ATTENDANCE SYSTEM USING LDA BASED FACE RECOGNISATION

A MINI PROJECT REPORT

Submitted by

SARMILAA R ***(211422104440)***

SAI SRUTHI N ***(211422104416)***

In a partial fulfillment for the award of the degree

Of

BACHELOR OF ENGINEERING

In

COMPUTER SCIENCE AND ENGINEERING

PANIMALAR ENGINEERING COLLEGE (An Autonomous Institution, Affiliated to Anna University, Chennai)

OCTOBER-2024

PANIMALAR ENGINEERING COLLEGE

(An Autonomous Institution, Affiliated to Anna University, Chennai)

BONAFIDE CERTIFICATE

Certified that this project report **"ATTENDANCE SYSTEM USING LDA BASED FACE RECOGNISATION"** is the bonafide work of **SARMILAA R (211422104440) & SAI SRUTHI N (211422104416)** who carried out the project work under my supervision.

SIGNATURE

Dr.L.JABASHEELA ,M.E.,Ph.D .,
HEAD OF THE DEPARTMENT

DEPARTMENT OF CSE,
PANIMALAR ENGINEERING COLLEGE ,
NASARATHPETTAI,
POONAMALLEE,
CHENNAI-600 123.

SIGNATURE

Dr.M.SHYMALA DEVI ,M.E..,Ph.D.,
PROFESSOR

DEPARTMENT OF CSE,
PANIMALAR ENGINEERING COLLEGE,
NASARATHPETTAI,
POONAMALLEE,
CHENNAI-600 123.

Certified that the above candidates were examined in the End Semester Project Viva-Voce Examination held on.....

INTERNAL EXAMINER

EXTERNAL EXAMINER

DECLARATION BY THE STUDENT

We **SARMILAA R (211422104440) & SAI SRUTHI N (211422104416)** hereby declare that this mini project report titled **"ATTENDANCE SYSTEM USING LDA BASED FACE RECOGNISATION" ,** Under the guidance of **Dr.M.SHYMALA DEVI ,M.E..,Ph.D.,** in the original work done by us and we have not plagiarized or submitted to any other degree in university by us.

1.SARMILAA.R

2.SAI SRUTHI N

ACKNOWLEDGEMENT

We express our deep gratitude to our respected Secretary and Correspondent **Dr.P.CHINNADURAI, M.A., Ph.D.** for his kind words and enthusiastic motivation, which inspired us a lot in completing this project.

We would like to extend our heartfelt and sincere thanks to our Directors **Tmt.C.VIJAYARAJESWARI**, **Thiru.C.SAKTHIKUMAR,M.E.,** and **Tmt. SARANYASREE SAKTHIKUMAR B.E.,M.B.A.,** for providing us with the necessary facilities for completion of this project.

We also express our gratitude to our Principal **Dr.K.Mani, M.E., Ph.D.** for his timely concern and encouragement provided to us throughout the course.

We thank the HOD of CSE Department, **Dr.L.JABASHEELA , M.E .,Ph.D.,** for the support extended throughout the project.

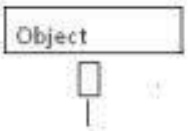
We would like to thank **Dr. M. Shyamala Devi, M.E., Ph.D.,** Professor of CSE for advice and suggestions for the successful completion of the project.

TABLE OF CONTENTS

ABSTRACT

Face recognition is one of the biometric technique that involves safe database access. Face recognition demands a larger database to hold the data. Mostly, outsourcing concept can only be implemented in large-scale companies. As a result of this, many traditional third-party companies handling the services and creating goods. In order to quickly deliver the results, this system can be conveniently used and accessed on remote computers, laptops in all types of institutions. This project analysis is to refuse the outsourcing concept and facilitate a system in which a student information can be accessed through facial recognition, able to record attendance. For that, we implement the algorithm known as the Normal discriminant function, which was utilised to circumvent this. This system is advantageous because it makes it simpler to know the list of pupils by face data entry and that data is controlled by the source operator. Face recognition, as we all know, entails two steps: first, faces must be detected, and then the faces must be identified using the database that already exists. To prevent impersonation, we're going to deploy technologies notably , retinal scanning and thermal imaging, 2D and 3D mapping. The PCA technique is being used in the feature extraction approach since it performs better in situations when it is small scale. The LDA algorithm will be used to reduce image dimensionality. This system helps us to store the student images based on Eigen and dimensional values and can be stored as 100 images for each student . Also this system provides the student attendance details with specified time and date.

LIST OF SYMBOLS

S.NO	SYMBOL NAME	NOTATION	DESCRIPTION
1.	Initial Activity		This shows the Starting point or first activity of flow.
2.	Final Activity		The end of the Activity diagram is shown by a bull's eye symbol.
3.	Activity		Represented by a rectangle with rounded edges.
4.	Decision		A logic where a decision is to be made.
5.	Use Case		Describe the interaction between a user and a system.
6.	Actor		A role that a user plays with respect to system.
7.	Object		A Real time Entity.
8.	Message		To send message between the life of an object.

LIST OF FIGURES

CHAPTER 1

1.1 INTRODUCTION

Attendance maintanence is an important task in all the institution to check the performance of the students. Every institute has its own way to do so . Some use the old paper or file based approach and some have adopted methods of automatic attendance using some biometric techniques. There are many automatic methods available for this purpose. Face recognition is one of the biometric technique . It's frequently used in identity authentication, attendance systems ,mobile lock patterns and so on .It is considered to be one of the most successful application of image analysis and processing ,this is the main reason behind the great attention it has been govern in the past several years.

1.2 BACKGROUND

Face recognition is crucial in daily life in order to identify family, friends or someone we are familiar with. We might not perceive that several steps have actually taken in order to identify human faces. Human intelligence allows us to receive information and interpret the information in the recognition process. We receive information through the image projected into our eyes, by specifically retina in the form of light. Light is a form of electromagnetic waves which are radiated from a source onto an object and projected to human vision. Robinson-Riegler, G., & Robinson-Riegler, B. (2008) mentioned that after visual processing done by the human visual system, we actually classify shape, size, contour and the texture of the object in order to analyse the information. The analysed information will be compared to other representations of objects or face that exist in our memory to recognize. In fact, it is a hard challenge to build an automated system to have the same capability as a human to recognize faces.

However, we need large memory to recognize different faces, for example, in the Universities, there are a lot of students with different race and gender, it is impossible to remember every face of the individual without making mistakes. In order to overcome human limitations, computers with almost limitless memory, high processing speed and power are used in face recognition systems. The human face is a unique representation of individual identity. Thus, face recognition is defined as a biometric method in which identification of an individual is performed by comparing real-time capture image with stored images in the database of that person.

CHAPTER 2

LITERATURE SURVEY

1. E.Varadharajan et.al explained the automatic Attendance Management system based on Face Detection. The author describes how faces are sensed and then cut, before which background subtraction is performed on the image in order to improve system performance efficacy. The erudite authors recommend the use of Eigen face for its simplicity and quality of performance in facial recognition. The document also concluded with the observation that in the case of women, the detection and recognition rate of the face with a veil was 45% and 10%, while in the case of women it was 93% and 87% without the veil. The identification and recognition levels, on the other hand, were 79% and 65% for bearded men.

Online International Conference on Green Engineering and Technologies (IC-GET) 2016

2. Smit Hapani et.al has magnified the system which approbated the model which contributes face distinguishing. Haar classifiers which uses cascade approach and followed by recognition which uses Fisher face. The system optimally offers efficacy up to 50% within 15 pupils when modelling with more than one face with respective to variations such as cap, spectacles. The proposed system makes use of classroom through video source, and these resulting frames are used to identify the faces. Thus, by following the procedures there by increasing the rate and accuracy of overall model.

Fourth International Conference on Computing Communication Control and Automation (ICCUBEA) 2018

3. Sathyanarayana n et.al launched Automated Attendance system using facial recognition. The system specifies algorithms such as Jones' Purple algorithm for face detection and MSE (medium square error) face recognition. The document stated and elaborated about the system's level of security and accuracy improves as the number of training images increases. The machine is also checked for different face angles and alignment up to 60 degrees can be identified. It is observed that when the system is tested with an image of six students, the system recognizes five students with 70% efficiency.

International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE) ISSN: 0976-1353 Volume 25 Issue 6 – MAY 2018.

4. D. Nithya has introduced Automated Attendance System which works on MATLAB. Extraction of the functions is accomplished by analysing the main components. The Eigen facial approach is utilized for its ease, speed and learning ability. The difference between the training values and the test image is calculated using the Euclidean distance.

International Journal of Engineering Research & Technology (IJERT) Vol. 4 Issue 12, December-2015

5. Professor Arun Katara et.al [17] implemented a real time assistance system which can perform multiple facial recognition using the Raspberry PI model and Raspberry PI camera. For face pin-pointing it uses Open-CV libraries and for face recognition the combination of feature extraction methods such as principal component analysis along with LBP is implemented. Since the system can identify faces from a distance of 4 feet to 7 feet, the facial recognition efficacy is limited, and is suggested to be improved. Capture a video of classroom using a video camera and followed by processing images for facial recognition.

International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169

6. Nilesh D. Veer et.al an automatic attendance system has been developed in which a video is collected as input. frames are captured when there is human presence detected. For face detection, Viola Jones is used, and PCA is used for face recognition, which also uses LBP for threshold purposes. The facial recognition rate is nearly 100% for a small number of students and the attendance of the student is recorded along with the entry time of the student.

IEEE International Conference On Recent Trends in Electronics Information Communication Technology, May 20-21, 2016, India

7. Refik Samet et.al has implemented a fully cell phone automatic attendance system. This is achieved using the Viola-Jones algorithm along with Ada-boost training for face finding, since according to the authors, they should work better in the real-life scenario. For the purposes of recognition, the Euclidean distance was determined for the 3 recognition methods, namely its Eigen face, Fisher face and LBP. A comparison of precision was made for all of the above-mentioned recognition techniques. The smartphone application was developed for the automatic attendance generating system.

2017 International Conference on Cyberworlds

8. Tripathi et.al claimed a real time system which can follow through the presence of the students in a classroom. The necessary supported images for this model was brought at a constant rate through a webcam until the system is turned off. The author scanned through several techniques in order for face detection and encourage them in recognition. Pupils are distinguished with the help of the Ada boost and Haar cascade classifier. Although for face exposer and recollection, the author made use of OpenCV libraries but still for in depth insight he made a quick use of PCA and LDA. The document also emphasized about the difference between LDA and PCA. In the end author confidently inclined towards the system's accuracy and noted that identification rate is entirely dependent on the database and the size of the used image.

International Journal of Artificial Intelligence & Applications (IJAIA), Vol.2, No.3, July 2011

9. Firoz Mahmud et.al approbated use of 2 database types including UMIST database and ORL database. PCA and LDA both are used for face knowing purposes. The accuracy of the face recognition is determined using the above listed algorithms, depending on the face alignment. It is observed that front aligned faces have a much better accuracy of recognition than those of face side alignment.

International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT) 2015

10. Krishna Dharavath et.al has produced excellent pre- processing results on a noisy image. The methods suggested for pre-processing are face cropping, resizing, normalizing & filtering. A low pass filter is used to eliminate components of high frequency noise. PCA, DCT (Discrete Cosine Transform) and combined Spatial and Frequency Domain approach are compared before and after pre-processing. The proposed combined form has the highest rate of face recognition and is not much influenced by pre-processing. The major drawback is that facial detection is performed before the pre- processing of image. In multiple face recognition system, this is not expected as the image needs to be pre-processed first before any face detection or recognition.

International Conference on Advances in Signal Processing (CASP) Jun 2016

11. Ms. Pooja Humbe et.al made use of 360-degree rotating camera for building the model which detects the pupils in the class. This system without the software such as XAMPP controller, NetBeans, Java Advance for the front-end and back-end with MySQL could have been impossible as stated by author. The characteristics of face are being brought by principal component analysis (PCA). Once registered, the record containing the names of students attended will be sent through email to parents and teachers.

International Journal on Recent and Innovation Trends in Computing and Communication Volume: 5 Issue: 12

12. Akshara Jadhav et.al prompted face encounter algorithm Viola Jones and face recognition PCA algorithm with support for machine learning and SVM for extraction functionality. The author also incorporated reprocessing which includes the histogram equalization of the facial image extracted and is scaled to 100x100. The use of neural networks for facial recognition has been shown, and we can see the possibility of a semi-supervised learning approach that uses facial recognition support vector machines for satisfactory results. The process followed after the face is recognized is the subsequent processing in which attendance is generated weekly or monthly and can be sent to parents or guardians.

International Research Journal of Engineering and Technology (IRJET) 2019

13. Shireesha Chintalapati et.al defined the Viola Jones Face Detection Algorithm. The paper stated that this algorithm offers better results in various lighting conditions and the authors have clubbed multiple Haar classifiers to achieve better output rates up to 30-degree angles. The preprocessing phase relates to the histogram equalization of the facial image obtained in which it is scaled down to 100x100. Images are converted to grayscale; the equalization of histograms is applied and images are scaled to size of 100x100. The system employed the LBPH algorithm to extract the characteristics and the SVM classifier for classification purpose This document used a 80 person database (NITW database) with approximately 20 images of each individual collected for the project. This document sets out some performance evaluation conditions when combining LBPH and distance classifier, the false positive rate is 25 %, the object distance for correct recognition must be 4 feet, the training time being 563 milliseconds, 95 %of recognition percentage for static images, the recognition percentage (real-time video) was 78 %, the occluded faces 2.3% In Microsoft Visual C and the EmguCV container the GUI is developed using the WinForms application.

IEEE International Conference on Computational Intelligence and Computing Research 2013

14. A. Majumdar and R.K Ward discussed how well they had done than PCA. To boost dispersion, they used the Fisher face subspace and LDA, and they also used KNN. Consequently, better results were obtained by using the Pseudo- Fisher facial technique. This article examines several methods that various authors consider to improve the rate of detection and recognition. The results show that Viola Jones, who uses Haar Cascade, is consistent in all the papers reviewed and offers a good detection rate whereas Fisher Face's LDA algorithm provides better performance and faster results.

International Conference of Acoustics, Speech and Signal Processing (ICASSP) 2008

15. Kennedy Okokpujie et.al describes a system that uses Viola Jones as a face detection tool and Fisher face algorithm for face recognition. Uses a webcam to build the database and to collect photos to process. It works well in good lighting conditions, but at different lighting conditions it decreases the face recognition rate (up to 54%). The system has access for the authority and the participants via the cell phone interface with the login credentials.

IEEE 3rd International Conference on Electro-Technology for National Development 2017

CHAPTER 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

Most of the existing system is to automate the attendance system by integrating the face recognition technology using Eigen-face database and Principle Component Analysis (PCA) algorithm with MATLAB GUI. The architecture of the system first, caputures the student image, pre process it, applied Eigen-face generated database then test the captured face image with Eigen-face image. When the similiarity distance test scored more than the threshold value of 0.3 then the face was not recognized finally attendance marking, was stored in Microsoft Excel sheet integrated with MATLAB GUI. The original face database consists of images of 15 persons each has 10 images with different position and direction.

LIMITATIONS OF EXISTING SYSTEM

- It is developed for a particular system so we implement this library OpenCV does not provide the same ease of use when compared to MATLAB. Open CV has a flann library of its own. This causes conflict issues when you try to use OpenCV library with the PCL library.
- Also the existing system only stores images of 15 persons upto 10 images per person. This result in delay of detecting the faces and also most of institutions has larger number of students so the existing system does not fit well for it .

3.2 PROPOSED SYSTEM

To overcome the drawbacks of the existing sytem, the proposed system was evolved. It main aims not to delay the time of detecting the face and also paves way to store more number of student images in the database. For that we implement a LDA (Linear-discriminant Algorithm) along with PCA and Eigen-face techniques in our system to overcomes all those drawbacks in the existing sytem. By using the LDA based system, a student face is stored as 100 different images in different postures also accommodates less storage space which makes great impact in duration of detecting the face of the particular student.

3.3 REQUIREMENT ANALYSIS

Requirement analysis is a technical specification of requirements for the software products. It is the first step in the requirement analysis process it lists the requirements of a particular software system including functional, performance and security requirements. The purpose of software requirements specification is to provide a detailed overview of the software project, its parameters and goals. This describes the project target audience and its users interface, hardware and software requirements.

3.3.1 SOFTWARE REQUIREMENTS

The software specification are the specification of the system. It should include both the specification and a definition of the requirements. It is a set of what the system should do rather than how it should do it. The software requirements provides the basis for creating the software requirement specification. It is useful in estimating cost, planning team activities, performing tasks and tracking the team’s progress throughout the development activity.

- Operating system: Windows 7 or 10
- Language : Python
- Libraries used : OpenCV-python, pillow pandas , OpenCV-contrib-pythonpymysql.
- Integrated development environment : Command prompt.

3.3.2 HARDWARE REQUIREMENTS

The Hardware requirements may serve as the basis for a contract for the implementation of the system and should therefore be a complete specification of the whole system. They are used by the software engineers as the starting point for the system design. It shows what the system do not and how it should be implemented.

- A standalone computer or laptop (8GB RAM or higher).
- High-quality wireless camera to capture images.
- Secondary memory to store all the images and database.

CHAPTER 4

SYSTEM DESIGN

.

ARCHITECTURE DIAGRAM

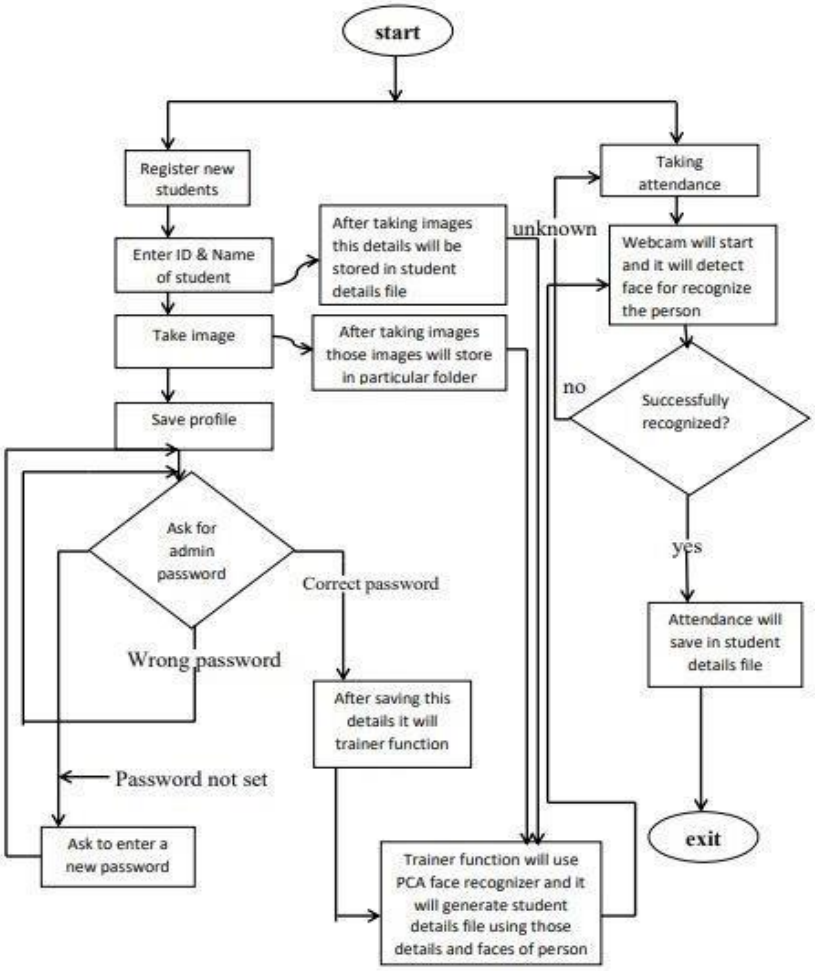


FIGURE 4.1 Architecture Diagram

Facial Recognition process generally includes three stages:

- Face Detection
- Feature Extraction

- Face Recognition

➢ Face Detection

It accepts the image as an input and checks if ‘Face’ appears in the image and calculates its position on the image. The output of this stage is ‘Patches’ which contains ‘Face’ and Face alignment is done which acts as pre-processing stage for Feature Extraction.

➢ Feature Extraction

Face Patch is transformed in to a set of Fiducial Points corresponding to their locations or it is transformed into vectors with specific dimension.

➢ Face Recognition

This step includes recognition of Face from the database. When the system receives Face image, it undergoes Face Detection and Feature Extraction process. Then, the features are compared with each Face in the Database using the nodal points on the Face.

By using the above processes the faces have been recognized. Here we once we have enrolled the new students in the application using the respective module, the students details can be stored in the database. Once we take the attendance the faces have been recognized and their attendance have been stored in the respective database (excel sheets).

4.2 UML DIAGRAM

4.2.1 USE CASE DIAGRAM

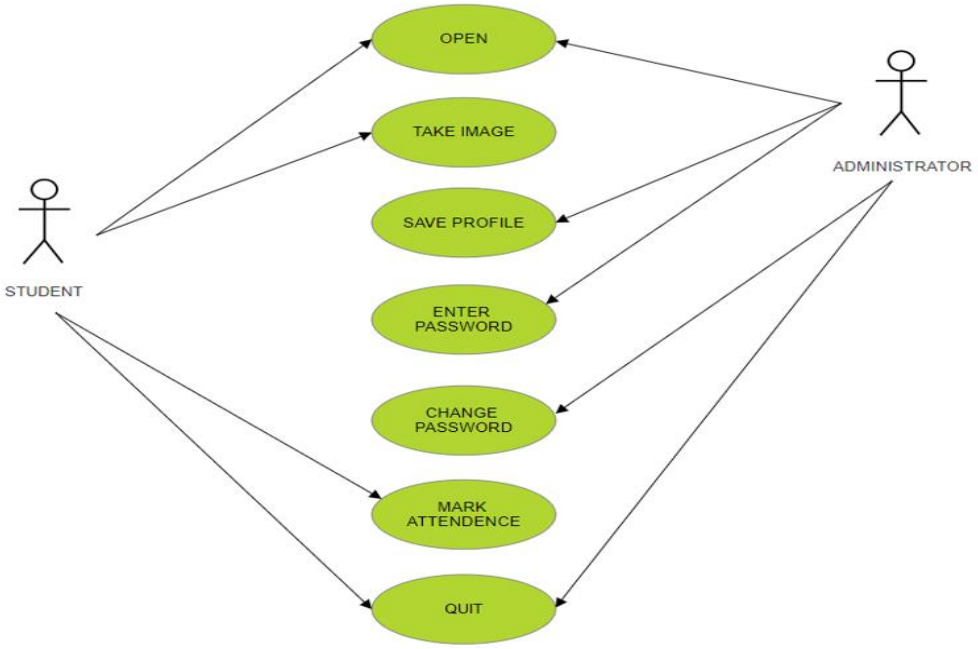


FIGURE 4.2 Use case diagram

4.2.2 CLASS DIAGRAM

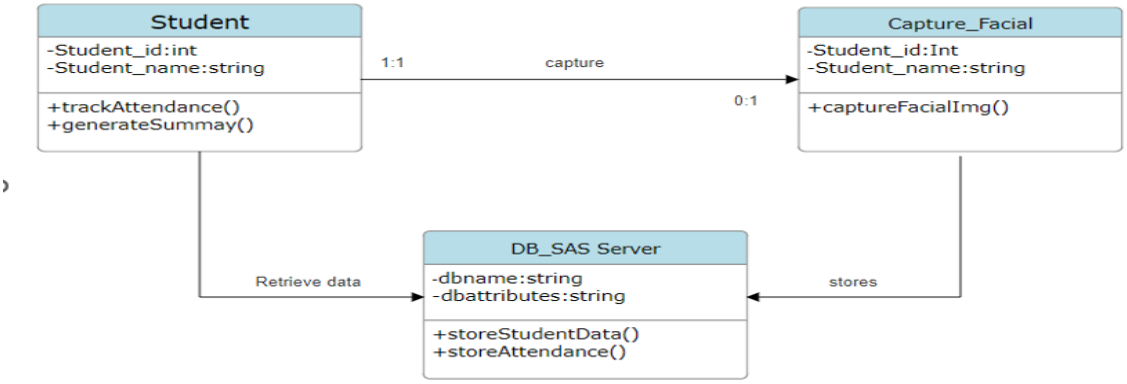


FIGURE 4.3 Class diagram

4.2.3 SEQUENCE DIAGRAM

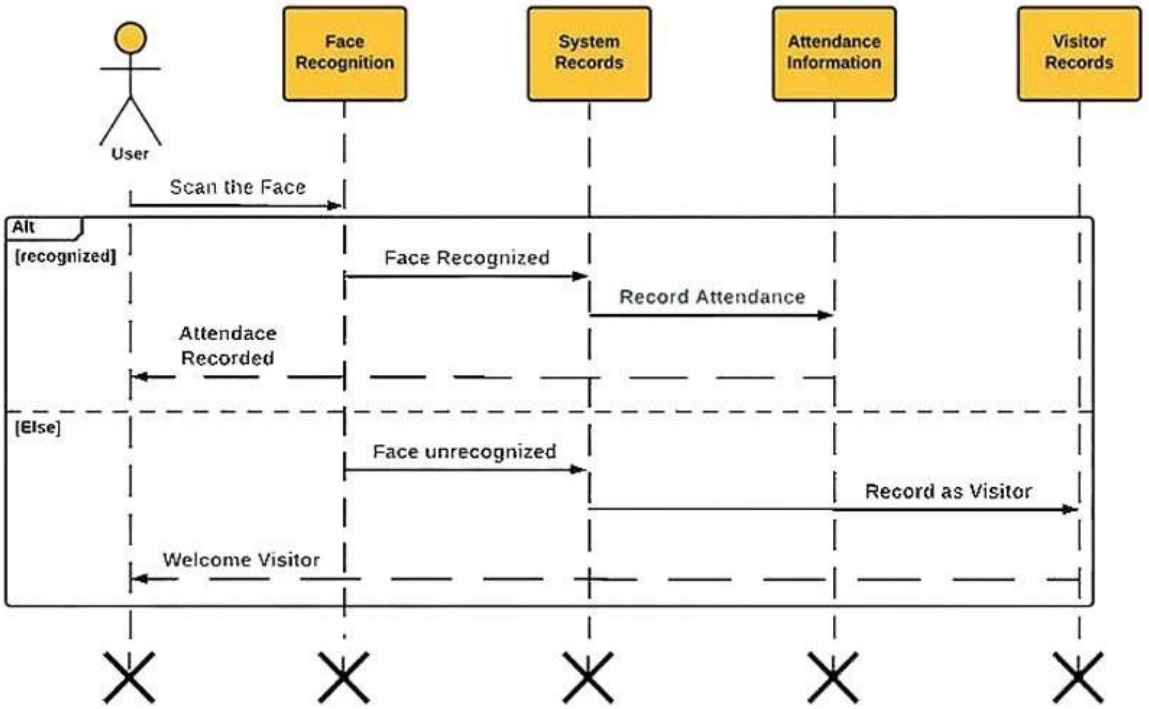
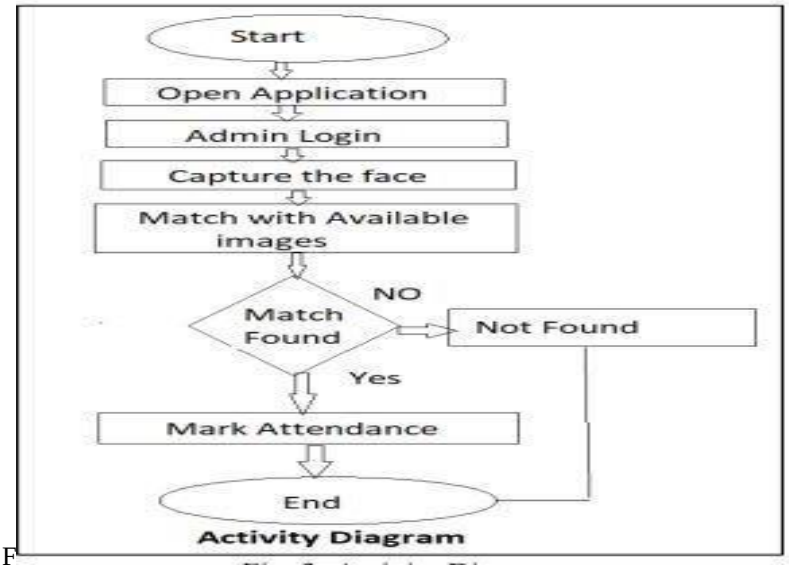


FIGURE 4.4 Sequence diagram

4.2.4 ACTIVITY DIAGRAM



F

FIGURE 4.5 Activity diagram

CHAPTER 5
SYSTEM IMPLEMENTATION

5.1 MODULE OVERVIEW

Module is a logical separation of functionality within a project. They are basically used for reusability and better code maintenance. There are four modules used here,

- Enrollment module
- Identification & Taking attendance module
- Verify password

5.2 MODULE DESCRIPTION

- **Enrollment Module**

  In this module, Administrator can add new students to the system by taking images of the student and provide student name and student id and then save students profile by verifying his authentication by providing the password. The images taken were stored to the database 100 images per student in different postures and expressions by using the algorithms we mentioned before.

- **Identification & Taking attendance module**

  In this module, attendance can be marked by clicking mark attendance button, a tab will open and take images of the student, detect and shows the name or id of that particular student. If the

student is not yet registered it shows the student name as unknown. If the student is already registered it saves the attendance of that particular student in a excel sheet with students name , id and time.

- **Change password**

In this module, Administrator can change the password for authentication by clicking the help option at left top corner of the page and then open the change password option. After opening, there will be a dialog box with three fields which contains old password, new password and confirm password . By providing the details administrator can able to change the password.

CHAPTER 6

SYSTEM TESTING

PERFORMANCE TESTING

A.ALGORITHM

The LDA-based Face Recognition Attendance Management System captures and preprocesses images to extract features for identifying individuals. Upon matching with the database, attendance is automatically logged and securely stored for administrator access.

B.MODEL EVALUATION

The model was evaluated using metrics such as accuracy, precision, recall, F1 score, processing time, false positive and negative rates, user satisfaction, and scalability to assess its performance. Results are detailed in Table II, highlighting the system's reliability and areas for improvement**.**

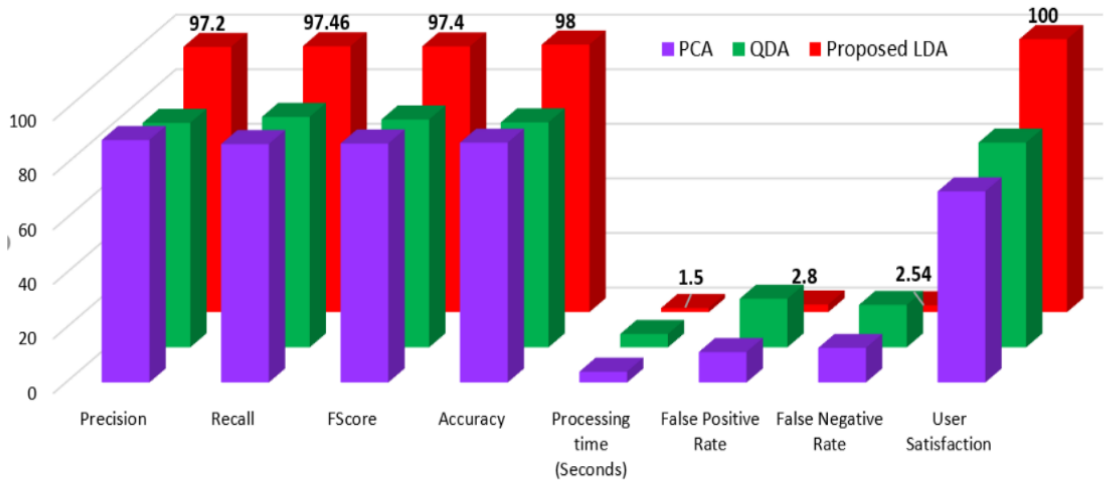


Figure 6 Performance Analysis of LDA Based Face Recognition Attendance Management System

.

CHAPTER 7

SCREENSHOTS

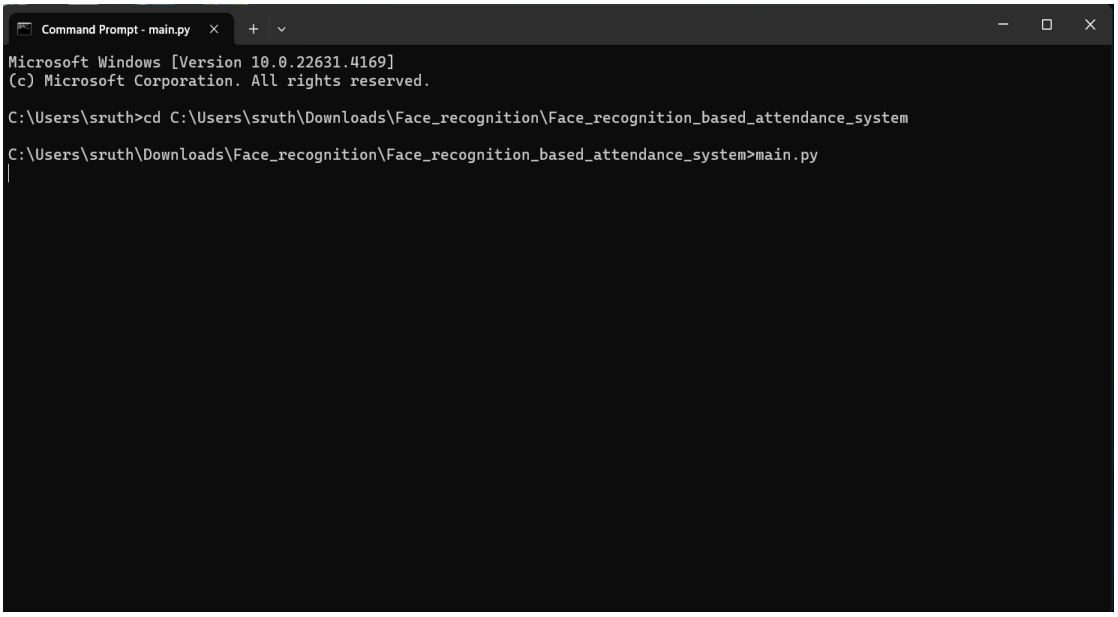


Figure 7.1 Execution page

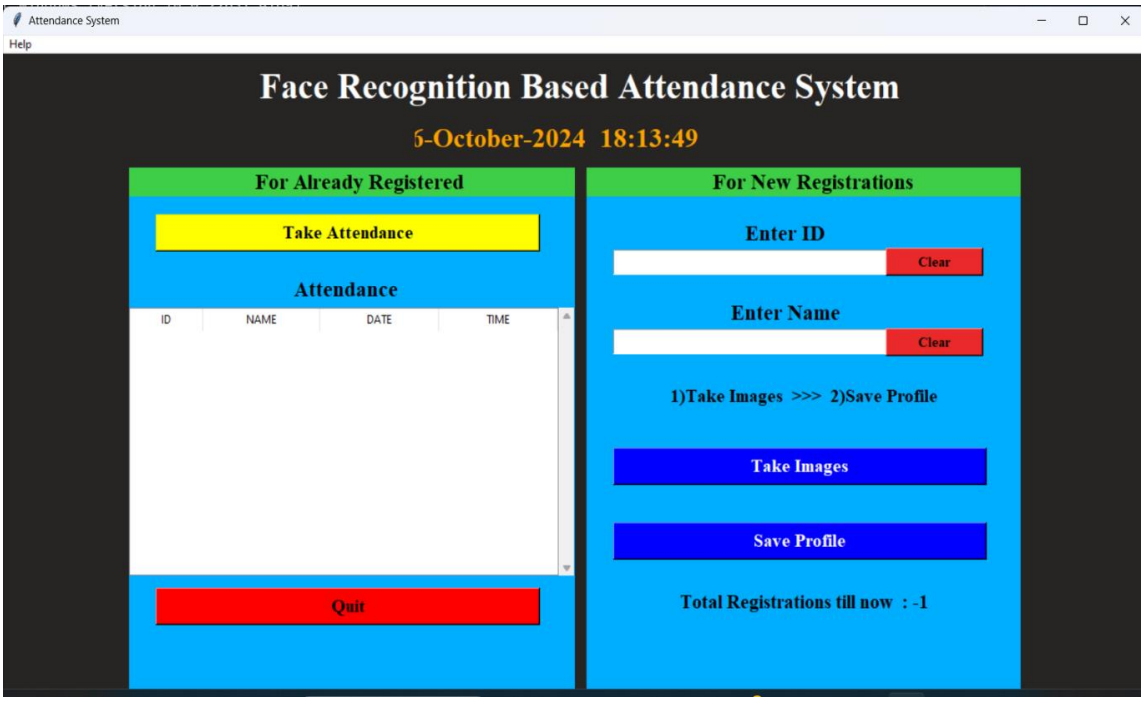


Figure 7.2 Home page

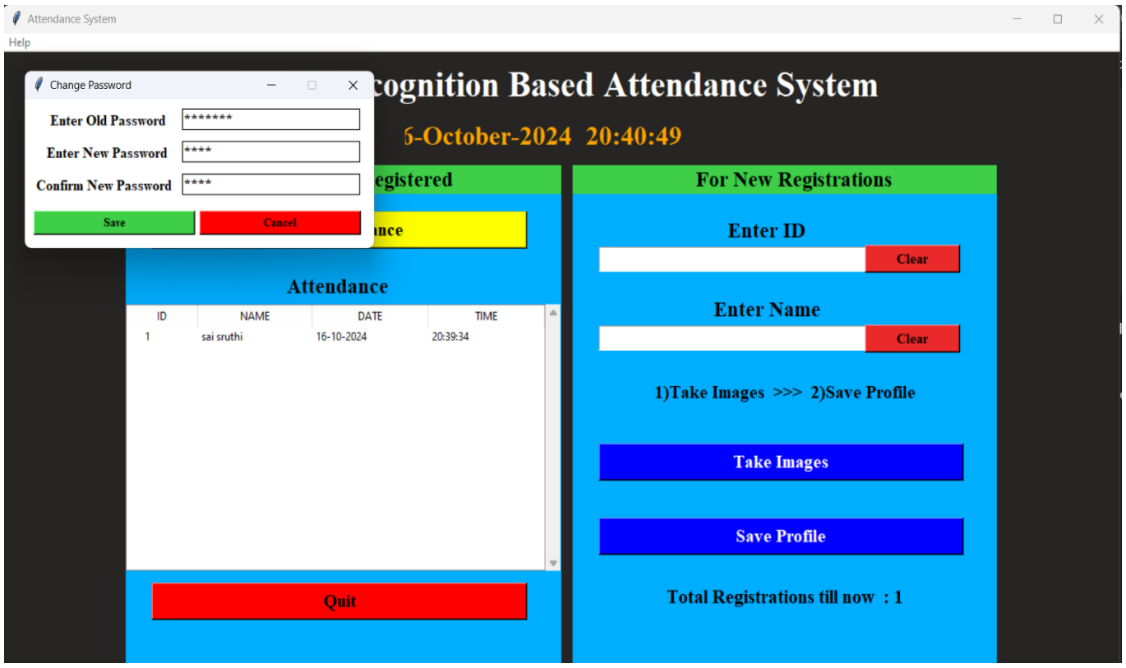


Figure 7.3 Change Password

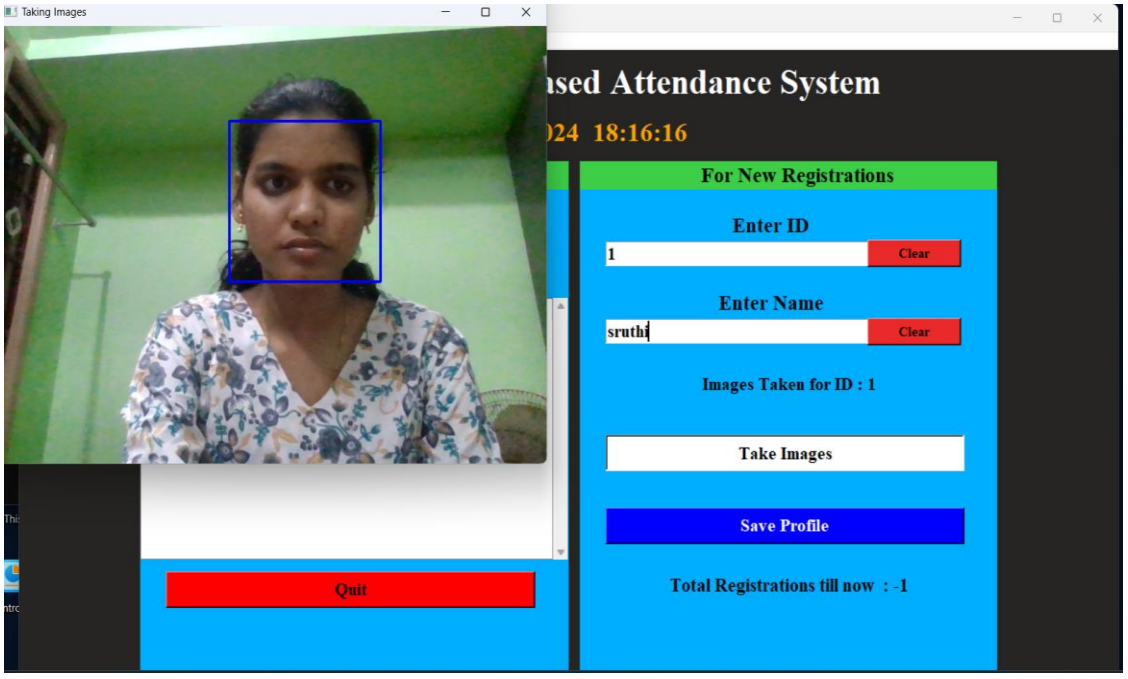


Figure 7.4 Taking images for registration for the particular student

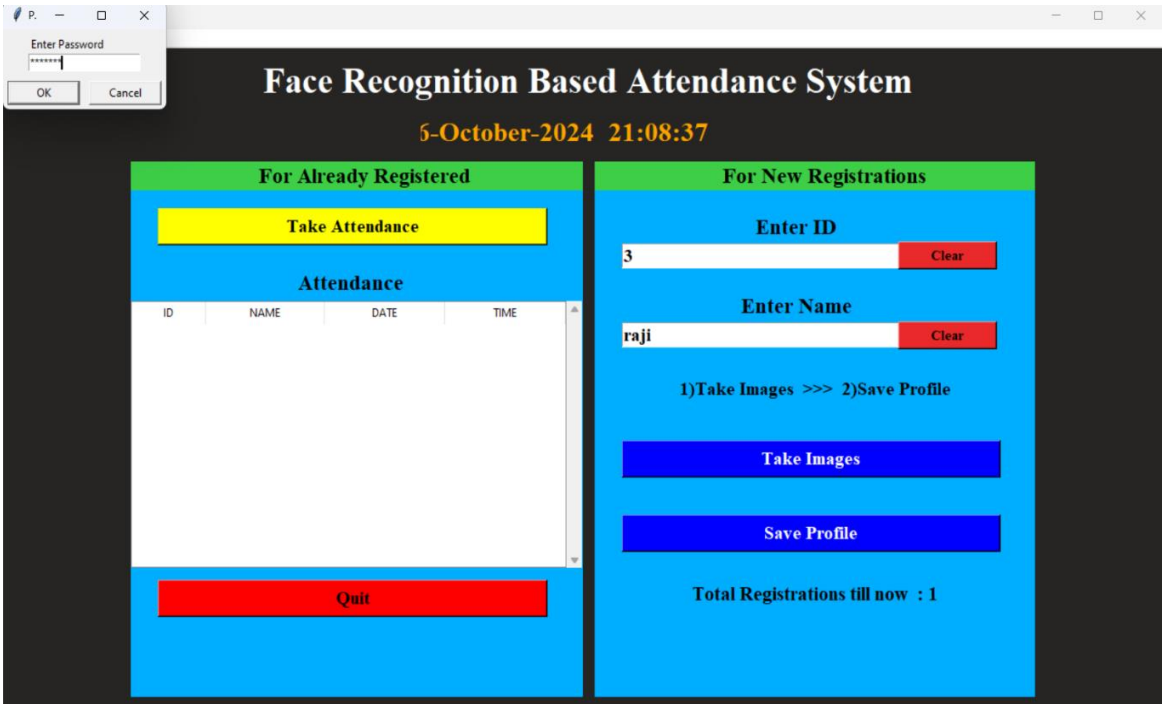


Figure 7.5 Authentication for saving the profile after taking images of student

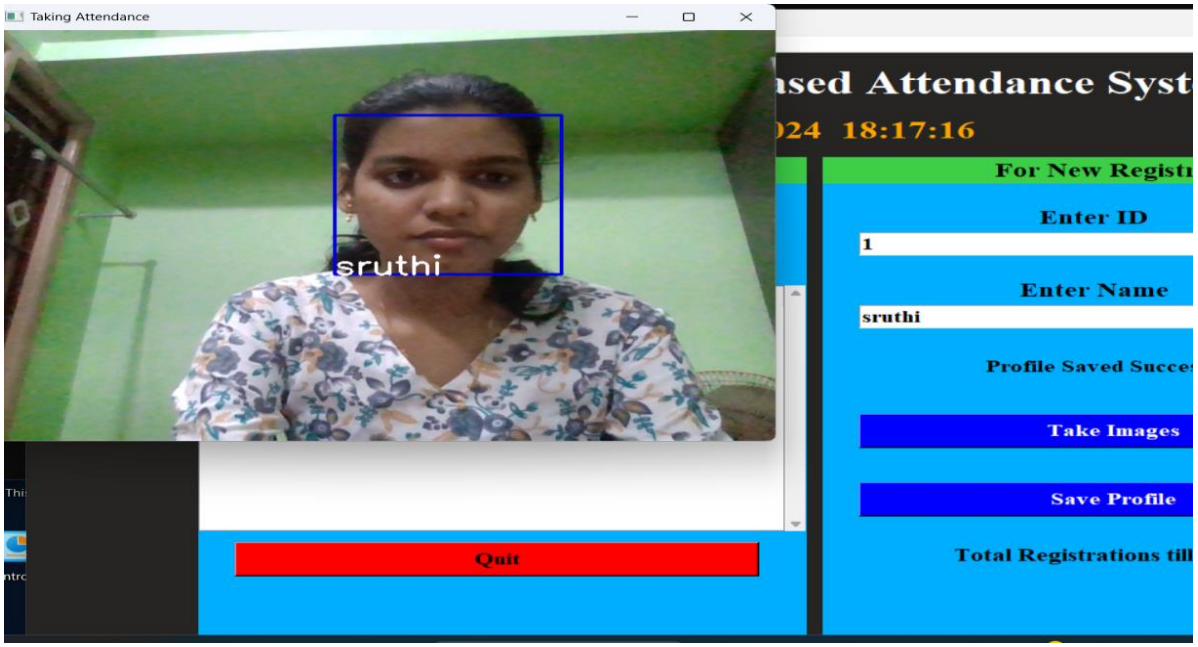


Figure 7.6 Taking attendance by recognizing face

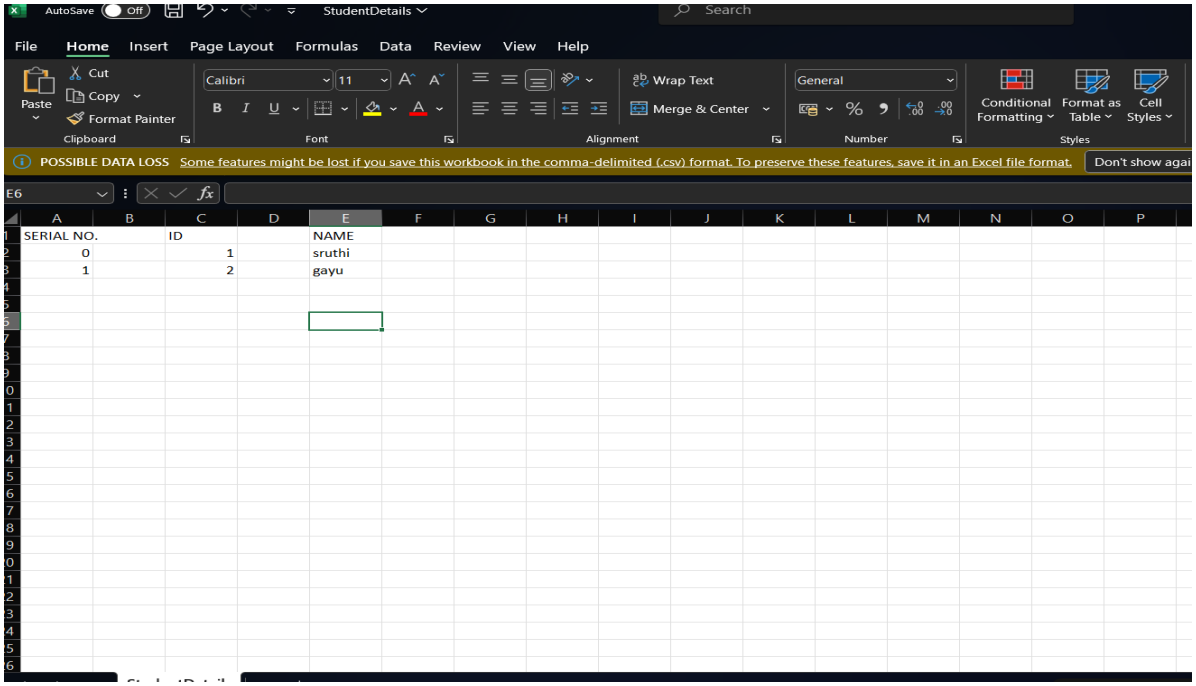


Figure 7.7 Registered student list saved in Student details excel sheet

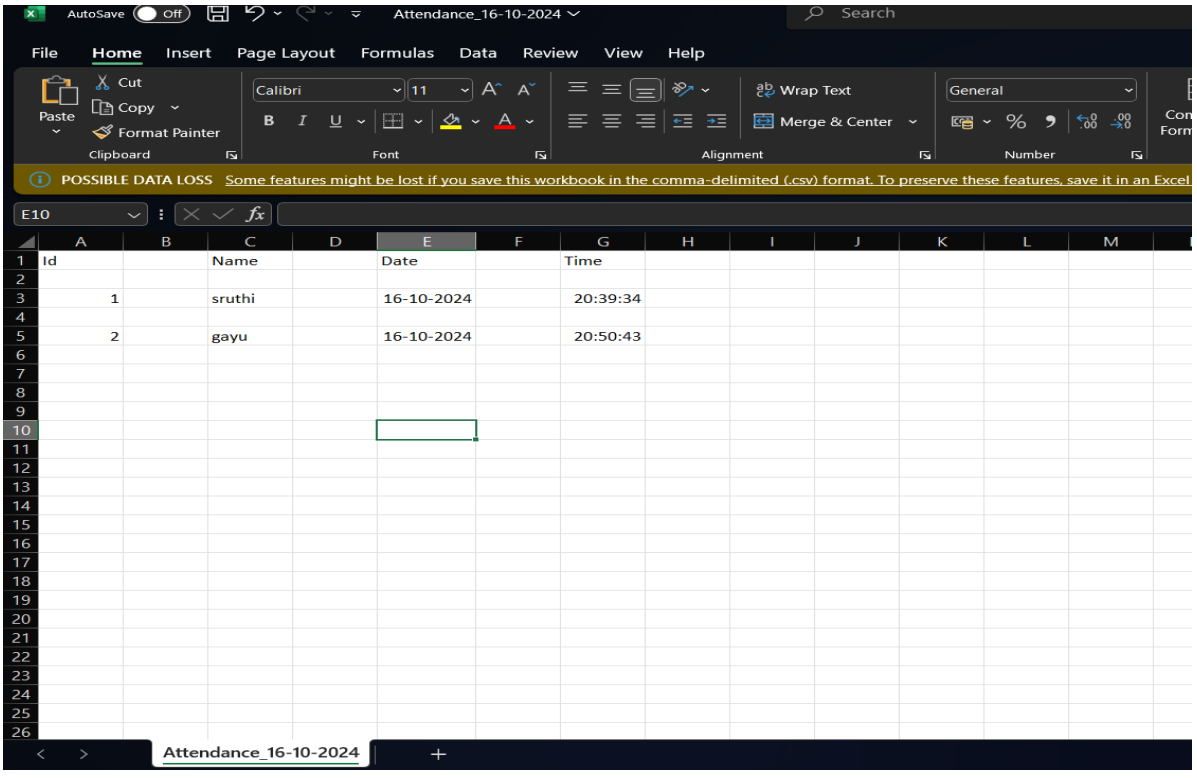


Figure 7.8 Attendance of students saved in Attendance (date) excel sheet

CHAPTER 8

CONCLUSION

Face recognition systems are part of facial image processing applications and their significance as a research area are increasing recently. Implementations of system are crime prevention, video surveillance, person verification, and similar security activities. The face recognition system implementation can be part of Universities. Face Recognition Based Attendance System has been envisioned for the purpose of reducing the errors that occur in the traditional (manual) attendance taking system. The aim is to automate and make a system that is useful to the organization such as an institute. The efficient and accurate method of attendance in the office environment that can replace the old manual methods. This method is secure enough, reliable and available for use. The Proposed algorithm introduced here utilizes the well known framework of linear discriminant analysis and it can be considered as a generalization of a number of techniques which are commonly in use. The use of Linear Discrimination Analyser (LDA) in the system increases its efficiency and reliability toward recognition of faces of the students to provide a secure system.

FUTURE ENHANCEMENT

The results are promising, however the future work will focus on adding some features to the page such as student details other than student id and name and detection of face when a student wearing a face masks and a feature to delete the student from the system & also giving authentication to parents to see the attendance report etc.,

CHAPTER 9

REFERENCES

[1] Cognitive psychology: Applying the science of the mind By Bridget Robinson Reigler & Greg Robinson Reigler.

[2] E.Varadharajan,R.Dharani, S.Jeevitha, B.Kavinmathi, S.Hemalatha – "Automatic attendance management system using face detection" 2016 Online International Conference on Green Engineering and Technologies (IC-GET)

[3] Smit Hapani, Nikhil Parakhiya, Prof. Nandana Prabhu, Mayur Paghda – "Automated Attendance System using Image Processing" l 2018 Fourth International Conference on Computing Communication Control and Automation.

[4] Sathyanarayana N, Ramya M R, Ruchitha C, and Shwetha H S – "Automatic Student Attendance Management System Using Facial Recognition" International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE) ISSN: 0976-1353 Volume 25 Issue 6 – MAY 2018.

[5] J. G. RoshanTharanga , S. M. S. C. Samarakoon , T. A. P. Karunarathne, K. L. P. M. Liyanage , M. P. A. W. Gamage, D. Perera – "Smart attendance using real time face recognition" May 2017

[6] Prof. Arun Katara1, Mr. Sudesh V. Kolhe2, Mr. Amar P. Zilpe3, Mr. Nikhil D. Bhele4, Mr. Chetan J. Bele – "Attendance System Using Face Recognition and Class Monitoring System" International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169.

[7] Nilesh D. Veer, B. F. Momin – “An automated attendance system using video surveillance camera”IEEE International Conference On Recent Trends in Electronics Information Communication Technology, May 20-21, 2016.

[8] Refik Samet, Muhammed Tanriverdi – “Face Recognition-Based Mobile Automatic Classroom Attendance Management System” 2017 International Conference on Cyberworlds.

[9] Firoz Mahmud, Mst. Taskia Khatun, Syed Tauhid Zuhori, Shyla Afroge, Mumu Aktar, Biprodip Pal – “Face Recognition using Principle Component Analysis and Linear Discriminant Analysis” 2nd Int'l Conf. on Electrical Engineering and Information & Communication Technology (ICEEICT) 2015.

[10] K. Susheel Kumar, Shitala Prasad, Vijay Bhaskar Semwal, R C Tripathi – “Real time face recognition using adaboost improved fast pca algorithm” International Journal of Artificial Intelligence & Applications (IJAIA), Vol.2, No.3, July 2011.

[11] Krishna Dharavath, G. Amarnath, Fazal A. Talukdar, Rabul H. Laskar – “Impact of Image Preprocessing on Face Recognition: A Comparative Analysis” International Conference on Communication and Signal Processing (IEEE 2014).

[12] Ms. Pooja Humbe, Ms. Shivani Kudale, Ms. Apurva Kamshetty, Ms. Akshata Jagtap, Prof. Krushna Belerao – “Automatic Attendance Using Face Recognition” International Journal on Recent and Innovation Trends in Computing and Communication Volume: 5 Issue: 12.

[13] Akshara Jadhav, Akshay Jadhav Tushar Ladhe, Krishna Yeolekar – “Automated attendance system using face recognition” International Research Journal of Engineering andTechnology (IRJET).

[14] Shireesha Chintalapati, M.V. Raghunadh – “Automated Attendance Management System Based On Face Recognition Algorithms” 2013 IEEE International Conference on Computational Intelligence and Computing Research.

[15] A. Majumdar and R. K. Ward – “Pseudo-Fisherface method for single image per person face recognition” ICASSP 2008.

[16] Kennedy Okokpujie, Etinosa Noma-Osaghae, Samuel John, Kalu-Anyah Grace, Imhade Okokpujie – “A Face Recognition Attendance System with GSM Notification” 2017 IEEE 3rd International Conference on Electro-Technology for National Development.

CHAPTER 10

APPENDIX

```
############################################ IMPORTING
################################################
import tkinter as tk
from tkinter import ttk
from tkinter import messagebox as mess
import tkinter.simpledialog as tsd
import cv2,os
import csv
import numpy as np
from PIL import Image
import pandas as pd
import datetime
import time

############################################ FUNCTIONS
################################################

def  assure_path_exists(path):
    dir = os.path.dirname(path)
    if not os.path.exists(dir):
        os.makedirs(dir)

#############################################################
######################

def tick():
```

```
    time_string = time.strftime('%H:%M:%S')
    clock.config(text=time_string)
    clock.after(200,tick)

##################################################################
########################

def contact():
    mess._show(title='Contact us', message="Please contact us on :
'xxxxxxxxxxxxx@gmail.com' ")

##################################################################
########################

def check_haarcascadefile():
    exists = os.path.isfile("haarcascade_frontalface_default.xml")
    if exists:
        pass
    else:
        mess._show(title='Some file missing', message='Please contact us
for help')
        window.destroy()

##################################################################
########################

def save_pass():
    assure_path_exists("TrainingImageLabel/")
    exists1 = os.path.isfile("TrainingImageLabel\psd.txt")
```

```
    if exists1:
        tf = open("TrainingImageLabel\psd.txt", "r")
        key = tf.read()
    else:
        master.destroy()
        new_pas = tsd.askstring('Old Password not found', 'Please enter a
new password below', show='*')
        if new_pas == None:
            mess._show(title='No Password Entered', message='Password not
set!! Please try again')
        else:
            tf = open("TrainingImageLabel\psd.txt", "w")
            tf.write(new_pas)
            mess._show(title='Password Registered', message='New password
was registered successfully!!')
            return
    op = (old.get())
    newp= (new.get())
    nnewp = (nnew.get())
    if (op == key):
        if(newp == nnewp):
            txf = open("TrainingImageLabel\psd.txt", "w")
            txf.write(newp)
        else:
            mess._show(title='Error', message='Confirm new password
again!!!')
            return
    else:
```

```
        mess._show(title='Wrong Password', message='Please enter correct
old password.')
        return
    mess._show(title='Password Changed', message='Password changed
successfully!!')
    master.destroy()

#############################################################
########################

def change_pass():
    global master
    master = tk.Tk()
    master.geometry("400x160")
    master.resizable(False,False)
    master.title("Change Password")
    master.configure(background="white")
    lbl4 = tk.Label(master,text='    Enter Old
Password',bg='white',font=('times', 12, ' bold '))
    lbl4.place(x=10,y=10)
    global old
    old=tk.Entry(master,width=25 ,fg="black",relief='solid',font=('times',
12, ' bold '),show='*')
    old.place(x=180,y=10)
    lbl5 = tk.Label(master, text='   Enter New Password', bg='white',
font=('times', 12, ' bold '))
    lbl5.place(x=10, y=45)
    global new
```

```
    new = tk.Entry(master, width=25, fg="black",relief='solid',
font=('times', 12, ' bold '),show='*')
    new.place(x=180, y=45)
    lbl6 = tk.Label(master, text='Confirm New Password', bg='white',
font=('times', 12, ' bold '))
    lbl6.place(x=10, y=80)
    global nnew
    nnew = tk.Entry(master, width=25, fg="black",
relief='solid',font=('times', 12, ' bold '),show='*')
    nnew.place(x=180, y=80)
    cancel=tk.Button(master,text="Cancel", command=master.destroy
,fg="black" ,bg="red" ,height=1,width=25 , activebackground = "white"
,font=('times', 10, ' bold '))
    cancel.place(x=200, y=120)
    save1 = tk.Button(master, text="Save", command=save_pass,
fg="black", bg="#3ece48", height = 1,width=25,
activebackground="white", font=('times', 10, ' bold '))
    save1.place(x=10, y=120)
    master.mainloop()

####################################################################
#########################

def psw():
    assure_path_exists("TrainingImageLabel/")
    exists1 = os.path.isfile("TrainingImageLabel\psd.txt")
    if exists1:
        tf = open("TrainingImageLabel\psd.txt", "r")
        key = tf.read()
```

```
    else:
        new_pas = tsd.askstring('Old Password not found', 'Please enter a
new password below', show='*')
        if new_pas == None:
            mess._show(title='No Password Entered', message='Password not
set!! Please try again')
        else:
            tf = open("TrainingImageLabel\psd.txt", "w")
            tf.write(new_pas)
            mess._show(title='Password Registered', message='New password
was registered successfully!!')
            return
    password = tsd.askstring('Password', 'Enter Password', show='*')
    if (password == key):
        TrainImages()
    elif (password == None):
        pass
    else:
        mess._show(title='Wrong Password', message='You have entered
wrong password')

######################################################################
############################

def clear():
    txt.delete(0, 'end')
    res = "1)Take Images >>> 2)Save Profile"
    message1.configure(text=res)
```

```
def clear2():
    txt2.delete(0, 'end')
    res = "1)Take Images >>> 2)Save Profile"
    message1.configure(text=res)

##################################################################
############################

def TakeImages():
    check_haarcascadefile()
    columns = ['SERIAL NO.', '', 'ID', '', 'NAME']
    assure_path_exists("StudentDetails/")
    assure_path_exists("TrainingImage/")
    serial = 0
    exists = os.path.isfile("StudentDetails\StudentDetails.csv")
    if exists:
        with open("StudentDetails\StudentDetails.csv", 'r') as csvFile1:
            reader1 = csv.reader(csvFile1)
            for   l   in   reader1:
                serial = serial + 1
        serial = (serial // 2)
        csvFile1.close()
    else:
        with open("StudentDetails\StudentDetails.csv", 'a+') as csvFile1:
            writer = csv.writer(csvFile1)
            writer.writerow(columns)
            serial = 1
        csvFile1.close()
```

```
    Id = (txt.get())
    name = (txt2.get())
    if ((name.isalpha()) or (' ' in name)):
        cam = cv2.VideoCapture(0)
        harcascadePath = "haarcascade_frontalface_default.xml"
        detector = cv2.CascadeClassifier(harcascadePath)
        sampleNum = 0
        while (True):
            ret, img = cam.read()
            gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
            faces = detector.detectMultiScale(gray, 1.3, 5)
            for (x, y, w, h) in faces:
                cv2.rectangle(img, (x, y), (x + w, y + h), (255, 0, 0), 2)
                # incrementing sample number
                sampleNum = sampleNum + 1

                # saving the captured face in the dataset folder TrainingImage
                cv2.imwrite("TrainingImage\ " + name + "." + str(serial) + "." +
Id + '.' + str(sampleNum) + ".jpg",

                            gray[y:y + h, x:x + w])
                # display the frame
                cv2.imshow('Taking Images', img)
            # wait for 100 miliseconds
            if cv2.waitKey(100) & 0xFF == ord('q'):
                break
            # break if the sample number is morethan 100
            elif sampleNum > 100:
                break
        cam.release()
        cv2.destroyAllWindows()
```

```
        res = "Images Taken for ID : " + Id
        row = [serial, '', Id, '', name]
        with open('StudentDetails\StudentDetails.csv', 'a+') as csvFile:
            writer = csv.writer(csvFile)
            writer.writerow(row)
        csvFile.close()
        message1.configure(text=res)
    else:
        if (name.isalpha() == False):
            res = "Enter Correct name"
            message.configure(text=res)

##################################################################
##############################

def TrainImages():
    check_haarcascadefile()
    assure_path_exists("TrainingImageLabel/")
    recognizer = cv2.face_LBPHFaceRecognizer.create()
    harcascadePath = "haarcascade_frontalface_default.xml"
    detector = cv2.CascadeClassifier(harcascadePath)
    faces, ID = getImagesAndLabels("TrainingImage")
    try:
        recognizer.train(faces, np.array(ID))
    except:
        mess._show(title='No Registrations', message='Please Register
someone first!!!')
        return
    recognizer.save("TrainingImageLabel\Trainner.yml")
```

```
    res = "Profile Saved Successfully"
    message1.configure(text=res)
    message.configure(text='Total Registrations till now : ' + str(ID[0]))

##################################################################
#################################3

def getImagesAndLabels(path):
    # get the path of all the files in the folder
    imagePaths = [os.path.join(path, f) for f in os.listdir(path)]
    # create empth face list
    faces = []
    # create empty ID list
    Ids = []
    # now looping through all the image paths and loading the Ids and the
images
    for imagePath in imagePaths:
        # loading the image and converting it to gray scale
        pilImage = Image.open(imagePath).convert('L')
        # Now we are converting the PIL image into numpy array
        imageNp = np.array(pilImage, 'uint8')
        # getting the Id from the image
        ID = int(os.path.split(imagePath)[-1].split(".")[1])
        # extract the face from the training image sample
        faces.append(imageNp)
        Ids.append(ID)
    return faces, Ids
```

```
##################################################################
################################

def TrackImages():
    check_haarcascadefile()
    assure_path_exists("Attendance/")
    assure_path_exists("StudentDetails/")
    for k in tv.get_children():
        tv.delete(k)
    msg = ''
    i = 0
    j = 0
    recognizer = cv2.face.LBPHFaceRecognizer_create() #
cv2.createLBPHFaceRecognizer()
    exists3 = os.path.isfile("TrainingImageLabel\Trainner.yml")
    if exists3:
        recognizer.read("TrainingImageLabel\Trainner.yml")
    else:
        mess._show(title='Data Missing', message='Please click on Save
Profile to reset data!!')
        return
    harcascadePath = "haarcascade_frontalface_default.xml"
    faceCascade = cv2.CascadeClassifier(harcascadePath);

    cam = cv2.VideoCapture(0)
    font = cv2.FONT_HERSHEY_SIMPLEX
    col_names = ['Id', '', 'Name', '', 'Date', '', 'Time']
    exists1 = os.path.isfile("StudentDetails\StudentDetails.csv")
    if exists1:
```

```
        df = pd.read_csv("StudentDetails\StudentDetails.csv")
    else:
        mess._show(title='Details Missing', message='Students details are
missing, please check!')
        cam.release()
        cv2.destroyAllWindows()
        window.destroy()
    while True:
        ret, im = cam.read()
        gray = cv2.cvtColor(im, cv2.COLOR_BGR2GRAY)
        faces = faceCascade.detectMultiScale(gray, 1.2, 5)
        for (x, y, w, h) in faces:
            cv2.rectangle(im, (x, y), (x + w, y + h), (225, 0, 0), 2)
            serial, conf = recognizer.predict(gray[y:y + h, x:x + w])
            if (conf < 50):
                ts = time.time()
                date = datetime.datetime.fromtimestamp(ts).strftime('%d-%m-
%Y')
                timeStamp =
datetime.datetime.fromtimestamp(ts).strftime('%H:%M:%S')
                aa = df.loc[df['SERIAL NO.'] == serial]['NAME'].values
                ID = df.loc[df['SERIAL NO.'] == serial]['ID'].values
                ID = str(ID)
                ID = ID[1:-1]
                bb = str(aa)
                bb = bb[2:-2]
                attendance = [str(ID), '', bb, '', str(date), '', str(timeStamp)]

            else:
```

```
                Id = 'Unknown'
                bb = str(Id)
            cv2.putText(im, str(bb), (x, y + h), font, 1, (255, 255, 255), 2)
        cv2.imshow('Taking Attendance', im)
        if (cv2.waitKey(1) == ord('q')):
            break
    ts = time.time()
    date = datetime.datetime.fromtimestamp(ts).strftime('%d-%m-%Y')
    exists = os.path.isfile("Attendance\Attendance_" + date + ".csv")
    if exists:
        with open("Attendance\Attendance_" + date + ".csv", 'a+') as
csvFile1:
            writer = csv.writer(csvFile1)
            writer.writerow(attendance)
        csvFile1.close()
    else:
        with open("Attendance\Attendance_" + date + ".csv", 'a+') as
csvFile1:
            writer = csv.writer(csvFile1)
            writer.writerow(col_names)
            writer.writerow(attendance)
        csvFile1.close()
    with open("Attendance\Attendance_" + date + ".csv", 'r') as csvFile1:
        reader1 = csv.reader(csvFile1)
        for lines in reader1:
            i = i + 1
            if (i > 1):
                if (i % 2 != 0):
                    iidd = str(lines[0]) + '   '
```

```
                tv.insert('', 0, text=iidd, values=(str(lines[2]), str(lines[4]),
str(lines[6])))
    csvFile1.close()
    cam.release()
    cv2.destroyAllWindows()

########################################## USED STUFFS
##########################################

global key
key = ''

ts = time.time()
date = datetime.datetime.fromtimestamp(ts).strftime('%d-%m-%Y')
day,month,year=date.split('-')

mont={'01':'January',
      '02':'February',
      '03':'March',
      '04':'April',
      '05':'May',
      '06':'June',
      '07':'July',
      '08':'August',
      '09':'September',
      '10':'October',
      '11':'November',
      '12':'December'
      }
```

```
########################################## GUI FRONT-END
##########################################

window = tk.Tk()
window.geometry("1280x720")
window.resizable(True,False)
window.title("Attendance System")
window.configure(background='#262523')

frame1 = tk.Frame(window, bg="#00aeff")
frame1.place(relx=0.11, rely=0.17, relwidth=0.39, relheight=0.80)

frame2 = tk.Frame(window, bg="#00aeff")
frame2.place(relx=0.51, rely=0.17, relwidth=0.38, relheight=0.80)

message3 = tk.Label(window, text="Face Recognition Based Attendance
System" ,fg="white",bg="#262523" ,width=55 ,height=1,font=('times',
29, ' bold '))
message3.place(x=10, y=10)

frame3 = tk.Frame(window, bg="#c4c6ce")
frame3.place(relx=0.52, rely=0.09, relwidth=0.09, relheight=0.07)

frame4 = tk.Frame(window, bg="#c4c6ce")
frame4.place(relx=0.36, rely=0.09, relwidth=0.16, relheight=0.07)
```

```
datef = tk.Label(frame4, text = day+"-"+mont[month]+"-"+year+" | ",
fg="orange",bg="#262523" ,width=55 ,height=1,font=('times', 22, ' bold
'))
datef.pack(fill='both',expand=1)

clock = tk.Label(frame3,fg="orange",bg="#262523" ,width=55
,height=1,font=('times', 22, ' bold '))
clock.pack(fill='both',expand=1)
tick()

head2 = tk.Label(frame2, text="                For New Registrations
", fg="black",bg="#3ece48" ,font=('times', 17, ' bold ') )
head2.grid(row=0,column=0)

head1 = tk.Label(frame1, text="                For Already Registered
", fg="black",bg="#3ece48" ,font=('times', 17, ' bold ') )
head1.place(x=0,y=0)

lbl = tk.Label(frame2, text="Enter ID",width=20 ,height=1 ,fg="black"
,bg="#00aeff" ,font=('times', 17, ' bold ') )
lbl.place(x=80, y=55)

txt = tk.Entry(frame2,width=32 ,fg="black",font=('times', 15, ' bold '))
txt.place(x=30, y=88)

lbl2 = tk.Label(frame2, text="Enter Name",width=20 ,fg="black"
,bg="#00aeff" ,font=('times', 17, ' bold '))
lbl2.place(x=80, y=140)
```

```
txt2 = tk.Entry(frame2,width=32 ,fg="black",font=('times', 15, ' bold ') )
txt2.place(x=30, y=173)

message1 = tk.Label(frame2, text="1)Take Images >>> 2)Save Profile"
,bg="#00aeff" ,fg="black" ,width=39 ,height=1, activebackground =
"yellow" ,font=('times', 15, ' bold '))
message1.place(x=7, y=230)

message = tk.Label(frame2, text="" ,bg="#00aeff" ,fg="black"
,width=39,height=1, activebackground = "yellow" ,font=('times', 16, '
bold '))
message.place(x=7, y=450)

lbl3 = tk.Label(frame1, text="Attendance",width=20 ,fg="black"
,bg="#00aeff" ,height=1 ,font=('times', 17, ' bold '))
lbl3.place(x=100, y=115)

res=0
exists = os.path.isfile("StudentDetails\StudentDetails.csv")
if exists:
    with open("StudentDetails\StudentDetails.csv", 'r') as csvFile1:
        reader1 = csv.reader(csvFile1)
        for l in reader1:
            res = res + 1
    res = (res // 2) - 1
    csvFile1.close()
else:
    res = 0
message.configure(text='Total Registrations till now : '+str(res))
```

```
###################### MENUBAR
#################################

menubar = tk.Menu(window,relief='ridge')
filemenu = tk.Menu(menubar,tearoff=0)
filemenu.add_command(label='Change Password', command =
change_pass)
filemenu.add_command(label='Contact Us', command = contact)
filemenu.add_command(label='Exit',command = window.destroy)
menubar.add_cascade(label='Help',font=('times', 29, ' bold
'),menu=filemenu)

################## TREEVIEW ATTENDANCE TABLE
####################

tv= ttk.Treeview(frame1,height =13,columns = ('name','date','time'))
tv.column('#0',width=82)
tv.column('name',width=130)
tv.column('date',width=133)
tv.column('time',width=133)
tv.grid(row=2,column=0,padx=(0,0),pady=(150,0),columnspan=4)
tv.heading('#0',text ='ID')
tv.heading('name',text ='NAME')
tv.heading('date',text ='DATE')
tv.heading('time',text ='TIME')

###################### SCROLLBAR
################################
```

```
scroll=ttk.Scrollbar(frame1,orient='vertical',command=tv.yview)
scroll.grid(row=2,column=4,padx=(0,100),pady=(150,0),sticky='ns')
tv.configure(yscrollcommand=scroll.set)

###################### BUTTONS
##################################

clearButton = tk.Button(frame2, text="Clear", command=clear
,fg="black" ,bg="#ea2a2a" ,width=11 ,activebackground = "white"
,font=('times', 11, ' bold '))
clearButton.place(x=335, y=86)
clearButton2 = tk.Button(frame2, text="Clear", command=clear2
,fg="black" ,bg="#ea2a2a" ,width=11 , activebackground = "white"
,font=('times', 11, ' bold '))
clearButton2.place(x=335, y=172)
takeImg = tk.Button(frame2, text="Take Images", command=TakeImages
,fg="white" ,bg="blue" ,width=34 ,height=1, activebackground =
"white" ,font=('times', 15, ' bold '))
takeImg.place(x=30, y=300)
trainImg = tk.Button(frame2, text="Save Profile", command=psw
,fg="white" ,bg="blue" ,width=34 ,height=1, activebackground =
"white" ,font=('times', 15, ' bold '))
trainImg.place(x=30, y=380)
trackImg = tk.Button(frame1, text="Take Attendance",
command=TrackImages ,fg="black" ,bg="yellow" ,width=35
,height=1, activebackground = "white" ,font=('times', 15, ' bold '))
trackImg.place(x=30,y=50)
```

```
quitWindow = tk.Button(frame1, text="Quit", command=window.destroy
,fg="black" ,bg="red" ,width=35 ,height=1, activebackground = "white"
,font=('times', 15, ' bold '))
quitWindow.place(x=30, y=450)

###################### END
#######################################

window.configure(menu=menubar)
window.mainloop()

#############################################################
########################################
```

REGISTRATION FORM
International Conference on
Numerical Optimization in Engineering & Sciences (NOIEAS-2024)
December 5-7, 2024

Name : **R.Sarmilaa**

Designation : **III B. E CSE Student**

Affiliation : **Panimalar Engineering College**

Corresponding Address : **Bangalore Trunk Road,**
Varadharajapuram, Poonamallee, Chennai – 600 123.

Phone : **044 -26490404**

Email : **shyamalapmr@gmail.com**

Mobile No. : **9962028328**

Title of the paper : **LDA based Face Recognition Attendance Management System**

No. of Accompanying persons: **Online Presentation**

Details of the payment :

UTR /Trn. No.: **464864289232** Date : **08.10.2024** Amount : **Rs 3,540**

Bank Name: **Indian Bank** Branch Name : **Poonamalle**

9/10/24.

Forwarding of HoD with Seal

Dr. A.R. Mohamed Shanavas M.E., Ph.D.
Professor & Head
Department of Computer Science and Engineering
Panimalar Engineering College
Chennai - 600123

HoD Signature with Date

(Only for students)

R Sarmilaa

Student Signature

Mode: Online Presentation

Note:

- Xerox/Scanned Copy of Receipts of online transfer to be attached with this Registration Form.
- Co-authors attending the conference have to register separately.
- For Accompanying persons, No kit will be given.
- Conference details are given in the website.

Past Conference (NOIEAS-2019) proceedings is available in the following link:

https://link.springer.com/book/10.1007/978-981-15-3215-3