**ABSTRACT**

**PROJECT TOPIC: “ATTENDANCE SYSTEM USING FACE RECOGNITION”.**

**DOMAIN: (ML) Image Processing**

**In an educational organization there is an issue for filling the attendance given the manipulation and the physical absence of the student. The common attendance system using paper and pen approach along with the data entry is very time consuming and prone to errors.**

**The purpose of this project is to get a more reliable and accurate attendance system. We intend to bring a Face Recognition Attendance System opens a terminal i.e., a camera which scans the face and keeps the attendance record of the student.**

**From this project, we hope to build an attendance system which is more reliable and has better accuracy.**

**GROUP MEMBERS:**

**SILASH TRIKHATRI (17IT101001)**

**TABLE OF CONTENTS**

**Chapter No. Title Page No.**

**Acknowledgements i**

**Abstract ii**

**Symbols, Acronyms and Abbreviations iii**

**List of Figures iv**

**List of Tables v**

**Chapter 1 Introduction**

* 1. Introduction
  2. Applications
  3. Overview of Report
  4. Concepts
     1. Image Processing
     2. Face Recognition

**Chapter 2 Literature Review**

* 1. A Review on Face Recognition Attendance System
  2. Web Based Intelligence Surveillance System

**Chapter 3 Problem Statements and Objectives**

3.1 Problem Definition

3.2 Objectives

3.3 Procedure

3.4 Gantt Chart

**Chapter 4 Theoretical Analysis/Project Details**

4.1 Methodology

4.2 Modules

4.3 Image Capturing

4.4 Storing the Attendance

4.5 Codes Implemented

**Chapter 5 Results and Discussion**

5.1 Description of Findings

5.2 Limitations and Further works

**Chapter 6 Conclusion**

5.1 Conclusion

5.2 Future Scopes

**References**

**List of Figures**

S No. Title Page No.

1. Gantt Chart

2. Modules

3.

4.

5.

**List of Tables**

S No. Title Page No.

1.

**CHAPTER 1**

**INTRODUCTION**

* 1. **Introduction**

As we all know, each and every organization requires a proper attendance system. The maintenance of the attendance system is done with traditional method that is pen and paper approach. But the pen and paper approach comes with various limitations such as manipulation, time consuming and prone to errors.

With this project, we are trying to cover the flaws that is in the traditional approach. We intend to bring an Attendance System which is done via **Face Recognition** method. The Face Recognition process is more reliable and is less prone to errors. This process solves the manipulative problem of the pen and paper method. The **Attendance** **System** **Using Face Recognition** which opens the camera which scans the face and stores the attendance record of the student.

* 1. **Applications**

The purpose of this new attendance system is to create a better attendance system than the one that is currently in use. The application of this new attendance system can be seen in the various areas. The following areas are given below -

1. Educational Organisation
2. Business Organisation
3. Government Departments
4. Banks

And various others.

* 1. **Overview of Report**

First, we register the name and registration number of the students and then we capture the faces of the student using haarcascade we capture the faces if their eyes are inside the face that was detected. We take around 50 to 100 pictures of the face from the same person in different angles after taking the name, registration number and picture we make the those faces in their numeric value and we make machine learn those value and names of the student. To make the machine learning easier we train our dataset into an XML file. After training the dataset and saving in XML file we take that file and open up the program in which we take the live video from the webcam of the PC and then predict the values that are stored in xml format and if the values match then it will display the name and registration number of the student. If the name of the student is displayed and the value is true the names will be automatically save in the temporary text file, after closing the program it runs another program without clicking a button which encrypts the names of the student who are present into base 64. After the file is encrypted, we send the encrypted file into admin system which will consist the program to decrypt those names. After the names are decrypted, names will all be automatically segregated into their classes and year. After the names are segregated, only when the registered timing of the classes is correct then students will be marked present, if the attendance is done late then system will show error or if the attendance is trying to be input in lunch time or short break or after the school hours the attendance will not be accepted by the system. Every hour attendance is registered in every hour and particular date and there will be a program to combine all those attendance in one excel file for that a person is needed because every new day date is updated.

* 1. **Concepts**

**1.4.1 Image Processing**

Digital Image Processing means processing digital image by means of digital computer. We can also say that it is a use of computer algorithms, in order to get enhanced image either to extract some useful information.

Image processing mainly include the following steps:

1. Importing the image via image acquisition tools.
2. Analyzing and manipulating the image.
3. Output in which result can be altered image or a report which is based on analyzing that image.

What is an Image?

An image is defined as a two-dimensional function F (x, y)

Where x and y are spatial coordinates and the amplitude of F at any pair of coordinates (x, y) is called the intensity of that image at that point. When x, y and amplitude values of F are finite, we call it a digital image.

**1.4.2 Face Recognition**

Face Recognition is an Al-based computer technology that can identify and locate the presence of human faces in digital photos and videos.

**CHAPTER 2**

**LITERATURE SURVEY**

In this project, we have gone for the literature survey and took some paper for the reference.

* 1. **A Review on Face Recognition Attendance System**

Research Inventy: International Journal of Engineering and Science.

* 1. **Web Based Intelligence Surveillance System**

Research Inventy: International Journal of Computer Applications

**CHAPTER 3**

**PROBLEM STATEMENTS AND OBJECTIVES**

**3.1 Problem Identification**

* Time Consuming while transferring large amount of data from hard copies to soft copies.
* Prone to Errors
* Physical Absence
* Data Manipulation
* Manual Workload

**3.2 Objectives­**

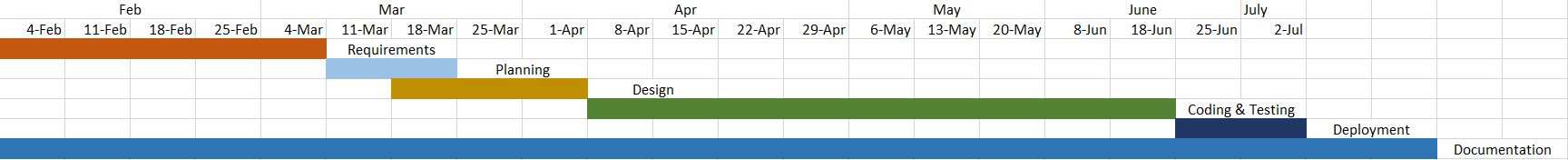
* Reduce Manual Work
* Time Efficiency
* Accuracy
* Assured Physical Presence

**3.3 Procedure**

* Group discussion was conducted.
* The domain & topic was finalized.
* For Image Capturing Webcam was used.
* For Codes: Python, XML

Software Packages: cv2, numpy, etc.

* For storing the data excel sheet & folders used.
* The project was concluded thereafter.
  1. **Gantt Chart**



**Fig (1). Gantt chart**

**CHAPTER 4**

**THEORITICAL ANALYSIS / PROJECT DETAILS**

**4.1 Methodology**

* The face of the individual(s) is read or scanned by the webcam.
* Image is captured through the webcam and the face is recognised.
* The attendance is stored after the face is read.
* The details are also stored in an excel format.

**4.2 Modules**

Encrypt

Names

Capturing

Faces

Convert into excel with date and hour

Training captured faces

Decrypt

Names

Distinguishing classes

Recognizing faces

**Fig 2: Working of project**

**4.3 Image Capturing**

When the program is executed it opens 2 terminals , one opens the webcam which captures image on live video and in another terminal it shows the student names and whether or not the students have been marked present or not. It captures the face of a certain person and stores it into dataset using certain id for each images to identify.

**4.4 Storing the Attendance**

We use xlwt and xlrd to write in an excel document. Package datetime is used to capture current date and time. In this process, the names of the present student which was written in text document will be committed and saved in an excel document. It also captures the face of a certain person and stores it into a folder using certain id for each images to identify.

**4.5 Codes Implemented**

**Face Detection**

import cv2

import numpy as np

import NameFind

WHITE = [255, 255, 255]

face\_cascade = cv2.CascadeClassifier('har/haarcascade\_frontalface\_default.xml')

eye\_cascade = cv2.CascadeClassifier('har/haarcascade\_eye.xml')

cap = cv2.VideoCapture(0)

ID=NameFind.AddName()

Count = 0

while Count < 100:

ret, img = cap.read()

gray = cv2.cvtColor(img, cv2.COLOR\_BGR2GRAY)

faces = face\_cascade.detectMultiScale(gray, 1.3, 5)

for (x, y, w, h) in faces:

FaceImage = gray[y - int(h / 2): y + int(h \* 1.5), x - int(x / 2): x + int(w \* 1.5)]

Img = (NameFind.DetectEyes(FaceImage))

cv2.putText(gray, "FACE DETECTED", (x+(w//2), y-5), cv2.FONT\_HERSHEY\_DUPLEX, .4, WHITE)

if Img is not None:

frame = Img

else:

frame = gray[y: y+h, x: x+w]

cv2.imwrite("dataset/User." + str(ID) + "." + str(Count) + ".jpg",frame)

cv2.waitKey(300)

cv2.imshow("CAPTURED PHOTO", frame)

Count = Count + 1

cv2.imshow('Face Recognition System Capture Faces', gray)

if cv2.waitKey(1) & 0xFF == ord('q'):

break

cap.release()

cv2.destroyAllWindows()

**Training faces**

import os

import cv2

import numpy as np

from PIL import Image

LBPHFace = cv2.face.LBPHFaceRecognizer\_create(2, 2, 7, 7, 15)

path = 'dataset'

def getImageWithID (path):

imagePaths = [os.path.join(path, f) for f in os.listdir(path)]

FaceList = []

IDs = []

for imagePath in imagePaths:

faceImage = Image.open(imagePath).convert('L')

faceImage = faceImage.resize((110,110))

faceNP = np.array(faceImage, 'uint8')

ID = int(os.path.split(imagePath)[-1].split('.')[1])

FaceList.append(faceNP)

IDs.append(ID)

cv2.imshow('Training Set', faceNP)

cv2.waitKey(1)

return np.array(IDs), FaceList

IDs, FaceList = getImageWithID(path)

LBPHFace.train(FaceList, IDs)

print('LBPH FACE RECOGNISER COMPLETE...')

LBPHFace.write('trainer/train.xml')

#LBPHFace.write('trainer/trainer.yml')

print ('ALL XML FILES SAVED...')

cv2.destroyAllWindows()

**Face recognization**

import os

import cv2

import numpy as np

import NameFind

face\_cascade = cv2.CascadeClassifier('har/haarcascade\_frontalface\_default.xml')

eye\_cascade = cv2.CascadeClassifier('har/haarcascade\_eye.xml')

recognise = cv2.face.LBPHFaceRecognizer\_create(2, 2, 7, 7, 15)

recognise.read('trainer/train.xml')

cap = cv2.VideoCapture(0)

# cap = cv2.VideoCapture('TestVid.wmv') # Video object

while True:

ret, img = cap.read()

gray = cv2.cvtColor(img, cv2.COLOR\_BGR2GRAY)

faces = face\_cascade.detectMultiScale(gray, 1.3, 5)

for (x, y, w, h) in faces:

# Eyes should be inside the face.

gray\_face = gray[y: y+h, x: x+w]

eyes = eye\_cascade.detectMultiScale(gray\_face)

for (ex, ey, ew, eh) in eyes:

ID, conf = recognise.predict(gray\_face)

NAME = NameFind.ID2Name(ID, conf)

NameFind.DispID(x, y, w, h, NAME, gray)

cv2.imshow('Face Recognition System', gray)

if cv2.waitKey(1) & 0xFF == ord('q'):

os.system('encoding.py')

break

cap.release()

cv2.destroyAllWindows()

**Encrypting**

import os

import base64

previousatt="text/encodedface.txt"

if os.path.isfile(previousatt):

os.remove(previousatt)

else:

print("Creating new attendance sheet")

fob=open('text/encodedface.txt','a')

f=open('detect.txt','r')

lines = f.read().split("\n")

print (lines)

names=[]

def enterData(z):

if z in names:

pass

else:

names.append(z)

z=str(z)

file=open("text/encodedface.txt","a")

file.write(z)

#file.write("\t")

file.write("\n")

file.close()

return names

for i in range(0,len(lines)):

data=lines[i].encode('utf-8')

name=str(base64.b64encode(data).decode())

print(name)

enterData(name)

f.close()

os.remove('detect.txt')

**Decrypting**

import base64

import os

f=open('text/encodedface.txt','r')

lines = f.read().split("\n")

maindata=[]

for i in range(0,len(lines)):

maindata.append(base64.b64decode(lines[i]))

print(maindata)

fob=open('text/DecodedResults.txt','w+')

for i in range(0,len(maindata)):

fob.write(str(maindata[i]) + '\n')

fob.close()

os.system('comparingclasses.py')

**Namefind**

import os

import cv2

import math

import time

now\_time = time.clock()

#face = cv2.CascadeClassifier('haarcascade\_frontalcatface.xml')

#glass\_cas = cv2.CascadeClassifier('haarcascade\_eye\_tree\_eyeglasses.xml')

face = cv2.CascadeClassifier(cv2.data.haarcascades + 'haarcascade\_frontalcatface.xml')

glass\_cas = cv2.CascadeClassifier(cv2.data.haarcascades + 'haarcascade\_eye\_tree\_eyeglasses.xml')

WHITE = [255, 255, 255]

def FileRead():

Info = open("Names.txt", "r")

NAME = []

while (True):

Line = Info.readline()

if Line == '':

break

NAME.append (Line.split(",")[1].rstrip())

return NAME

Names = FileRead()

previousatt="detect.txt"

if os.path.isfile(previousatt):

os.remove(previousatt)

else:

print("Creating new attendance sheet")

fob=open('detect.txt','a')

#f=open('detect.txt','r')

#lines = f.read().split("\n")

#print (lines)

names=[]

def enterData(z):

if z in names:

pass

else:

names.append(z)

z=str(z)

file=open("detect.txt","a")

file.write(z)

file.write("\t")

file.write("\n")

file.close()

print(z)

return names

def ID2Name(ID, conf):

if ID > 0:

NameString = "Name: " + Names[ID-1]

print(Names)

#z = Names

enterData(Names)

else:

NameString = " Face Not Recognised "

return NameString

def AddName():

Name = input('Enter Your Name ')

Info = open("Names.txt", "r+")

ID = ((sum(1 for line in Info))+1)

Info.write(str(ID) + "," + Name + "\n")

print ("Name Stored in " + str(ID))

Info.close()

return ID

def DispID(x, y, w, h, NAME, Image):

#def DispID(x, y, w, h):

#

Name\_y\_pos = y - 10

Name\_X\_pos = x + w//2 - (len(NAME)\*7//2)

if Name\_X\_pos < 0:

Name\_X\_pos = 0

elif (Name\_X\_pos +10 + (len(NAME) \* 7) > Image.shape[1]):

Name\_X\_pos= Name\_X\_pos - (Name\_X\_pos +10 + (len(NAME) \* 7) - (Image.shape[1]))

if Name\_y\_pos < 0:

Name\_y\_pos = Name\_y\_pos = y + h + 10

# ---------------------------

draw\_box(Image, x, y, w, h)

cv2.rectangle(Image, (Name\_X\_pos-10, Name\_y\_pos-25), (Name\_X\_pos +10 + (len(NAME) \* 7), Name\_y\_pos-1), (0,0,0), -2) # Draw a Black Rectangle over the face frame

cv2.rectangle(Image, (Name\_X\_pos-10, Name\_y\_pos-25), (Name\_X\_pos +10 + (len(NAME) \* 7), Name\_y\_pos-1), WHITE, 1)

cv2.putText(Image, NAME, (Name\_X\_pos, Name\_y\_pos - 10), cv2.FONT\_HERSHEY\_DUPLEX, .4, WHITE) # Print the name of the ID

def draw\_box(Image, x, y, w, h):

cv2.line(Image, (x, y), (x + (w//5) ,y), WHITE, 2)

cv2.line(Image, (x+((w//5)\*4), y), (x+w, y), WHITE, 2)

cv2.line(Image, (x, y), (x, y+(h//5)), WHITE, 2)

cv2.line(Image, (x+w, y), (x+w, y+(h//5)), WHITE, 2)

cv2.line(Image, (x, (y+(h//5\*4))), (x, y+h), WHITE, 2)

cv2.line(Image, (x, (y+h)), (x + (w//5) ,y+h), WHITE, 2)

cv2.line(Image, (x+((w//5)\*4), y+h), (x + w, y + h), WHITE, 2)

cv2.line(Image, (x+w, (y+(h//5\*4))), (x+w, y+h), WHITE, 2)

# -

def DispID2(x, y, w, h, NAME, Image):

Name\_y\_pos = y - 40

Name\_X\_pos = x + w/2 - (len(NAME)\*7/2)

if Name\_X\_pos < 0:

Name\_X\_pos = 0

elif (Name\_X\_pos +10 + (len(NAME) \* 7) > Image.shape[1]):

Name\_X\_pos= Name\_X\_pos - (Name\_X\_pos +10 + (len(NAME) \* 7) - (Image.shape[1]))

if Name\_y\_pos < 0:

Name\_y\_pos = Name\_y\_pos = y + h + 10

cv2.rectangle(Image, (Name\_X\_pos-10, Name\_y\_pos-25), (Name\_X\_pos +10 + (len(NAME) \* 7), Name\_y\_pos-1), (0,0,0), -2) # Draw a Black Rectangle over the face frame

cv2.rectangle(Image, (Name\_X\_pos-10, Name\_y\_pos-25), (Name\_X\_pos +10 + (len(NAME) \* 7), Name\_y\_pos-1), WHITE, 1)

cv2.putText(Image, NAME, (Name\_X\_pos, Name\_y\_pos - 10), cv2.FONT\_HERSHEY\_DUPLEX, .4, WHITE) # Print the name of the ID

def DispID3(x, y, w, h, NAME, Image):

Name\_y\_pos = y - 70

Name\_X\_pos = x + w//2 - (len(NAME)\*7//2)

if Name\_X\_pos < 0:

Name\_X\_pos = 0

elif (Name\_X\_pos +10 + (len(NAME) \* 7) > Image.shape[1]):

Name\_X\_pos= Name\_X\_pos - (Name\_X\_pos +10 + (len(NAME) \* 7) - (Image.shape[1]))

if Name\_y\_pos < 0:

Name\_y\_pos = Name\_y\_pos = y + h + 10

cv2.rectangle(Image, (Name\_X\_pos-10, Name\_y\_pos-25), (Name\_X\_pos +10 + (len(NAME) \* 7), Name\_y\_pos-1), (0,0,0), -2)

cv2.rectangle(Image, (Name\_X\_pos-10, Name\_y\_pos-25), (Name\_X\_pos +10 + (len(NAME) \* 7), Name\_y\_pos-1), WHITE, 1)

cv2.putText(Image, NAME, (Name\_X\_pos, Name\_y\_pos - 10), cv2.FONT\_HERSHEY\_DUPLEX, .4, WHITE)

def DrawBox(Image, x, y, w, h):

cv2.rectangle(Image, (x, y), (x + w, y + h), (255, 255, 255), 1)

def DetectEyes(Image):

Theta = 0

rows, cols = Image.shape

glass = glass\_cas.detectMultiScale(Image)

for (sx, sy, sw, sh) in glass:

if glass.shape[0] == 2:

if glass[1][0] > glass[0][0]:

DY = ((glass[1][1] + glass[1][3] / 2) - (glass[0][1] + glass[0][3] / 2))

DX = ((glass[1][0] + glass[1][2] / 2) - glass[0][0] + (glass[0][2] / 2))

else:

DY = (-(glass[1][1] + glass[1][3] / 2) + (glass[0][1] + glass[0][3] / 2))

DX = (-(glass[1][0] + glass[1][2] / 2) + glass[0][0] + (glass[0][2] / 2))

if (DX != 0.0) and (DY != 0.0):

Theta = math.degrees(math.atan(round(float(DY) / float(DX), 2)))

print ("Scanning.. " + str(Theta))

M = cv2.getRotationMatrix2D((cols / 2, rows / 2), Theta, 1)

Image = cv2.warpAffine(Image, M, (cols, rows))

# cv2.imshow('ROTATED', Image)

Face2 = face.detectMultiScale(Image, 1.3, 5)

for (FaceX, FaceY, FaceWidth, FaceHeight) in Face2:

CroppedFace = Image[FaceY: FaceY + FaceHeight, FaceX: FaceX + FaceWidth]

return CroppedFace

def tell\_time\_passed():

print ('TIME PASSED ' + str(round(((time.clock() - now\_time)/60), 2)) + ' MINS')

**Comparing classes**

import difflib

import os

previousatt="text/BscIT6thsempresent.txt"

if os.path.isfile(previousatt):

os.remove(previousatt)

file1 = ('text/BscIT6thsem.txt')

file2 = ('text/DecodedResults.txt')

file1\_lines = open(file1).read().split('\n')

file2\_lines = open(file2).read().split('\n')

file1\_lines\_set = set(file1\_lines)

file2\_lines\_set = set(file2\_lines)

file1\_added = file1\_lines\_set - file2\_lines\_set

file1\_removed = file2\_lines\_set - file1\_lines\_set

#----------------------------

names=[]

def enterData(z):

if z in names:

pass

else:

names.append(z)

z=str(z)

file=open("text/BSCIT6thsempresent.txt","a")

file.write(z)

file.write("\n")

file.close()

return names

#-------------------------------------------

for line in file1\_lines:

if line in file1\_added:

print('-', line.strip())

elif line in file1\_removed:

print ('+', line.strip())

else:

enterData(line)

for line in file2\_lines:

if line in file1\_added:

print('-',line.strip())

elif line in file1\_removed:

print('+',line.strip())

else:

enterData(line)

#os.system('2.py')

import difflib

import os

previousatt2="text/Bca6thpresent.txt"

if os.path.isfile(previousatt2):

os.remove(previousatt2)

file12 = ('text/Bca6thsem.txt')

file22 = ('text/DecodedResults.txt')

file12\_lines = open(file12).read().split('\n')

file22\_lines = open(file22).read().split('\n')

file12\_lines\_set = set(file12\_lines)

file22\_lines\_set = set(file22\_lines)

file12\_added = file12\_lines\_set - file22\_lines\_set

file12\_removed = file22\_lines\_set - file12\_lines\_set

#----------------------------

names=[]

def enterData(z):

if z in names:

pass

else:

names.append(z)

z=str(z)

file=open("text/Bca6thpresent.txt","a")

file.write(z)

file.write("\n")

file.close()

return names

#-------------------------------------------

for line in file12\_lines:

if line in file12\_added:

print('-', line.strip())

elif line in file12\_removed:

print ('+', line.strip())

else:

enterData(line)

for line in file22\_lines:

if line in file12\_added:

print('-',line.strip())

elif line in file12\_removed:

print('+',line.strip())

else:

enterData(line)

**Converting into excel**

import xlwt

import xlrd

from datetime import datetime

import time

import datetime

import os

#-------------------------------------

dt\_date = datetime.datetime.now()

#print ("The Current date is:" ,dt\_date)

#print("In specified format:", dt\_date.strftime("%A, %d %b %Y,\t %T"))

print("time",dt\_date.strftime("%T"))

#----------------------------------------

date = xlwt.easyxf(num\_format\_str='D-MMM-YY')

time = xlwt.easyxf(num\_format\_str='H:M')

#td = time.strftime("%d-%m-%Y")

a = input ("enter the file name with .txt")

name=input("Enter the name of the class and semester")

book = xlwt.Workbook()

ws = book.add\_sheet('First Sheet')

f = open('text/'+a, 'r+')

data = f.readlines()

#ws.write(0, 0, datetime.now(), date)

check = dt\_date

#---------------------------------------------

if check.hour == 9 and check.minute <= 30:

h = ('First hour')

elif check.hour == 10 and check.minute >= 20:

h = ('Second hour')

elif check.hour == 11 and check.minute >= 20:

h = ('Third hour')

elif check.hour == 12 and check.minute >= 10:

h = ('Fourth hour')

elif check.hour == 14 and check.minute >= 00:

h = ('Fifth hour')

elif check.hour == 15 and check.minute <= 40:

h = ('Sixth hour')

else:

print("Error in timings")

h = ('wrong time')

print(h)

#-------------------------------------

for i in range(len(data)):

row = data[i].split()

ws.write(i,3,"Present")

ws.write(i,4,dt\_date,date)

#ws.write(i,4,dt\_date)

ws.write(i,5,dt\_date,time)

ws.write(i,6,h)

for j in range(len(row)):

ws.write(i, j, row[j])

#ws.write(i,j+1,datetime.now(),date)

book.save(os.path.join('attendence',(name+" "+dt\_date.strftime("%A, %d %b %Y")+" "+h+'.xls')))

#book.save(name+" "+check+'.xls')

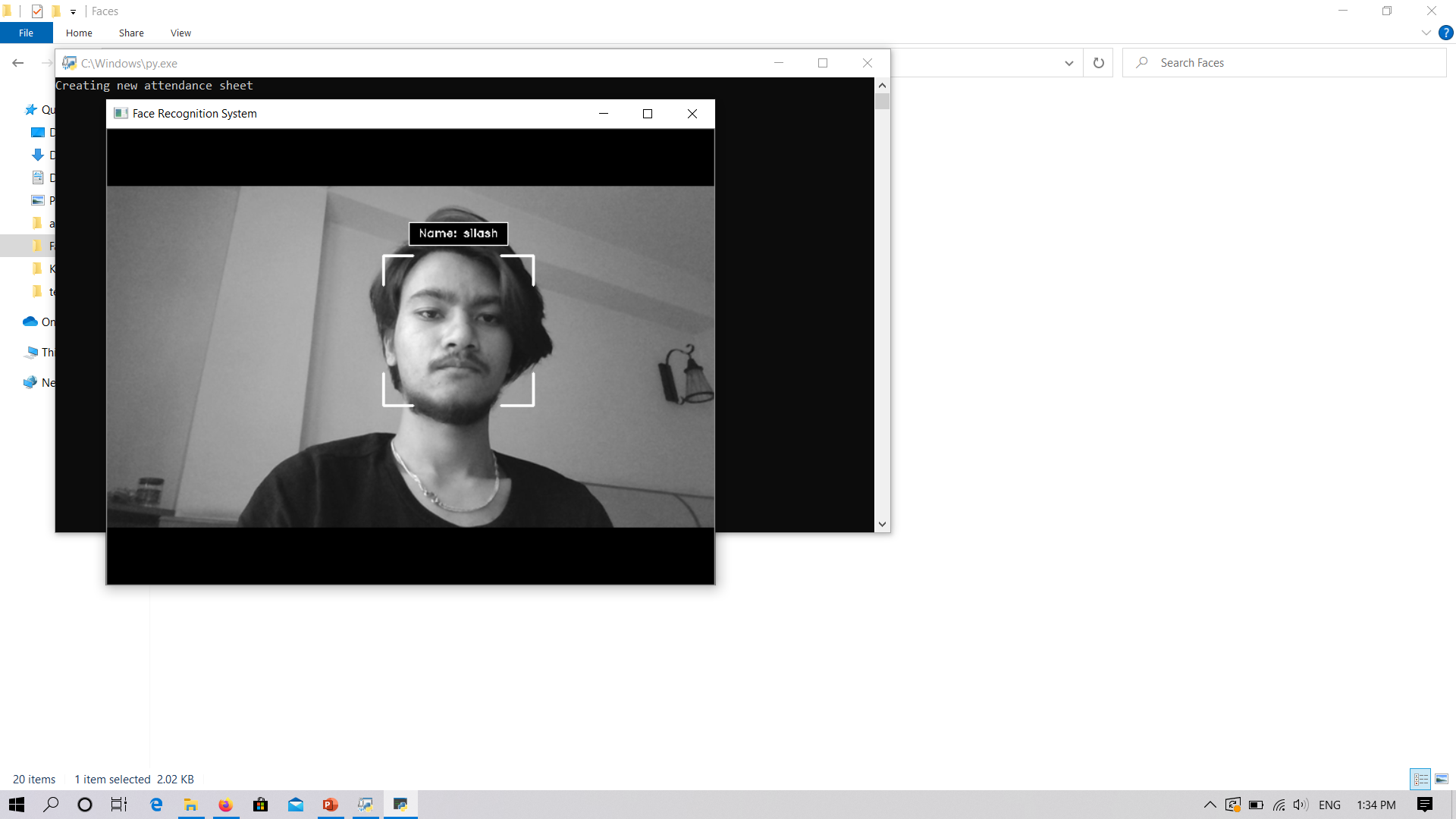
print("File "+name+" has been saved")

f.close()

**CHAPTER 5**

**RESULTS AND DISCUSSION**

**5.1 Results**



A screenshot of a social media post

Description automatically generated

A screenshot of a computer screen

Description automatically generated**A screenshot of a cell phone

Description automatically generated**

A screenshot of a cell phone

Description automatically generated

**CHAPTER 6**

**CONCLUSION**

**6.1 Conclusion**

An attendance monitoring system is very important in any type of organisation. A face recognizing smart attendance system is really efficient, effective, provides the accurate results. The face recognition process possesses a really great advantage over the traditional method in terms of ensuring the physical presence.

In this project report, we are able to create a better attendance system than the existing. However, the complexity might increase a bit but regardless, the results are more positive and accurate and there is no manipulation of data.

**6.2 Future Scopes**

A further extension to this project research is capable of providing improved and productive results. The further work on this project can be done by adding more dynamicity into it.

We can link the attendance system with the ERP system and also increase the effectiveness of the automatic face recognition techniques.

**REFERENCES**

**[1] Akshada Deshmukh, Harsalata Wadaskar, Leena Zade, Neha Dhekate and Preetee Karmore “*Webcam Based Intelligence Surveillance System.*” Research Inventy: International Journal of Engineering and Science. (2013)**

**[2] Shubhi Shriwastav, Dinesh Chandra Jain “*A Review on Face Recognition Attendance System*.” Research Inventy: International Journal of Computer Applications. (2016)**

**Internet sources:**

**[1]** [**https://github.com/ShubhamSonkesriya/Face-Recognition-Attendance-System**](https://github.com/ShubhamSonkesriya/Face-Recognition-Attendance-System)

**ACCESSED DATE: 15th MARCH, 2020**

**[2]** [**https://ijeat.org/wpcontent/uploads/papers/v8i35/C11230283519.pdf**](https://ijeat.org/wpcontent/uploads/papers/v8i35/C11230283519.pdf)

**ACCESSED DATE: 21st MARCH, 2020**