Major Project Report

On

Ocular Disease Recognition With Ensemble Techniques

Submitted by

Pratik Hublikar

&

Shubhankar Joshi

Under the Guidance of

Prof. Dr. Archana Chaudhari

In the partial fulfillment for the Degree of

Bachelor of Technology

In

Instrumentation & Control

Department of Instrumentation and Control

Bansilal Ramnath Agarwal Charitable Trust's

Vishwakarma Institute of Technology, Pune-37

(An Autonomous Institute affiliated to Savitribai Phule Pune University)

Academic Year 2022-23

Certificate

This is to certify that the Project entitled *Ocular Disease Recognition With Ensemble Techniques* submitted by

1. Shubhankar Joshi
2. Pratik Hublikar

Is a record of bonafide work carried out by above students under our supervision in their partial fulfillment for the award of the Degree of Bachelor of Technology (B.Tech) in Instrumentation and Control at Vishwakarma Institute of Technology, Pune.

Dr. Archana Chaudhari
College Guide
Department of Instrumentation
Vishwakarma Institute of Technology Pune

Prof. Manisha Mehtre
Major Project Coordinator
Department of Instrumentation
Vishwakarma Institute of Technology Pune

Dr. Shilpa Sondkar
Head of Department
Department of Instrumentation
Vishwakarma Institute of Technology

Acknowledgement

We express our sincere gratitude to Dr. R. M. Jalnekar, Director VIT, Pune, Dr. Shilpa Sondkar, Head, Dept. of Instrumentation and Control, VIT, Pune, Prof. Dr Manisha Mhetre, Project Coordinator, Prof. Dr. Archana Chaudhari, Project Guide.

All the dignitaries have been of great assistance (directly and indirectly) during our Major Project. Their guidance and encouragement have made this project a good experience.

Pratik Hublikar

Shubhankar Joshi

Index

List Of Images

List Of Tables

Abstract

Retinal pathologies are the most common cause of childhood blindness worldwide. Rapid and automatic detection of diseases is critical and urgent in reducing the ophthalmologist's workload. Ophthalmologists diagnose diseases based on pattern recognition through direct or indirect visualization of the eye and its surrounding structures. Dependence on the fundus of the eye and its analysis make the field of ophthalmology perfectly suited to benefit from deep learning algorithms. Each disease has different stages of severity that can be deduced by verifying the existence of specific lesions and each lesion is characterized by certain morphological features where several lesions of different pathologies have similar characteristics. We note that patients may be simultaneously affected by various pathologies, and consequently, the detection of eye diseases has a multi-label classification with a complex resolution principle. The solutions we presented here are three major Deep learning models. First model is a binary classification model which classifies into two classes 'Disease' and ‘Normal’. Second model is a multi-class classification model which classifies the images into different eye diseases. Third model is a multi-class classification model which classifies into different stages of Diabetes Retinopathy. The basic methodology while building these models is as follows: First, we study the different characteristics of lesions and define the fundamental steps of data processing. We then identify the different libraries and modules needed to execute deep learning solutions. Finally, we investigate the principles of experimentation involved in evaluating the various methods, the public database used for the training and validation phases, and report the final detection accuracy with other important metrics.

Keywords: Image classification, Deep learning, Retinography, Convolutional neural networks, Eye diseases, Medical imaging analysis.

Chapter 1

Introduction

The retina is a layer of tissue at the back of the eye that perceives incoming light and sends it out images in our brain. In its center there is a tissue where the macula is located, which provides sharp vision that helps us with tasks as necessary as driving and reading. This valuable tissue can be affected by different disorders or diseases that can affect vision. Today, retinal pathologies are already the most common cause of childhood blindness all over the world. As countries become richer and per capita income rises, the prevalence of blindness decreases and the causes that produce blindness change. In the poorest nations in the world, the leading cause of blindness is cataract. In a country with an average gross domestic product as in India, the leading cause of blindness is glaucoma and diabetic retinopathy. Due to economic improvement, cataract surgery is mostly accessible and its incidence is lower. In countries with a high GDP, glaucoma and cataracts are still very common and important pathologies, but blindness is due to other diseases of the retina such as diabetic retinopathy which can be prevented and is treatable in their first stages (Gilbert, C. 2001)[1].

Diabetes is a growing problem in developing countries. In India, it is estimated that 8 to 10% of the population is diabetic and its prevalence is growing. Although the studies based on the population suggest that diabetic retinopathy is not a major cause of blindness in India at present, it is likely to be in the future. In 2010, around 10 million cataract operations per year worldwide and by 2020, expected to exceed 30 millions. Almost all of the growth has occurred in developing countries. More surgery from Cataract leads to complications in the posterior segment of the eye, such as the detachment of retina .These complications are very treatable, as long as a retinal surgeon is present, qualified and well equipped. Considering the observable trend, it is likely that retinal diseases are already an important and growing problem in all parts of the world (Yorston, D. 2003)[2].

It is true that there are many retinal degenerations for which there is no cure. Despite this, patients can benefit greatly from receiving an accurate diagnosis, with an explanation detailed and a clear prognosis (often a disease affects an eye and it is possible to prevent the other). For disease detection in developed countries, ophthalmologists use one standard medical imaging tool called "fundus photography or retinography". Through of a quick and simple procedure, the doctor or specialist, is able to obtain a photograph in high-quality color of the fundus of the eye where its morphology and structures (nerve optic, blood vessels, macula, retina, etc.) as shown in Figure 1, providing a source important information about the patient's health (Bonet, 2018)[3] (Saine and Tyler, 2002)[4].

The fundus image in Figure 1 shows the macula in the center of the image, the optic disc located towards the side of the nose, the arteries and the different veins.

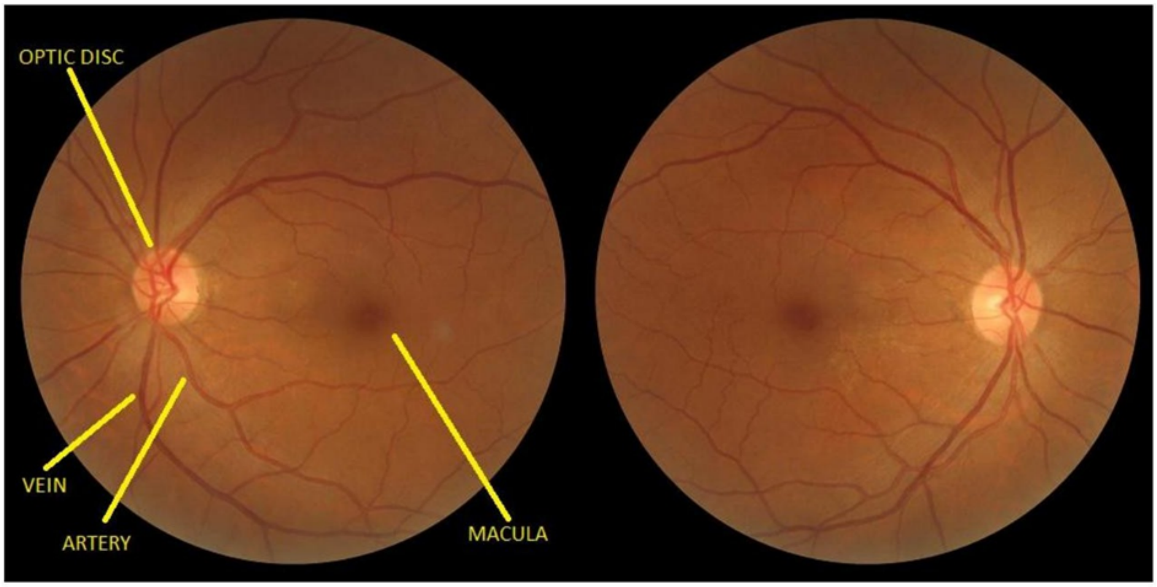


Figure 1. Left and right eye retinography (seen from the front) without any abnormalities.

Due to advances in technology, equipment to treat retinal diseases, although still expensive, are now much more suitable for use in developing countries. However, the shortage of qualified personnel continues to be an important limitation facing future challenges. This means we need more trained ophthalmologists subspecialized in retinal diseases. (Ophthalmology, 2019)[5]. The procedure for obtaining the images is done using a retinal camera as shown in Figure 2. To be able to capture the fundus of the eye, the doctor applies a few drops to the eye to dilate the pupil and wait a few minutes while they take effect. At the time of the exam, the patient must look at the camera, and the specialist can capture the fundus to obtain images to be analyzed later (Garcia, 2010)[6].

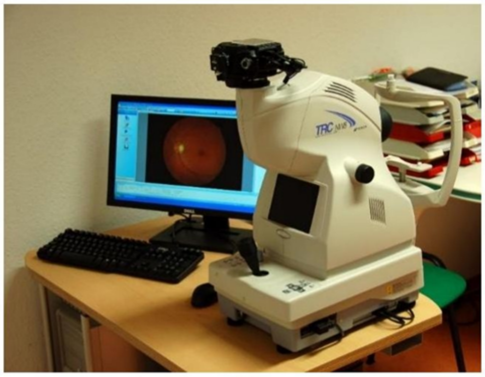


Figure 2. Retinal camera (Roletschek, 2019)

The procedure, which can be performed annually, provides an effective, safe and economical way to avoid blindness as soon as possible caused by diseases such as: diabetes, glaucoma, cataract and macular degeneration among others. The fundus examination has been a key aspect in the diagnosis of eye diseases, and fundamentally, in the early diagnosis of general diseases that manifest in the retina. The retina is an extension of the brain and everything that affects it will manifest itself in it, such as neurodegenerative or vascular diseases (Ophthalmology, 2019)[5]. As fundamental advantages offered by the procedure, we can consider the ease of obtaining and the rapid diagnosis of diseases and thus be able to rule out the appearance of injuries and prevent its progression. As the main disadvantage we can mention that mydriasis (which occurs after putting the drops to dilate the eye) lasts about 4 hours, which limits activity of the patient until the eye returns to its normal state. The following Figure 3 groups together a set of common pathologies that can be detected by the fundus test (Ophthalmological, 2019)[5]:

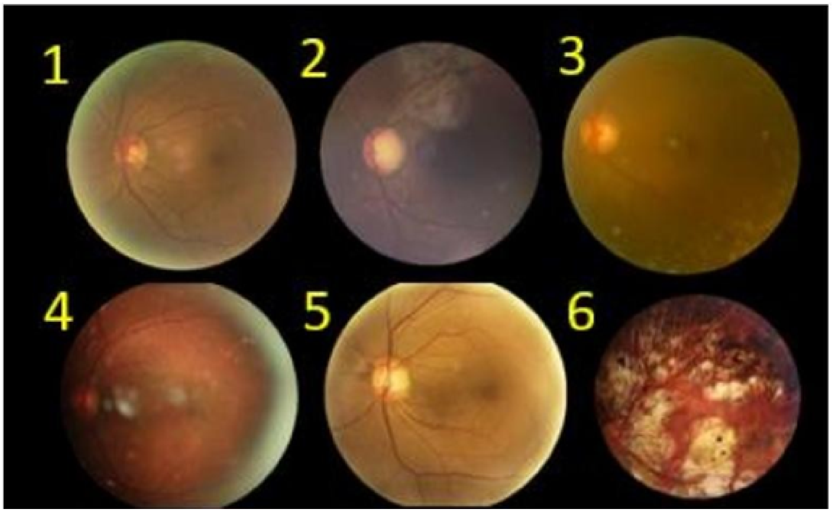


Figure 3. (1) Diabetic retinopathy, (2) Glaucoma, (3) Cataract, (4) Degeneration of the macula, (5) Hypertension, (6) Myopia,

Some of the pathologies contained in Figure 3 can cause blindness if the disease progresses or becomes complicated. People with diabetes and the elderly have more chances of developing diabetic retinopathy (DR) and if it is not controlled in time, it can lead to blindness. Glaucoma is a disease that causes progressive damage to the optic nerve and a constant increase in intraocular pressure. Glaucoma has no definitive cure, however, through surgery its consequences can be greatly alleviated. This one pathology is one of the main causes of vision loss in the world.

Cataract affects older adults and causes a decrease in vision due to the physiological change inside the eye where a series of patches develop that make it difficult vision. The macula is responsible for central vision and vision is distorted if fluids get accumulated there. Macular degeneration (AMD) usually occurs in people over the age of seventy and there are usually no symptoms during the early stages of the disease, which makes the fundus an important process in the early detection of eye diseases.

Hypertension is a silent disease, but it is capable of affecting the entire organ causing its destruction over time. This pathology is capable of changing the morphological structures of the blood vessels, such as, for example, changes in diameter, alteration of the tortuosity, and can produce cerebral vascular diseases, cerebrovascular accidents and heart attacks.

Myopia is an eye disorder that causes a significant loss of vision, a cause of progressive retinal epithelial pigment thinning and attenuation. Mainly it alters the vision of distant objects making them blurry and causes alterations that can be seen in the fundus obviously given that the optic nerve is usually inclined and shows a surface sclera as shown in Figure 3 (Garrido, 2011)[7].

Digital fundus cameras save the images directly to the computer and then they are evaluated by specialists for the generation of a diagnosis. The current standard for the disease classification from fundus photographs, includes manual estimation of the locations of the injuries and the analysis of their degree of severity, which require a lot of time by the ophthalmologist, also incurring high costs in the health system. Therefore, it would be important to have automatic methods to do the analysis. The generation of a computer-aided diagnosis provides us with an interesting challenge of medical imaging for the automatic detection of eye diseases from the images of in some situations, for the detection of anomalies through the evaluation of lesions and landmarks anatomical features of the fundus image, and which present a reliable analysis of the results according to its context. (CI Sánchez et al. 2011)[8] (CI Sánchez et al. 2012)[9].

According to the paper published by Zhou et al, (2019)[10], there are two important research areas in the treatment of medical images of the fundus of the eye. The first details the classification of diseases according to their degree of severity and the second on the classification based on the segmentation of lesions through the analysis of the lowest level features, i.e. analyzing the pixels of the image. Both areas can be seen as a generic classification problem. A classification, where nowadays deep learning methods are being applied, both in Ophthalmology and in other fields of medicine, which have had a great advance. Ophthalmologists diagnose diseases based on the recognition of patterns through the direct or indirect visualization of the eye and its surrounding structures. Diagnostic technologies provide an information accompaniment that helps the doctor with their decision-making. This dependency with the fundus image and its analysis makes the field of ophthalmology perfectly suitable to benefit from deep learning algorithms. The incorporation of algorithms of deep learning is starting to be implemented in different areas of ophthalmology and can potentially change the type of work performed by ophthalmologists (MD Abramoff, Y. Lou, A. Erginay, et al 2016)[11] (Parampal S, et al 2018)[12] (A. Lee et al 2017)[13]

Deep learning is being applied to fundus photographs, optical coherence tomography and other visual fields, achieving robust classification performance in the detection of diabetic retinopathy, retinopathy of prematurity, glaucoma, macular edema and age related macular degeneration. Deep learning on eye images can be used in conjunction with telemedicine as a possible solution to select, diagnose and control eye diseases for patients in primary care (Ting DSW, Pasquale LR, Peng L, et al 2019)[14]. More specifically, through machine learning schemes that use convolution-based neural networks (CNNs). The networks use a set of filters of image processing to extract various types of features that the network considers the existence of pathological signs. Feature extraction happens after your learning about a training set and that is an integral part of the methods of pattern classification. Deep learning, then, can be seen as a brute force algorithm which is able to determine the most appropriate filters and image processing tools, which can quantify various characteristics of different diseases (Hijazi et al., 2015)[15]. CNNs have become the standard image classification method using machine learning.

The solutions we presented here are three major Deep learning models. First model is a binary classification model which classifies into two classes 'Disease' and ‘Normal’. Second model is a multi-class classification model which classifies the images into different eye diseases. Third model is a multi-class classification model which classifies into different stages of Diabetes Retinopathy. The basic methodology while building these models is as follows: First, we study the different characteristics of lesions and define the fundamental steps of data processing. We then identify the different libraries and modules needed to execute deep learning solutions. Finally, we investigate the principles of experimentation involved in evaluating the various methods, the public database used for the training and validation phases, and report the final detection accuracy with other important metrics. The models are developed in Pytorch in Python with Transfer Learning and are trained on single cloud GPU’s. Pytorch is one of the open source software library for numerical computation using flow charts of data and is able to deploy computations on one or more CPUs/GPUs with a single API

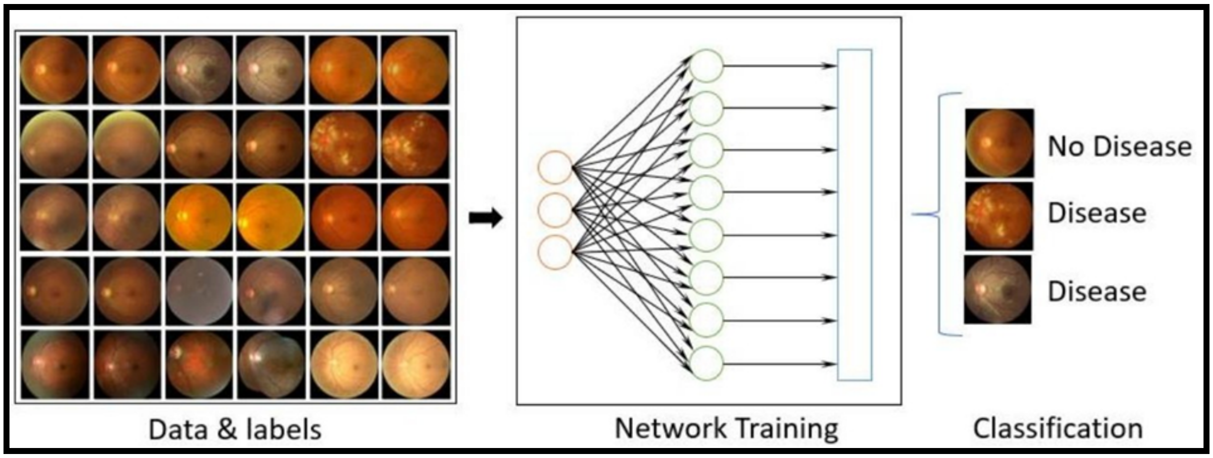


Figure 4. Classification of eye diseases using CNNs.

Chapter 2

Literature Review

Deep Learning for Ocular Disease Recognition: An Inner-Class Balance [16]

Description: In this paper they have presented an ocular disease detection system using deep learning algorithms. The ODIR database has been used which has eight categories of ocular disease with around 5000 values. The authors found out that the data was unstable so they divided the data into two equal classes 'Diseased ' and ‘Normal’.These classes were then trained using pre pre-trained VGG-19 model.
VGG-19 is a CNN-based model that uses 3 × 3 filters with a single stride and always employs the same padding and max pooling layers of 2 × 2 filters with a stride of 2, instead of having a huge number of hyperparameters. In the architecture, the convolution and max pooling layers were organized in a similar manner. They used two FC layers in the model.The model was configured to use the Adam optimizer and a binary cross-entropy loss function. Additionally, the sigmoid activation function was also used.

Results: They achieved the highest accuracy for normal versus myopia, which is 98.10%, and also got a 94.03% accuracy for the normal versus cataract class. Furthermore, they got a 90.94% accuracy rate for the normal versus glaucoma class

A Survey of Convolutional Neural Networks: Analysis, Applications and Prospects [17]

Description: The article aims to provide some novel ideas and prospects in the fast-growing field of CNNs. The types of convolution operations are discussed along with a brief history, applications of one dimensional, two dimensional and multidimensional CNNs in time series prediction, signal identification, image classification, object detection, image segmentation and face recognition, human action recognition and object recognition, and experimental results for various CNN models such as VGGNet, R-CNN, YOLO, SSD, etc. The hardware implementation of CNNs is also discussed which encompasses the information about NVIDIA’s premium GPU architectures (Turing and Ampere) built for training and inferencing of Deep Neural Networks with efficient acceleration libraries like cuBLAS and cuDNN. Implementation of deep neural networks on field programmable gate arrays (FPGA) is also discussed. To conclude, the article states that even though CNNs possess many benefits and have been widely used, they have the potential to be refined further in terms of model size, security and NAS (Network Architecture Search).

EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks [18]

Description: This paper introduces the EfficientNet architecture of convolutional neural networks which is a scaling method that uniformly scales all dimensions of depth, width, resolution using a compound coefficient. FOr example, if 2^N times more computational resources are required, then the network depth can be simple increased by α^N, width by β^N and image size by γ^N, were α, β and γ are constant coefficients determined by small grid search on the original small model. The compound scaling method discussed in the paper is justified by the intuition that if the input image is bigger, then the network needs more layers to increase the receptive field and more channels to capture more fine grained patterns on bigger images. The base EfficientNet-B0 network is based on the inverted bottleneck residual blocks of MobileNetV2, in addition to squeeze-and-excitation blocks.

Results: It was observed that EfficentNets transfer well and achieve state-of-the-art accuracy on CIFAR-100 dataset (91.7%), Flowers dataset (98.8%), and three other transfer learning datasets with an order of magnitude of fewer parameters.

Detection of Hard Exudates in Retinal Fundus Images Using Deep Learning [19]

Description: In this paper they have proposed a deep learning algorithm to detect the hard exudates in fundus images of retina. Hard exudates develop when Diabetic Retinopathy is present. They have used the IDRiD dataset which has 54 fundus images.They created respective ground truth images having intensity 1 for exudate pixels and 0 for remaining pixels.Training set is of 40 images and testing set is of 14 images. They created 200000 image patches of 32x32.From a single image 2500 exudate patches and 2500 background patches were extracted.The network is trained on these 200000 images.

They have proposed 8 layered Convolutional neural network.The feature map size is halved using maxpool operation after every two convolutional layers, but, at the last convolution layer the feature map size remained same. Training images were divided into 5 sets of 40000 images and each set is trained for 500 epochs one after another.The network is trained for 3 complete streaks, which is 1500 epochs.

Results: The model has predicted all the test image patches with an accuracy of 98.6%.

Detection of Age related Macular Degeneration via Deep Learning. [20]

Description: This paper studies the efficacy of Deep Convolutional Neural Networks for the detection Age related Macular Degeneration. In this paper they have used the NIH AREDS dataset which has 5600 images. The images were assigned in 4 categories with category 1 corresponding to no evidence of AMD, category 2 corresponding to early stage AMD, category 3 corresponding to intermediate stage AMD, and category 4 corresponding to one of the advanced forms of AMD.

For the feature selection they used the OF DCNN (Over Feat) pre-trained model. Features were calculated by tapping into the OF network layer 19. From their observation they found that the features near the center of the macula tend to be more important for classifying the severity of AMD. To account for that fact, their approach uses different concentric square image grid areas (windows) within the fundus image and concatenates the results into a single feature vector. The final output feature vector is used as input to a linear support vector machine for purposes of classification.

Their main aim was to identify the people having intermediate AMD i.e. category 3. Therefore they focused on a set of two class problems where they test early vs intermediate i.e {1 & 2} vs. {3 & 4}; {1 & 2} vs. {3}; {1} vs. {3}; and {1} vs. {3 & 4}.

Results: They got varied accuracy between 92% to 95% for the above 4 category tests.The results show that pre-trained DCNN features trained on general purpose images, effectively transfer to, and can be used for training AMD severity LSVM classifiers, with good ensuing performance

Very Deep Convolutional Neural Networks for Large-Scale Image Recognition [21]

Description: The article introduces VGG16 and VGG19 models, with 16 and 19 layers respectively, which are high accuracy CNN models used for a variety of image classification tasks. Details regarding the architecture of the model, their specific configurations, their training and testing methods and their implementation details such as the GPU configurations are also highlighted. The basis for the classification framework included fixing the training image size to 224*224 pixels, which corresponded to single scale training. For experimentation, the ILSVRC-2012 dataset was used, which includes 1000 classes split into training (1.3M images), validation (50K images), and testing (100K images with held-out class labels). Classification performance of the models were evaluated with the top-1 and top-5 errors, where the former is multi-class classification error and the latter is computed as the proportion of images such that the ground truth category is outside the top-5 predicted categories. To improve the performance of the models a ConvNet fusion was implemented that gave significantly better results. When comparing with the state-of-the-art it was seen that very deep convolutional networks (up to 19 layers) provided the best results.

Results: The model was evaluated using single scale evaluation, multi scale evaluation, multi-crop evaluation and a ConvNet fusion. For single scale evaluation, it was observed that the classification error decreases with the increased ConvNet depth: from 11 layers to 19 layers. The performance of the model significantly improved when outputs of several models were combined by averaging their soft-max class posteriors.

Rethinking the Inception Architecture for Computer Vision [22]

Description: The authors of the paper introduce the InceptionV3 model architecture of convolutional neural networks used widely for image classification. InceptionV3 is a superior version of the previous InceptionV1 which was introduced as GoogLeNet. The InceptionV3 model has a total of 42 layers and a much lower error rate than its predecessors. The major architecture modifications made for InceptionV3 include: factorization into smaller convolutions where convolutions of size 5x5 were reduced to 3x3 which reduced the number of parameters to large extent, spatial factorization into asymmetric convolutions which further improves the operation speed by converting an nxn convolution matrix into an nx1 and a 1xn matrix, utility of auxiliary classifiers which improve convergence of very deep neural network and is mainly used to tackle the problem of vanishing gradient in very deep neural networks, and efficient grid size reduction, which was accomplished by expanding the activation dimension of the network filters. Even though the InceptionV3 model has 42 layers, which is a bit higher than the previous InceptionV1 and InceptionV2, the efficiency of the model is quite remarkable.

Results: The InceptionV3 model performed significantly better than VGGNet, GoogLeNet, PReLU with a top-1 error of 17.2% and a top-5 error of 3.58%.

Ocular Disease Recognition [23]

Description: In this paper a deep learning model is being created with a combination of CNN architectures. The model is capable enough to identify six eye diseases. The ODIR database has been used which has eight categories of ocular disease with around 5000 values:(Diabetes, Glaucoma, Cataract, Related Macular Degeneration, Myopia, Hypertensive Retinopathy)

At first they implemented the LeNet5 CNN architecture which consists of 3 convolutional layers, 2 subsampling layers, 2 fully connected layers and 1 flatten layer. All layers use tanh activation function, except one fully connected layer which uses softmax.

Later,they implemented the AlexNet30 model which has 3 convolutional layers, 3 max-pooling layers, 2 normalization layers, 2 fully connected layers, and 1 softmax layer.

They noticed both models gave better for specific diseases, For example AlexNet performs better in recognition of myopia, cataract and a normal eye. So they decided to make a custom model with a combination of these two models.
Results: The combination of two models gave an accuracy 94% with the test set.

Deep Learning [24]

Description: The authors began by providing a general overview of machine learning systems, which are used to recognize objects, transcribe speech to text, and match news items to user interests. They went on to discuss the advantages of Deep Learning over traditional machine learning methods. Deep Learning algorithms can automatically learn intricate structures in high-dimensional data, whereas traditional machine learning methods rely heavily on domain experts like engineers to manually extract features. The authors stated that deep learning is making significant progress in solving problems that have eluded the artificial intelligence community for many years, and they believe that deep learning will have many more successes in the future because it requires very little manual engineering. The paper provided a detailed explanation of the backpropagation algorithm. Working backwards, the backpropagation algorithm calculates and propagates the derivatives or gradients of an objective function from the output of the neurons in the network. The gradients propagated backwards are critical for Deep Learning algorithms to learn the data representation. It was explained how gradients are used to adjust the weights, using the analogy of weights as knobs that serve as the algorithm's input–output function and a hilly landscape in the high-dimensional space of weight values that represents the objective function when averaged across all training examples. The paper then discussed the various Deep Learning models that are currently in use, such as Convolutional Neural Networks (ConvNets), which are commonly used in the fields of language processing and computer vision. The authors provided detailed insights into ConvNets, for example, how it uses natural properties of compositional hierarchies, local connections, shared weights, and pooling. The paper also discusses other Deep Learning models besides ConvNets, such as Recurrent Neural Networks (RNNs) and the Long Short Term Memory (LSTM) model.

Results: The authors predict and express optimism about the future of Deep Learning research. According to their assessment, unsupervised learning will be far more important in the long run than supervised learning. End-to-end trained systems that combine ConvNets with RNNs that use reinforcement learning will lead to advances in vision and natural language understanding, and deep learning will have a significant impact on language understanding in the coming years.

Ocular Diseases Diagnosis in Fundus Images using a Deep Learning: Approaches, tools and Performance evaluation [25]

Description:

This is a survey of ocular pathology detection methods based on deep learning. First, they studied the existing methods either for lesion segmentation or pathology classification. Pre-processing of images was the first step in all projects. However, the preprocessing objective varied with respect to the methods. Some methods proposed processing to enhance fundus image quality. In some cases, the original images are converted from RGB to Hue Saturation Intensity color space, and then denoised using median filter

The glaucoma can be directly detected through a method based on 18-layers CNN.The diabetic retinopathy lesions exudates, hemorrhages, micro aneurysms are able to segment through a 10-layer CNN. AlexNet, VggNet, GoogleNet and ResNet are some models used.

Chapter 3

Introduction to Ensemble Techniques

3.1 Introduction

Ensemble techniques in machine learning refer to the process of combining multiple models to improve the accuracy and robustness of predictions. Ensemble methods can be used with a variety of machine learning algorithms, including decision trees, random forests, and neural networks. The general idea behind ensemble techniques is that by combining the predictions of multiple models, we can reduce the risk of errors or biases that may be present in any one individual model.

There are two main types of ensemble learning:

Bagging: This involves training multiple instances of the same type of model on different subsets of the training data. Each model produces a prediction, and the final prediction is the average of all the predictions. Bagging helps to reduce overfitting and improve the accuracy of the model.

Boosting: This involves training a series of weak models on the same dataset. Each model is trained on the errors made by the previous model, and the final prediction is a weighted average of all the models. Boosting helps to improve the accuracy of the model by focusing on the areas where the previous models made errors.

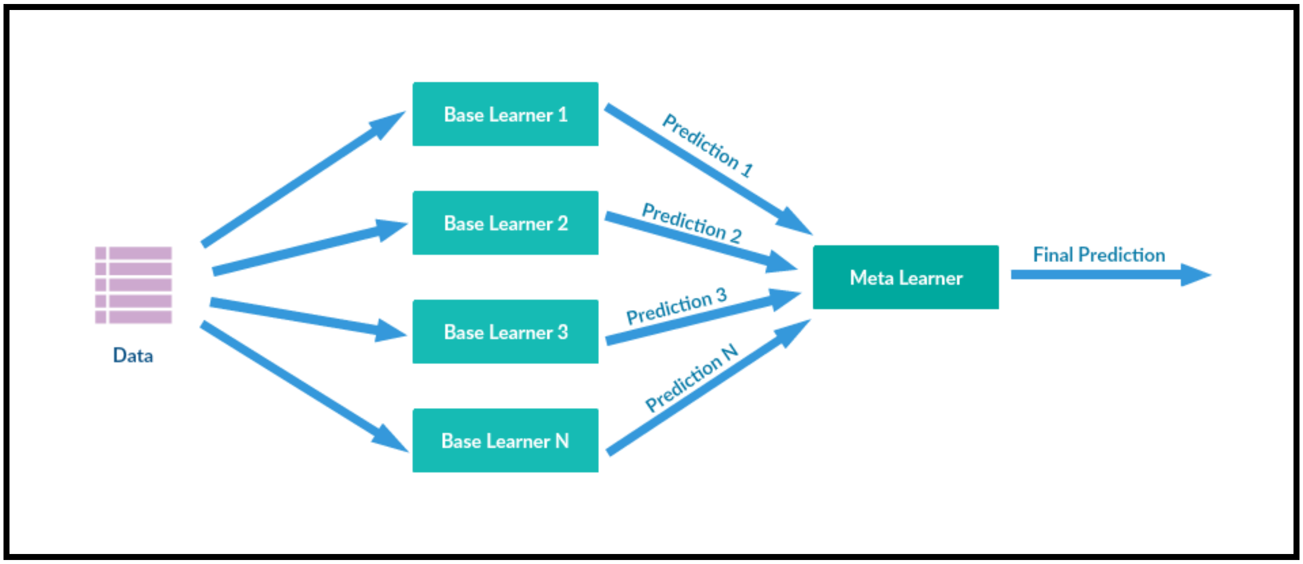


Fig 5 Block diagram of Ensemble Learning

3.2 Model Architectures

Let’s have a look at the architecture of the models used with Transfer Learning:

A. ResNet50 -

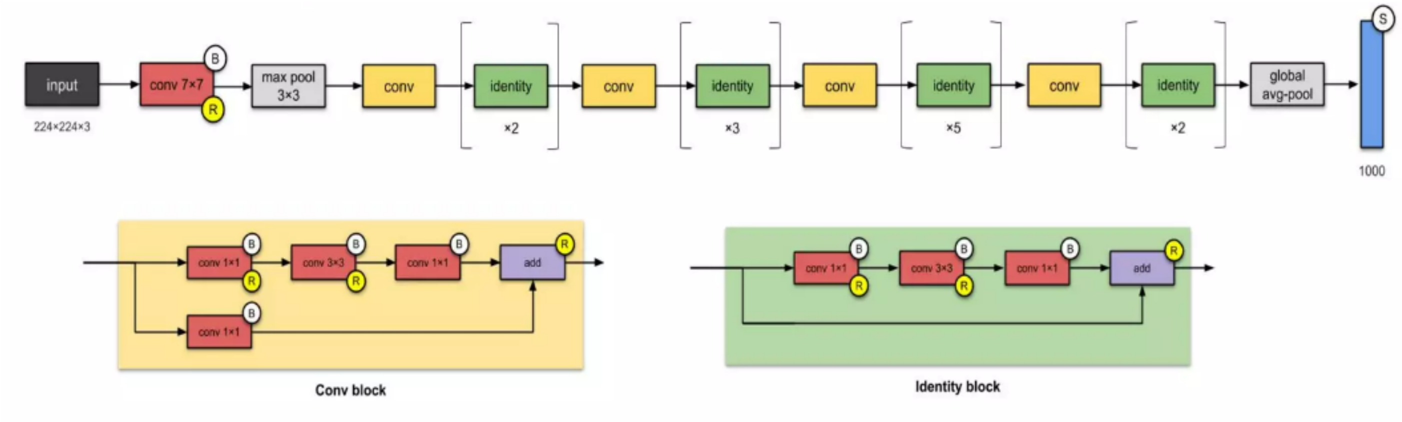


Fig 6 ResNet Architecture

ResNet50 is a convolutional neural network architecture consisting of 50 layers developed by Microsoft researchers for image classification tasks. It features residual connections, which allow information to be passed directly from one layer to another without being transformed, reducing the vanishing gradient problem and improving training convergence. The architecture is composed of convolutional layers, max pooling layers, fully connected layers, and a softmax layer for classification. It is trained using the cross-entropy loss function and stochastic gradient descent optimization.

The ResNet50 architecture consists of several blocks, each composed of multiple layers. Here is a block-wise description of the ResNet50 architecture:

Input layer: The input is an RGB image of size 224x224.

Convolutional block: The first block consists of a convolutional layer followed by a batch normalization layer and a ReLU activation function. This is repeated three times, each time with a different number of filters.

Identity blocks: The next three blocks are identical and each contains three convolutional layers with batch normalization and ReLU activation. The first layer has a filter size of 64, while the second and third layers have a filter size of 64 or 256, depending on whether the block is the first one in the sequence.

Projection block: This block is used to increase the dimensionality of the feature maps. It consists of a convolutional layer with a stride of 2, followed by batch normalization and ReLU activation. Then, there are two identity blocks with filter sizes of 128 or 512.

Projection blocks: Two more projection blocks are added, each with a stride of 2 and increasing the feature maps' dimensionality. They are composed of a convolutional layer with batch normalization and ReLU activation, followed by three identity blocks with filter sizes of 256 or 1024.

Projection blocks: The final two projection blocks are similar to the previous two, but with filter sizes of 512 or 2048.

Average pooling: The output of the final projection block is fed into an average pooling layer of size 7x7.

Output layer: The output of the average pooling layer is then flattened and fed into a fully connected layer with 1000 units, followed by a softmax layer for classification.
The ResNet50 architecture has a total of 50 layers, including convolutional, batch normalization, activation, and fully connected layers.

B. EfficientNet B6 -

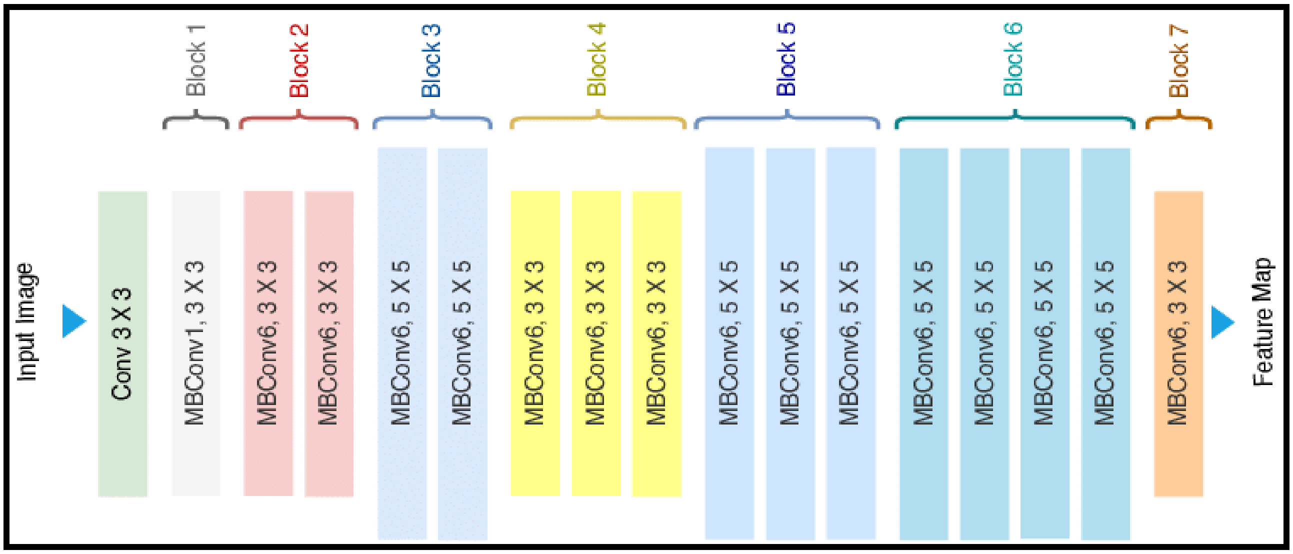


Fig 7 EfficientNet Architecture

EfficientNet is a family of convolutional neural network architectures developed by Google in 2019, that achieved state-of-the-art results on image classification tasks while being more efficient in terms of computation and memory usage than other popular models like ResNet or Inception.

The EfficientNet architecture uses a combination of scaling methods to improve performance and efficiency. It includes compound scaling, which scales the depth, width, and resolution of the network together, rather than independently. This approach allows the model to achieve better performance while using fewer parameters and less computational resources.

The architecture is composed of several building blocks, including convolutional layers, batch normalization, activation functions, and pooling layers. It also includes a novel block called the "MBConv block," which stands for Mobile Inverted Residual Bottleneck Convolution. This block uses depthwise separable convolution, which reduces the number of parameters and computation required by traditional convolutional layers, and inverted residual connections, which allows for better information flow between layers.

The EfficientNet family includes several models with varying levels of complexity and performance, from EfficientNet-B0, which is a small and efficient model, to EfficientNet-B7, which is a larger and more powerful model. These models have achieved state-of-the-art performance on several image classification tasks, including the ImageNet dataset, while using fewer parameters and less computation than other popular models.

The EfficientNetB6 architecture consists of several blocks, each composed of multiple layers. Here is a block-wise description of the EfficientNetB6 architecture:

Input Layer : The input image is passed through a series of convolutional and pooling layers to reduce its resolution and increase the number of channels.

Blocks: The network consists of 28 blocks, with each block consisting of several layers. Each block is composed of a combination of inverted bottleneck blocks and linear bottleneck blocks.

Inverted bottleneck blocks: The inverted bottleneck blocks consist of a 1x1 convolution layer that expands the number of channels, followed by a depthwise convolution layer that performs spatial filtering, and a 1x1 convolution layer that compresses the number of channels. The output of this block is added to the input of the block to create a residual connection.

Linear bottleneck blocks: The linear bottleneck blocks are similar to the inverted bottleneck blocks, except that they do not have the first 1x1 convolution layer for channel expansion.

Head: The final output of the last block is passed through a global average pooling layer to generate a feature vector, which is then passed through a fully connected layer to produce the final
Model Size: EfficientNet-B6 has approximately 84 million parameters and its input size is 528x528.

C. ResNext50 -

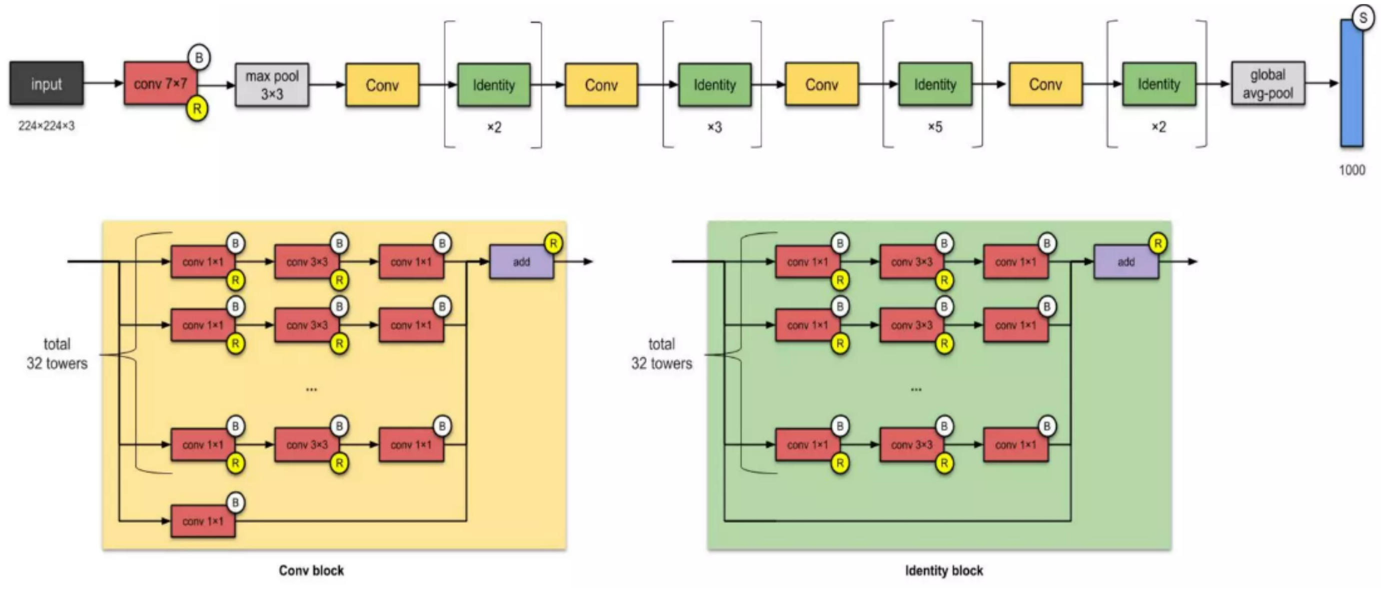


Fig 8 ResNext50 Architecture

ResNext is a neural network architecture that builds upon the ResNet (Residual Network) model by introducing a novel module called a "cardinality" module. This module increases the network's capacity to learn by allowing it to learn multiple different representations of the same feature map. The ResNext model consists of a series of ResNet-like blocks with multiple parallel branches. Each branch is responsible for learning a specific subset of features, and the cardinality module allows the model to aggregate these features in a way that improves overall performance. The cardinality module uses a split-transform-merge strategy to create a multi-branch structure. In the split stage, the input feature map is divided into multiple smaller feature maps, each of which is processed by a separate branch. In the transform stage, each branch performs its own set of operations on its corresponding feature map. Finally, in the merge stage, the feature maps from all branches are concatenated to form a single output feature map. Overall, the ResNext architecture achieves state-of-the-art performance on a wide range of computer vision tasks, including image classification, object detection, and semantic segmentation. Its ability to efficiently learn multiple representations of the same feature map makes it well-suited for tasks that require fine-grained classification or detection.

D. EfficientNet V2 S -

EfficientNet V2 S is a convolutional neural network architecture that was introduced in a paper titled "Scaling EfficientNet to Even Smaller Models" by Tan et al. in 2021. This model is designed to be a smaller and more efficient version of the original EfficientNet architecture, while still maintaining a high level of accuracy in image classification tasks. The "S" in the model name stands for "Small" and refers to the fact that it is the smallest variant in the EfficientNet V2 family. EfficientNet V2 S has a total of 13 layers and is composed of a stem, six intermediate blocks, and a classification head. The stem consists of a series of convolutional layers that extract features from the input image. The intermediate blocks are composed of depth wise-separable convolutions, which allow for efficient use of computational resources while still maintaining a high level of accuracy. The final classification head consists of a global average pooling layer followed by a fully connected layer. EfficientNet V2 S has 66 million parameters and achieves state-of-the-art performance on a number of benchmark datasets, including ImageNet and CIFAR-10. It is designed to be efficient enough to run on devices with limited computational resources, such as mobile phones or embedded systems, while still delivering high accuracy in image classification tasks.

EfficientNet V2 S is a deep neural network architecture that is designed to be highly efficient and accurate for various computer vision tasks. The model is an improved version of the original EfficientNet architecture with significant improvements in both accuracy and efficiency. EfficientNet V2 S is composed of a series of blocks, each of which is responsible for performing a specific operation. These blocks are arranged in a hierarchical manner, with the output of one block being fed as input to the next block in the sequence. Here is a detailed description of the EfficientNet V2 S architecture, block by block:

Stem Block: The stem block serves as the input layer of the EfficientNet V2 S model. It consists of a series of convolutional layers, batch normalization, and activation functions. The stem block is responsible for processing the raw input image and extracting low-level features such as edges and corners.

Inverted Residual Block: The inverted residual block is a building block of the EfficientNet V2 S model. It consists of three major components: a depthwise convolution, a pointwise convolution, and a skip connection. The depthwise convolution is responsible for performing spatial convolutions while the pointwise convolution is responsible for performing channel-wise convolutions. The skip connection allows for the flow of information from one layer to another without modification, which helps preserve important features.

Squeeze-and-Excitation Block: The Squeeze-and-Excitation (SE) block is a module that is inserted after the inverted residual block. It is designed to help the model learn to focus on important features and suppress less important features. The SE block consists of two major components: a squeeze operation and an excitation operation. The squeeze operation reduces the number of channels in the feature map, while the excitation operation learns a set of weights to scale the channels based on their importance.

Stem Reduction Block: The stem reduction block is similar to the stem block but with a reduced number of channels. It is designed to reduce the dimensionality of the feature maps before they are passed to the next block.

Inverted Residual Block with SE: This block is similar to the inverted residual block, but with the addition of an SE block. The SE block helps the model learn to focus on important features and suppress less important features.

Feature Mix Layer: The feature mix layer is the final layer of the EfficientNet V2 S model. It takes the output of the previous block and performs a global average pooling operation, which produces a fixed-length vector of features. This vector is then passed through a fully connected layer to produce the final output.

In summary, the EfficientNet V2 S architecture is composed of six blocks: stem block, inverted residual block, SE block, stem reduction block, inverted residual block with SE, and feature mix layer. These blocks are designed to work together to extract meaningful features from the input image and produce accurate predictions for a variety of computer vision tasks.

Chapter 4

Proposed Methodology

4.1 Training Process

We built three Ensemble models using four models with transfer learning. The first Ensemble model is a binary classification model which has labels *‘Disease’ and ‘Normal’*. To build this ensemble model ResNet50, EfficientNet V2 S and ResNext50 were trained individually on the ODIR dataset. Ocular Disease Intelligent Recognition (ODIR) is a structured ophthalmic database of 5,000 patients with age, color fundus images of eyes having 8 types of diseases. All images of the diseased patients were clubbed together and labeled as ‘Disease’ in the preprocessing.

The second Ensemble model is a multi classification model which has labels *'Age_Macular_Degeneration', 'Cataract', 'Diabetic_Retinopathy', 'Glaucoma', 'Myopia', 'Normal'.* To build this ensemble model ResNet50, EfficientNet V2 S, EfficientNetB6 and ResNext50 were trained individually on the ODIR dataset.

The third Ensemble model is a multi classification model which has labels *‘Normal’, ‘Stage 1’ , ‘Stage 2’, ‘Stage 3’ and ‘Stage 4’.* To build this ensemble model ResNet50, EfficientNet V2 S, EfficientNetB6 and ResNext50 were trained individually on the Diabetic Retinopathy Balanced dataset. The dataset contains 35000 images having 7000 images per class in the training set and 1000 in the testing set. The dataset is a preprocessed dataset which has several data augmentations.

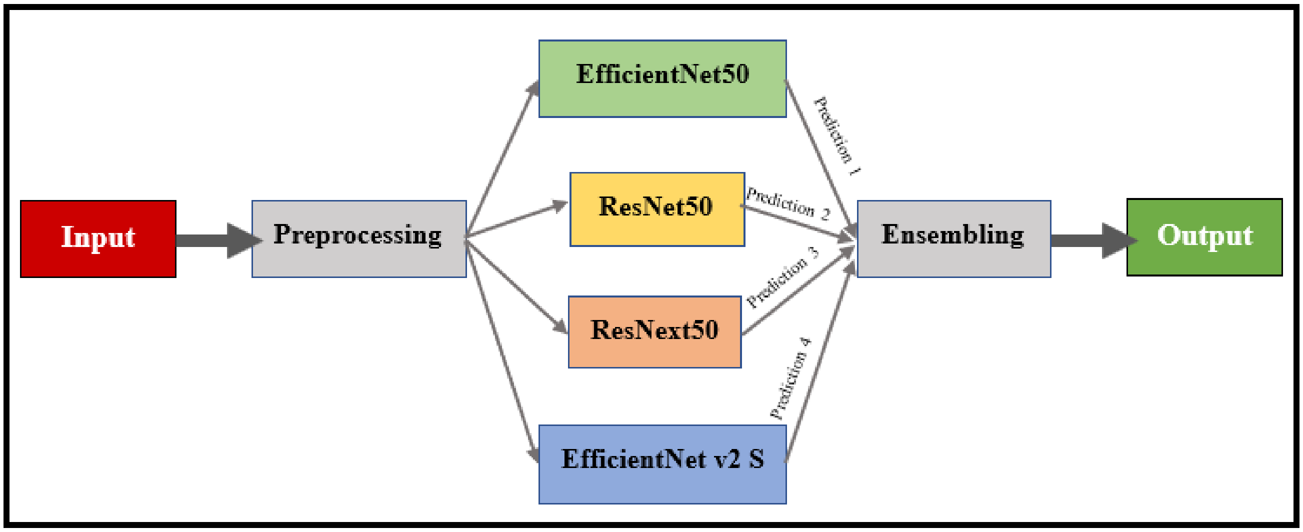


Fig 9. Training Process Block Diagram

All the above three Ensemble models were trained on a Nvidia T4, Nvidia P100 and Local Nvidia RTX 3060 laptop GPU. The frontend of these models was built with Gradio.

A. Databases

There are mainly two databases used in this project. The first one is the Ocular Disease Intelligent Recognition (ODIR) dataset. It It is a structured ophthalmic database of 5,000 patients with age, color fundus photographs from left and right eyes and doctors' diagnostic keywords from doctors.This dataset is meant to represent "real-life" set of patient information collected by Shanggong Medical Technology Co., Ltd. from different hospitals/medical centers in China. The patient data is classified into 8 labels known as -

1. Normal (N),
2. Diabetes (D),
3. Glaucoma (G),
4. Cataract (C),
5. Age related Macular Degeneration (A),
6. Hypertension (H),
7. Pathological Myopia (M),
8. Other diseases/abnormalities (O)

The second database is the Diabetic Retinopathy Balanced dataset.This database contains the images of different stages in Diabetic Retinopathy. This dataset contains 35000 images having 7000 images per class in the training set and 1000 in the testing set. This is a preprocessed dataset which has several data augmentation such horizontal flips, random rotations, vertical flips and grayscaling. The dataset is classified into 5 labels known as -

1. Normal
2. Stage 1
3. Stage 2
4. Stage 3
5. Stage 4

Both the above datasets are available on kaggle.com We have also combined some images from the *Preprocess eye fundus image dataset* which is also available on kaggle. All the images in the both dataset are gone through preprocessing functions.

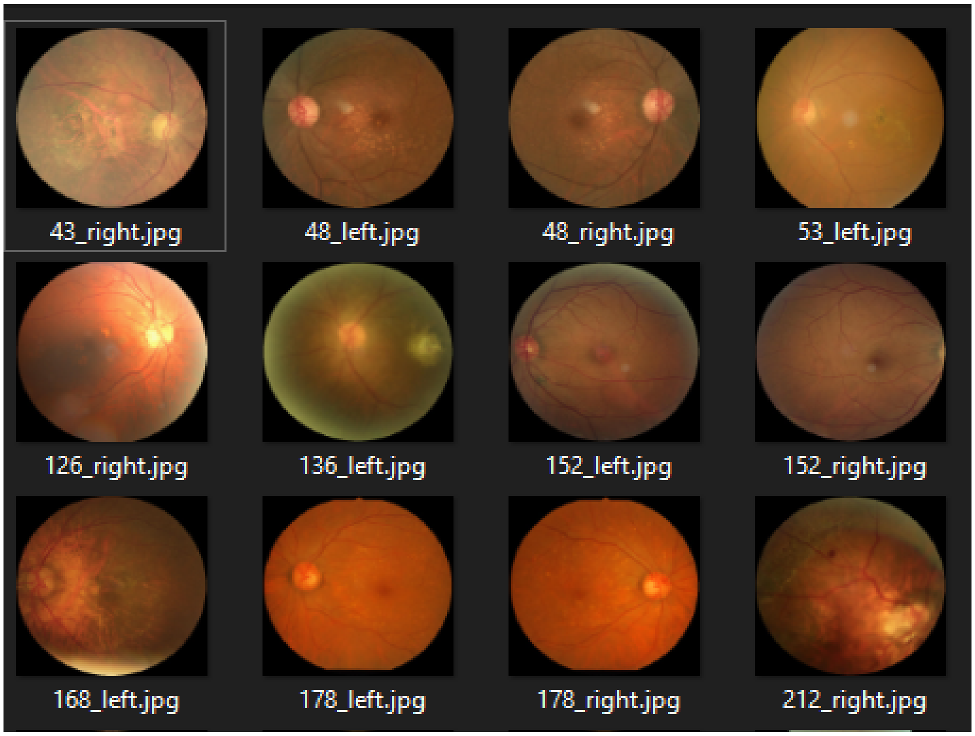


Fig 10 Dataset Image

B. Steps of Implementation

The following are steps followed in the training process.

STEP 1 Preprocessing

A preprocessing function was applied on the images in the ODIR dataset. In this function the images are converted from RGB into Lab. Lab is a conversion of the same information to a lightness component L*, and two color components - a* and b*. Lightness is kept separate from color, so that you can adjust one without affecting the other. In the function a Contrast-limited adaptive histogram equalization (CLAHE) filter was applied on the L* channel.

The preprocessed dataset was loaded with ImageFolder Class from pytorch into training testing and validation. (70,10,20). The Image Folders were converted into Dataloaders in Pytorch which convert data into an iterable for training loops.While loading the Dataloader we applied a specific transform on all the images by using the class *transforms* from *torchvision* library. In the transform we have resized the image into (224,224) and applied random rotations and flips. The batch size was chosen to be 8 for smooth computation. The loss function in the training was CrossEntropy function and the optimizer was Adam. The learning rate in most cases was 10^{-4} .

Algorithm for Preprocessing Function

1 Accept input path, output path, and clip limit as input parameters

2 Load the input image using cv2.imread() function and assign it to variable 'image'

3. Convert the input image from BGR to LAB color model using cv2.cvtColor() function and assign it variable 'image_lab'

4. Split the image into L, A, and B channels using cv2.split() function and assign them to variables 'l_channel', 'a_channel', and 'b_channel', respectively

5. Apply Contrast Limited Adaptive Histogram Equalization (CLAHE) to the lightness channel using cv2.createCLAHE() and cv2.apply() functions with the given clip limit and assign it to variable 'cl'

6. Merge the CLAHE enhanced L channel with the original A and B channel using cv2.merge() function and assign it to variable 'merged_channels'

7. Convert the image from LAB color model back to RGB color model using cv2.cvtColor() function and assign it to variable 'final_image'

8. Write the final image to the output path using cv2.imwrite() function

STEP 2 Training and Validation

The training process involves feeding the model input data and labels, and adjusting its internal parameters to minimize the difference between its predicted outputs and the actual outputs. While in the validation phase we evaluate the performance of the trained model on a separate dataset, known as the validation set. The purpose of the validation set is to evaluate the performance of the model on unseen data and to prevent overfitting to the training data.

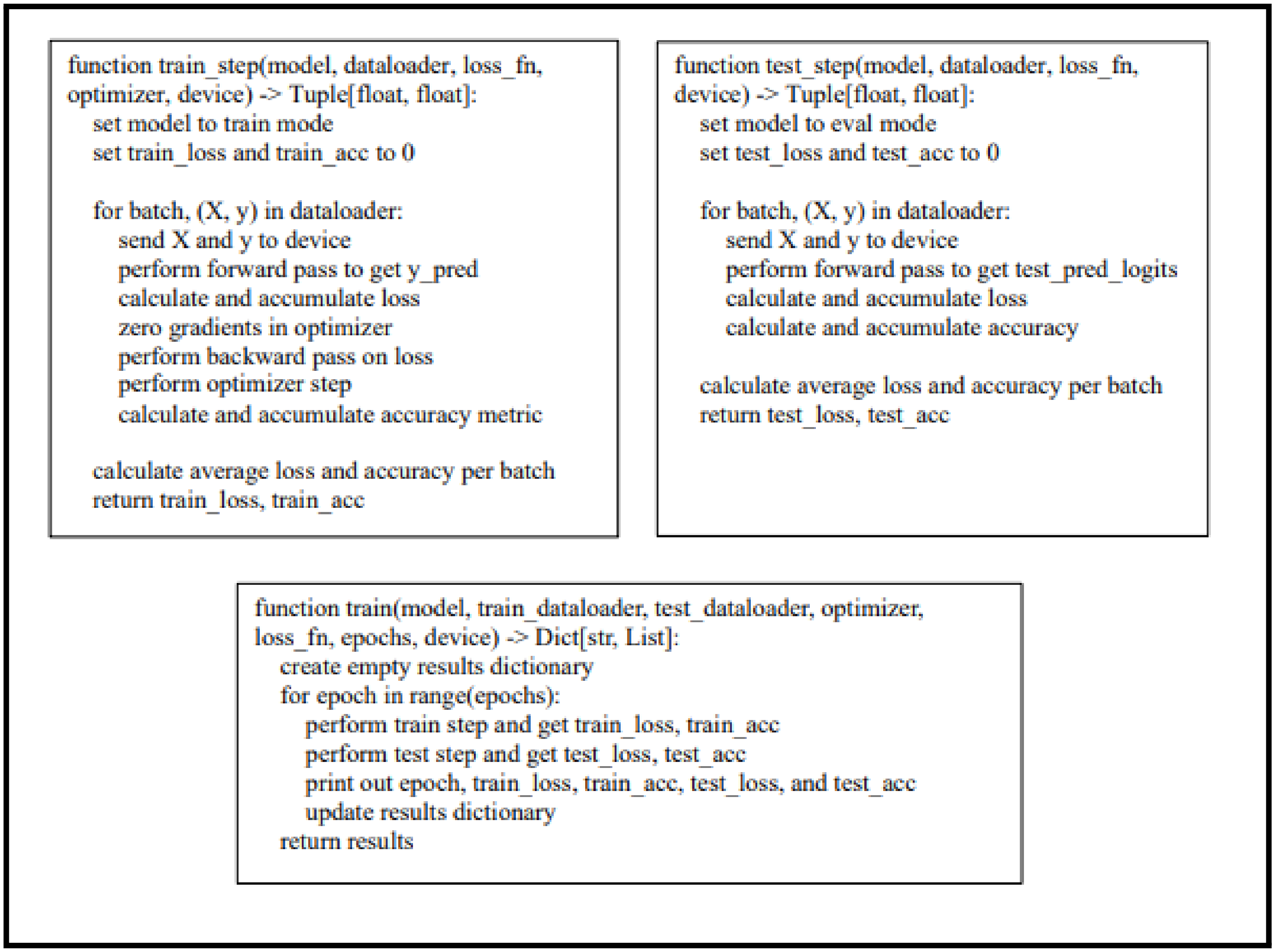
For training and validation we defined a function called *train* which calculates the loss and accuracy for both. In the *train* function we call the *train_step* function which calculates the training loss and accuracy per epoch. Similarly we have a *test_step* function which calculates the validation loss and accuracy per epoch. We conducted around hundreds of experiments to adjust the hyper parameters such as batch size, learning rate, number of epochs. We also tried different optimizers and loss functions. In our initial phase of experimentation we kept the standard batch size of 32 with learning rate of 0.001 and noticed that the model was not learning any specific patterns because of the large number of images in the batches. Then in the later phase, the models were trained with a batch size of 8 images which showed significant improvement in the results

For the case of Disease vs Normal, we trained ResNext50 and EfficientNet B6 with transfer learning for 15 epochs each, with a learning rate of 0.0001, whereas, ResNet50 and EfficientNet V2 S were trained for 21 epochs each, with a learning rate of 0.00001.

For the case of Multiclass Classification of Eye Diseases, we trained ResNext50 and EfficientNet B6 with transfer learning for 25 epoch each, with a learning rate of 0.0001, whereas, ResNet50 and EfficientNet V2 S were trained for 32 epochs each, with a learning rate of 0.0001.

For the case of Multiclass Classification of Diabetic Retinopathy Stages, we trained ResNet50 and ResNext50 with transfer learning for 75 epochs each, with a learning rate of 0.0001, whereas, EfficientNet B6 and EfficientNet V2 S were trained for 55 epochs each, with a learning rate of 0.0001.

Pseudo Code for Training and Validation



```
function train_step(model, dataloader, loss_fn,
optimizer, device) -> Tuple[float, float]:
    set model to train mode
    set train_loss and train_acc to 0

    for batch, (X, y) in dataloader:
        send X and y to device
        perform forward pass to get y_pred
        calculate and accumulate loss
        zero gradients in optimizer
        perform backward pass on loss
        perform optimizer step
        calculate and accumulate accuracy metric

    calculate average loss and accuracy per batch
    return train_loss, train_acc
```

```
function test_step(model, dataloader, loss_fn,
device) -> Tuple[float, float]:
    set model to eval mode
    set test_loss and test_acc to 0

    for batch, (X, y) in dataloader:
        send X and y to device
        perform forward pass to get test_pred_logits
        calculate and accumulate loss
        calculate and accumulate accuracy

    calculate average loss and accuracy per batch
    return test_loss, test_acc
```

```
function train(model, train_dataloader, test_dataloader, optimizer,
loss_fn, epochs, device) -> Dict[str, List]:
    create empty results dictionary
    for epoch in range(epochs):
        perform train step and get train_loss, train_acc
        perform test step and get test_loss, test_acc
        print out epoch, train_loss, train_acc, test_loss, and test_acc
        update results dictionary
    return results
```

4.2 GUI Design

To design the user interface of the project we have used Gradio library in Python. .Gradio is an open-source Python library that allows developers to quickly build custom interfaces for machine learning models. It provides an easy-to-use web interface for running and testing machine learning models, without requiring any web development experience. Gradio allows users to input data and receive predictions from their machine learning models in real-time. It also provides a variety of visualization options to help users better understand their models and their predictions. Gradio can be used for a wide range of machine learning applications, including image recognition, natural language processing, and time series analysis. Developers can quickly build interfaces for their models using just a few lines of code, and users can interact with those interfaces without needing to understand any of the underlying code or algorithms. Gradio is also highly customizable, with support for custom CSS and JavaScript to modify the appearance and behavior of the interface. It supports a wide range of machine learning frameworks and can be used with models developed in TensorFlow, PyTorch, Scikit-learn, and many other popular machine learning libraries. Overall, Gradio is a powerful tool for developers and researchers looking to create interactive web interfaces for their machine learning models.

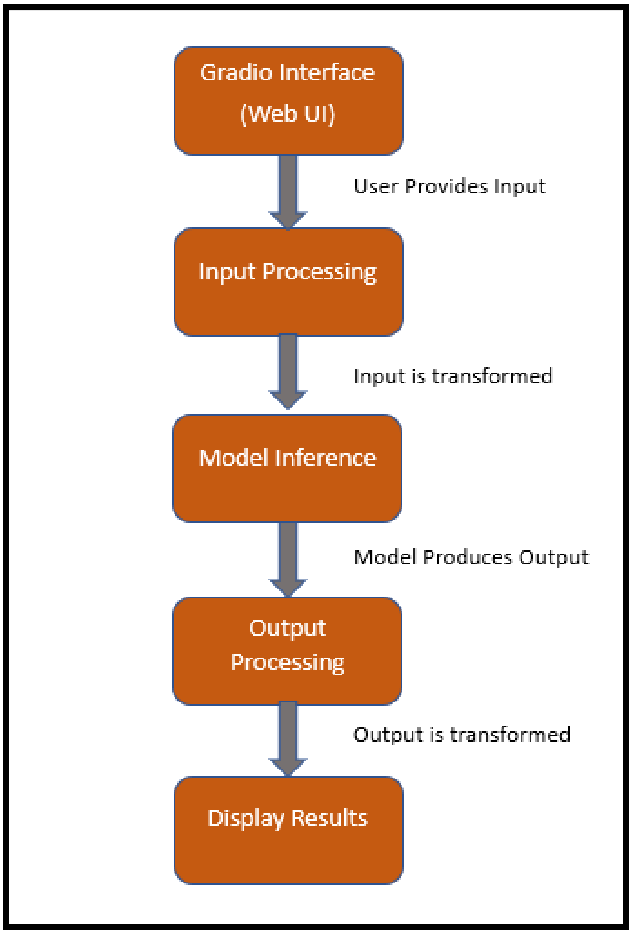
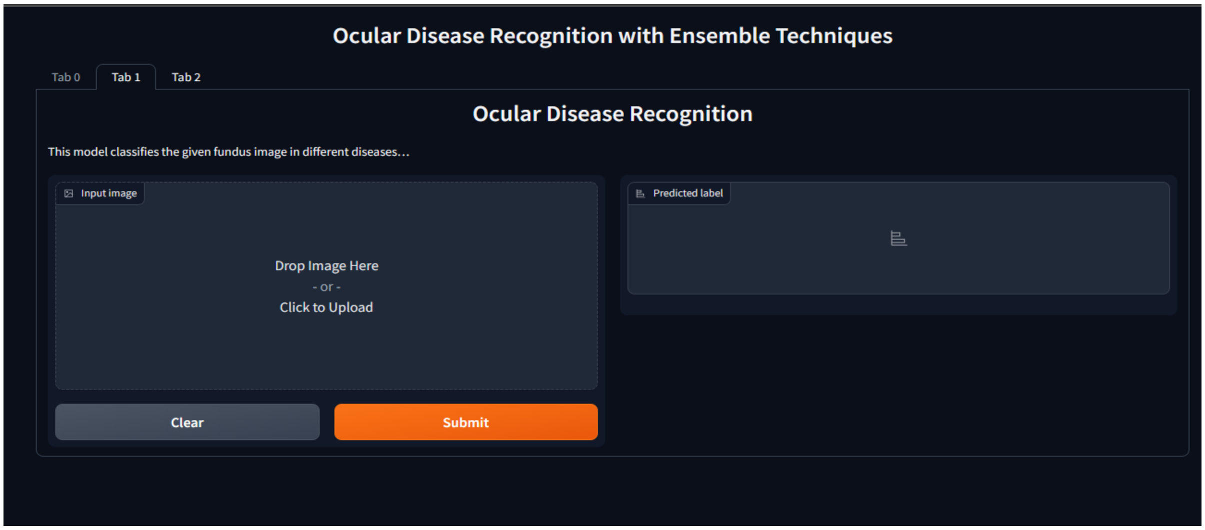


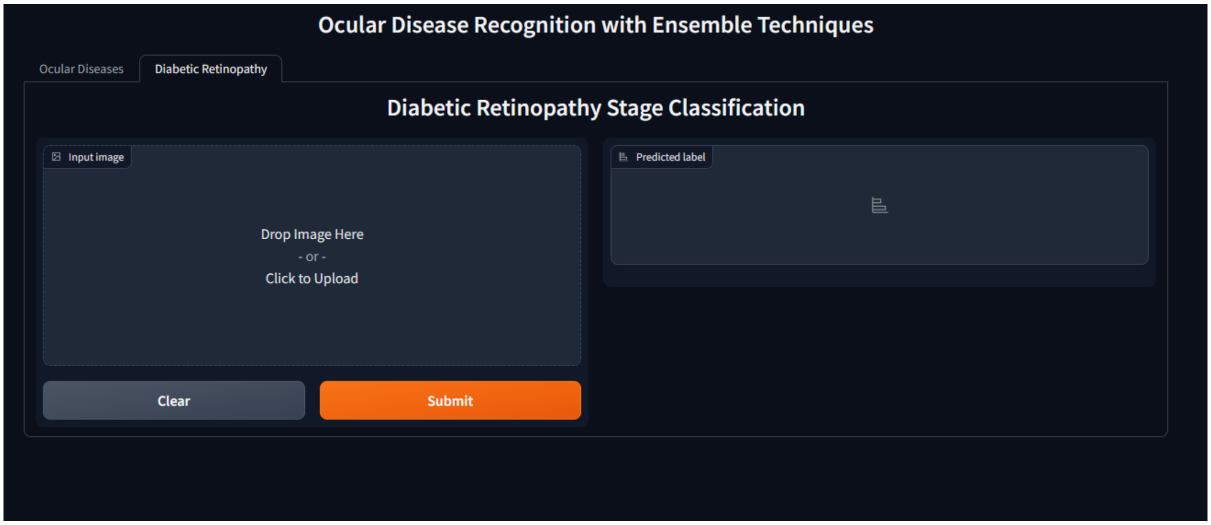
Fig 11 GUI Flowchart


Ocular Disease Recognition with Ensemble Techniques

Ocular Disease Recognition

This model classifies the given fundus image in different diseases...

Fig 12 Ocular Disease Recognition Tab on Gradio


Ocular Disease Recognition with Ensemble Techniques

Diabetic Retinopathy Stage Classification

Fig 13 Diabetes Retinopathy Disease Recognition Tab on Gradio

Chapter 5

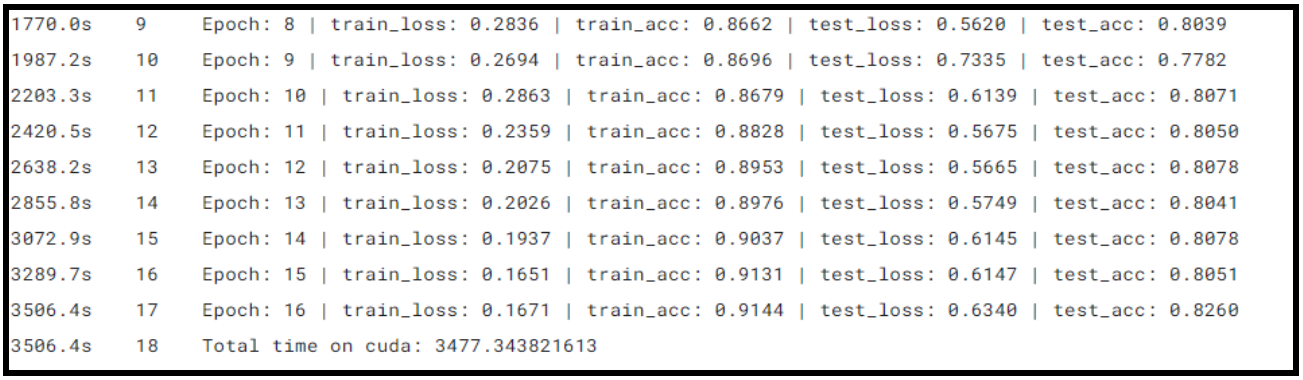
Results and Discussion

The models built during these experiments were evaluated using various performance metrics such as Precision, Recall and Accuracy. Precision is a performance metric that measures the proportion of true positive predictions among all positive predictions made by a model. It is a measure of how accurate the positive predictions made by a model are. Mathematically, precision is defined as: *precision = true positives / (true positives + false positives)* ,where true positives are the number of correctly predicted positive instances, and false positives are the number of incorrectly predicted positive instances. In other words, precision is a measure of how precise or exact the positive predictions made by a model A high precision means that the model makes few false positive predictions, i.e., it is good at identifying the true positive instances from the negative ones. Conversely, a low precision means that the model makes many false positive predictions, i.e., it is not very good at identifying the true positive instances. The precision of Ensemble for Disease Vs Normal is around 82%, Ocular Multiclass is around 83% and Diabetic Retinopathy Multi Stage around 81%.

Recall measures the proportion of true positive predictions among all actual positive instances in the data. It is a measure of how well the model can identify the positive instances in the data. Mathematically, recall is defined as: *recall = true positives / (true positives + false negatives)* where true positives are the number of correctly predicted positive instances, and false negatives are the number of positive instances that were incorrectly predicted as negative by the model. In other words, recall is a measure of how many of the positive instances in the data the model can correctly identify. A high recall means that the model can identify a large portion of the positive instances in the data, while a low recall indicates that the model misses many of the positive instances, i.e., it has a high number of false negatives. The recall of Ensemble for Disease Vs Normal is around 87%, Ocular Multiclass is around 84% and Diabetic Retinopathy Multi Stage around 81%.

Accuracy measures the proportion of correct predictions made by a model among all the predictions made. It is a measure of how well the model can correctly classify instances. Mathematically, accuracy is defined as: *accuracy = (true positives + true negatives) / (true positives + false positives + true negatives + false negatives)* ,where true positives are the number of correctly predicted positive instances, true negatives are the number of correctly predicted negative instances, false positives are the number of negative instances that were incorrectly predicted as positive, and false negatives are the number of positive instances that were incorrectly predicted as negative. The accuracy of Ensemble for Disease Vs Normal is around 84%, Ocular Multiclass is around 86% and Diabetic Retinopathy Multi Stage around 83%. We noticed that the ensembled showed better performance in all the above metrics in comparison to its individual component models.

This is the output log of one of the models we trained using our methods.



```
1770.0s    9     Epoch: 8 | train_loss: 0.2836 | train_acc: 0.8662 | test_loss: 0.5620 | test_acc: 0.8039
1987.2s    10    Epoch: 9 | train_loss: 0.2694 | train_acc: 0.8696 | test_loss: 0.7335 | test_acc: 0.7782
2203.3s    11    Epoch: 10 | train_loss: 0.2863 | train_acc: 0.8679 | test_loss: 0.6139 | test_acc: 0.8071
2420.5s    12    Epoch: 11 | train_loss: 0.2359 | train_acc: 0.8828 | test_loss: 0.5675 | test_acc: 0.8050
2638.2s    13    Epoch: 12 | train_loss: 0.2075 | train_acc: 0.8953 | test_loss: 0.5665 | test_acc: 0.8078
2855.8s    14    Epoch: 13 | train_loss: 0.2026 | train_acc: 0.8976 | test_loss: 0.5749 | test_acc: 0.8041
3072.9s    15    Epoch: 14 | train_loss: 0.1937 | train_acc: 0.9037 | test_loss: 0.6145 | test_acc: 0.8078
3289.7s    16    Epoch: 15 | train_loss: 0.1651 | train_acc: 0.9131 | test_loss: 0.6147 | test_acc: 0.8051
3506.4s    17    Epoch: 16 | train_loss: 0.1671 | train_acc: 0.9144 | test_loss: 0.6340 | test_acc: 0.8260
3506.4s    18    Total time on cuda: 3477.343821613
```

Fig 14 Output of Train function

5.1 Performance Metrics

Following are the results of all the models we evaluated during the course of this project:

Table 1. Disease Vs Normal

Model	Accuracy (%)	Precision (%)	Recall (%)
ResNet50	80.55	80.89	85.06
ResNext50	81.88	82.23	84.01
EfficientNet V2 S	82.11	81.23	84.55
Ensemble	83.85	82.01	87.84

Table 2. Multiple Ocular Disease Classification

Model	Accuracy (%)	Precision (%)	Recall (%)
ResNet50	79.67	77.55	78.78
ResNext50	82.83	81.84	83.35
EfficientNet V2 S	78.98	78.12	80.89
EfficieNetB6	80.81	79.84	82.56
Ensemble	85.60	83.45	84.67

Table 3. Diabetic Retinopathy Multiclass Classification

Model	Accuracy (%)	Precision (%)	Recall (%)
ResNet50	**79.77**	**80.54**	**80.06**
ResNext50	**78.25**	**80.87**	**80.31**
EfficientNet V2 S	**73.11**	**79.12**	**80.11**
Ensemble	**82.34**	**80.81**	**80.28**

In the results, we can see that the Ensemble model has performed well in comparison to other models. The main reason for this is reduced overfitting. Ensembling can help to reduce overfitting by combining the predictions of multiple models that have been trained on different subsets of the data. Also, each individual model in an ensemble captures different patterns in the data, which helps to improve the overall predictive accuracy of the ensemble. The impact of random noise in the data is also reduced by averaging the predictions of multiple models. Overall, ensembling is a powerful technique for improving the performance of machine learning models, particularly in situations where a single model may struggle to capture all of the important patterns in the data.

5.2 Confusion Matrices

A confusion matrix is a table used to evaluate the performance of a classification model. It displays the number of true positives, true negatives, false positives, and false negatives for each class in the model's predictions. The matrix is commonly used to calculate metrics such as accuracy, precision, and recall. Here we have displayed confusion matrices of all our models.

🟥 **- Disease Vs Normal** 🟪 **- Ocular Multi Class** 🟦 **- Diabetes Multi Stages**

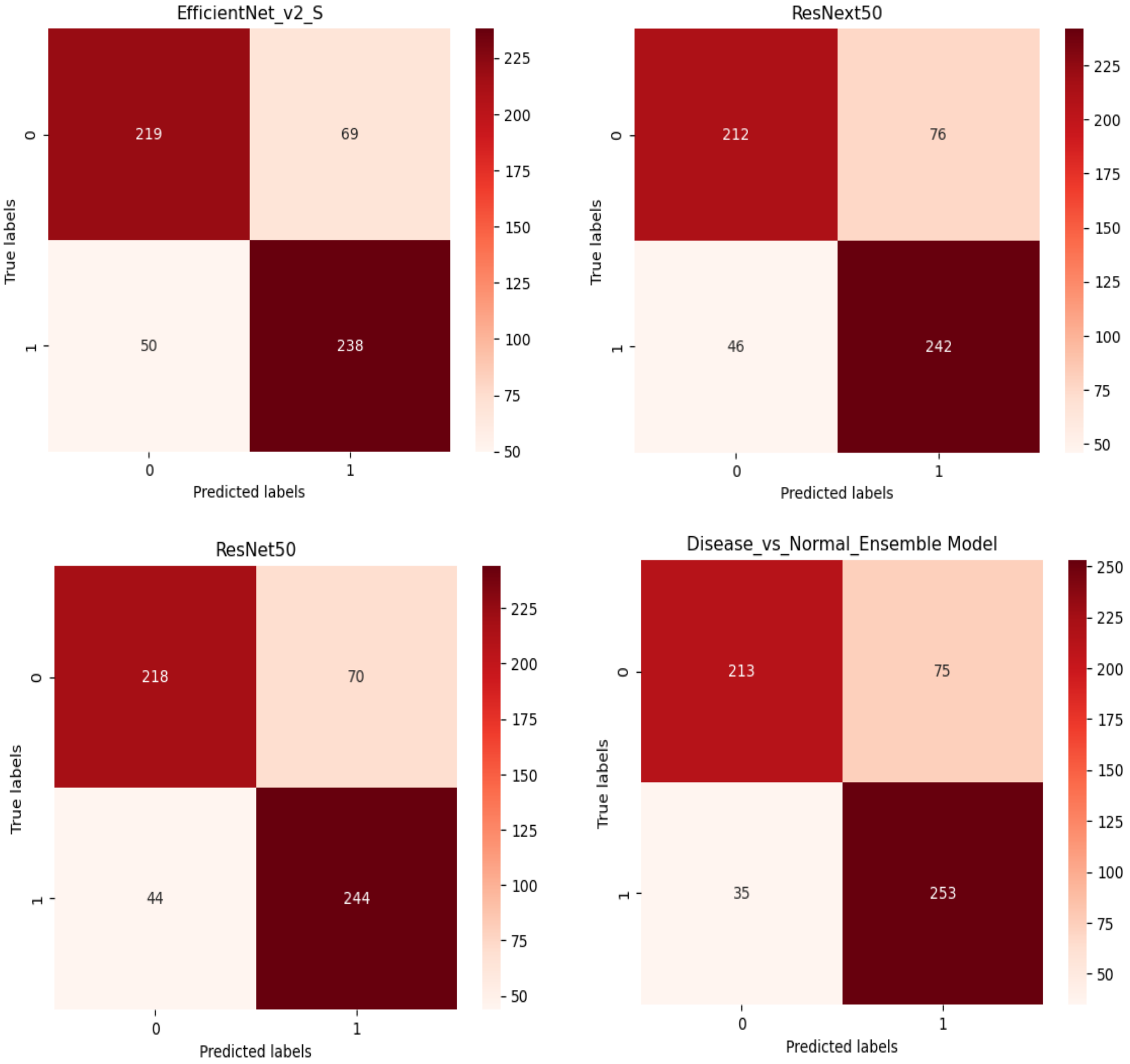


Fig 15 Disease Vs Normal Confusion Matrices

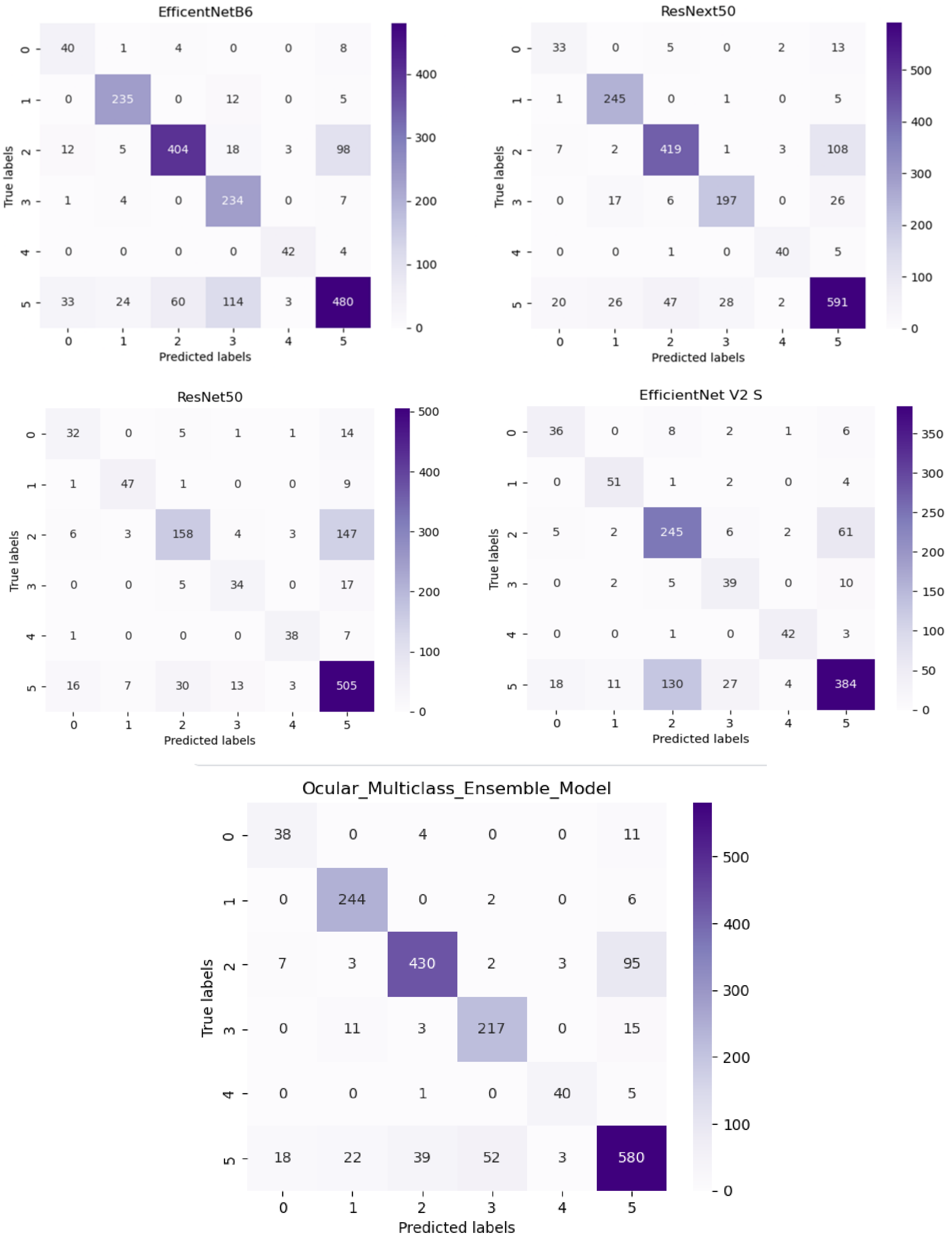
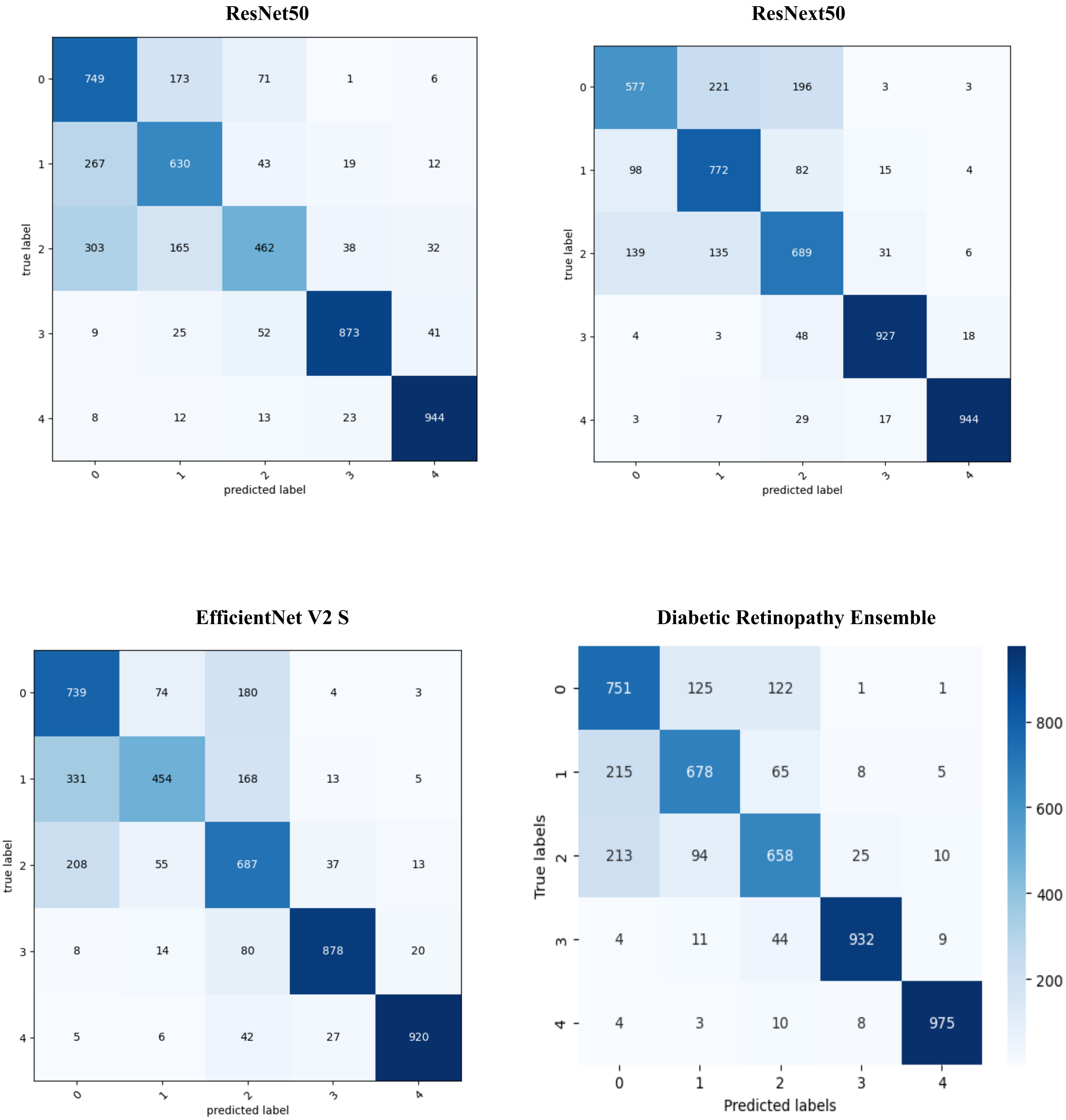


Fig 16 Ocular Multiclass Disease Confusion Matrices



ResNet50

true label \ predicted label	0	1	2	3	4
0	749	173	71	1	6
1	267	630	43	19	12
2	303	165	462	38	32
3	9	25	52	873	41
4	8	12	13	23	944

ResNext50

true label \ predicted label	0	1	2	3	4
0	577	221	196	3	3
1	98	772	82	15	4
2	139	135	689	31	6
3	4	3	48	927	18
4	3	7	29	17	944

EfficientNet V2 S

true label \ predicted label	0	1	2	3	4
0	739	74	180	4	3
1	331	454	168	13	5
2	208	55	687	37	13
3	8	14	80	878	20
4	5	6	42	27	920

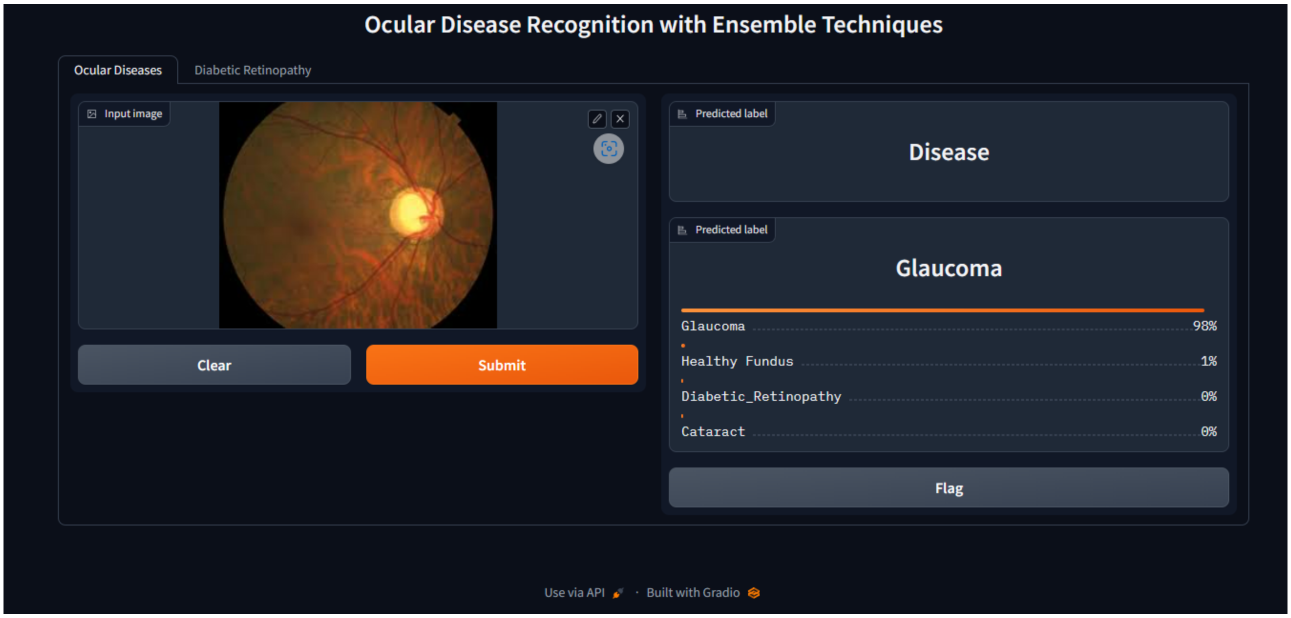
Diabetic Retinopathy Ensemble

True labels \ Predicted labels	0	1	2	3	4
0	751	125	122	1	1
1	215	678	65	8	5
2	213	94	658	25	10
3	4	11	44	932	9
4	4	3	10	8	975

Fig 17 Diabetic Retinopathy Multiclass Disease Confusion Matrices

5.3 GUI Output

Following are the outputs of our user interface which is hosted on Hugging Face Gradio. The different prediction probabilities are displayed where the highest one is the class that the model has predicted. These probabilities are calculated by the Softmax function in the model output layer.



Predicted label

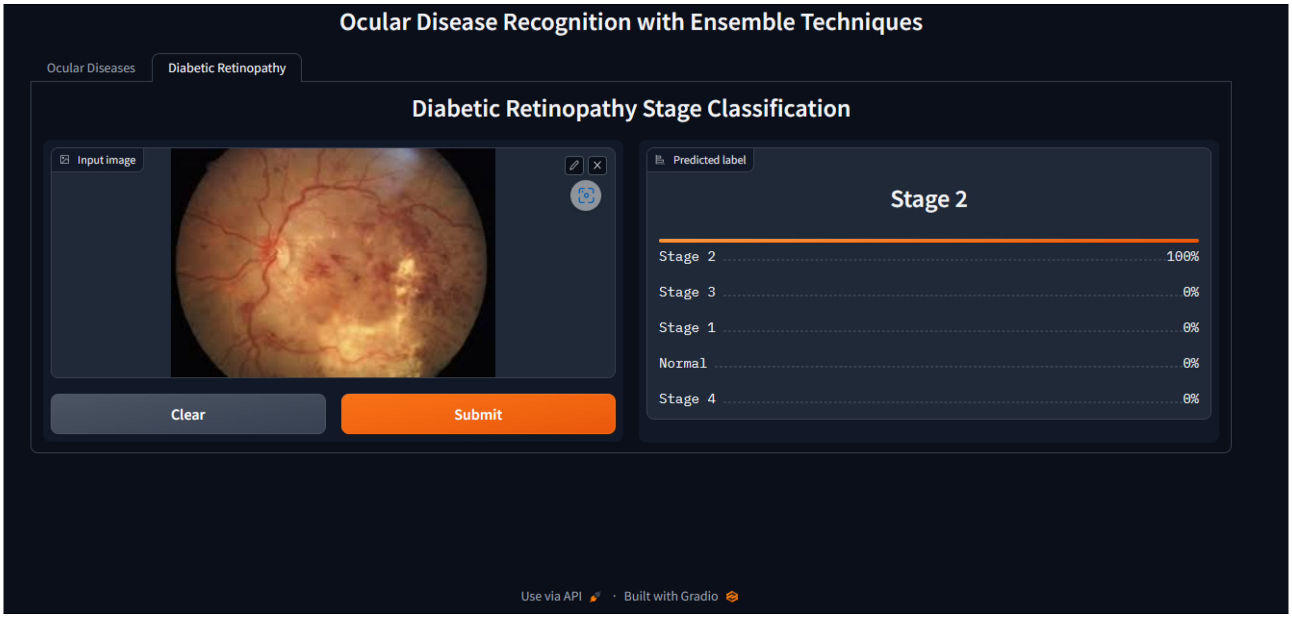
Disease

Predicted label

Glaucoma

```
Glaucoma ........................ 98%
Healthy Fundus .................. 1%
Diabetic_Retinopathy ............ 0%
Cataract ........................ 0%
```

Fig 18 Ocular Disease Recognition GUI


Ocular Disease Recognition with Ensemble Techniques
Ocular Diseases
Diabetic Retinopathy
Diabetic Retinopathy Stage Classification
Input image
Clear
Submit
Use via API
Built with Gradio

Predicted label

Stage 2

```
Stage 2 ......................... 100%
Stage 3 ......................... 0%
Stage 1 ......................... 0%
Normal .......................... 0%
Stage 4 ......................... 0%
```

Fig 19 Diabetic Retinopathy Disease Recognition GUI

Chapter 6

Conclusions

In conclusion, ocular disease recognition holds great promise for improving the accuracy and efficiency of diagnosing eye diseases. By leveraging advanced machine learning algorithms and computer vision techniques, these projects have the potential to provide fast and reliable diagnosis of ocular diseases, thus helping to improve patient outcomes. Furthermore, as these projects continue to evolve and improve, they may also help to reduce the workload of ophthalmologists and increase the accessibility of eye care in remote or underserved areas. While there are still some challenges to be addressed, such as the need for large and diverse datasets, these projects represent an exciting area of innovation and research in the field of ophthalmology. Overall, the development and deployment of ocular disease recognition projects offer a promising future for improving eye health outcomes and advancing the field of eye care.

Our results show the feasibility of classifying multiple eye diseases using standard deep learning solutions using transfer learning with loaded models with non-medical learning and the use of data augmentation to correct an imbalance problem found in the data.

We faced several difficulties in this project, some of which include:

- Limited and imbalanced datasets: One of the biggest challenges we faced was the availability of limited and imbalanced datasets. We overcame it by applying preprocessing and data augmentation procedures.
- Variation in image quality: Images of the eyes in the dataset vary in quality, making it difficult for the model to identify the features necessary for accurate diagnosis. This particularly was solved by our preprocessing function which changes the contrast levels of the images.
- Generalizability: The data available in the dataset had its inherent properties which made it difficult for the model to recognise different patterns. This problem was resolved to a certain extent by using ensemble techniques, but due to inherent randomness of the deep learning models more analysis is required.

We believe that we have achieved all the objectives defined in this work. First we did the initial learning that this project requires of us and then we studied the problem and proposed deep learning solutions for the problem of medical diagnosis. Several other deep learning methods that are available can also be implemented that may or may not prove beneficial, we leave this for the industry to explore.

References

1. Gilbert C, Foster A. Childhood blindness in the context of VISION 2020--the right to sight. Bull World Health Organ. 2001;79(3):227-32. Epub 2003 Jul 7. PMID: 11285667; PMCID: PMC2566382.
2. Yorston, D. (2003). Retinal Diseases and VISION 2020. Community Eye Health. Rabinovich, A. (2014). Going Deeper with Convolutions. 2003;16(46):19–20.
3. Bonet, E. (2018). What is an eyeshadow? Ophthalmology Service. Foundation Children's Hospital of. Barcelona, p.1.
4. Saine, P. and Tyler, M. (2002). Ophthalmic photography. Boston [Mass.]: Butterworth Heinemann.
5. Ophthalmological (2019). Technology for the revision of the retina - Advanced Ophthalmological Area.
6. Garcia , B. , De Juana , P. , Hidalgo , F and Bermejo , T. (2010). Ophthalmology. Pharmacy Hospital Volume II. Published by the SEFH. Chapter 15
7. Garrido, R. (2011). Descriptive epidemiology of the refractive state in students university students Complutense University of Madrid, p.339.
8. CI Sánchez, M. Niemeijer, AV Dumitrescu, MSA Suttorp-Schulten, MD Abràmoff and B. van Ginneken. "Evaluation of a Computer-Aided Diagnosis system for Diabetic Retinopathy screening on public data", Investigative Ophthalmology and Visual Science 2011;52:4866-4871.
9. CI Sánchez, M. Niemeijer, I. Išgum, AV Dumitrescu, MSA Suttorp-Schulten, MD Abramoff and B. van Ginneken. "Contextual computer-aided detection: Improving bright lesion detection in retinal images and coronary calcification identification in CT scans", Medical Image Analysis 2012;16(1):50-62
10. Zhou, Y., He, X., Huang, L., Liu, L., Zhu, F., Cui, S., Shao, L. (2019). Collaborative Learning of Semi-Supervised Segmentation and Classification for Medical Images. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
11. M.D. Abramoff, Y. Lou, A. Erginay, et al. Improved automated detection of diabetic retinopathy on a publicly available dataset through integration of deep learning Invest Ophthalmol Vis Sci. (2016), pp. 5200-5206
12. Parampal S. Grewal, Faraz Oloumi, Uriel Rubin, Matthew T.S. Tennant, Deep learning in ophthalmology: a review, Canadian Journal of Ophthalmology, Volume 53, Issue 4, 2018,Pages 309-313, ISSN 0008-4182,
13. A Lee, M. Seattle, P. Taylor, U. Kingdom ‘Machine Learning has arrived!’ Ophthalmology. (2017), pp. 1726-1728.
14. Ting DSW, Pasquale LR, Peng L, et al Artificial intelligence and deep learning in ophthalmology British Journal of Ophthalmology 2019;103:167-175.
15. Hijazi, S., Kumar, R. Rowen, C. Using Convolutional Neural Networks for Image Recognition. 2015. Cadence.

16. M d Shakib Khan, Nafisa Tafshir, Kazi Nabiul Alam. Deep Learning for Ocular Disease Recognition: An Inner-Class Balance. Hindawi Computational Intelligence and Neuroscience Volume 2022
17. Li Z, Liu F, Yang W, Peng S, Zhou J. A survey of convolutional neural networks: analysis, applications, and prospects. IEEE transactions on neural networks and learning systems. 2021 Jun 10.
18. Tan M, Le Q. Efficientnet: Rethinking model scaling for convolutional neural networks. International conference on machine learning 2019 May 24 (pp. 6105-6114). PMLR.
19. Benzamin A., Chakraborty C., “Detection of Hard Exudates in Retinal Fundus Images Using Deep Learning”, 2018 IEEE International Conference on System, Computation, Automation and Networking (ICSCA), 1 – 5, 2018.
20. Burlin P., Freund D. E., Joshi N., Wolfson Y., Bressler N. M., “DETECTION OF AGE-RELATED MACULAR DEGENERATION VIA DEEP LEARNING”, 2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI)(2016)
21. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556. 2014 Sep 4.
22. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the inception architecture for computer vision. InProceedings of the IEEE conference on computer vision and pattern recognition 2016 (pp. 2818-2826).
23. Ioana Madalina Tugui, Adrian Iftene,Ocular Disease Recognition, Proceedings of Symposium on Logic and Artificial Intelligence SLAI2022, January 12-16, 2022
24. LeCun Y, Bengio Y, Hinton G. Deep learning. nature. 2015 May 28;521(7553):436-44.
25. Y. Elloumi, M. Akil, and H. Boudegga, “Ocular diseases diagnosis in fundus images using deep learning: approaches, tools and performance evaluation,” in Proc. Real-Time Image Processing And Deep Learning. 109960T, Maryland, USA, 2019