REAL TIME FACE ATTENDANCE SYSTEM

A MINI PROJECT REPORT

Submitted by

KARTHICK ARAVIND B [REGISTER NO: 211419104318]

T. PENIEL BRANHAM [REGISTER NO: 211419104321]

in 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)

JUNE 2022

PANIMALAR ENGINEERING COLLEGE

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

BONAFIDE CERTIFICATE

Certified that this mini project report **"REAL TIME FACE ATTENDANCE SYSTEM"** is the bonafide work of **"Karthick Aravind B [211419104318] T. Peniel Branham [211419104321]"** who carried out the projectwork under my supervision.

SIGNATURE

Dr. S. MURUGAVALLI, M.E., Ph.D.,
HEAD OF THE DEPARTMENT

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

SIGNATURE

Dr. G. SENTHIL KUMAR, M.E.,Ph.D,
PROFESSOR,

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

Certified that the above mentioned students were examined in End Semester mini project (CS8611) held on _____

INTERNAL EXAMINER

EXTERNAL EXAMINER

DECLARATION BY THE STUDENTS

We hereby "**Karthick Aravind B [211419104318], T. Peniel Branham [211419104321]"** declare that this project report titled "**REAL TIME FACE ATTENDANCE SYSTEM**", under the guidance of **Dr. G. Senthil Kumar** is the original work done by us and we have not plagiarized or submitted to any other degree in any university by us.

KARTHICK ARAVIND B

T. PENIEL BRANHAM

ACKNOWLEDGEMENT

We would like to 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 express our sincere thanks to our Directors **Tmt. C. VIJAYARAJESWARI**, **Dr. C. SAKTHI KUMAR, M.E., Ph.D.** and Dr. **SARANYASREE SAKTHI KUMAR B.E., M.B.A., Ph.D.,** for providing us with the necessary facilities to undertake this project.

We also express our gratitude to our Principal **Dr. K. MANI, M.E., Ph.D.** who facilitated us in completing the project.

We thank the Head of the CSE Department, **Dr. S. MURUGAVALLI, M.E., Ph.D.,** for the support extended throughout the project.

We would like to thank our supervisor **Dr. G. SENTHIL KUMAR,** and coordinator **Dr. S. BALAJI** and all the faculty members of the Department of CSE for their advice and encouragement for the successful completion of the project.

KARTHICK ARAVIND B

T. PENIEL BRANHAM

ABSTRACT

Face is the representation of one's identity. Hence, we have proposed an automated student attendance system based on face recognition. Face recognition system is very useful in life applications especially in security control systems. The airport protection system uses face recognition to identify suspects and FBI (Federal Bureau of Investigation) uses face recognition for criminal investigations. In our proposed approach, firstly, video framing is performed by activating the camera through a user- friendly interface. The face ROI is detected and segmented from the video frame by using Viola-Jones algorithm. In the pre-processing stage, scaling of the size of images is performed if necessary in order to prevent loss of information. The median filtering is applied to remove noise followed by conversion of color images to grayscale images. After that, contrast-limited adaptive histogram equalization (CLAHE) is implemented on images to enhance the contrast of images. In face recognition stage, enhanced local binary pattern (LBP) and principal component analysis (PCA) is applied correspondingly in order to extract the features from facial images. In our proposed approach, the enhanced local binary pattern outperform the original LBP by reducing the illumination effect and increasing the recognition rate. Next, the features extracted from the test images are compared with the features extracted from the training images. The facial images are then classified and recognized based on the best result obtained from the combination of algorithm, enhanced LBP and PCA. Finally, the attendance of the recognized student will be marked and saved in the excel file. The student who is not registered will also be able to register on the spot and notification will be given if students sign in more than once. The average accuracy of recognition is 100 % for good quality images, 94.12 % of low-quality images and 95.76 % for Yale face database when two images per person are trained.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS, ABBREVIATIONS

SERIAL NO.	ABBREVATION	EXPANSION
1	LBP	Local Binary Pattern
2	SMQT	Successive mean quantization transform
3	SNOW	Sparse network of winnows
4	CNN	Convolutional Neural Network
5	PCA	Principal Component Analysis
6	ANN	Artificial Neural Network
7	RFID	Radio Frequency Identification
8	R	Radius
9	M	length and height of images
10	N	total number of images
11	Γi	column vector

CHAPTER 1
INTRODUCTION

INTRODUCTION:

1.1 Aims and Objectives

The main objective of this project is to develop face recognition based automated student attendance system. In order to achieve better performance, the test images and training images of this proposed approach are limited to frontal and upright facial images that consist of a single face only. The test images and training images have to be captured by using the same device to ensure no quality difference. In addition, the students have to register in the database to be recognized. The enrolment can be done on the spot through the user-friendly interface.
Expected achievements in order to fulfill the objectives are:

- To detect the face segment from the video frame.
- To extract the useful features from the face detected.
- To classify the features in order to recognize the face detected.
- To record the attendance of the identified student.

1.2 Problem Statement

Traditional student attendance marking technique is often facing a lot of trouble. The face recognition student attendance system emphasizes its simplicity by eliminating classical student attendance marking technique such as calling student names or checking respective identification cards. There are not only disturbing the teaching process but also causes distraction for students during exam sessions. Apart from calling names, attendance sheet is passed around the classroom during the lecture sessions. The lecture class especially the class with a large number of students might find it difficult to have the attendance sheet being passed around the class. Thus, face recognition student attendance system is proposed in order to replace the manual signing of the presence of students which are burdensome and causes students get distracted in order to sign for their attendance. Furthermore, the face recognition based automated student attendance system able to overcome the problem of fraudulent approach and lecturers does not have to count the number of students several times to ensure the presence of the students.

The paper proposed by Zhao, W et al. (2003) has listed the difficulties of facial identification. One of the difficulties of facial identification is the identification between known and unknown images. In addition, paper proposed by Pooja G.R et al. (2010) found out that the training process for face recognition student attendance system is slow and time-consuming. In addition, the paper proposed by Priyanka Wagh et al. (2015) mentioned that different lighting and head poses are often the problems that could degrade the performance of face recognition based student attendance system.

Hence, there is a need to develop a real time operating student attendance system which means the identification process must be done within defined time constraints to prevent omission. The extracted features from facial images which represent the identity of the students have to be consistent towards a change in background, illumination, pose and expression. High accuracy and fast computation time will be the evaluation points of the performance.

1.3 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 (Margaret Rouse, 2012).

Nowadays, face recognition system is prevalent due to its simplicity and awesome performance. For instance, airport protection systems and FBI use face recognition for criminal investigations by tracking suspects, missing children and drug activities (Robert Silk, 2017). Apart from that, Facebook which is a popular social networking website implement face recognition to allow the users to tag their friends in the photo for entertainment purposes (Sidney Fussell, 2018). Furthermore, Intel Company allows the users to use face recognition to get access to their online account (Reichert, C., 2017). Apple allows the users to unlock their mobile phone, iPhone X by using face recognition (deAgonia, M., 2017).

The work on face recognition began in 1960. Woody Bledsoe, Helen Chan Wolf and Charles Bisson had introduced a system which required the administrator to locate eyes, ears, nose and mouth from images. The distance and ratios between the located features and the common reference points are then calculated and compared. The studies are further enhanced by Goldstein, Harmon, and Lesk in 1970 by using other features such as hair colour and lip thickness to automate the recognition. In 1988, Kirby and Sirovich first suggested principle component analysis (PCA) to solve face recognition problem.

CHAPTER 2
LITERATURE SURVEY

LITERATURE SURVEY:

2.1 Student Attendance System

Arun Katara et al. (2017) mentioned disadvantages of RFID (Radio Frequency Identification) card system, fingerprint system and iris recognition system. RFID card system is implemented due to its simplicity. However, the user tends to help their friends to check in as long as they have their friend's ID card. The fingerprint system is indeed effective but not efficient because it takes time for the verification process so the user has to line up and perform the verification one by one. However for face recognition, the human face is always exposed 2and contain less information compared to iris. Iris recognition system which contains more detail might invade the privacy of the user. Voice recognition is available, but it is less accurate compared to other methods. Hence, face recognition system is suggested to be implemented in the student attendance system.

Table 2.1 Advantages & Disadvantages of Different Biometric System (Arun Katara et al., 2017)

System type	**Advantages**	**Disadvantages**
RFID card system	Simple	Fraudulent usage
Fingerprint system	Accurate	Time-consuming
Voice recognition system	-	Less accurate compared to others
Iris recognition system	Accurate	Privacy Invasion

2.2 Face Detection

Difference between face detection and face recognition are often misunderstood. Face detection is to determine only the face segment or face region from image, whereas face recognition is to identify the owner of the facial image. S.Aanjanadevi et al. (2017) and Wei-Lun Chao (2007) presented a few factors which cause face detection and face recognition to encounter difficulties. These factors consist of background, illumination, pose, expression, occlusion, rotation, scaling and translation.

Table 2.2 Factors Causing Face Detection Difficulties (S.Aanjanadevi et al., 2017)

Background	Variation of background and environment around people in the image which affect the efficiency of face recognition.
Illumination	Illumination is the variation caused by various lighting environments which degrade the facial feature detection.
Pose	Pose variation means different angle of the acquired the facial image which cause distortion to recognition process, especially for Eigen face and Fisher face recognition method.
Expression	Different facial expressions are used to express feelings and emotions. The expression variation causes spatial relation change and the facial-feature shape change.
Occlusion	Occlusion means part of the human face is unobserved. This will diminish the performance of face recognition algorithms due to deficiency information.
Rotation, scaling and translation	Transformation of images which might cause distortion of the original information about the images.

2.2.1 Viola-Jones Algorithm

Viola-Jones algorithm which was introduced by P. Viola, M. J. Jones (2001) is the most popular algorithm to localize the face segment from static images or video frame. Basically the concept of Viola-Jones algorithm consists of four parts. The first part is known as Haar feature, second part is where integral image is created, followed by implementation of Adaboost on the third part and lastly cascading process

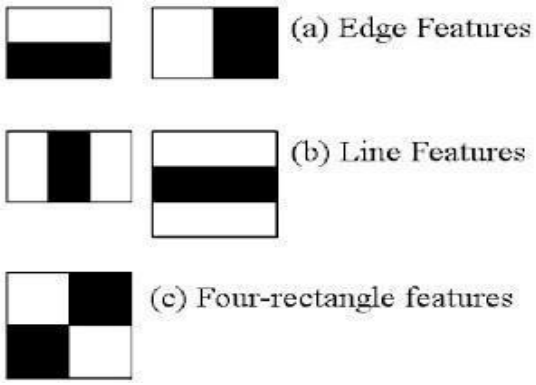

Figure 2.1 Haar Feature (Docs.opencv.org, 2018)

Viola-Jones algorithm analyses a given image using Haar features consisting of multiple rectangles (Mekha Joseph et al., 2016). Figure 2.1 shows several types of Haar features. The features perform as window function mapping onto the image. A single value result, which representing each feature can be computed by subtracting the sum of the white rectangle(s) from the sum of the black rectangle(s) (Mekha Joseph et al., 2016).

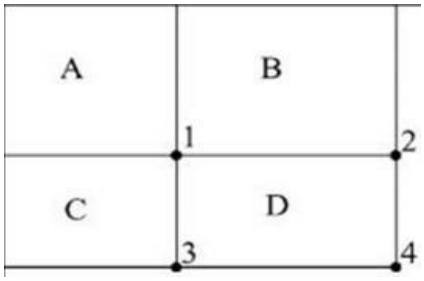

Figure 2.2 Integral of Image (Srushti Girhe et al., 2015)

2.3 Pre-Processing

Subhi Singh et al. (2015) suggested cropping of detected face and colour image was converted to grayscale for pre-processing. They also proposed affine transform to be applied to align the facial image based on coordinates in middle of the eyes and scaling of image to be performed. Arun Katara et al (2017), Akshara Jadhav et.al (2017), Shireesha Chintalapati, and M.V. Raghunadh (2013), all of the 3 papers have proposed histogram equalization to be applied to facial image, and scaling of images was performed for pre-processing.

Pre-processing enhances the performance of the system. It plays an essential role to improve the accuracy of face recognition. Scaling is one of the important pre- processing steps to manipulate the size of the image. Scaling down of an image increases the processing speed by reducing the system computations since the number of pixels are reduced. The size and pixels of the image carry spatial information. Gonzalez, R. C. and Woods (2008) mentioned spatial information is a measure of the smallest discernible detail in an image. Hence, spatial information has to be manipulated carefully to avoid distortion of images to prevent checkerboard effect. The size should be same for all the images for normalization and standardization purposes. Subhi Singh et al (2015) proposed PCA (Principal Component Analysis) to extract features from facial images, same length and width of image is preferred, thus images were scaled to 120 × 120 pixels.

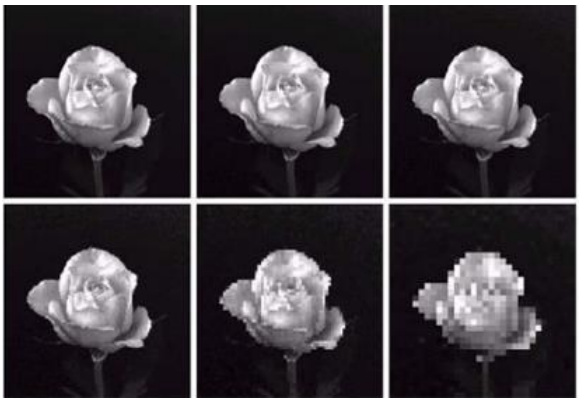

Figure 2.3 Images Show Checkerboard Effect Significantly Increasing from Left to Right (Gonzalez, R. C., & Woods, 2008)

CHAPTER 3
SYSTEM ANALYSIS

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

- The Existing RFID card system is implemented due to its simplicity. However, the user tends to help their friends to check in as long as they have their friend's ID card.
- The fingerprint system is indeed effective but not efficient because it takes time for the verification process so the user has to line up and perform the verification one by one.
- Iris recognition system which contains more detail might invade the privacy of the user and Difficult to Implement.
- Voice recognition is available, but it is less accurate compared to other methods.

ADVANTAGES:

- RFID card system - Simple
- Fingerprint system - Accurate
- Iris recognition System - Accurate

DISADVANTAGES:

- RFID card system - Fraudulent usage
- Fingerprint system - Time-consuming
- Voice recognition System - Less accurate compared to others
- Iris recognition System - Privacy Invasion

3.2 PROPOSED SYSTEM

The main task of our proposed system is to detect and recognize the image of the student and mark the attendance accordingly in the excel file. Also can capture the new entries if needed. After successful recognition of a student, the system automatically updates the attendance in the database. The proposed system improves the performance of existing attendance management systems by eliminating manual calling. The proposed system is divided into major 3 modules which are as follows:

A. Admin Module

In this module, one has to provide the login credentials which involves id and password which will be matched with the one that is stored in database.

B. Student Detail Module

Student details like enrollment, name etc. can be edited, added, update, delete and can search student based on details.

C. Attendance Module

This will mark the attendance if the face of student match with the database else not.

ADVANTAGES:

- High detection speed
- High accuracy
- High speed in training and recognition.
- Capable to deal with lighting problem in object detection.

3.3 DEVELOPMENT ENVIRONMENT

SOFTWARE REQUIREMENT:

- Operating System Windows 10 SP1+,8, 64-bit version.
- Microsoft Visual Studio code
- Anaconda prompt (miniconda3)
- Programming Language: Python
- Database: MySQL
- OpenCV
- Pillow
- Pandas
- NumPy

HARDWARE REQUIREMENT:

- Memory (RAM): 8 GB
- Hard Disk: 32 GB
- Processor: Minimum 2.3 GHz

CHAPTER 4
SYSTEM DESIGN

SYSTEM DESIGN:

4.1 UML DIAGRAMS

A UML diagram is a diagram based on the UML (Unified Modelling Language) with the purpose of visually representing a system along with its main actors, roles, actions, artefacts or classes, in order to better understand, alter, maintain, or document information about the system

4.1.1 Use case diagram

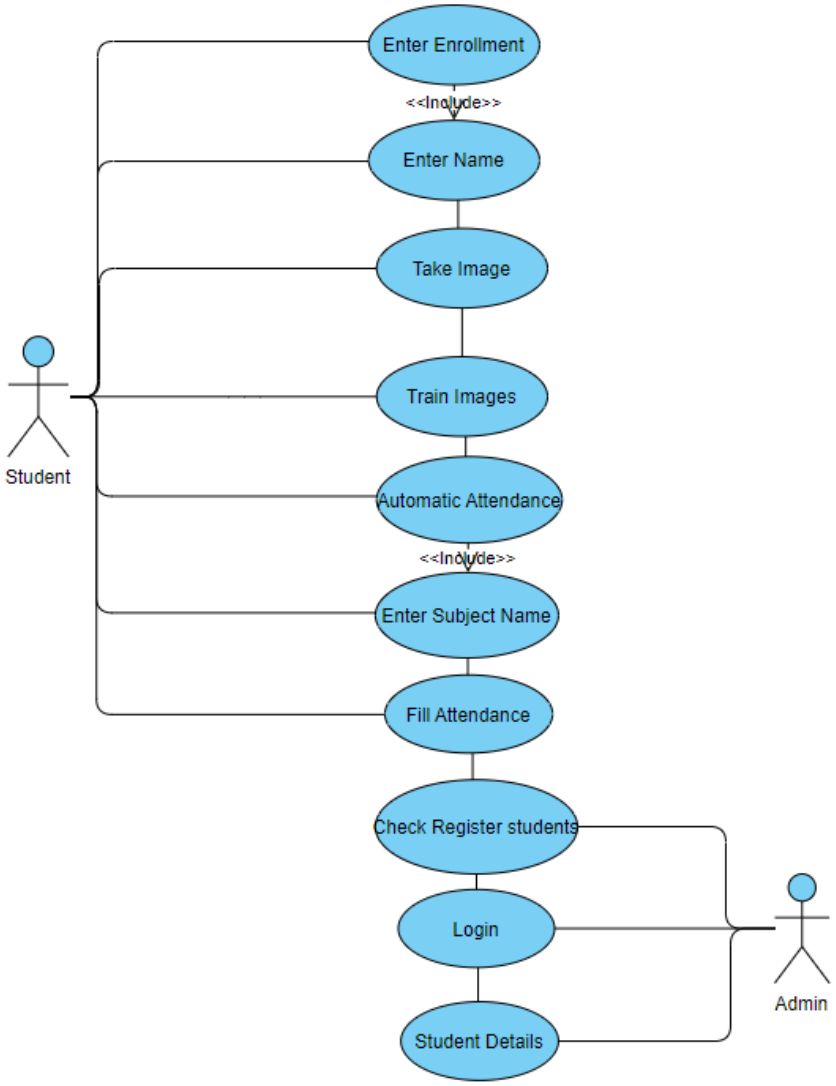

Fig 4.1.1 Use case diagram for Real Time Face Attendance System

4.1.2 Class diagram

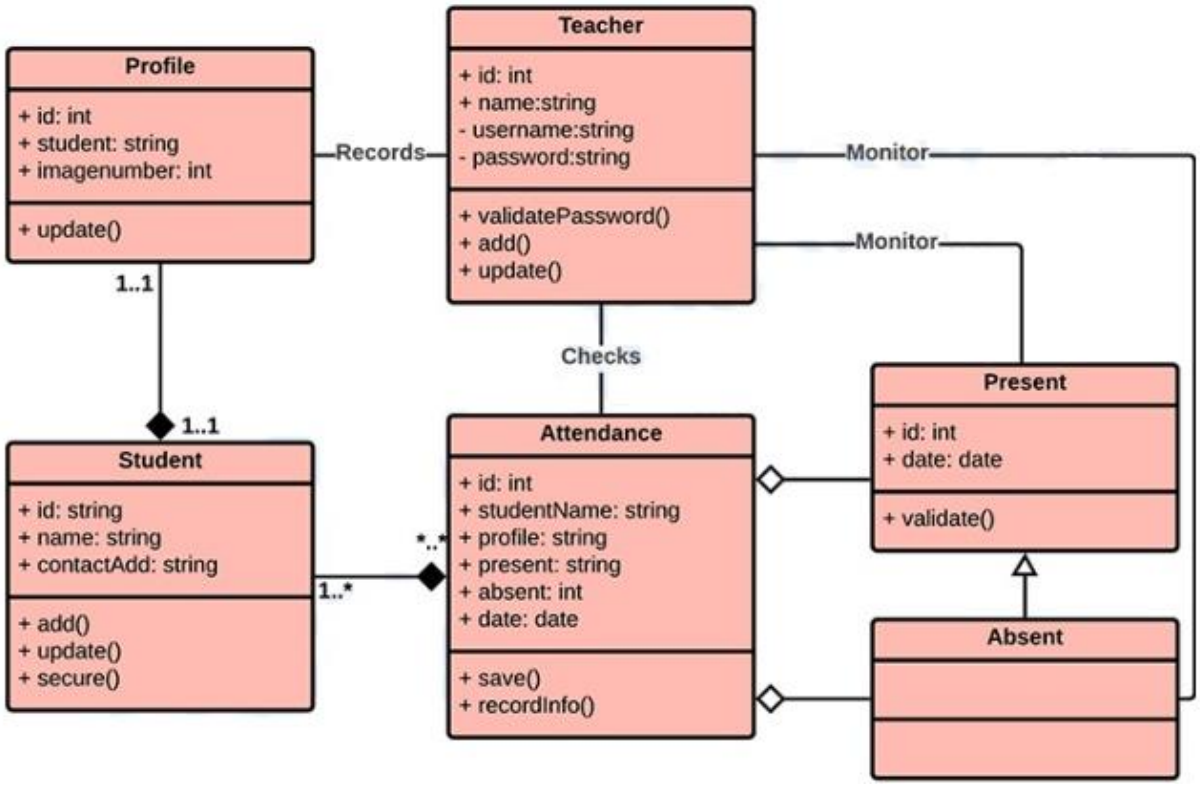

Fig 4.1.2 Class diagram for Real Time Face Attendance System

4.1.3 Sequence diagram

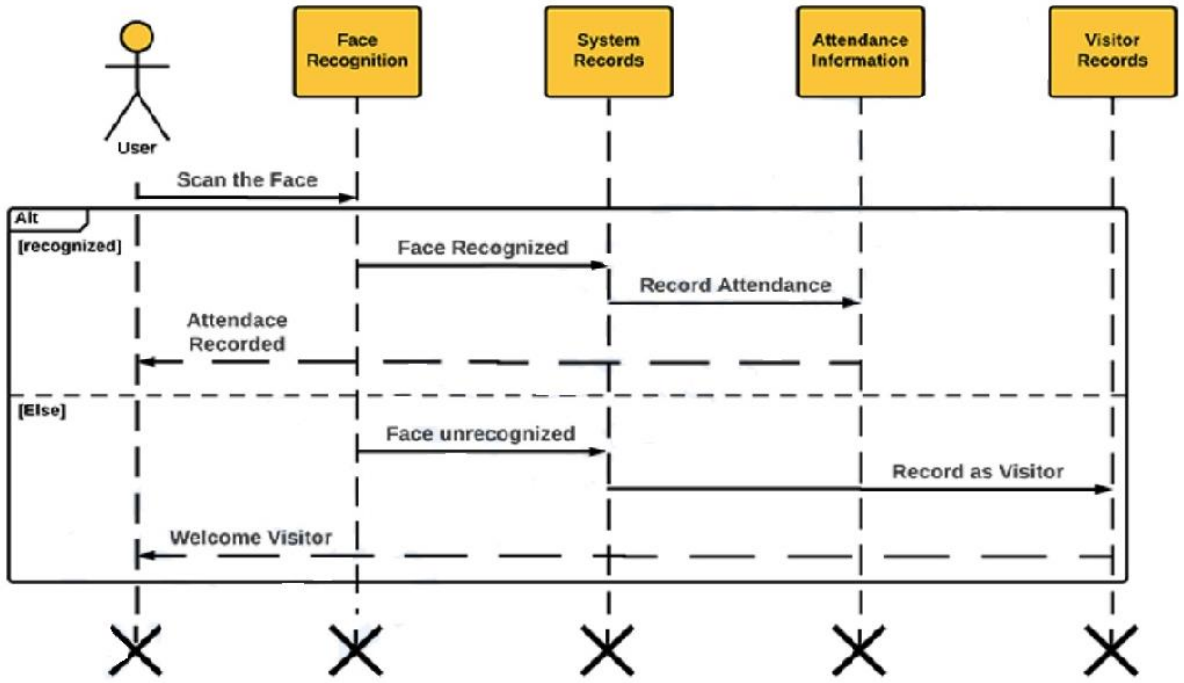

Fig 4.1.3 Sequence diagram for Real Time Face Attendance System

4.1.4 Activity diagram

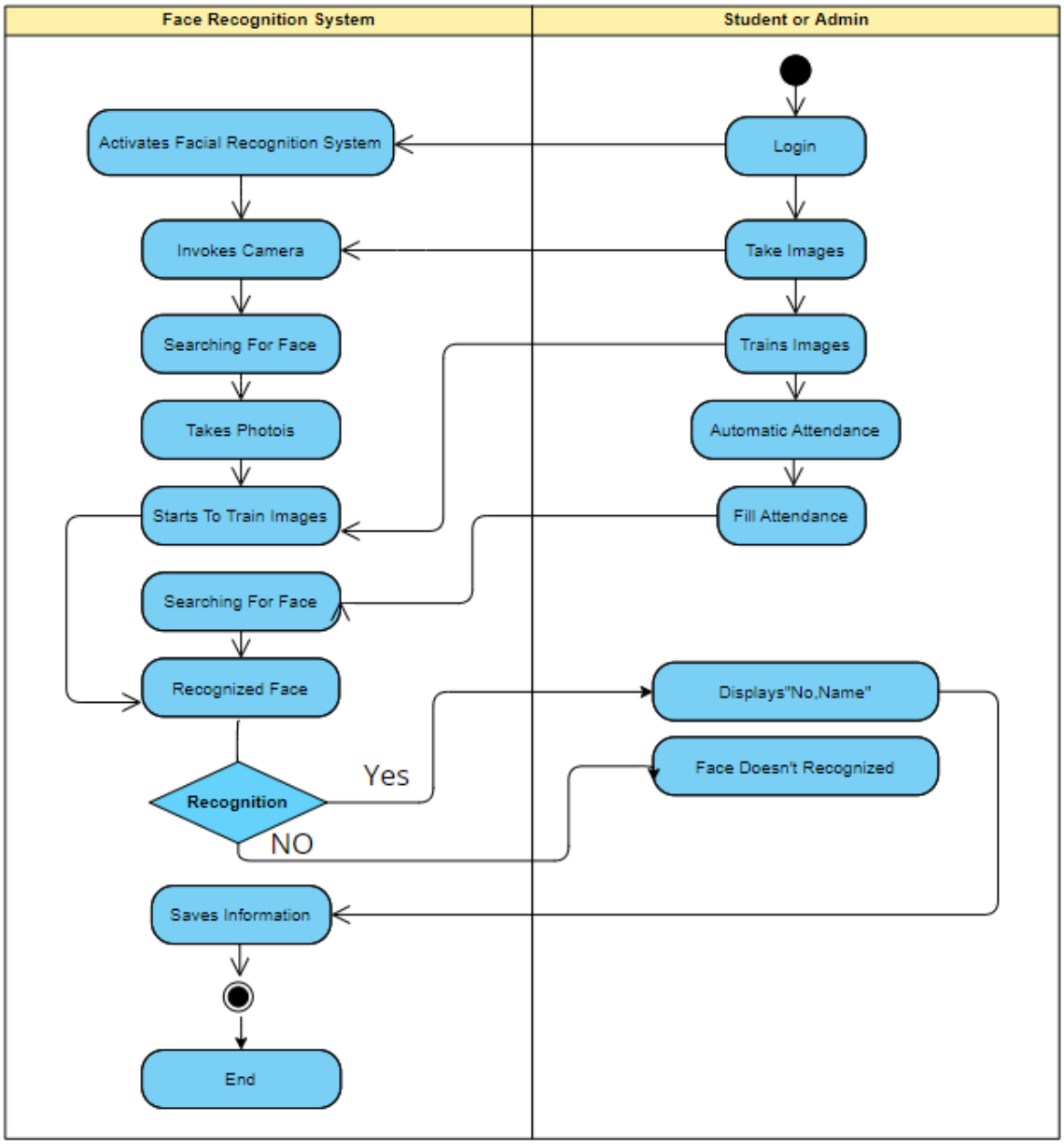

Fig 4.1.4 Activity diagram for Real Time Face Attendance System

CHAPTER 5
SYSTEM ARCHITECTURE

SYSTEM ARCHITECTURE:

5.1 ARCHITECTURE OVERVIEW

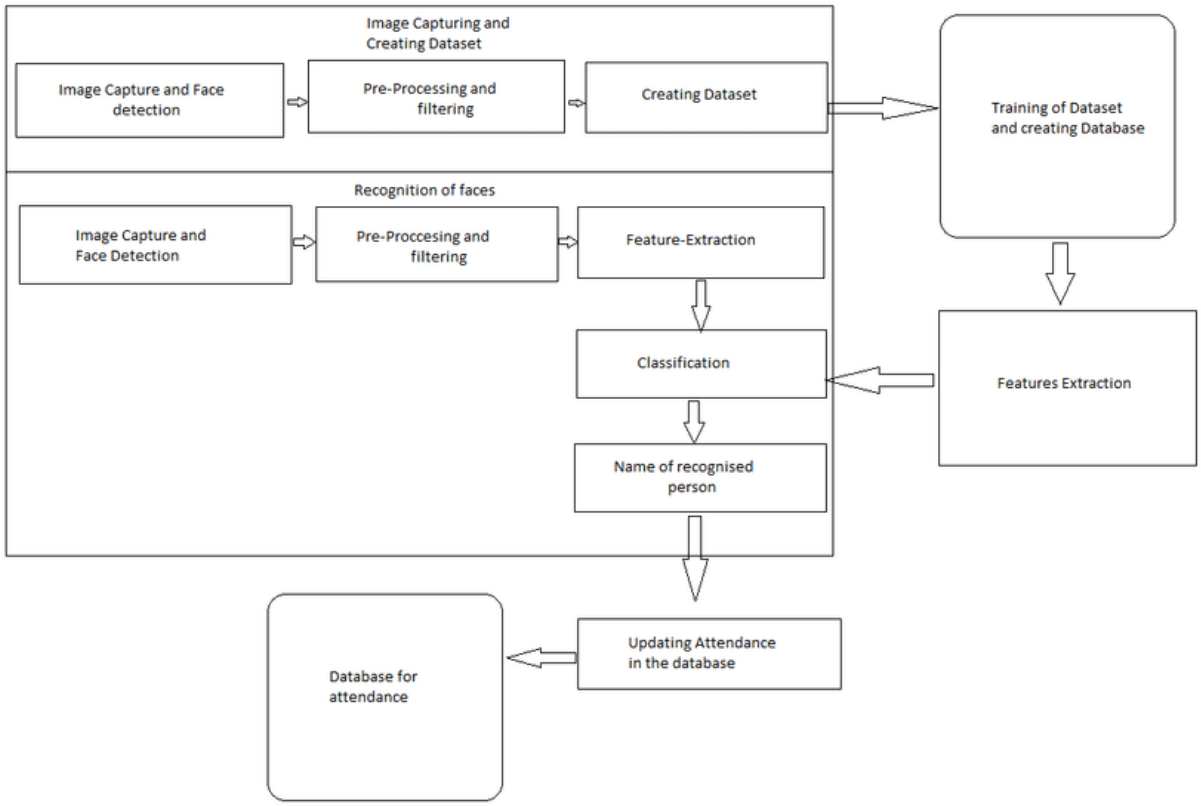

Fig 5.1 Architecture diagram for the Real Time Face Attendance System

The Architecture diagram for the Real Time Face Attendance System is to identify the individuals by comparing their input image obtained from recording video frame with respect to train image. This proposed approach able to detect and localize face from an input facial image, which is obtained from the recording video frame. Besides, The test images and training images have to be captured by using the same device to ensure no quality difference. In addition, the students have to register in the database to be recognized. The enrolment can be done on the spot through the user-friendly interface.

5.2 MODULE DESCRIPTION

- Sign in module
  - Can enter the Enrollment
  - Can enter the Name

- Training module
  - Can take several images for Training the machine

- Automatic Attendance module
  - Can Detecting the Face
  - Can Generate Attendance by Detecting the Face

- Subject module
  - Can Enter Subject Name
  - Can Check sheet

- Check Register module
  - Displays Enrollment, Name, Date, Time

CHAPTER 6
METHODOLOGY

METHODOLOGY:

6.1 Methodology Flow

The approach performs face recognition based student attendance system. The methodology flow begins with the capture of image by using simple and handy interface, followed by pre-processing of the captured facial images, then feature extraction from the facial images, subjective selection and lastly classification of the facial images to be recognized. Both LBP and PCA feature extraction methods are studied in detail and computed in this proposed approach in order to make comparisons. LBP is enhanced in this approach to reduce the illumination effect. An algorithm to combine enhanced LBP and PCA is also designed for subjective selection in order to increase the accuracy. The details of each stage will be discussed in the following sections.

6.2 Input Images

Although our own database should be used to design real time face recognition student attendance system, the databases that are provided by the previous researchers are also used to design the system more effectively, efficiently and for evaluation purposes

Yale face database is used as both training set and testing set to evaluate the performance. Yale face database contains one hundred and sixty-five grayscale images of fifteen individuals. There are eleven images per individual; each image of the individual is in different condition. The conditions included centre-light, with glasses, happy, left-light, without glasses, normal, right-light, sad, sleepy, surprised and wink. These different variations provided by the database is able to ensure the system to be operated consistently in variety of situations and conditions.

For our own database, the images of students are captured by using laptop built in camera and mobile phone camera. Each student provided four images, two for training set and two for testing set. The images captured by using laptop built in camera are categorized as low quality images, whereas mobile phone camera captured images are categorized as high quality images. The high quality images consists of seventeen students while low quality images consists of twenty-six students.

Fig 6.1 Sample of High Quality Images

Fig 6.2 Sample of Low Quality Images

6.3 Face Detection

Viola-Jones object detection framework will be used to detect the face from the video camera recording frame. The working principle of Viola-Jones algorithm is mentioned in Chapter 2. The limitation of the Viola-Jones framework is that the facial image has to be a frontal upright image, the face of the individual must point towards the camera in a video frame

6.3.1 Pre-Processing

Testing set and training set images are captured using a camera. There are unwanted noise and uneven lighting exists in the images. Therefore, several pre-processing steps are necessary before proceeding to feature extraction

Pre-processing steps that would be carried out include scaling of image, median filtering, conversion of color images to grayscale images and adaptive histogram equalization. The details of these steps would be discussed in the later sections.

6.3.2 Scaling of Image

Scaling of images is one of the frequent tasks in image processing. The size of the images has to be carefully manipulated to prevent loss of spatial information. (Gonzalez, R. C., & Woods, 2008), In order to perform face recognition, the size of the image has to be equalized. This has become crucial, especially in the feature extraction process, the test images and training images have to be in the same size and dimension to ensure the precise outcome. Thus, in this proposed approach test images and train images are standardize at size 250 × 250 pixels.

6.4 Feature Extraction

Different facial images mean there are changes in textural or geometric information. In order to perform face recognition, these features have to be extracted from the facial images and classified appropriately. In this project, enhanced LBP and PCA are used for face recognition. The idea comes from nature of human visual perception which performs face recognition depending on the local statistic and global statistic features. Enhanced LBP extracts the local grayscale features by performing feature extraction on a small region throughout the entire image. On the other hand, PCA extracts the global grayscale features which means feature extraction is performed on the whole image

6.4.1 Working Principle of Original LBP

LBP is basically a texture based descriptor which it encoded local primitive into binary string. (Timo Ojala et al., 2002). The original LBP operator works on a 3 × 3 mask size. 3 × 3 mask size contains 9 pixels. The center pixel will be used as a threshold to convert the neighboring pixels (the other 8 pixels) into binary digit. If the neighboring pixel value is larger than the center pixel value, then it is assigned to 1, otherwise it is assigned to 0. After that, the neighborhoods pixel bits are concatenated to a binary code to form a byte value representing the center pixel. Figure 6.3 shows an example of LBP conversion

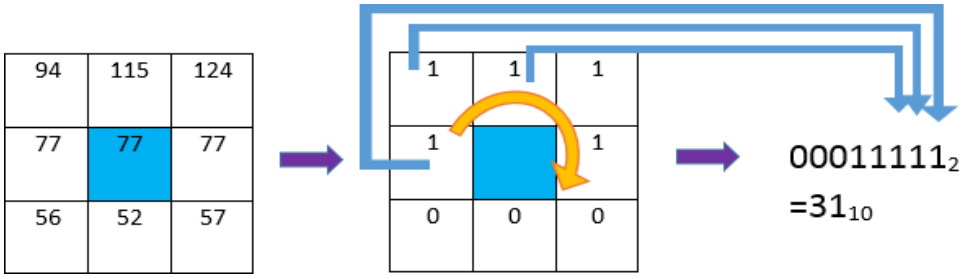

Fig 6.3 Example of LBP Conversion

6.4.2 Working Principle of Proposed LBP

The original LBP operator is composed of 3 × 3 filter size with 9 pixels. Instead of the circular pattern, it looks more rectangular in shape. The 9 pixels adjacent to each other means every detail will be taken as sampling points even the non-essential details. It is more affected by uneven lighting condition because the small filter size emphasizes small scale detail (Lee and Li, 2007), even the shadow created by non-uniform lighting condition. In our proposed approach, a larger radius size, R is implemented in LBP operator. In the paper of Md. Abdur Rahim et.al (2013), the equation of modifying the radius size has been introduced. However, the paper did not mention the effect of changing the radius size. In the proposed approach, analysis is done on different radius sizes in order to enhance the system and reduce the illumination effect.

Basically, the increasing in the size of the radius means extending the circular pattern of LBP externally. The green spots within the blocks indicate the sampling pixels to be encoded into binary string. For the sampling pixel located in between the blocks, it indicates the average pixel value is computed from the adjacent pixels (diagonal).

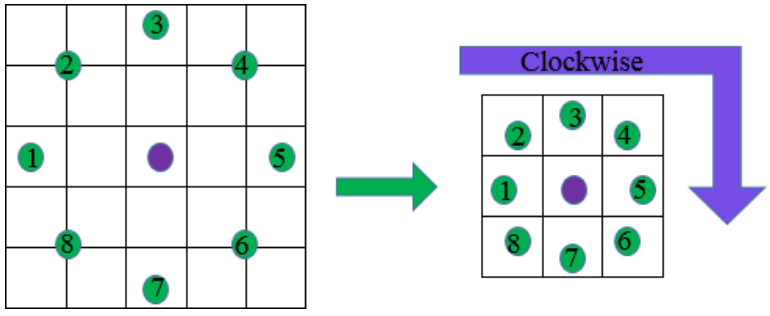

Fig 6.4 Proposed LBP Operator with Radius 2 and Its Encoding Pattern.

6.4.3 Working Principle of PCA

In this proposed approach, PCA face recognition is studied, as it is one of the popular face recognition methods that was suggested and used by the previous researchers. The accuracy of PCA is computed in order to compare with the enhanced LBP.

PCA includes a few steps which will briefly be described in the following paragraphs. For PCA, the image scale, length (M) and height (M) is not so important. This is because PCA is mostly dealing with number of total images, N instead of M. However, same size of test image and training image is a must for PCA computation. Same length and height of the image is assumed in the following equation for illustration. Given a training set of N images with size $M \times M$, the first step of PCA is to convert two dimensional vectors to one dimensional vector. The one dimensional vector can be either column vector or row vector. In this approach, the column vector conversion is done. For each facial image with matrix notation $M \times M$ will be converted to column vector Γi, with dimension $M^2 \times 1$.There are N facial images, each face is represented by column vector Γ1, Γ2, Γ3, .., ΓN. Feature vector of each face is stored in this column vector. The dimension reduced face matrix is constructed by concatenating every single column vector

CHAPTER 7

SYSTEM IMPLEMENTATION

7.SYSTEM IMPLEMENTATION:

7.1 Program / Coding:

```
import tkinter as tk
from tkinter import *
import cv2
import csv
import os
import numpy as np
from PIL import Image,ImageTk
import pandas as pd
import datetime
import time
#####Window is our Main frame of system
window = tk.Tk()
window.title("Face Attendance using Machine Learning")
window.geometry('1280x720')
window.configure(background='snow')
def manually_fill():
global sb
sb = tk.Tk()
sb.title("Please enter subject Name :-")
sb.geometry('580x320')
sb.configure(background='snow')
def err_screen_for_subject():
def ec_delete():
ec.destroy()
global ec
ec = tk.Tk()
ec.geometry('300x100')
ec.title('Warning!!')
```

```
ec.configure(background='snow')
Label(ec, text='Please enter your subject name!!!', fg='red', bg='white', font=('times', 16, '
bold ')).pack()
Button(ec, text='OK', command=ec_delete, fg="black", bg="lawn green", width=9,
height=1, activebackground="Red",
font=('times', 15, ' bold ')).place(x=90, y=50)
def fill_attendance():
ts = time.time()
Date = datetime.datetime.fromtimestamp(ts).strftime('%Y_%m_%d')
timeStamp = datetime.datetime.fromtimestamp(ts).strftime('%H:%M:%S')
Time = datetime.datetime.fromtimestamp(ts).strftime('%H:%M:%S')
Hour, Minute, Second = timeStamp.split(":")
####Creatting csv of attendance
##Create table for Attendance
date_for_DB = datetime.datetime.fromtimestamp(ts).strftime('%Y_%m_%d')
#global cursor
global subb
subb=SUB_ENTRY.get()
DB_table_name = str(subb + "_" + Date + "_Time_" + Hour + "_" + Minute + "_" + Second)
import pymysql.connections
global cursor
###Connect to the database
try:
global mycursor
connection = pymysql.connect(host='localhost', user='root', password='root',
db='manually_fill_attendance')
mycursor = connection.cursor()
except Exception as e:
print(e)
sql = "CREATE TABLE " + DB_table_name + """
(ID INT NOT NULL AUTO_INCREMENT, ENROLLMENT varchar(100) NOT NULL,
```

```
NAME VARCHAR(50) NOT NULL, DATE VARCHAR(20) NOT NULL, TIME VARCHAR(20) NOT NULL,PRIMARY KEY (ID) );"""
try:
mycursor.execute(sql)  ##for create a table
except Exception as ex:
print(ex)
if subb==":
err_screen_for_subject()
else:
sb.destroy()
MFW = tk.Tk()
#MFW.title("Manually attendance of "+ str(subb))
MFW.geometry('880x470')
MFW.configure(background='snow')
def del_errsc2():
errsc2.destroy()
def err_screen1():
global errsc2
errsc2 = tk.Tk()
errsc2.geometry('330x100')
errsc2.title('Warning!!')
errsc2.configure(background='snow')
Label(errsc2, text='Please enter Student & Enrollment!!!', fg='red', bg='white',
font=('times', 16, ' bold ')).pack()
Button(errsc2, text='OK', command=del_errsc2, fg="black", bg="lawn green", width=9, height=1,
activebackground="Red", font=('times', 15, ' bold ')).place(x=90, y=50)
def testVal(inStr, acttyp):
if acttyp == '1':  # insert
if not inStr.isdigit():
return False
```

```
return True
ENR = tk.Label(MFW, text="Enter Enrollment", width=15, height=2, fg="white", bg="blue2",
 font=('times', 15, ' bold '))
ENR.place(x=30, y=100)
STU_NAME = tk.Label(MFW, text="Enter Student name", width=15, height=2, fg="white", bg="blue2",
font=('times', 15, ' bold '))
STU_NAME.place(x=30, y=200)
global ENR_ENTRY
ENR_ENTRY = tk.Entry(MFW, width=20,validate='key', bg="yellow", fg="red", font=('times', 23, ' bold '))
ENR_ENTRY['validatecommand'] = (ENR_ENTRY.register(testVal), '%P', '%d')
ENR_ENTRY.place(x=290, y=105)
def remove_enr():
ENR_ENTRY.delete(first=0, last=22)
STUDENT_ENTRY = tk.Entry(MFW, width=20, bg="yellow", fg="red", font=('times', 23, ' bold ')) STUDENT_ENTRY.place(x=290, y=205)
def remove_student():
STUDENT_ENTRY.delete(first=0, last=22)
####get important variable
def enter_data_DB():
ENROLLMENT = ENR_ENTRY.get()
STUDENT = STUDENT_ENTRY.get()
if ENROLLMENT==":
err_screen1()
elif STUDENT==":
err_screen1()
else:
time = datetime.datetime.fromtimestamp(ts).strftime('%H:%M:%S')
Hour, Minute, Second = time.split(":")
```

```
Insert_data = "INSERT INTO " + DB_table_name + "
(ID,ENROLLMENT,NAME,DATE,TIME) VALUES (0, %s, %s, %s,%s)"
VALUES = (str(ENROLLMENT), str(STUDENT), str(Date), str(time))
try:
mycursor.execute(Insert_data, VALUES)
except Exception as e:
print(e)
ENR_ENTRY.delete(first=0, last=22)
STUDENT_ENTRY.delete(first=0, last=22)
def create_csv():
import csv
mycursor.execute("select * from " + DB_table_name + ";")
csv_name='./Manually Attendance/'+DB_table_name+'.csv'
with open(csv_name, "w") as csv_file:
csv_writer = csv.writer(csv_file)
csv_writer.writerow([i[0] for i in mycursor.description])  # write headers
csv_writer.writerows(mycursor)
O="CSV created Successfully"
Notifi.configure(text=O, bg="Green", fg="white", width=33, font=('times', 19, 'bold'))
Notifi.place(x=180, y=380)
import csv
import tkinter
root = tkinter.Tk()
root.title("Attendance of " + subb)
root.configure(background='snow')
with open(csv_name, newline="") as file:
reader = csv.reader(file)
r = 0
Notification.place(x=250, y=400)
except FileExistsError as F:
f = 'Student Data already exists'
```

```
Notification.configure(text=f, bg="Red", width=21)
Notification.place(x=450, y=400)

###for choose subject and fill attendance
def subjectchoose():
    def Fillattendances():
        sub=tx.get()
        now = time.time()  ###For calculate seconds of video
        future = now + 20
if time.time() < future:
if sub == '':
err_screen1()
else:
recognizer = cv2.face.LBPHFaceRecognizer_create()
# cv2.createLBPHFaceRecognizer()
try:
recognizer.read("TrainingImageLabel/Trainner.yml")
except:
e = 'Model not found,Please train model'
Notifica.configure(text=e, bg="red", fg="black", width=33, font=('times', 15, 'bold'))
Notifica.place(x=20, y=250)
harcascadePath = "haarcascade_frontalface_default.xml"
faceCascade = cv2.CascadeClassifier(harcascadePath)
df = pd.read_csv("StudentDetails/StudentDetails.csv")
cam = cv2.VideoCapture(0)
font = cv2.FONT_HERSHEY_SIMPLEX
col_names = ['Enrollment', 'Name', 'Date', 'Time']
attendance = pd.DataFrame(columns=col_names)
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:
global Id
Id, conf = recognizer.predict(gray[y:y + h, x:x + w])
if (conf <70):
print(conf)
global Subject
global aa
global date
global timeStamp
Subject = tx.get()
ts = time.time()
date = datetime.datetime.fromtimestamp(ts).strftime('%Y-%m-%d')
timeStamp = datetime.datetime.fromtimestamp(ts).strftime('%H:%M:%S')
aa = df.loc[df['Enrollment'] == Id]['Name'].values
global tt
tt = str(Id) + "-" + aa
En = '15624031' + str(Id)
attendance.loc[len(attendance)] = [Id, aa, date, timeStamp]
cv2.rectangle(im, (x, y), (x + w, y + h), (0, 260, 0), 7)
cv2.putText(im, str(tt), (x + h, y), font, 1, (255, 255, 0,), 4)
else:
Id = 'Unknown'
tt = str(Id)
cv2.rectangle(im, (x, y), (x + w, y + h), (0, 25, 255), 7)
cv2.putText(im, str(tt), (x + h, y), font, 1, (0, 25, 255), 4)
if time.time() > future:
break
attendance = attendance.drop_duplicates(['Enrollment'], keep='first')
cv2.imshow('Filling attedance..', im)
key = cv2.waitKey(30) & 0xff
```

```
if key == 27:
break
ts = time.time()
date = datetime.datetime.fromtimestamp(ts).strftime('%Y-%m-%d')
timeStamp = datetime.datetime.fromtimestamp(ts).strftime('%H:%M:%S')
Hour, Minute, Second = timeStamp.split(":")
fileName = "Attendance/" + Subject + "_" + date + "_" + Hour + "-" + Minute + "-" +
Second + ".csv"
attendance = attendance.drop_duplicates(['Enrollment'], keep='first')
print(attendance)
attendance.to_csv(fileName, index=False)
##Create table for Attendance
date_for_DB = datetime.datetime.fromtimestamp(ts).strftime('%Y_%m_%d')
DB_Table_name = str( Subject + "_" + date_for_DB + "_Time_" + Hour + "_" + Minute +
"_" + Second)
import pymysql.connections
###Connect to the database
try:
global mycursor
connection = pymysql.connect(host='localhost', user='root', password='', db='Face_reco_fill')
mycursor = connection.cursor()
except Exception as e:
print(e)
sql = "CREATE TABLE " + DB_Table_name + """ (ID INT NOT NULL
AUTO_INCREMENT, ENROLLMENT varchar(100) NOT NULL, NAME VARCHAR(50)
NOT NULL, DATE VARCHAR(20) NOT NULL, TIME VARCHAR(20) NOT NULL,
PRIMARY KEY (ID)
);"""
####Now enter attendance in Database
insert_data = "INSERT INTO " + DB_Table_name + "
(ID,ENROLLMENT,NAME,DATE,TIME) VALUES (0, %s, %s, %s,%s)"
```

```
username = un_entr.get()
password = pw_entr.get()
if username == 'admin' :
if password == 'admin':
win.destroy()
import csv
import tkinter
root = tkinter.Tk()
root.title("Student Details")
root.configure(background='snow')
cs = './StudentDetails/StudentDetails.csv'
with open(cs, newline="") as file:
reader = csv.reader(file)
r = 0
for col in reader:
c = 0
for row in col:
# i've added some styling
label = tkinter.Label(root, width=8, height=1, fg="black", font=('times', 15, ' bold '),
bg="lawn green", text=row, relief=tkinter.RIDGE)
label.grid(row=r, column=c)
c += 1
r += 1
root.mainloop()
else:
valid = 'Incorrect ID or Password'
Nt.configure(text=valid, bg="red", fg="black", width=38, font=('times', 19, 'bold'))
Nt.place(x=120, y=350)
else:
valid ='Incorrect ID or Password'
Nt.configure(text=valid, bg="red", fg="black", width=38, font=('times', 19, 'bold'))
```

```
Nt.place(x=120, y=350)
Nt = tk.Label(win, text="Attendance filled Successfully", bg="Green", fg="white",
width=40,
height=2, font=('times', 19, 'bold'))
# Nt.place(x=120, y=350)

un = tk.Label(win, text="Enter username", width=15, height=2, fg="white", bg="blue2",
font=('times', 15, ' bold '))
un.place(x=30, y=50)

pw = tk.Label(win, text="Enter password", width=15, height=2, fg="white", bg="blue2",
font=('times', 15, ' bold '))
pw.place(x=30, y=150)

def c00():
un_entr.delete(first=0, last=22)

un_entr = tk.Entry(win, width=20, bg="yellow", fg="red", font=('times', 23, ' bold '))
un_entr.place(x=290, y=55)

def c11():
pw_entr.delete(first=0, last=22)

pw_entr = tk.Entry(win, width=20,show="*", bg="yellow", fg="red", font=('times', 23, ' bold
'))
pw_entr.place(x=290, y=155)

c0 = tk.Button(win, text="Clear", command=c00, fg="black", bg="deep pink", width=10,
height=1,
activebackground="Red", font=('times', 15, ' bold '))
c0.place(x=690, y=55)
```

```
c1 = tk.Button(win, text="Clear", command=c11, fg="black", bg="deep pink", width=10,
height=1,
activebackground="Red", font=('times', 15, ' bold '))
c1.place(x=690, y=155)
Login = tk.Button(win, text="LogIn", fg="black", bg="lime green", width=20,
height=2,
activebackground="Red",command=log_in, font=('times', 15, ' bold '))
Login.place(x=290, y=250)
win.mainloop()
###For train the model
def trainimg():
recognizer = cv2.face.LBPHFaceRecognizer_create()
global detector
detector = cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
try:
global faces,Id
faces, Id = getImagesAndLabels("TrainingImage")
except Exception as e:
l='please make "TrainingImage" folder & put Images'
Notification.configure(text=l, bg="SpringGreen3", width=50, font=('times', 18, 'bold'))
Notification.place(x=350, y=400)
recognizer.train(faces, np.array(Id))
try:
recognizer.save("TrainingImageLabel/Trainner.yml")
except Exception as e:
q='Please make "TrainingImageLabel" folder'
Notification.configure(text=q, bg="SpringGreen3", width=50, font=('times', 18, 'bold'))
Notification.place(x=350, y=400)

res = "Model Trained"  # +",".join(str(f) for f in Id)
```

```
Notification.configure(text=res, bg="SpringGreen3", width=50, font=('times', 18, 'bold'))
Notification.place(x=250, y=400)

def getImagesAndLabels(path):
imagePaths = [os.path.join(path, f) for f in os.listdir(path)]
# create empth face list
faceSamples = []
# 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 = detector.detectMultiScale(imageNp)
# If a face is there then append that in the list as well as Id of it
for (x, y, w, h) in faces:
faceSamples.append(imageNp[y:y + h, x:x + w])
Ids.append(Id)
return faceSamples, Ids

window.grid_rowconfigure(0, weight=1)
window.grid_columnconfigure(0, weight=1)

def on_closing():
from tkinter import messagebox
```

```
if messagebox.askokcancel("Quit", "Do you want to quit?"):
window.destroy()
window.protocol("WM_DELETE_WINDOW", on_closing)

message = tk.Label(window, text="Face Attendance ", bg="white", fg="black", width=50,
height=3, font=('times', 30, 'bold'))

message.place(x=80, y=20)

Notification = tk.Label(window, text="All things good", bg="Green", fg="white", width=15,
height=3, font=('times', 17, 'bold'))

lbl = tk.Label(window, text="Enter Enrollment", width=20, height=3, fg="black",
bg="white", font=('times', 15, ' bold '))
lbl.place(x=200, y=200)

def testVal(inStr,acttyp):
if acttyp == '1': #insert
if not inStr.isdigit():
return False
return True

txt = tk.Entry(window, validate="key", width=20, bg="white", fg="green", font=('times', 25,
' bold '))
txt['validatecommand'] = (txt.register(testVal),'%P','%d')
txt.place(x=550, y=210)

lbl2 = tk.Label(window, text="Enter Name", width=20, fg="black", bg="white", height=3,
font=('times', 15, ' bold '))
lbl2.place(x=200, y=300)
```

```
txt2 = tk.Entry(window, width=20, bg="white", fg="green", font=('times', 25, ' bold '))
txt2.place(x=550, y=310)
clearButton = tk.Button(window,
text="Clear",command=clear,fg="black"  ,bg="gray"  ,width=10  ,height=1
,activebackground = "Red" ,font=('times', 15, ' bold '))
clearButton.place(x=950, y=210)
clearButton1 = tk.Button(window,
text="Clear",command=clear1,fg="black"  ,bg="gray"  ,width=10 ,height=1,
activebackground = "Red" ,font=('times', 15, ' bold '))
clearButton1.place(x=950, y=310)
AP = tk.Button(window, text="Check Register
students",command=admin_panel,fg="black"  ,bg="pale goldenrod"  ,width=19 ,height=1,
activebackground = "Red" ,font=('times', 15, ' bold '))
AP.place(x=990, y=500)
takeImg = tk.Button(window, text="Take
Images",command=take_img,fg="black"  ,bg="pale green"  ,width=20  ,height=3,
activebackground = "Red" ,font=('times', 15, ' bold '))
takeImg.place(x=90, y=500)
trainImg = tk.Button(window, text="Train Images",fg="black",command=trainimg
,bg="khaki1"  ,width=20  ,height=3, activebackground = "Red" ,font=('times', 15, ' bold '))
trainImg.place(x=390, y=500)
FA = tk.Button(window, text="Automatic
Attendance",fg="black",command=subjectchoose  ,bg="alice blue"  ,width=20  ,height=3,
activebackground = "Red" ,font=('times', 15, ' bold '))
FA.place(x=690, y=500)
#quitWindow = tk.Button(window, text="Manually Fill Attendance",
command=manually_fill  ,fg="black"  ,bg="thistle"  ,width=20  ,height=3, activebackground
= "Red" ,font=('times', 15, ' bold '))
#quitWindow.place(x=990, y=500)
```

CHAPTER 8
SYSTEM TESTING

SYSTEM TESTING:

8.1 TEST CASES:

TEST CASE ID	TESTCASE/ACTION TO BE PERFORMED	EXPECTED RESULT	ACTUAL RESULT	PASS/ FAIL
1	Entering a Number in a the Enrollment Text Box	Display Student Number	Display Student Number	Pass
2	Entering a Name in a the Name Text Box	Display Name Number	Display Name Number	Pass
3	Giving the details and Clicking "Take Images" Button	Image saved for Enrollment Student	Image saved for Enrollment Student	Pass
4	Clicking the "Train images" Button	Model Trained	Model Trained	Pass
5	Clicking the "Automatic Attendance" Button	Model Trained	Model Trained	Pass
6	Entering the Subject Name in Pop up box	Name Entered	Data was Entered	Pass
7	Clicking the "Fill Attendance" Button in the Pop up box	Attendance filled Successfully	Attendance filled Successfully	Pass
8	Clicking the "Check Register student" Button	Display Student Detail	Display Student Detail	Pass
9	Giving the Admin details and Clicking "Login" Button in pop up box	Display Student Attendance Detail	Display Student Attendance Detail	Pass

CHAPTER 9
CONCLUSION

CONCLUSION:

9.1 Results & Discussion

In this proposed approach, face recognition student attendance system with user-friendly interface is designed by using MATLAB GUI(Graphic User Interface). A few buttons are designed in the interface, each provides specific function, for example, start button is to initialize the camera and to perform face recognition automatically according to the face detected, register button allows enrolment or registrations of students and update button is to train the latest images that have been registered in the database. Lastly, browse button and recognize button is to browse facial images from selected database and recognized the selected image to test the functionality of the system respectively.

This proposed approach provides a method to perform face recognition for the student attendance system, which is based on the texture based features of facial images. Face recognition is the identification of an individual by comparing his/her real-time captured image with stored images in the database of that person. Thus, the training set has to be chosen based on the latest appearance of an individual other than taking important factors for instance illumination into consideration.

The proposed approach is being trained and tested on different datasets. Yale face database which consists of one hundred and sixty-five images of fifteen individuals with multiple conditions is implemented. However, this database consists of only grayscale images. Hence, our own database with color images which is further categorized into high quality set and the low quality set, as images are different in their quality: some images are blurred while some are clearer. The statistics of each data set have been discussed in the earlier chapter

9.2 Conclusion

In this approach, a face recognition based automated student attendance system is thoroughly described. The proposed approach provides a method to identify the individuals by comparing their input image obtained from recording video frame with respect to train image. This proposed approach able to detect and localize face from an input facial image, which is obtained from the recording video frame. Besides, it provides a method in pre-processing stage to enhance the image contrast and reduce the illumination effect. Extraction of features from the facial image is performed by applying both LBP and PCA. The algorithm designed to combine LBP and PCA able to stabilize the system by giving consistent results. The accuracy of this proposed approach is 100 % for high-quality images, 92.31 % for low-quality images and 95.76 % of Yale face database when two images per person are trained.

As a conclusion for analysis, the extraction of facial feature could be challenging especially in different lighting. In pre-processing stage, Contrast Limited Adaptive Histogram Equalization (CLAHE) able to reduce the illumination effect.

9.3 Future Enhancements

- The future work is to improve the recognition rate of algorithms when there are unintentional changes in a person like using scarf, beard etc..
- The face recognition model would be done more precisely so that maximum accuracy can be achieved.
- The marked attendance is stored in only one excel, rather different excels for different date can be created.

The above mentioned are the future enhancements that can be done to make this project much more dynamic.

CHAPTER 10
APPENDICES

10.1 Sample Screens

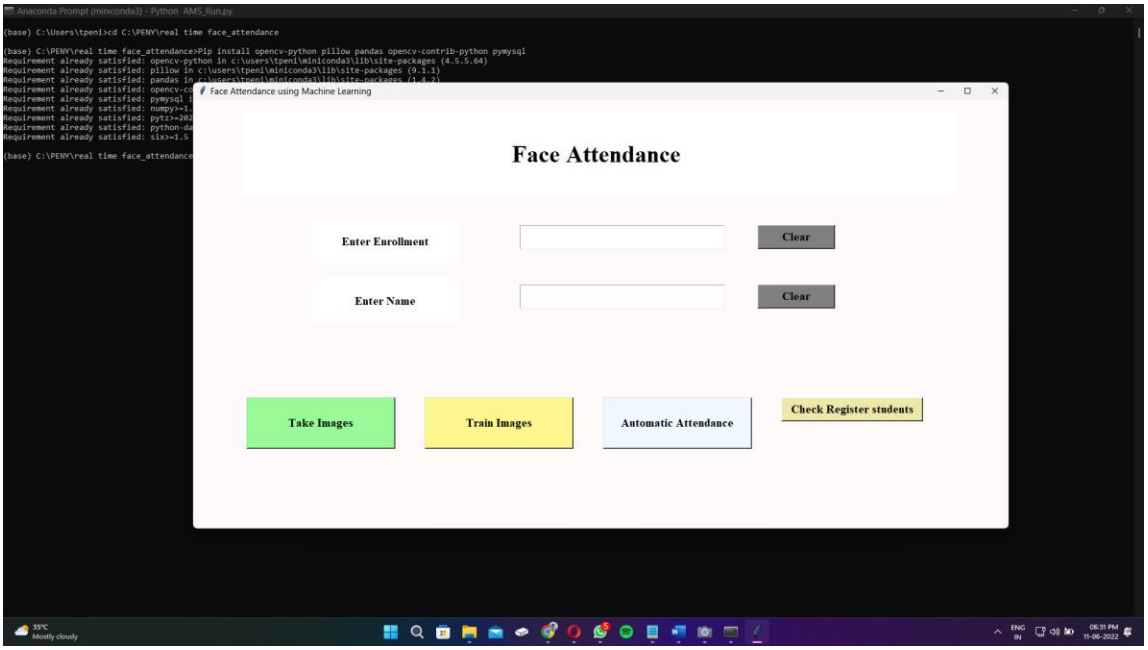

Fig 10.1 First window for Real Time Face Attendance System

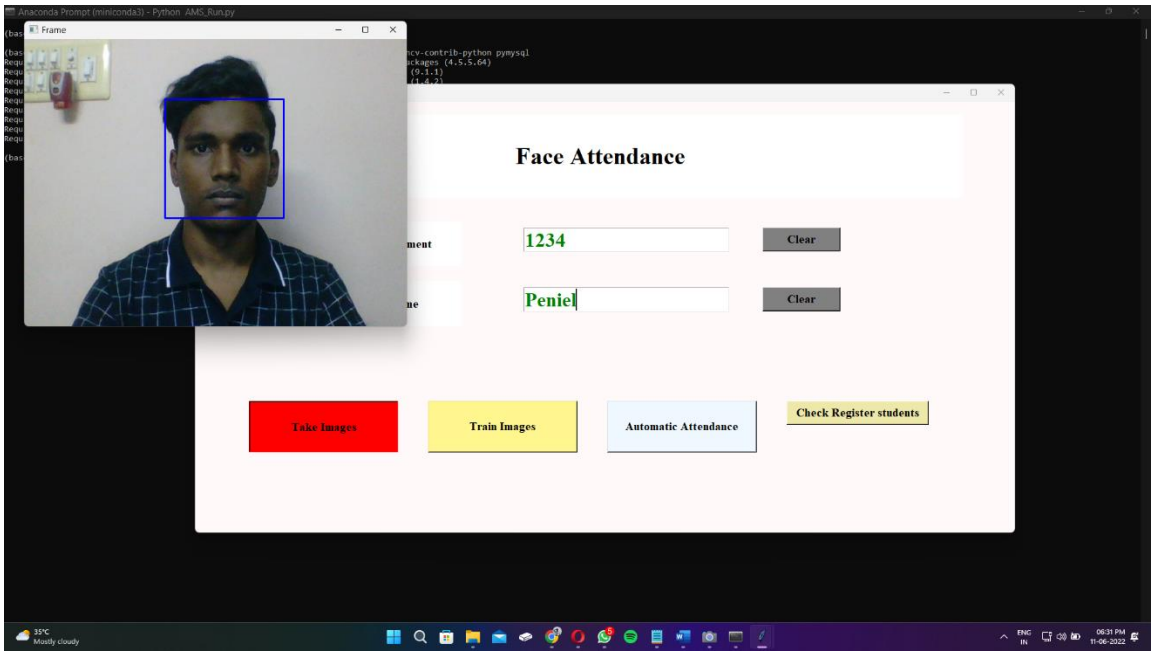

Fig 10.2 Register the Details to Capture the Student Face

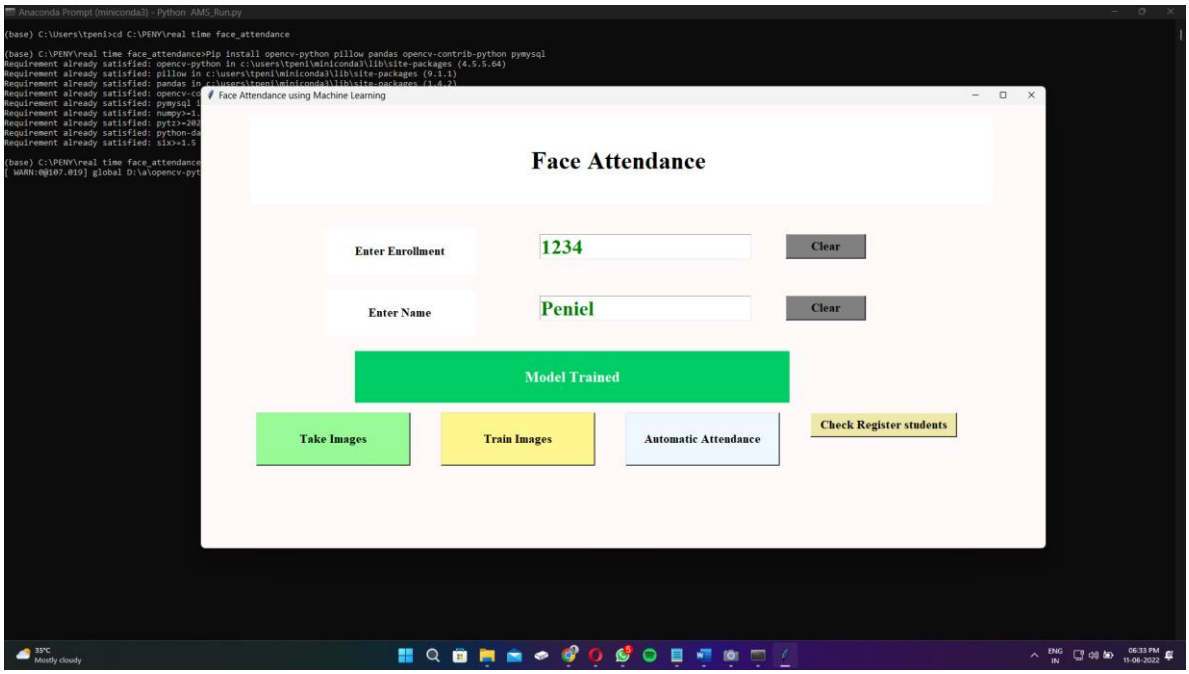

Fig 10.3 Student Face Trained

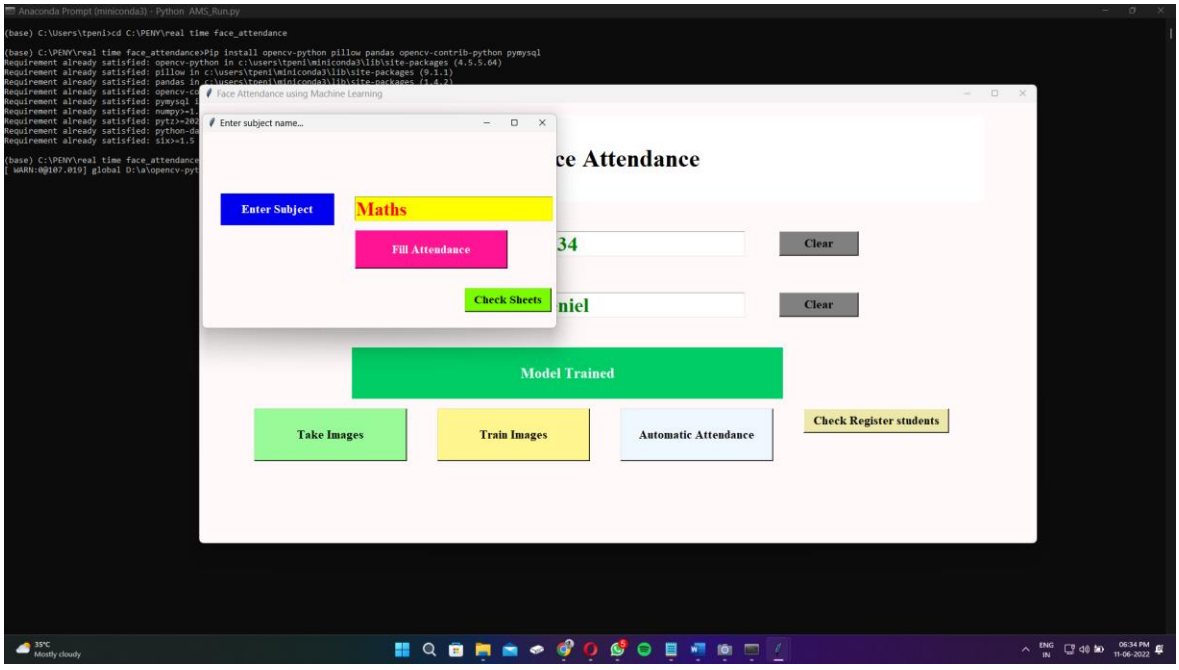

Fig 10.4 Enter the subject Details to Fill Attendance

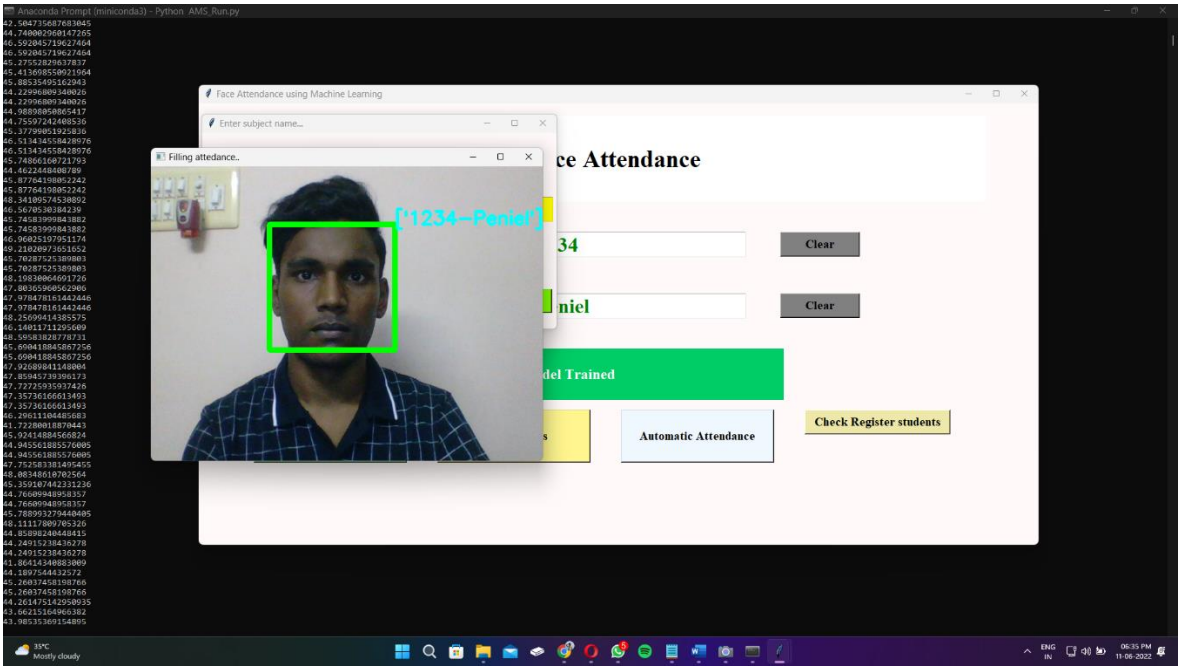

Fig 10.5 Face Identified

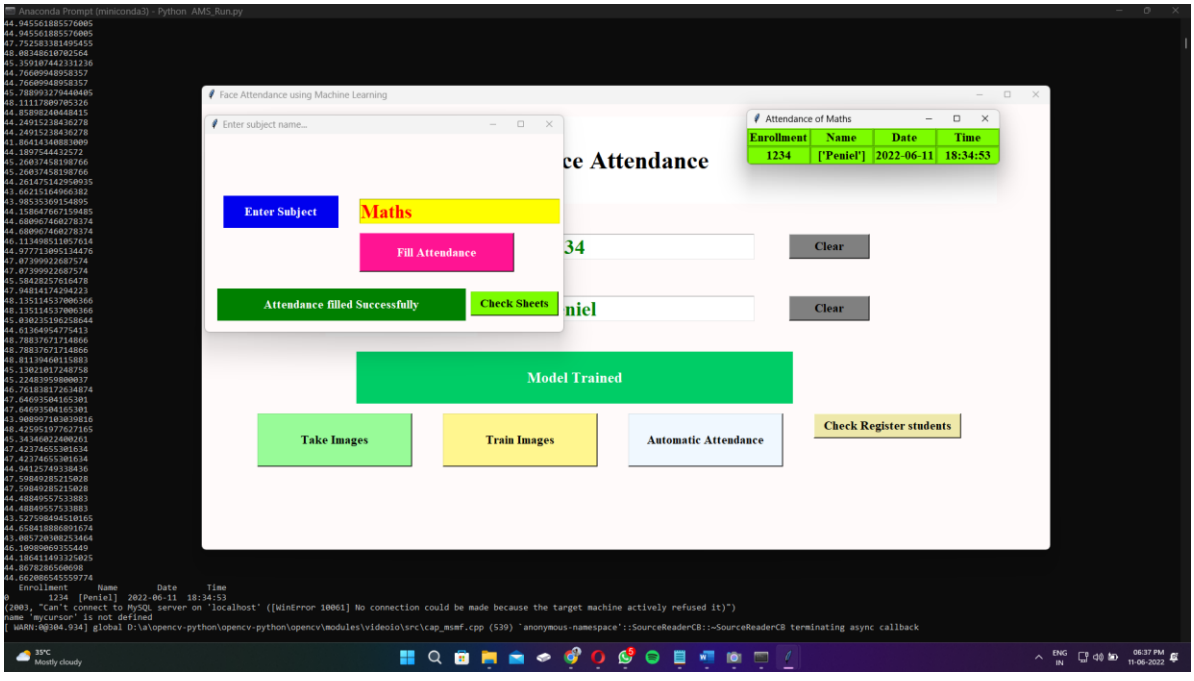

Fig 10.6 Attendance Filled Automatically

REFERENCES

Robinson-Riegler, G., & Robinson-Riegler, B. (2008). Cognitive psychology: applying the science of the mind. Boston, Pearson/Allyn and Bacon.

Margaret Rouse. (2012). What is facial recognition? - Definition from WhatIs.com. [online] Available at:
http://whatis.techtarget.com/definition/facial-recognition [Accessed 25 Mar. 2018].

Solon, O. (2017). Facial recognition database used by FBI is out of control, House committee hears. [online] the Guardian. Available at:
https://www.theguardian.com/technology/2017/mar/27/us-facial-recognition- database-fbi-drivers-licenses-passports [Accessed 25 Mar. 2018].

Robert Silk. (2017). Biometrics: Facial recognition tech coming to an airport near you: Travel Weekly. [online] Available at:
http://www.travelweekly.com/Travel-News/Airline-News/Biometrics-Facial- recognition-tech-coming-airport-near-you [Accessed 25 Mar. 2018].

Sidney Fussell. (2018). NEWS Facebook's New Face Recognition Features: What We Do (and Don't) Know. [online] Available at:
https://gizmodo.com/facebooks-new-face-recognition-features-what-we-do-an- 1823359911 [Accessed 25 Mar. 2018].

deAgonia, M. (2017). Apple's Face ID [The iPhone X's facial recognition tech explained]. [online] Computerworld. Available at: https://www.computerworld.com/article/3235140/apple-ios/apples-face-id-the- iphone-xs-facial-recognition-tech-explained.html [Accessed 25 Mar. 2018].

Ashley DuVal. (2012).Face Recognition Software -History of Forensic Psychology. [online] Available at:
http://forensicpsych.umwblogs.org/research/criminal-justice/face-recognition- software/ [Accessed 12 Apr. 2018].

Jesse Davis West. (2017). History of Face Recognition - Facial recognition software. [online] Available at: https://www.facefirst.com/blog/brief-history-of-face- recognition-software/ [Accessed 25 Mar. 2018].

Pooja G.R, et al. (2010). An automated Attendance System Using Image Processing. International Journal Of Advanced Networking & Applications.

Wagh, P., Thakare, R., Chaudhari, J. and Patil, S. (2015). Attendance system based on face recognition using eigen face and PCA algorithms. International Conference on Green Computing and Internet of Things.

S.Aanjanadevi, Dr.V.Palanisamy,R.Anandha Jothi. (2017). A Study on Secure Online Voting

System using Biometrics Face Detection and Recognition Algorithms.

Ojala, T., Pietikainen, M. and Maenpaa, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(7), pp.971-987.

Md. Abdur Rahim (2013), Face Recognition Using Local Binary Patterns. Global Journal of Computer Science and Technology Graphics & Vision V13 (4) Version 1.0.

Kasar, M., Bhattacharyya, D. and Kim, T. (2016). Face Recognition Using Neural Network: A Review. International Journal of Security and Its Applications, 10(3), pp.81-100.

Taigman, Y. et al., 2014. DeepFace: Closing the Gap to Human-Level Performance in Face Verification. IEEE Conference on Computer Vision and Pattern Recognition.

Divyarajsinh N Parmar et al. (2013).Face Recognition Methods and Applications. Int.J.Computer Technology & Applications,Vol 4 (1),84-86.

Deepesh Raj (2011), A Realtime Face Recognition system using PCA and various Distance Classifiers. CS676 : Computer Vision and Image Processing. Available at: http://home.iitk.ac.in/~draj/cs676/project/index.html [Accessed March 25, 2018].

Kamila, NK. (2015.). Handbook of research on emerging perspectives in intelligent pattern recognition, analysis, and image processing.

Lee, S. and Li, S. (2007). Advances in Biometrics. Berlin: Springer.

Kalyan Sourav Dash. (2014). Face recognition using PCA - File Exchange - MATLAB Central. [online] Mathworks.com. Available at: https://www.mathworks.com/matlabcentral/fileexchange/45750-face-recognition- using-pca [Accessed 11 Apr. 2018].

Mathworks.com. (2018). Detect objects using the Viola-Jones algorithm - MATLAB. [online] Available at:https://www.mathworks.com/help/vision/ref/vision.cascadeobjectdetector- system-object.html [Accessed 22 Apr. 2018].

Cvc.cs.yale.edu. (1997). Yale Face Database. [online] Available at: http://cvc.cs.yale.edu/cvc/projects/yalefaces/yalefaces.html [Accessed 22 Apr. 2018].
Luxand.com. (2018). Luxand - Face Recognition, Face Detection and Facial Feature Detection Technologies. [online] Available at: https://www.luxand.com/ [Accessed 22 Apr. 2018].

Kihwan Kim. (2011). Computer Vision Final Project:Viola-Jones & Morphology- based Face Detector. [online] Available at: https://www.cc.gatech.edu/~kihwan23/imageCV/Final2005/FinalProject_KH.htm [Accessed 22 Apr. 2018].