# University of Mumbai

# Face Recognition based Attendance System

Submitted in partial fulfillment of requirements

for completion of

## Mini-Project

by

**Dhruv Doshi**

**Roll No: 1814002**

**Shubham Bhakuni**

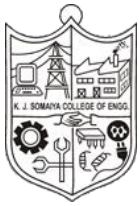
**Roll No: 1814006**

**Labdhi Jain**

**Roll No: 1814015**

Guide

Prof. Nandana Prabhu




**Department of Information Technology**

**K. J. Somaiya College of Engineering, Mumbai-77**

**(Autonomous College Affiliated to University of Mumbai)**

**Batch 2021**

# K. J. Somaiya College of Engineering, Mumbai-77

(Autonomous College Affiliated to University of Mumbai)

## Certificate

This is to certify that the report entitled **Face Recognition based Attendance System** is bona fide record of Mini-Project work done by **Dhruv Doshi, Shubham Bhakuni and Labdhi Jain** in the Sem VI, year 2021 under the guidance of **Prof. Nandana Prabhu** of Department of Information Technology in partial fulfillment of requirement for the completion of Mini-Project.

_____ _____

Guide Head of the Department

Date: 18/04/2021

Place: Mumbai-77

# K. J. Somaiya College of Engineering, Mumbai-77

(Autonomous College Affiliated to University of Mumbai)

## Certificate of Approval of Examiners

We certify that this report entitled **Face Recognition based Attendance System** is bona fide record of Mini-Project work done by Dhruv Doshi, Shubham Bhakuni and Labdhi Jain.

This project is approved for the award of credits for completing Mini-Project course

_____

Internal Examiner

_____

External Examiner

Date: 18/04/2021

Place: Mumbai-77

# K. J. Somaiya College of Engineering, Mumbai-77

(Autonomous College Affiliated to University of Mumbai)

## DECLARATION

We declare that this written report submission represents the work done based on our and / or others' ideas with adequately cited and referenced the original source. We also declare that we have adhered to all principles of intellectual property, academic honesty and integrity as we have not misinterpreted or fabricated or falsified any idea/data/fact/source/original work/ matter in our submission.

We understand that any violation of the above will be cause for disciplinary action by the college and may evoke the penal action from the sources which have not been properly cited or from whom proper permission is not sought.

| | |
|---|---|
| **Signature of the Student**<br>**1814002**<br>**Roll No.** | **Signature of the Student**<br>**1814006**<br>**Roll No.** |
| **Signature of the Student**<br>**1814015**<br>**Roll No.** | **Signature of the Student**<br><br>**Roll No.** |

**Date: 18/04/2021**

**Place: Mumbai-77**

**Dedicated to**

***Family, friends and teachers…***

# Abstract

Manual attendance is not only time consuming but also disturbs the entire class. Taking attendance manually is old school now and there seems to be a need to replace that system with a new and modernized attendance system which is more efficient, exciting and easy to implement compared to the current system of taking attendance. This is where the Face Recognition based Attendance System comes into the picture. It boasts many advantages in comparison to the manual form of attendance like readymade attendance sheet in Excel format, no need to maintain a physical register, no need to waste time in calling out students for marking their attendance to name a few.

In this mini project we have implemented the concepts which we have learned in our previous semesters. We have implemented this entire mini project in Python programming language which we have learned in semester 3 and 4. We have implemented basic machine learning concepts which we learned in semester 4 and 5. We have also learned the documentation process in our current semester 6. We had to learn new concepts of face recognition and go through the face recognition libraries for this project.

This Face Recognition based Attendance System has a promising future since it can be used in various places like schools, colleges, tuition classes, offices and other such similar places. Since the user base varies so much, this we have made this system in such a way that it can be used by an individual of any age group and any background. We have also shown in the report that why did we select face recognition over any other biometric alternatives. The system we have made can be considered as a prototype version which can be deployed in a module consisting of dedicated hardware. This would ensure portability and convenience for the users.

# Contents

# List of Figures

# List of Tables

## Nomenclature

| | |
|---|---|
| CNN | Convolutional Neural Network |
| KNN | K Nearest Neighbours |
| OpenCV | Open source computer vision |
| RFID | Radio Frequency Identification |

# CHAPTER 1

## Introduction

### 1.1 Problem Definition:

Manual attendance not only disturbs the faculty while teaching but also a distraction for the students during exam sessions. The attendance sheet is sometimes passed around the class or names have to be called out. When the number of students is more, these methods become very time consuming. Thus, an attendance system based on face recognition is proposed in this mini project. This will also solve the problem of proxies and the faculty will not have to count the number of students present in the classroom in order to confirm the presence of all students.

In this project first we have to detect the presence of a face on the screen, capture and preprocess multiple images of the face, train a model using those images, recognize the same face again with considerable accuracy and finally mark the attendance of the student in an excel sheet (the student's roll number, name, date and time).

### 1.2 Motivation:

Taking attendance manually each and every day has become tiresome for the faculty members. In this world of digitalization, marking attendance manually is considered old school. It would be both time and energy saving if the attendance system is made automated. This is what made us think about using biometrics of the student such as the facial features to record the attendance daily. This face recognition system would not only avoid the tedious task of maintaining a register for attendance but marking attendance on an excel sheet would enable the faculty to calculate the aggregate attendance and other such parameters with just a few clicks. This face recognition attendance system will be extremely convenient to use for a person of any age group. Hence, we wish to make the monotonous task of taking attendance fun, interesting and easy for everyone.

## 1.3 Scope of the project:

### 1.3.1 Objective

The face recognition based attendance system would help the faculty or organization to take attendance in a new, easy and exciting manner compared to the manual form of attendance.

### 1.3.2 Deliverables

- Checking the camera feature
- Detecting the face in the camera display screen
- Capturing images of the user's face
- Training model for the saved images
- Recognizing existing user's face
- Marking attendance in excel sheet after recognition is successful

### 1.3.3 Milestones

- Complete planning of the project by 18th February 2021.
- Start exploring OpenCV library and look for the best method for training the model for face recognition on 18th February 2021.
- Go through the OpenCV modules which are essential for the project in detail and finalize the face recognition training model by 15th March 2021.
- Start the implementation of the project on 15th March 2021.
- Start the testing process, preparation of the report and presentation from the 1st week of April 2021.
- The complete project including report and presentation should be completed in the 3rd week of April 2021.

## 1.4 Functional and non-functional Requirements:

**Functional requirements:**

| Sr. No. | Name | Description |
|---|---|---|
| 1. | Check camera | The user should be able to check if the camera is working properly and if the face detection is working properly. |
| 2. | Capture face | A user should be able to capture his face if he is using the system for the first time. System should take 50 pictures of the user for training purposes. |
| 3. | Train images | Captured images must be trained. |
| 4. | Recognize face | The user's face should be recognized the next time that user appears in front of the camera. User should be able to see his name on the screen. |
| 5. | Mark attendance | Once the required threshold is crossed, user's name, ID, and date should be marked in an excel sheet and that sheet should be saved locally. |

**Non-functional requirements:**

| Sr. No. | Name | Description |
|---|---|---|
| 1. | Reliability | The system should be reliable enough to work properly whenever required. The system should be able to differentiate between faces and should only mark the attendance if the threshold configured for the system is crossed. |
| 2. | Data recovery | Data recovery is not necessary |

| | | |
|---|---|---|
| | | since this is a prototype system, all the data is stored locally and no data backups are made. |
| 3. | Performance | A known face should be recognized within 4 seconds of detection. |
| 4. | Storage | The images captured should take minimum space to minimize the total storage required by the system. |
| 5. | Usability | The system should be easy to use for any age group since the user base could highly vary depending on where this system is implemented. |

## 1.5 Organization of report:

The report starts by displaying the certificates, declarations and the abstract which gives a gist of the project which will be explained in detail ahead. Then comes the introduction section, i.e. Chapter 1, which explains the problem definition, motivation behind the project, scope of the project, functional and non-functional requirements and organization of report.

Then the research done for this project and the concepts learned are explained in short in the section Background work which forms the Chapter 2 of this report. In Chapter 3 Implementation explains the technologies used, algorithm or the methodology used in this project and the sample images taken which show how use this system.

Chapter 4 Results and discussions mentions the findings we obtained while developing the system and while testing the system. The Chapter 5 Conclusion explains what we are concluding from the results we obtained and also the things we learnt in the course of this mini project.

At the end of this report, the references used to develop this system are mentioned and finally we have mentioned the individuals who we would like to thank for the successful completion of this mini project.

# CHAPTER 2

## Background Work

Arun Katara et al. (2017) mentioned disadvantages of RFID (Radio Frequency Identification) card system, fingerprint system and iris recognition system. RFID card systems are implemented due to their simplicity. But it is possible that a user helps his friend by swiping his/her card along with the user's own card and making an attendance entry for both. Fingerprint system is effective but it can take multiple attempts to read the fingerprint and the users have to line up and perform the verification one by one. Sometimes when the finger is dirty or sweaty, it becomes difficult for the reader to verify the fingerprint accurately which leads to multiple attempts and hence becomes time consuming. However, in face recognition systems, the user's face is always exposed but contains less information than iris. Even though iris recognition consists of more information it could invade the privacy of the user. Voice recognition is also available but when compared to other biometric options, it is less reliable and less accurate. Hence, we come to a conclusion that face recognition systems take lesser details of the user than iris, it is easier to use when compared to fingerprint recognition, it is more reliable than RFID card system and finally it is more accurate than voice recognition. Thus, face recognition based attendance system is implemented in this mini project.

*Documentation of face_recognition library* [2] shows how to implement face recognition step by step. It describes how to find faces in a photograph, also how to find faces using deep learning (convolutional neural networks). Next, finding and recognizing unknown faces in a photograph based on the photos of known people and recognizing faces in live video using the computer's webcam using OpenCV. Recognizing faces with a K-nearest neighbor classifier is also described in this documentation.

*An article on Medium* titled "Machine Learning is Fun! Part 4: Modern Face Recognition with Deep Learning" [10] gives an in detailed explanation of the theory behind face recognition. It explains functions like finding all the faces in a photo which works on a method invented in 2015 called Histogram of Oriented Gradients or HOG for short (Fig. 1). Next it explains posing and projecting faces which is done using an algorithm called face landmark estimation. There are lots of ways to do this but in this article uses the approach invented in 2014 by Vahid Kazemi

and Josephine Sullivan. The basic idea in this is to come up with 68 specific points called *landmarks* that exist on every face as shown in fig. 2. This algorithm tries to bring the eyes and mouth in the center.

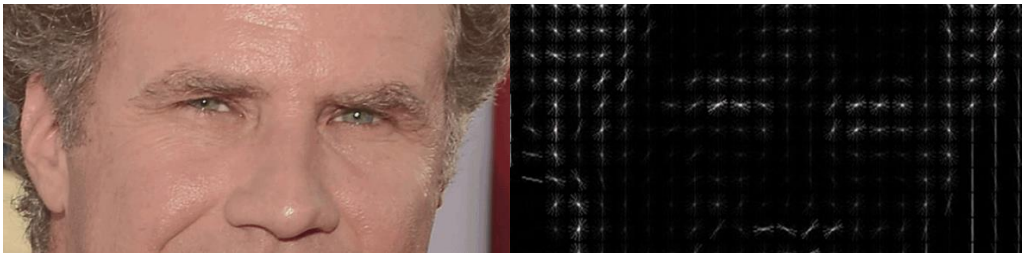


Fig. 1: HOG representation

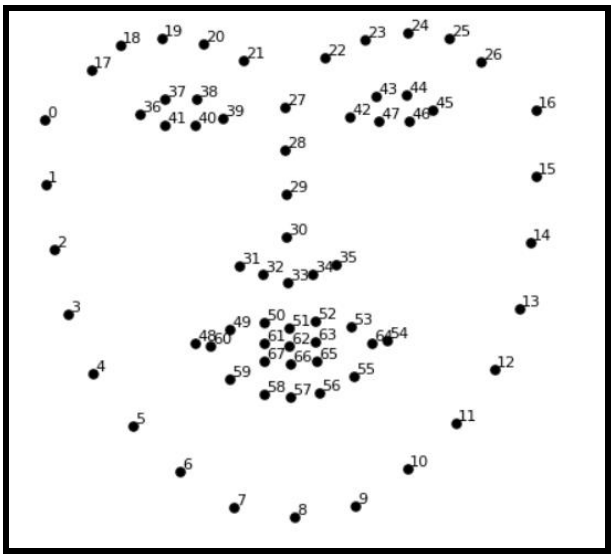


Fig. 2: The 68 landmarks we will locate on every face. This image was created by Brandon Amos of CMU who works on OpenFace.

Next is the extraction of a few basic measurements of the face which is termed as Encoding faces. Then we can measure the unknown face in the same way and find the known face with the closest measurements. This calculation of the most optimal features for comparison is done using Deep Convolutional Neural Networks. The idea of reducing complicated raw data like a picture into a list of computer-generated numbers comes up a lot in machine learning. And the final task to be done is to find the person in our database of known people who has the closest measurements to our test image. This can be done using any machine learning algorithm like linear SVM classifier.

# CHAPTER 3

## Implementation

The Face Recognition based Attendance system is implemented using a combination of various technologies and algorithms. This section gives a short description of the technologies and algorithms used and then a step by step illustration of the usage of this system.

### 3.1 Technologies used:

### 3.1.1 Python 3.7

Python is an interpreted high-level general-purpose programming language. Python's design philosophy emphasizes code readability with its notable use of significant indentation. Its language constructs as well as its object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

Since 2003, Python has consistently ranked in the top ten most popular programming languages in the TIOBE Programming Community Index where, as of February 2021, it is the third most popular language (behind Java, and C). It was selected Programming Language of the Year (for "the highest rise in ratings in a year") in 2007, 2010, 2018, and 2020 (the only language to do so four times).

An empirical study found that scripting languages, such as Python, are more productive than conventional languages, such as C and Java, for programming problems involving string manipulation and search in a dictionary, and determined that memory consumption was often "better than Java and not much worse than C or C++".

### 3.1.2 OpenCV 4.4.x

Officially launched in 1999 the OpenCV project was initially an Intel Research initiative to advance CPU-intensive applications, part of a series of projects including real-time ray tracing and 3D display walls. The main contributors to the project included a number of optimization experts in Intel Russia, as well as Intel's Performance Library Team.

OpenCV is written in C++ and its primary interface is in C++, but it still retains a less comprehensive though extensive older C interface. All of the new developments and algorithms appear in the C++ interface. There are bindings in Python, Java and MATLAB/OCTAVE. The API for these interfaces can be found in the online documentation. Wrappers in several programming languages have been developed to encourage adoption by a wider audience. In version 3.4, JavaScript bindings for a selected subset of OpenCV functions was released as OpenCV.js, to be used for web platforms.

### 3.1.3 dlib 19.18.0

DLib is an open source C++ library implementing a variety of machine learning algorithms, including classification, regression, clustering, data transformation, and structured prediction. DLib provides a good framework for developing machine learning applications in C++.

DLib is much like DMTL in that it provides a generic high-performance machine learning toolkit with many different algorithms, but DLib is more recently updated and has more examples. DLib also contains much more supporting functionality.What makes DLib unique is that it is designed for both research use and creating machine learning applications in C++.

### 3.1.4 face_recognition 1.3.0

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 Labeled Faces in the Wild benchmark.

### 3.1.5 Pillow 8.0.1

The Python Imaging Library adds image processing capabilities to your Python interpreter. This library provides extensive file format support, an efficient internal representation, and fairly powerful image processing capabilities. The core image library is designed for fast access to data stored in a few basic pixel formats. It should provide a solid foundation for a general image processing tool.

### 3.1.6 Numpy 1.18.4

NumPy is a Python library used for working with arrays. It also has functions for working in domain of linear algebra, fourier transform, and matrices. NumPy was created in 2005 by Travis Oliphant. It is an open source project and you can use it freely. NumPy stands for Numerical Python. In Python we have lists that serve the purpose of arrays, but they are slow to process. NumPy aims to provide an array object that is up to 50x faster than traditional Python lists. The array object in NumPy is called ndarray, it provides a lot of supporting functions that make working with ndarray very easy. Arrays are very frequently used in data science, where speed and resources are very important.

### 3.1.7 cmake 3.18.2.post1

CMake is used to control the software compilation process using simple platform and compiler independent configuration files, and generate native makefiles and workspaces that can be used in the compiler environment of your choice. The suite of CMake tools were created by Kitware in response to the need for a powerful, cross-platform build environment for open-source projects such as ITK and VTK. The CMake python wheels provide CMake 3.18.4.

## 3.2 Algorithm:

### 3.2.1 Haar Cascades for Object Detection:

Object Detection is a computer technology related to computer vision, image processing and deep learning that deals with detecting instances of objects in images and videos.

The algorithm uses edge or line detection features proposed by Viola and Jones in their research paper. The Model is trained and available at https://github.com/opencv/opencv/tree/master/data/haarcascades.

Haar cascade works by finding edge by comparing pixel values across the image and if the average between light and dark pixel comes approximately 1, an edge is detected. Various haarCascade models are generated to identify along with a face in an image, particular features such as nose, eyes, etc.

### 3.2.2 KNN Classifier for image recognition from sklearn.neighbors

Here we have used Neighbors-based classification is a type of *instance-based learning* or *non-generalizing learning*: it does not attempt to construct a general internal model, but simply stores instances of the training data. Classification is computed from a simple majority vote of the nearest neighbors of each point: a query point is assigned the data class which has the most representatives within the nearest neighbors of the point.

Here we have used Ball Tree Algorithm, which is a space partitioning algorithm, which partitions data points into a nested set of hyperspheres known as "balls".The resulting data structure has characteristics that make it useful for a number of applications, most notably nearest neighbor search.

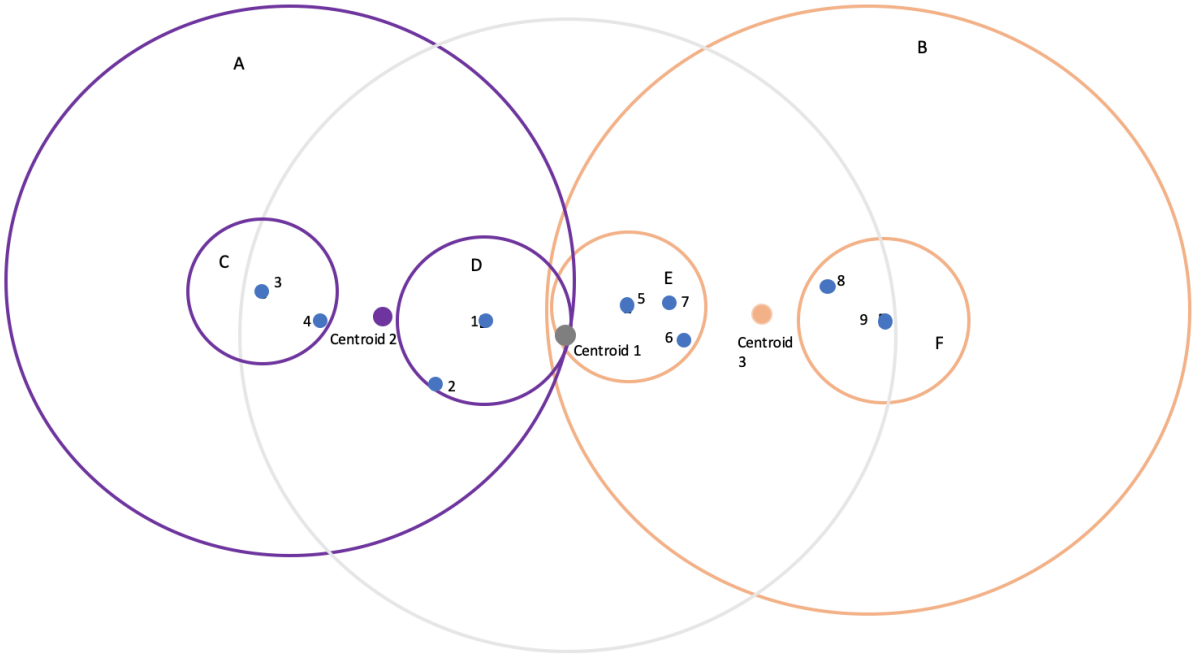



Fig 3. KNN classifier example

## 3.3 Activity diagram:

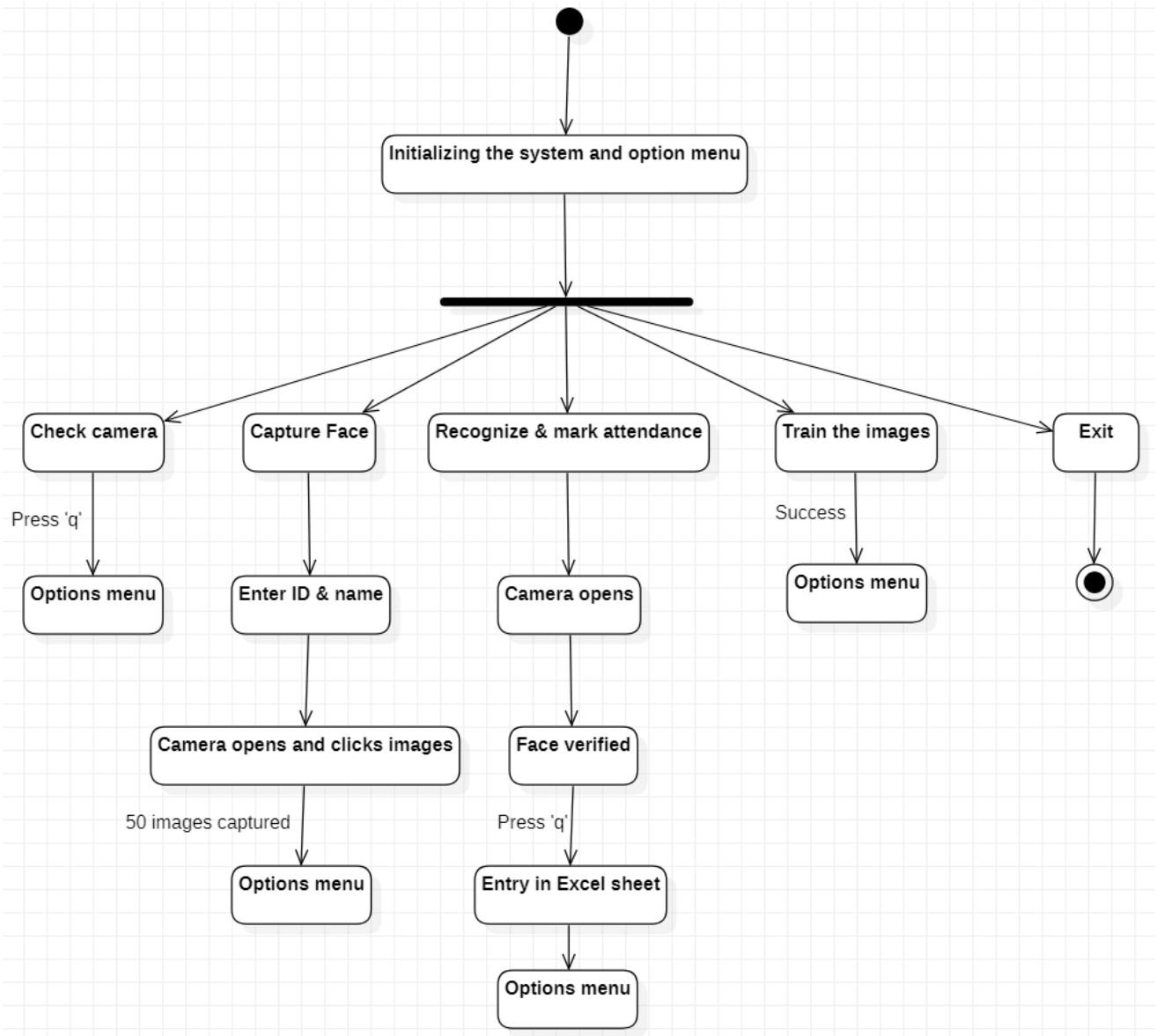



Fig 4. Activity diagram

## 3.4 Procedure (step by step):

**Select an option**


Fig 5. Select an option

**Option 1- Check device's camera is selected**

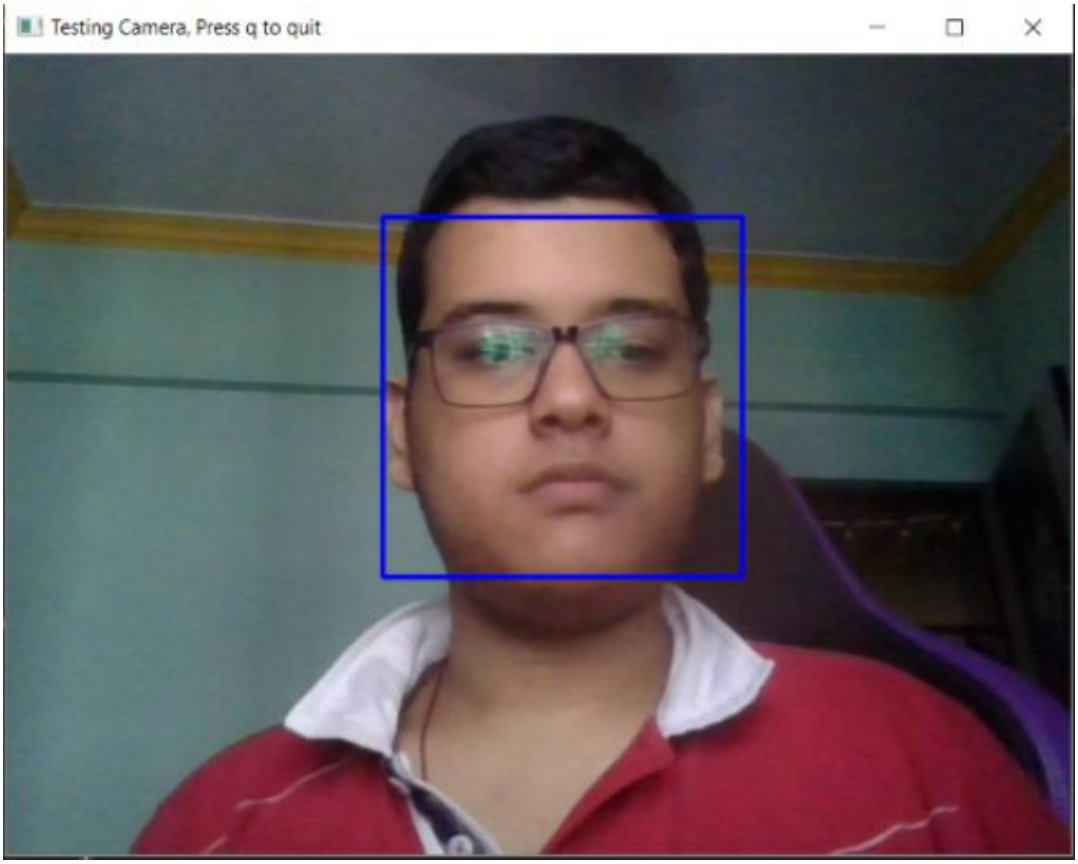



Fig 6. Check camera

**Camera is released after user presses ‘q’**

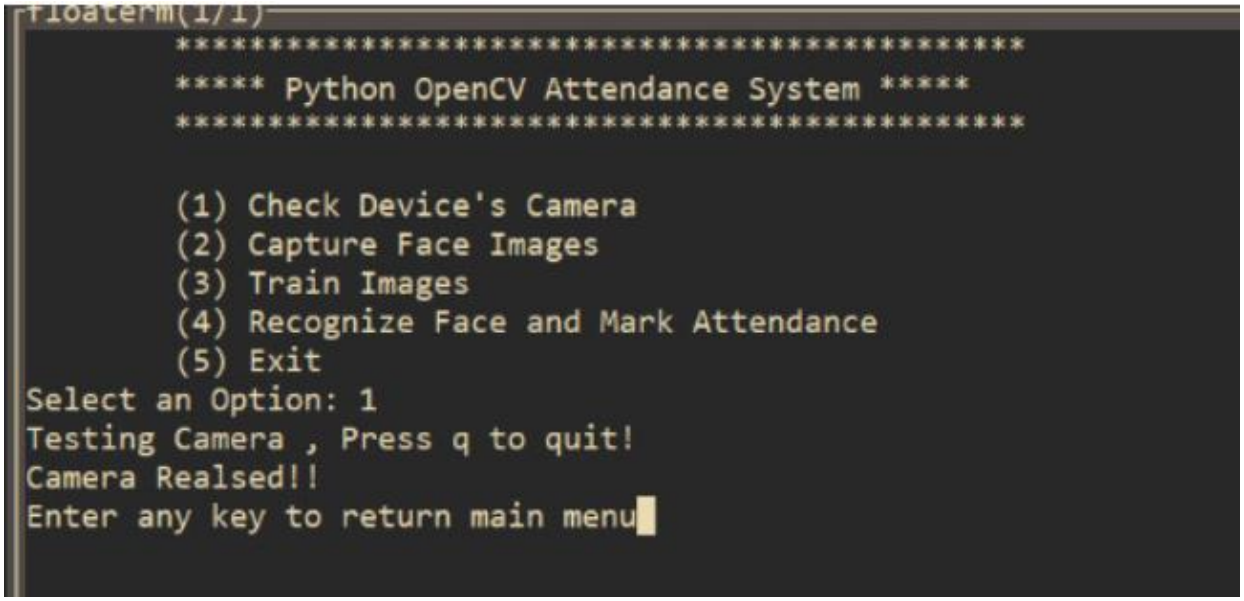



Fig 7. Camera released

**Option 2- Capture Face Images is selected**

**(User must enter roll number (ID) and name before he can capture face images)**

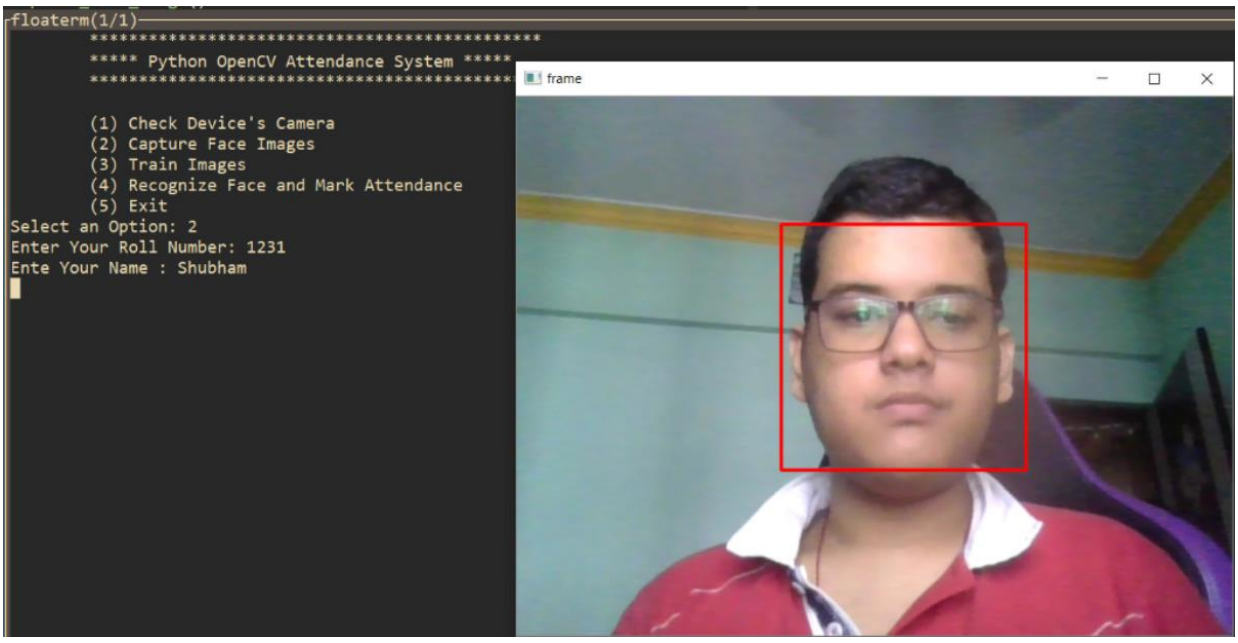



Fig 8. Capture face

**Face captured successfully**

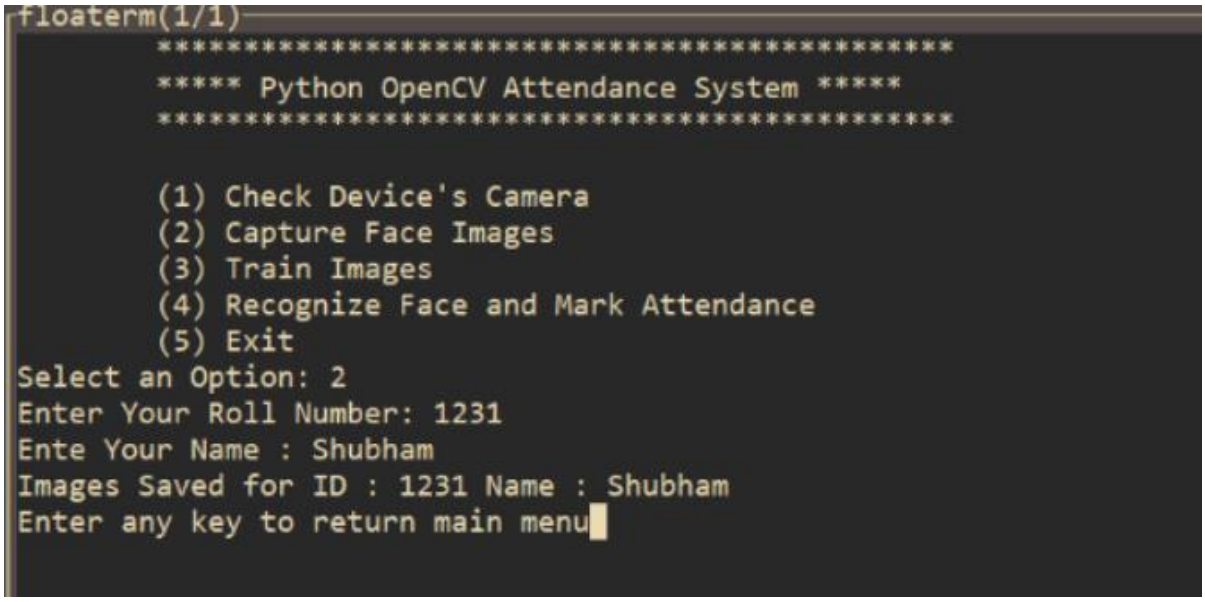



Fig 9. Face capture complete

**Option 3- Train Images**

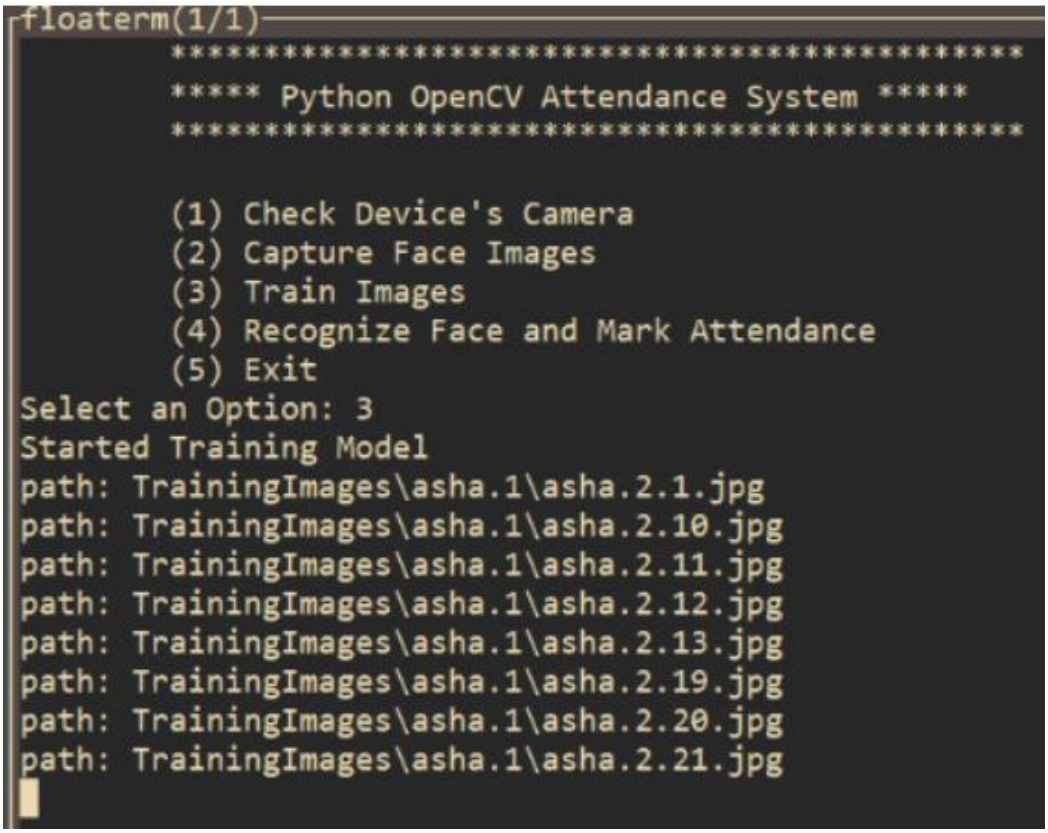



Fig 10. Training images

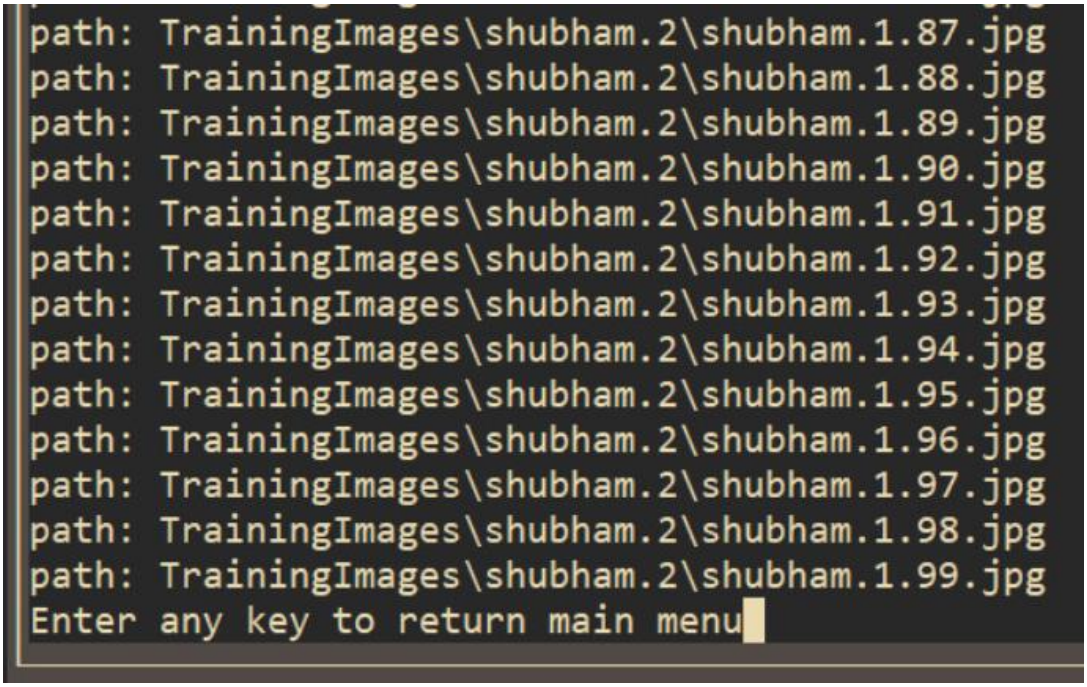



Fig 11. Training images

**Option 4- Recognize Face and Mark Attendance**

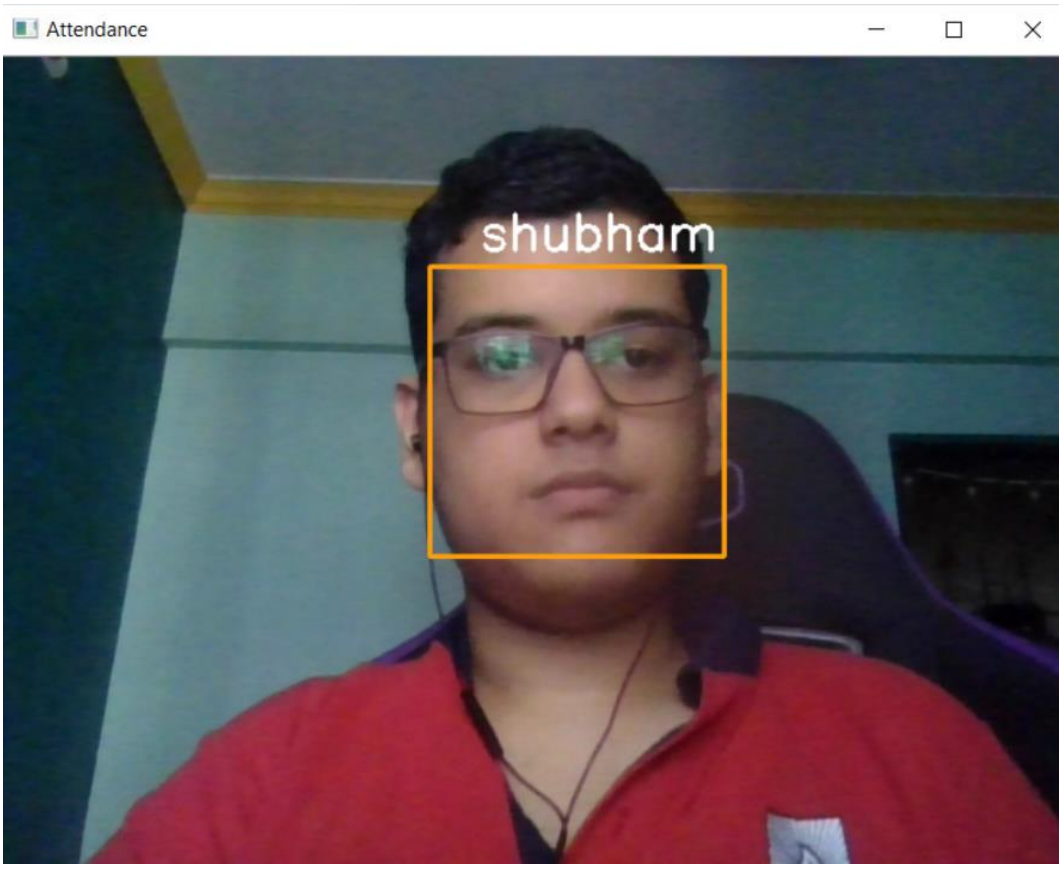



Fig 12. Recognize face

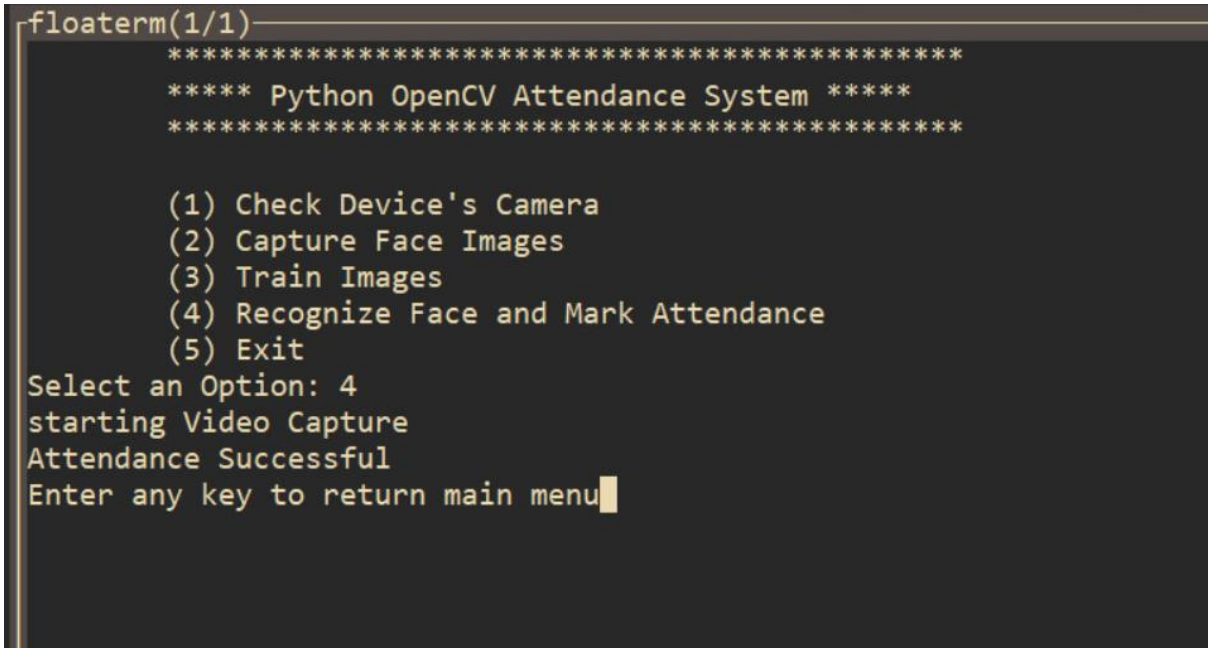



Fig 13. Attendance marked

**Attendance is entered successfully in the Excel Sheet along with ID, name, date and time.**

| | A | B | C | D |
|---|---|---|---|---|
| 1 | Id | Name | Date | Time |
| 2 | 2 | shubham | 19-04-2021 | 12:53:35 |
| 3 | 1 | harsh | 19-04-2021 | 12:56:09 |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |
| 7 | | | | |
| 8 | | | | |
| 9 | | | | |
| 10 | | | | |

Fig 14. Excel sheet

## 3.4 Testing

**We test whether the system is able to perform well when we change the number of users in front of the camera. The performance criteria for the testing will be that if the system can accurately mark the attendance of the 'n' number of users in front of the camera in the excel sheet or not.**

### 3.4.1 Test case 1 (TC01):

Test case name: No user test

No face is detected when there is no one in front of the camera

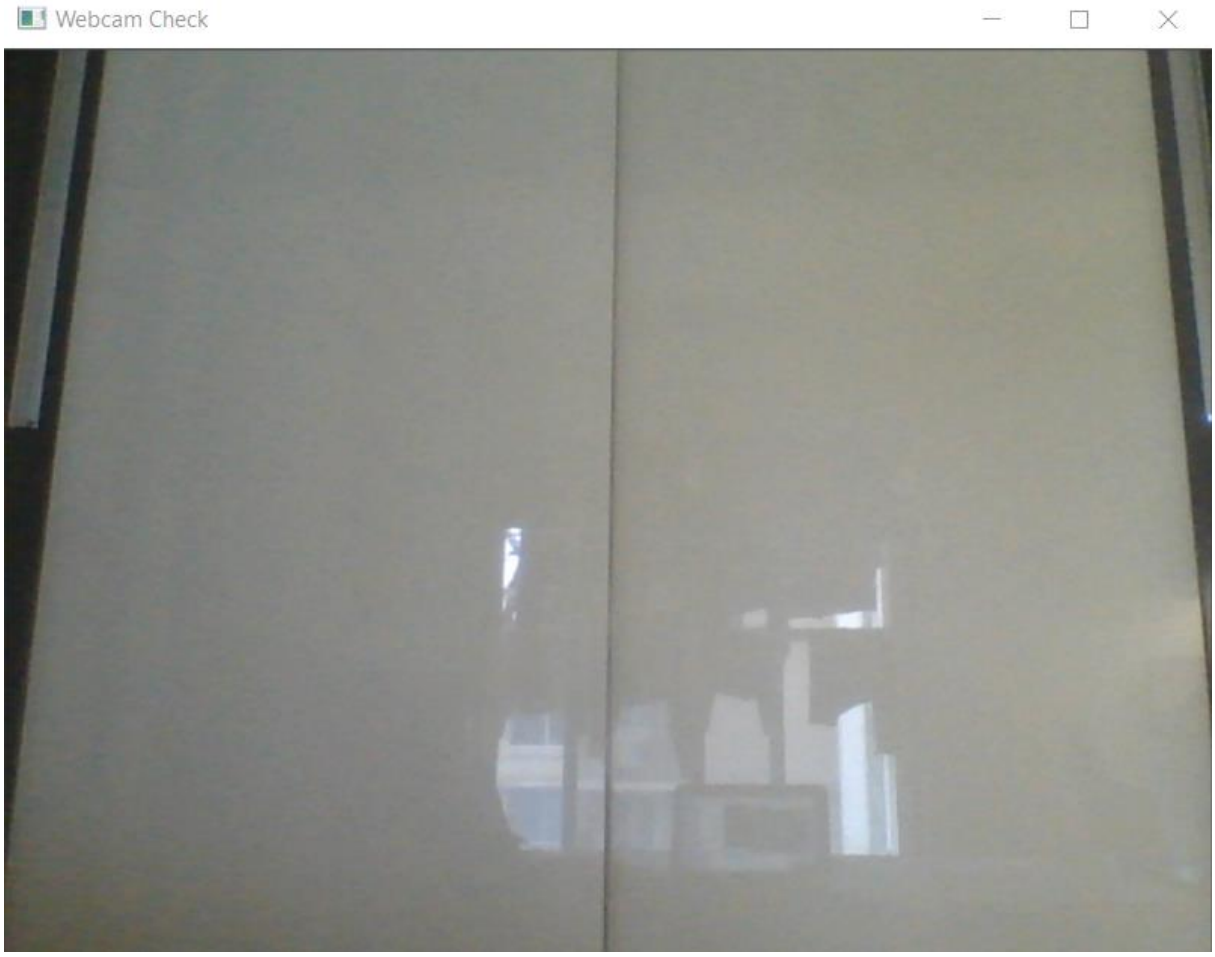



Fig 15. No people in frame

### 3.4.2 Test case 2 (TC02):

Test case name: 1 user test

One user stands in front of the camera

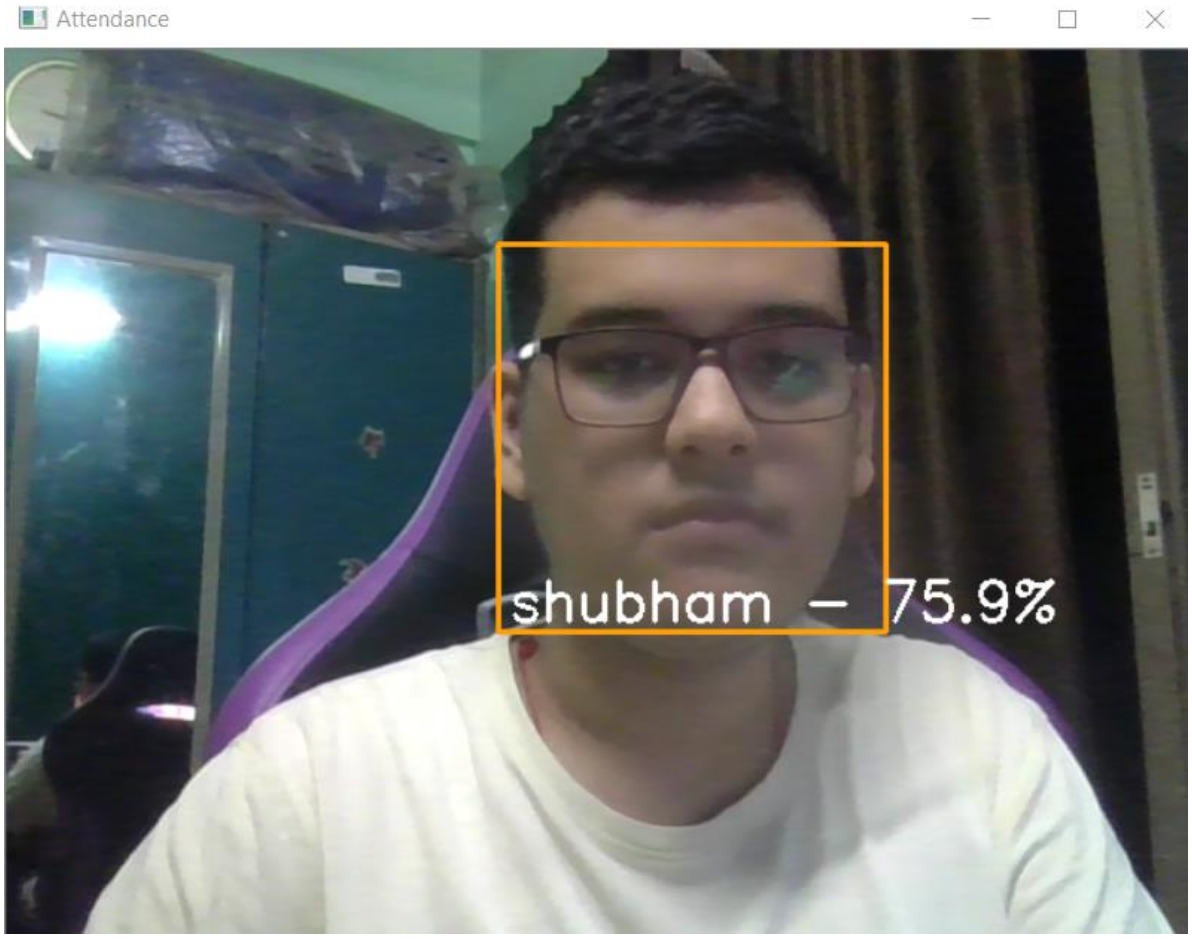



Fig 16. One person in the frame

| | A | B | C | D |
|---|---|---|---|---|
| 1 | Id | Name | Date | Time |
| 2 | 2 | shubham | 21-04-2021 | 16:27:51 |
| 3 | | | | |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |

Fig 17. Entry of one person

### 3.4.3 Test case 3 (TC03):

Test case name: 2 users test

Two users stand in front of the camera

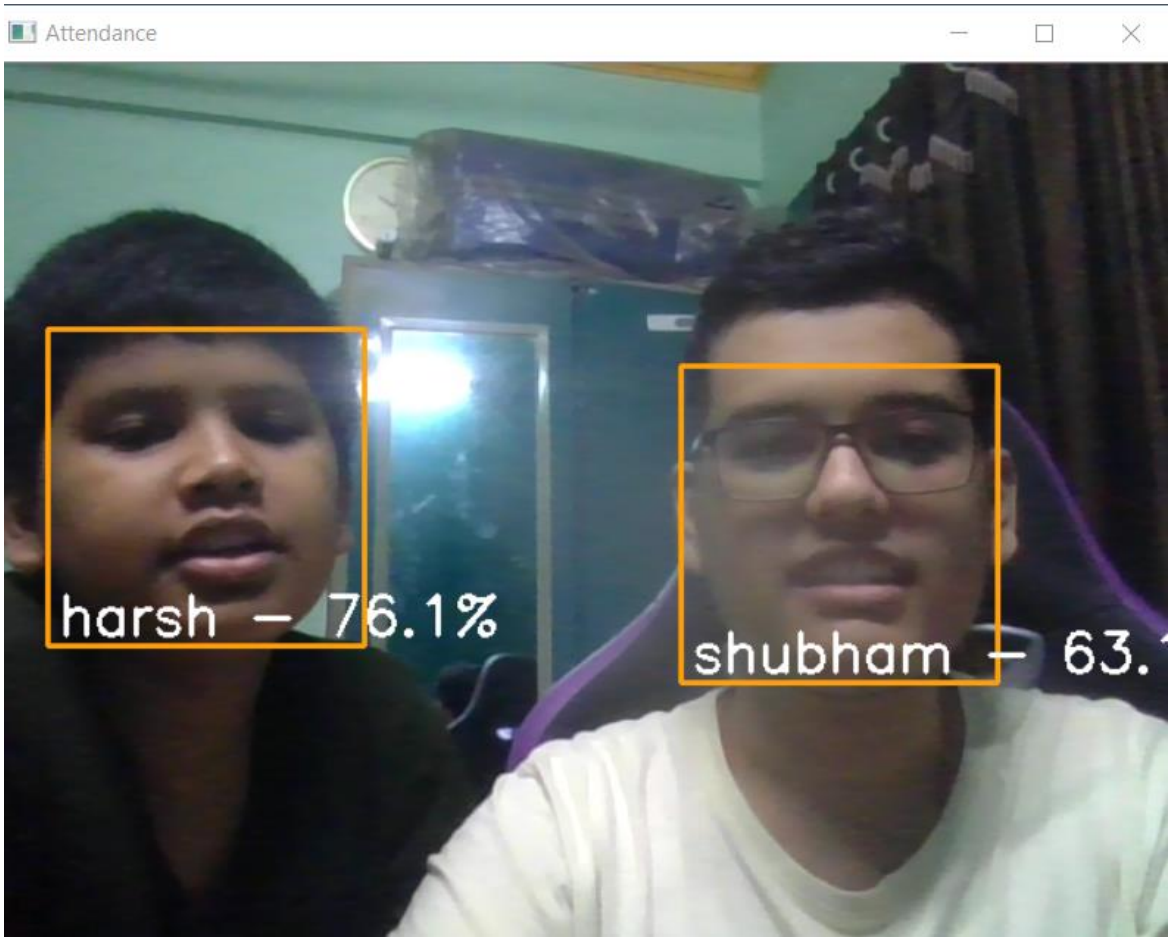



Fig 18. Two people in the frame

J14

| | A | B | C | D |
|---|---|---|---|---|
| 1 | Id | Name | Date | Time |
| 2 | 2 | shubham | 21-04-2021 | 16:27:51 |
| 3 | 3 | harsh | 21-04-2021 | 21:49:33 |
| 4 | | | | |
| 5 | | | | |
| 6 | | | | |
| 7 | | | | |

Fig 19. Entry of two people

### 3.4.4 Test case 4 (TC04):

Test case name: 3 users test

Three users stand in front of the camera

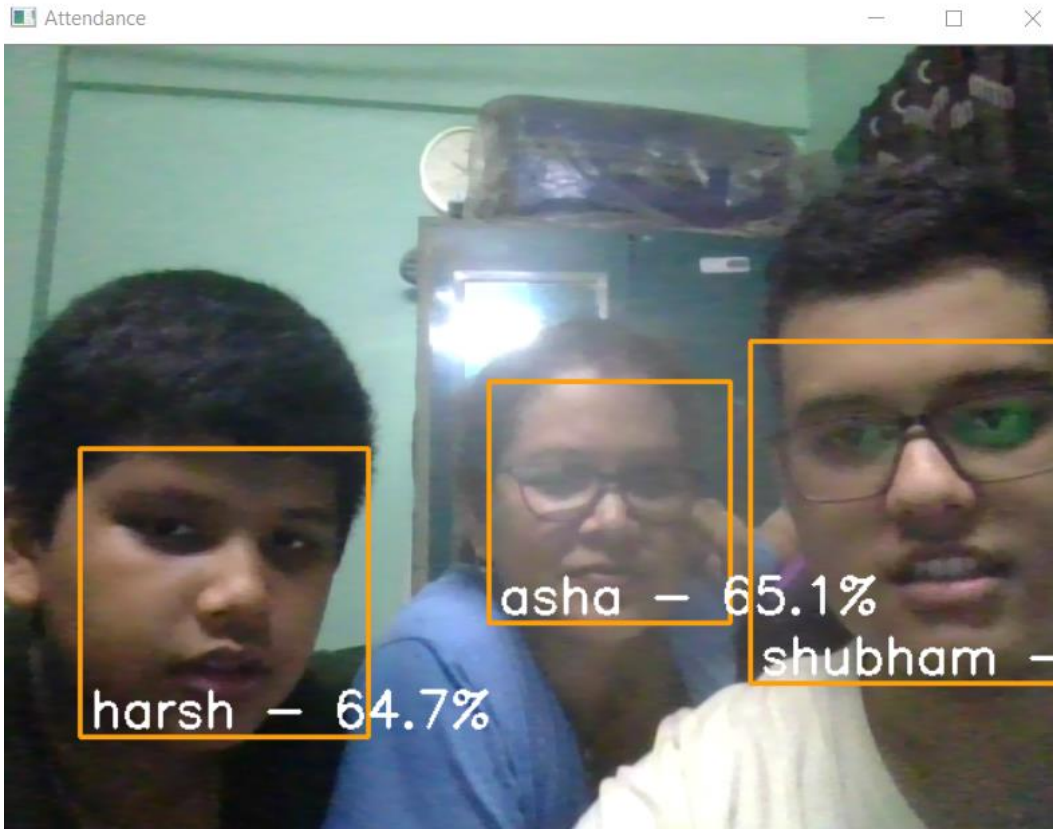



Fig 20. Three people in the frame

G15

| | A | B | C | D |
|---|---|---|---|---|
| 1 | Id | Name | Date | Time |
| 2 | 2 | shubham | 21-04-2021 | 16:27:51 |
| 3 | 3 | harsh | 21-04-2021 | 21:49:33 |
| 4 | 1 | asha | 21-04-2021 | 21:50:14 |
| 5 | | | | |
| 6 | | | | |
| 7 | | | | |

Fig 21. Entry of three people

### 3.4.5 Test case 5 (TC05):

Test case name: 4 users test

Four users stand in front of the camera

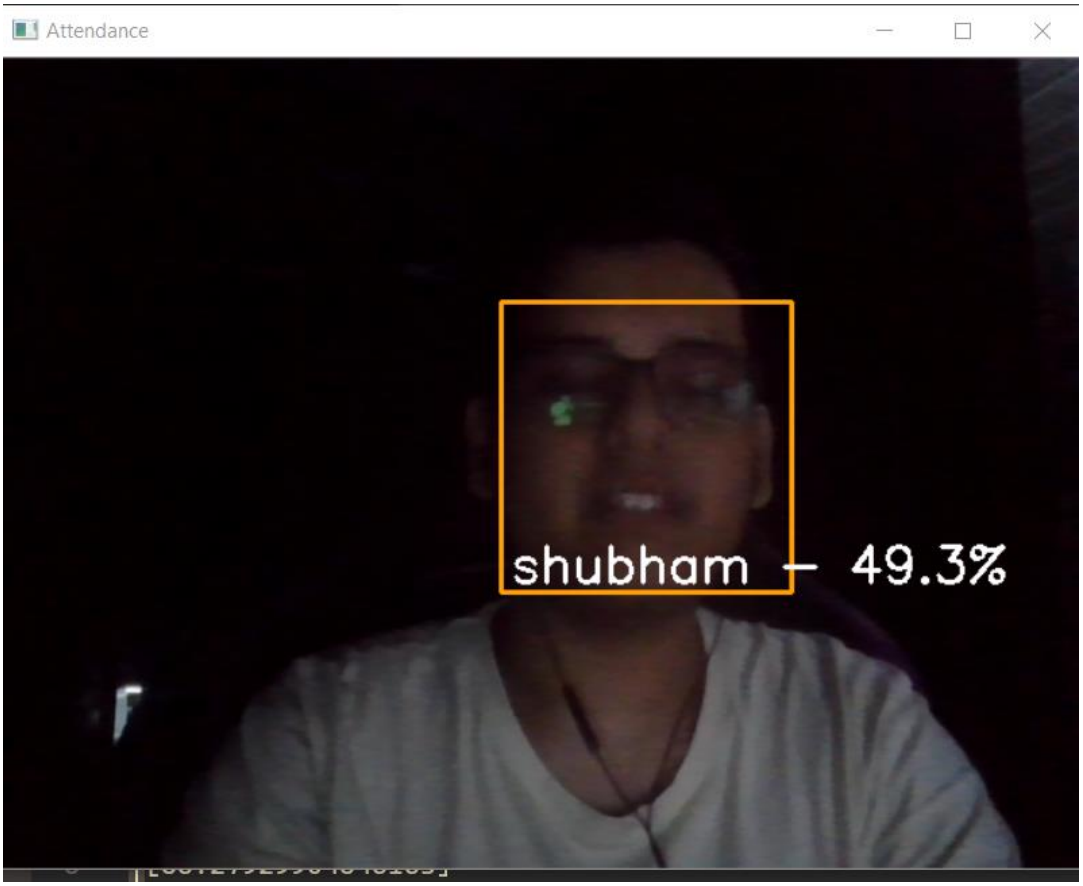



Fig 22. Peformance in bad lighting

## 3.5 Implementation schedule (Gantt Chart):

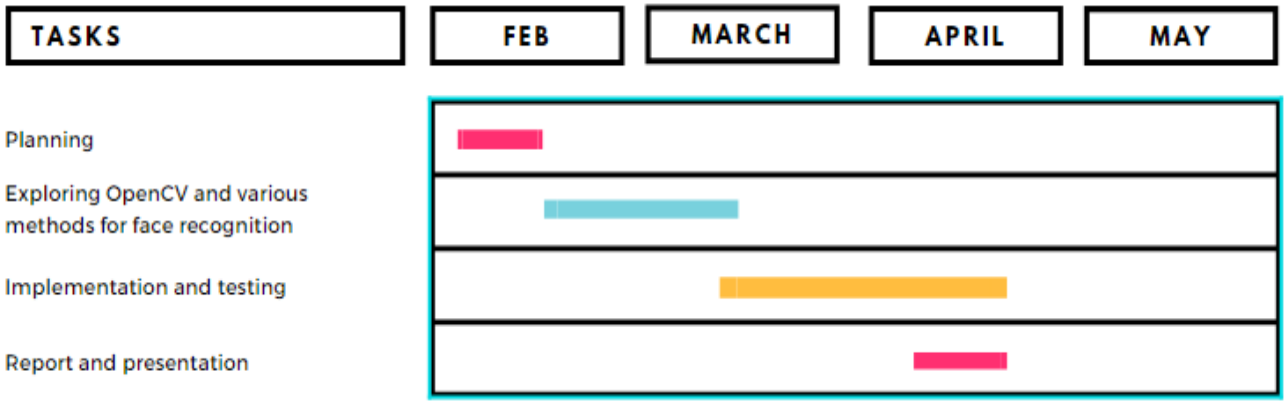



Fig 23. Gantt chart

# CHAPTER 4

## Results and Inferences

### 4.1 Results:

**We obtained the following results from the testing of the system we performed:**

| Test case no. | Test name | Expected result | Observed result | Pass/ Fail |
|---|---|---|---|---|
| TC01 | No user test | No face is detected and no entry is made in the excel sheet | No face is detected and no entry is made in the excel sheet | Pass |
| TC02 | 1 user test | 1 face is detected and 1 entry of that user is made in the excel sheet | 1 face is detected and 1 entry of that user is made in the excel sheet | Pass |
| TC03 | 2 users test | 2 faces are detected and 2 entries of those users are made in the excel sheet | 2 faces are detected and 2 entries of those users are made in the excel sheet | Pass |
| TC04 | 3 users test | 3 faces are detected and 3 entries of those users are made in the excel sheet | 3 faces are detected and 3 entries of those users are made in the excel sheet | Pass |
| TC05 | Bad lighting condition | Face is accurately detected | Face is recognized with an accuracy of about 49%<br><br>Face is sometimes not recognized or incorrectly recognized. | Fail |

**Result of the overall system:**

| Test case no. | Test name | Expected result | Observed result | Pass/ Fail |
|---|---|---|---|---|
| TC06 | Good lighting condition | Face is accurately detected | Face is recognized with an accuracy of about 60% to 70% | Pass |
| TC07 | Average efficiency | We obtain the average efficiency by comparing 10 face recognition cases. | Efficiency of about 60% is obtained. | Pass |

## 4.2 Inferences:

The classifier is able to identify individuals in front of the camera with good accuracy but this accuracy is highly dependent upon the quality of training images provided to the system. The model was taking relatively more amount of time during training on our personal computers which are not very efficient for this training task. But if there is dedicated hardware for performing only processing this training function the process would be faster. Another alternative is that the images could be trained using NVIDIA GPU using CUDA cores for faster processing.

We can infer that the system gives an efficiency of about 60% but the peak efficiency obtained was about 80% which we are considering good enough for an attendance system. No doubt the efficiency of the system needs improvement. Lighting condition plays an important role. While testing in bad lighting conditions, the performance of the system was found out to be below average. Hence, this system needs improvement when the images it gets are taken in dark, dull or a badly lit environment.

## 4.3 Limitations of this project:

- *System cannot distinguish between a real person's face (actually standing in front of the camera) and a face shown in a photograph.*

  This forms one of the major limitations of this system since it is possible for a person to enter proxy attendance for someone else using his/her photograph. This could lead to fraudulent entries. This limitation can be removed by using 3D imaging techniques which could differentiate between a 2D face shown in a photograph and a 3D face of an actual person standing in front of the camera.

- *System performance is below average when the lighting conditions are bad.*

  As elaborated in the previous section this system is unable to perform well when the training images provided are taken in dark, dull or poorly lit conditions or the user to trying to get is face verified in those similar conditions. In both cases, the system performs poorly. This limitation could be solved by applying preprocessing algorithms to the input images.

- *After face recognition, 'q' must be pressed to mark the attendance in the excel sheet: it doesn't happen automatically.*

  This is a minor issue in the system. After the face is verified by the system, it should automatically mark the attendance and close the camera. But in this system, the user will have to press a button in order to close the camera after the face is verified.

# CHAPTER 5

## Conclusion

### 5.1 Conclusion:

In this modernizing world, this face recognition based attendance system seems promising enough to replace the manual form of attendance. This system doesn't only make the dull task of marking attendance interesting but also makes it easier to manage and analyze the attendance. We have successfully completed the following functionalities that were desired in the beginning of the project:

- Checking the camera feature
- Detecting the face in the camera display screen
- Capturing images of the user's face
- Training model for the saved images
- Recognizing existing user's face
- Marking attendance in excel sheet after recognition is successful

Due to time constraints we were unable to fulfil the optional requirement of sending an email to the faculty after the attendance sheet is ready. The face_recognition libraries which we have used has given good accuracy and good results throughout our testing.

### 5.2 Scope for future work:

This mini project on Face Recognition based Attendance System should be treated as a prototype of a full-fledged attendance system which can be deployed in a dedicated device (which includes camera and software) for face recognition or in a laptop/personal computer which would then use its web camera for detecting and recognizing faces. This attendance system can be used in schools, colleges, tuition centers, offices and other such places. Many other features could be added to this mini project like sending the excel sheet to the supervisor via email. Accuracy of the system could be improved and better image preprocessing could be done to improve the problem of images taken in poorly lit conditions.

## 5.3 Learnings:

The project helped in learning about various new concepts and libraries. Firstly, OpenCV was explored and the functionalities required for the project were studied in further detail. Second, the working of Haar Cascade algorithm for face detection and recognition was studied. But as mentioned above, Haar cascade's accuracy for face recognition was not good enough. Next, the face_recognition library along KNN classifier was explored for image recognition from sklearn.neighbor and its different functionalities. But before implementing this library the knowledge of the working of face detection and recognition was required. Hence multiple articles on Medium were referred to understand the same. How to plan and manage a project in a given time period was also learned, i.e. giving an apt amount of time to each and every aspect of the project. Hence, this project helped in learning many new things and also brushed up some concepts which were known earlier.

# REFERENCES


[1] Arun Katara, Mr. Sudesh V. Kolhe, Mr. Amar P. Zilpe, Mr. Nikhil D. Bhele, Mr. Chetan J. Bele. (2017). "Attendance System Using Face Recognition and Class Monitoring System", International Journal on Recent and Innovation Trends in Computing and Communication

[2] Adam Geitgey. "face-recognition 1.3.0", Accessed on April 15, 2021. Available: https://pypi.org/project/face-recognition/

[3] Ageitgey. "Face_recognition", Accessed on April 15, 2021. Available: https://github.com/ageitgey/face_recognition

[4] Ageitgey. "Find-faces-in-picture", Accessed on April 15, 2021. Available:https://github.com/ageitgey/face_recognition/blob/master/examples/find_faces_in_picture.py

[5] Ageitgey. "Find-faces-in-picture-cnn", Accessed on April 15, 2021. Available:https://github.com/ageitgey/face_recognition/blob/master/examples/find_faces_in_picture_cnn.py

[6] Ageitgey. "find-facial-features-in-picture", Accessed on April 15, 2021. Available:https://github.com/ageitgey/face_recognition/blob/master/examples/find_facial_features_in_picture.py

[7] Ageitgey. "recognize-faces-in-pictures", Accessed on April 15, 2021. Available:https://github.com/ageitgey/face_recognition/blob/master/examples/recognize_faces_in_pictures.py

[8] Ageitgey. "facerec-from-webcam", Accessed on April 15, 2021. Available:https://github.com/ageitgey/face_recognition/blob/master/examples/facerec_from_webcam.py

[9] Ageitgey. "face-recognition-knn", Accessed on April 15, 2021. Available:https://github.com/ageitgey/face_recognition/blob/master/examples/face_recognition_knn.py

[10] Adam Geitgey. “Machine Learning is Fun! Part 4: Modern Face Recognition with Deep Learning”, Accessed on April 15, 2021. Available:https://medium.com/@ageitgey/machine-learning-is-fun-part-4-modern-face-recognition-with-deep-learning-c3cffc121d78

[11] K. M. H. Mubin. “FRAS”, Accessed on April 15, 2021. Available:https://github.com/kmhmubin/Face-Recognition-Attendance-System

[12] https://www.canva.com/graphs/gantt-charts/

# ACKNOWLEDGEMENTS

We would like to thank everyone who had contributed to the successful completion of this mini project. First, we would like to express our utmost gratitude to our mini project guide Prof. Nandana Prabhu for helping us improve the project and bringing it to the stage it is currently at by guiding us throughout the development of this project. We would also like to thank our college K.J. Somaiya College of Engineering and our department of Information technology for providing us with this opportunity.