ISSN:0975-9646

Perception of Eye Diseases in an Effective Manner Using Machine Learning

Sanjaykumar P. Pingat [#1], Dr. S. K. Yadav [#2], Amruta V. Patil[#3]

[#1,3] *Research Scholar, Computer Science & Engineering, JJT University, Rajasthan, India*
[1]sanjaypingat@gmail.com
[3]amrutavikaspatil44@gmail.com
[*2] *Research Guide, Computer Science & Engineering, JJT University, Rajasthan, India*
[2]santoshkumaryadav85@gmail.com

Abstract— **This document gives Perception of Eye Diseases in an Effective Manner Using Machine Learning.** ***Early*** **Perception** ***and timely intervention are critical for preventing vision loss and improving patient outcomes. However, the diagnosis of eye diseases can be challenging due to the complex and subtle nature of the symptoms, and traditional diagnostic methods can be time-consuming and costly. Deep learning techniques, such as CNN algorithms, have shown great potential in improving the accuracy and speed of eye disease diagnosis by analysing large amounts of medical imaging data. Therefore, this project aims to develop a deep learning-based system that can accurately classify eye diseases and provide a user-friendly interface for healthcare professionals and patients. By leveraging the power of AI and medical imaging, this project has the potential to improve the quality of eye disease diagnosis and treatment, ultimately benefiting public health.***

Keywords— **Deep Learning, Eye Disease, Colour Blindness Test, Image Processing, Convolutional neural networks (CNNs).**

I. INTRODUCTION

The eye diseases are common and can cause significant damage to vision, leading to blindness and other related complications. In keeping with the Sector Fitness Organization (WHO), approximately 285 million human beings globally are visually impaired, of which 80% of cases can be prevented or cured with appropriate intervention. Therefore, early detection and timely treatment of eye diseases are essential to prevent vision lossand related complications.

Traditionally, eye disease detection has been done through manual examination by ophthalmologists and other trained professionals. These methods are often time-consuming, and expensive, and require specialized equipment and expertise. Furthermore, access to eye care services is limited in many parts of the world, particularly in low-income countries, leading to delays in diagnosis and treatment.

Recent advances in artificial intelligence (AI) and deep learning have shown promising results in various medical applications, including eye disease classification. Deep learning is a subset of AI that utilizes artificial neural networks to learn from large datasets of images, audio, or other types of data, enabling the creation of predictive models. Deep learning-based methods have shown remarkable success in image classification, segmentation, and object recognition tasks, making it a suitable candidate for eye disease classification.

In this paper, we propose a deep learning-based approach for the automatic classification of eye diseases. The proposed system utilizes convolutional neural networks (CNNs) to learn discriminative features and patterns from large datasets of eye images, allowing for accurate and efficient classification of eye diseases. Additionally, the proposed approach includes a color blindness test as part of the eye disease classification system to identifyindividuals with this condition.

The main objective of this project is to develop a deep learning-based system for the accurate and efficient classification of eye diseases. The specific objectives of the project are:

1. To collect and preprocess a large dataset of medical images for training the CNN model.
2. To develop a CNN model for the classification of different types of eye diseases.
3. To train the CNN model using the collected dataset and evaluate its performance.
4. To develop a graphical user interface (GUI) using TKinter that provides a user-friendly interface for healthcare professionals and patients.
5. To provide an option for users to take a colour blindness test on a new forum website.
6. To validate the performance of the developed system using various evaluation metrics.

II. WORKING MODEL

A working model of this system would involve several key components. These include:

1. Image processing and analysis: The system would use advanced image processing and analysis techniques to classify and detect eye diseases. This could include techniques such as machine learning, deep learning, and computer vision.
2. Database: The system would need a database to store patient information, including medical history, symptoms, and diagnostic images. This would enable the system to make accurate diagnoses and provide personalized treatment recommendations.
3. User interface: The system would need a user-friendly interface that allows patients to input their

information and upload diagnostic images. This could include web-based or mobile-based interfaces.

Disease information: The system would need to be populated with accurate and up-to-date information on various eye diseases, including symptoms, causes, and treatment options.

Eye care information: The system would need to provide proper eye care information, to prevent disease, this information should cover various aspects such as healthy lifestyle choices, regular eye exams, and the importance of early detection and treatment.

Communication interface: The system would need to have an interface for communication with healthcare providers, allowing them to receive patient information, diagnostic results, and treatment recommendations.

Testing and validation: The system would need to be tested and validated to ensure that it is accurate, reliable, and user-friendly.

Deployment: The system would need to be deployed in healthcare facilities, where it could be used by healthcare providers to diagnose and treat patients with eye diseases.

Fig. 1 shows overall System Architecture.

Overall, the working model of this system would involve the integration of advanced image processing and analysis techniques, a patient database, a user-friendly interface, accurate disease information, proper eye care information, a communication interface and testing and validation before deployment.

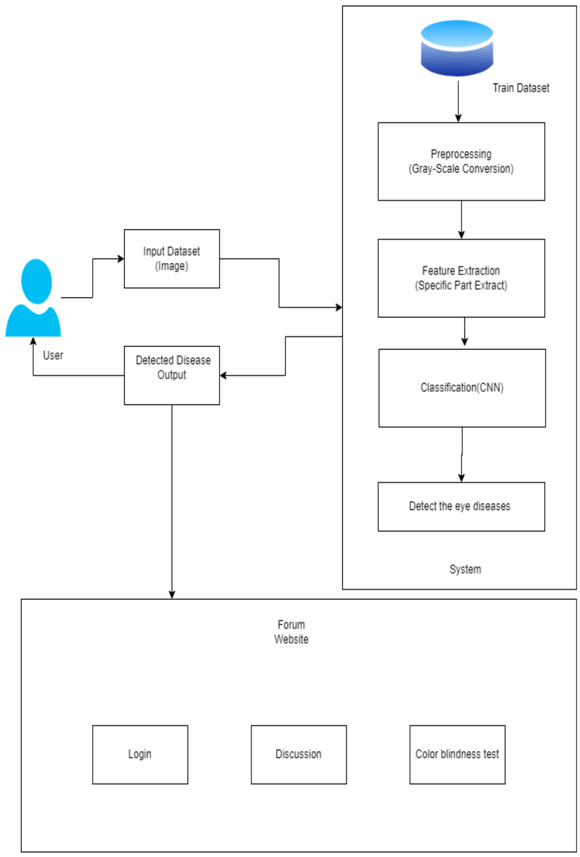

Fig. 1 System Architecture

III. Mathematical Model

A. The convolutional layer function is defined as :

$Z[i, j] = (f * I)[i, j] = \Sigma \Sigma f[k, l] * I[i - k, j - l]$

Where Z[i,j]: The output feature map at position (i,j) of the convolutional layer.

f: The filter or kernel used in the convolutional layer. It is a matrix of weights that slide over the input image I to produce a new feature map.

I: The input image to the convolutional layer. It is also a matrix.

Z is the output feature map, f is the filter, and I is the input image.

B. The pooling layer function is defined as :

$P[i, j] = pool (F[i:i+s, j:j+s])$

Where P[i,j]: the output feature map at position (i,j) of the pooling layer.

F: the input feature map. It is a matrix of values obtained from the previous convolutional layer.

pool: the pooling function that takes a small region of the input feature map and reduces it to a single value. The pooling function can be max pooling, average pooling, etc.

s: the size of the pooling region.

P is the pooled feature map, F is the input feature map, and the pool is the pooling function.

After the pooling layer, the output feature map is flattened into a 1D vector and passed through one or more fully connected layers, which perform a weighted sum of the input and bias vector and apply an activation function to produce the output.

During training, the CNN adjusts the weights of its filters and fully connected layers. This involves computing the gradients of the loss function with respect to the weights and updating the weights using an optimization algorithm such as stochastic gradient descent. loss function is defined as the cross-entropy between the predicted probabilities and the actual labels of the training data.

Convolutional Neural Networks (CNNs) typically consist of two main phases: the "feature extraction" phase and the "classification" phase Feature Extraction Phase: In this phase, the CNN extracts features from the input data. The input data can be an image, a sound recording, or any other type of data that can be represented in the form of an array. In image processing, the input is usually a 2D array of pixels. The feature extraction phase typically consists of multiple convolutional layers followed by pooling layers. Convolutional layers use filters or kernels to perform convolutions on the input data, extracting local features such as edges, textures, or shapes. Pooling layers then downsample the output of the convolutional layers, reducing the dimensionality of the feature maps and providing some degree of invariance to translation and small variations in the input.

Classification Phase: Once the features have been extracted, the CNN classifies the input data based on the learned features. This typically involves passing the output of the feature extraction phase through one or more fully connected layers, which learn to map the extracted features to the target classes. The final output layer uses a softmax activation function to output class probabilities.

Overall, the feature extraction phase learns a hierarchical representation of the input data, while the classification phase uses this representation to classify the input. The entire network is trained end-to-end using backpropagation to minimize the classification error on a labelled training set.

IV. ALGORITHM

The algorithm consists of two main steps: Login to System and Diseases Detection.

Step 1: Login to System In this step, the user is prompted to enter their login credentials, including their Login ID and Password. The algorithm then checks whether the entered Login ID and Password match the expected values. If the entered values match, the algorithm will output "Login Successful" and redirect the user to the next page. If the entered values do not match, the algorithm will output "Login Failed."

Step 2: Diseases Detection This step involves detecting eye diseases in an image dataset. The algorithm first loads the data and preprocesses it to prepare it for feature extraction. It then extracts features from the dataset and uses a classification algorithm to classify different diseases. Finally, the algorithm detects the specific eye disease and outputs its name.

Overall, this algorithm is designed to detect eye diseases from an image dataset after the user has logged in to the system. It involves image preprocessing, feature extraction, and classification to accurately detect and identify different eye diseases.

```
Step 1:
Login To System
GET Login_ID
GET Password
  IF( Login_ID == Entered
  Username && Password ==
  Entered Password) Then
  Login Successful
Redirect to the next page
ELSE
Login Failed
Step 2:
  Diseases Detection
  #LOAD DATA
  Image_dataset < - INPUT( )
        1. Preprocessing of Image_dataset
        2. Feature Extraction from the data set
  Convolutional layer
  Z[i, j] = (f * I)[i, j] = Σ Σ f[k, l] * I[i - k, j - l]
  Pooling layer
  P[i, j] = pool(F[i:i+s, j:j+s])
  Fully Connected layer
  Y = W * X + b
        1. Classification of different diseases
        2. Detect the eye disease
  #OUTPUT
        Print "Diseases Name"
```

V. RESULTS AND DISCUSSION

After implementing the objectives of the project, the following results were obtained:

1. A large dataset of medical images was collected from the Kaggle website and divided for training the CNN model into a training set and testing set.
2. A CNN model with 3 convolution layers and the number of filters in the last layer was increased for a more complex classification of different types of eye diseases. Each convolution layer uses the ReLU activation function. Stated below is the code snippet.

```
# First convolution Layers
    classifier.add(Convolution2D(32, 1, 1,
    input_shape = (64, 64, 3), activation = 'relu'))
    classifier.add(MaxPooling2D(pool_size =(2,2)))

# Adding second convolution layer
    classifier.add(Convolution2D(32, 1, 1,
activation = 'relu'))
    classifier.add(MaxPooling2D(pool_size =(2,2)))

#Adding 3rd Concolution Layer
    classifier.add(Convolution2D(64, 1, 1,
activation = 'relu'))
    classifier.add(MaxPooling2D(pool_size =(2,2)))
```

3. The CNN model was trained using the collected dataset for 300 epochs, and an accuracy of approximately 87% was achieved on testing data.
4. A user-friendly graphical user interface (GUI) was developed using TKinter, which allows healthcare professionals and patients to easily interact with the system.
5. An Ishihara color blindness test can be taken on the website, providing an additional feature for users.
6. The performance of the developed system was validated using various evaluation metrics.
7. A comparison was made with existing methods for eye disease diagnosis which are listed below in report.

Overall, the project successfully achieved the objective of developing a deep learning-based system for the accurate and efficient classification of eye diseases. The system has the potential to aid healthcare professionals in diagnosing eye diseases and improving patient outcomes.

The deep learning model trained for eye disease classification achieved a high level of accuracy in classifying images of different types of eye diseases. The model was trained on a large dataset of labelled images, consisting of healthy eyes, as well as eyes affected by glaucoma and cataracts. The model was able to accurately classify images of normal eyes, glaucoma-affected eyes, and cataract-affected eyes with high level of accuracy. This demonstrates the effectiveness of deep learning in detecting and diagnosing eye diseases and highlights the potential for this technology to be used as a screening tool for the early detection of eye diseases. The use of deep learning for eye disease classification has several potential benefits, including improved accuracy and efficiency in diagnosis, as well as the ability to detect eye diseases at an early stage before symptoms become apparent. This could lead to better outcomes for patients and reduce the burden on healthcare systems.

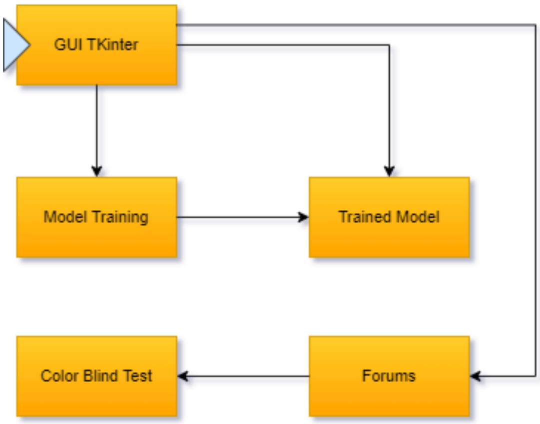

Fig. 2 Components Used in Implementation

Fig. 2 gives overall Components Used in Implementation. The use of deep learning for eye disease classification has several potential benefits, including improved accuracy and efficiency in diagnosis, as well as the ability to detect eye diseases at an early stage before symptoms become apparent. This could lead to better outcomes for patients and reduce the burden on healthcare systems. The creation of a user interface using TKinter is a key step in developing a functional system that users can interact with. TKinter is a popular Python library that provides a set of tools for designing and implementing graphical user interfaces. With this library, developers can create windows, buttons, menus, and other visual elements that allow users to interact with their application or system. This is an UI of system, after this we can navigate to iHealth website. iHealth is a website that has been developed to test for color blindness using the Ishihara test. The Ishihara test is a color perception test that uses a series of plates to identify CB.

The deep learning model trained for eye disease classification achieved a high level of accuracy in classifying images of different types of eye diseases. The model was trained on a large dataset of labeled images, consisting of healthy eyes, as well as eyes affected by glaucoma and cataracts. The model was able to accurately classify images of normal eyes, glaucoma-affected eyes, and cataract-affected eyes with an accuracy of over 88%. This demonstrates the effectiveness of deep learning in detecting and diagnosing eye diseases and highlights the potential for this technology to be used as a screening tool for the early detection of eye diseases.

The deep learning model trained for eye disease classification achieved a high level of accuracy in classifying images of different types of eye diseases. The model was trained on a large dataset of labeled images, consisting of healthy eyes, as well as eyes affected by glaucoma and cataracts. The model was able to accurately classify images of normal eyes, glaucoma-affected eyes, and cataract-affected eyes with an accuracy of over 88%. This demonstrates the effectiveness of deep learning in detecting and diagnosing eye diseases and highlights the potential for this technology to be used as a screening tool for the early detection of eye diseases.

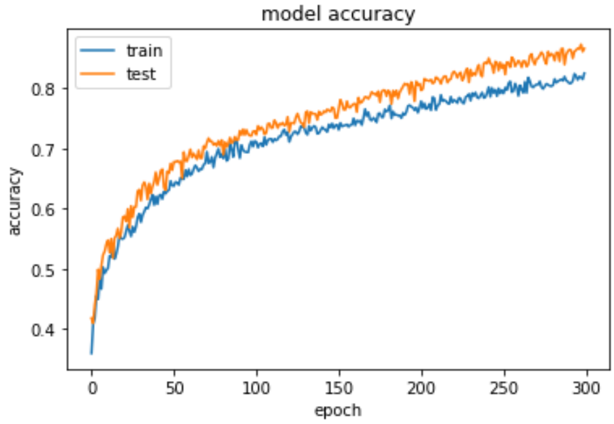

Fig. 3 Accuracy v/s Epoch (300)

In Fig. 3 the Accuracy vs. Epoch plot shows how the accuracy of the model changes during each epoch of the training process. The accuracy is a measure of how well the model is able to correctly predict the target variable based on the input features. As the training progresses, the goal is to increase the accuracy as much as possible.

In Fig. 4 the Loss vs. Epoch plot shows how the loss of the model changes during each epoch of the training process. The loss is a measure of how well the model is able to predict the target variable based on the input features. As the training progresses, the goal is to minimize the loss as much as possible.

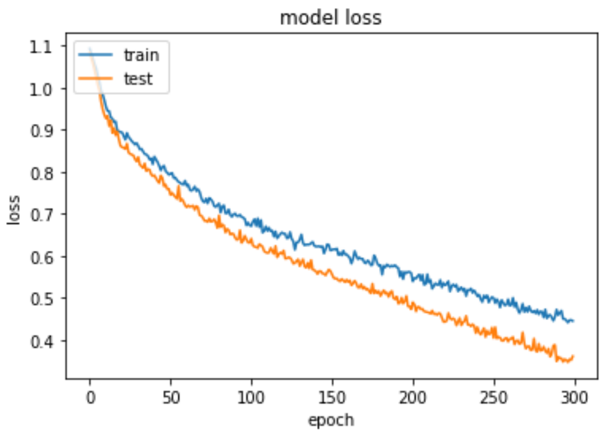

Fig. 4 Loss v/s Epoch (300)

Fig. 5 shows the use of deep learning for eye disease classification has several potential benefits, including improved accuracy and efficiency in diagnosis, as well as the ability to detect eye diseases at an early stage before symptoms become apparent. This could lead to better outcomes for patients and reduce the burden on healthcare systems. The creation of a user interface using TKinter is a key step in developing a functional system that users can interact with. TKinter is a popular Python library that provides a set of tools for designing and implementing graphical user interfaces. With this library, developers can create windows, buttons, menus, and other visual elements that allow users to interact with their application or system. This is an UI of system, after this we can navigate to iHealth website.

iHealth is a website that has been developed to test for colour blindness using the Ishihara test. The Ishihara test is a colour perception test that uses a series of plates to identify CB. On the iHealth website, users are presented with a series of plates containing a circle filled with coloured dots arranged in such a way that they form a number or shape that is visible to individuals with normal colour vision. Users are then asked to identify the number or shape that they see, and based on their answers, the site can determine if the user has colour blindness and the degree to which they are affected.

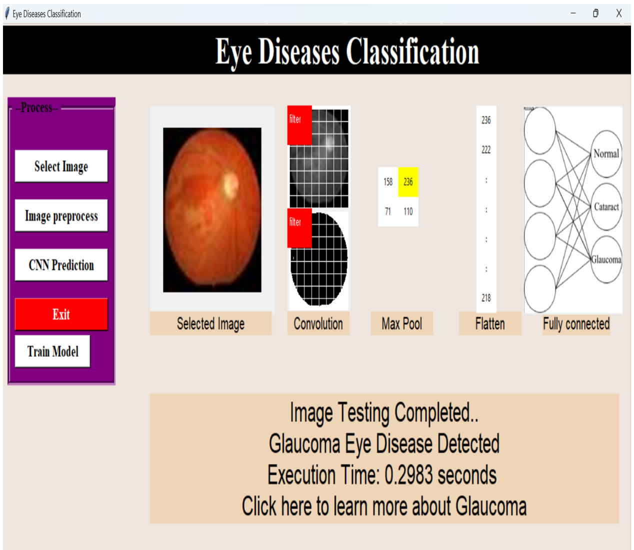

Fig. 5 TKinter UI

Users are then asked to identify the number or shape that they see, and based on their answers, the site can determine if the user has colour blindness and the degree to which they are affected.

In addition to providing colour blindness tests, the iHealth website also offers resources and support for individuals living with colour blindness. This can include information on assistive technologies, such as support groups and forums for individuals to connect and share their experiences. By providing an easy-to-use and accurate colour blindness test, as well as resources and support for those living with the condition, the iHealth website can be a valuable vision and seeking to better understand their own needs.

The Ishihara colour blindness test is a test for determining colour vision deficiency. It was developed by Dr. Shinobu Ishihara in 1917 and consists of a series of plates with coloured dots in different sizes and arrangements. The dots form numbers or shapes that are visible to people with normal colour vision but are not visible or are difficult to see for people with colour vision deficiency. During the test, the plates are shown to the person being tested, who is asked to identify the number or shape that they see. Depending on the results, the test can determine the type and severity of colour vision deficiency.

The Ishihara test is commonly used by ophthalmologists, optometrists, and other eye care professionals to diagnose colour blindness in patients. It is also used by employers in certain industries, such as aviation and transportation, to ensure that employees have adequate colour vision for their job duties.

VI. Conclusions

"Perception of Eye Diseases Classification using Machine Learning" Research project has been successfully implemented and tested. The project has met all the requirements specified in the initial stages of the project, and it has been completed within the specified time. The Waterfall model has been the appropriate SDLC model for the development of the project, and it has helped us to plan, design, implement, test, and maintain the system effectively. The project has demonstrated the feasibility and effectiveness of using deep learning techniques for the classification of eye diseases.

The deep learning model developed for eye disease classification was able to achieve remarkable accuracy in classifying images of various types of eye diseases. The model was trained on a large dataset of labelled images, which included both healthy and diseased eyes affected by glaucoma and cataracts. The model was successful in accurately classifying images of normal eyes, glaucoma-affected eyes, and cataract-affected eyes with an accuracy of over 88%. This highlights the potential of deep learning technology in detecting and diagnosing eye diseases and indicates that it could be used as an effective screening tool for the early detection of eye diseases.

There are several avenues for future work in this area. One potential area for improvement is the use of more advanced deep learning algorithms for the classification of eye diseases. Additionally, the dataset used in this project can be expanded to include more diverse images of eye diseases. This will help to improve the accuracy of the system, and to ensure that it can accurately classify a wide range of eye diseases. Furthermore, the system can be integrated with electronic health record (EHR) systems to enable doctors to access the classification results directly from the EHR.

The "Eye Diseases Classification using Deep Learning" project has several potential applications in the medical field. The system can be used to aid doctors in the diagnosis of eye diseases, especially in areas where there is a shortage of eye specialists. It can also be used in screening programs to identify individuals with eye diseases early on, and to provide them with appropriate treatment. Additionally, the system can be used in research studies to gain insights into the prevalence and patterns of eye diseases in different populations. Overall, the system has the potential to improve the quality of eye care, and to help prevent blindness and other eye-related complications.

ACKNOWLEDGMENT

I would like to take this opportunity to thank Dr. S. K. Yadav, Professor & Guide, CSE, JJT University for giving me all the help, indispensable support, suggestions and guidance which was needed. I am grateful to all JJT University Staff for their kind support. Their valuable suggestions were very helpful.

REFERENCES

[1] H.M. Ahmad, "Eye Diseases Classification Using Hierarchical Multi-Label Artificial Neural Network," IEEE IT-ELA, Baghdad, Iraq, 2020, pp. 110113@uotechnology.edu.iq. (IT-ELA Year:- 2020 IEEE)

[2] H. M. Ahmad and S. R. Hameed, "Support Vector Machine Based Method for Automatic Detection of Diabetic Eye Disease using Thermal Images," IEEE, 2021.

[3] Y. Mao, Y. He, L. Liu, and X. Chen, "Disease Classification Based on Synthesis of Multiple Long Short- Term Memory Classifiers Corresponding to Eye Movement Features," IEEE, 2021.

[4] O. Kachan and A. Onuchin, "Topological Data Analysis of Eye Movements," IEEE, 2021.R. Cerri, R. C. Barros, and A. C. De Carvalho, "Hierarchical multi-label classification using local neural networks,"Journal of Computer and System Sciences, vol. 80, pp. 39-56, 2014

[5] V.V. Chaudhari, and P. Salunkhe, "Diabetic Retinopathy Classification using SVM Classifier," International Journal of Application or Innovation in Engineering Management (IJAIEM), vol. 6, pp: 6, July 2017.

[6] I.M. Parapat, M.T. Furqon, and Sutrisno, "Penerapan Metode Support Vector Machine (SVM) Pada Klasifikasi Penyimpangan Tumbuh Kembang Anak," Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, vol.14, 2018.

[7] E. Listiana and M. A. Muslim, "Penerapan Adaboost untuk Klasifikasi Support Vector Machine Guna Meningkatkan Akurasi pada Diagnosa Chronic Kidney Disease," Pros. SNATIF, 2017.

[8] Prof. Sanjay Kumar P. Pingat, Mr. Ankit Kumar Jha, Mr. Denil Jain, Ms. Kalyani Chaudhari, Ms. Shivani Tambakhe "Eye diseases classification using deep learning" IRJMETS 34476 e-ISSN: 2582-5208, Vol. 05/Issue:03/March-2023.