# VISVESVARAYA TECHNOLOGICAL UNIVERSITY

JNANA SANGAMA, BELAGAVI -590 014

**Mini Project Report on**

# FACE RECOGNITION AND ATTENDANCE SYSTEM

**Submitted by**


| | |
|---|---|
| **ANU H A** | **1AY19EC018** |
| **ASHMITHA SHETTY** | **1AY19EC022** |
| **ASHMITHA G** | **1AY19EC023** |
| **SANGEETHA N R** | **1AY19EC073** |


**Under the Guidance of**


**Mr. Lakshmikanth Reddy**
**Assistant Professor**
**Department of Electronics and Communication Engineering**


ACHARYA

**2021-2022**


**DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING**

**Acharya Institute of Technology**

**Acharya Dr. Sarvepalli Radhadrishnan Road, Soladevanahalli, Bengaluru-560107**

# ACHARYA INSTITUTE OF TECHNOLOGY

**Hesaraghatta Road, Dr. Sarvepalli Radhakrishnan Road**
**Bengaluru 560107**
**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING.**

## CERTIFICATE

Certified that the Mini Project work entitled **"FACE RECOGNITION AND ATTENDANCE SYSTEM"** is carried out by **ANU H A(1AY19EC018), ASHMITHA SHETTY (1AY19EC022), ASHMITHA G (1AY19EC023) and SANGEETHA N R (1AY19EC073)** in the partial fulfillment for the award of the degree of Bachelor of Engineering in Electronics and Communication Engineering of Visvesvaraya Technological University, Belagavi during the year **2021-2022**. It is certified that all corrections/suggestions indicated for the assessment have been incorporated in the report deposited in the departmental library. The Mini Project Report has been approved as it satisfies the academic requirement in respect of Mini Project work **(18ECMP68)** prescribedfor the Bachelor of Engineering Degree.

| | | |
|---|---|---|
| ..... | ..... | ..... |
| Signature of Guide | Signature of HOD | Signature of Principal |
| Mr.Lakshmikanth Reddy | Dr. Rajeswari | Dr. Rajath Hegde M M |
| Assistant Professor | Prof and Head of the Dept | Principal, AIT |
| Dept of ECE, AIT | Dept of ECE, AIT | |

**External Viva**

| **Name of the Examiners** | **Signature with Date** |
|---|---|
| **1. .....** | **.....** |
| **2. .....** | **.....** |

# ACHARYA INSTITUTE OF TECHNOLOGY

**Hesaraghatta Road, Dr. Sarvepalli Radhakrishnan Road**
**Bengaluru 560107**
**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING.**


2021-2022

## DECLARATION

We, ANU H A [1AY19EC018], ASHMITHA ARUN SHETTY[1AY19EC022], ASHMITHA G[1AY19EC023], SANGEETHA N R[1AY19EC073] hereby declare that the Mini project work entitled **"FACE RECOGNITION AND ATTENDANCE SYSTEM"** has been independently carried out by us under the supervision of **Mr.Lakshmikanth Reddy , Assistant Professor**, Department of Electronics and Communication Engineering, Acharya Institute of Technology in partial fulfillment of the requirement for the award of the degree of **Bachelor of Engineering** in **Electronics and Communication Engineering** by **Visvesvaraya Technological University, Belagavi** during the year **2021-22.**

**Place:** Bangalore

**Date:**

| | |
|---|---|
| ANU H A | 1AY19EC018 |
| ASHMITHA SHETTY | 1AY19EC022 |
| ASHMITHA G | 1AY19EC023 |
| SANGEETHA N R | 1AY19EC073 |

# ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of a task would be incomplete without the mention of the people who made it possible and without their constant guidance and encouragement success would not have been possible.

I am grateful to the institute **Acharya Institute of Technology** and management for its ideas and inspiration for having provided us with the good infrastructure, laboratory, facilities, and inspiring staff which has made this Mini Project work successful.

I would like to express my sincere gratitude to **Dr. Rajath Hegde M M, Principal, AIT and Prof. Mari Gowda C K, Vice Principal, AIT** for all the facilities that has been extended by them throughout my work.

I heartily thank and express my sincere gratitude to **Dr. Rajeswari, Professor and HOD, Dept. of ECE, AIT** for her valuable support and a constant source of enthusiastic inspiration to steer us forward.

I would like to express my sincere gratitude to the Internal Guide **Mr.Lakshmikanth Reddy, Assistant Professor, Dept. of ECE, AIT** for his/her invaluable guidance and support.

I would like to express my sincere gratitude to the Mini Project Coordinators **Mrs. Pranita Niraj Palsapure, Assistant Professor and Mr. Sandeep kumar K, Assistant Professor, Dept. of ECE, AIT** for their valuable guidance and support.

Finally, I would like to express my sincere gratitude to my parents, all teaching and non-teaching faculty members, and friends for their moral support, encouragement, and help throughout the completion of the Mini Project work.

| | |
|---|---|
| **ANU H A** | **1AY19EC018** |
| **ASHMITHA SHETTY** | **1AY19EC022** |
| **ASHMITHA G** | **1AY19EC023** |
| **SANGEETHA N R** | **1AY19EC073** |

# ABSTRACT

In the fastmoving modern world everything is changed to provide a better life for humanity. New developments make this as a reality. So, we were decided to develop the Real Time criminal detection and face recognition system. The importance of the automatic face detection and tracking system has increased as it is needed for video surveillance and new user interfaces. And providing higher security to the country. This system suitable for all people, but some people's face damaged on acid attack, war and etc. so we were used to password for these peoples. In this research our effort is to develop a system for face detection and recognition in the real time. That will be efficient and give solution for many problems. The growing interest in computer vision of the past decade. Fueled by the steady doubling rate of computing power every 13 months, face detection and recognition has transcended from an esoteric to a popular area of research in computer vision and one of the better and successful applications of image analysis and Algorithm based understanding. Because of the intrinsic nature of the problem, computer vision is not only a computer science area of research, but also the object of neuro-scientific and psychological studies, mainly because of the general opinion that advances in computer image processing and understanding research will provide insights into how our brain work and vice versa. Because of general curiosity and interest in the matter, the author has proposed to create an application that would allow user access to a particular machine based on an in-depth analysis of a person's facial features. This application will be developed using PyCharm Open Source computer vision project.

# TABLE OF CONTENTS

# TABLE OF FIGURES:

# CHAPTER 1

## 1.1 INTRODUCTION:

Face recognition is the technique in which the identity of a human being can be identified usingOnes individual face can be identified using OPENCV with high accuracy. We will create Attendance project that will use a webcam to detect a faces and record the attendance live in a excel sheet. Such kind of systems can be used in photos, videos, or in real time machines. The objective of this article is to provide a simpler and easy method in machine technology. With the help of such a technology one can easily detect the face by the help of dataset in similar matching appearance of a person.

The method in which with the help of python and Open CV in deep learning is the most efficient way to detect the face of the person. This method is useful in many fields such as the military, for security, schools, colleges and universities, airlines, banking, online web applications, gaming etc. this system uses powerful python algorithm through which the detection and recognition of face is very easyand efficient.

The growing interest in computer vision of the past decade by the steady doubling rate of computingpower every 13 months, face detection and recognition has transcended from an esoteric to a popular area of research in computer vision and one of the better and successful applications of image analysis and algorithm based understanding. Because of the intrinsic nature of the problem, computer vision is not only a computer science area of research, but also the object of neuro-scientific and psychological studies, mainly because of the general opinion that advances in computer image processing and understanding research will provide insights into how our brain work and vice versa. Because of generalcuriosity and interest in the matter, the author has proposed to create an application that would allow user access to a particular machine based on an in-depth analysis of a person's facial features. This application will be developed using PyCharm computer vision .OpenCV libraries.

Face recognition is a part of biometric identification that extracts the facial features of a face, and then stores it as a unique face print to uniquely recognize a person. Biometric face recognition technology has gained the attention of many researchers because of its wide application. Face recognition technology is better than other biometric based recognition techniques like finger-print, palm-print, iris because of its non-contact process. Facebook uses the facial recognition technique for automating the process of tagging people.

## 1.2 HISTORY:

The implementation of facial recognition has seen many iterations, which saw roots in the 1960s when facial recognition was manually implemented by Woodrow Wilson Bledsoe. Bledsoe is largely considered the father of facial recognition for developing a system that classified photos of faces through a RAND tablet , which was a graphical computer input device. With this device, Bledsoe manually recorded the coordinate locations of facial features such as a person's mouth, nose, eyes.. Facial recognition was incrementally refined throughout the 1970s by the likes of Goldstein, Harmon, but largely remained a manually computed process. In what was to become known as the Eigenface approach, Kirby implemented an approach that started as a low-dimensional representation of facial images. Through this, they demonstrated that an analysis of features on a set of images could form a set of basic features. They also established that less than a hundred values were needed to accurately code a normalized image of a face.

## 1.3 MOTIVATION:

The main intention of this project is to solve the issues encountered in the manual attendance system while reproducing a brand new innovative smart system that can provide convenience to the institution. In this project, an application will be developed which is capable of recognising the identity of each individuals and eventually record down the data into a database system. Apart from that, an excel sheet is created which shows the students attendance with time.

## 1.4 OBJECTIVE:

- Save Images to Database
- Detect Faces
- Match Detected Faces to Database
- Recognize Faces
- Provides accurate information about them
- Record the attendance live in a excel sheet.

# CHAPTER 2:
# LITERATURE REVIEW:

Krishna Mridha, Nabhan Tawjih Yousef[1]"Study and Analysis of Implementing a Smart Attendance Management System Based on Face Recognition Technique usingOpenCV and Machine Learning" We can make our Attendance Management System(AMS) intelligent by using a face-to-face recognition strategy. For that, we have to fixa CCTV camera in the classroom at any best point, which makes a person's picture ata fixed time and tests a face-to-face image. Traditionally, student attendance at the institutes is manually reported on the attendance sheets. It's not a productive operation, because it takes 5 or more minutes for attendance.

R. Chellappa, C. Wilson, and S. Sirohey. "Human and machine recognition of faces: A survey". IEEE, 83:705–740, 1995[2] A face image can be represented by a point in a feature space such asspanned by a number of eigenfaces. In methods based on nearest neighbor classification, the representational capacity of face database depends on how prototypical face images are chosen to account for possible image variations and also how many prototypical images or their featurepoints are available

Xiaojun Bai, Feihu Jiang,Tianyi Shi, Yuang Wu "Design of Attendance System Basedon Face Recognition and Android Platform"[3] Aiming at the disadvantages of traditional manual attendance, this paper proposes a face recognition based attendancescheme. Through mobile platforms and face recognition technology to optimize the manual attendance process. This designdivides into the face recognition system of check on work attendance information input, attendancesign-in and attendance recordthree function module, and introduces a principle of face detection and classification,analyzes the process of the construction of the face recognition classifier, the last on

the Android platform design and implement a face recognition system of check on work attendance, by comparing the experiment results of face recognition accuracy, verify the feasibilityof this scheme.

Jaehoon Paul Jeong; Minho Kim; Yeonghyeon Lee; Patrick Lingga "IoT-Based Automatic Attendance System with Photo Face Recognition in Smart Campus"[4] The existing attendance checking system uses student identification (ID) cards or mobile applications. If the attendance is processed by an automatic attendance checking system,the attendance checking process can be more convenient for both students and professorsthan manual attendance checking. For getting these advantages, this paper proposes an IoT-based Automatic Attendance System (IAAS). IAAS is an attendance checking system using face recognition technology. The image data of students are collected by acapturing device (e.g., smartphone and tablet PC) and processed by the face recognitionsystem. This system checks who are attending the class and send their attendance to an attendance database system through an email notification.

Shubhobrata Bhattacharya; Gowtham Sandeep Nainala; Prosenjit Das; Aurobinda Routray "SmartAttendance Monitoring System (SAMS): A Face Recognition Based Attendance System for Classroom Environment"[5] In present academic system, regular class attendance of students' playsa significant role in performance assessment and quality monitoring. The conventional methods practiced in most of the institutions are bycalling names or signing on papers, which is highly time-consuming and insecure. This article presents the automatic attendance management system for convenience or data reliability. The system is developed by the integration of ubiquitous components to makea portable device for managing the students' attendance using Face Recognition technology.

# CHAPTER 3

# Tools and Technology

## 3.1 Python

Python being a very powerful programming languages and one of the programming languages that are being used all over the world has proven to give best results in the face recognition and detection systems. Together face recognition and detection becomes very easy and fruitful with the help of the python programming language and OpenCV.

The Python interpreter and the extensive standard library are freely available in source or binary for all major platforms from the Python Web site, https://www.python.org/, andmay be freely distributed. The same site also contains distributions of and pointers to many free third-party Python modules, programs and tools and additional documentation.

The Python interpreter is easily extended with new functions and data types implemented in C or C++ (or other languages callable from C). Python is also suitable as an extension language for customizable applications.

## 3.2 PyCharm

PyCharm is the most popular IDE used for Python scripting language.

This chapterwill give you an introduction to PyCharm and explains its features.

PyCharm offers some of the best features to its users and developers in the following aspects:

- Code completion and inspection
- Advanced debugging
- Support for web programming and frameworks such as Django and Flask

### Features of PyCharm:

Besides, a developer will find PyCharm comfortable to work with because of the features mentioned below:

#### Code Completion

PyCharm enables smoother code completion whether it is for built in or for an external package.

#### SQLAlchemy as Debugger

You can set a breakpoint, pause in the debugger and can see the SQL representation of theuser expression for SQL Language code.

## Git Visualization in Editor

When coding in Python, queries are normal for a developer. You can check the last commit easily in PyCharm as it has the blue sections that can define the difference between the last commit and the current one.

## Code Coverage in Editor

You can run.py files outside PyCharm Editor as well marking it as code coverage details elsewhere in the project tree, in the summary section etc.

## Package Management

All the installed packages are displayed with proper visual representation. This includes list of installed packages and the ability to search and add new packages.

## Local History

Local History is always keeping track of the changes in a way that complements like Git. Local history in PyCharm gives complete details of what is needed to rollback and what is tobe added.

## Refactoring

Refactoring is the process of renaming one or more files at a time and PyCharm includes various shortcuts for a smooth refactoring process.

## 3.3 OpenCV

OpenCV was started at Intel in 1999 by Gary Bradski for the purposes of accelerating research in and commercial applications of computer vision in the world and, for Intel, creating a demand for ever more powerful computers by such applications. Vadim Pisarevskyjoined Gary to manage Intel's Russian software OpenCV team. Over time the OpenCV teammoved on to other companies and other Research. Several of the original team eventually ended up working in robotics and found their way to Willow Garage. In 2008, Willow Garagesaw the need to rapidly advance robotic perception capabilities in an open way that leveragesthe entire research and commercial community and began actively supporting OpenCV, withGary and Vadim once again leading the effort. Intel's open-source computer-vision library can greatly simplify computer vision programming. It includes advanced capabilities - face detection, face tracking, face recognition, Kalman filtering, and a variety of artificial intelligence (AI) methods - in ready-to use form. In addition, it provides many basiccomputer-vision algorithms via its lower-level APIs.


The OpenCV has the ability to perform on multi-level framework. OpenCV has a number of significant abilities as well as utilities which appears from the outset. The OpenCV helps in recognizing the frontal face of the person and also creates XML documents for several areas such as the parts of the body. OpenCV has the advantage of being a multi- platform framework; it supports both Windows and Linux, and more recently, Mac OS X. OpenCV has so many capabilities it can seem overwhelming at first. A good understanding

of how these methods work is the key to getting good results when using OpenCV. Fortunately, only a select few need to be known beforehand to get started.

OpenCV is a cross-platform library using which we can develop real-time computer vision applications. It mainly focuses on image processing, video capture and analysis including features like face detection and object detection. Let's start the chapter by defining the term "Computer Vision". Computer Vision Computer Vision can be defined as a discipline that explains how to reconstruct, interrupt, and understand a 3D scene from its 2D images, in terms of the properties of the structure present in the scene. It deals with modelling and replicating human vision using computer software and hardware.

Computer Vision overlaps significantly with the following fields:

- Image Processing: It focuses on image manipulation.
- Pattern Recognition: It explains various techniques to classify patterns.
- Photogrammetry: It is concerned with obtaining accurate measurements from images.

## Features of OpenCV Library

Using OpenCV library, you can –

1. Read and write images
2. Capture and save videos
3. Process images (filter, transform)
4. Perform feature detection
5. Detect specific objects such as faces, eyes, cars, in the videos or images.
6. in it. Analyse the video, i.e., estimate the motion in it, subtract the background, and track objects

OpenCV was originally developed in C++. In addition to it, Python and Java bindings were provided. OpenCV runs on various Operating Systems such as windows, Linux, OSx, FreeBSD, Net BSD, Open BSD, etc.

## 3.4 Dlib

Dlib is a modern C++ toolkit containing machine learning algorithms and tools for creating complex software in C++ to solve real world problems. It is used in both industry and academia in a wide range of domains including robotics, embedded devices, mobile phones, and large high performance computing environments. Dlib's open source licensing allows you to use it in any application, free of charge.


Major Features:

### Documentation

Unlike a lot of open-source projects, this one provides complete and precise documentation for every class and function. There are also debugging modes that check the documented preconditions for functions. When this is enabled, it will catch the vast majority of bugs caused by calling functions incorrectly or using objects in an incorrect manner.

Lots of example programs are provided

I consider the documentation to be the most important part of the library. So if youfind anything that isn't documented, isn't clear, or has out of date documentation,tell me and I will fix it.

### High Quality Portable Code

Good unit test coverage. The ratio of unit test lines of code to library lines of codeis about 1 to 4.

The library is tested regularly on MS Windows, Linux, and Mac OS X systems. However, it should work on any POSIX system and has been used on Solaris, HPUX, and the BSDs.

No other packages are required to use the library. Only APIs that are provided by an out of the box OS are needed.

There is no installation or configure step needed before you can use the library. Seethe 'How to compile' page for details.

All operating system specific code is isolated inside the OS abstraction layers which are kept as small as possible. The rest of the library is either layered on top of the OS abstraction layers or is pure ISO standard C++.

## 3.5 Face recognition Library

Deep learning evolved lately in the process of the recognition systems. Hence deep learning along with the face recognition together work as the deep metric learning systems. In short deep learning in face detection and recognition will broadly work on two areas the first one being accepting the solidary input image or any other relevant picture and the second being giving the best outputs or the results of the image of the picture. We would be using dlib facial recognition framework that would be the easy way to organize the face evaluation. The two main significant libraries used in the system are dlib and face_recognition.

Recognize and manipulate faces from Python or from the command line with the world's simplest face recognition library. Built using dlib's state-of-the-art face recognition built with deep learning.

The model has an accuracy of 99.38% on the labelled faces in the Wild benchmark. This also provides a simple face_recognition command line tool that lets you do face recognition on a folder of images from the command line!

## 3.6 Python Modules-

Pillow, NumPy, click, colorama, pip, setuptools.

### 1. Pillow:

Python Imaging Library (abbreviated as PIL) is a free library for the Python programming language that adds support for opening, manipulating, and saving manydifferent image fileformats.

### 2. NumPy:

NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematicalfunctions to operate on these arrays.

## 3. Click:

Click is a Python package for creating beautiful command line interfaces in a composableway with as little code as necessary. It's the “Command Line InterfaceCreation Kit”. It's highly configurable but comes with sensible defaults out of thebox.

## 4. Colorama:

Colorama module is a Cross-platform printing of colored text can then be done using Colorama's constant shorthand for ANSI escape sequences:

Example 1: Pythonprogram to print red text with green background.

## 5.PiP:

The standard package manager for python is pip. It allows you to install and manage packages that aren't part them

## 6.Setuptools:

Setuptools is a package development process library designed to facilitate packaging Python projects by enhancing the Python standard library distutils (distribution utilities). It includes: Python package and module definitions. Distribution package metadata.

## 3.7 BLOCK DIAGRAM:

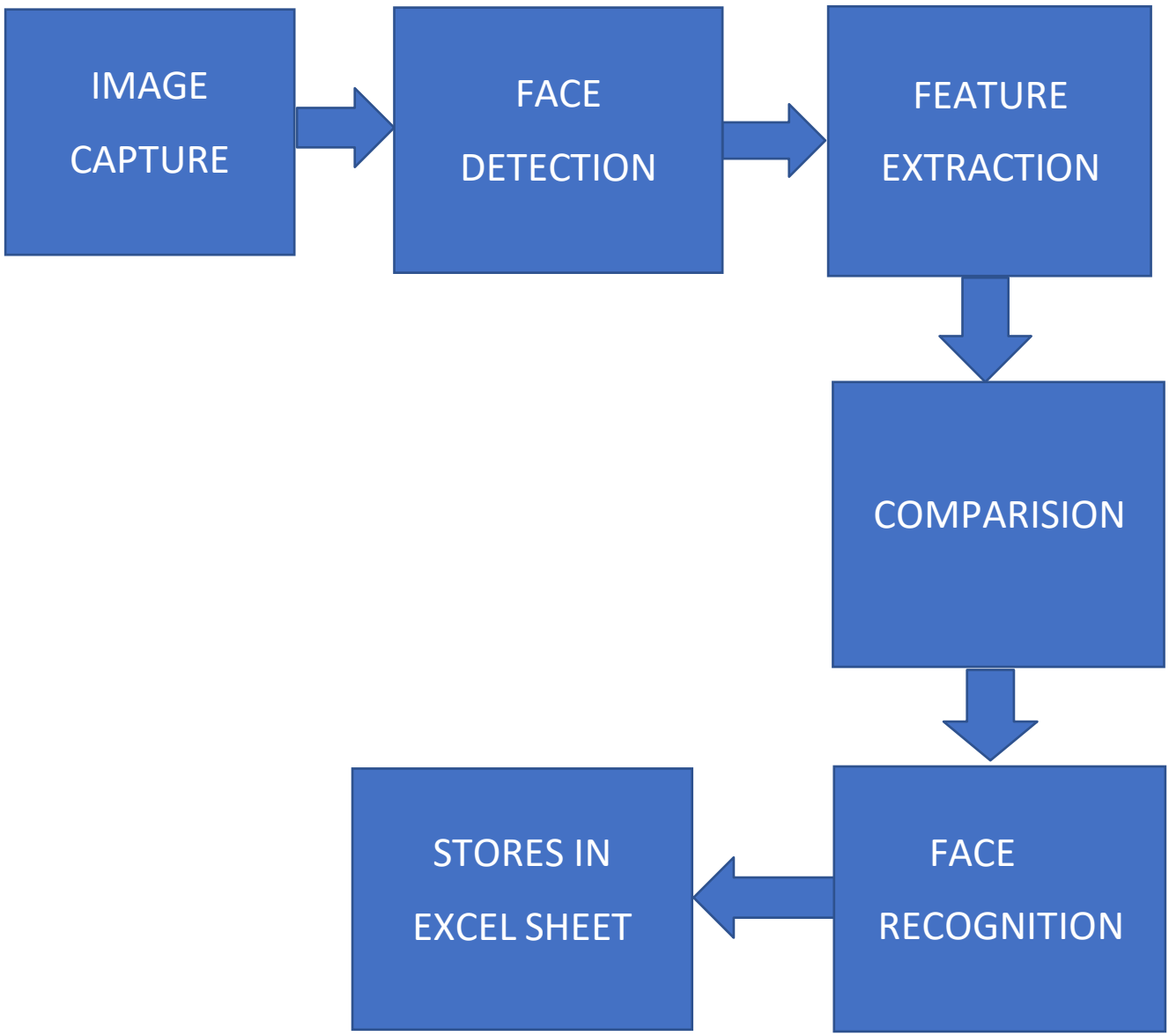



FIGURE 5: Block Diagram

# CHAPTER 4

## METHODOLOGY:

Importing

First, we will import the relevant libraries:

```
import face_recognition
import cv2
import numpy as np
```

Step 1:

### Loading Images and Converting to RGB

The Face Recognition package consists of a load image function that loads the image.Once imported the image has to be converted to RGB.

FIGURE 1: Sample image

Step 2:

## Find Faces Locations and Encodings

In the second step we will use the true functionality of the face recognition library. First, we will find the faces in our images. This is done using HOG (Histogram of Oriented Gradients) at the backend. Once we have the face they are warped to removeunwanted rotations. Then the image is fed to a pretrained neural network that out puts128 measurements that are unique to that particular face.

The parts that the model measures is not known as this is what the model learns by itself when it was trained. Once we have the face locationsand the encodings we candraw rectangles around our faces.

FIGURE 2: Detecting the face

Step 3:

## Compare Faces and Find Distance

Once we have the encodings for both faces, then we can compare these 128 measurements of these two faces to find similarities. To compare the package uses one of the most common machine Learning methods linearSVM Once we have the encodings for both faces, then we can compare these 128 measurements of these two faces to classifier. We can use the compare faces function to find if the faces match. This function returns True or False. Similarly, we can use the face_distance function to find how likely is the faces matchin termsof numbers. This is helpful particularly when there are multiple faces to detect from.

If we run this with the test image, we get the value True, indicating that the face found is of

Elon Musk. The is distance between the faces is 0.44. The lower the distance the better the match .Let's try it with another testing image. This time we will test it with an image of Bill Gates. As we can see the result is False and the distance is much higher than before indicating a bad match.


FIGURE 3: Comparing the face with that stored in database.

## Attendance Project

Now using the methods we have seen above, we will develop an attendance system where the user is automatically logged when they are detected in the camera. We will store the name along with the time when they appeared.

### Importing Images

As we have imported before we can use the same face_recognition.load_image_file() function to import our images. But when we have multiple images, importing them individually can become messy. Therefore, we will write a script to import all images in a given folder at once. For this we will need the OS library so we will import that first. We will store all the images in one list and their names in another.

## Compute Encodings

Now that we have a list of images, we can iterate through those and create a corresponding encoded list for known faces. To do this we will create a function. As earlier we will first convert it into RGB and thenfind its encoding using the face encodings() function. Then we will append each encoding to our list.

## Webcam Image

First we will read the image from the webcam and then resize it to quarter the size. This is done to increasethe speed of the system. Even though the image being used is 1/4 th of the original, we will still use the original size while displaying. Next, we will convert it to RGB.

## Find Matches

Now we can match the current face encodings to our known faces encoding list to find the matches. We will also compute the distance. This is done to find the best match in case more than one face is detected at a time.

## Marking Attendance

Lastly, we are going to add the automated attendance code. We will start by writing a function that requiresonly one input which is the name of the user. First, we open our Attendance file which is in csv format. Then we read all the lines and iterate through each line using a for loop. Next, we can split using comma ‘,’. This will allow us to get the first element which is the name of the user. If the user in the camera alreadyhas an entry in the file, then nothing will happen. On the other hand, if the user is new then the name of the user along with the current time stamp will be stored. We can use the datetime class in the date time package to get the current time.

# FINAL OUTPUT SNAPSHOTS:

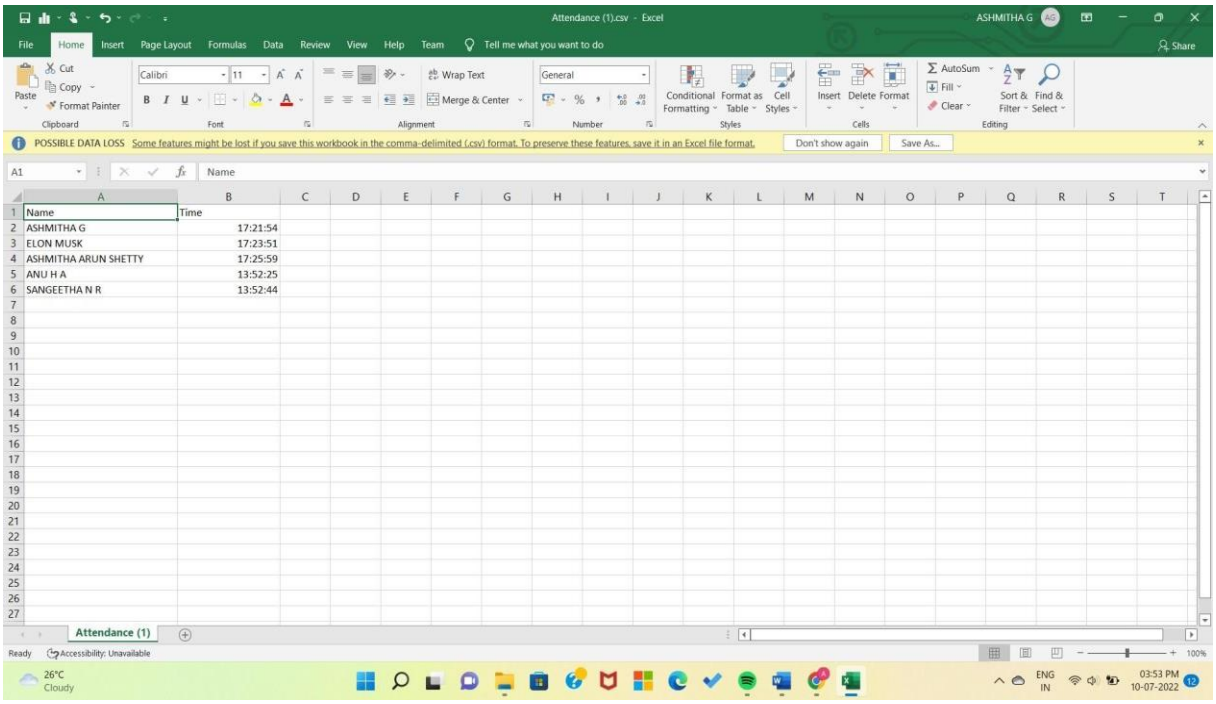



FIGURE 4: Sample Excel sheet

Here attendance is stored live in an excel sheet.

# 4.1 PROGRAMMING PART-1:

**Basic.py**

```
import cv2

import face_recognition

imgElonimport cv2= face_recognition.load_image_file('ImagesBasic/Elon

Musk.jpg')imgElon = cv2.cvtColor(imgElon,cv2.COLOR_BGR2RGB)

imgTest = face_recognition.load_image_file('ImagesBasic/Bill

gates.jpg')imgTest = cv2.cvtColor(imgTest,cv2.COLOR_BGR2RGB)


faceLoc = face_recognition.face_locations(imgElon)[0]
encodeElon=face_recognition.face_encodings(imgElon)[0]

cv2.rectangle(imgElon,(faceLoc[3],faceLoc[0]),(faceLoc[1],faceLoc[2]),(255,0,255),2)


faceLocTest = face_recognition.face_locations(imgTest)[0] encodeTest
=face_recognition.face_encodings(imgTest)[0]

cv2.rectangle(imgTest,(faceLocTest[3],faceLocTest[0]),(faceLocTest[1],faceLocTest[2]),(255,
0,255),2)


results = face_recognition.compare_faces([encodeElon],encodeTest)

faceDis = face_recognition.face_distance([encodeElon],encodeTest) print(results,faceDis)

cv2.putText(imgTest,f'{results}{round(faceDis[0],2)}',(50,50),cv2.FONT_HERSHEY_COMP
LEX,1,(0,0,255),2)

cv2.imshow('Elon

Musk',imgElon)

cv2.imshow('Elon Test',imgTest)

cv2.waitKey(0)
```

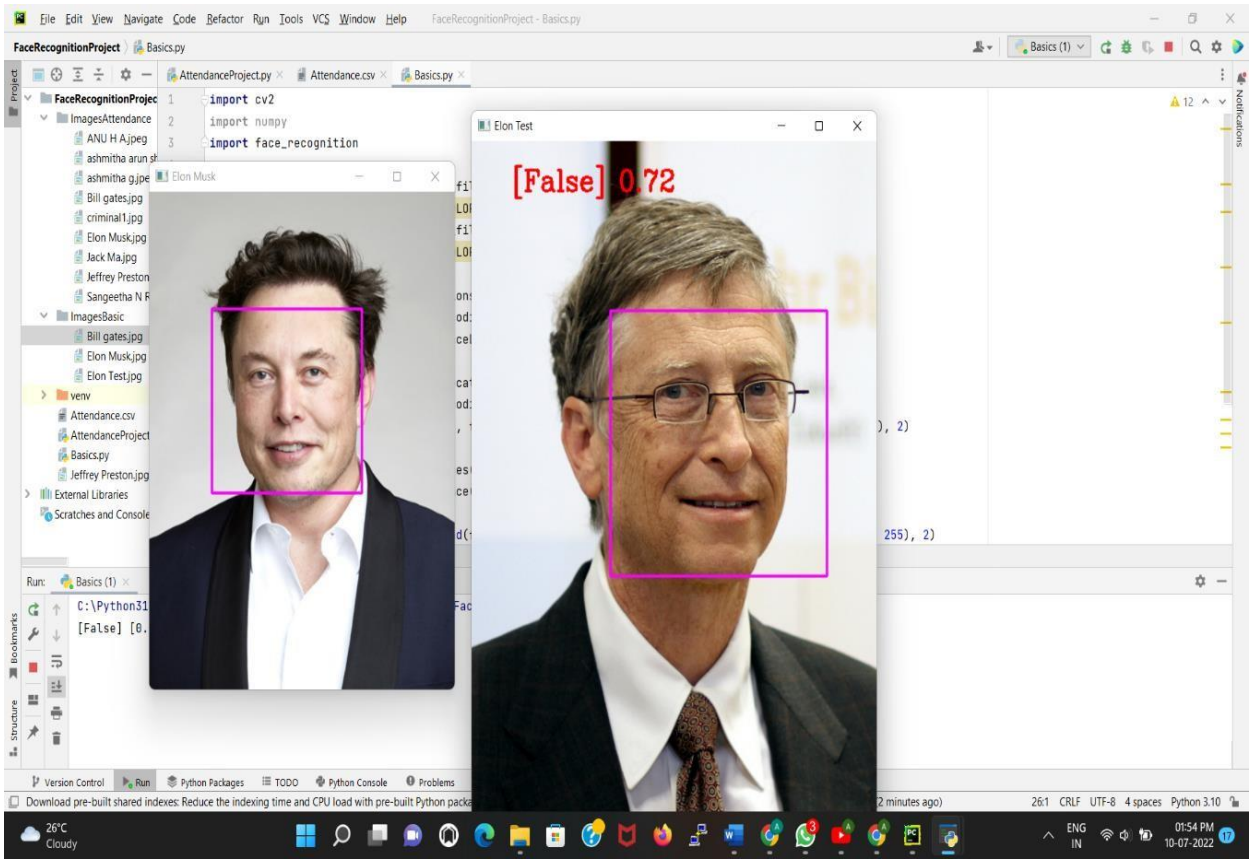



FIGURE 6: Matching the faces

**Basic.py output**

Here we are checking for programming working .

# PROGRAMMING PART-2:

```
import cv2
import numpy as np
import face_recognition
import os
from datetime import datetime

# from PIL import ImageGrab

path = 'ImagesAttendance'
images = []
classNames = []
myList = os.listdir(path)
print(myList)

imgElon = face_recognition.load_image_file('ImagesBasic/Elon Musk.jpg')
imgElon = cv2.cvtColor(imgElon, cv2.COLOR_BGR2RGB)
imgTest = face_recognition.load_image_file('ImagesBasic/Bill gates.jpg')
imgTest = cv2.cvtColor(imgTest, cv2.COLOR_BGR2RGB)

for cl in myList:
    curImg = cv2.imread(f'{path}/{cl}')
    images.append(curImg)
    classNames.append(os.path.splitext(cl)[0])
print(classNames)
```

```
def findEncodings(images):
    encodeList = []
    for img in images:
        img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
        encode = face_recognition.face_encodings(img)[0]
        encodeList.append(encode)
    return encodeList


def markAttendance(name):
    with open('Attendance.csv', 'r+') as f:
        myDataList = f.readlines()
        nameList = []
        for line in myDataList:
            entry = line.split(',')
            nameList.append(entry[0])
        if name not in nameList:
            now = datetime.now()
            dtString = now.strftime('%H:%M:%S')
            f.writelines(f'\n{name},{dtString}')


encodeListKnown = findEncodings(images)
print('Encoding Complete')
cap = cv2.VideoCapture(0)
```

```
  while True:
success, img = cap.read()
# img = captureScreen()
imgS = cv2.resize(img, (0, 0), None, , 0.25, 0.25)
imgS = cv2.cvtColor(imgS, cv2.COLOR_BGR2RGB)



facesCurFrame = face_recognition.face_locations(imgS)
encodesCurFrame = face_recognition.face_encodings(imgS, facesCurFrame)

for encodeFace, faceLoc in zip(encodesCurFrame, facesCurFrame):
    matches = face_recognition.compare_faces(encodeListKnown,encodeFace)
    faceDis = face_recognition.face_distance(encodeListKnown,encodeFace)
    print( faceDis)
    matchIndex = np.argmin(faceDis)

    if matches[matchIndex]:
        name = classNames[matchIndex].upper()
        print(name)
        y1, x2, y2, x1 = faceLoc
        y1, x2, y2,  x1 = y1 * 4, x2 * 4, y2 * 4, x1 * 4
        cv2.rectangle(img, (x1, y1), (x2, y2), (0, 255, 0), 2)
        cv2.rectangle(img, (x1, y2 - 35), (x2, y2), (0, 255, 0), cv2.FILLED)
        cv2.putText(img, name, (x1 + 6, y2 - 6), cv2.FONT_HERSHEY_COMPLEX, 1, (255, 255, 255), 2)
        markAttendance(name)

cv2.imshow('Webcam', img)
cv2.waitKey(1)
```

# CHAPTER 5

## 5.1 TEST IMAGES

### Test data image 1:

Figure 7: test data image 1

The above figure indicates the image which we have given for testing process to detect the image which we have trained.

## Test data image 2:

Figure 8: Test data image 2

The above figure2 indicates the image which we have given for testing process to detect the image which we have trained.

## 5.2 RESULT AND DISCUSSION :

The proposed system is a software system which will mark attendance using facial recognition. In this project we used OpenCV module integrated with Python which will helps the institution to make the attendance process easy and efficient. The system comprises of Computer with webcam and Wi-Fi module or Internet.

Steps of Working:

-Initiate the attendance.py python script.

- Create a DATASET of the students by uploading their images.
- A picture of the class is taken, and python file is initiated.
- Attendance is taken by cropping the faces in the picture and comparing with the faces in the database.
- If a face is matched, the responding name is marked in a EXCEL file with the current date and time.

## DEMONSTRATION AND SCREENSHOTS:

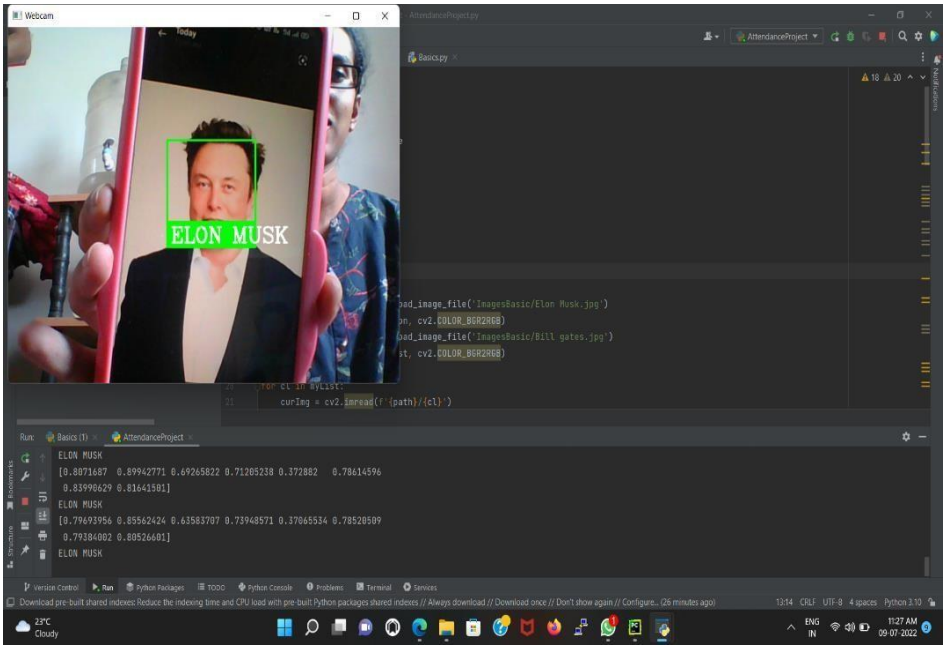



Figure 9: code snippet 1 1

The above figure indicates the pycharm command prompt.

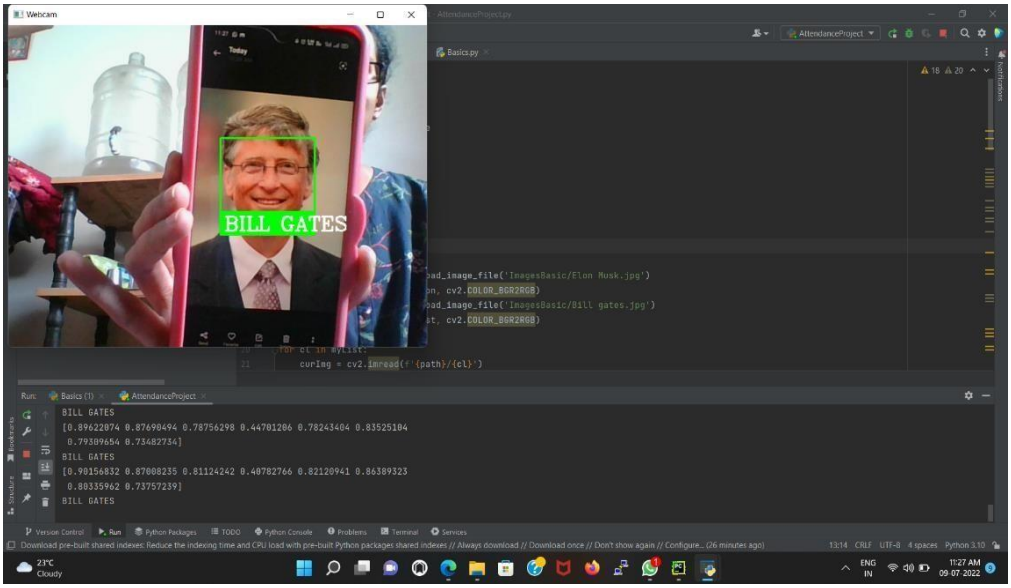
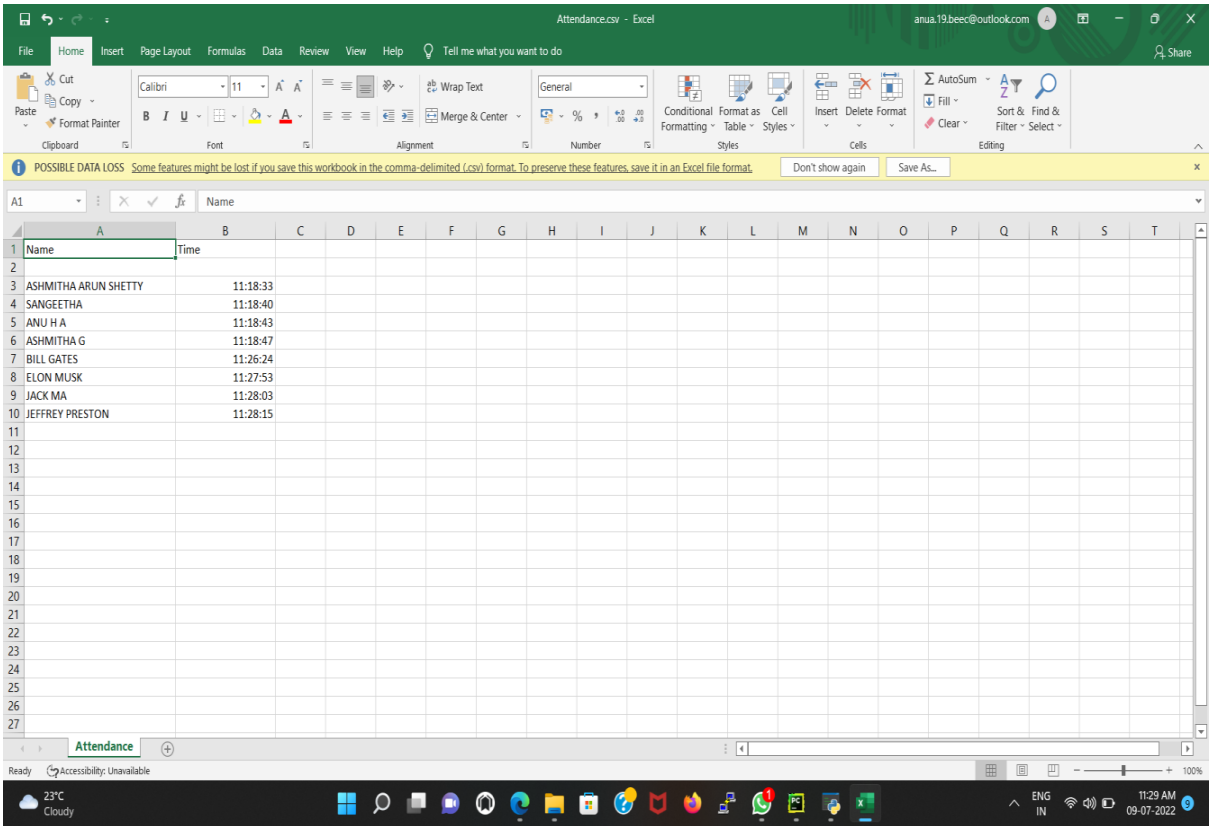



Figure 10: code snippet 2 1

The above figure indicates the pycharm command prompt.

# FINAL OUTPUT SNAPSHOTS

figure 11: output snippet 1




Here attendance is stored live in excel sheet.

# CHAPTER 6

## 6.1 ADVANTAGES AND APPLICATIONS:

**ADVANTAGES:**

- Automated time tracking system
- Cost effective
- Touchless sign in system: A post pandemic Requirement
- Facial Recognition with ageing changes and accessories
- More accurate and better worker attendance
- The ubiquity of cameras on Mobile devices
- Easy to manage
- Smart Integration

**APPLICATIONS:**

- Banking using ATM
- Voter Verification
- Residential/Office Security
- Smart Security system
- Attendance and leave management
- Information security
- Access Contro

## 6.2 FUTURE ENHANCEMENTS:

The system can be made more flexible and scalable using these recommendations. Please note that the system implemented here is just a prototype of idea presented via this project. The recommendations are as follows:

- The system can be extended to more number of students with freedom to change list of students according to class changes.
- The system can be made more flexible to allow updating of templates in case student incurs amount of change in his facial features.
- The system can also be extended to allow better face recognition algorithm in which even rotational features of face can be detected efficiently.

### Future uses of facial recognition:

- Events – both on a local and global scale – has been one of the most impacted industriesas restrictions on numbers, social distancing, mandatory masks and other safety precautions have meant that organisers have had to rethink the way events are managed. Many events over the past 18 months have been conducted virtually, however, as we start to open up and international travel starts again, event organisers are turning to facial recognition as a way of managing access, check- in, visitor experience and safety.

- Car owners can also set up permissions or restrictions for other people such as family members. For example, they could set up certain restrictions on their children learningto drive such as a time or speed limit or deny access without an adult present. If an unauthorised person enters the car, the system can notify the owner or block the car from starting. This helps prevent theft and gives owners better control of their cars.

- Preventing the frauds at ATMs in India. A database of all customers with ATM cards in India can be created and facial recognition systems can be installed. So, whenever user will enter in ATM his photograph will be taken to permit the access after it is being matched with stored photo from the database

# 6.3 CONCLUSION:

Before the development of this project. There are many loopholes in the process of taking attendance using the old method which caused many troubles to most of the institutions. Therefore, the facial recognition feature embedded in the attendance monitoring system can not only ensure attendance to be taken accurately and also eliminated the flaws in the previous system. By using technology to conquer the defects cannot merely save resources but also reduces human intervention in the whole process by handling all the complicated task to the machine. The only cost to this solution is to have sufficient space in to store all the faces into the database storage. In this project, the face database is successfully built. Apart from that, the face recognizing system is also working well. At the end, the system not only resolve troubles that exist in the old model but also provide convenience to the user to access the information collected by mailing the attendance sheet to the respected faculty.

# 6.4 REFERENCES:

1] M.A. Turk and A.P. Pentland, “Face Recognition Using Eigenfaces”,IEEE Conf. on Computer Vision and Pattern Recognition, pp. 586-591,1991. Year:2021

2] Paul Viola, Matthew Jones Conference paper- IEEE Computer SocietyConference on Computer Vision and Pattern Recognition.“Rapid objectdetection using a boosted cascade of simple features.”

3] Maliha Khan,Sudeshna Chakraborty,Rani Astya,Shaveta Khepra "Face Detection and Recognition Using OpenCV" Year: 2019 | Conference Paper | Publisher: IEEE

4] Earle Kit M. Sanchez,Noel B.Linsangan,Randy E.Angelia "Three Triangle Method for Face Recognition using Dlib and OpenCV" Year: 2020 | ConferencePaper | Publisher: IEEE

5] B. Gomathy,K.B.Sri Sathya,Sathish.J,Santhosh.S,Sabari Krishna.S "Face Recognition based student detail collection using OpenCV"Year: 2022 | Conference Paper | Publisher: IEEE

6] P Apoorva,H.C Impana,S.L Siri,M.R Varshitha,B Ramesh "Automated Criminal Identification by Face Recognition using Open Computer VisionClassifiers" 2019 3rd International Conference on Computing Methodologies and Communication Year: 2019 | Conference Paper | Publisher: IEEE

7] Vijay Kumar Sharma "Designing of Face Recognition System" 2019 International Conference on Intelligent Computing and Control Systems Year: 2019 | Conference Paper | Publisher: IEE