

# Ocular Disease Detection Using Advanced Neural Network Based Classification Algorithms

Nadim Mahmud Dipu

Department of Electrical and Computer  
Engineering  
North South University  
Dhaka, Bangladesh  
nadim.dipu@northsouth.edu

Sifatul Alam Shohan

Department of Electrical and Computer  
Engineering  
North South University  
Dhaka, Bangladesh  
sifatul.shohan@northsouth.edu

K.M.A Salam

Department of Electrical and Computer  
Engineering  
North South University  
Dhaka, Bangladesh  
kazi.salam@northsouth.edu

**Abstract**—One of the most challenging tasks for ophthalmologists is early screening and diagnosis of ocular diseases from fundus images. However, manual diagnosis of ocular diseases is difficult, time-consuming and it can be prone to errors. That is why a computer-aided automated ocular disease detection system is required for the early detection of various ocular diseases using fundus images. Due to the enhanced image classification capabilities of deep learning algorithms, such a system can finally be realized. In this study, we present four deep learning-based models for targeted ocular tumor detection. For this study, we trained the cutting-edge image classification algorithms such as Resnet-34, EfficientNet, MobileNetV2, and VGG-16 on the ODIR dataset consisting of 5000 fundus images that belong to 8 different classes. Each of these classes represents a different ocular disease. The VGG-16 model achieved an accuracy of 97.23%; the Resnet-34 model reached an accuracy of 90.85%; the MobileNetV2 model provided an accuracy of 94.32%, and the EfficientNet classification model achieved an accuracy of 93.82%. All of these models will be instrumental in building a real-time ocular disease diagnosis system.

**Keywords**—Ocular Disease Classification, Color Fundus Photography, Ocular Disease Detection, Convolutional Neural Networks, EfficientNet, VGG-16, Resnet-34, MobileNetV2, Transfer Learning

## I. INTRODUCTION

Various ocular diseases are capable of causing permanent and irreversible damage to the patient's vision, and in extreme cases, it can even lead to blindness [1-3]. Although effective treatments are available for these ocular diseases, these treatment options can only be implemented if the disease is diagnosed as early as possible. Ocular diseases are primarily diagnosed using color fundus photography or CFP [4]. This technique is utilized in order to record the interior surface of the human eye so that various types of possible ocular diseases can be detected [5].

Although this method of diagnosis is effective, it's still quite difficult to detect certain ocular diseases using CFP. Some of the most prevalent ocular diseases, such as cataracts, myopia, and diabetic retinopathy are difficult to diagnose as they show very few initial symptoms. [6] Moreover, the process of manually inspecting and detecting ocular diseases is a laborious task, and this process is not that accurate [7].

In recent times deep learning-based neural network models have shown promising results in medical image classification and object detection. [8-10] Moreover, that is why convolutional neural network-based models have been extensively studied for ocular disease detection [9] [11-12].

Meng et al. [13] proposed a two-stage process of utilizing convolutional neural networks (CNN) on fundus images in order to perform Optic Disc (OD) localization. Automatic ocular disease classification models have been proposed by He et al. [14] that are based on knowledge distillation. This system is built by training and optimizing two deep networks sequentially.

Roy et al. [15] suggested a fully convolutional deep architecture called ReLayNet for segmenting retinal layers and fluids from Optical Coherence Tomography (OCT) scans. This technique utilizes an encoder-decoder network for semantic segmentation on OCT scans.

Liefers et al. [16] used a fully convolutional neural network that had dilated convolution filters in order to implement a pixel-wise classification on Optical Coherence Tomography (OCT) scans. The performance of this model was evaluated on a dataset consisting of 400 OCT scans of patients who were affected by varying stages of age-related macular degeneration.

Lee et al. [17] proposed a CNN-based model that can detect intra-retinal fluid on OCT images. This model was trained on 1,289 OCT scans, and the images segmented by the CNN model received a cross-validated Dice score of 0.911.

A novel convolutional multi-task architecture was proposed by Ployat et al. [18] that takes a supervised learning approach. This model is trained to perform three tasks simultaneously and those tasks involve segmentation of bright lesions, segmentation red lesions, and lesion detection. The area under ROC curve of this model was 0.839.

Hu et al. [19] proposed a retinal vessel segmentation technique that's implemented using a convolutional neural network and fully connected conditional random fields (CRFs). The accuracy and effectiveness of this model was evaluated on the color fundus images taken from STARE [20] and DRIVE [21] datasets.

Gulshan et al. [22] proposed a deep learning-based algorithm for automating the process of diabetic macular edema and diabetic retinopathy detection. This task was done using an optimized neural network-based image classification model.

Li et al. [23] proposed a deep learning-based system to detect Glaucomatous Optic Neuropathy (GON). This study was also done on color fundus photographs. The researchers had trained a classification model that was trained on 8000 color fundus images. The model achieved a sensitivity score of 95.6%, specificity of 92.00%, and an AUC score of 0.986.



Karri et al. [24] presented an algorithm that can identify different retinal pathologies from optical coherence tomography images (OCT) images. This algorithm was developed by fine-tuning a pre-trained convolutional neural network called GoogleNet [25]. The dataset used in this study had four distinct classes that included dry age-related macular degeneration, diabetic macular edema, and no pathology.

Although almost all of these studies have shown promising results, only a few of the existing studies have addressed the task of classifying multiple ocular diseases from fundus images. Furthermore, an automated ocular disease diagnostic tool will require a robust model that has been thoroughly trained on multiple ocular diseases so that it can detect diseases from color fundus images.

The models that we have discussed so far are highly effective at performing specific classification or segmentation tasks such as segmenting retinal vessels and classifying a specific ocular disease. However, they cannot be used as a generalized ocular disease detection system.

Our task was to classify ocular diseases from color fundus photographs as effectively as possible. Although various CNN-based classification models have been used for ocular disease classification before, the latest, state-of-the-art classification models such as EfficientNet [28] and VGG-16 [29] have not been extensively studied in this regard. These models have been highly effective at classification tasks performed on various other medical imaging datasets. That is why we chose to use these models in order to determine their performance of on ODIR dataset. This way we can figure out which model would be ideal for building an autonomous ocular disease detection system.

## II. DATASET

For this study, we have used the Ocular Disease Intelligent Recognition (ODIR) dataset. [26] It is one of the largest publicly available multiclass ocular disease detection datasets in the world. This dataset was compiled by Shangong Medical Technology Co, limited by taking collecting fundus images from different hospitals in China. The fundus images of this dataset are split into eight different ocular disease classification categories. These categories include seven disease classes that are diabetes (D), cataract (C), glaucoma (G), age-related muscular degeneration (A), myopia (M), hypertension (H), and other abnormalities/diseases (O). In total, this dataset contains 5000 cases of color fundus photographs (CFPs), and it is split into training and testing subsets. Roughly 3500 cases are used for training, and the rest are used for testing. Some sample images of the ODIR dataset can be observed in figure 1.

The class distribution of the images is illustrated in the bar chart shown in figure 2. And we can see the details regarding the image distribution of the dataset in table I.

## III. METHODOLOGY

### A. Classification Using Resnet-34

Resnet refers to a convolutional neural network architecture that's extensively used as a classification model. [31]

The Resnet-34 model that we used in our study was pretrained on a large image classification dataset known as Imagenet [27]. The Imagenet dataset contains 14 million images that are categorized into 1000 classes.

This model is then further trained on the ODIR dataset so that it can be used for classifying ocular diseases. This process of training a pretrained image classifying model on custom images is called transfer learning.

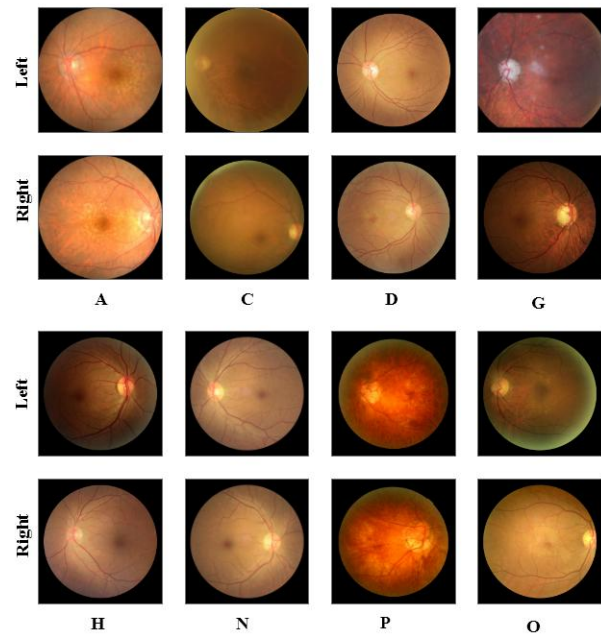


Fig. 1. A look at the fundus images of the ODIR dataset.

TABLE I. DISTRIBUTION OF THE IMAGES IN THE DATASET

No.	Labels	Training Cases	Off-Sie Training Cases	On-Sie Training Cases	All Cases
1	N	1,135	161	324	1,620
2	D	1,131	162	323	1,616
3	G	207	30	58	307
4	C	211	32	64	243
5	A	171	25	47	295
6	H	94	14	30	138
7	M	177	23	49	249
8	O	944	134	268	1,346

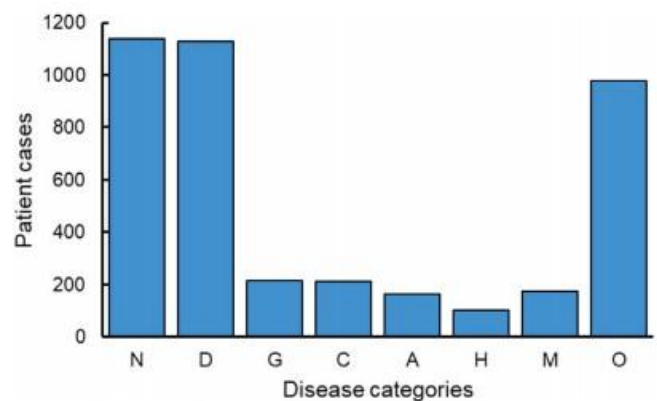


Fig. 2. Bar chart representing the distribution of the dataset.

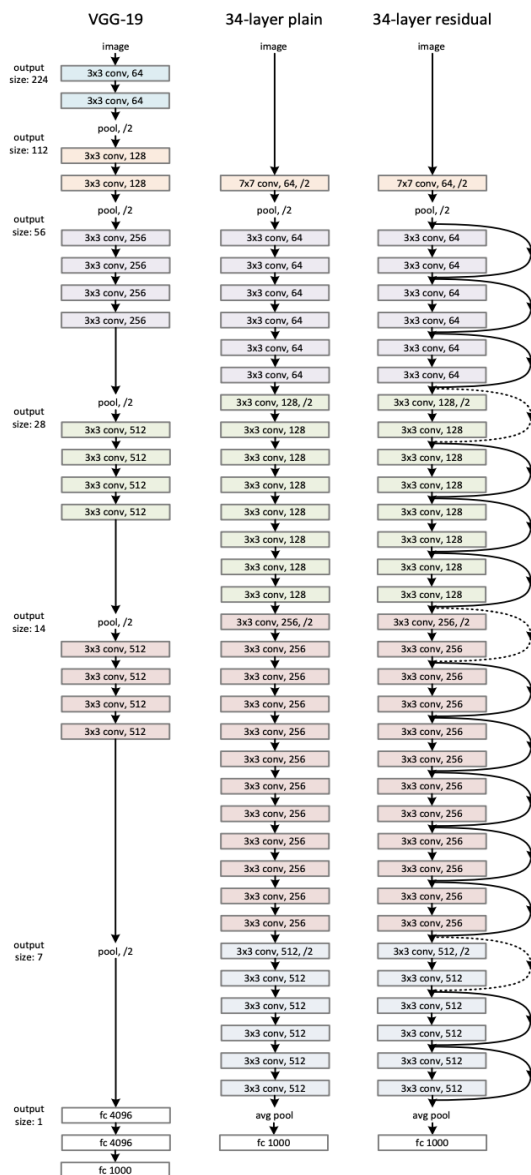


Fig. 3. Resnet-34 Architecture.

The overall architecture of the Resnet-34 model is shown in figure 3. The architecture of the Resnet-34 model might seem a bit complicated, which is why a more simplified illustration is shown in figure 4. At first, we loaded the dataset into a colab notebook for training the Resnet-34 model. For creating this model, we utilized the FastAi library. After that, we downloaded a custom pretrained Resnet image classification model.

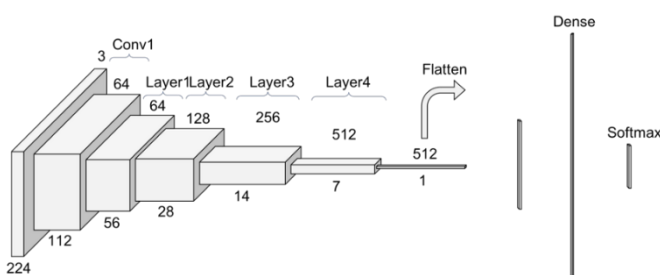


Fig. 4. Simplified representation of Resnet-34 model.

Next, we initialized our classification model by fine-tuning its final layer while the rest of the model was frozen.

This way, the model was able to learn about all of the necessary pretrained features. After that the model was run for 50 epochs. We used a training callback function called “early stopping” that stops the training process if the validation loss of the model does not decrease for more than 20 epochs.

After that, we unfroze the model’s parameters and then proceeded to calculate the ideal learning rate. The model was further training for 50 epochs to ensure our classification model provided the maximum performance.

### B. Classification Using EfficientNet

EfficientNet is one of the most sophisticated models out there when it comes to custom image classification. It is an open-source, state-of-the-art CNN-based model that was developed by Google Brain. In order to create this model, we used the Keras deep learning framework, and we implemented it in Google Colab. We used a supervised learning approach to training the EfficientNet model on the ODIR dataset.

This model was trained by passing the features of the training images into the deep neural network, and its task is to provide the probabilities of the test images belonging to a particular class. In this case, the class that has the highest probability according to the model is considered to be the model’s prediction. The architecture of the EfficientNet model is illustrated in figure 5.

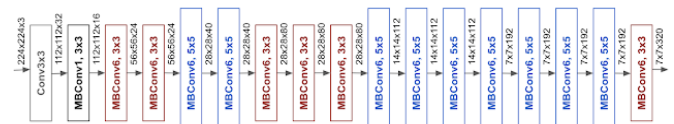


Fig. 5. EfficientNet Architecture

EfficientNet was developed in order to test how to effectively scale the overall size of the convolutional neural networks (CNNs). The comparison of the various scaling methods used in EfficientNet is shown in figure 6. Just like Resnet-34, the EfficientNet model is also pretrained and benchmarked on the ImageNet dataset. That's why it has a strong understanding of the general features that are required to classify the images.

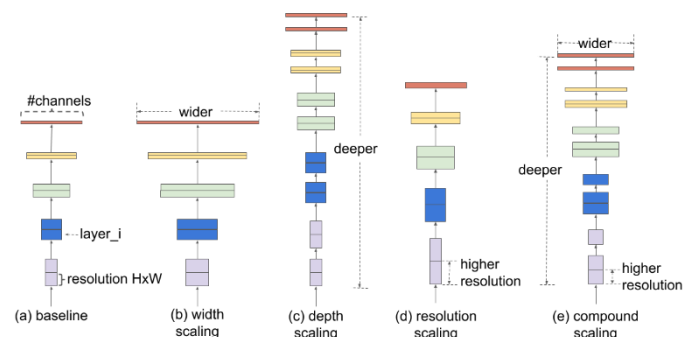


Fig. 6. Scaling of EfficientNet architecture.

While creating and running this model, we enabled the GPU environment on our Google Colab notebook. Next, we had to ensure our model was running on the TensorFlow 1.x environment and Keras 2.3.1 was installed. After that, we imported the EfficientNetB0 model from the Keras library. Then we set the input resolution of the images to be 150 x 150.



The next task was to import our ocular disease dataset into the notebook and then utilize transfer learning in order to classify ocular diseases from color fundus images. After importing the fundus images, we passed the data through a training generator function to prepare it for training.

Next, we set the number of epochs for which our model will train on the dataset to be 150, and the final layer of the EfficientNet model was removed so that it can be replaced by eight layers that correspond to the eight classes of our dataset. Finally, we trained the model and then evaluated its performance using the images inside the test directory.

### C. Classification Using MobileNetV2

MobileNetV2 is an image classification model that was developed by Google, and its task is to provide efficient real-time classification even in constrained computing environments such as smartphones. [30] This model is quite similar to the previous two models in the sense that it also utilizes transfer learning, and it's pretrained on the ImageNet dataset as well. The architecture of the MobileNetV2 model is illustrated in figure 7.

This image classification framework uses an inverted residual structure in which the input and output layers of the residual blocks comprise thin bottleneck layers. Moreover, the convolutions used in this model are quite lightweight and it does not have non-linearities in its narrow layers.

In order to implement the MobileNetV2 model on the ODIR dataset, we had to import the dataset into our Colab notebook and then convert the images into a TensorFlow dataset. We built the TensorFlow dataset by using the ImageFolder API provided by the TensorFlow framework. After that, we instantiated the MobileNetV2 classification model in which the classification layers were dependent on the last layer before the flatten operation was performed on it. Then we set the compiled model using categorical cross entropy as the loss function and accuracy as the evaluation metric. Finally, we generated the Accuracy vs. Epoch and Cross Entropy vs. Epoch graphs in order to evaluate the efficiency of our model.

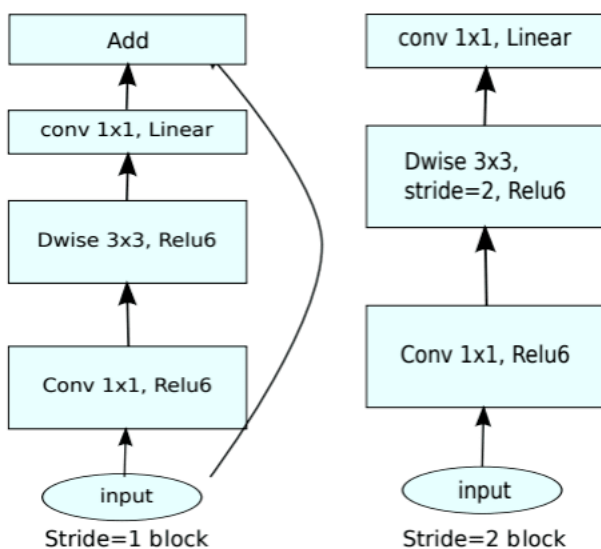


Fig. 7. MobileNetV2 architecture.

### D. Classification Using VGG16 Model

The VGG-16 model was developed by researchers of University of Oxford and the VGG-16 paper was published in 2015. It's widely regarded as one of the best image classification models out there, and it achieved 92.7% accuracy on the ImageNet dataset. It is built using a large number of tiny convolutional filters that allow the model to learn about complicated pixel relational data.

At first, we download the required libraries and dependencies to make sure that the environment is compatible with the VGG-16 classification model. The VGG-16 model expects the data to have an input size of 224 x 224. Moreover, that is why we had to resize our training images accordingly.

After that, we performed some preprocessing on the images in order to make them suitable for the VGG-16 model. This was done using the ImageDataGenerator module of the Keras library. Those preprocessing steps involved setting the re-scale value to 1/255, shear range to 0.2, zoom range to 0.2, and the value of horizontal flip to true. The task of the ImageDataGenerator function is to generate the preprocessed images based on the parameters that we have set so that those images can be fed to the VGG-16 model. Some samples of the preprocessed images can be seen in figure 8.

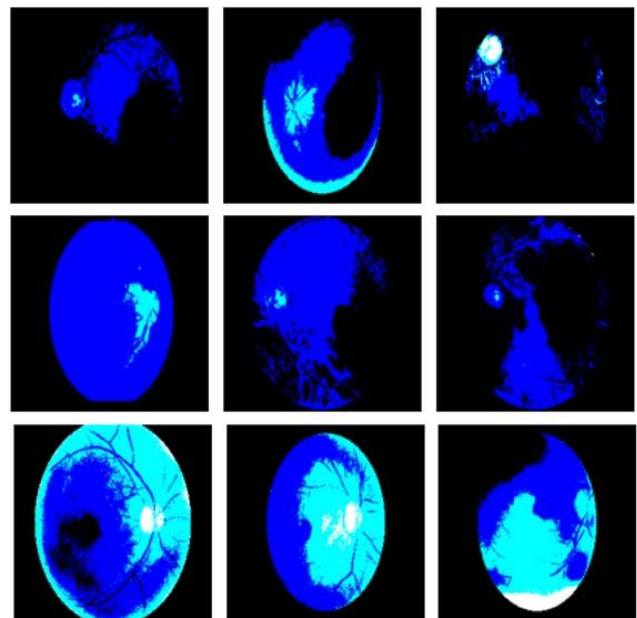


Fig. 8. A look at the preprocessed ODIR datasets.

In order to make our VGG-16 model train and predict on eight different ocular disease classes, we had to append two Dense layers to the existing VGG-16 architecture. The overall architecture of the VGG-16 model after modification is shown in table II.

This model is set to use the Adam optimizer and categorical cross entropy loss function. It also used the softmax activation function. After those parameters were set, we ran the model on the training set for 150 epochs. When the training was finished, we evaluated its performance on the validation set.

### E. Evaluation of The Models

In order to analyze the results provided by the classification models in a comprehensible way, we have used evaluation metrics such as accuracy, precision, recall, F1 score, etc. The formulas of these evaluation metrics are shown below:

$$Accuracy = \frac{(TP+FN)}{TP+TN+FP+FN} \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$F1 = \frac{2*Precision*Recall}{Precision+Recall} \quad (4)$$

Here,

TP = True Positive (The total number of images that are correctly detected to be positive)

FP = False Positive (The total number of images that are predicted to be positive but actually are negative)

TN = True Negative (The number of images that are accurately predicted to be negative)

FN = False Negative (The number of images that are incorrectly predicted to be negative)

TABLE II. VGG-16 ARCHITECTURE

Layer (type)	Output Shape	Param #
block1 conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1 conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2 conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3 conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4 conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
flatten (Flatten)	(None,)	0

	25088)	
fc1 (Dense)	(None, 4096)	102764544
fc2 (Dense)	(None, 4096)	16781312
Total params: 134,260,544 Trainable params: 134,260,544 Non-trainable params: 0		

## IV. EXPERIMENTAL RESULTS

In total we had to perform four different experiments in order to determine the performance of the four classification models (Resnet-34, EfficientNet, MobileNetV2, VGG-16).

### A. Resnet-34 Classification Model

We trained the Resnet-34 model for up to 200 epochs for the ocular disease classification task. Its performance was impressive, and it achieved an accuracy of 93.47% on the training set. Moreover, it achieved an accuracy of 90.85% on the test set, which comprised of previously unseen images. We had split the dataset consisting of 5000 color fundus photographs into a training set comprising 3500 images (70% of the total images) and a test set of 1500 images (30% of the total images) of the total MRI scans. We primarily evaluated our four models on the test set in order to make it easy for us to compare and contrast the models with one another. This was done to simulate how the Resnet-34 model would perform in a real-life scenario with previously unseen color fundus images. The output generated by the Resnet-34 model on the test set is shown in figure 10.

The confusion matrix for test set is shown in Figure 9. Out of the eight classes the most successful prediction made on the other diseases class. Overall, the performance of this model was quite satisfactory on the test set.

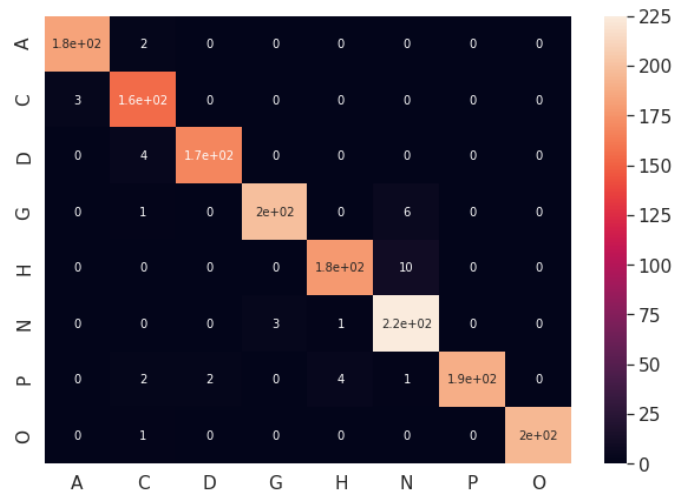


Fig. 9. Confusion matrix produced by the Resnet-34 classification model

A confusion matrix is a type of layout that provides visualization of the performance of an algorithm. Each row of the confusion matrix represents the instances in a true/actual class. And each column of the matrix represents the instances in a predicted class. The values located at the main diagonal of the matrix represents the instances at which the model was able to accurately predict the class to which an image belongs to. On the other hand, all of the other nonzero values in the confusion matrix represents the instances at which the model had incorrectly classified an image.

The precision of the Resnet-34 model is 93.70%, and its recall was 92.65%. Furthermore, the model achieved an F1 score of 93.17%. The class-wise accuracy, precision, recall and F1 score of this model is given in table number III.

TABLE III. PERFORMANCE OF THE RESNET-34 MODEL

Class	Accuracy	Precision	Recall	F1 Score
AMD	99.64%	0.99	0.98	0.98
Cataract	99.64%	0.98	0.99	0.98
Diabetes`	92%	0.57	0.97	0.72
Glaucoma	92.07%	0.98	0.61	0.75
Hypertension	99.34%	0.98	0.99	0.98
Normal	99.39%	0.98	0.98	0.98
Myopia	99.52%	0.99	0.97	0.98
Other	99.82%	0.98	1.0	0.99

### B. EfficientNet Classification Model

Just like the Resnet-34 model we also trained the Efficient-NetB0 model for 500 epochs. It performed exceptionally well on the training set as well as the test set. It achieved an accuracy of 94.97% on the training set. And it achieved an accuracy of 93.82% on the test set.

The confusion matrix produced by the EfficientNetB0 model on the test set is shown in Figure 11. This model provided the most successful prediction on the — class.

The EfficientNet model had an overall precision of 92.73%, and its recall was 96.25%. Furthermore, this model achieved an F1 score of 93.74%. The class-wise analysis of the accuracy, precision, recall, and F1 score of the EfficientNet model is given in table number IV. As mentioned before, we used the softmax function as a loss function for the EfficientNet model. The figure 12 shows the training and validation accuracy of the model. Here the dots represent the training accuracy and the curve below shows the validation accuracy. We can observe from this figure that the training accuracy drastically increases as the number of epochs increases. The validation accuracy increases as well but it sometimes declines as well during the training process.

And the graph shown in figure 13 illustrates the training and validation loss of the EfficientNet model. Here the x-axis represents the number of epochs and the y-axis represents the training and validation loss of the model.

We can clearly see that both the training and validation losses of the EfficientNet model drastically decreases as the number of epochs increase.

### C. MobileNetV2 Classification Model

The performance of the MobileNetV2 model was fairly close to the previous two classification models. It achieved an accuracy of 95.56% on the training set. And it achieved an accuracy of 94.32% on the test set.

The training and validation accuracy graph as well as the training and validation loss graph is illustrated in figure 14.

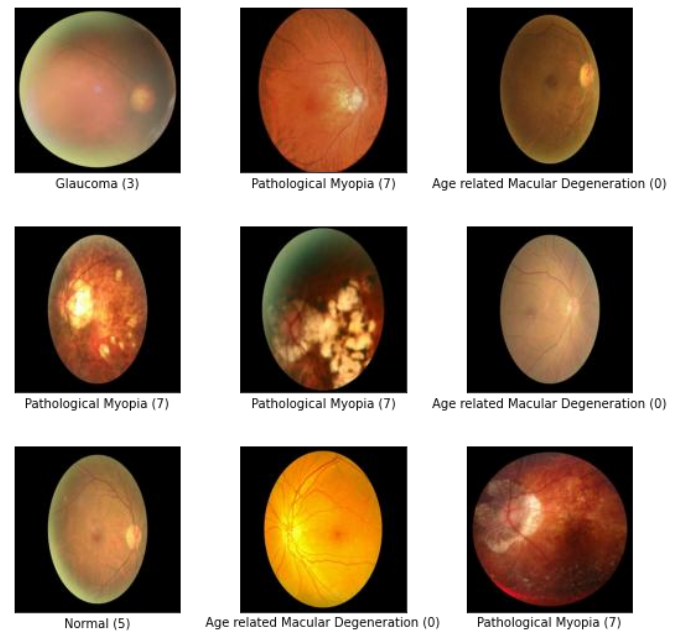


Fig. 10. Output generated by Resnet-34.

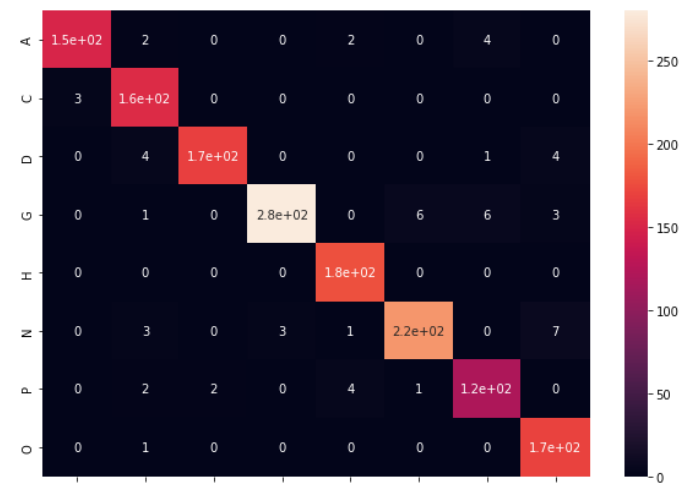


Fig. 11. Confusion Matrix generated by EfficientNet

The MobileNetV2 model had a precision score of 93.33%, and its recall was 89.67%. Furthermore, this model achieved an F1 score of 91.46%. The details of the class-wise accuracy, precision, recall and F1 score of the model is shown in table number V.

TABLE IV. PERFORMANCE OF THE EFFICIENTNET MODEL

Class	Accuracy	Precision	Recall	F1 Score
AMD	99.27%	0.95	0.98	0.96
Cataract	98.93%	0.98	0.92	0.95
Diabetes	99.27%	0.95	0.99	0.97
Glaucoma	98.27%	0.95	0.99	0.97
Hypertension	98.73%	1.0	0.96	0.98
Normal	99.53%	0.94	0.97	0.98
Myopia	98.67%	0.93	0.92	0.92
Other	99%	0.99	0.92	0.96

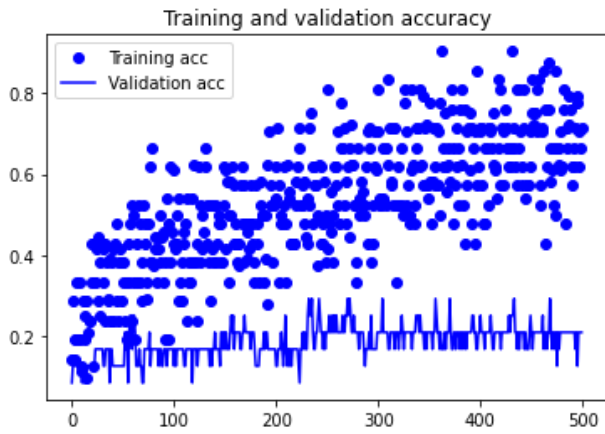


Fig. 12. Training and Validation Accuracy Graph of the EfficientNet model.

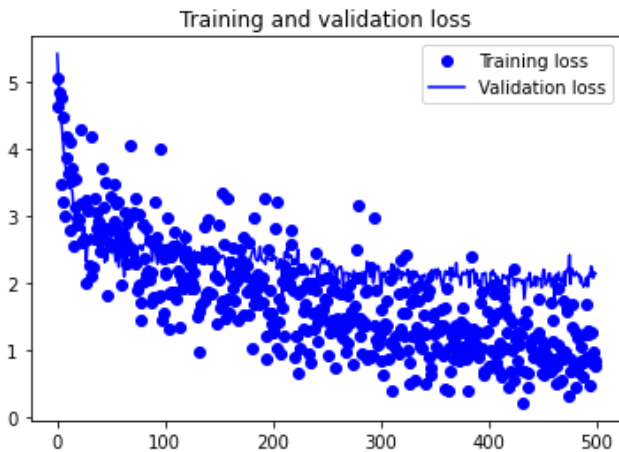


Fig. 13. Training and Validation Loss of EfficientNet Model.

#### D. VGG-16 Classification Model

Out of the four classification models the VGG-16 model had the best performance in terms of accuracy. It achieved an accuracy of 98.65% on the training set. The accuracy achieved on the test set was 97.23%. The confusion matrix of the VGG-16 model is shown in figure 15.

We can observe from figure 14 that the training as well as the validation accuracy rises exponentially as the number of epochs increase. Furthermore, both training and validation loss (cross entropy) decreases as the number of epochs rises. This happens because as the model gets trained for more and more epochs it learns more about the features of the images and it gets better at differentiating between the images belonging to different classes, thus increasing its accuracy.

The VGG-16 model had a precision score of 96.73%, and its recall was 93.76%. Furthermore, this model achieved an F1 score of 95.22%. The class-wise details of this model are included in table number VI.

Our models have outperformed some of the existing solutions to ocular disease detection and classification. For instance, He et al. [9] had achieved an F1 score of 90.4% of the ODIR dataset using their ResNet-34 model.

However, our Resnet-34 model achieved an F1 score of 93.17%.

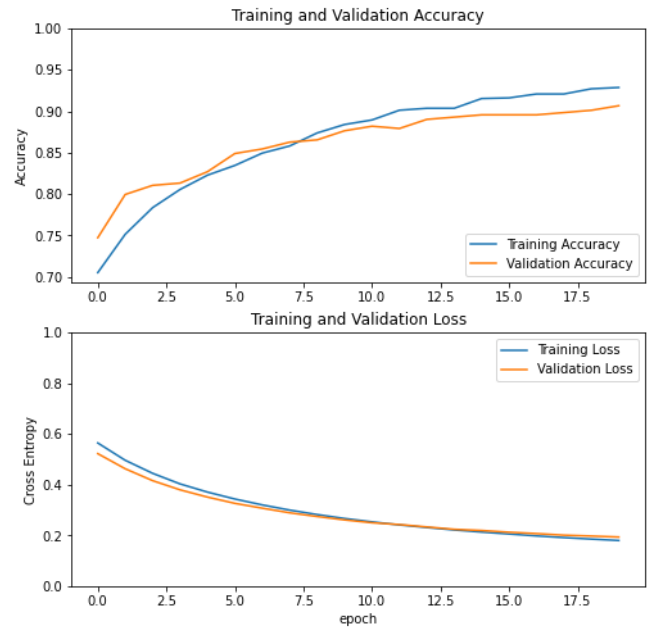


Fig. 14. Training and Validation Accuracy and Loss Graph of the VGG-16 Model.

TABLE V. PERFORMANCE OF THE MOBILENETV2 MODEL

Class	Accuracy	Precision	Recall	F1 Score
AMD	98.73%	0.94	0.94	0.94
Cataract	97.73%	0.97	0.91	0.94
Diabetes	98.47%	0.91	0.96	0.93
Glaucoma	98.67%	0.94	0.99	0.93
Hypertension	92.47%	0.92	0.93	0.91
Normal	98.13%	0.93	0.95	0.94
Myopia	98.20%	0.92	0.88	0.90
Other	98.87%	0.99	0.91	0.95

TABLE VI. PERFORMANCE OF THE VGG-16 MODEL

Class	Accuracy	Precision	Recall	F1 Score
AMD	98.27%	0.94	0.90	0.92
Cataract	97.6%	0.91	0.88	0.89
Diabetes	97.87%	0.88	0.93	0.91
Glaucoma	97.67%	0.92	0.95	0.93
Hypertension	96.89%	0.87	0.92	0.90
Normal	97.93%	0.93	0.94	0.93
Myopia	97.00%	0.85	0.84	0.94
Other	98.33%	0.98	0.89	0.93

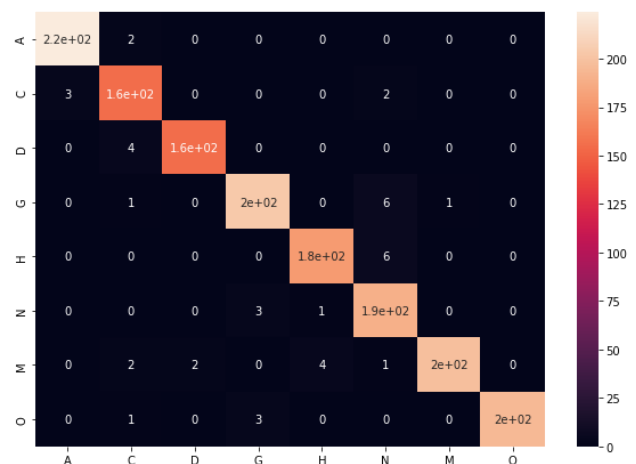


Fig. 15. Confusion matrix produced by VGG-16.



This was because we had trained our model for more epochs and we had fine-tuned our model for better gaining accuracy.

## V. CONCLUSION

In this study, we have developed four neural network-based ocular disease, classification models. Those models are Resnet-34, EfficientNet, MobileNetV2 and VGG-16. Out of which, the VGG-16 provided the best accuracy of 97.23% when it comes to classifying ocular diseases from fundus photographs. The performance of the other models was also satisfactory. We have performed extensive experiments on the publicly available ODIR- 2019 dataset to validate our proposed method's effectiveness. Our proposed method can generate more impressive results than the existing CNN-based ocular disease classification models while at the same time requiring lower computational power.

The best part about our proposed method is that it can easily be extended to other types of medical image-based disease classification. Furthermore, the models described in this study can be used in order to build a user-friendly, real-time ocular diseases classification system. Such a system will be a great help to the medical professionals and it will revolutionize the field of ocular disease diagnosis.

## REFERENCES

- [1] Bourne, R. R., Stevens, G. A., White, R. A., Smith, J. L., Flaxman, S. R., Price, H., Jonas, J. B., Keeffe, J., Leasher, J., Naidoo, K., Pesudovs, K., Resnikoff, S., & Taylor, H. R. (2013). Causes of vision loss worldwide, 1990–2010: a systematic analysis. *The Lancet Global Health*, 1(6). [https://doi.org/10.1016/s2214-109x\(13\)70113-x](https://doi.org/10.1016/s2214-109x(13)70113-x)
- [2] Sommer, A., Tielsch, J. M., Katz, J., Quigley, H. A., Gottsch, J. D., Javitt, J. C., Martone, J. F., Royall, R. M., Witt, K. A., & Ezrine, S. (1991). Racial Differences in the Cause-Specific Prevalence of Blindness in East Baltimore. *New England Journal of Medicine*, 325(20), 1412–1417. <https://doi.org/10.1056/nejm199111143252004>
- [3] Congdon, N., O'Colmain, B., Klaver, C. C., Klein, R., Muñoz, B., Friedman, D. S., Kempen, J., Taylor, H. R., Mitchell, P., & Eye Diseases Prevalence Research Group (2004). Causes and prevalence of visual impairment among adults in the United States. *Archives of ophthalmology* (Chicago, Ill. : 1960), 122(4), 477–485.
- [4] Application of Ocular Fundus Photography and Angiography. (2014). *Ophthalmological Imaging and Applications*, 154–175. <https://doi.org/10.1201/b17026-12>
- [5] Rowe, S., MacLean, C. H., & Shekelle, P. G. (2004). Preventing Visual Loss From Chronic Eye Disease in Primary Care. *JAMA*, 291(12), 1487. <https://doi.org/10.1001/jama.291.12.1487>
- [6] Kessel, L., Erngaard, D., Flesner, P., Andresen, J., Tendal, B., & Hjortdal, J. (2015). Cataract surgery and age-related macular degeneration. An evidence-based update. *Acta Ophthalmologica*, 93(7), 593–600. <https://doi.org/10.1111/aos.12665>
- [7] Li, N., Li, T., Hu, C., Wang, K., & Kang, H. (2021). A Benchmark of Ocular Disease Intelligent Recognition: One Shot for Multi-disease Detection. *Benchmarking, Measuring, and Optimizing*, 177–193. [https://doi.org/10.1007/978-3-030-71058-3\\_11](https://doi.org/10.1007/978-3-030-71058-3_11)
- [8] Miranda, E., Aryuni, M., & Irwansyah, E. (2016, November). A survey of medical image classification techniques. In 2016 International Conference on Information Management and Technology (ICIMTech) (pp. 56-61). IEEE.
- [9] He, J., Li, C., Ye, J., Qiao, Y., & Gu, L. (2021). Multi-label ocular disease classification with a dense correlation deep neural network. *Biomedical Signal Processing and Control*, 63, 102167
- [10] Li, C., Ye, J., He, J., Wang, S., Qiao, Y., & Gu, L. (2020, April). Dense correlation network for automated multi-label ocular disease detection with paired color fundus photographs. In 2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI) (pp. 1-4). IEEE
- [11] Tayal, A., Gupta, J., Solanki, A., Bisht, K., Nayyar, A., & Masud, M. (2021). DL-CNN-based approach with image processing techniques for diagnosis of retinal diseases. *Multimedia Systems*, 1-22.
- [12] Akil, M., Elloumi, Y., & Kachouri, R. (2020). Detection of retinal abnormalities in fundus image using CNN deep learning networks.
- [13] Meng, X., Xi, X., Yang, L., Zhang, G., Yin, Y., Chen, X. (2018). Fast and effective optic disk localization based on convolutional neural network. *Neurocomputing*, 312, 285–295. <https://doi.org/10.1016/j.neucom.2018.05.114>
- [14] He, J., Li, C., Ye, J., Qiao, Y., Gu, L. (2021). Self-speculation of clinical features based on knowledge distillation for accurate ocular disease classification. *Biomedical Signal Processing and Control*, 67, 102491. <https://doi.org/10.1016/j.bspc.2021.102491>
- [15] Roy, A. G., Conjeti, S., Karri, S. P., Sheet, D., Katouzian, A., Wachinger, C., Navab, N. (2017). ReLayNet: retinal layer and fluid segmentation of macular optical coherence tomography using fully convolutional networks. *Biomedical Optics Express*, 8(8), 3627. <https://doi.org/10.1364/boe.8.003627>
- [16] Lee, C. S., Tying, A. J., Deruyter, N. P., Wu, Y., Rokem, A., Lee, A. Y. (2017). Deep-learning based, automated segmentation of macular edema in optical coherence tomography. *Biomedical Optics Express*, 8(7), 3440. <https://doi.org/10.1364/boe.8.003440>
- [17] Payout, C., Duval, R., Cheriet, F. (2019). A Novel Weakly Supervised Multitask Architecture for Retinal Lesions Segmentation on Fundus
- [18] Images. *IEEE Transactions on Medical Imaging*, 38(10), 2434–2444. <https://doi.org/10.1109/tmi.2019.2906319>
- [19] Hu, K., Zhang, Z., Niu, X., Zhang, Y., Cao, C., Xiao, F., Gao, X. (2018). Retinal vessel segmentation of color fundus im-ages using multiscale convolutional neural network with an im-proved cross-entropy loss function. *Neurocomputing*, 309, 179–191. <https://doi.org/10.1016/j.neucom.2018.05.011>
- [20] M. D. M. Goldbaum, STARE Dataset Website, Clemson University, Clemson, SC, USA, 1975
- [21] J. J. Staal, M. D. Abramoff, M. Niemeijer, M. A. Viergever, and B. van Ginneken, Digital Retinal Image for Vessel Extraction (DRIVE) Database, Image Sciences Institute, University Medical Center Utrecht, Utrecht, The Netherlands, 2004
- [22] Gulshan, V., Peng, L., Coram, M., Stumpe, M. C., Wu, D., Narayanaswamy, A., Venugopalan, S., Widner, K., Madams, T., Cuadros, J., Kim, R., Raman, R., Nelson, P. C., Mega, J. L., Webster, D. R. (2016). Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs. *JAMA*, 316(22), 2402. <https://doi.org/10.1001/jama.2016.17216>
- [23] Li, Z., He, Y., Keel, S., Meng, W., Chang, R. T., He, M. (2018). Efficacy of a Deep Learning System for Detecting Glaucomatous Optic Neu-ropathy Based on Color Fundus Photographs. *Ophthalmology*, 125(8), 1199–1206. <https://doi.org/10.1016/j.ophtha.2018.01.023>
- [24] Karri, S. P., Chakraborty, D., Chatterjee, J. (2017). Transfer learning based classification of optical coherence tomography images with diabetic macular edema and dry age-related macular degeneration. *Biomed-ical Optics Express*, 8(2), 579. <https://doi.org/10.1364/boe.8.000579>
- [25] Zhong, Z., Jin, L., & Xie, Z. (2015, August). High performance offline handwritten chinese character recognition using googlenet and directional feature maps. In 2015 13th International Conference on Document Analysis and Recognition (ICDAR) (pp. 846-850). IEEE.
- [26] Li, N., Li, T., Hu, C., Wang, K., & Kang, H. (2021). A Benchmark of Ocular Disease Intelligent Recognition: One Shot for Multi-disease Detection. *Benchmarking, Measuring, and Optimizing*, 177–193. [https://doi.org/10.1007/978-3-030-71058-3\\_11](https://doi.org/10.1007/978-3-030-71058-3_11)
- [27] J. Deng, W. Dong, R. Socher, L. Li, Kai Li and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 248-255, doi: 10.1109/CVPR.2009.5206848.
- [28] Tan, M., & Le, Q. (2019, May). Efficientnet: Rethinking model scaling for convolutional neural networks. In International Conference on Machine Learning (pp. 6105-6114). PMLR.
- [29] Qassim, H., Verma, A., & Feinzimer, D. (2018, January). Compressed residual-VGG16 CNN model for big data places image recognition. In 2018 IEEE 8th Annual Computing and



- Communication Workshop and Conference (CCWC) (pp. 169-175). IEEE.
- [30] Targ, S., Almeida, D., & Lyman, K. (2016). Resnet in resnet: Generalizing residual architectures. arXiv preprint arXiv:1603.08029.
- [31] Tan, M., & Le, Q. (2019, May). Efficientnet: Rethinking model scaling for convolutional neural networks. In International Conference on Machine Learning (pp. 6105-6114). PMLR.