MACHINE LEARNING BASED IRIS RECOGNITION MODERN VOTING SYSTEM

A Major Project Report Submitted to

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD

In Partial Fulfillment of the requirement For the Award of the Degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

Submitted by

SANIYA TAHREEN	**(20N01A0593)**
PUNNAM SHIVANI	**(20N01A0585)**
NUNE PAVAN KUMAR	**(21N05A0510)**
SHAIK SHARUKH AHMED	**(21N05A0511)**
THADAGONDA SHASHIDHAR	**(20N01A05A5)**

Under the supervision of

Mr. MAJOJU SRIDHAR KUMAR

Assistant Professor

SREE CHAITANYA
EDUCATIONAL INSTITUTIONS

Department of Computer Science and Engineering

SREE CHAITANYA COLLEGE OF ENGINEERING

(Affiliated to JNTUH, HYDERABAD)
THIMMAPUR, KARIMNAGAR, TELANGANA-505 527

MAY-2024

SREE CHAITANYA COLLEGE OF ENGINEERING
(Affiliated to JNTUH, HYDERABAD)
THIMMAPUR, KARIMNAGAR, TELANGANA-505 527
Department of Computer Science and Engineering

CERTIFICATE

This is to certify that the major project report entitled **“MACHINE LEARNING BASED IRIS RECOGNITION MODERN VOTING SYSTEM”** is being submitted by **SANIYA TAHREEN (20N01A0593), PUNNAM SHIVANI (20N01A0585), NUNE PAVAN KUMAR (21N05A0510), SHAIK SHARUKH AHMED (21N05A0511), THADAGONDA SHASHIDHAR (20N01A05A5),** for partial fulfillment of the requirement for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering** discipline to the **Jawaharlal Nehru Technological University, Hyderabad** during the academic year 2023-2024 is a bonafide work carried out by them under my guidance and supervision.

The result embodied in this report has not been submitted to any other University or institution for the award of any degree of diploma.

Project Guide	**Head Of the Department**
Mr. MAJOJU SRIDHAR KUMAR	**Dr. KHAJA ZIAUDDIN**
Assistant Professor	Associate Professor
Department of CSE	Department of CSE

External Examiner

MSR PROJECTS

(AN SSI CERTIFIED INSTITUTION)

TRAINING | DEVELOPMENT | PLACEMENTS

Date:

TO WHOM SO EVER IT MAY CONCERN

This is to certify that **SANIYA TAHREEN (20N01A0593), PUNNAM SHIVANI (20N01A0585), NUNE PAVAN KUMAR(21N05A0510), SHAIK SHARUKH AHMED (21N05A0511), THADAGONDA SHASHIDHAR (20N01A05A5), IV-B.Tech-CSE, Sree Chaitanya College of Engineering, Karimnagar**, have done their project work / Internship on the subject name of "**MACHINE LEARNING BASED IRIS RECOGNITION MODERN VOTING SYSTEM".**

They have done the project / Internship under the guidance of **Mr. K.MANOJ KUMAR**, in **MSR PROJECTS**, Hyderabad.

During the period of their project work with us, we found their conduct and character are Good.

We wish all the best for their future endeavors.

MSR PROJECTS ★ HYDERABAD ★

HR-Manager
(P. MADHURI)

SREE CHAITANYA COLLEGE OF ENGINEERING

(Affiliated to JNTUH, HYDERABAD)
THIMMAPUR, KARIMNAGAR, TELANGANA-505 527

Department of Computer Science and Engineering

DECLARATION

We, **SANIYA TAHREEN (20N01A0593),PUNNAM SHIVANI(20N01A0585), NUNE PAVAN KUMAR(21N05A0510), SHAIK SHARUKH AHMED (21N05A0511), THADAGONDA SHASHIDHAR (20N01A05A5),** are students of Bachelor of Technology in Computer Science and Engineering, during the academic year:2023-2024, hereby declare that the work presented in this major project work entitled **“MACHINE LEARNING BASED IRIS RECOGNITION MODERN VOTING SYSTEM”** is the outcome of our own bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics and carried out under the supervision of **Mr. MAJOJU SRIDHAR KUMAR, Assistant Professor.**

It contains no material previously published or written by another person nor material which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

SANIYA TAHREEN	**(20N01A0593)**
PUNNAM SHIVANI	**(20N01A0585)**
NUNE PAVAN KUMAR	**(21N05A0510)**
SHAIK SHARUKH AHMED	**(21N05A0511)**
THADAGONDA SHASHIDHAR	**(20N01A05A5)**

Date:

Place:

SREE CHAITANYA
EDUCATIONAL INSTITUTIONS

SREE CHAITANYA COLLEGE OF ENGINEERING
(Affiliated to JNTUH, HYDERABAD)
THIMMAPUR, KARIMNAGAR, TELANGANA-505 527
Department of Computer Science and Engineering

ACKNOWLEDGEMENTS

The satisfaction that accomplishes the successful completion of any task would be incomplete without the mention of the people who make it possible and whose constant guidance and encouragement crown all the effortswith success.

We would like to express our sincere gratitude and indebtedness to Major project supervisor of **Mr. MAJOJU SRIDHAR KUMAR, Assistant Professor**, Department of Computer Science and Engineering, Sree Chaitanya College of Engineering, LMD Colony, Karimnagar for his valuable suggestions and interest throughout the course of this major project report.

We are also thankful to Head of the department **Dr. KHAJA ZIAUDDIN, Associate Professor & HOD**, Department of Computer Science and Engineering, Sree Chaitanya College of Engineering, LMD Colony, Karimnagar for providing excellent infrastructure and a nice atmosphere for completing this major project report successfully.

We sincerely extend out thanks to **Dr. G. VENKATESWARLU, Principal,** Sree Chaitanya College of Engineering, LMD Colony, Karimnagar, for providing all the facilities required for completion of this major project report.

We convey our heartfelt thanks to the lab staff for allowing us to use the required equipment whenever needed.

Finally, we would like to take this opportunity to thank our family for their support through the work.

We sincerely acknowledge and thank all those who gave directly or indirectly their support in completion of this major project report.

SANIYA TAHREEN	**(20N01A0593)**
PUNNAM SHIVANI	**(20N01A0585)**
NUNE PAVAN KUMAR	**(21N05A0510)**
SHAIK SHARUKH AHMED	**(21N05A0511)**
THADAGONDA SHASHIDHAR	**(20N01A05A5)**

ABSTRACT

A paper ballot or an Electronic Voting Machine (EVM) based on Direct Response Electronic (DRE) or Identical Ballot Boxes have traditionally been used for voting. This study recommends a digital voting system based on Machine Learning algorithm that uses Iris recognition to address the flaws in the current voting process in order to fix the traditional voting system's flaws. A program called the Iris recognition-based Voting System identifies people based on the iris pattern of their eyes. Iris recognition is an automated biometric identification technology that analyses video evidence of one or both of an individual's iris to identify complex patterns that are distinct, stable, and visible from a distance. A voter may only cast one ballot, where the proposed technology prohibits multiple votes from the same person because it can spot duplicate entries. Additionally, this technique does away with the need for the user to carry a voter ID that has the relevant information since the Aadhar is incorporated with the voter ID thus enhancing the digitalization by means of digital verification of biometric and iris pattern available in Aadhar card of every user. At the voting venue, a simple iris scan will allow the voter's iris to be collected and used as identification. The iris recognition process consists of the following four steps: image acquisition, iris segmentation, feature extraction, and pattern matching. Iris recognition is one of the most trustworthy biometric modalities due to its high identification rate. Thereby this system eliminates the major drawbacks of traditional voting systems and enhances the digital voting by incorporating the modern transformation.

CONTENTS

LIST OF FIGURES

CHAPTER-1

INTRODUCTION

The election system is the pillar of the every democracy. The depth of democracy is voting. The voting process must be reliable, and the voting record must be accurately and reasonably recorded. The success of democratic administration is totally dependent on the results of the election. The election process provides the right to every citizen of a country to select a legitimate representative among themselves who can guide the democratic system towards the welfare of the society.

The voting system has observed many effective changes over the past few decades, right from the traditional paper ballot voting to electronic voting and now towards the online voting. The voting system is improving step by step; advancement in the new system eliminates the drawbacks of the previous system. Every system tries to overcome the loop holes of the previous system.

The primary goal of this paper is to understand the traditional voting system with the recently proposed voting system.In modern world, many new techniques such as voting process play an important role in any democratic country. Democracy is meant to allow people to vote freely and the election result is accepted by voters group.

The biometric process has been mainly used to recognize individual types of physical aspects and features. For this purpose, a tremendous amount of acknowledgement technologies have been generally provided with the actual fingerprint, iris procedures and voice acknowledgement. The biometric mainly deals with the proper technical and technological fields for the body controls and body dimensions. The authentication system is based on the appropriate biometric security system that has increased the actual importance within all countries.

The used system has been shown the proper valid and best impressive performance based on all these procedures and aspects. For this purpose, the fingerprint is the only procedure for providing the proper security techniques to provide the true uniqueness and the strong privacy properties of the entire system.

The exceptional fingerprint assurance or the proper kind of imprint approval has been mainly insinuating the automated methods and procedures to ensure similarity between the two people fingerprints.

The entire chapter has been generally provided with the actual purpose of the fundamental research that is overall dependent on the research objectives and respective research questions. In this chapter, the research framework of the entire study has also been provided. The fundamental research has described all the factors that are responsible for this recognition process.

Iris recognition is a method of biometric authentication that uses pattern-recognition techniques based on high-resolution images of the irises of an individual's eyes. Iris is a muscle within the eye that regulates the size of pupil, controlling the amount of light that controls the eye.

1.1 OVERVIEW

The project aims to revolutionize the voting process by introducing a digital voting system based on machine learning and iris recognition technology. By utilizing iris patterns for identification, the system ensures the integrity and security of the voting process, effectively preventing duplicate votes and eliminating the need for physical voter IDs. This innovative approach not only enhances the efficiency and accuracy of voting but also contributes to the digitalization of electoral practices. Through four key steps—image acquisition, iris segmentation, feature extraction, and pattern matching—the system captures and analyzes iris data to authenticate voters at polling stations. By incorporating Aadhar data with the voter ID, it streamlines the verification process, making it seamless and reliable. With its high identification rate and advanced technology, the system has the potential to revolutionize the voting experience, paving the way for more transparent and secure elections.

1.2 MOTIVATION

The motivation for this project stems from the recognition of the limitations and vulnerabilities present in traditional voting systems. Instances of fraud, logistical challenges, and inefficiencies in voter identification have raised concerns about the integrity and transparency of electoral processes. By harnessing advancements in machine learning and biometric technologies, such as iris recognition, the project aims to address these issues and modernize the voting experience.

The goal is to create a digital voting system that is not only secure and accurate but also user-friendly and accessible to all citizens. Ultimately, the project is driven by the desire to strengthen democratic practices, restore trust in electoral systems, and ensure that every vote counts.

1.3 EXISTING SYSTEM

India being a democracy that too world's largest, still conducts its elections using either Secret Ballet Voting or Electronic Voting Machines (EVM) both of which involves high costs, manual labor and are inefficient. So, the system must be optimized to be made efficient which would not leave room for unwanted means of voting. The most familiar issue faced by the election commission is inappropriate confirmation with respect to the arrangement of casting the votes, duplication or illegal casting of votes.

1.4 PROPOSED SYSTEM

Proposed voting method, we use a biometric system that uses multiple sources of biometric behavior. This can be done by combining multiple features of an individual or multiple bio-extraction and matching algorithms running on the same biometrics. This system improves the accuracy of matching the data for the biometric system in the voting process. Since there is no way for any candidate to provoke government-issued biometric records before the election process, we use iris recognition and fingerprint scanning for accuracy and reasonable voting result. Many types of layers are applied in the "convolution neural network (CNN)" for the purpose of the "multimodal biometric human authentication" process. The authentication process mainly has been done with respect to the face, veins, iris scanner, fingerprints and palm for increasing the robustness and visibility of the entire recognition system. The entire recognition process is very much tricky for hacking and copying.

1.5 OBJECTIVES

Use of machine learning technique for iris recognition is the most accurate, recognition rate wass high and the Use of iris for identification is the most secure method of identification. The segmentation methods used for iris detection and segmentation had an effect on accuracy and the speed of execution there is a need to use multiple techniques for user authentication.

CHAPTER-2

LITERATURE SURVEY

Literature survey is the most important step in software development process. Before developing the tool, it is necessary to determine the time factor, economy and company strength. Once these things are satisfied, ten next steps are to determine which operating System and Language can be used for developing the tool.

"Smart Voting" is used to identify people who are trying to vote a second time, and once the fingerprint print And iris are scanned, authentication is complete, and the user is locked into login.. Face detection, which is the major part of this project is done by using the Haar Cascade method. It is a machine learning object detection algorithm used to identify objects in an image or video. The process of election data is recorded, stored and preceded as digital information. Electronic voting system is used to fling vote as well as counting number of votes. The electronic voting system uses AVISPA technique

Canny Edge detection algorithm for localizing the iris and pupils.. Iris recognition system consists of five stages, such as, image acquisition, segmentation, normalization, feature extraction and matching.. In security of voting system by bringing advanced technologies of neural networks with multimodal biometrics (face recognition, fingerprint scan, retina scan etc).

Iris recognition refers to the automated method of verifying a match between two human IRIS. Iris scanner Capture the iris image and compare or match to database.. RFID tags have been used. Each and every tag contains the information related to individual voters.

The voter identity card is replaced by smart card in which all the detail of the person is updated. Only the specified person can poll using their smart card[9]. The incorporation of biometric technologies can be as simple as using a single biometric. However, a single biometric measure is always subject to security breaches, if not properly attended and administered..

M. Ibrahim, K. Ravindran, H. Lee, O. Farooqui and Q. H. Mahmoud, "ElectionBlock: An Electronic Voting System using Blockchain and Fingerprint Authentication", ***2021 IEEE*** **18th International Conference on Software Architecture Companion (ICSA-C)*, pp. 123-129, 2021.***

Voting is the method of choice used to make a large number of democratic decisions amongst many groups of people. Regardless of whether the method is used in professional or casual scenarios, it provides a fair and efficient way to determine a decision based on the majority. In smaller groups, keeping track of voter decisions is not a difficult task, however, in situations where there are hundreds of thousands of voters, keeping a precise record of voter decisions becomes important and more difficult. The advancements in blockchain technology provide a potential solution to the record-keeping problem of contemporary voting procedures, as blockchain technology by design, excels in applications where multiple users are working on immutable data. In this paper we discuss the design and development of ElectionBlock, a voting system that provides its own blockchain, running on a centralized network of nodes, with the integration of a biometric scanner, to maintain vote integrity and distinguish between registered and unregistered voters. This scheme allows data immutability while providing the user with security and control over their ballot. Experimental results demonstrate the potential for scalability of the system to handle a high volume of votes from multiple servers while maintaining data integrity, performance, and security. This paper will address the considerations taken to develop and implement the centralized and independent blockchain network for use as a voting platform with the integration of biometrics for the purpose of enhanced user security.

Shubham Gupta, Divanshu Jain and Milind Thomas Themalil, "Electronic Voting Mechanism using Microcontroller ATmega328P with Face Recognition", Proceedings of the Fifth International Conference on Computing Methodologies and Communication (ICCMC2021), pp. 1471-147, 2021.

An electronic voting system based upon face recognition algorithm, camera and ATmega328P is reported in this paper. Election is the main concern of any country when to elect someone. Also, it is very important to conduct the reliable, secure, fast and fair election so that the people can have faith in the system and they can elect the person to whom they want along with the low cost of paper work and manpower. In the present work, an electronic voting system is developed that includes the more secure version of existing system and the use of modern technology such as face recognition and IoT. The current work proposes an electronic system inclusive of multipled verification layers such that process reliability is ensured. In this process, each voter is registered into the system database well before the time of election. Now at the voting time, In the first step voter must verify his/her government identity such as Aadhar card or voting card with his/her proper picture, once it is verified, he/she moves to the second step. In second step voter has to go under the face reorganization process. Once the corresponding matching or verification is done, the voter will move to next step to cast vote to the candidate from the electronic voter machine. The cast vote is shown on display for the satisfaction of voters. Then the voting data is continuously uploaded on ThingSpeak server (can use dedicated server of Election department). The central office of election department can monitor the data in more reliable way so that no discrepancy/modification can take place later.

S. Jehovah Jireh Arputhamoni and A. Gnana Aravanan, "Online Smart Voting System Using Biometrics Based Facial and Fingerprint Detection on Image Processing and CNN", Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks *(ICICV)*, 2021.

India being a democratic country, still conducts its elections by using voting machines, which involves high cost and manual labor. Web-based system enables voter to cast their votes from anywhere in the world. Online website has a prevented IP address generated by the government of India for election purpose. People should register the name and address in the website. Election commission will collect the fingerprint and face image from the voters. The database or server will store the images. When the images are obtained on the casting day, it will be compared with database and provides a secured voting on the Election Day. System utilizes faces and fingerprints to unlock the voting system, similar to the mobile phone are used. The current system requires the physical presence of voter, which is inconvenient to many voters. The process consumes less time as well. Using the detection of face and fingerprint images, the number of fake voters can be reduced. The eyes and eyebrows distance remains constant with growing age to make the system more secure. This research work utilizes ten print image to detect the correct name of voter.

CHAPTER-3

PROBLEM DEFINITION

Security is one of the most important requirements in the design and implementation of electronic services. Access to and the protection of data during transmission and storage are critical issues. In this work, the security of Electronic Voting System (E-Voting system) is proposed to handled and process. Secured E-voting system is proposed, designed and implemented for private purposes. The system is proposed to protect mainly using iris identification as an authentication method.in addition to protection of transmission channel and stored data. Iris recognition is implemented using one of machine learning model.

CHAPTER-4

SYSTEM ANALYSIS

4.1 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are,

- Economical Feasibility
- Technical Feasibility
- Social Feasibility

4.1.1 Economical Feasibility

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

4.1.2 Technical Feasibility

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

4.1.3 Social Feasibility

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity.

The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

4.2 SYSTEM DESIGN AND DEVELOPMENT

4.2.1 Input Design

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

4.5.2 Output Design

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

- Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

- Select methods for presenting information.
- Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

- Convey information about past activities, current status or projections of the
- Future.
- Signal important events, opportunities, problems, or warnings.
- Trigger an action.

Confirm an action.

CHAPTER-5

SOFTWARE AND HARDWARE REQUIREMENTS

5.1 HARDWARE REQUIRMENTS

• System	:	Intel i3
• Hard Disk	:	1 TB.
• RAM	:	4 GB.

5.2 SOFTWARE REQUIRMENTS

• Operating system	:	Windows 10.
• Coding Language	:	Python.
• Front-End	:	Python

5.3 FUNCTIONAL REQUIREMENTS

- Graphical User interface with the User.

5.4 NON-FUNCTIONAL REQUIREMENTS

- **Maintainability:** Maintainability is used to make future maintenance easier, meet new requirements. Our project can support expansion.

- **Robustness:** Robustness is the quality of being able to withstand stress, pressures or changes in procedure or circumstance. Our project also provides it.

- **Reliability:** Reliability is an ability of a person or system to perform and maintain its functions in circumstances. Our project also provides it.

- **Size:** The size of a particular application plays a major role, if the size is less then efficiency will be high. The size of database we have developed is 5.05 MB.

- **Speed:** If the speed is high then it is good. Since the no of lines in our code is less, hence the speed is high.

- **Power Consumption:** In battery-powered systems, power consumption is very important. In the requirement stage, power can be specified in terms of battery life.

However the allowable wattage can't be defined by the customer. Since the no of lines of code is less CPU uses less time to execute hence power usage will be less.

CHAPTER-6

DESIGN AND IMPLEMENTATION

6.1 ARCHITECTURE OF THE PROPOSED SYSTEM

System design is the process of designing the overall architecture of a system, including the hardware, software, and communication systems that make up the system. It involves understanding the needs and constraints of the system, identifying the components that need to be built or procured, and determining how those components will work together to achieve the desired outcomes.

6.1.1 Architecture

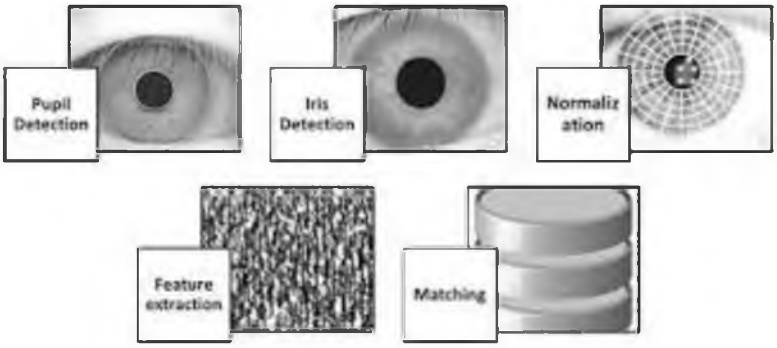


Fig: 6.1 Architecture Diagram

The algorithm starts with image normalization based on Daugman's rubber sheet model. This method uses information about centers and radiuses of iris and pupil. This solution was used because it can guarantee presentation clarity as well as it was much easier to work on transformed sample rather than the original iris sample. It is connected with the fact that it was much easier to analyze such samples.

Before we will be ready to obtain all significant information and create sufficient feature vector, preprocessing algorithms have to be applied in the normalized image. All these actions are required because iris sample is not adapted to easily extract the most important features. At the beginning of the preprocessing stage, we used histogram equalization. After this operation, we obtained the image in which the most significant iris points have been strengthened. (It is connected with the fact that the proposed operation can highlight the most important parts of the processed image.) This step allowed to observe them even by human eye. The images after normalization and after histogram equalization arepresented.

6.2 MODULE DESCRIPTION

Convolutional Neural Network(CNN) :

A Convolutional Neural Network (CNN) is a type of deep learning algorithm that is particularly well-suited for image recognition and processing tasks. It is made up of multiple layers, including convolutional layers, pooling layers, and fully connected layers.

The convolutional layers are the key component of a CNN, where filters are applied to the input image to extract features such as edges, textures, and shapes. The output of the convolutional layers is then passed through pooling layers, which are used to down-sample the feature maps, reducing the spatial dimensions while retaining the most important information. The output of the pooling layers is then passed through one or more fully connected layers, which are used to make a prediction or classify the image.

CNNs are trained using a large dataset of labeled images, where the network learns to recognize patterns and features that are associated with specific objects or classes. Once trained, a CNN can be used to classify new images, or extract features for use in other applications such as object detection or image segmentation.

CNNs have achieved state-of-the-art performance on a wide range of image recognition tasks, including object classification, object detection, and image segmentation. They are widely used in computer vision, image processing, and other related fields, and have been applied to a wide range of applications, including self-driving cars, medical imaging, and security systems.

- A convolutional neural network, or CNN, is a deep learning neural network sketched for processing structured arrays of data such as portrayals.
- CNN are very satisfactory at picking up on design in the input image, such as lines, gradients, circles, or even eyes and faces.
- This characteristic that makes convolutional neural network so robust for computer vision.
- CNN can run directly on a underdone image and do not need any preprocessing.
- A convolutional neural network is a feed forward neural network, seldom with up to 20.

- The strength of a convolutional neural network comes from a particular kind of layer called the convolutional layer.
- CNN contains many convolutional layers assembled on top of each other, each one competent of recognizing more sophisticated shapes.
- With three or four convolutional layers it is viable to recognize handwritten digits and with 25 layers it is possible to differentiate human faces.
- The agenda for this sphere is to activate machines to view the world as humans do, perceive it in a alike fashion and even use the knowledge for a multitude of duty such as image and video recognition, image inspection and classification, media recreation, recommendation systems, natural language processing, etc.

Convolutional Neural Network Design :

- The construction of a convolutional neural network is a multi-layered feed-forward neural network, made by assembling many unseen layers on top of each other in a particular order.
- It is the sequential design that give permission to CNN to learn hierarchical attributes.
- In CNN, some of them followed by grouping layers and hidden layers are typically convolutional layers followed by activation layers.

The pre-processing needed in a ConvNet is kindred to that of the related pattern of neurons in the human brain and was motivated by the organization of the Visual Cortex.

6.3 SYSTEM WORKFLOW

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. An interpreted language, Python has a design philosophy that emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer lines of code than might be used in languages such as C++or Java. It provides constructs that enable clear programming on both small and large scales. Python interpreters are available for many operating systems. C Python, the reference implementation of Python, is open source software and has a community-based development model, as do nearly all of its variant implementations. CPython is managed by the non-profit Python Software Foundation.

Python features a dynamic type system and automatic memory management. It supports ultiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.

Python Identifiers

A Python identifier is a name used to identify a variable, function, class, module or other object. An identifier starts with a letter A to Z or a to z or an underscore (_) followed by zero or more letters, underscores and digits (0 to 9).

Python does not allow punctuation characters such as @, $, and % within identifiers. Python is a case sensitive programming language. Thus, Manpower and manpower are two different identifiers in Python.

Here are naming conventions for Python identifiers −

- Class names start with an uppercase letter. All other identifiers start with a lowercase letter.
- Starting an identifier with a single leading underscore indicates that the identifier is private.
- Starting an identifier with two leading underscores indicates a strongly private identifier.
- If the identifier also ends with two trailing underscores, the identifier is a language-defined special name.

What is Python

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.

It is used for:

- web development (server-side),
- software development,
- mathematics,
- system scripting.

What can Python do

- Python can be used on a server to create web applications.
- Python can be used alongside software to create workflows.
- Python can connect to database systems. It can also read and modify files.

- Python can be used to handle big data and perform complex mathematics.
- Python can be used for rapid prototyping, or for production-ready software development.

Why Python

- Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc).
- Python has a simple syntax similar to the English language.
- Python has syntax that allows developers to write programs with fewer lines than some other programming languages.
- Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.
- Python can be treated in a procedural way, an object-orientated way or a functional way.

Good to know

- The most recent major version of Python is Python 3, which we shall be using in this tutorial. However, Python 2, although not being updated with anything other than security updates, is still quite popular.
- In this tutorial Python will be written in a text editor. It is possible to write Python in an Integrated Development Environment, such as Thonny, Pycharm, Netbeans or Eclipse which are particularly useful when managing larger collections of Python files.

Python Syntax compared to other programming languages

- Python was designed for readability, and has some similarities to the English language with influence from mathematics.
- Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
- Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.

Introduction

Python applications will often use packages and modules that don’t come as part of the standard library. Applications will sometimes need a specific version of a library, because the application may require that a particular bug has been fixed or the application may be written using an obsolete version of the library’s interface.

This means it may not be possible for one Python installation to meet the requirements of every application. If application A needs version 1.0 of a particular module but application B needs version 2.0, then the requirements are in conflict and installing either version 1.0 or 2.0 will leave one application unable to run.

The solution for this problem is to create a virtual environment, a self-contained directory tree that contains a Python installation for a particular version of Python, plus a number of additional packages.

Different applications can then use different virtual environments. To resolve the earlier example of conflicting requirements, application A can have its own virtual environment with version 1.0 installed while application B has another virtual environment with version 2.0. If application B requires a library be upgraded to version 3.0, this will not affect application A's environment.

Creating Virtual Environments

The module used to create and manage virtual environments is called venv. venv will usually install the most recent version of Python that you have available. If you have multiple versions of Python on your system, you can select a specific Python version by running python3 or whichever version you want.

To create a virtual environment, decide upon a directory where you want to place it, and run the venv module as a script with the directory path:

```
python3 -m venv tutorial-env
```

This will create the tutorial-env directory if it doesn't exist, and also create directories inside it containing a copy of the Python interpreter, the standard library, and various supporting files.

A common directory location for a virtual environment is .venv. This name keeps the directory typically hidden in your shell and thus out of the way while giving it a name that explains why the directory exists. It also prevents clashing with .env environment variable definition files that some tooling supports.

Once you've created a virtual environment, you may activate it.

On Windows, run:

```
tutorial-env\Scripts\activate.bat
```

On Unix or MacOS, run:

```
source tutorial-env/bin/activate
```

(This script is written for the bash shell. If you use the csh or fish shells, there are alternate activate.csh and activate.fish scripts you should use instead.)

Activating the virtual environment will change your shell's prompt to show what virtual environment you're using, and modify the environment so that running python will get you that particular version and installation of Python. For example:

```
$ source ~/envs/tutorial-env/bin/activate
(tutorial-env) $ python
Python 3.5.1 (default, May  6 2016, 10:59:36)
  ...
>>> import sys
>>>sys.path
['', '/usr/local/lib/python35.zip', ...,
'~/envs/tutorial-env/lib/python3.5/site-packages']
>>>
```

Managing Packages with pip

You can install, upgrade, and remove packages using a program called pip. By default pip will install packages from the Python Package Index, <https://pypi.org>. You can browse the Python Package Index by going to it in your web browser, or you can use pip's limited search feature:

```
 (tutorial-env) $ pip search astronomy
skyfield             - Elegant astronomy for Python
gary                - Galactic astronomy and gravitational dynamics.
novas                - The United States Naval Observatory NOVAS astronomy library
astroobs             - Provides astronomy ephemeris to plan telescope observations
PyAstronomy           - A collection of astronomy related tools for Python.
...
```

pip has a number of subcommands: "search", "install", "uninstall", "freeze", etc. (Consult the Installing Python Modules guide for complete documentation for pip.)

You can install the latest version of a package by specifying a package's name:

```
 (tutorial-env) $ pip install novas
Collecting novas
  Downloading novas-3.1.1.3.tar.gz (136kB)
Installing collected packages: novas
  Running setup.py install for novas
Successfully installed novas-3.1.1.3
```

You can also install a specific version of a package by giving the package name followed by == and the version number:

```
 (tutorial-env) $ pip install requests==2.6.0
Collecting requests==2.6.0
  Using cached requests-2.6.0-py2.py3-none-any.whl
Installing collected packages: requests
Successfully installed requests-2.6.0
```

If you re-run this command, pip will notice that the requested version is already installed and do nothing. You can supply a different version number to get that version, or you can run pip install --upgrade to upgrade the package to the latest version:

```
 (tutorial-env) $ pip install --upgrade requests
Collecting requests
Installing collected packages: requests
Found existing installation: requests 2.6.0
Uninstalling requests-2.6.0:
Successfully uninstalled requests-2.6.0
Successfully installed requests-2.7.0
```

pip uninstall followed by one or more package names will remove the packages from the virtual environment.

distribution taken from the supported_dists parameter.

Deprecated since version 2.6.

platform.linux_distribution(distname='', version='', id='', supported_dists=('SuSE', 'debian', 'redhat', 'mandrake', ...), full_distribution_name=1)

Tries to determine the name of the Linux OS distribution name.

Using the Python Interpreter

Invoking the Interpreter

The Python interpreter is usually installed as /usr/local/bin/python3.8 on those machines where it is available; putting /usr/local/bin in your Unix shell's search path makes it possible to start it by typing the command:

python3.8 to the shell. 1 Since the choice of the directory where the interpreter lives is an installation option, other places are possible; check with your local Python guru or system administrator. (E.g., /usr/local/python is a popular alternative location.)

On Windows machines where you have installed Python from the Microsoft Store, the python3.8 command will be available. If you have the py.exe launcher installed, you can use the py command. See Excursus: Setting environment variables for other ways to launch Python.

Typing an end-of-file character (Control-D on Unix, Control-Z on Windows) at the primary prompt causes the interpreter to exit with a zero exit status. If that doesn't work, you can exit the interpreter by typing the following command: quit().

The interpreter's line-editing features include interactive editing, history substitution and code completion on systems that support the GNU Readline library. Perhaps the quickest check to see whether command line editing is supported is typing Control-P to the first Python prompt you get. If it beeps, you have command line editing; see Appendix Interactive Input Editing and History Substitution for an introduction to the keys. If nothing appears to happen, or if ^P is echoed, command line editing isn't available; you'll only be able to use backspace to remove characters from the current line.

The interpreter operates somewhat like the Unix shell: when called with standard input connected to a tty device, it reads and executes commands interactively; when called with a file name argument or with a file as standard input, it reads and executes a script from that file.

A second way of starting the interpreter is python -c command [arg] ..., which executes the statement(s) in command, analogous to the shell's -c option. Since Python statements often contain spaces or other characters that are special to the shell, it is usually advised to quote command in its entirety with single quotes.

Some Python modules are also useful as scripts. These can be invoked using python -m module [arg] ..., which executes the source file for module as if you had spelled out its full name on the command line.

When a script file is used, it is sometimes useful to be able to run the script and enter interactive mode afterwards. This can be done by passing -i before the script.

All command line options are described in Command line and environment.

Argument Passing

When known to the interpreter, the script name and additional arguments thereafter are turned into a list of strings and assigned to the argv variable in the sys module. You can access this list by executing import sys. The length of the list is at least one; when no script and no arguments are given, sys.argv[0] is an empty string.

When the script name is given as '-' (meaning standard input), sys.argv[0] is set to '-'. When -c command is used, sys.argv[0] is set to '-c'. When -m module is used, sys.argv[0] is set to the full name of the located module. Options found after -c command or -m module are not consumed by the Python interpreter's option processing but left in sys.argv for the command or module to handle.

Interactive Mode

When commands are read from a tty, the interpreter is said to be in interactive mode. In this mode it prompts for the next command with the primary prompt, usually three greater-than signs (>>>); for continuation lines it prompts with the secondary prompt, by default three dots (...). The interpreter prints a welcome message stating its version number and a copyright notice before printing the first prompt:

```
$ python3.8
Python 3.8 (default, Sep 16 2015, 09:25:04)
[GCC 4.8.2] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

The Interpreter and Its Environment

Source Code Encoding

By default, Python source files are treated as encoded in UTF-8. In that encoding, characters of most languages in the world can be used simultaneously in string literals, identifiers and comments — although the standard library only uses ASCII characters for identifiers, a convention that any portable code should follow. To display all these characters properly, your editor must recognize that the file is UTF-8, and it must use a font that supports all the characters in the file.

To declare an encoding other than the default one, a special comment line should be added as the first line of the file. The syntax is as follows:

```
# -*- coding: encoding -*-
```

where encoding is one of the valid codecs supported by Python.

For example, to declare that Windows-1252 encoding is to be used, the first line of your source code file should be:

```
# -*- coding: cp1252 -*-
```

One exception to the first line rule is when the source code starts with a UNIX "shebang" line. In this case, the encoding declaration should be added as the second line of the file. For example:

```
#!/usr/bin/env python3
# -*- coding: cp1252 -*-
```

Packages and Versions

```
astunparse           ==1.6.3
certifi              ==2022.12.7
charset-normalizer     ==3.0.1
contourpy            ==1.0.7
cycler               ==0.11.0
Django               ==3.0.4
et-xmlfile           ==1.1.0
fonttools            ==4.38.0
gast                 ==0.3.3
google-pasta         ==0.2.0
grpcio               ==1.51.1
h5py                 ==2.10.0
idna                 ==3.4
importlib-metadata     ==6.0.0
joblib               ==1.2.0
Keras                ==2.3.1
Keras-Applications     ==1.0.8
Keras-Preprocessing    ==1.1.2
kiwisolver           ==1.4.4
Markdown             ==3.4.1
matplotlib           ==3.6.3
mysql-connector-python ==8.0.32
mysqlclient          ==2.1.1
numpy                ==1.21.4
oauthlib             ==3.2.2
openpyxl             ==3.1.0
opt-einsum           ==3.3.0
packaging            ==23.0
pandas               ==1.3.5
Pillow               ==9.4.0
pip                  ==20.1.1
```

protobuf	==3.20.3
pyparsing	==3.0.9
opencv-contrib-python	4.5.2.52

6.4 SYSTEM DESIGN

6.4.1 UML Diagrams

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

Goals:

The Primary goals in the design of the UML are as follows:

- Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
- Provide extendibility and specialization mechanisms to extend the core concepts.
- Be independent of particular programming languages and development process.
- Provide a formal basis for understanding the modeling language.
- Encourage the growth of OO tools market.
- Support higher level development concepts such as collaborations, frameworks, patterns and components.
- Integrate best practices.

Use Case Diagram

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

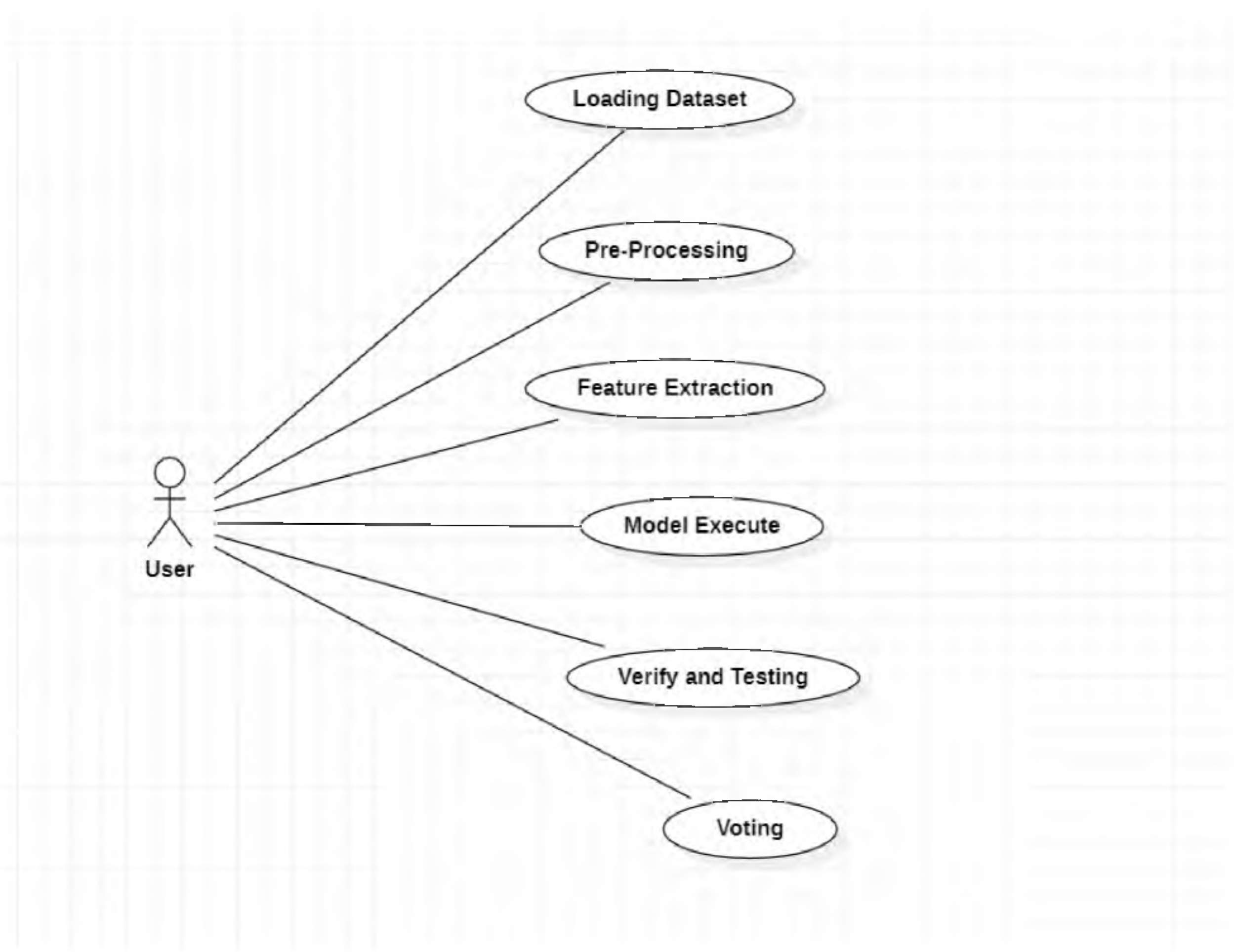


Fig: 6.2 Use case Diagram

Class Diagram

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

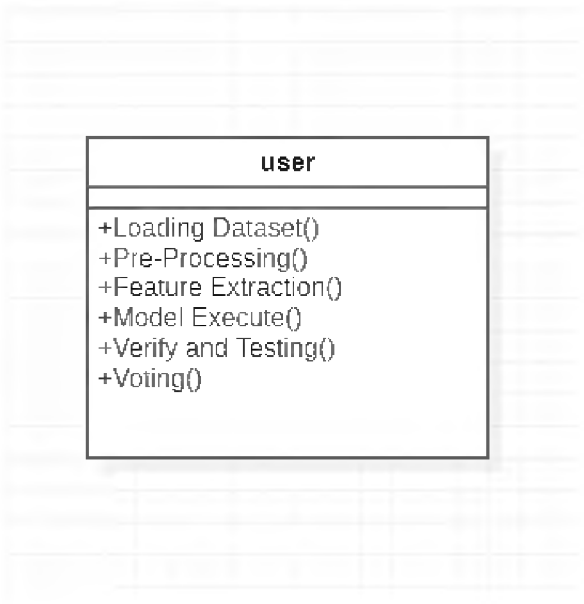


Fig: 6.3 Class Diagram

Sequence Diagram

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

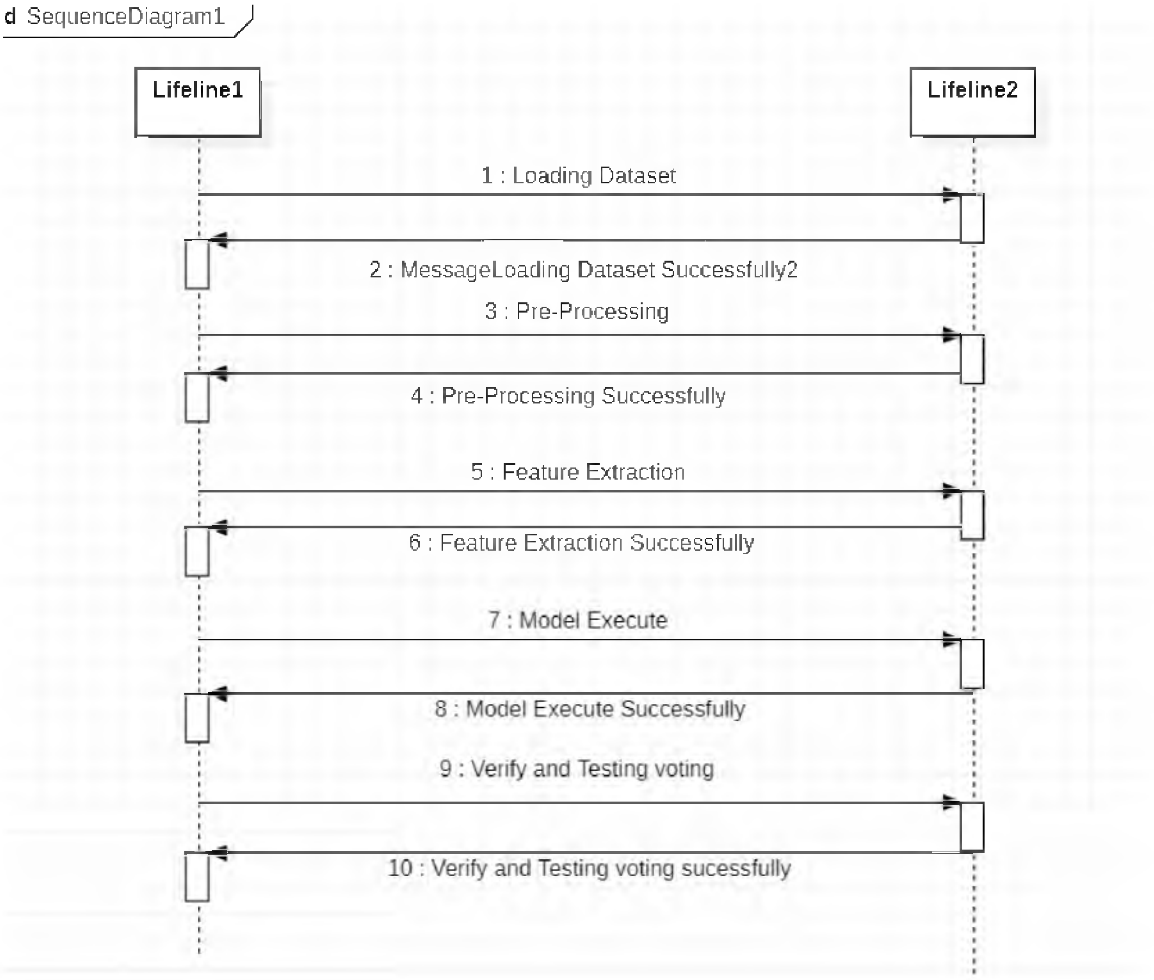

d SequenceDiagram1
Lifeline1
Lifeline2
1 : Loading Dataset
2 : MessageLoading Dataset Successfully2
3 : Pre-Processing
4 : Pre-Processing Successfully
5 : Feature Extraction
6 : Feature Extraction Successfully
7 : Model Execute
8 : Model Execute Successfully
9 : Verify and Testing voting
10 : Verify and Testing voting sucessfully

Fig: 6.4 Sequence Diagram

State Diagram:

A state diagram, as the name suggests, represents the different states that objects in the system undergo during their life cycle. Objects in the system change states in response to events. In addition to this, a state diagram also captures the transition of the object's state from an initial state to a final state in response to events affecting the system.

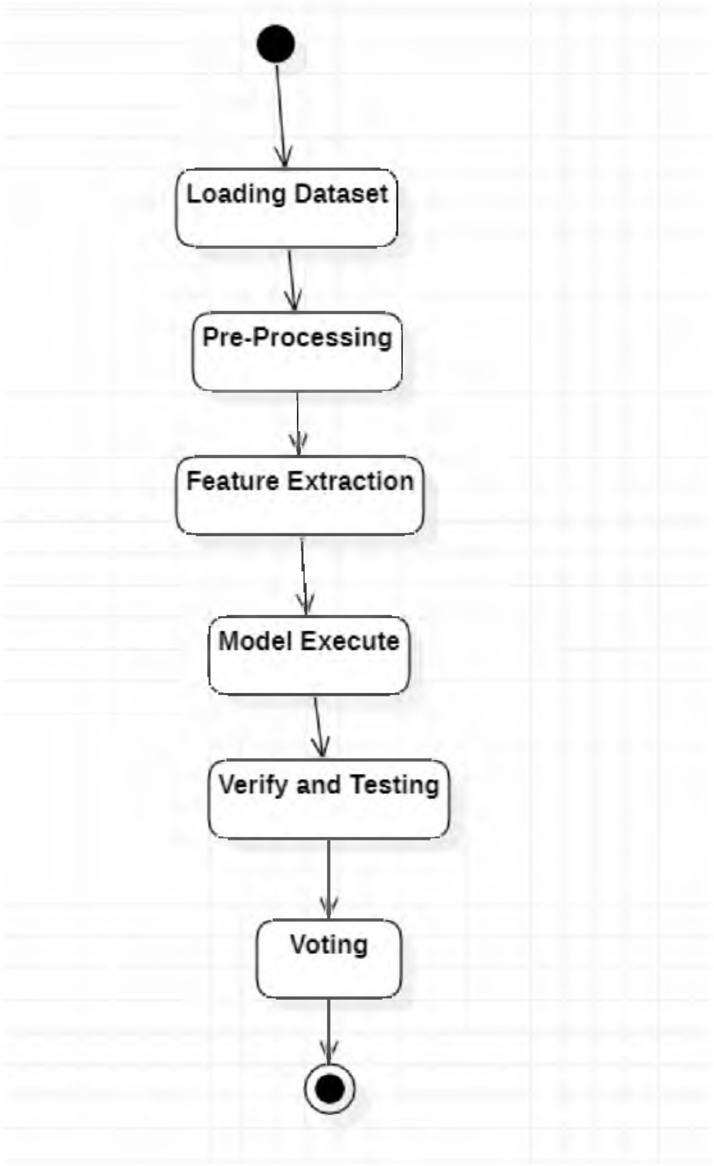


Fig:6.5 State Diagram

Collaboration Diagram:

A collaboration diagram groups together the interactions between different objects. The interactions are listed as numbered interactions that help to trace the sequence of the interactions. The collaboration diagram helps to identify all the possible interactions that each object has with other objects

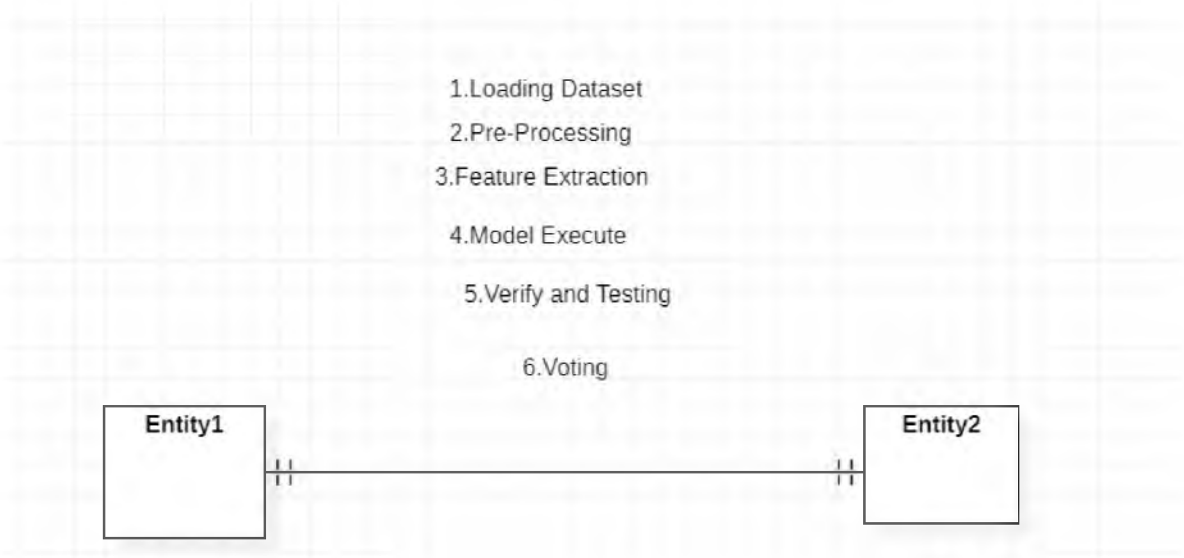


Fig:6.6 Collaboration Diagram

Flow Chat

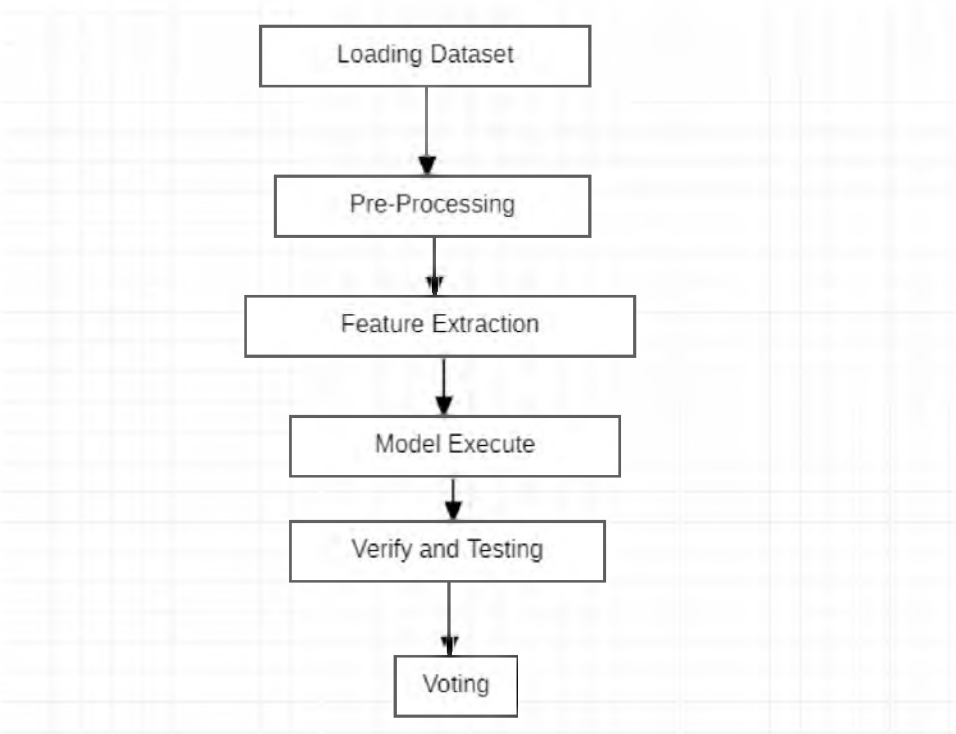


Fig:6.7 Flow Chat Diagram

6.5 SAMPLE CODE

```
main = tkinter.Tk()
main.title("Iris Recognition using Machine Learning Technique") #designing main screen
main.geometry("1300x1200")
global filename
global model
def getIrisFeatures(image):
global count
img = cv2.imread(image,0)
img = cv2.medianBlur(img,5)
cimg = cv2.cvtColor(img,cv2.COLOR_GRAY2BGR)
circles=cv2.HoughCircles(img,cv2.HOUGH_GRADIENT,1,10,param1=63,param2=70, minRadius=0,maxRadius=0)
if circles is not None:
height,width = img.shape
r = 0
mask = np.zeros((height,width), np.uint8)
for i in circles[0,:]:
cv2.circle(cimg,(i[0],i[1]),int(i[2]),(0,0,0))
cv2.circle(mask,(i[0],i[1]),int(i[2]),(255,255,255),thickness=0)
blank_image = cimg[:int(i[1]),:int(i[1])]
masked_data = cv2.bitwise_and(cimg, cimg, mask=mask)
_,thresh = cv2.threshold(mask,1,255,cv2.THRESH_BINARY)
contours=cv2.findContours(thresh,cv2.RETR_EXTERNAL,cv2.CHAIN_APPROX_SIMPLE)
x,y,w,h = cv2.boundingRect(contours[0][0])
crop = img[y:y+h,x:x+w]
r = i[2]
cv2.imwrite ("test.png",crop)
else:
count = count + 1
miss.append(image)
```

```
return cv2.imread("test.png")
def uploadDataset():
global filename
filename = filedialog.askdirectory(initialdir=".")
text.delete('1.0', END)
text.insert(END,filename" loaded\n\n");
def loadModel():
global model
text.delete('1.0', END)
X_train = np.load('model/X.txt.npy')
Y_train = np.load('model/Y.txt.npy')
print(X_train.shape)
print(Y_train.shape)
text.insert(END,'Dataset contains total '+str(X_train.shape[0])+' iris images from
'+str(Y_train.shape[1])+"\n")
if os.path.exists('model/model.json'):
with open('model/model.json', "r") as json_file:
loaded_model_json = json_file.read()
model = model_from_json(loaded_model_json)
model.load_weights("model/model_weights.h5")
model._make_predict_function()
print(model.summary())
f = open('model/history.pckl', 'rb')
data = pickle.load(f)
f.close()
acc = data['accuracy']
accuracy = acc[59] * 100
text.insert(END,"CNN Model Prediction Accuracy = "+str(accuracy)+"\n\n")
text.insert(END,"See Black Console to view CNN layers\n")
```

CHAPTER-7

SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

TYPES OF TESTS

Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested.

Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box. you cannot "see" into it. The test provides inputs and responds to outputs without considering how the software works.

7.1 UNIT TESTING

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

7.2 INTEGRATION TESTING

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

7.3 ACCEPTANCE TESTING

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

7.4 TEST CASES:

S.no	Test Case	Excepted Result	Result
1	Upload the dataset	Dataset uploaded successfully	pass
2	Generate CNN model	CNN model Generated successfully	pass
3	Load CNN model	CNN model Loaded successfully	pass
4	Generate loss and Accuracy graph	loss and Accuracy graph Generated successfully	pass
5	Upload Test image	Test images uploaded successfully	pass
6	Recognize image	image Recognize successfully	pass
7	Recognize iris through image	iris Recognized through image successfully	pass

Table:7.1 Test Cases 1

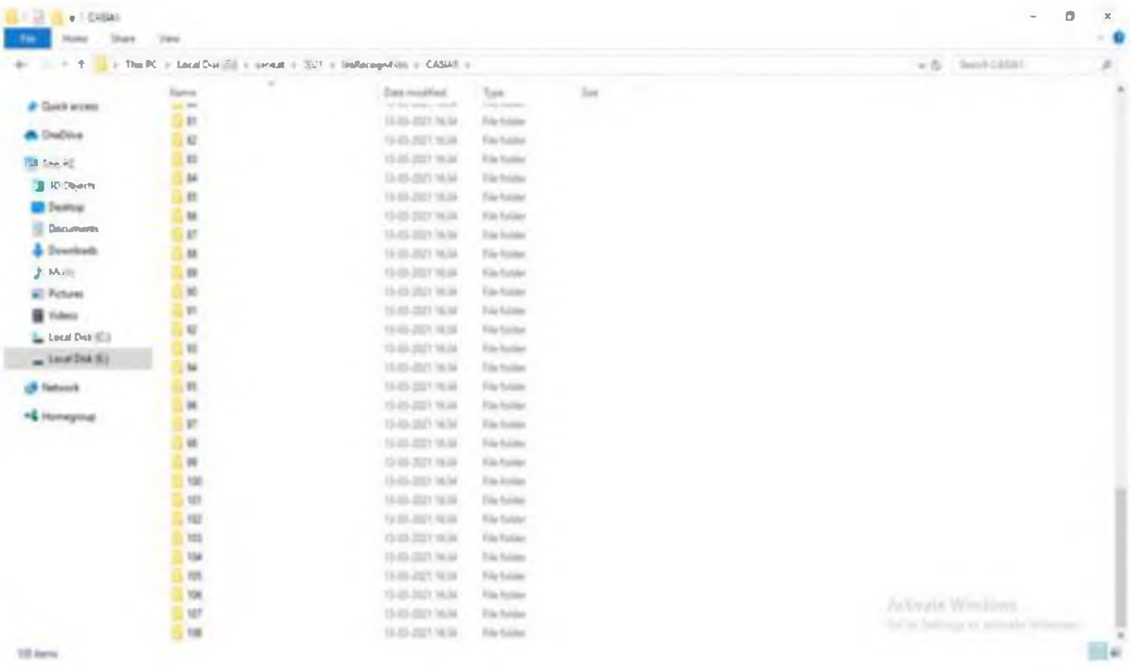
S.no	Test Case	Excepted Result	Result	Remarks(IF Fails)
1.	Loading Dataset	Loading Dataset Success	Pass	Image must be resize resolution will get better results
2.	Preprocessing	Loading the preprocessing	Pass	Loading the preprocessing Dataset
3.	Feature extractment	Processing Feature extractments	pass	Processing Feature extractments dataset
4.	Training and testing slipt	Detected images draw square and writing stress emotions	Pass	Images must be clearly to detect facial expression
5.	Model execution	PyImagelibaray will load the process and start the live	Pass	If library not available then failed
6.	Algorithm	If tensorflow not installed then it will fail	Pass	Depends on system configuration and tensorflow library
7.	Results	Load the dataset and process the Algorithm	Pass	The dataset must be media folder
8.	Predict Train and Test data	Predicted and original salary will be displayed	Pass	Trains and test size must be specify otherwise failed

Table:7.2 Test Cases 2

CHAPTER-8

OUTPUT SCREENS

8.1 DETERMINATION OF THE INFORMATION THROUGH THE CASIA IRIS DATASET



Name	Date modified	Type	Size
81	13-03-2021 16:34	File folder	
82	13-03-2021 16:34	File folder	
83	13-03-2021 16:34	File folder	
84	13-03-2021 16:34	File folder	
85	13-03-2021 16:34	File folder	
86	13-03-2021 16:34	File folder	
87	13-03-2021 16:34	File folder	
88	13-03-2021 16:34	File folder	
89	13-03-2021 16:34	File folder	
90	13-03-2021 16:34	File folder	
91	13-03-2021 16:34	File folder	
92	13-03-2021 16:34	File folder	
93	13-03-2021 16:34	File folder	
94	13-03-2021 16:34	File folder	
95	13-03-2021 16:34	File folder	
96	13-03-2021 16:34	File folder	
97	13-03-2021 16:34	File folder	
98	13-03-2021 16:34	File folder	
99	13-03-2021 16:34	File folder	
100	13-03-2021 16:34	File folder	
101	13-03-2021 16:34	File folder	
102	13-03-2021 16:34	File folder	
103	13-03-2021 16:34	File folder	
104	13-03-2021 16:34	File folder	
105	13-03-2021 16:34	File folder	
106	13-03-2021 16:34	File folder	
107	13-03-2021 16:34	File folder	
108	13-03-2021 16:34	File folder	

Fig:8.1 CASIA IRIS Dataset

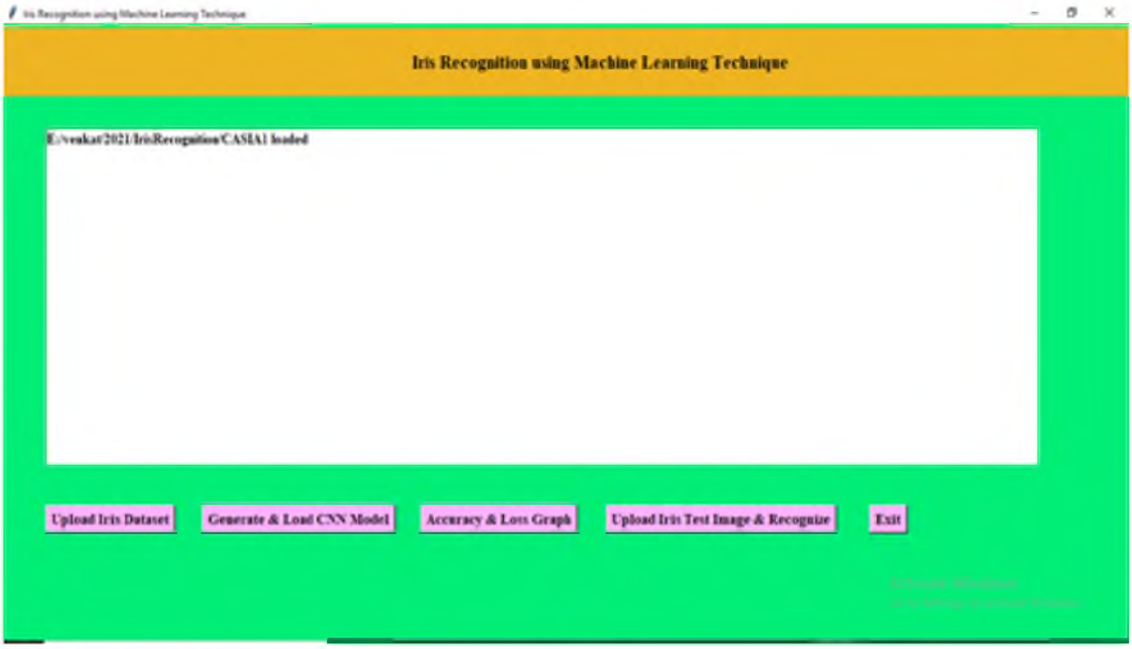
(Source: Provided)

The above figure has been shown in the dataset with the particular user id of 108 persons. The entire project has been mainly done for the best purpose of recognizing all people with respect to Iris. The entire implementation has been mainly done by the "CASIA iris dataset". The dataset must contain the particular images of 108 numbers of peoples of the place (Luz, 2019). With respect to this particular dataset the system user can be able to attend the proper training on the "convolutional neural network (CNN)" model for all the members of the organization. With the help of this "convolutional neural network (CNN)" model the system user can predict and recognize all the persons. To provide the appropriate training on the "convolutional neural network (CNN)" model for extracting the respective iris factors through the Hough Circles algorithm to describe all the iris cycles from the respective eye images.

Implementation of the Hough Circles algorithm for extracting the features

The Hough circles transform is the standard values of the "computer vision algorithm" for determination of the various types of parameters within the system. The system has determined various types of aspects based on the simple types of geometric objects like the circles and lines for representation of the image. The "circular Hough transform" has been mainly employed for deducing all the centre coordinates and radius of the pupil regions and the iris regions. The automatic segmentation algorithm has been mainly based on the proper notification of the Hough transformation through Wildes (Lv et al. 2018). The "circular Hough transform" is mainly involved for generating the particular edge map with respect to the canny edge detection process. This method is very much applicable and necessary for completing the entire task of finding out the respective iris from the image.

8.2 GENERATING THE CONVOLUTIONAL NEURAL NETWORK (CNN) MODEL FROM THE PROVIDED DATASET


Iris Recognition using Machine Learning Technique
Iris Recognition using Machine Learning Technique
Upload Iris Dataset
Generate & Load CNN Model
Accuracy & Loss Graph
Upload Iris Test Image & Recognize
Exit

E:/venkat/2021/IrisRecognition/CASIA1 loaded

Fig:8.2 Screenshot of generation of the Convolutional Neural Network (CNN) model (Source: Provided)

The above figure has been mainly described the loading process of the respective dataset. From this particular dataset the iris image can be easily generated by the "convolutional neural model (CNN)".

8.3 GENERATION OF THE LOSS GRAPH AND ACCURACY CHECK

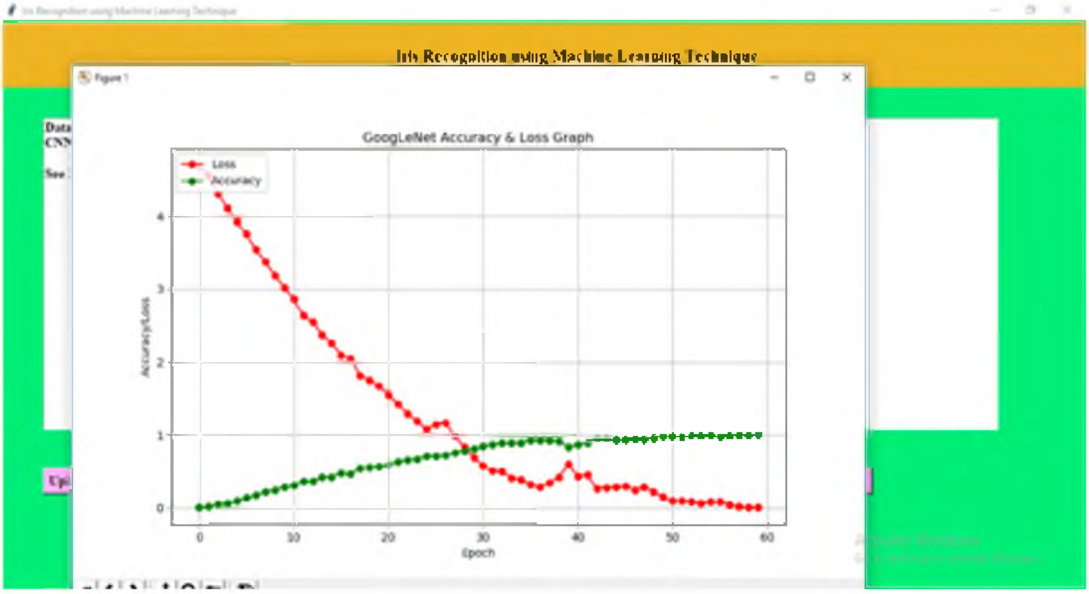


Fig:8.3 Generation of the LOSS Graph and accuracy check

(Source: Provided)

All the under attached image have been described the actual process of the iris recognition system of all the 108 peoples of the organization. The screenshots are attached below.

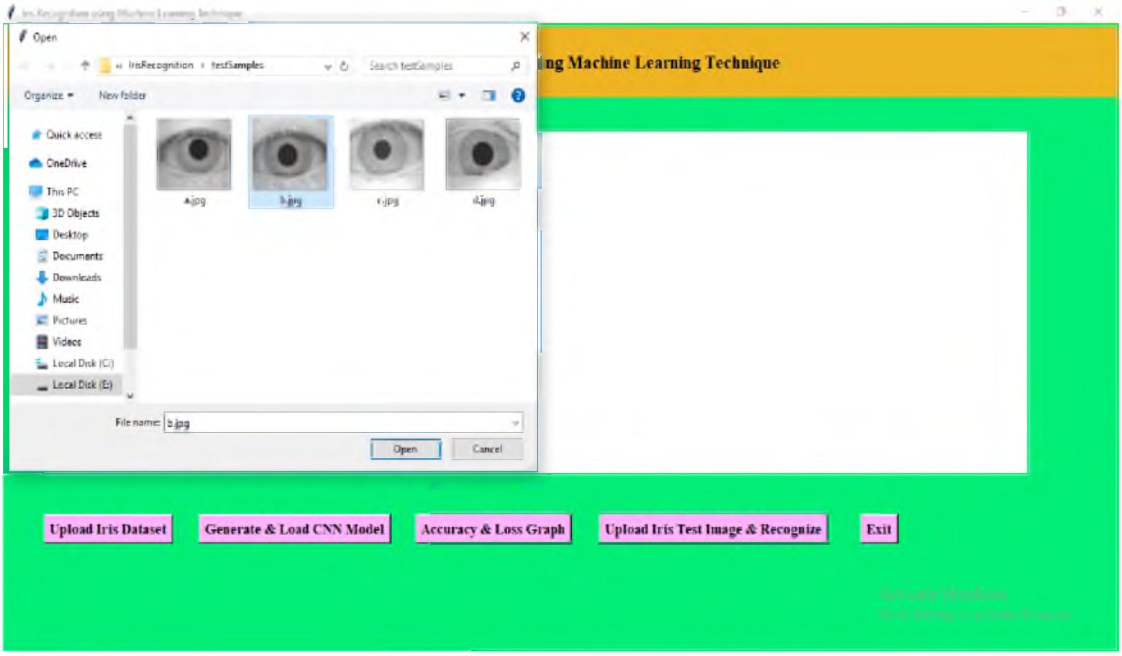
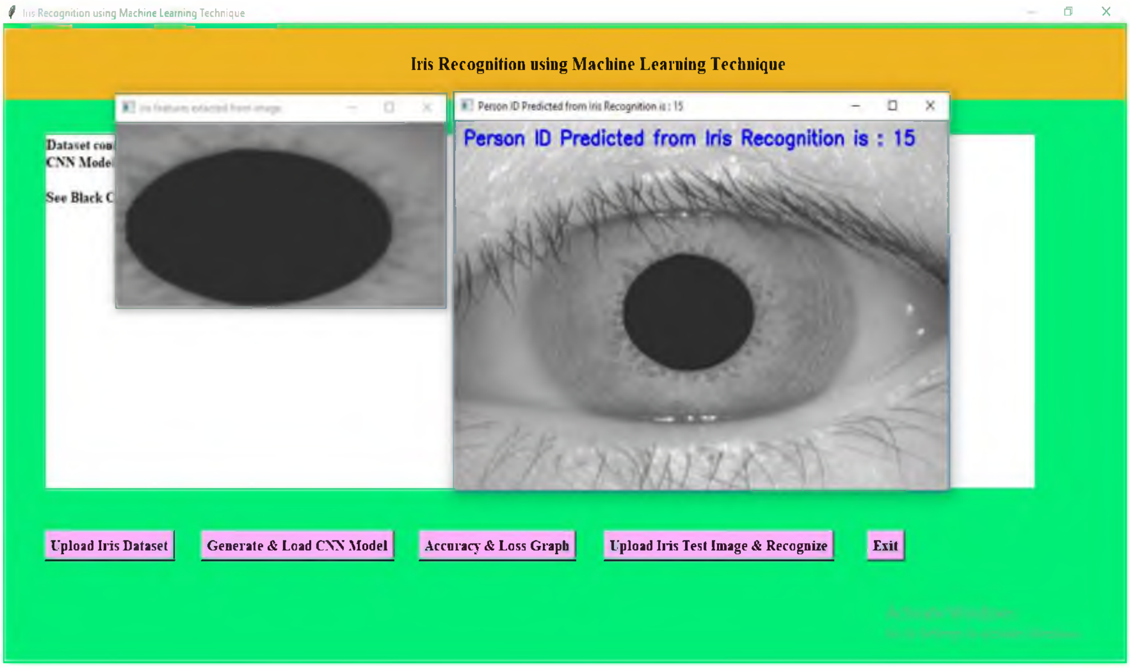


Fig:8.4 Recognition process

(Source: Provided)

The above figure has been provided the accurate loss graph from the provided dataset and the check the accuracy level through the “convolutional neural model (CNN)”. In this case the dotted red line has been represented as the loss value and factors of “convolutional neural model (CNN)”. And from the graph it has been properly proved that the iteration loss in the initial cases is greater than 3.9%. But with respect to the increased value of epoch, the loss value has been reduced to the value of zero. In the figure, the green line has been represented with the proper value of accuracy. In the graph, the X-axis has been shown the actual value of epoch and the Y-axis has been shown the proper value of accuracy and respective loss values. With the help of all these values, the system user can be easily able to easily recognize the proper ID from the iris picture through the “convolutional neural model (CNN)” model. The results that have been generally out from the model are totally correct for recognition of human identification.



Iris Recognition using Machine Learning Technique

Iris Recognition using Machine Learning Technique

Person ID Predicted from Iris Recognition is : 15

Person ID Predicted from Iris Recognition is : 15

Dataset con
CNN Mode

See Black C

Iris features extracted from image
Upload Iris Dataset
Generate & Load CNN Model
Accuracy & Loss Graph
Upload Iris Test Image & Recognize
Exit

Fig:8.5 Recognition process

(Source: Provided)

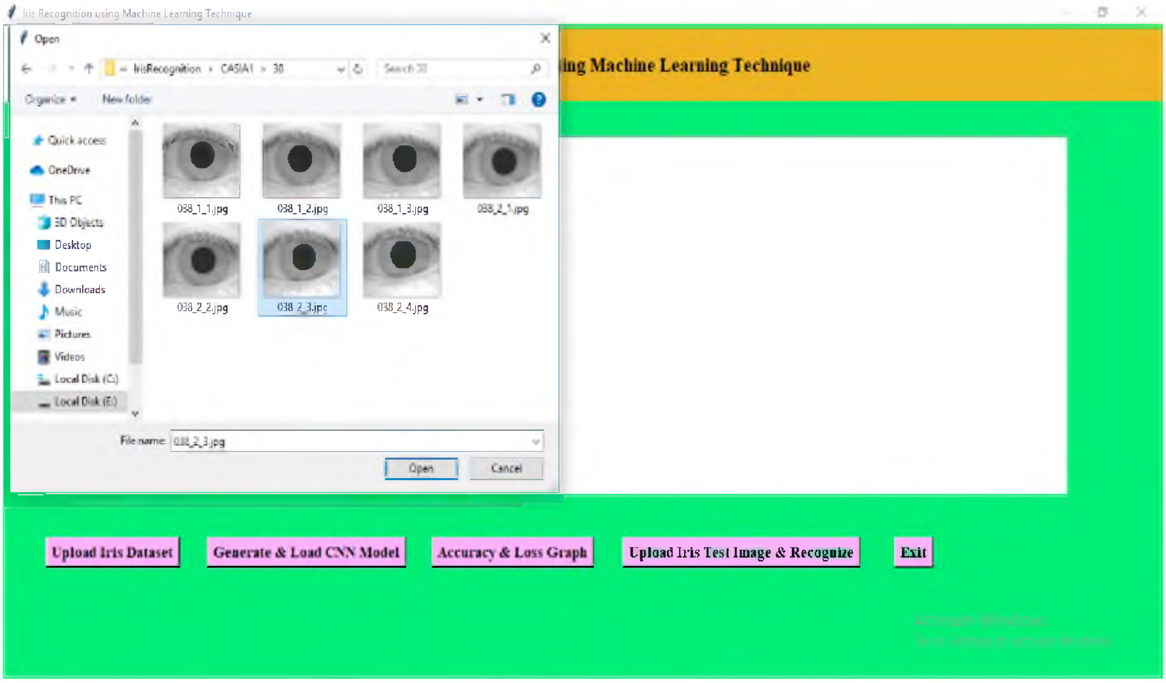
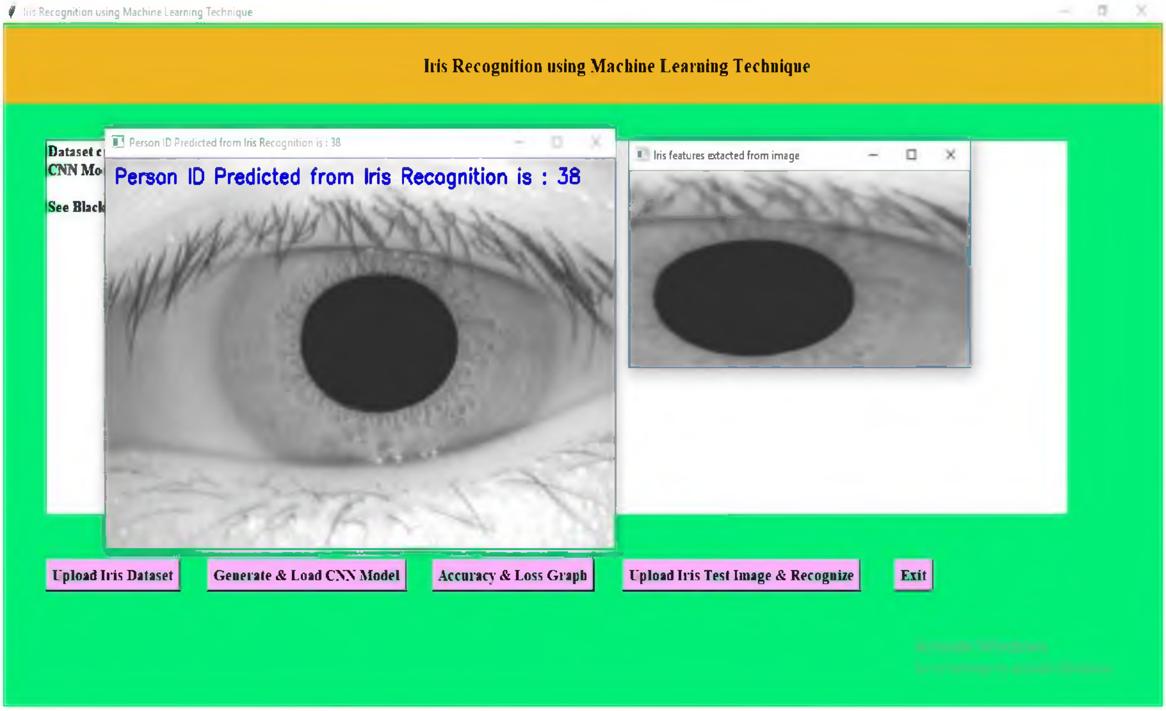


Fig:8.6 Recognition process

(Source: Provided)


Iris Recognition using Machine Learning Technique
Iris Recognition using Machine Learning Technique
Dataset c
CNN Mo
See Black
Person ID Predicted from Iris Recognition is : 38
Iris features extacted from image
Upload Iris Dataset
Generate & Load CNN Model
Accuracy & Loss Graph
Upload Iris Test Image & Recognize
Exit

Person ID Predicted from Iris Recognition is : 38

Fig:8.7 Accurate recognition image

(Source: Provided)

With respect to all these screenshots the possible way for getting the accurate iris recognition process of employee can be easily get.

CHAPTER-9

CONCLUSION AND FUTURE ENHANCEMENT

This particular piece of the study explains the critical technologies and software that are important to conduct the entire research work. This chapter helps to provide a thorough insight into the methods and approaches that are followed for completing the entire research. The chapter also discusses the importance of data collection and how the data is collected for any kind of software research work, as it can be seen from the above chapter that the entire research work has been conducted by maintaining all the standard ethics. However, there are some limitations that this particular research has faced, which will be tried to overcome during the future upgrade. Moreover, the chapter highlights the essential parts such as “convolution neural network (CNN)” architecture and algorithms, processing and extraction of images, decision making, etc., that have been included.

REFERENCES

[1] Aman Jatain, Yojna Arora, Jitendra Prasad, Sachin Yadav, Konark Shivam,Department of Computer Science, Amity University, Gurgaon, Haryana,Design and Development of Biometric Enabled Advanced Voting System May 2020.

[2] Chandra Keerthi Pothina , Atla Indu Reddy, Ravikumar CV,Electronics and Communication Engineering, Vellore Institute of Technology, Vellore,Smart Voting System using Facial Detection April 2020.

[3] Jayapriya J, Roghini M, Jayanthi S,Department of CSE, Agni College of Technology, Tamil Nadu, India, A Survey on Biometric Voting System using Iris Recognition Mar 2020.

[4] Kennedy Okokpujie, Samuel Ndueso John, Etinosa NomaOsaghae,Charles Ndujiuba, Department of Electrical and Information Engineering,Covenant University, Ota, Ogun State, Nigeria, An Enhanced Voters Registration And Authentication Application Using Iris Recognition Technology February 2019.

[5] Md. Mahiuddin,Department of Computer Science and Engineering International Islamic University Chittagong

[6] (IIUC) Chittagong, Bangladesh,Design a Secure Voting System Using Smart Card and Iris Recognition February 2019.

[7] PayalDeora,Abhishek kumar, Gayathri.M,Malathy.C,CSE, SRM Institute of Science and Technology, Chennai,Secured Voting System with Multimodal Biometric Technique using ANN,2018.

[8] Saravanan.N, Pavithra.K, Nandhini.C,Dept. of MCA Priyadarshini Engineering College, vaniyambadi, Tamilnadu, India,Iris Based EVoting System Using Aadhar Database,April-2017.

[9] Sarankumar.V, Sasikumar.M, Ramprabu.K, Sathishkumar.A, Mr.S.Gladwin Moses Stephen,Department of Electronics and Communication Engineering, Akshaya College of Engineering and Technology, Coimbatore, March-2017 .