

Attendance Monitoring System

By

**PRAGATI DOBARIYA [15BCE154]
MANALI GOKANI [15BCE157]**



**DEPARTMENT OF COMPUTER ENGINEERING
Ahmedabad 382481**

ATTENDANCE MONITORING SYSTEM

Minor Project

Submitted in fulfillment of the requirements

For the degree of

Bachelor of Technology in Computer Engineering

By

**PRAGATI DOBARIYA [15BCE154]
MANALI GOKANI [15BCE157]**

Guided By
DR.VIJAY UKANI
[DEPARTMENT OF COMPUTER ENGINEERING]



**DEPARTMENT OF COMPUTER ENGINEERING
Ahmedabad 382481**

CERTIFICATE

This is to certify that the Project entitled “ATTENDANCE MONITORING SYSTEM” submitted by Pragati Dobariya [15BCE154] & Manali Gokani [15BCE157], towards the partial fulfillment of the requirements for the degree of Bachelor of Technology in Computer Engineering of Nirma University is the record of work carried out by her under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for examination.

Dr. Vijay Ukani
Associate Professor
Department of Computer Engineering
Institute of Technology,
Nirma University,
Ahmedabad

Dr. Sanjay Garg
H.O.D.
Department of Computer Engineering,
Institute of Technology,
Nirma University,
Ahmedabad

ACKNOWLEDGEMENT

I sincerely thank our department for the academic advancement it has provided to us during the third year and finally provided us an opportunity for the project work. We Pragati Dobariya and Manali Gokani are sincerely thankful to Dr. Vijay Ukani for giving us the necessary guidance and suggestion to complete this project and term report on time. I might want to express my true on account of my relatives and my college mate for input on it and recommendations for making my venture more plausible.

ABSTRACT

In this Minor project we had implemented firstly to detect each face and train the face data into the folder using trainer program in python using OpenCV. After that faces from the folder had made available to be recognized now at present each faces are recognized and displaying each person's name according to the faces of the person stored in the folder. Objective of this project is to mark the attendance of the students whose face is recognized by the faces which are trained. This project is completed till the faces are recognized and it's displayed as present if the faces are recognized. Important finding is that by this project if done perfectly, it would be used in many organizations, institutes. This project can be used for many other applications where face recognition can be used for. This project uses algorithm of Haar's cascades for face detection and also use fisher faces with create lbph [Local Binary Patterns Histograms] recognizer for face recognition.

CONTENTS

Certificate

Acknowledgement

Abstract

Table of Contents

List of figures

Chapter 1	Introduction	07
1.1	Introduction	
1.2	Techniques	
Chapter 2	Literature Survey	08
2.2	Implementation and working	
Chapter 3	Algorithms	11
3.1	Basics	
3.2	Types	
Chapter 4	Haar Cascade Algorithm	12
4.1	Introduction	
4.2	How it works?	
4.2.1	Software & Hardware Requirements	
Chapter 5	List of figures	13
Chapter 6	Conclusion	18
Chapter 7	References	19

Chapter 1 Introduction

1.1 Introduction

A confront acknowledgment framework is a PC framework which distinguishes a man from picture or video from a video source. This is finished by contrasting chosen confront sympathy from picture in the database. It is utilized as a part of security frameworks and in numerous different spots. It can likewise be utilized as a part of the place of fingerprints and eye acknowledgment framework. Face acknowledgment framework has many preferences contrasted with unique mark and eye acknowledgment framework. It is significantly more renowned in light of its contactless procedure Face acknowledgment framework can catch pictures from the separation with touching it. Thus, confront acknowledgment I utilized as a part of wrongdoing division since it records confront pictures in database and later can be utilized for distinguishing other individual.

1.2 Techniques

Face acknowledgment framework distinguish confront empathy by utilizing points of interest from the face. For instance, it might utilize confront position, estimate, shape, nose, eyes and so forth. These can be utilized to look for other spared pictures in the database. Numerous calculations standardize the face pictures and just spare information of face pictures by packing it. The caught picture is then contrasted and face information.

So examine will transform this fantasy innovation into reality. This component if executed on cell phones will enable clients to dispose of recollecting pins, passwords or example which can be effectively speculated by any of our companion or associate. This is quicker contrasted with writing the secret key or stick. Despite the fact that this may flop now and again like the instance of indistinguishable twins, yet isn't exceptionally regular issue.

1.3 Tools

This paper depicts a framework that can identify and track human face continuously utilizing haar-like highlights where the discovery calculation depends on wavelet change. In PC vision, low level handling includes picture preparing errands in which the nature of the picture is enhanced for the advantage of human eyewitnesses and more elevated amount schedules to perform better [Viola and Jones, 2001]. Middle of the road level preparing includes the procedures of highlight extraction and example location undertakings.

OpenCV (Open Source Computer Vision Library) is a library of programming capacities basically went for ongoing PC vision, created by Intel and now upheld by Willow Garage [Lu et al., 1999]. It is free for use under the open source BSD permit. The library is cross-stage. It concentrates for the most part on continuous picture preparing. In the event that the library discovers Intel's Integrated Performance Primitives on the framework [Open Source Computer Vision Library Reference Manualintel; Gary Bradski and Adrian Kaehler O'Reilly, 2008], it will utilize these restrictive upgraded schedules to quicken.

Chapter 2 Literature Survey

Face Recognition is a standout amongst the most inclining subjects in the field of software engineering for examine works. It goes under biometrics strategy for one of a kind distinguishing proof which interestingly perceive people in light of their inborn physical or behavioral characteristics. In software engineering field this frameworks are utilized as a part of security frameworks for get to control. Different spots where this innovation is utilized are recognizing crooks, opening the PCs or cell phones and so on. This is thought to be a standout amongst the most difficult innovation as a result of issues, for example, light variety and stance variety. Face of a same individual may appear to be unique on account of progress in brightening.

Same face may appear to be unique if saw through various point. A wide range of sorts of thoughts were proposed and furthermore actualized to make this facial acknowledgment conceivable. Previously utilized strategies incorporate systems which depended on 2D design acknowledgment. This was not effective as it couldn't deal with the issues talked about before. Another technique was to utilize separate measuring calculations which can figure remove between imperative facial highlights like eyes, nose and lips. This technique was change over the past strategy however it was not sufficiently exact to be utilized as a part of securing things.

So examine will transform this fantasy innovation into reality. This component if executed on cell phones will enable clients to dispose of recollecting pins, passwords or example which can be effectively speculated by any of our companion or associate. This is quicker contrasted with writing the secret key or stick. Despite the fact that this may flop now and again like the instance of indistinguishable twins, yet isn't exceptionally regular issue.

2.1 Implementation and working

(Hanzratech Face Recognition 2015) Finding a valid resource for face recognition with OpenCV has been full of hassles. Thereby in this project, we have stumbled upon OpenCV with Python.

The entire project phase evolution can be categorized into below steps:

- A. To accumulate a fine database for face images with multiple photos tagged and pinned to a single face/person.
- B. Later, we trained the face recognizer by detecting faces in database Images
- C. And to derive the actual output we test the face recognizer to detect Faces. Fortunately the training fetched us with correct results.

About Yale Face Database and its integration to our project:

It comprises of 165 grayscale images of 15 persons in gif exclusive extension. In every photo, the person/face has variegated facial expressions – set of feelings like joyous, gloomy, lazy/boring, stagnant, poker face, etc. Out of those 11 images we took 10 images as input to put inside the trained face recognizer. The left one photo has been for a draft run to verify as well as rectify the inner mechanism of face recognition algorithm.

(Hanzratech Face Recognition 2015) Implementation:

After establishing database, the programming of face recognition algorithm needs to be carried out as follows:

A. Importing modules:

1. Cv2 – which comprises functions for face recognition.
2. OS - We have used this module to fetch the image names (in database Directory) and then fetch the individual number, to be labelled for the face.
3. Image – used to have .gif and grayscale format support.
4. Numpy - Our images are stored in numpy arrays and data structures.

B. Loading the face detection Cascade

Herein we have used haar cascades that is inbuilt in OpenCV. To load the cascade we have cv2.CascadeClassifier function which directly fetches path from the cascade xml file.

C. Creating the Face Recognizer Object

OpenCV has three face recognizer functions (along with object functions embedded as

FacRecognizer.train and FaceRecognizer.predict) named:

- i. CreateEigenFaceRecognizer ()
- ii. CreateFisherFaceRecognizer ()
- iii. CreateLBPHFaceRecognizer ()

(Real time face recognition) Amidst all three, we have stumbled upon createBPHFaceRecognizer () – AKA Local

Binary Patterns Histograms Face Recognizer. CreateLBPHFaceRecognizer () have been in wide use for texture/pattern matching and has lucid libraries for python developers.

D. Creating function to get the training set going:

- i. We initially defined get_images_and_labels function.
- ii. The function fetched us two row list, one consisting the recognized faces and III. Other row containing the labels for detected faces.
- iii. We append both the rows corresponding to regions of interest (in face, for all person's faces).
- iv. To read images directly with .gif support, we used PIL's image module.
- v. This image module got us compatible with the grayscale formats.
- vi. CascadeClassifier.detectMultiScale – detects faces in photo by returning the list of faces by rectangles in format.
- vii. To get the training set ready it is essential to fetch an absolute path. After the detected faces are retrieved – tuple 1 has faces of 1st individual and tuple-2 has face photos of 2nd individual.
- viii. With recognizer.train (photos, np.array (labels)) we made a testing phase setup. Later to test the face recognizer (usually .sad extension is used), FaceRecognizer.predict function is initialized. This later assigns needed labels and then all labels are assimilated in.
- ix. nbr_predicted.x
- x. To check the accuracy of face detection algorithm, we made sure its confidence value comes around 0.0. One can also deploy threshold to let that face not get recognized.

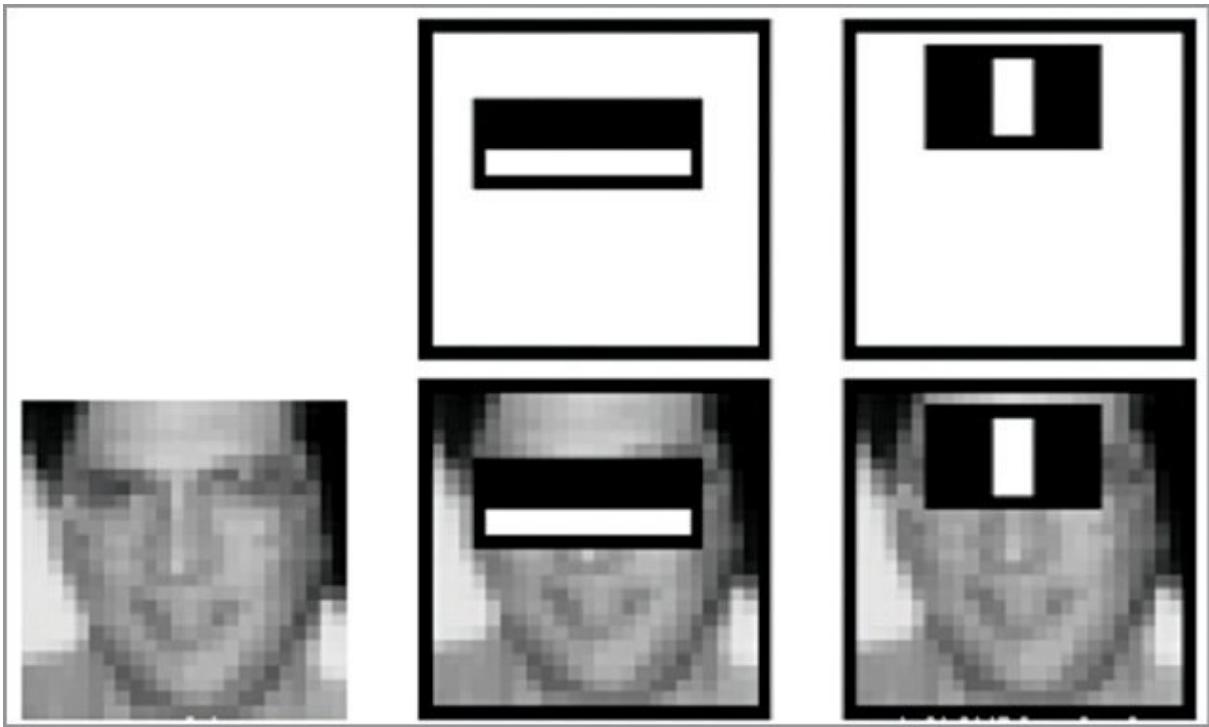


Fig 2.1: This Figure shows the process done in the face which are captured by using OpenCV program. Section defines the face detection , Section 2 ,3,4,5 shows how the eyes, nose ,forehead is detected by the haar cascade algorithm.

Working and Implementation of image and how it is processed by OpenCV.

The picture of the face caught by web-cam with the assistance of Processing, OpenCV experiences distinctive strides as specified underneath. Produce rectangle class which monitors the face facilitates. Make an occasion of the OpenCV library. This serial library is expected to speak with the OpenCV. Change Screen Size Parameters on differentiate/brilliance esteems. Change over the picture originating from webcam to greyscale organize. See whether any appearances were distinguished. In the event that a face is discovered, discover the midpoint of the main face in the edge. Control these qualities to discover the midpoint of the rectangle. See whether the Y part of the face is underneath the center of the screen, on the off chance that it is beneath the center of the screen.

Refresh the tilt position variable to bring down the tilt servo. See whether the Y part of the face is over the center of the screen. See whether the X segment of the face is to one side of the center of the screen. Refresh the container position variable to move the servo to one side. See whether the X segment of the face is to one side of the center of the screen. Refresh the skillet position variable to move the servo to one side. The dish and tilt position of the servo engine connected with web camera is specifically relative to the serial order of the directions to the OpenCV of the X and Y segments of the face from midpoint of the rectangle. It demonstrates the aftereffect of the face identification and demonstrates the face discovery and in addition following. By utilizing this approach it was discovered that time taken to distinguish the face was under 1 second which implies that this setup can be utilized as a part of ongoing. The identification effectiveness was incredibly enhanced by utilizing OpenCV. The normal edge rate was observed to be 15 fps.

Chapter 3 Algorithms

3.1 Basics

Popular face recognition prominent face acknowledgment calculations are PCA (Principle segment examination) utilizing Eigen esteems and vectors. The concealed markov model and chart coordinating utilizing Fisher confront calculation. The Multilinear subspace learning procedure utilizing tensor portrayal. Numerous different calculations are straight separate examination, Elastic Bunch, and so forth.

3.2 Types

Face Recognition algorithms can be divided into two main approaches

1. Geometric: The image which looks at distinguishing features such as skin, color, shape, etc.

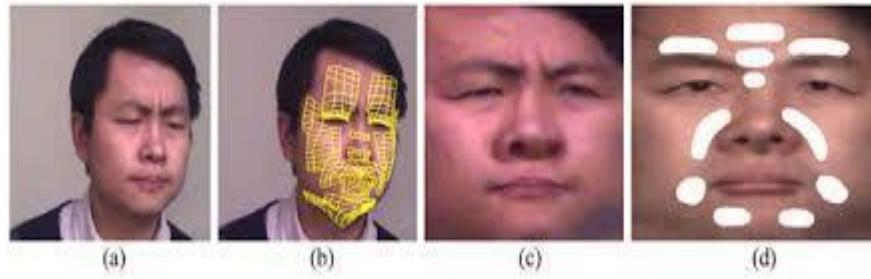


Fig 3.1 Geometric based on (a) skin (b) color (c) shape (d) facial regions which are selected for feature extraction

Photometric which compresses the images in data and saved it in database and compares the captured image with the saved data.

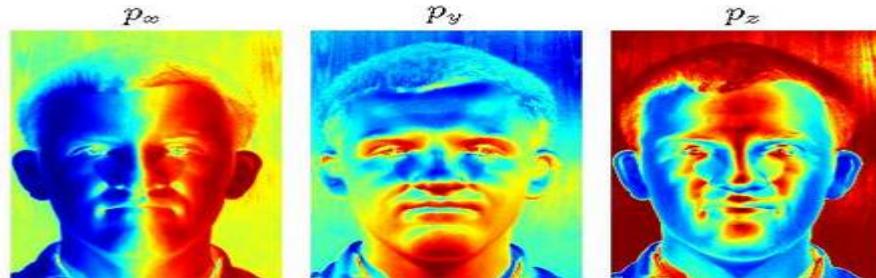


Fig 3.2 this figure show the image extraction of (P_∞) Face extraction, (P_y) Background subtraction, (P_z) Image compression which is done.

Chapter 4 Haar Cascade Algorithm

4.1 Introduction

Algorithms could also normalize the input face images and then wrap the face data, only saving the data useful for face recognition. An input image is compared with the face data when the programs runs.

How it works?

A Haar-like feature takes up a nearby rectangular region at a specific location in a detection window, adds the intensity of pixels in each region and proceeds to obtain the difference between the sums obtained. The image is divided into subsections and this difference is used to categorize the subsections. For instance, consider an image database which has human faces.[1] In all the faces, the common observation would be that the eye region would be darker than the cheek part. “Therefore a common haar feature for face detection is a set of two adjacent rectangles that lie above the eye and the cheek region. The position of these rectangles is defined relative to a detection window that acts like a bounding box to the target object (the face in this case).”

“Principal Component Analysis using Eigen faces”, “Linear Discriminate Analysis”, “Elastic Bunch Graph Matching using the Fisher face algorithm”, “the Hidden Markov model”, “the Multilinear Subspace Learning using tensor representation”, and the “neuronal motivated dynamic link matching” are some of the most popular recognition algorithms. The image below shows a demo of how the Eigen faces algorithm normalizes the images.

A Haar-like feature takes up a nearby rectangular region at a specific location in a detection window, adds the intensity of pixels in each region and proceeds to obtain the difference between the sums obtained.

$$\text{Sum} = I(C) + I(A) - I(B) - I(D)$$

Where points $\{A, B, C, D\}$ belong to the integral image $\{I\}$

4.2.1 Software & Hardware Requirements

Hardware requirements

- Intel dual core Processor or higher.
- minimum 512MB of RAM.
- minimum 120GB of Hard Disk.
- webcam or Laptop's camera.

Software Requirements

- Python
- OpenCV
- Libraries :- numpy, recognizer, scipy, pillow, os, date & time , cmake

Chapter 5 List of Figures

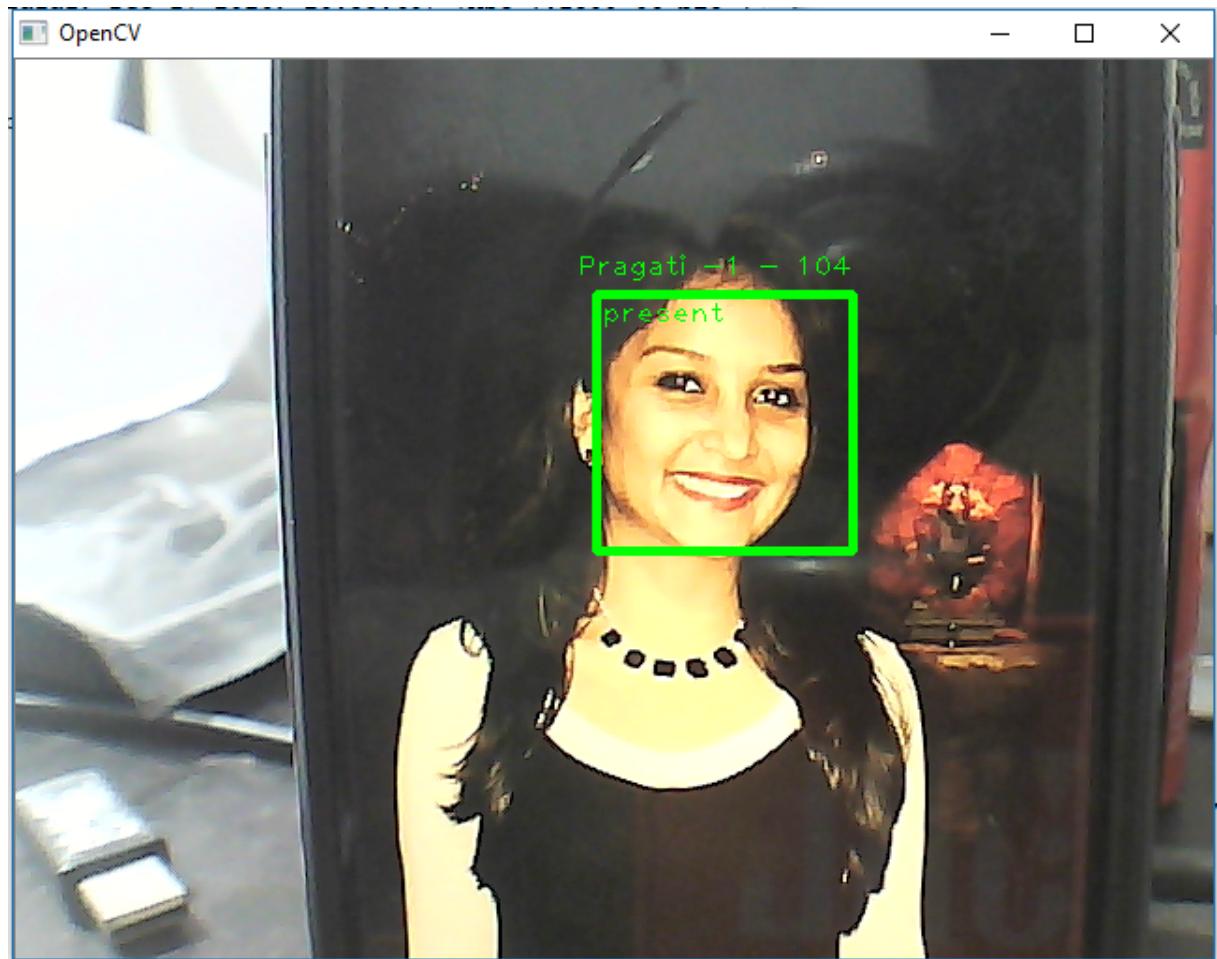


Fig 5.1 shows that the face is recognized when the image is trained and stored in the folder.

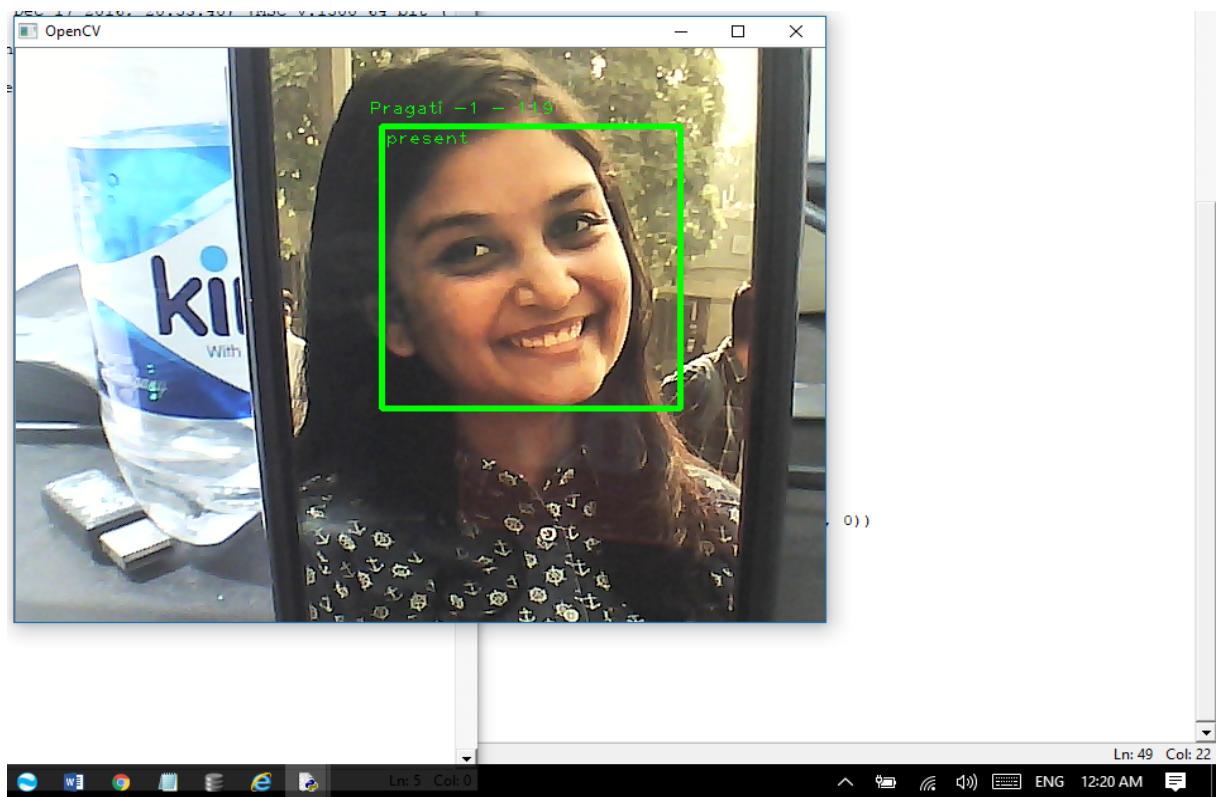


Fig 5. 2 shows the face detected and recognized by running the face recognition program.

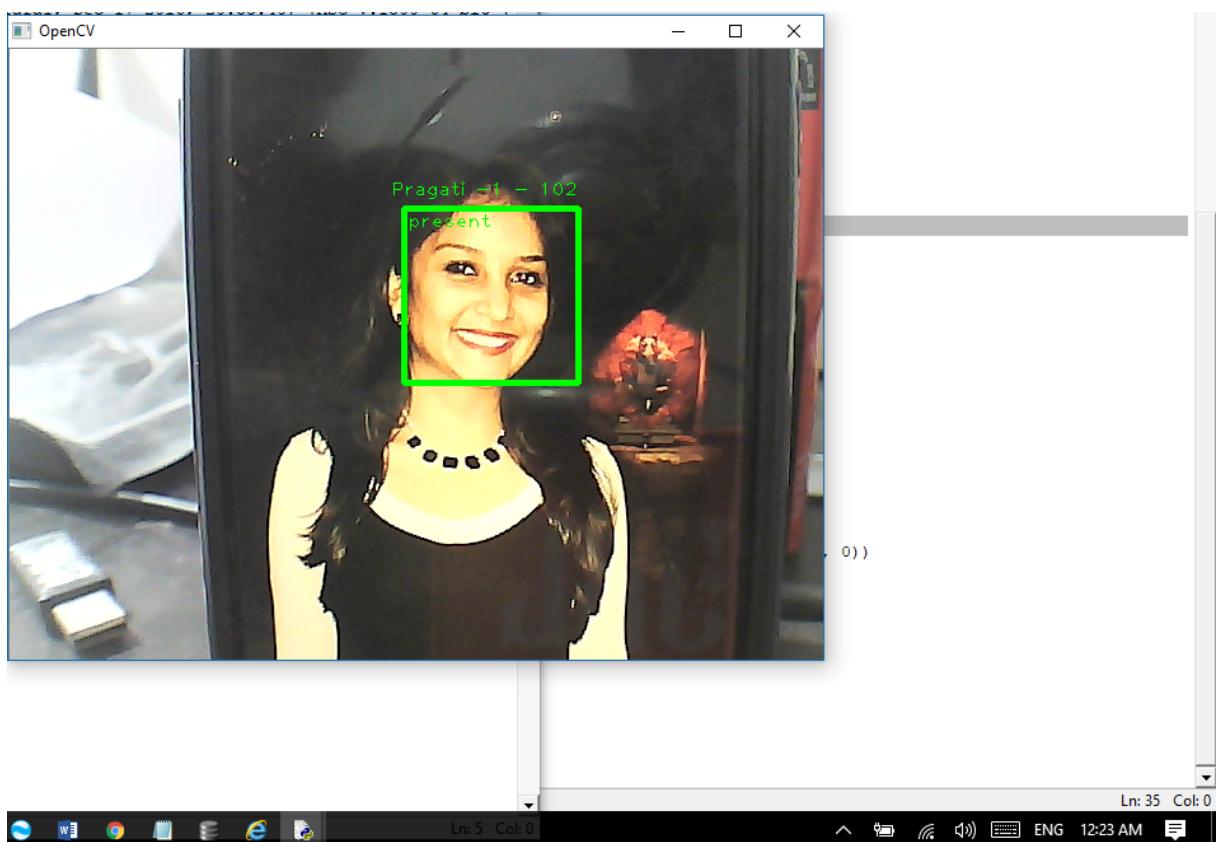


Fig 5. 3 shows the face detected and recognized by running the face recognition program.

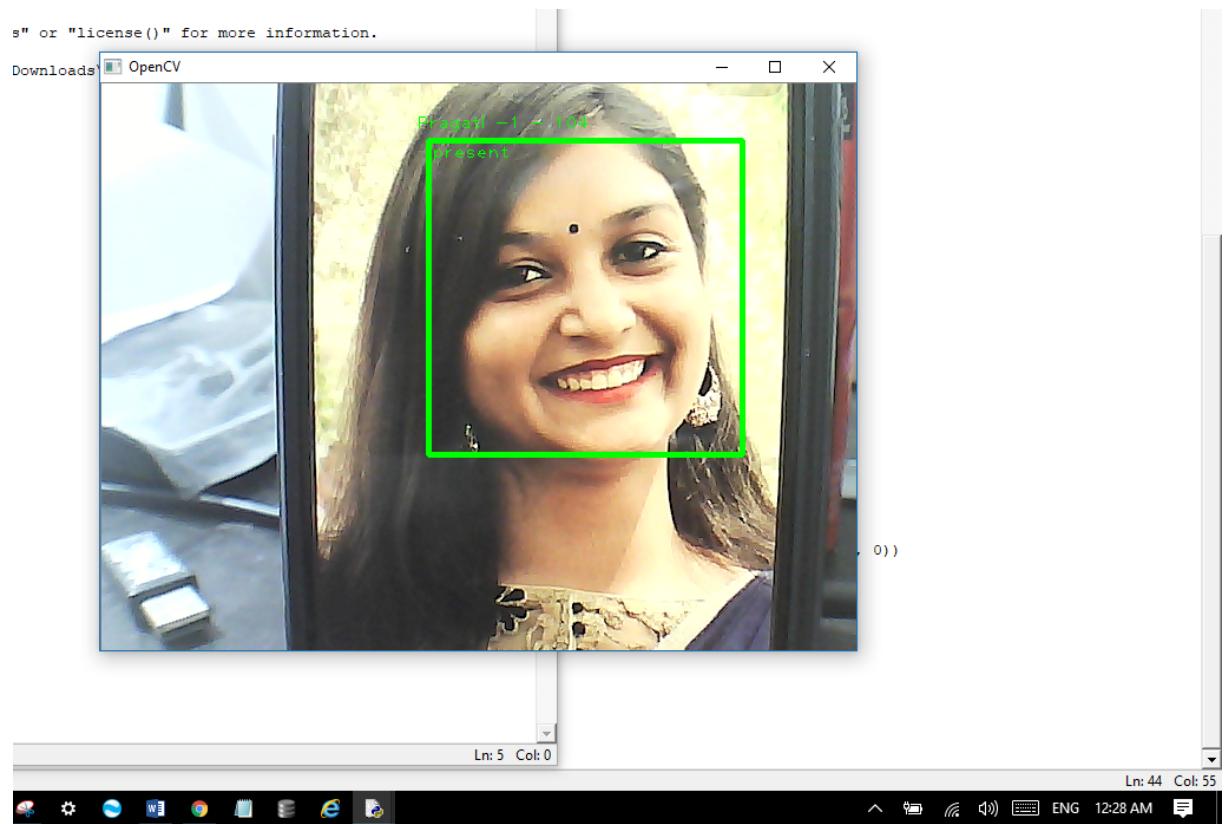


Fig 5.4 shows the face detected and recognized by running the face recognition program.

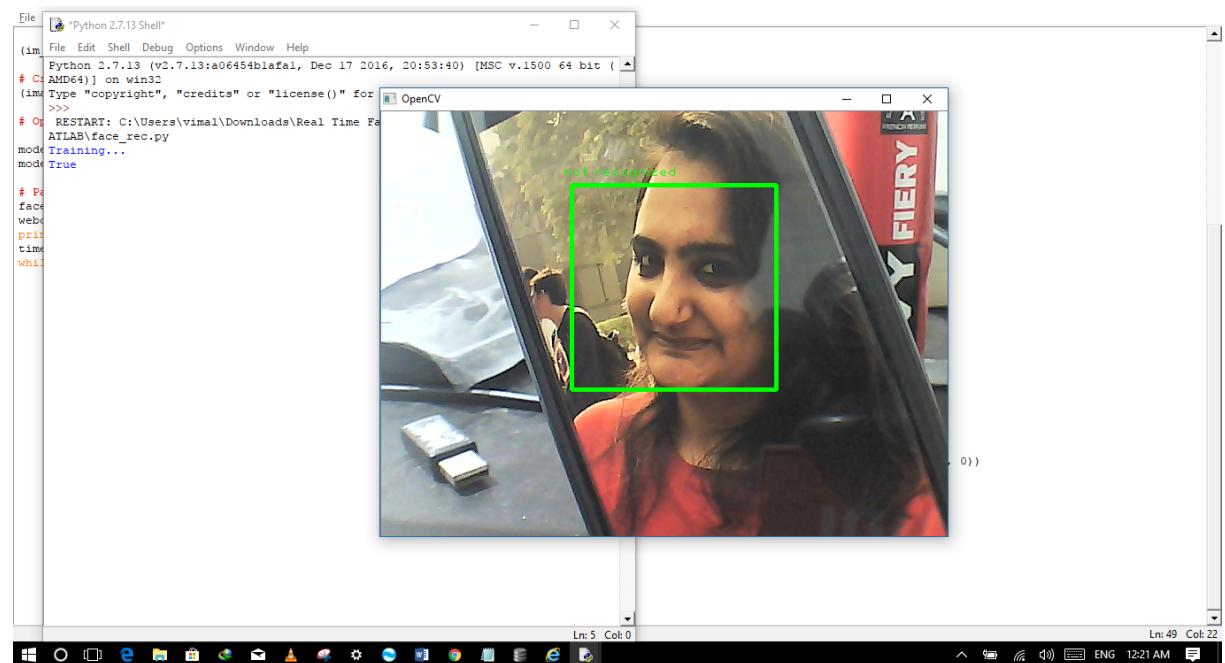


Fig-5.5 shows that the image is Not Recognized as the image was not stored and trained in database folder.

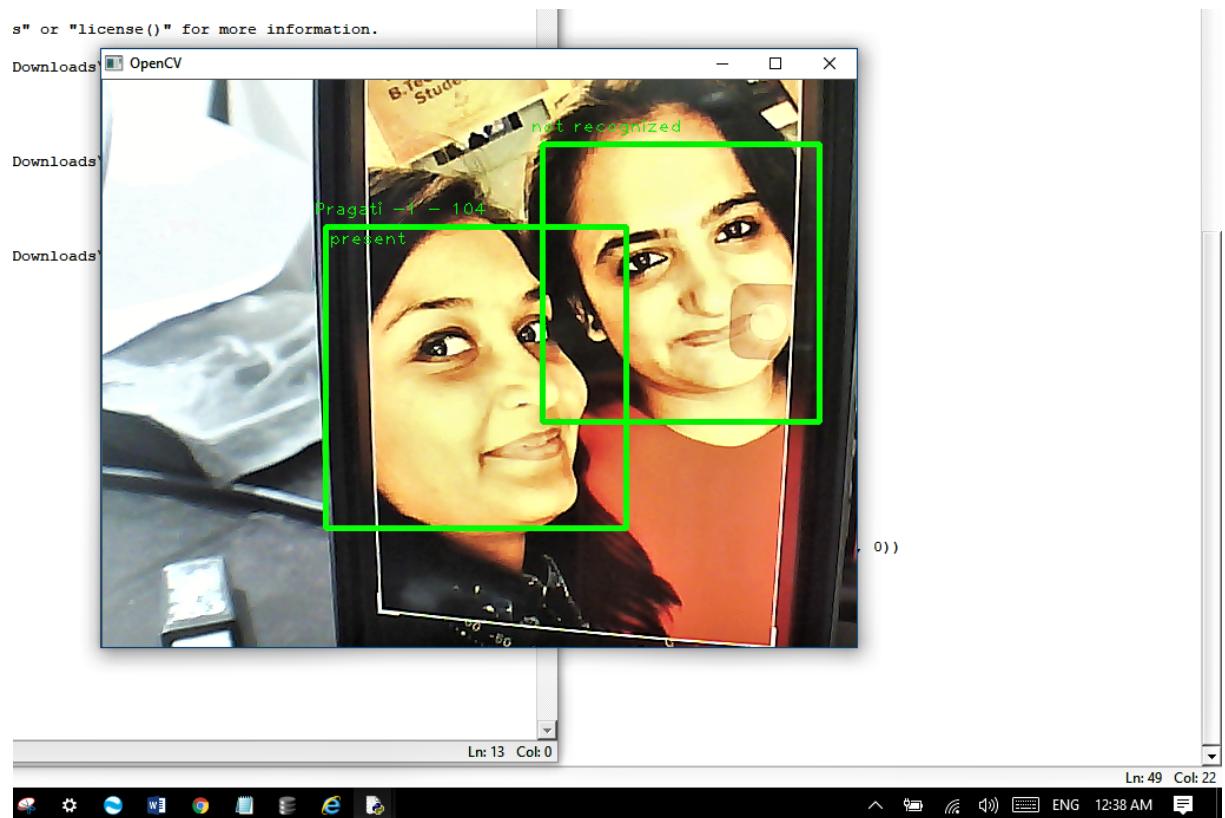


Fig 5.6 shows that one face is stored in the database folder and displays as present and name of the person and the second one's face is as not stored it is not recognizing..

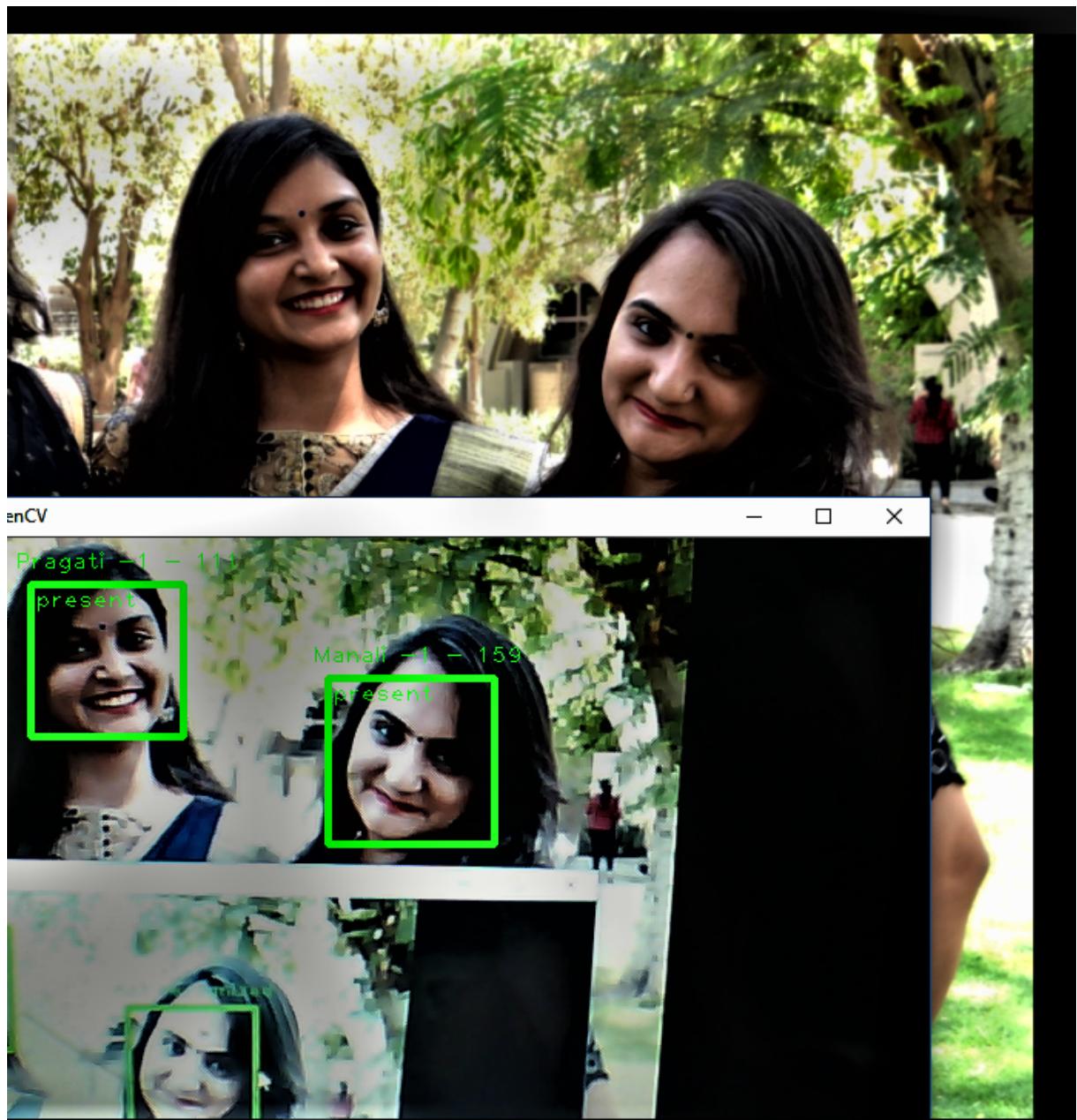


Fig 5.7 shows now both faces as recognized and present as the system identified the faces of both person from the database folder.

Chapter 6 Conclusion

Attendance Monitoring System using Face recognition program is far better than any other security system. In many other system many disadvantages such as contact system which means the body part must be in contact with the system to check for the security purpose. Whereas Face recognition system uses contactless system which means it does not require to be in the contact with the system. It checks from the distance. Face recognition system uses many techniques and types and algorithms with many advantages and disadvantages.

Chapter 7 References

- [1] (P. Viola & M. Jones (2001)), “Rapid Face Detection using Cascade of Simple Feature”, Conference on Computer Vision and Pattern Recognition. IEEE Press.
- [2] (Real time face recognition) <http://electronicsforu.com/electronics-projects/real-time-face-recognition-python-opencv>
- [3] (Hanzratech Face Recognition 2015)<http://hanzratech.in/2015/02/03/face-recognition-using-opencv.html>