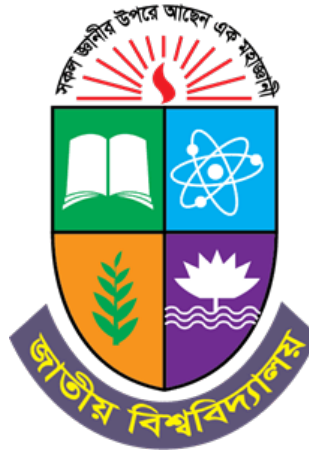# Smart Eye Checkup using Machine Learning

A Project presented to the National University in partial fulfillment of the requirement for the degree of Bachelor of Science (Hon's) in Computer Science & Engineering

Submitted By

Bepul Saha

Registration No: 17502004913

Session: 2017-2018

Department of Computer Science & Engineering

Daffodil Institute of IT, Dhaka

Under National University, Bangladesh

Date of Submission: September, 2023

DECLARATION

We pledge that the project work titled "Smart Eye Checkup" being submitted in partial fulfillment for the degree of B.Sc. (Hon's) in Computer Science Engineering is the original work carried out by me. It has not formed part of any other project work submitted for any degree or diploma, either in this or any other University.

_____

Bepul Saha

Reg: 17502004913

APPROVAL

The Project "Smart Eye Checkup" is submitted to the De- partment of Computer Science Engineering, DIIT under National University of Bangladesh in absolute fulfillment of the requirements for the degree of Bachelor of Science (Hon's) in Computer Science and Engineering and approved as to its style and content.

<div style="display:flex; justify-content:space-between;">

_____   _____

Examiner   Examiner

_____   _____

Project Supervisor   Head of Department

Poly Bhoumik   **Md. Imran Hossain**

Senior Lecturer, Department of CSE   Head of Department, CSE

Daffodil Institute of IT   Daffodil Institute of IT

</div>

ACKNOWLEDGEMENT

I would like to express my profound gratitude to Almighty. With the blessing of Almighty Allah I have successfully planned my project.

My sincere thanks to **Professor Dr. Mohammed Shakhawat Hossain**, Principal, DIIT who has allowed me to do this project and encouragement given to me. Also, thanks for his valuable guidance and support to meet the successful completion of my project.

My heartiest thanks **Poly Bhoumik**, Supervisor, Senior Lecturer, DIIT, Dhaka, for his patronage and for giving me an opportunity to undertake this Project.

I express my gratitude to **Md. Imran Hossain**, Senior Lecturer, Head of Departmrnt, DIIT, Dhaka, for having provided us with the facilities to do the project successfully.

I express my gratitude to **Saidur Rahman**, Senior Lecturer, DIIT, Dhaka, for having provided us with the facilities to do the project successfully.

I express my gratitude to **Safrun Nesa Saira**, Lecturer, DIIT, Dhaka, for having pro- vided us with the facilities to do the project successfully.

I express my gratitude to **Nushrat Jahan Sarkar**, Lecturer, DIIT, Dhaka, for having provided us with the facilities to do the project successfully.

I express my gratitude to **Mizanur Rahman**, Lecturer, DIIT, Dhaka, for having provided us with the facilities to do the project successfully.

I express my gratitude to **Moumita Akter**, Lecturer, DIIT, Dhaka, for having provided us with the facilities to do the project successfully.

I express my gratitude to **Md. Mushfiqur Rahman**, Lecturer, DIIT, Dhaka, for having provided us with the facilities to do the project successfully.

Last but not the least, I extend my sincere thanks to my family members and my friends for their constant support throughout this project.

ABSTRACT

Many eye diseases like cataracts, trachoma, or cornealulcer can cause vision problems. Progression of these eye diseases can only be prevented if they are recognized accurately at the early stage. Visually observable symptoms differ a lot among these eye diseases. However, a wide variety of symptoms is necessary to be analyzed for the accurate detection of eye diseases. In this paper, we propose a novel approach to provide an automated eye disease recognition system using visually observable symptoms applying digital image processing techniques and machine learning techniques such as deep convolution neural network (DCNN) and support vector ma- chine (SVM). We apply the principal component analysis and distributed stochastic neighbor embedding methods for better feature selection. The proposed system automatically divides the facial components from the frontal facial image and extracts the eye part.

Table of Contents

1. Introduction

2. Background Analysis

3. Hardware Analogy & Software

4. Algorithm & Design

5. Implementation

6. Result Analysis

7. Conclusion

8. References

List of Figures

Chapter 1
Introduction

1.1 Motivation

Eyes are an essential part of human life, each and every person relies on the eyes to see and sense the world around them. One of the most vital senses is sight because it accounts for 80% of all information, we take in. By taking proper care of our eyes, we will lower the risk of becoming blind and losing vision, while also keeping an eye out for any developing eye conditions like glaucoma and cataracts. Most people experience eye issues at some point in time. Some of the eye issues are minor and simple to cure at home which will go away on their own, other major eye issues need assistance from expert doctors. When these eye diseases are accurately diagnosed at an early stage, only then the progression of these eye diseases can be stopped. These eye diseases have a wide range of visually discernible symptoms. To accurately diagnose eye illnesses, it is required to analyze a wide range of symptoms. In this paper, our proposed model analyses and classifies eye diseases namely cataracts, crossed eyes, bulging eyes, uveitis, and conjunctivitis.[1]

1.2 Proposed System

In this paper, an automated eye disease recognition system is designed using various machine learning techniques. In our eye disease recognition system, the human eye part is obtained from the facial image automatically. The first stage is the capturing of image. This procedure is acquired by digital camera. Initially, the original image has been loaded as an input image and then the method detects the face from the input image by our algorithm. Our method scales the facial images at 500 × 500 pixels. Then the method segments the various facial components.[2] Once the eye part is segmented from the face part, we apply the eye parts of the image for learning. The phases of the proposed method are shown in Figure 1.
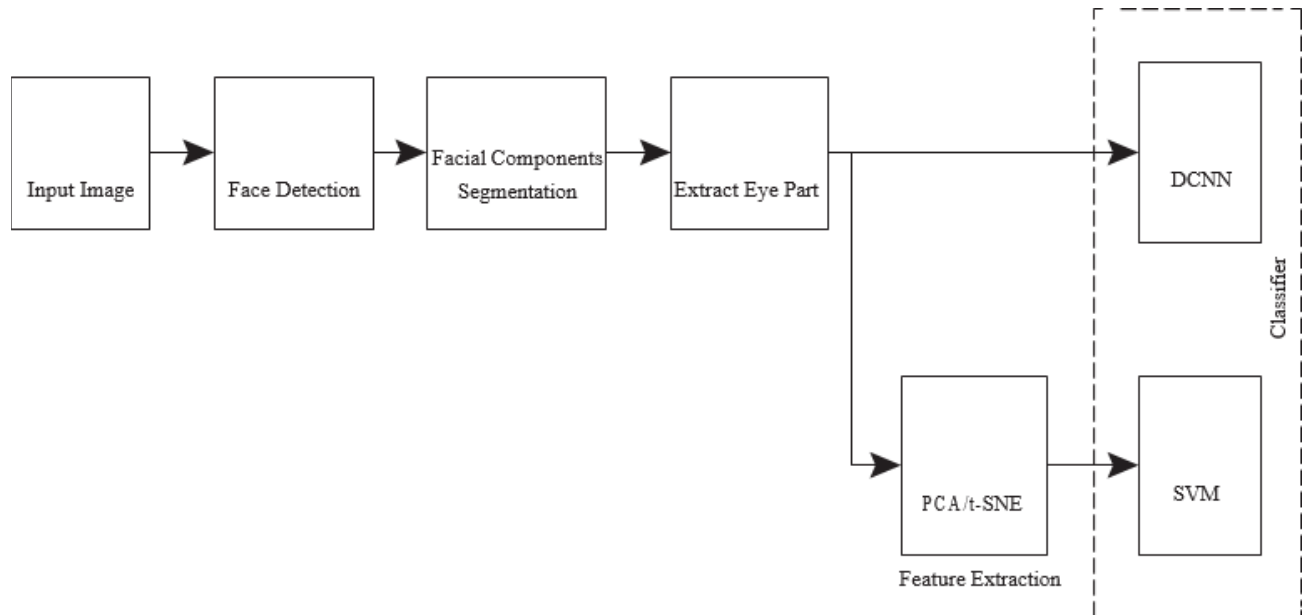

Figure 1.1: Proposed System

1.3   Methodology

1.3.1 Decision Tree Classifier

Several widely used techniques include decision tree classifier multilayer perceptron and Na¨ıve Bayes classifier.The output of a C4.5 decision tree classifier is structural data in the form of a binary tree. A C4.5 tree is modeled as follows. A training set is a set of base tuples to determine classes related to these tuples. A tuple X is represented by an adjective vector $X = (x1, x2, . . . , xn)$. Assume that a tuple belongs to a predefined class that is determined by an adjective called a class label. The training set is randomly selected from the base; this step is called the learning step. This technique is very efficient and extensively uses classification. The structure of the tree can be implemented with the following factors: 1. A node of the tree represents a test on an adjective; 2. A branch exiting from a node represents possible outputs of a test; 3. A leaf represents a class label. A decision tree includes a rule set by which objective functions can be predicted. The algorithm used for this model uses greedy techniques.[3]

1.3.2 Naïve Bayes Classifier

In machine learning, naive Bayes classifiers are a family of simple "probabilistic classifiers" based on applying Bayes' theorem with strong (naive) independence as- sumptions between the features. Naive Bayes has been studied extensively since the 1950s. It was introduced under a different name into the text retrieval community in the early 1960s and remains a popular (baseline) method for text categorization, the problem of judging documents as belonging to one category or the other (such as spam or legitimate, sports or politics, etc.) with word frequencies as the fea- tures. With appropriate pre- processing, it is competitive in this domain with more advanced methods including support vector machines.t also finds application in au- tomatic medical diagnosis.[4]

1.3.3 Neural Network

A neural network is a method in artificial intelligence that teaches computers to process data in a way that is inspired by the human brain. It is a type of machine learning process, called deep learning, that uses interconnected nodes or neurons in a layered structure that resembles the human brain.[5]

1.3.4 Random Forest Tree

Random forest is a flexible, easy-to-use machine learning algorithm that produces, even without hyper-parameter tuning, a great result most of the time. It is also one of the most-used algorithms, due to its simplicity and diversity (it can be used for both classification and regression tasks).[5]

1.4   Background of CNN

Convolutional networks are a type of sophisticated neural network. It was calculated using deep learning. The way this computation is done is that the model can initially take an input image and then assign significance to various arguments or points of view in that image so that the machine can choose to divide one class from the other. Prepossessing is this model's main requirement. CNN's layout includes network representations of neurons found in the human brain. Additionally, 2D structures of

information pictures have specific desirable positions. Here, slope-based augmentation is used. There are several layers in this model, including convolutional and subsampling layers.[6]

1.5 Advantages

• corneal topography.

• the interpretation of fundus photography and OCT scans.

• IOL power prediction.

• screening for and diagnosing eye problems such as Cataract

• predicting the speed of progression of diseases such as macular degeneration and glaucoma.

• predicting treatment outcomes.

• predicting the likelihood of needing treatment in the future.

At a more general level, AI may:

• helps us capture photographs and scans more accurately.

• act as an automatic scribe.

• predict future medical crises that a patient may face.

• improve patient flow.

• analyze and sort data so that the physician has to view only the most relevant information.

• help to discover new associations between diseases and the characteristics of a patient's eye

1.6 Disadvantages

• AI will miss things it's not looking for, giving patients a false sense of security.

• A machine will miss things that a human reader would notice.

• AI could result in people losing their jobs as machines do those same jobs faster.

• AI could cause surgeons' skills to atrophy as machines take over more respon- sibilities.

• Testing accuracy on similar data sets hides potential problems.

• You can't fix a problem you don't know about.

• A machine learning system will only be reliable on a population exactly like the one it learned.

• When the input to the system evolves, the learning has to be redone.

Chapter 2
Background Analysis

2.1 Related Work

Papers relating to the prediction of eye diseases using artificial intelligence and machine learning are reviewed in this section since these areas discuss the tools that are used for the proposed method for efficiently predicting the five eye diseases discussed above. Some papers discussing explainable artificial intelligence methods is also included since some results from these papers are used for a better interpretation of the results from the proposed method. The few works that have been performed in the area of eye disease prediction using symptomatic data are also considered here.[7]

One study assessed the prevalence of eye diseases in a certain low-income community in Bangladesh. The study was performed using in-person surveys and the results showed a higher prevalence of eye diseases than the world average. This reflects the difference in eye disease prevalence and diagnosis between individuals in low-income individuals and individuals with higher standards of living. The study found that there is a funding and communication gap between vision-related support for slum dwellers of Bangladesh and the Bangladesh community in general.[8]

2.2 Existing System

In this section, a brief overview of the five eye diseases that have been worked on is presented. For each disease, it describes the disease conditions, disease symptoms, and disease risk factors. The images of the selected five categories of eye diseases are shown in Figure 1, however, it has not used image data to identify or annotate the dataset. Therefore, the sources or the images are referenced in the footnotes. The next five subsections describe the diseases worked on.[9]
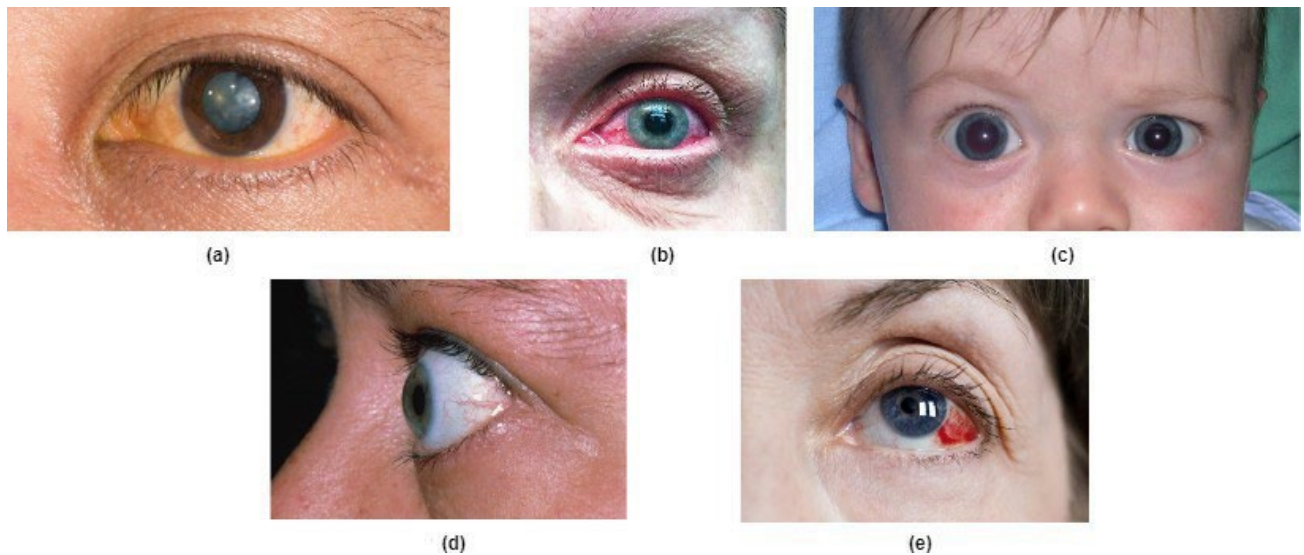

Figure 2.1: Typical images of eye diseases. (a) Cataracts, (b) Acute angle-closure glaucoma, (c) Primary congenital glaucoma, (d) Exophthalmos or bulging eyes, and (e) Ocular hypertension.

2.3 Cataracts

A dense and cloudy area that is usually formed on the lens of the eye is called a cataract. This cataract is an agglomeration of proteins situated in the eye that forms a lump-like mass which prevents the normal transmission of light through the lens to the retina. Some of the major symptoms appearing due to cataracts include degradation of night vision, blurry vision, faded colors, and double vision. The different sub-types of cataracts are nuclear, cortical, posterior, congenital, senile, traumatic, bilateral as well as other subtypes. These sub-types are grouped into one disease category for this study to minimize the complexity of associating the cataract with a specific sub-type. Smoking, older age, obesity, increased blood pressure, history of the same disease in family members, diabetes, and exposure to radiation from X-rays and cancer treatments are the main risk factors associated with cataracts. Cataracts are one of the most common eye diseases in low and middle-income countries.
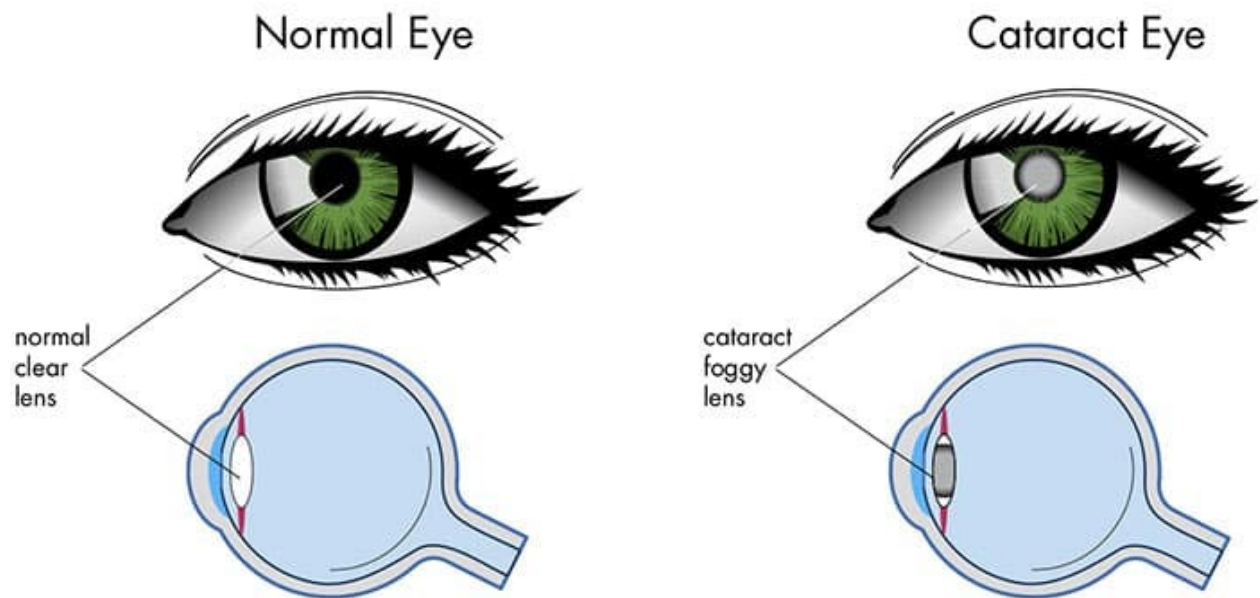
Figure 2.2: Cataract

Chapter 3
Hardware Analogy & Software

3.1 Hardware Specification

1. Intel Core i5

2. 8GB of RAM

3. Integrated GPU (2GB of VRAM)

4. 256GB SSD

3.1.1 Processor

A processor (CPU) is the logic circuitry that responds to and processes the basic instructions that drive a computer.

3.1.2 RAM

Random-access memory is a form of computer memory that can be read and changed in any order, typically used to store working data and machine code.

3.1.3 GPU

A graphics processing unit (GPU) is a specialized electronic circuit initially designed to accelerate computer graphics and image processing (either on a video card or embedded on the motherboards, mobile phones, personal computers, workstations, and game consoles).

3.1.4 Storage

Computer data storage is a technology consisting of computer components and recording media that are used to retain digital data.

3.1.5 Camera

A camera is an optical instrument used to capture and store images or videos, either digitally via an electronic image sensor, or chemically via a light-sensitive material such as photographic film. As a pivotal technology in the fields of photography and videography, cameras have played a significant role in the progression of visual arts, media, entertainment, surveillance, and scientific research.

3.2 Softwares

3.2.1 Python

Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation.

Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly procedural), object-oriented and functional programming. It is often described as a "batteries included" language due to its comprehensive standard library.

3.2.2 NumPy

NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms basic linear algebra, basic statistical operations, random simulation and much more.

3.2.3 Pandas

Pandas is an open-source Python package that is most widely used for data science/data analysis and machine learning tasks. It is built on top of another package named Numpy, which provides support for multi-dimensional arrays. As one of the most popular data- wrangling packages, Pandas works well with many other data science modules inside the Python ecosystem and is typically included in every Python distribution, from those that come with your operating system to commercial vendor distributions like ActiveState's ActivePython.

3.2.4 Open CV

Initially developed by Intel, OpenCV (Open Source Computer Vision) is a free cross-platform computer vision library for real-time image processing. The OpenCV software has become a de-facto standard tool for all things related to Computer Vision. In 2023, OpenCV is still highly popular, with over 29'000 downloads every week.

3.2.5 Scikit Learning

Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python.

3.2.6 Tensor Flow

TensorFlow is a free and open-source software library for machine learning and artificial intelligence. It can be used across a range of tasks but has a particular focus on training and inference of deep neural networks.

3.2.7 Node JS

Node.js is a cross-platform, open-source server environment that can run on Windows, Linux, Unix, macOS, and more. Node.js is a back-end JavaScript runtime environment, runs on the V8 JavaScript engine, and executes JavaScript code outside a web browser.

3.2.8 Jupyter Notebook

The Jupyter Notebook is an open source web application that you can use to create and share documents that contain live code, equations, visualizations, and text. Jupyter Notebook is maintained by the people at Project Jupyter.

3.2.9 Django

Django is a high-level Python web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel. It's free and open source.

Chapter 4
Algorithm & Design

4.1 Algorithm

Step 1: Choose a Dataset

Step 2: Prepare Dataset for Training

Step 3:Create Training Data

Step 4: Shuffle the Dataset

Step 5: Assigning Labels and Features

Step 6: Normalising and converting labels to categorical data

Step 7: Define, compile and train the CNN Model

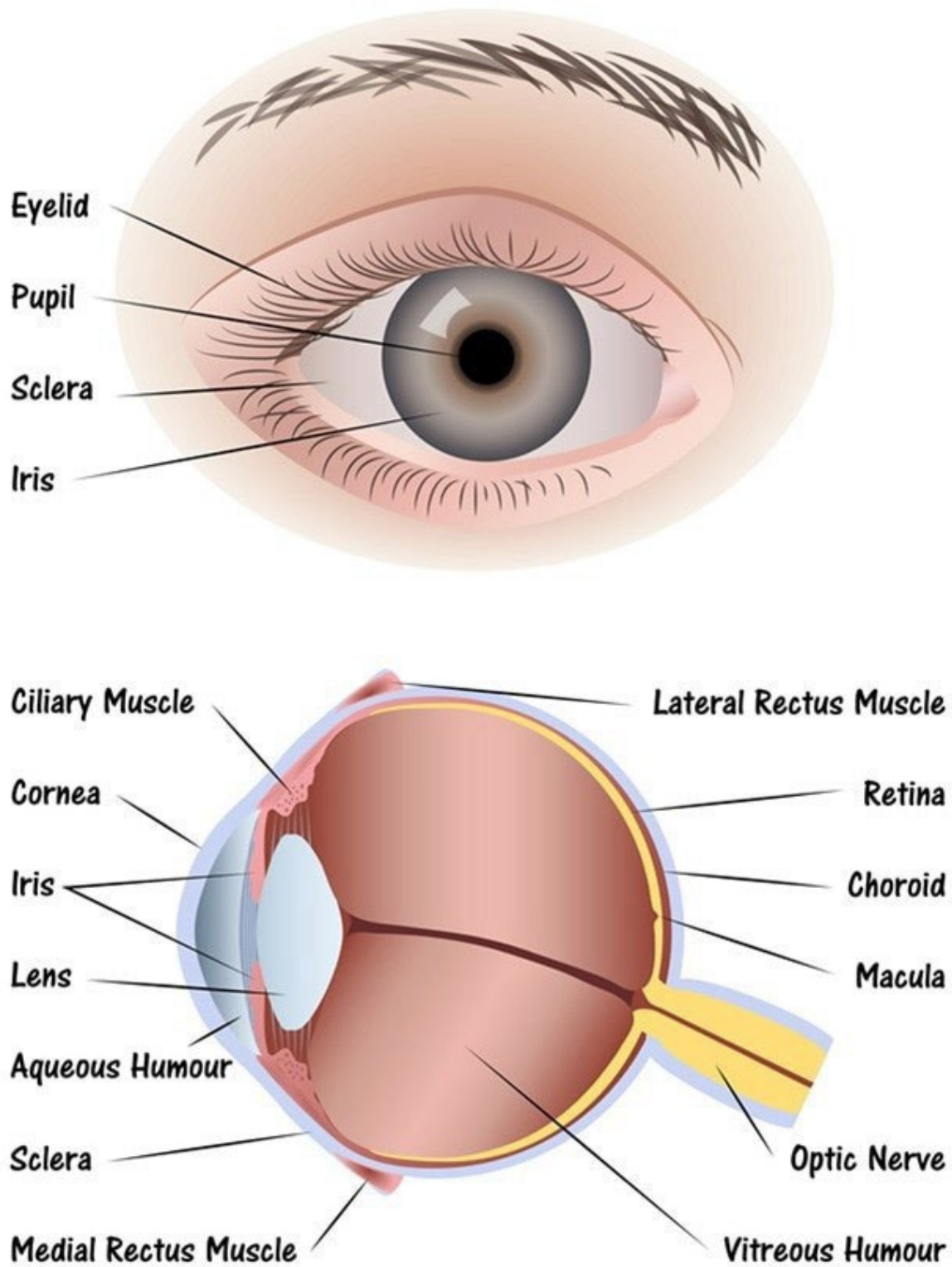Step 8: Accuracy and Score of mode
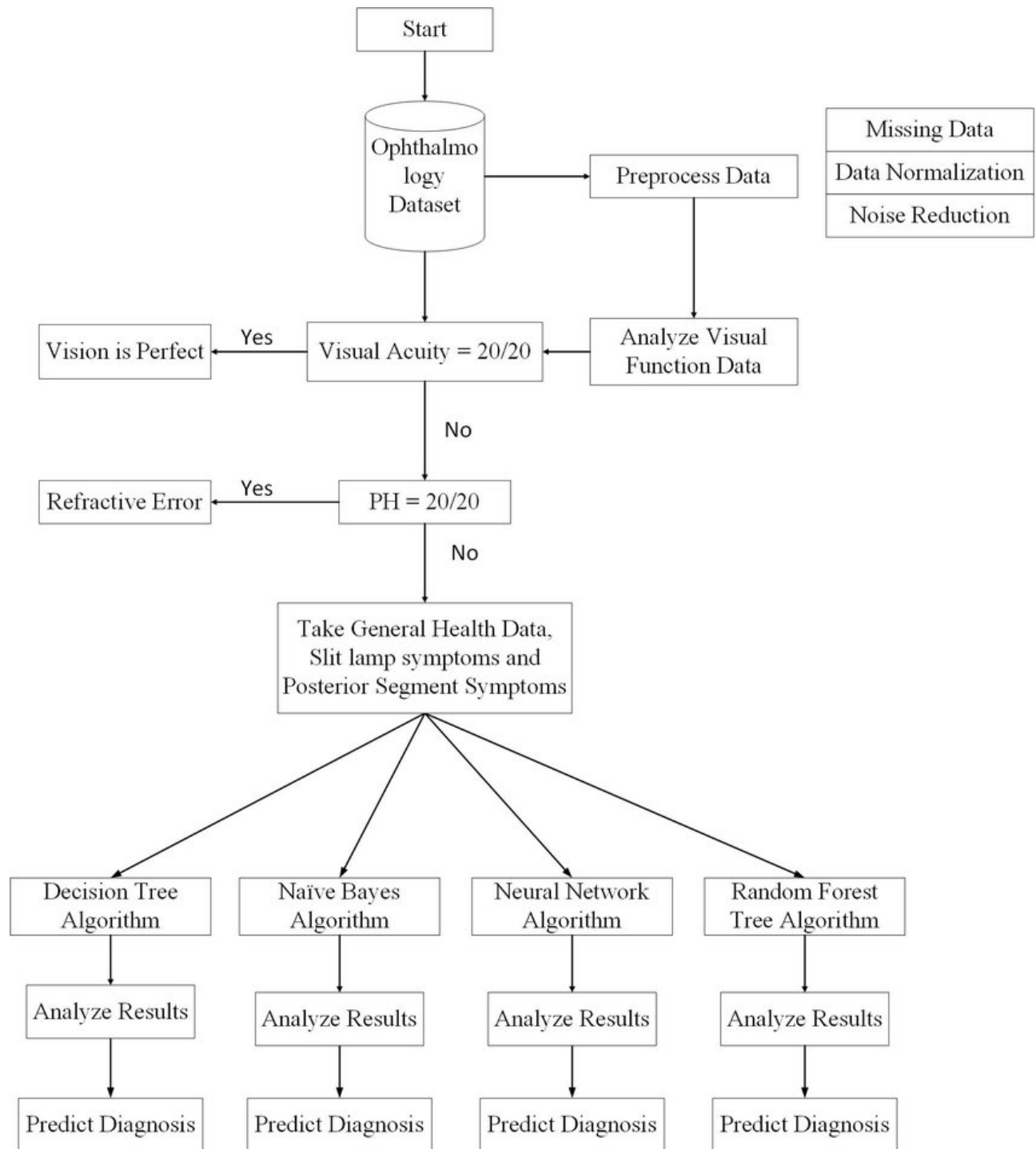
4.2 Diagrams

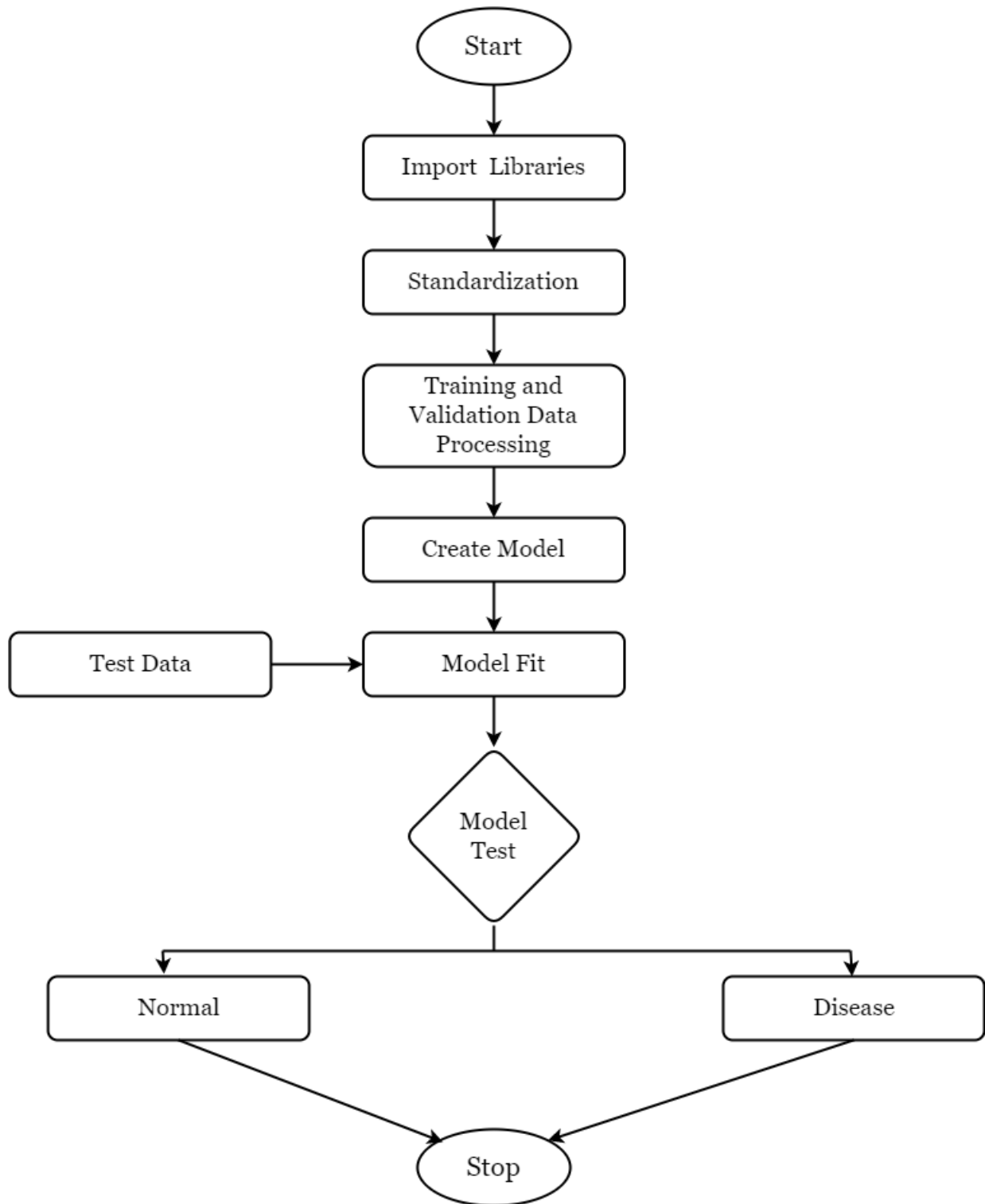Figure 4.1: Human Eye

Figure 4.2 System Framework
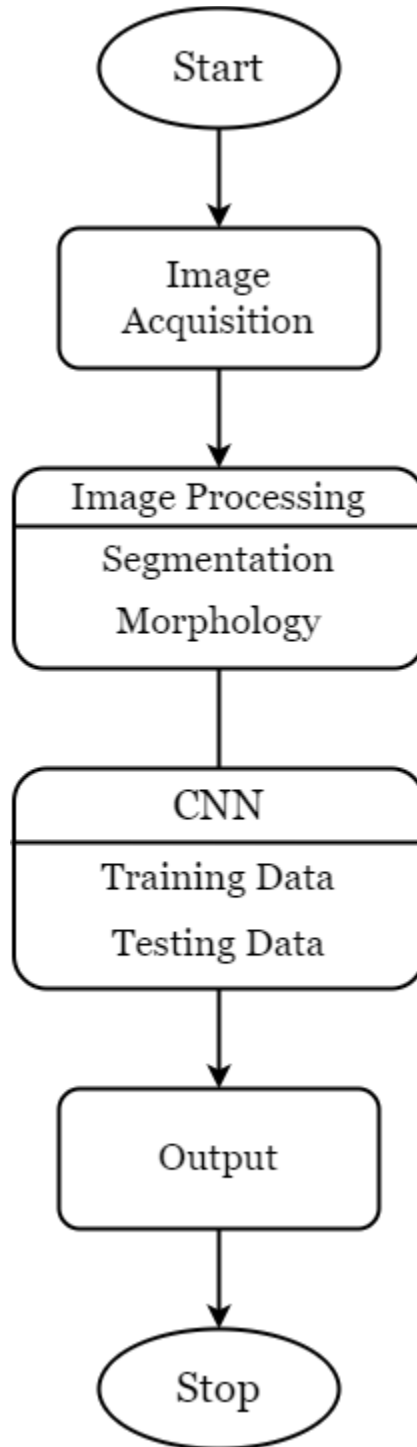
Figure 4.3: Work Flow Diagram

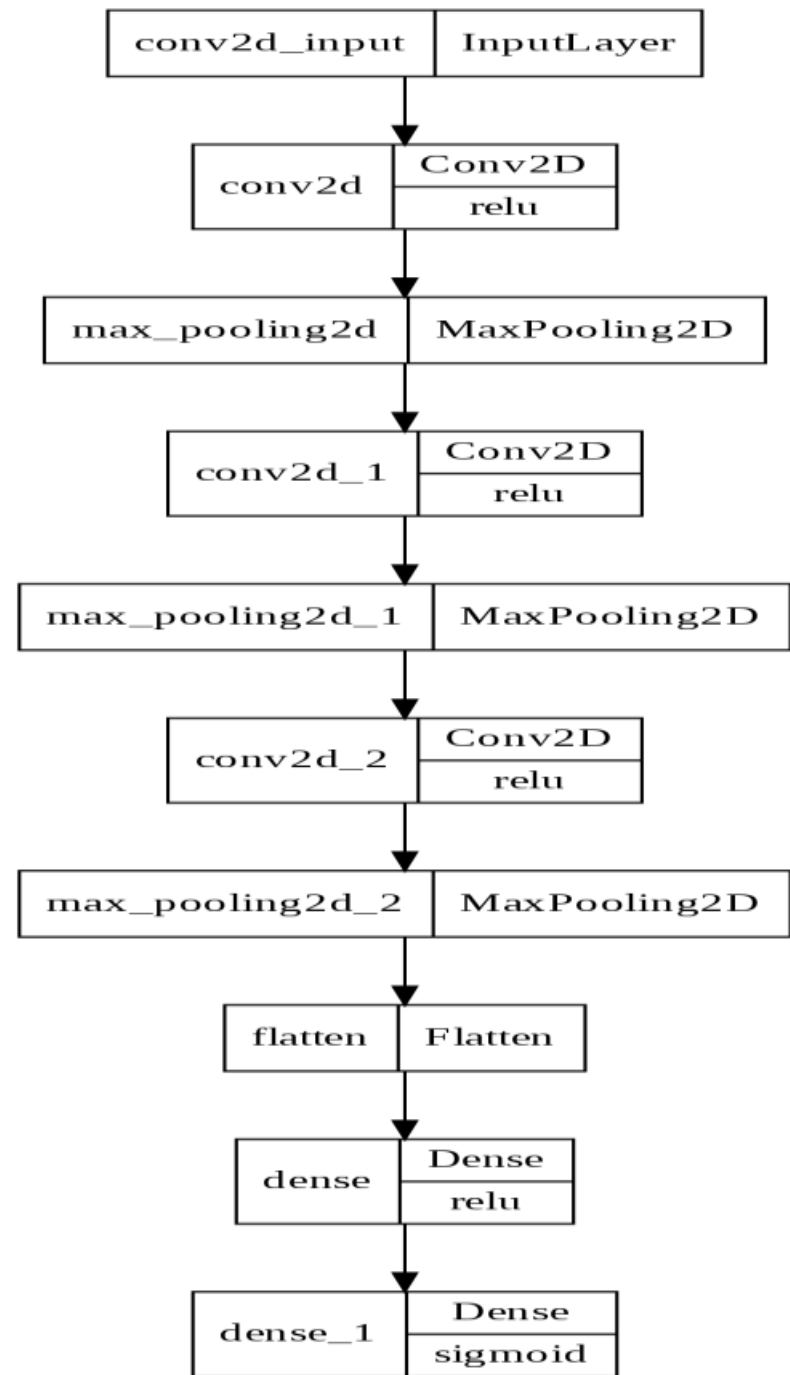Figure 4.4: Flow Cart of Training Process

Figure 4.5: Model Architecture

Chapter 5
Implementation

5.1 Dataset

A data set (or dataset) is a collection of data. In the case of tabular data, a data set corresponds to one or more database tables, where every column of a table represents a particular variable, and each row corresponds to a given record of the data set in question.

5.1.1 Cataract Dataset Sample

Figure 5.1: Cataract Dataset Sample

5.2 Model Training

Model training in machine language is the process of feeding an ML algorithm with data to help identify and learn good values for all attributes involved. There are several types of machine learning models, of which the most common ones are supervised and unsupervised learning.

5.2.1 Source Code

```python
import numpy as np

import pandas as pd

import cv2

import random

import os

from tqdm import tqdm

import matplotlib.pyplot as plt

from tensorflow.keras.preprocessing.image import ImageDataGenerator

from sklearn.model_selection import train_test_split

from sklearn.metrics import confusion_matrix, classification_report, accuracy_score

from mlxtend.plotting import plot_confusion_matrix

from efficientnet.tfkeras import EfficientNetB3

from tensorflow.keras import Sequential

from tensorflow.keras.layers import GlobalAveragePooling2D, Dense

from tensorflow.keras.optimizers import Adam

from tensorflow.keras.callbacks import ModelCheckpoint, EarlyStopping, ReduceLROnPlateau

import scipy


# Set a random seed for reproducibility

random.seed(42)


# Read the data

df = pd.read_csv('full_df.csv')
```

```python
# Helper function to check if 'cataract' is present in the text
def has_cataract(text):
    if "cataract" in text:
        return 1
    else:
        return 0


# Create columns to indicate presence of cataract
df["left_cataract"] = df["Left-Diagnostic Keywords"].apply(lambda x: has_cataract(x))
df["right_cataract"] = df["Right-Diagnostic Keywords"].apply(lambda x: has_cataract(x))


# Filter images with cataract and normal fundus
left_cataract = df.loc[(df.C == 1) & (df.left_cataract == 1)]["Left-Fundus"].values
right_cataract = df.loc[(df.C == 1) & (df.right_cataract == 1)]["Right-Fundus"].values
left_normal = df.loc[(df.C == 0) & (df["Left-Diagnostic Keywords"] == "normal fundus")]["Left-Fundus"].sample(250, random_state=42).values
right_normal = df.loc[(df.C == 0) & (df["Right-Diagnostic Keywords"] == "normal fundus")]["Right-Fundus"].sample(250, random_state=42).values


# Concatenate the cataract and normal fundus images
cataract = np.concatenate((left_cataract, right_cataract), axis=0)
normal = np.concatenate((left_normal, right_normal), axis=0)


# Load images and create the dataset
dataset_dir = "preprocessed_images"
image_size = 224
labels = []
dataset = []
```
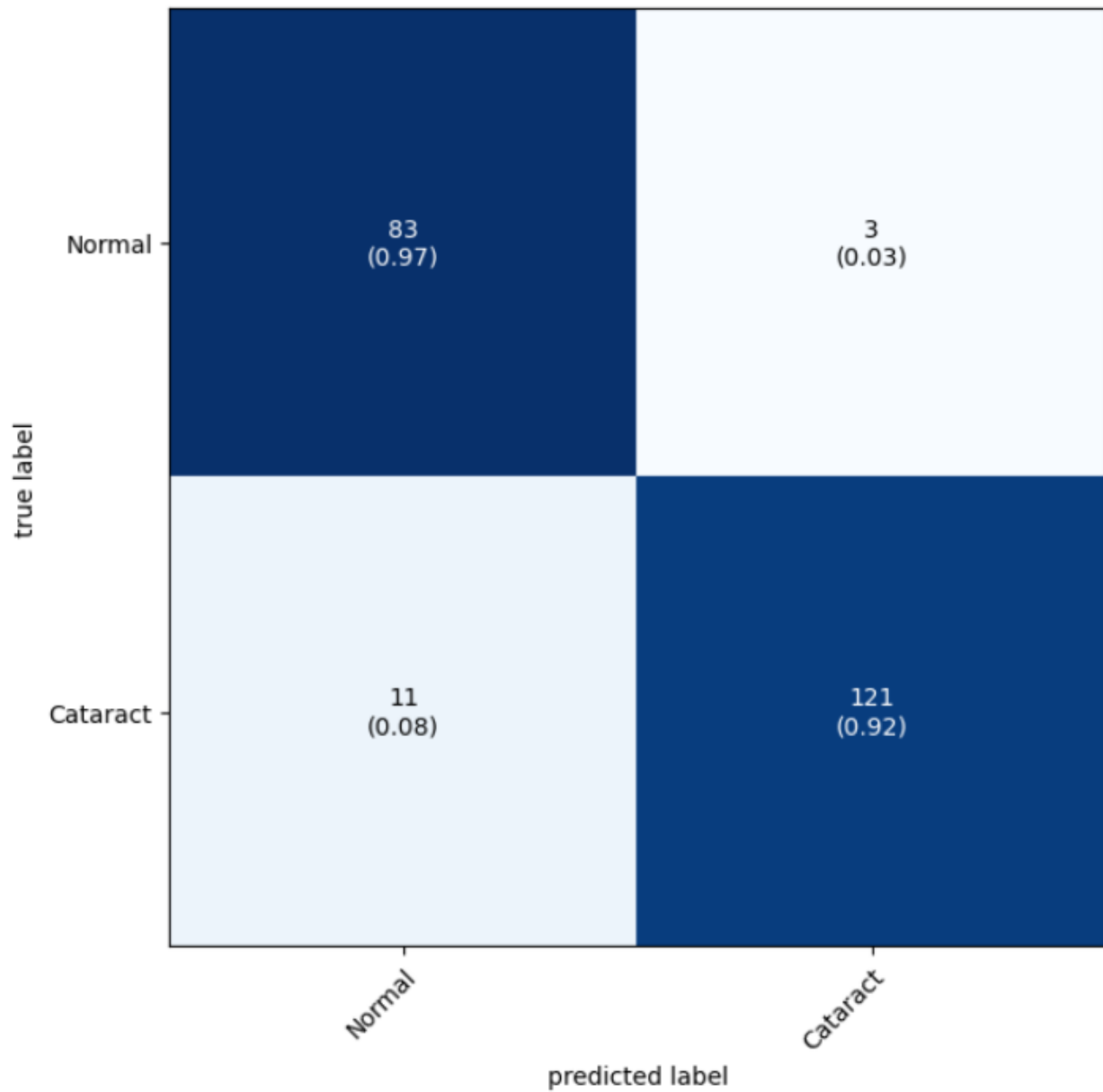
5.2.2 Trained Model Graph

Fig 5.2: Trained Model Graph

5.3 Website Design

Web design refers to the design of websites that are displayed on the internet. It usually refers to the user experience aspects of website development rather than software development.

5.3.1 Source Code

```
<!DOCTYPE html>

<html>

<head>

    <title>Cataract Detection</title>

    <!-- Bootstrap CSS -->

    <link          rel="stylesheet"          href="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/5.3.0/css/bootstrap.min.css">

</head>

<body>

    <!-- Navbar -->

    <nav class="navbar navbar-dark bg-dark">

        <div class="container">

            <span class="navbar-brand mb-0 h1">Cataract Detection</span>

        </div>

    </nav>


    <!-- Main Content -->

    <div class="container mt-5">

        <div class="row justify-content-center">

            <div class="col-md-6">

                <div class="card">

                    <div class="card-body">

                        <!-- Logo and Hospital Name -->

                        <div class="text-center mb-4">
```

```html
            <img src="path_to_logo.png" alt="Logo" width="100" height="100">

            <h4>Hospital Name</h4>

        </div>


        <!-- Upload Form -->

        <form action="/upload" method="post" enctype="multipart/form-data" id="upload-form">

            <div class="mb-3">

                <input class="form-control" type="file" name="file" id="file-input">

            </div>

            <div class="text-center">

                <button type="button" class="btn btn-primary" onclick="predictImage()">Upload Image</button>

            </div>

        </form>


        <!-- Uploaded Image Preview -->

        <div class="mt-4 text-center" id="image-preview" style="display: none;">

            <h5>Uploaded Image:</h5>

            <img src="" alt="Uploaded Image" id="uploaded-image" style="max-width: 100%;">

        </div>


        <!-- Prediction Result -->

        <div class="mt-4 text-center" id="prediction-result" style="display: none;">

            <h5>Prediction:</h5>

            <p id="prediction-text"></p>

        </div>

    </div>
```
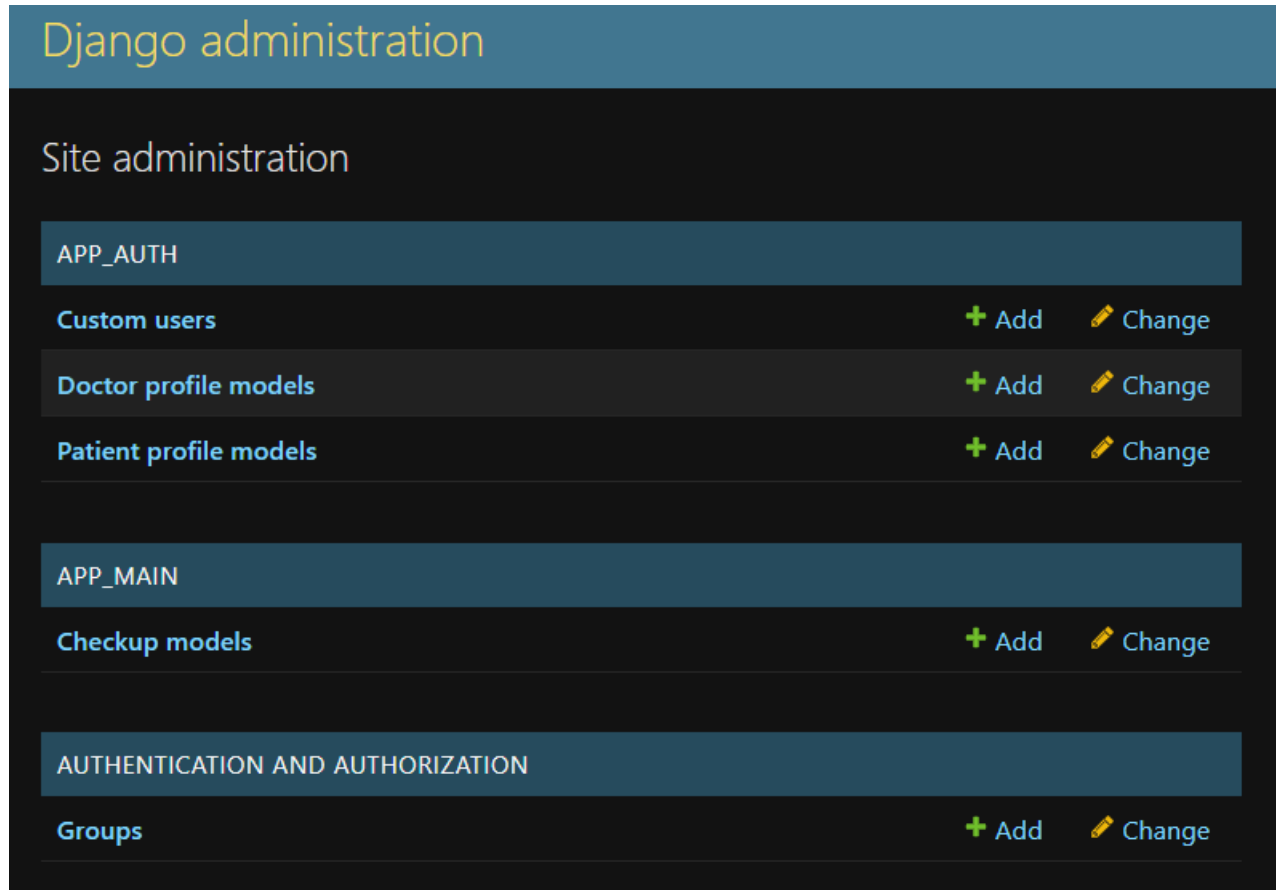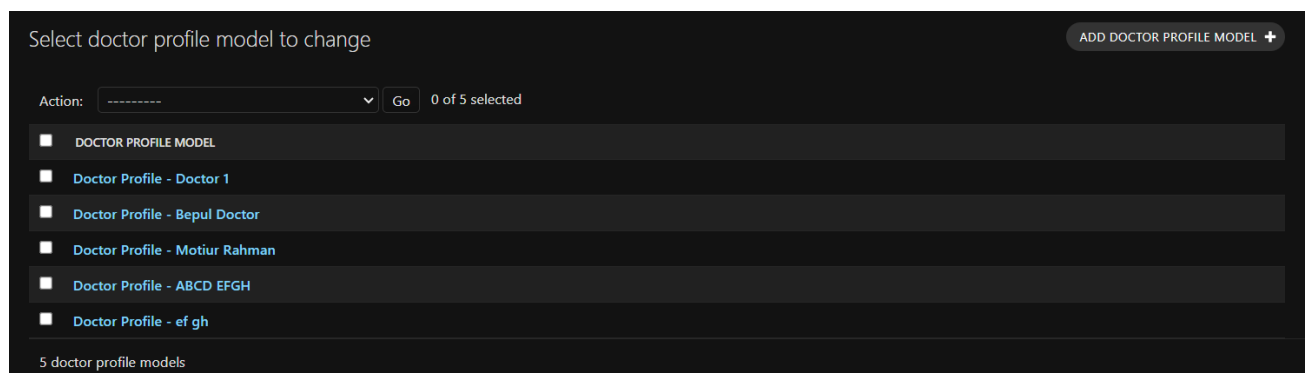
Chapter 6
Result Analysis

6.1 Admin Panel

6.1.1 Main Page

6.1.2 Doctor & Patient Profile Models

Select patient profile model to change

ADD PATIENT PROFILE MODEL +

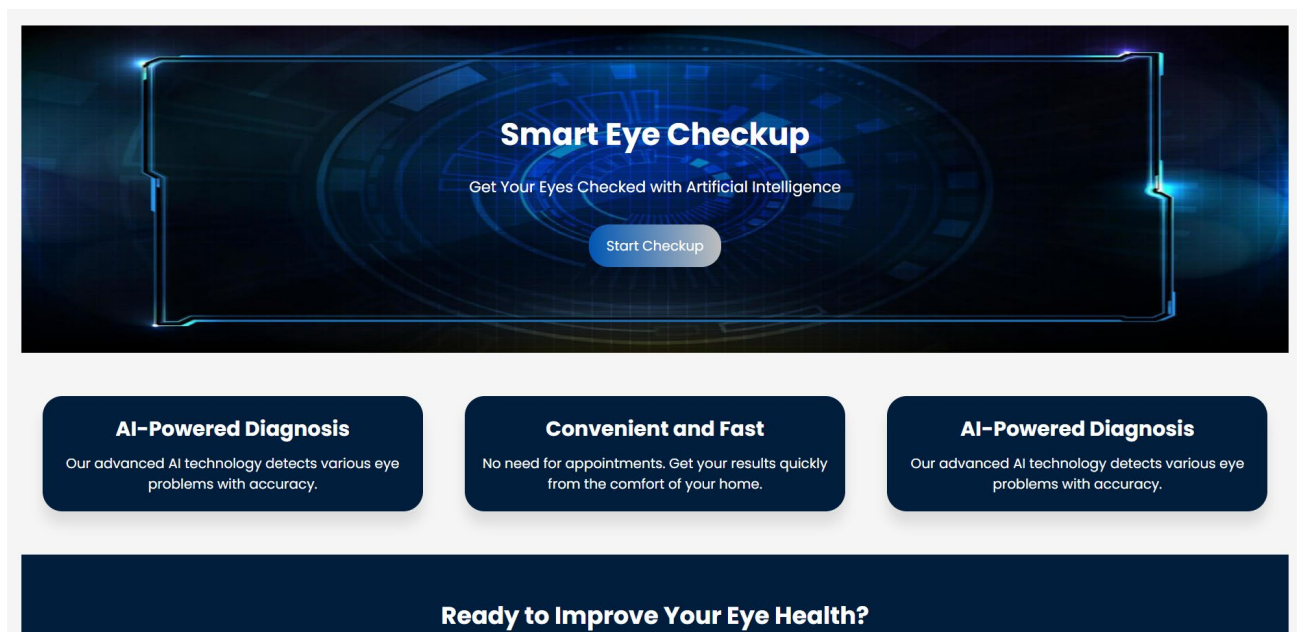Action: [---------] [Go] 0 of 2 selected

☐ PATIENT PROFILE MODEL

☐ Patient Profile - Bepul Saha

☐ Patient Profile - hjfsdf hdfjksdfv

2 patient profile models

6.2 Website

6.2.1 Homepage

Smart Eye Checkup
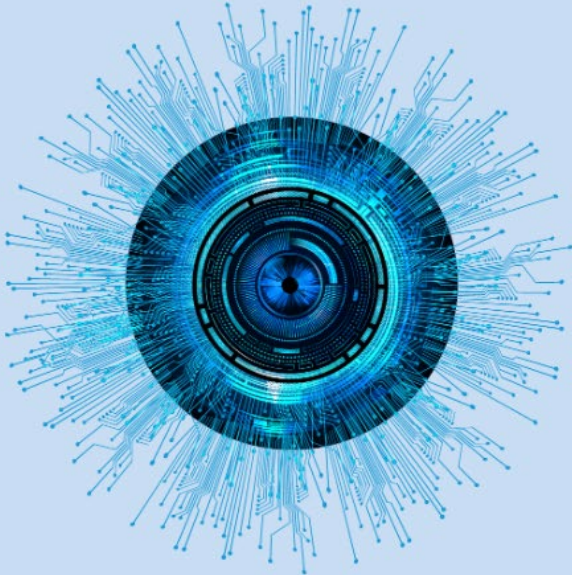
Get Your Eyes Checked with Artificial Intelligence

Start Checkup

AI-Powered Diagnosis
Our advanced AI technology detects various eye problems with accuracy.

Convenient and Fast
No need for appointments. Get your results quickly from the comfort of your home.

AI-Powered Diagnosis
Our advanced AI technology detects various eye problems with accuracy.

Ready to Improve Your Eye Health?

6.2.2 Login & Sign Up Panel
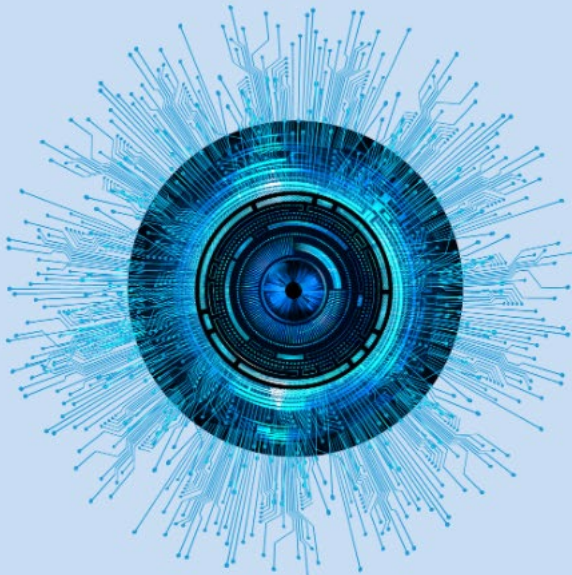
6.2.3 Dashboard
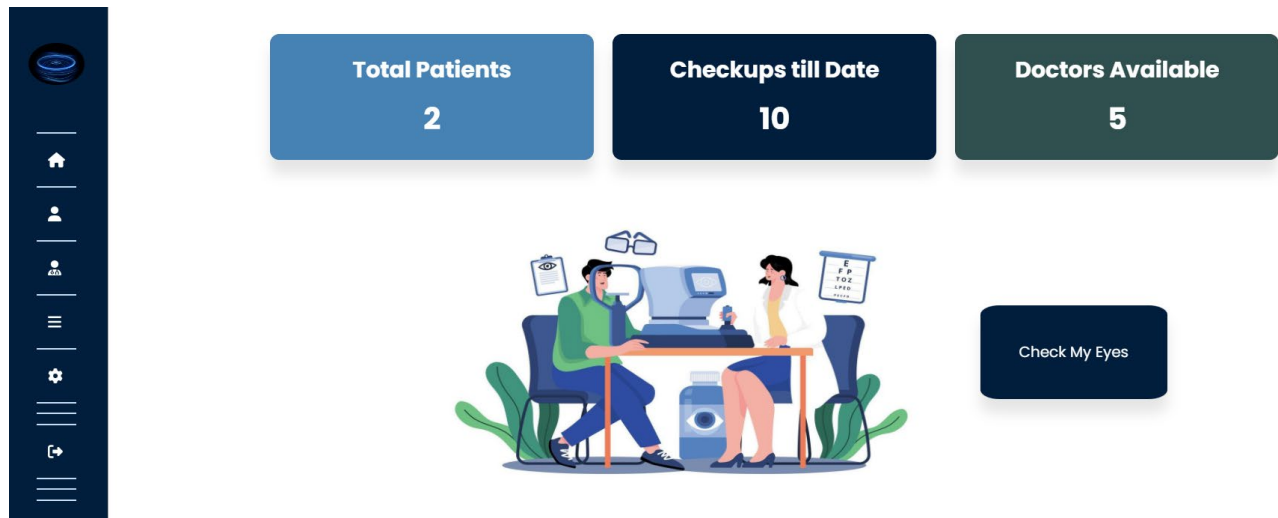
6.2.4 Eye Checkup

6.3 Result

Chapter 7
Conclusion

There are various models developed for eye disease detection in the recent years. In this paper, we have used the deep neural networks along with some core libraries like OpenCV, keras, TensorFlow, pandas, NumPy. We are successfully able to achieve all our objectives.

1. Deep learning model which classifies between normal eye and a diseased eye has been successfully built.

2. The model will be able to guide the users to know their eye condition (for specified eye disease namely cataract).

3. The model is cost effective and has simple interface.

4. The model shares the details of the nearby eye doctors to the use to get their eye check-up.

7.1 Future Scope

1. We can add more diseases to be detected by the model by collecting more datasets.

2. The admin panel will be automated by the time being it will be developed

3. Interact between Doctor & Patient will be more functionable.

7.2 Business Prospect

Speedy diagnosis. However, the recording of health data in a standard form still requires attention so that machine learning can be more accurate and reliable by considering multiple features. The aim of this study is to develop a general framework for recording diagnostic data in an international standard format to facilitate prediction of disease diagnosis based on symptoms using machine learning algorithms. Efforts were made to ensure error-free data entry by developing a user-friendly interface.

Chapter 8
References

8.1 References

[1]   (Walter, Image Processing, 2002) In October 2002, Thomas Walter, Jean-Claude Klein, Pascale Massin and Ali Erginay, Image Processing to the Diagnosis of Diabetic Retinopathy-- Detection of Exudates in color Fundus Images of Human Retina.

[2]   (Preethi, Glaucma, 2013) In the year 2013, Preethi and Jyothika Pruthi, Image processing Technique for Glaucoma Detection.

[3]   (Li, Splat, Nov 15) In November 2015, Li Tang, Meindert Niemeijer, Michael D. Abramoff, Splat feature classification: Detection of the presence of large retinal hemorrhages.

[4]   (Kev, Enc, 06) In December 2006 , Kevin Noronha,Jagadish Nayak,S.NBhatproposed a work on "Enhancement of retinal fundus Image to highlight the features for detection of abnormal eyes".

[5]   (Abbas, GD, 2017) In the year 2017,Abbas Q proposed work on "Glaucoma-Deep: Detection of Glaucoma Eye Disease on Retinal Fundus Images using Deep Learning".

[6]   (Lan,RIMLT, 16) In the year 2016,Mr.Langade Umesh,Ms.Malkar Mrunalini,Dr.Swati Shinde proposed a work on "Review of Image Processing and Machine Learning Techniques for Eye Disease Detection and Classification".

[7]   (Al, V. San) A Novel Approach for Classifying Medical Images Using Data Mining Techniques. Alamelu Mangai, Jagadish Nayak and V. Santhosh Kumar.

[8]   (SVMD, Inb) Detection of Retinal Hemorrhage in Fundus Images by Classifying the Splat Features Using SVM. Inbarathi.R, Karthikeyan.R

[9]   A Study on Retinal Disease Classification and Filtration Approaches. Parul1, Neetu Sharma.