IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 5, May 2025

DOI: 10.17148/IARJSET.2025.125357

# Eye Disease Detection

**Prof. Harish H K[1], Namratha S N[2], Keerthana C N[3],Prathiksha B S[4], Rohini R[5]**

Assistant Professor, Department of Computer Science and Engineering, Maharaja Institute of Technology Mysore, Mysore, Karnataka, India.1 Designation, Department, College, City, Country[1]

Undergraduate Students, Department of Computer Science and Engineering, Maharaja Institute of Technology Mysore, Mysore, Karnataka, India.[2-5]

**Abstract**: Diabetic retinopathy, glaucoma, and cataracts are among the most widespread eyerelated diseases, posing significant challenges in the realm of global public health. Early diagnosis and intervention are critical to preventing irreversible vision impairment. This study focuses on the development of an efficient deep learning algorithm for the detection of eye diseases depicted in fundus images.

## I. INTRODUCTION

Despite India's high burden of blindness, 85% of cases being treatable, there's a critical need for an eAcient diagnostic system. This study aims to create an automatic, deep learning-based solution to rapidly detect Diabetic Retinopathy, Glaucoma, and Cataracts in fundus images with high accuracy. Current categorization methods are limited, and the proposed deep convolutional neural network (DCNN) model offers a promising approach. The study seeks to achieve detection accuracies of 91% for Diabetic Retinopathy, 89% for Cataracts, and 85% for Glaucoma. An intuitive online interface will make accessing this diagnostic tool easy for users

EfficientNet, a convolutional neural network, utilizes "compound scaling" to balance model complexity, accuracy, and computational efficiency. This technique involves adjusting three crucial factors: width (channels), depth (layers), and resolution (input image size) of the network architecture. Width scaling enhances complexity, depth scaling captures intricate representations, and resolution scaling improves detail. The model employs Mobile Inverted Bottleneck (MBConv) layers, featuring depth-wise and point-wise convolutions, and Squeeze-and-Excitation (SE) blocks for optimization. The MBConv layer efficiently maintains representational power, and the SE block focuses on essential features. EfficientNet variants (B0, B1, etc.) offer different trade-offs between size and accuracy to cater to diverse requirements.

MobileNetV2, tailored for mobile devices, employs inverted residual blocks, linear bottlenecks, and skip connections to strike a balance between model size and accuracy. By leveraging depth-wise separable convolutions, its architecture ensures a lightweight design, ideal for resource-limited environments, while maintaining competitive performance in tasks such as image classification and object detection. In MobileNetV2, two types of blocks are employed: one with a stride of 1 (residual block) and the other with a stride of 2 for downsizing. Both block types consist of three layers: a 1×1 convolution with ReLU6, a depthwise convolution, and a 1×1 convolution without additional non-linearity. This design, excluding ReLU in the last layer, prevents diminishing the model's expressive power. The expansion factor t is set to 6 for all main experiments, resulting in an internal output of 64×6=384 channels when the input has 64 channels

## II. RELATED WORK

A easy to use website was build to present the model in an appealing way. The website allows for instant screening of fundus images taken with a fundus camera. The uploaded image undergoes preprocessing and is then processed by the Convolutional Neural Network (CNN) architecture, built using Keras and Tensorflow. The system, based on its training, provides results, which are displayed on the GUI along with corresponding confidence percentages.These percentages indicate the system's confidence in determining the presence or absence of a disease. Development Tools: Software tools used for development include Jupyter (an open-source platform) with Keras and TensorFlow for building, training, and testing the network. The GUI was developed with the help of FastAPI for backend deployment, React for GUI/frontend HTML5 was employed to design the webpage, defining the structure and behaviour of web page content.

It proposed a methodology for cataract detection and classification. While it contains technical details and observations related to the methodology. Cataract detection and classification are crucial tasks in the field of ophthalmology and

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 5, May 2025

DOI: 10.17148/IARJSET.2025.125357

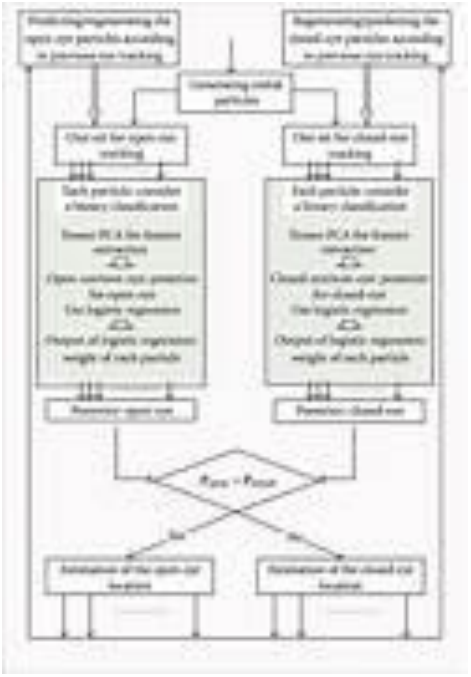

Figure 1 : Flowchart of Eye Disease Detection

A. Grayscale

i. Represent grayscale images using 8-bit brightness values ranging from 0 to 255.

ii. ii. Convert color images (24-bit) to grayscale images (8-bit) for further processing.Convert color images (24-bit) to grayscale images (8-bit) for further processing. These preprocessing steps are crucial for improving image quality and consistency, which is essential for effective feature extraction and the training of deep learning systems, particularly in cases where image data may have low fidelity and variations in quality.

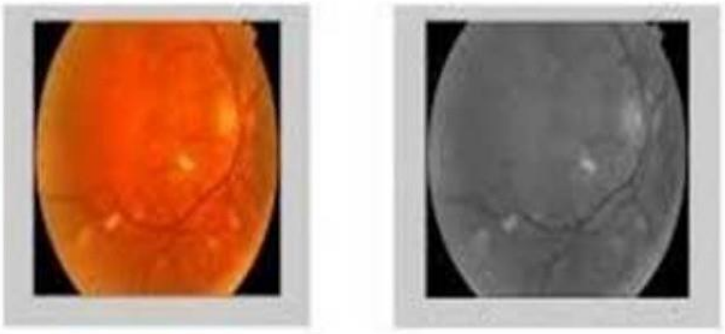

Figure 2: Color Space Conversion

The technique used in this neural network has a precise approach for image processing and the model development. The images preprocessed from the training dataset are given to the proposed deep convolutional network model. A sequence of multiple convolution, pooling, and Rectified Linear Unit (ReLU) layers are present in this model, through which features are extracted from the images. These layers form the hidden layers, which helps in abstracting high- level features from input data. The depth of the neural network affects the training process and thus influences the feature extraction. We have

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 5, May 2025

DOI: 10.17148/IARJSET.2025.125357

connected and pooling layers. Convolutional maps are generated by convolu- tional layers by convolving with input pixels and ReLU activation function is utilized to generate feature maps. Diverse pooling layers, each having distinct filters, are used to identify specific features and image components. The output from tha base model is passed through various dense and dropout layers. Finally, the input image is categorized among 4 classes for disease classification.

Various regularization techniques, such as L2 kernel regularization and L1 activity and bias regularization, are applied within the dense layer to combat overfitting and enhance the model's ability to generalize. We have also used dropout layers with a rate of 0.45 to further prevent overfitting by randomly dropping out some layers during the training process. Adam optimizer is used with a standard learning rate of 0.001 for model training. Model is further optimized using categorical crossentropy loss function to compute difference between predicted and actual class distributions.

In that phase, several critical parameters were determined to fine-tune the training process of a deep-learning model. The number of samples processed per iteration was regulated by using batch size of 40 during training. Such a value affects the speed and accuracy of the training operation. By conducting the training for 30 epochs, which is the complete number of passes over the whole training dataset, the model parameters were optimized iteratively. Furthermore, a patience value of 1 was allotted, implying the number of epochs to be patient for the improvement in monitored outputs before trying to revise the learning rate.

For cases in which monitored statistics do not record progress for 3 epochs, training is over to keep the model from overfitting and increasing the stop patience value. In addition, a threshold of 0.9 was defined in order to dynamically change training by monitoring metrics dependent on the performance of the model. If the training accuracy falls below this point, the monitoring is conducted based on accuracy; otherwise, it is based on validation loss. The learning rate is reduced by 0.5 factor after reaching the patience threshold. Furthermore, training is managed through an inquiry function every 5 epochs which is designated by ask epoch value.

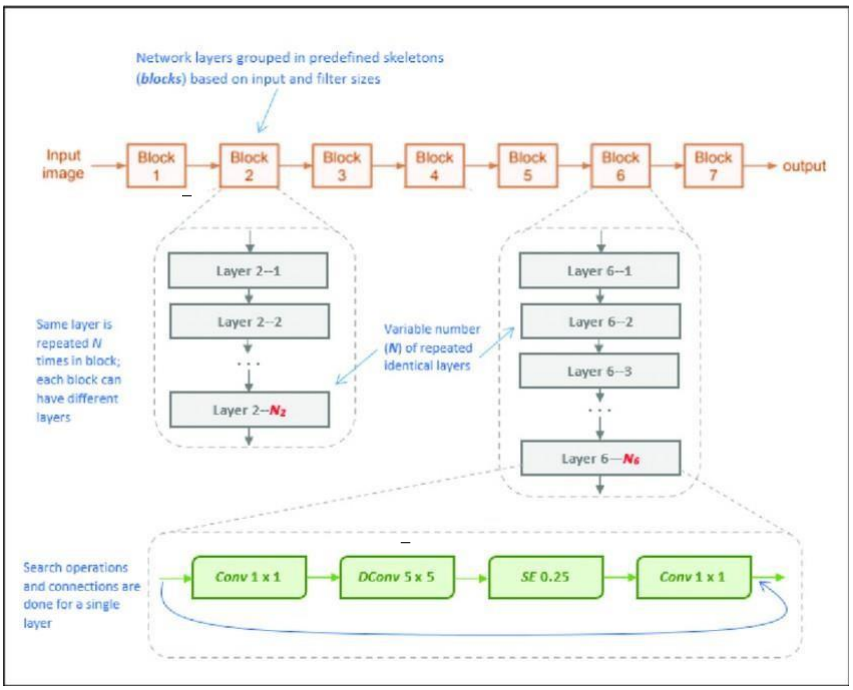


Figure 3: NASNet Architecture

NASNet (Neural Architecture Search Network) is a neural network architecture designed through automated neural architecture search. Developed by Google's DeepMind and Google Brain teams, NASNet employs a repeating cell structure containing normal and reduction cells, each discovered through reinforcement learning during the search process. The architecture is efficient and scalable, adapting well to various tasks and computational resources. NASNet demonstrates competitive performance across various computer vision tasks, including object detection and image classification. Its flexibility lies in automatically discovering architectures suited to different problem domains.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 5, May 2025

DOI: 10.17148/IARJSET.2025.125357


Figure 4 : EfficientNetB3 Model Performance

The number of samples processed per iteration was regulated by using batch size of 40 during training. Such a value affects the speed and accuracy of the training operation. By conducting the training for 30 epochs, which is the complete number of passes over the whole training dataset, the model parameters were optimized iteratively. Furthermore, a patience value of 1 was allotted, implying the number of epochs to be patient for the improvement in monitored outputs before trying to revise the learning rate. For cases in which monitored statistics do not record progress for 3 epochs, training is over to keep the model from overfitting and increasing the stop patience value. In addition, a threshold of 0.9 was defined in order to dynamically change training by monitoring metrics dependent on the performance of the model. If the training accuracy falls below this point, the monitoring is conducted based on accuracy; otherwise, it is based on validation loss. The learning rate is reduced by 0.5 factor after reaching the patience threshold. Furthermore, training is managed through an inquiry function every 5 epochs which is designated by ask epoch value.

## III. DATASET

**Dataset**: The dataset is classified into 4 classes: diabetic retinopathy, glaucoma, cataract, and normal eye. The datasets are taken from various contributors across the kaggle website. Source [17] dataset is used in this research. This dataset contain the above four classes, but there are a few technical issues in the dataset for the particular disease- Glaucoma. Hence the dataset for this disease is referred from various different available datasets . Similar findings are evident in other research papers, as observed in reference [13]. Thus, the final total number of images are 3721, containing eye diseases and the normal eye images with following distribution:
Data is partitioned in ratio of 80-10-10 for the following three processes:

A. Training Dataset

It is the fundamental data which is used for training the model. Maximum percentage of the overall dataset is used for training purposes. The content of supervised learning is an output variable for one or multiple input variables.

B. Validation Dataset

A small percentage of dataset is used for validation purposes in DNN models. Once the model is trained and the model is ready for prediction, validation data is used to evaluate the model performance on unfamiliar data. Thus, cross-validation is performed to check the accuracy of the model on unseen data.

## IV. ROPOSED METHODOLOGY

The technique used in this neural network has a precise approach for image processing and the model development. The images preprocessed from the training dataset are given to the proposed deep convolutional network model. A sequence of multiple convolution, pooling, and Rectified Linear Unit (ReLU) layers are present in this model, through which features are extracted from the images. These layers form the hidden layers, which helps in abstracting high- level features from input data. The depth of the neural network affects the training process and thus influences the feature extraction.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 5, May 2025

DOI: 10.17148/IARJSET.2025.125357

We have used transfer learning techniques to effectively increase the accuracy of the model keeping the computational resources minimal.

In that phase, several critical parameters were determined to fine-tune the training process of a deep-learning model. The number of samples processed per iteration was regulated by using batch size of 40 during training. Such a value affects the speed and accuracy of the training operation. By conducting the training for 30 epochs, which is the complete number of passes over the whole training dataset, the model parameters were optimized iteratively. Furthermore, a patience value of 1 was allotted, implying the number of epochs to be patient for the improvement in monitored outputs before trying to revise the learning rate. For cases in which monitored statistics do not record progress for 3 epochs, training is over to keep the model from overfitting and increasing the stop patience value. In addition, a threshold of 0.9 was defined in order to dynamically change training by monitoring metrics dependent on the performance of the model. If the training accuracy falls below this point, the monitoring is conducted based on accuracy; otherwise, it is based on validation loss. The learning rate is reduced by 0.5 factor after reaching the patience threshold. Furthermore, training is managed through an inquiry function every 5 epochs which is designated by ask epoch value.

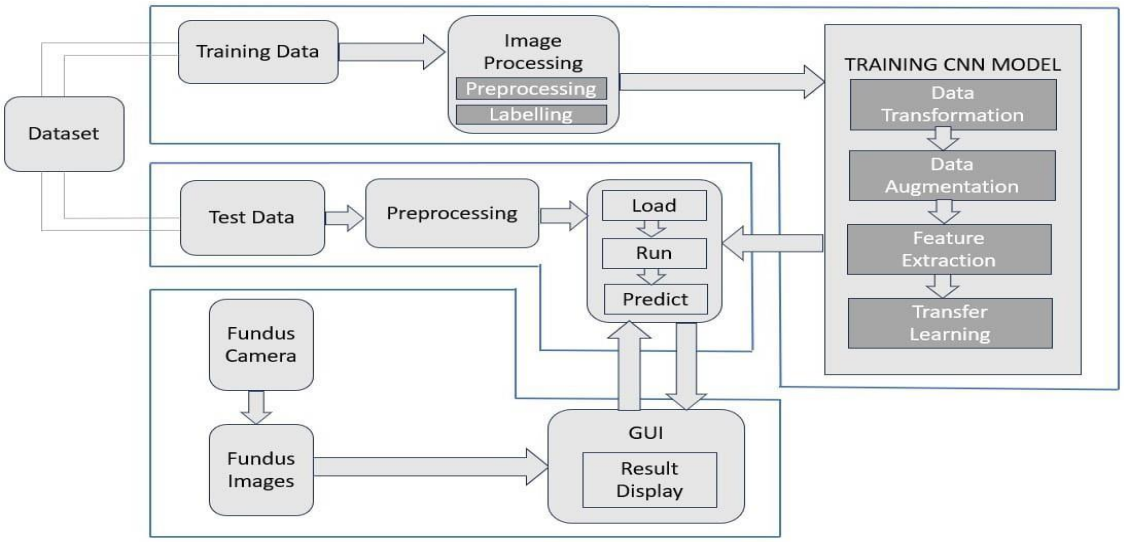


Figure 5 : Model speification

## V. EVALUATION

Evaluation metrics rely on the information provided by the confusion matrix. In assessing method performance, three key parameters are utilized: Accuracy, Precision, and Recall..

1. **Accuracy**

Accuracy is calculated by summing the number of true positives (TP) and true negatives (TN), then dividing this sum by the total number of instances. The formula for accuracy is as follows:

$$accuracy = (TP + TN)/(TP + FP + TN + FN)$$

2. **Precision Value**

Precision is determined by dividing the number of true positive predictions by the total number of positive predictions made by the classifier. The formula for precision is:

$$precision = (TP)/(TP + FP)$$

3. **Recall**

Recall measures the ability of a model to correctly I dentify positive samples. It is calculated by dividing the number of true positive predictions by the total number of actual positive instances.

$$Recall = (TP)/(TP + FN)$$

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 5, May 2025

DOI: 10.17148/IARJSET.2025.125357

## VI. RESULT

The product underwent evaluation by testing it with both images from the test dataset and those uploaded in real-time for the applicable diseases. In all cases, the output disease is predicted along with the confidence percentage. The training of the model automatically halts when there is no significant improvement in the accuracy and loss curves, based on a callback algorithm. Thus the training process stopped after 15 epochs and achieved test accuracy of 95.3%. Performance analysis parameters

```
Do you want model asks you to halt the training [y/n] ?
 n
 Epoch     Loss   Accuracy  V_loss    V_acc     LR      Next LR  Monitor  % Improv  Duration
 1 /30     5.891   82.787   3.73299  87.822   0.00100  0.00100  accuracy     0.00   1312.75
 2 /30     2.040   92.155   1.71805  89.930   0.00100  0.00100  val_loss    53.98   1262.72
 3 /30     1.013   93.326   1.12924  85.012   0.00100  0.00100  val_loss    34.27   1279.15
 4 /30     0.815   90.486   0.71615  88.993   0.00100  0.00100  val_loss    36.58   1335.61
 5 /30     0.511   94.116   0.52596  89.696   0.00100  0.00100  val_loss    26.56   1320.57
 6 /30     0.407   95.521   0.44242  92.272   0.00100  0.00100  val_loss    15.88   1322.47
 7 /30     0.358   95.814   0.48965  88.993   0.00100  0.00050  val_loss   -10.68   1303.24
 8 /30     0.303   96.897   0.38206  91.101   0.00050  0.00050  val_loss    13.64   1404.25
 9 /30     0.246   97.804   0.39001  93.208   0.00050  0.00025  val_loss    -2.08   1389.84
10 /30     0.227   98.390   0.33583  93.208   0.00025  0.00025  val_loss    12.10   1362.90
11 /30     0.196   98.858   0.35290  92.974   0.00025  0.00013  val_loss    -5.08   1285.77
12 /30     0.172   99.444   0.32637  93.911   0.00013  0.00013  val_loss     2.82   1352.01
13 /30     0.164   99.590   0.33194  94.614   0.00013  0.00006  val_loss    -1.71   1344.87
14 /30     0.164   99.297   0.33237  93.443   0.00006  0.00003  val_loss    -1.84   1340.08
15 /30     0.160   99.707   0.33032  93.911   0.00003  0.00002  val_loss    -1.21   1371.81
 training has been halted at epoch 15 after 3 adjustments of learning rate with no improvement
training elapsed time was 5.0 hours, 39.0 minutes, 52.56 seconds)
```

Figure 6: EfficientNetB3 Model Training

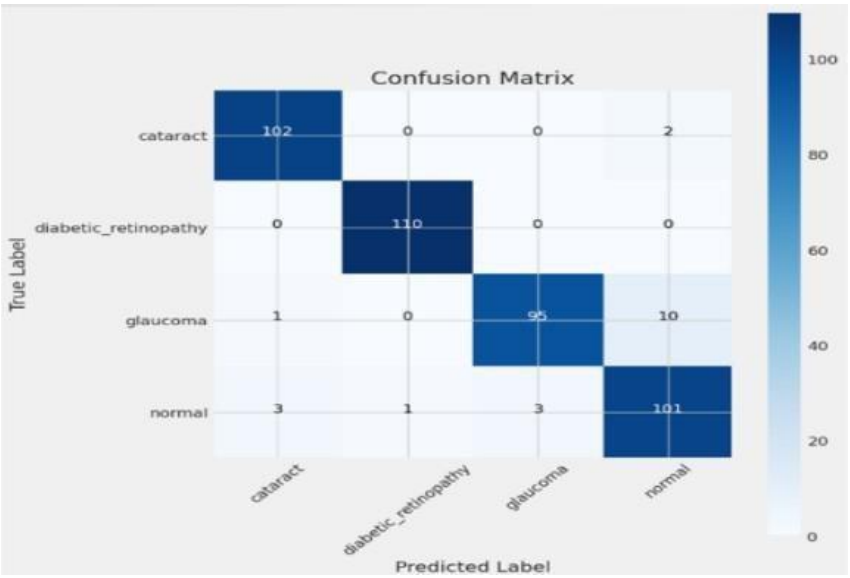


Figure 7: Confusion Matrix

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 5, May 2025

DOI: 10.17148/IARJSET.2025.125357


Figure 8: GUI showing detected eye diseases

| | Precision | Recall | F1_score | Support |
|---|---|---|---|---|
| Cataract | 0.96 | 0.98 | 0.97 | 104 |
| Diabetic Retinopathy | 0.99 | 1 | 1 | 110 |
| Glaucoma | 0.97 | 0.9 | 0.93 | 106 |
| Normal Eye | 0.89 | 0.94 | 0.91 | 108 |

Figure 9: Confusion Matrix

## VII. CONCLUSION

The project has achieved significant success by successfully deploying a multifaceted optical disease detection system catering to four distinct classes: cataract, glaucoma, diabetic retinopathy, and normal ocular conditions. Furthermore, an intuitive graphical user interface (GUI) has been developed to expedite and optimize the preliminary identification process of these eye ailments. Its utility extends to enabling expedited verification by healthcare practitioners and empowering patients with the means for early monitoring, thereby fortifying proactive healthcare management practices.

## REFERENCES

[1]. D. Shamia, S. Prince and D. Bini, ”An Online Platform for Early Eye Disease Detection using Deep Convolutional Neural Networks,” 2022 6th International Conference on Devices, Circuits and Systems (ICDCS), Coimbatore, India, 2022, pp. 388-392, doi: 10.1109/ICDCS54290.2022.9780765.

[2]. G. Ramanathan, D. Chakrabarti, A. Patil, S. Rishipathak and S. Kharche, ”Eye Disease Detection Using Machine Learning,” 2021 2nd Global Conference for Advancement in Technology (GCAT), Bangalore, India, 2021, pp. 1-5, doi: 10.1109/GCAT52182.2021.9587740.

DOI: 10.17148/IARJSET.2025.12xx

[3]. D. Helen and S. Gokila, ”EYENET: An Eye Disease Detection System using Convolutional Neural Network,” 2023 2nd International Conference on Edge Computing and Applications (ICECAA), Namakkal, India, 2023, pp. 839-842, doi: 10.1109/ICECAA58104.2023.10212139.

[4]. G. V. Datta, S. R. Kishan, A. Kartik, G. B. Sai and S. Gowtham, ”Glaucoma Disease Detection Using Deep Learning,” 2023 Fifth International Conference on Electrical, Computer and Communication Technologies (ICECCT), Erode, India, 2023, pp. 1-6, doi: 10.1109/ICECCT56650.2023.10179802.

[5]. K. Prasad, P. S. Sajith, M. Neema, L. Madhu and P. N. Priya, ”Multiple eye disease detection using Deep Neural Network,” TENCON 2019 - 2019 IEEE Region 10 Conference (TENCON), Kochi, India, 2019, pp. 2148-2153, doi: 10.1109/TENCON.2019.8929666.

[6]. K. Vayadande, V. Ingale, V. Verma, A. Yeole, S. Zawar and Z. Jamadar, ”Ocular Disease Recognition using Deep Learning,” 2022 International Conference on Signal and Information Processing (IConSIP), Pune, India, 2022, pp. 1-7, doi: 10.1109/ICoNSIP49665.2022.10007470.

[7]. S. A. Toki, S. Rahman, S. M. Billah Fahim, A. Al Mostakim and M. K. Rhaman, ”RetinalNet-500: A newly developed CNN Model for Eye Dis- ease Detection,” 2022 2nd International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC), Cairo, Egypt, 2022, pp. 459-463, doi: 10.1109/MIUCC55081.2022.9781785.

[8]. Atul and S. Dhingra, ”Classification of Diabetic Retinopathy Disease with Improved Transfer Learning Techniques using EfficientNets,” 2022 10th In- ternational Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 2022, pp. 1-5, doi: 10.1109/ICRITO56286.2022.9965087

[9]. Amin, Javeria & Sharif, Muhammad & Yasmin, Mussarat. (2016). A Review on Recent Developments for Detection of Diabetic Retinopathy. Scientifica. 2016. 1-20. 10.1155/2016/6838976.

[10]. R. S. Salvi, S. R. Labhsetwar, P. A. Kolte, V. S. Venkatesh and A. M. Baretto, ”Predictive Analysis of Diabetic Retinopathy with Transfer Learn- ing,” 2021 4th Biennial International Conference on Nascent Technologies in Engineering (ICNTE), NaviMumbai, India, 2021, pp. 1-6, doi: 10.1109/IC-NTE51185.2021.9487789

[11]. Roy, Amit & Abdullah, Riasat & Ahmed, Fahim & Mashfi, Md.Shahriar & Khan, Sazid & Karim, Dewan. (2023). RetNet: Retinal Disease Detection using Convolutional Neural Network. 1-6. 10.1109/ECCE57851.2023.10101661

[12]. W. Liu, Z. Zhao and M. Levkiv, ”Automatic Diagnosis of Diabetic Retinopathy Based on EfficientNet,” 2023 17th International Conference on the Experience of Designing and Application of CAD Systems (CADSM), Jaroslaw, Poland, 2023, pp. 64-67, doi: 10.1109/CADSM58174.2023.10076511

 