Attendance System Using Face Recognition

A

Project Report Submitted in partial fulfillment of the

Requirements for the award of the degree of Bachelor of Technology

In

Computer Science & Engineering

By

Kashish Bajaj (161319)

Sidhant Sharma (161293)

Under the supervision of

Dr. RajinderSandhu

To

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology

Waknaghat, Solan – 173234, Himachal Pradesh

ittfdshshI

Candidate's Declaration

I hereby declare that the work presented in this report entitled **"Attendance System using Face Recognition"** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science &Engineering** submitted in the department of Computer Science & Engineering and Information Technology, Jaypee University of Information Technology Waknaghat, Solan is an authentic record of my own work carried out over a period From January 2020 to June 2020 under the supervision of **Dr. Rajinder Sandhu**(Assistant Professor, Department of CSE & IT).

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Sidhant

(Student Signature)

Kashish Bajaj, 161319

Sidhant Sharma, 161293

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

(Supervisor Signature)

Dr. RajinderSandhu

Assistant Professor

Department of Computer Science& Engineering

Dated: ..../June/2020

Acknowledgement

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of people whose ceaseless cooperation made it possible, whose constant guidance and encouragement crown all efforts with success. I am grateful to my project guide **Dr. Rajinder Sandhu** for the guidance, inspiration and constructive suggestions that helped me in the preparation of the project.

I also thank our colleagues who have helped me in successful completion of the project.

Kashish Bajaj

Sidhant Sharma

Table of Contents

List of Tables

List of Figures

Abstract

Uniqueness of a person’s face is the representation of his identity. In this project, face of person is used to mark attendance without human intervention. Attendance of the student is vital for college, university and school. Prior method of taking attendance is by saying name or roll number of the student and the attendance is marked. Assume that the time interval for one subject is around 1 hour and to record attendance takes 7-10 min. For every tutor this is overwhelming task. Face Recognition, an automatic procedure is used in this project which uses image processing. In this project Face Detection and Face Recognition is used. Face Detection is used to trace the position of face region and Face Recognition is used for marking attendance. The database of all the students is stored and when the face of the person matches with one of the faces stored in the database then the attendance is marked. This method is better than other old methods of taking attendance as it costs less time than other methods and it is difficult to mark false attendance. Moreover, this method could also be extended for use in examination halls to curb cases of impersonation as the system will be able to single out the imposters who won’t have been captured during the enrolment process.

Chapter 1
Introduction

1.1 Background

Attendance is an important measure in the aspect of both teacher and student for any institute or to an employer and an employee, and also they have to keep thee record of the same.

There are lot of issues in taking the attendance manually such as proxies and a lot of energy required to call out the name and roll numbers. So there is a need of the Automatic attendance which can solve all these problems.

Talking about the Automatic System it started from a system such as Biometric attendance i.e. attendance using the finger prints or the technique is RFID system. The only disadvantage of this system is that it takes a lot of time that one has to stand in the long waiting queue and also a source of the communicable diseases.

This project gives an idea of automatic attendance system platform that can also be used for purposes other than the attendance such as in exam sessions or other teaching programs. This can also remove the hassle and time consuming activities such as checking of ID cards etc. This can also help in taking the database of the students as well as staff up to date. This is will also help in registrations as it is done on the spot using facial Recognition

Face Recognition is difficult in daily life to identify people. Human intelligence systems allows us to get information and understand that in the recognition process. We receive information through the eyes of the human, especially retina is the main source of the information. Light act as an electromagnetic wave which helps us to recognize different objects.

After the processing is done by the human system, we classify shapes, sizes and the texture of the object to obtain the information. Once the information is studied the results will be compared and the face memories that exists in the database. It is difficult to create such advance system which can identify faces as clear as done by the humans. As we need large storage to store the recognized and assigned faces as there are number of students in schools, colleges, universities etc.

Every human face has a unique identity. Thus, in this method the real time face of the person is compared with the present in the database.

Nowadays, Face Recognition system is very useful due to its simple and high performance. This system is used in airports and also FBI use this system for investigations of the criminals. Facebook also implements Face Recognition to allow the users to tag other people in the photo for entertainment purposes. Furthermore, Intel Company allows the users to access their account using Face Recognition. Many mobile companies add this feature in their mobile phones as an unlock feature.

1.2 Problem Statement

Earlier the student attendance system face a lot of problems. This system helps the society by eliminating the earlier style of attendance marking process that includes the calling of name or roll no. of by checking the ID cards of the students. These older system cause disturbance in the teaching as 5-10 minutes from the class of 50 minutes wasted in this process also during examination this process hassles the students break their concentration which is needed at a that time. Circulation of the attendance sheets in class having large number of students takes a lots of time and distract the student and also result in proxies.

Thus, Attendance system using face recognition helpful to replace the manual system of taking attendance without being disturbed. Moreover, this system based automated attendance has ability to give the solution to the problem of proxies and improve the style of store data in an organized manner and also helps to reduce the human error.

The Face Recognition system takes time at initial steps i.e. it acts like one time investment and multiple returns. This system takes time during the training process i.e. the system has to be train to the face s of all the students once during registration which takes upto1-2 min per student.

Also we need to develop a real time attendance system and the cameras to be fitted in every classroom that must cost the institution at once and the lights are full effective or the other way is the quality of cameras to be high.

1.3 Objectives

The main objective of the project is to develop system that is able to mark attendance on the basis of Face recognition. The project is expected the fulfil features as listed below:

- To detect the face of the student
- To extract the features needed by the programme from the captured face.
- To classify the features as needed to recognize the face.
- To mark attendance in the database.

1.4 Scope of the project

We will design the system which compromises of two modules. The first module help us to start camera and start taking pictures of the class in order to capture faces and store them in the specific location using Computer Vision Face Detection Algorithm and other Face Features Extraction techniques. The second model comprises of the backend application in computer that help in recognizing the images captured at that time along with comparison, of the images with the images in database and mark the attendance.

Chapter 2
Literature Review

In this section we discussed about digital image processing and how we have used this in detecting

and recognizing faces. We also have discussion different methods of face recognition.

2.1 Students Attendance System

Firstly talking about RFID (Radio Face Recognition Frequency Identification) card system and fingerprint system have various negative effects. This card system is implemented because of its simple and straight features. In face recognition system one can misuse it by using their Face Recognition end’s identity card. The system of marking attendance using finger patterns is much secure but the only disadvantage is that is time taking process. As this is face recognition it is less secure and informative as the main information is found in one’s iris. But we cannot use Iris Recognition system as it is again privacy rules. Other option is using voice but that is least secure as well as contains many errors. So keeping all the things in mind from security purpose, privacy issues as well as fast process face recognition is the best and is suggested to mark attendance.

Table 2.1: List of Advantages & Disadvantages of Different Biometric Systems

Systems	**Advantage**	**Disadvantage**
Radio Face Recognition Frequency Identification card system	Simpler and Straight forward	Face Recognition audient use

Finger pattern system	Accuracy	Time taking process
Voice Recording System	Accuracy	Less accurate as compared other systems
Iris recognition System	Accuracy	Priority Inversion

2.2 Digital Image Processing

This process involves the image processing of digital images in computers. This method consists of three major applications that are as follows:

- This helps in improving of picture details for human understanding.
- This is used for image processing for different autonomous machine applications.
- This also contains storage for resources and transmission.

2.2.1 Representation of image in Digital Computer

Image can be represented with the help of 2-Dimensional light intensity function

$fn(a ,b)=r1(a ,b)\times i1(a ,b)$-(2.0)

In the above equation r1 (a, b) is defined as surface reflectivity of the given image point. i1 (a, b) is the measure of strength of the incident light on the image. The Digital image fn (a, b) is described in both the special co-ordinates by the grids and the quantization brightness. For an effective we can also represent the image in matrices form in which rows gives the point of the image and columns gives us gray level values at that point of the image. These elements are also known as pixels opels.

The image size used by given processing application if approx. 256*256, and the size of the elements is of 640*480 pix. The main process of the quantization for the matrices is done using 8bits for the black and white image and for color image it uses 24 bits as 8 bits is used for each plane i.e. RGB.

2.2.2 Steps involves in Digital Image Processing

The steps of digital image processing is given as follows:

- Acquisition of image – The imaging sensor along with the ability to convert signal that is produced by the sensor.
- Preprocessing of image – It helps in increasing the quality of the image.
- Segmentation of image– It also helps in partitions of image and input of the image into various Parts of objects.
- Recognition and Interpretation of image– It helps in assigning a label to the object which is based on the information that is provided by its descriptor.

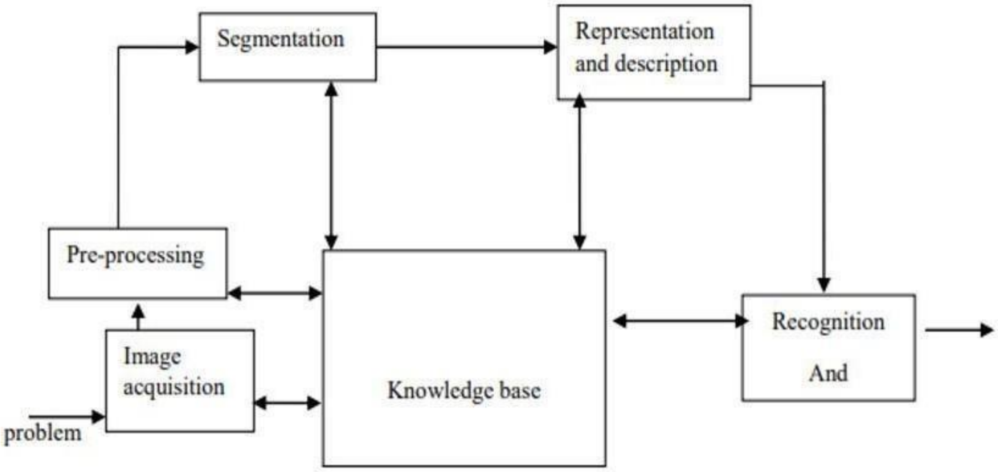


Fig. 2.1 Digital Image Processing Steps

The above diagram shows that how Digital image processing works and its proper steps

2.3 Definition of Terms

2.3.1 Face Detection

Detecting of face is the process that involves detecting and locating the faces that are present in the image and differentiate in terms of expression or age. Moreover this detection is able to function without the interference of the light conditions.

2.3.2 Face Recognition

Recognition of the face is system that involves recognize patterns in the image, the face is represented in 3 Dimensional with different types of light conditions, pose and other factors

That may be helpful to get best results. This process is mainly to check whether the real time captured face is already present the database or not.

2.3.3 Difference between Face Detection and Face Recognition

Detecting of Face is the process in which the program runs to check that weather the captured image is already present in database or not. Whereas Recognizing of Face is the process in which the program runs to check that whom image is being captured in the real-time and all the details from the database. The Face Recognition’s output is the final decision i.e. face known or face not known.

2.3.4 Method of Face Detection

The main aim of face detector is to check whether the image capture has any of human face if the human face is presented in the image then the face detection process checks various conditions to classify the image and extract various features such as skin of the face. Shape of the head and combine these result for future use. Many face detecting algorithms works without use of the parameters. Major algorithms involves Viola Jones object detection Face Recognition framework, which is based on Haar Cascades.

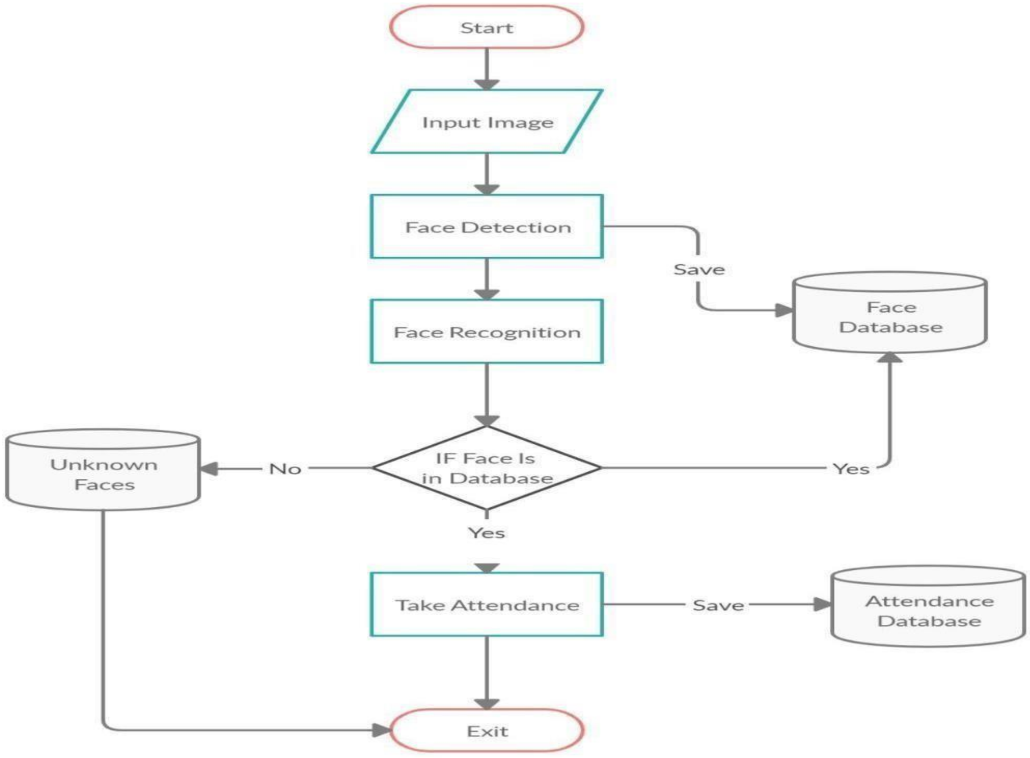


Fig. 2.2 Flow Chart

This Flow chart gives us proper information that how attendance is marked using face recognition.

Table 2.2: List of Advantages & Disadvantages of Face Detection Methods

Face Detection Method	**Advantages**	**Disadvantages**
Viola Jones Algorithm	1. Detection speed is high 2. Highly accurate	1. Training time is more 2. Head pose conditions are less 3.Not able to detect dark faces.

Local Binary Pattern Histogram	1. Simpler calculations 2. Tolerance power is very high	1. Has only been used for grey and binary images. 2. Not good as Viola Jones
Ada Boost Algorithm	No prerequisite needed.	Training time is more
Features and SNOW Classifier Method	1. Has ability to use only faces from the image 2. Fast Calculations	The empty image also being considered
Neural-Network	Accuracy is more is clarity is there.	1. Process of detecting faces is slow and difficult. 2. Not good as Viola-Jones

The given table differentiates between various Face Detection techniques.

The best among above mentioned Viola-Jones algorithm was firstly introduced by P. Viola and M. J. Jones in 2001and this is the most popular algorithm to be used as face detecting and recognition. The base of the Viola-Jones algorithm has 4 parts.

The first part of the algorithm is Haar feature and 2nd part is that where integral image is created, and the third part involves the implementation of Adaboost and last part involves the process of cascading.

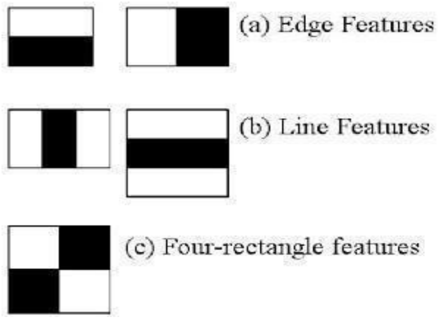


Fig. 2.3 VariousHaar Feature

This figure gives information about different Haar Features.

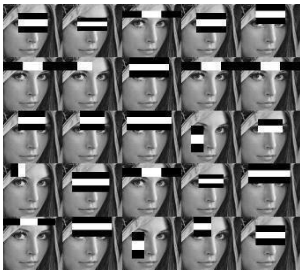
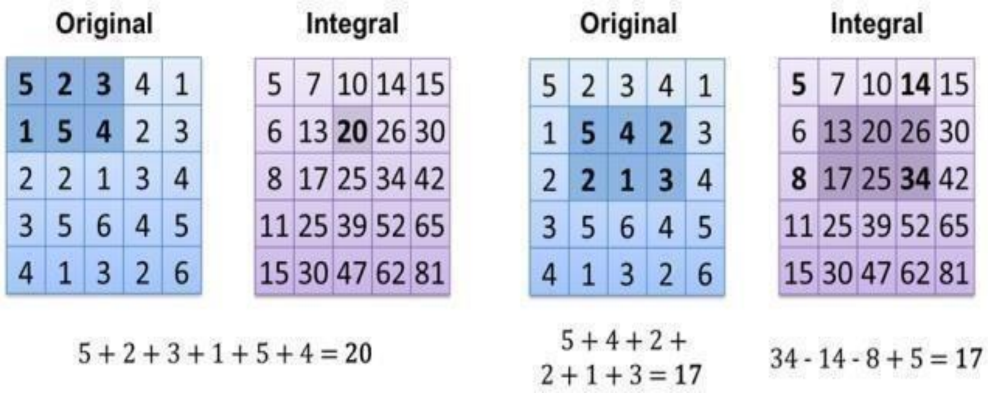


Fig. 2.4 Haar Features

This figure gives information how image is being recognized on basis of Various Haar features.

his algorithm helps us to analyze the image with the help of Haar features that consists of many

rectangles. In the given figures there is example of various type of Haar features.



Original

5	**2**	**3**	4	1
1	**5**	**4**	2	3
2	2	1	3	4
3	5	6	4	5
4	1	3	2	6

Integral

5	7	10	14	15
6	13	**20**	26	30
8	17	25	34	42
11	25	39	52	65
15	30	47	62	81

5 + 2 + 3 + 1 + 5 + 4 = 20

Original

5	2	3	4	1
1	**5**	**4**	**2**	3
2	**2**	**1**	**3**	4
3	5	6	4	5
4	1	3	2	6

Integral

5	7	10	**14**	15
6	13	20	26	30
8	17	25	**34**	42
11	25	39	52	65
15	30	47	62	81

5 + 4 + 2 +
2 + 1 + 3 = 17

34 - 14 - 8 + 5 = 17

Fig. 2.5 Integral of Image

The value calculated by integrating of the images in the specific location is calculated as the sum of the pixels on the left side and the top of the location of the image. For more detailed understanding the integrated value of the image calculated at location one is the sum of the pixel of the rectangle A present in the image. The value for other locations will be calculated cumulatively. As Value calculated at location two is equal to value at one plus value at two i.e. (A+B) and value at location three is equal to (A+B+C) and so on.

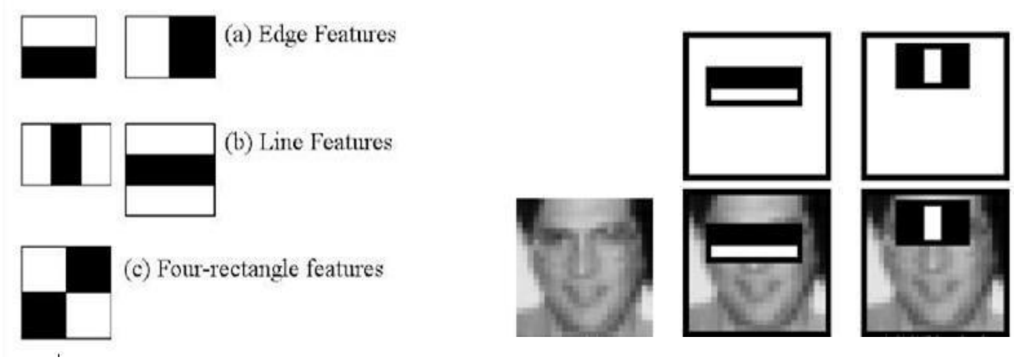


Fig 2.6 Haar Feature Example

For proper classification of image and to increase the accuracy of the result all the features are applied while training of the image. While recognizing the features that has least error is considered that helps us to get maximum accuracy and it also helps us to give the best classification of images containing face or not.

Finally the results calculated by taking the sum of all the groups. Each group taking along is termed as a weak classifier while all these are grouped together then this result in strong classifiers. The accuracy of the result is up to 97%. Number of features on which image was trained is approx.6000.

As to calculate results using all the features, these features are combined in number of groups and then the stages of classification started by using one group at a time. If there is an error occurred in earlier stage then the classification stops right there and do not move further.

All the 6000 features are combined to form approx... 40 groups that is result is classified in 40 stages, first five groups contains one, ten, twenty five, twenty five and fifty features. According to research approx. 10features are calculated per sub window.

2.3.5 Local Binary Patterns Histogram

LBP (Local Binary Pattern) is a system that helps in labeling the pixels of an image by off putting the pixels of each neighbored and results in binary format. This is very simple to use and very well organized function. The very first description of LBP is given in year 1994 and is very helpful for the classification of the text. It combines with histogram and forms it HOG (Histogram oriented gradient descriptor). It helps in removing the appearance on some data sets. While combining of LPB and the histogram we are able to give the representation of images of face with a data vector.

2.3.6 Transfer learning

It is the process that mainly works in storing the previous results and train itself by the knowledge gained by it and help to solve more similar problems as fast as the system was already trained. The concept of transfer learning is under research under Machine Learning technology which can be more widely used further. For e.g. If the machine is trained to classify cars in an image it can also be helpful to classify other vesicles. This concept of somewhere connected with the history of psychological literature.

The concept of transfer learning can be called as a subset of the machine learning as it is the feature that it can train itself with the help of the previous results. For example, the image recognition model is a subset of Inception-v3 which consists of two parts as given below:

- The extraction of features from the image with the help of convolution neural network.
- Classification of the extracted features of image using softmax layers.

The concept if inception model has achieved very accurate results for the recognition of the images it also works on pre training phenomenon. This model firstly extract the main features of the image and then classify into different categories.

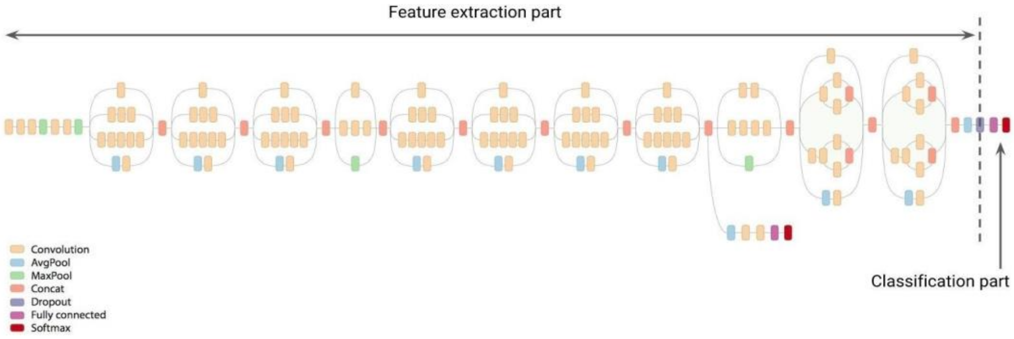


Fig 2.7 Transfer Learning Result

In this process whenever the new model is to be build the model start training itself using previous results in order to give most accurate results in the least time units. The training of the model also takes less time.

2.3.7 Triplet loss

It is a function used to calculate loss and is used in artificial neural networks. This function firstly compares the base line inputs with both positive and truthy input as well as to negative or falsy input after that triplet loss function activated and help to minimize the distance of base from positive side and maximize it from the negative side to get the better and accurate results.

It is also used as to learn the similarity. For e.g. word embedding, metric learning etc. This loss function can also be called as the Euclidean distance function as it calculates the Euclidean distance formulae.

This function can also be used as a cost function to calculate the sum of all the losses, and further used for minimizing the loss and maximizing the profits.

The index given to each input is considered as a triplet. The triplet input consists of the anchor or base line input, the positive input and the negative input. This triplet then sent to a network function and the results obtained from that is used in the triplet loss function.

In the field of computer vision it believes that the results of this function is not too accurate and many researchers obtain that the pre trained model is somewhat vary results with the training of models from the scratch and the method which is fully combined with deep learning of this function has released in the year 2017.

2.3.8 Imagenet

Image net is an image which is prepared with the help of word net. Word net is describes as number of words which is called as synest. There are a lot of synest present in the word net approximately around 100000 most of the synest are nouns and to describe one synest there are average 1000 ImageNet given.

Fig 2.8 Image Net example

2.3.9 Inception model

Inception Modules are used in Convolutional Neural Networks to allow for extra well-organized Computation and deeper Networks through a dimensionality cutback with stacked 1×1 convolutions. The modules were planned to solve the problem of computational expense, as well as over fit, amid other issues. The solution is to take several kernel filter sizes within the CNN, and rather than stacking them in a queue, order them to do functioning on the same level.

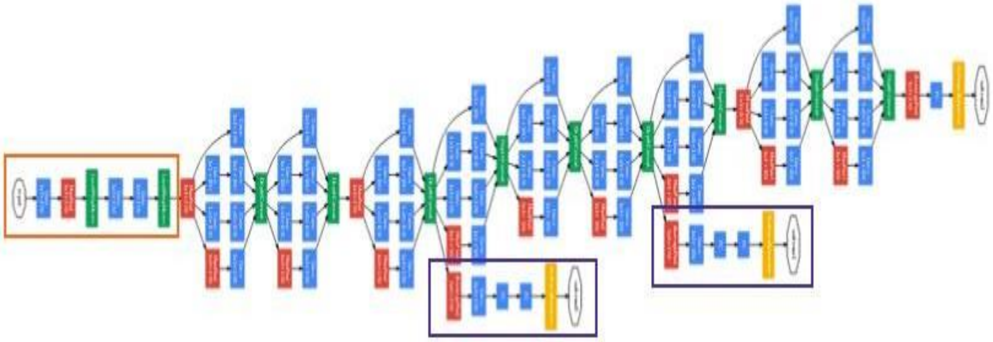


Fig 2.9 Inception Model

2.3.10 working of inception network

Inception models are the part of CNN (Convolution neural network).

The basic function of the inception network is to inspect the results into as deeper as it can in order to get more and more accurate results it performs the operations in three different size i.e. in 1*1 3*3 and 5*5. This process is helpful in calculation the most accurate results as it takes all the conditions and then compute the final result but the n of the disadvantage is that it is a little expensive.

In order to get these accurate results in cheap rates the network hast to add the extra layer of 1*1 convolution before the computation of the 3*3 as well as 5*5 layers.

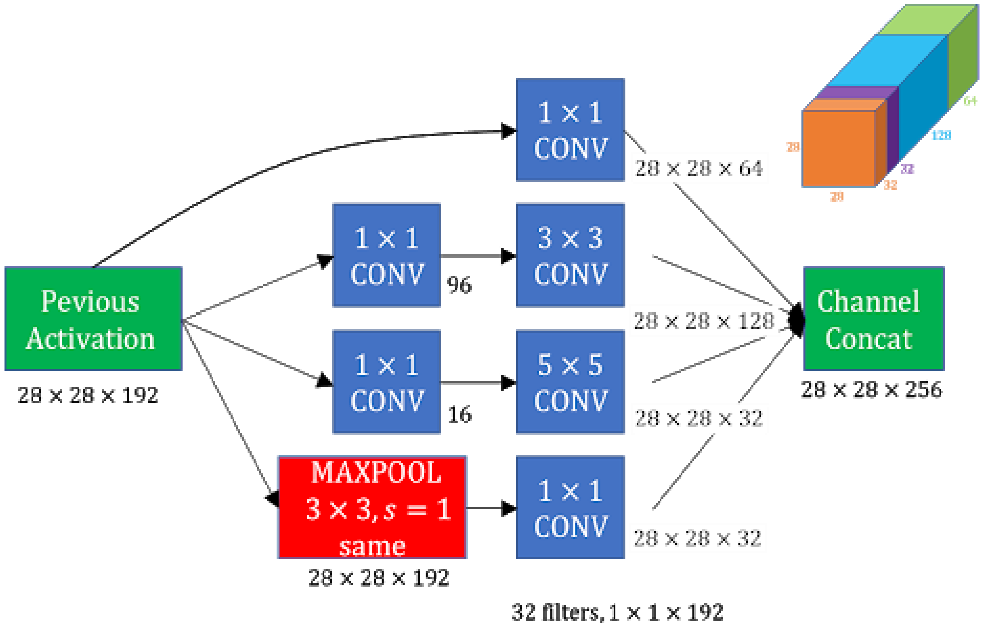


Fig 2.10 Inception Network explanation

Chapter 3

Proposed Framework

3.1 Introduction

The technique of face detection contains the organization out of the image into two different course: One of the course contains faces by turning the background into clutter. This process is difficult as there is very vast type of data involved in it as age, color of the skin etc. This process become more difficult in next step as it has to differ taking more conditions such as traits of the image or in terms of the geometry or there may be the chance of the incomplete cover. An ideal Face detection algorithm or the system has ability to capture and detect the face presence under many conditions and having any type of background provided.

The process of Face Detection can be done as follows:

Firstly the random images is taken as inputs and output gives us a binary value 0 or 1 i.e. No or Yes that gives us value of presence of face. Second step involves the localization of the face that takes an image as input and gives us output the position of faces present in that image. The output given is in the form of (x, y, Width, Height) of the image.

After taking picture the algorithmic system will compare the equality of the pictures in its database and give the most linked result. We will use Windows operating system, open CV platform and will do the coding in python language.

3.2 Model Implementation

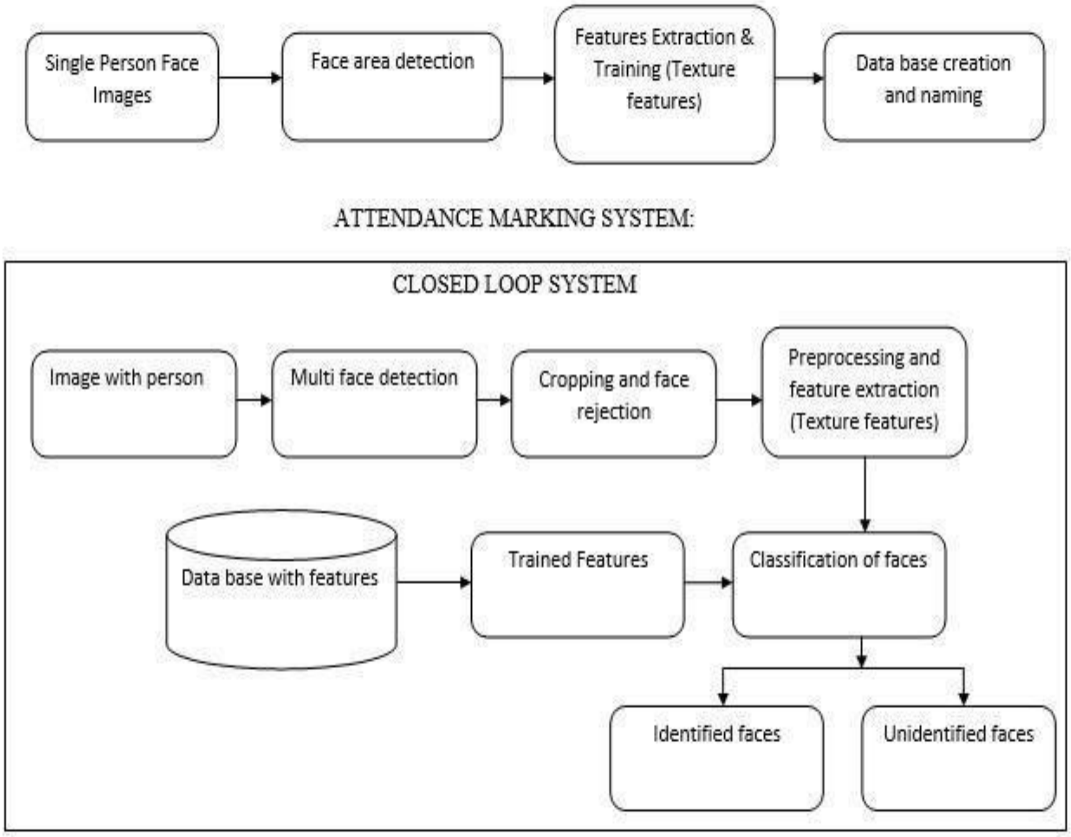

Single Person Face Images
Face area detection
Features Extraction & Training (Texture features)
Data base creation and naming
ATTENDANCE MARKING SYSTEM:
CLOSED LOOP SYSTEM
Image with person
Multi face detection
Cropping and face rejection
Preprocessing and feature extraction (Texture features)
Data base with features
Trained Features
Classification of faces
Identified faces
Unidentified faces

Fig. 3.1 Model Implement

The main mechanism used to implement the process is using library that is OpenCV (open source computer vision library). This library has a goal to provide a simpler version of computer vision in recognition of the faces. This structure is helpful for users to get fairly refined applications at a very fast pace. The library has more than 600 Functions than can be used for various purposes. The fundamental technology used behind Recognition of face is OpenCV. In this the user has to stand in front of the camera at a distance that is less than 60cm and the image is being captured and sent as a input to the library. Only the facial features is extracted from the image and that extracted features then converted into gray scale image and get stored. Then the functioning of the PCA(Principal Component Analysis) algorithm is started which convert the stored gray scale image into the Eigen value and after that store it that values in the xml file. When any user uses it after training the same process occurs but rather than storing the calculated Eigen value is now compared to the values present in the database.

3.3 Software Used

This system is built with the help of number of tools which provide us platform to run our algorithms or store data or the functions for the front end services some of the main tools are as follows:

OpenCV: We used OpenCV 3 dependency for python 3. OpenCV is nothing but a python library which is used for processing of image there are many predefined functions that can be used to perform various operations on the image. There are such number of programs that we can get expected results with the help of inbuilt functions and nothing else. This library can be used as an open source with no cost. Example of some supported functions are given below:

- **Derivation**: It gives us the gradient or laplacian computing.
- **Hough transforms:** It gives us lines, segments, circles, and geometrical shapes detection
- **Histograms**: It gives us computing, equalization, and object localization with back projection algorithm
- **Segmentation**: It gives thresholding, distance transform, foreground background detection, watershed segmentation
- **Filtering**: It gives us linear and nonlinear filters, morphological operations

- **Cascade detectors**: It gives us detection of face, eye, car plates
- **Interest points**: It gives us detection and matching
- **Video processing:** It gives us optical flow, background subtraction, camshaft (object tracking)
- **Photography**: It gives us panoramas realization, high definition imaging (HDR), image in painting

Python IDE: There are lots of IDEs for python. Some of them are PyCharm, Thonny, Ninja, and Spyder etc. Ninja and Spyder both are very excellent and Face free but we used Spyder as it feature- rich than ninja. Spyder is a little bit heavier than ninja but still much lighter than PyCharm. We used python idle and installed necessary libraries.

Microsoft Excel: This is a kind of spreadsheet that is developed by the Microsoft for all kinds of operating systems. The main features of MS Excel is that it contains various mathematical functions that can help us in calculations and there are lot of tools that can be used to plot graphs in different forms that helps us to analyze data and many other tools such as pivot tables, programming language named Visual basics for various applications.

3.4 Experimental Results

The step of the experiments process is given below:

3.4.1 Face Detection process:

This turn on the camera and started capturing images

```
    Begin:
```

- Preprocessing and extraction of features is done using image of the face.
- Eigen value is then calculated and compared with the values present in the database.
- If the value that is calculated in real time does not match with any of the values present in database then it saves it as a new face image to existing database.
- If the value that is calculated in real time matches with existing database values then this step will be done.

End

3.4.2 Face Recognition process:

This process of recognition of faces is done using inception network algorithm
The steps for face recognition are as follows:

Begin:

- The face information is matched with the information in the database.
- The log table will be updated with matched face and the system's time.

End

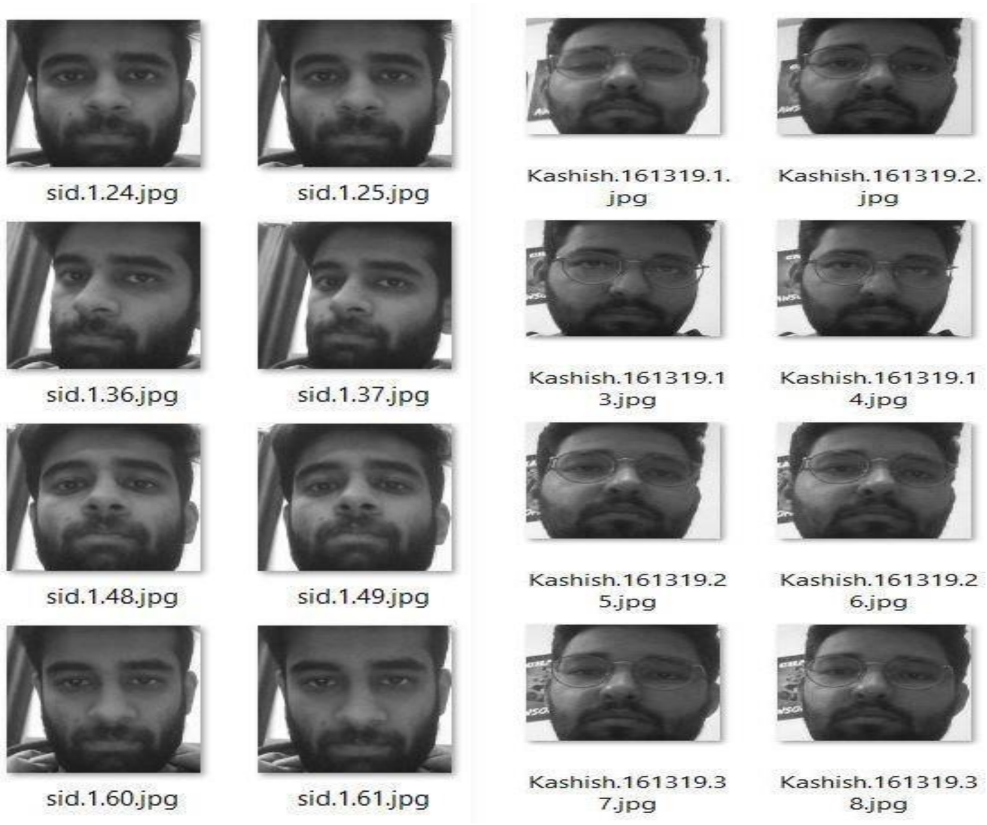


Fig. 3.2 Dataset sample

The diagram above represents the results of the function that is used to capture face into a grey scale image of 50x50 pixels.

Table 3.1: Accuracy Measure

Orientations of Face	**Rate of Detection**	**Rate of Recognition**
0º	98.6%	95.2%
19º	81%.	79%
55º	59.5%	58.3%
73º	0.01%	0.01%
90º(Profile face)	0.03%	0.03%

We have taken a set of different experiments to illustrate that the method is how much accurate.31 number of images of 11 people are used in training set.

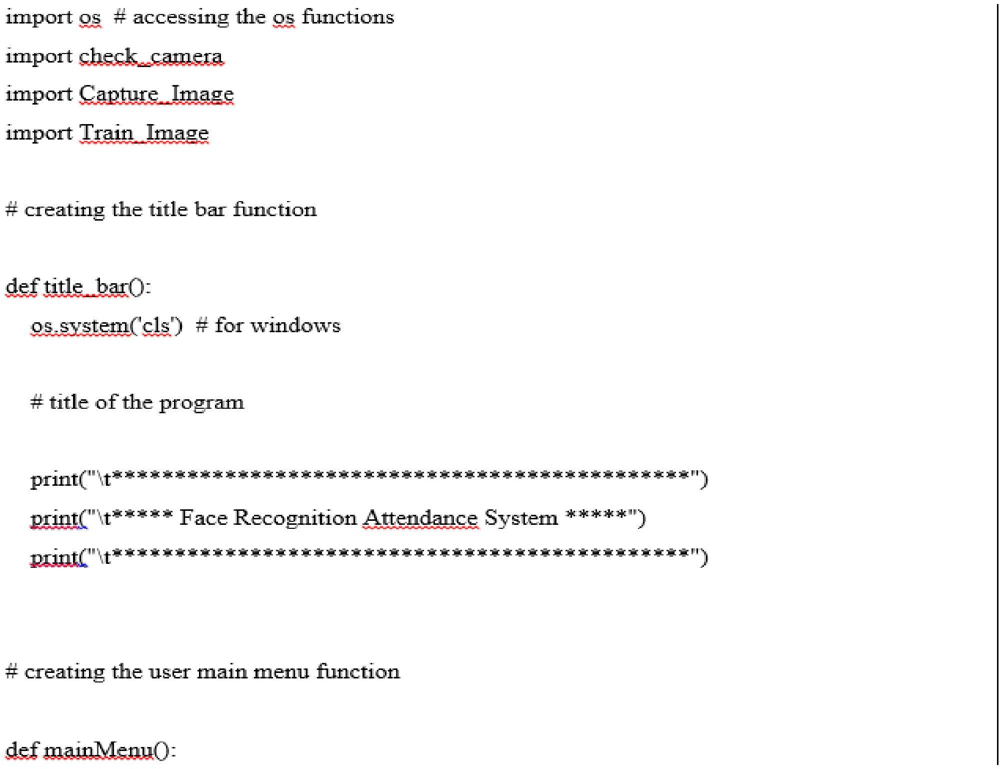
Chapter 4
Code Implementation

This chapter includes the different codes for various features and functioning of the project along with the outputs figures captured in running environment

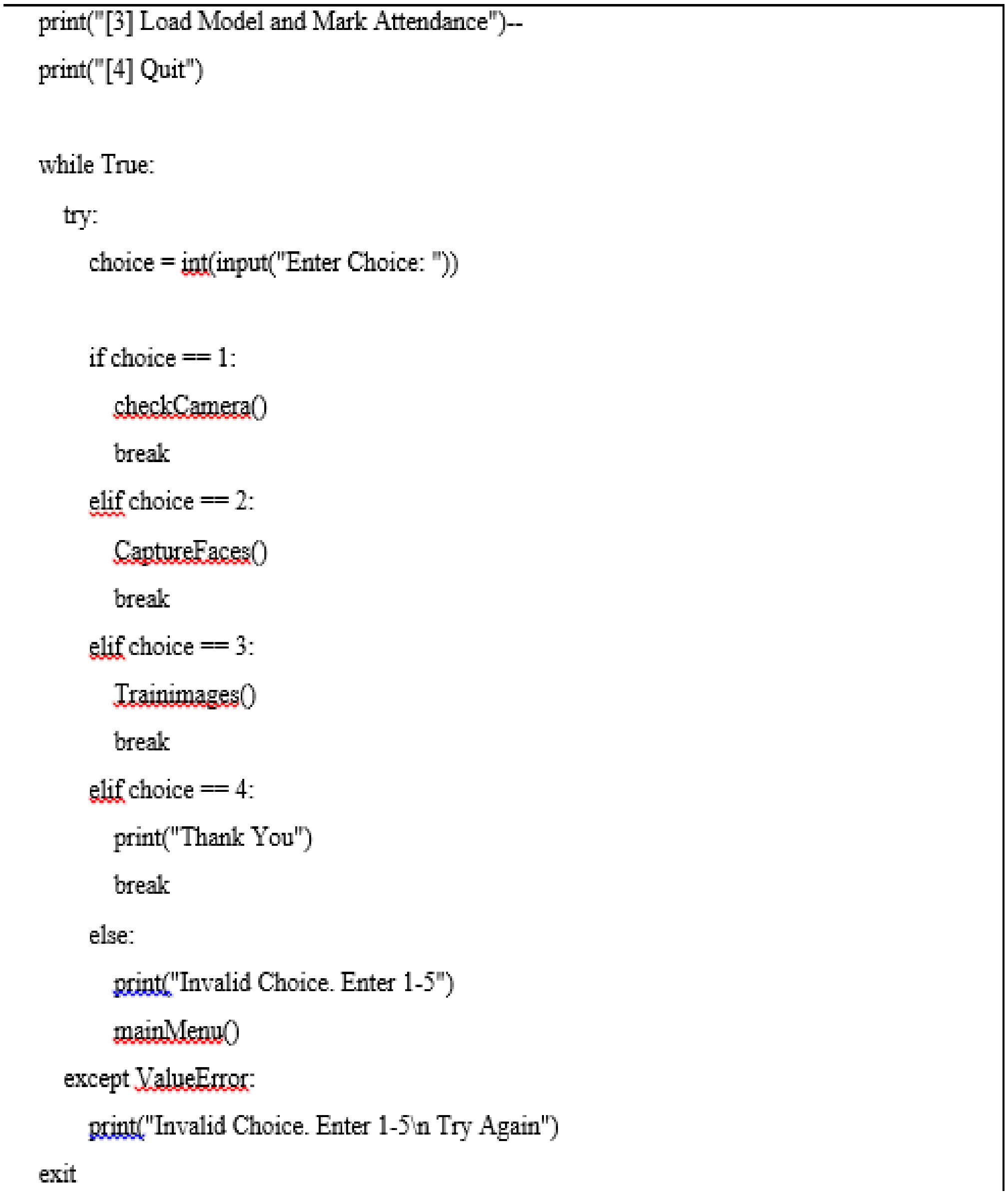
4.1 main.py

This file is used to create the basic framework of project. A user menu is created in this file, so that user can call all the other functions from a single file. User can select up to four options which are: check camera, capture faces, load model and mark attendance and quit. Each option calls another function which is implementing in different python file. Menu is implemented by using a simple while loop, so that the functionalities can be used again and again.

Code:-

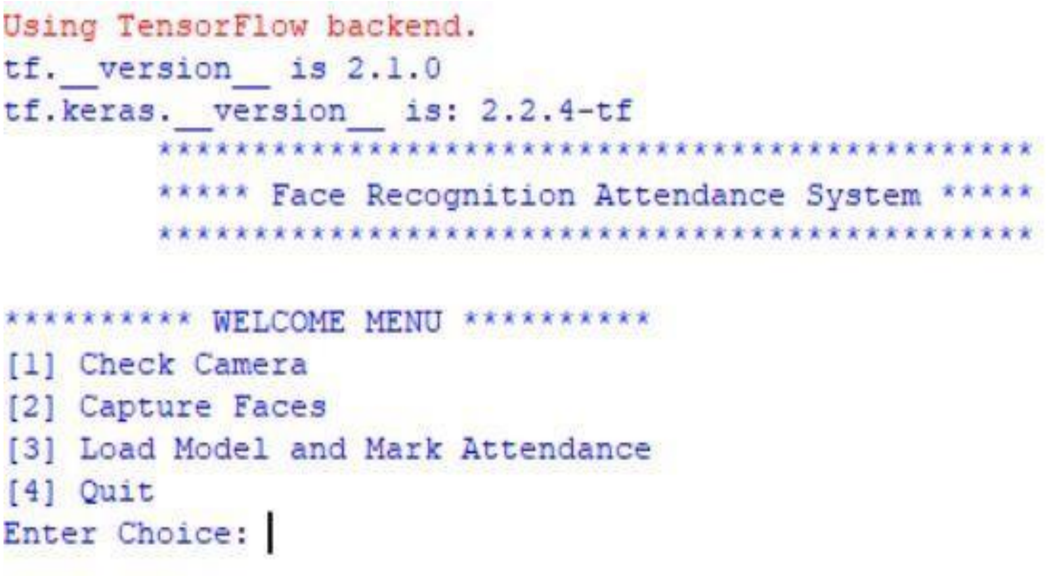


```
import os  # accessing the os functions
import check_camera
import Capture_Image
import Train_Image

# creating the title bar function

def title_bar():
    os.system('cls')  # for windows

    # title of the program

    print("\t************************************************")
    print("\t***** Face Recognition Attendance System *****")
    print("\t************************************************")


# creating the user main menu function

def mainMenu():
```



```
print("[3] Load Model and Mark Attendance")--
print("[4] Quit")

while True:
    try:
        choice = int(input("Enter Choice: "))

        if choice == 1:
            checkCamera()
            break
        elif choice == 2:
            CaptureFaces()
            break
        elif choice == 3:
            Trainimages()
            break
        elif choice == 4:
            print("Thank You")
            break
        else:
            print("Invalid Choice. Enter 1-5")
            mainMenu()
    except ValueError:
        print("Invalid Choice. Enter 1-5\n Try Again")
exit
```

```
def checkCamera():
    check_camera.camer()
    key = input("Enter any key to return main menu")
    mainMenu()

def CaptureFaces():
    Capture_Image.takeImages()
    key = input("Enter any key to return main menu")
    mainMenu()

def Trainimages():
    Train_Image.TrainImages()
    key = input("Enter any key to return main menu")
    mainMenu()

# ---------------main driver ------------------
mainMenu()
```



```
Using TensorFlow backend.
tf.__version__ is 2.1.0
tf.keras.__version__ is: 2.2.4-tf
        **********************************************
        ***** Face Recognition Attendance System *****
        **********************************************

********** WELCOME MENU **********
[1] Check Camera
[2] Capture Faces
[3] Load Model and Mark Attendance
[4] Quit
Enter Choice: 
```

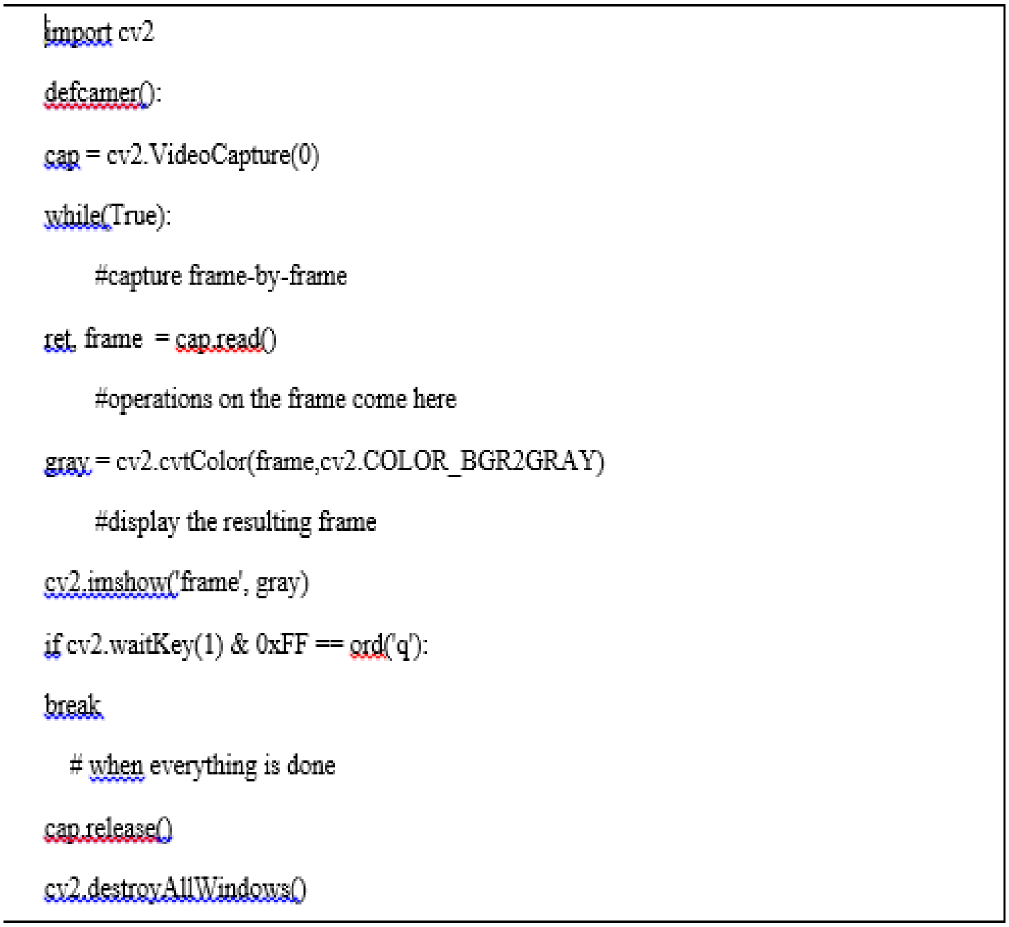
Fig 4.1 Output of main.py

4.2 check_camera.py

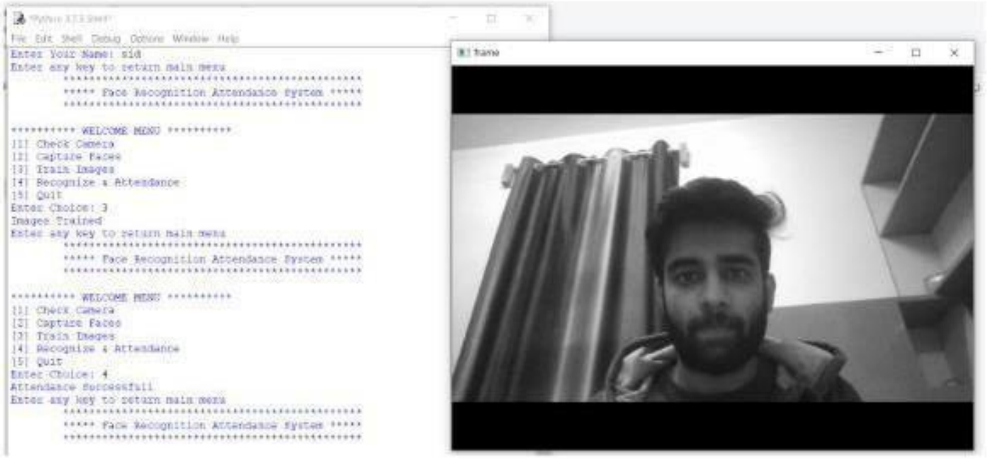
This file is used to check whether camera is working or not. OpenCV is used to implement major part of function.

A while loop is used to click continuous images until user presses ‘q’ to quit. An object of open CV library is used to read the images. Images are captured from the front camera of the laptop. The whole process is done only to check whether the camera of a device is working or not.

Code:-



```
import cv2
defcamer():
cap = cv2.VideoCapture(0)
while(True):
    #capture frame-by-frame
ret, frame  = cap.read()
    #operations on the frame come here
gray = cv2.cvtColor(frame,cv2.COLOR_BGR2GRAY)
    #display the resulting frame
cv2.imshow('frame', gray)
if cv2.waitKey(1) & 0xFF == ord('q'):
break
  # when everything is done
cap.release()
cv2.destroyAllWindows()
```



```
Enter Your Name: sid
Enter any key to return main menu
        ********************************************
        ***** Face Recognition Attendance System *****
        ********************************************

********** WELCOME MENU **********
[1] Check Camera
[2] Capture Faces
[3] Train Images
[4] Recognize & Attendance
[5] Quit
Enter Choice: 3
Images Trained
Enter any key to return main menu
        ********************************************
        ***** Face Recognition Attendance System *****
        ********************************************

********** WELCOME MENU **********
[1] Check Camera
[2] Capture Faces
[3] Train Images
[4] Recognize & Attendance
[5] Quit
Enter Choice: 4
Attendance Successfull
Enter any key to return main menu
        ********************************************
        ***** Face Recognition Attendance System *****
        ********************************************
```

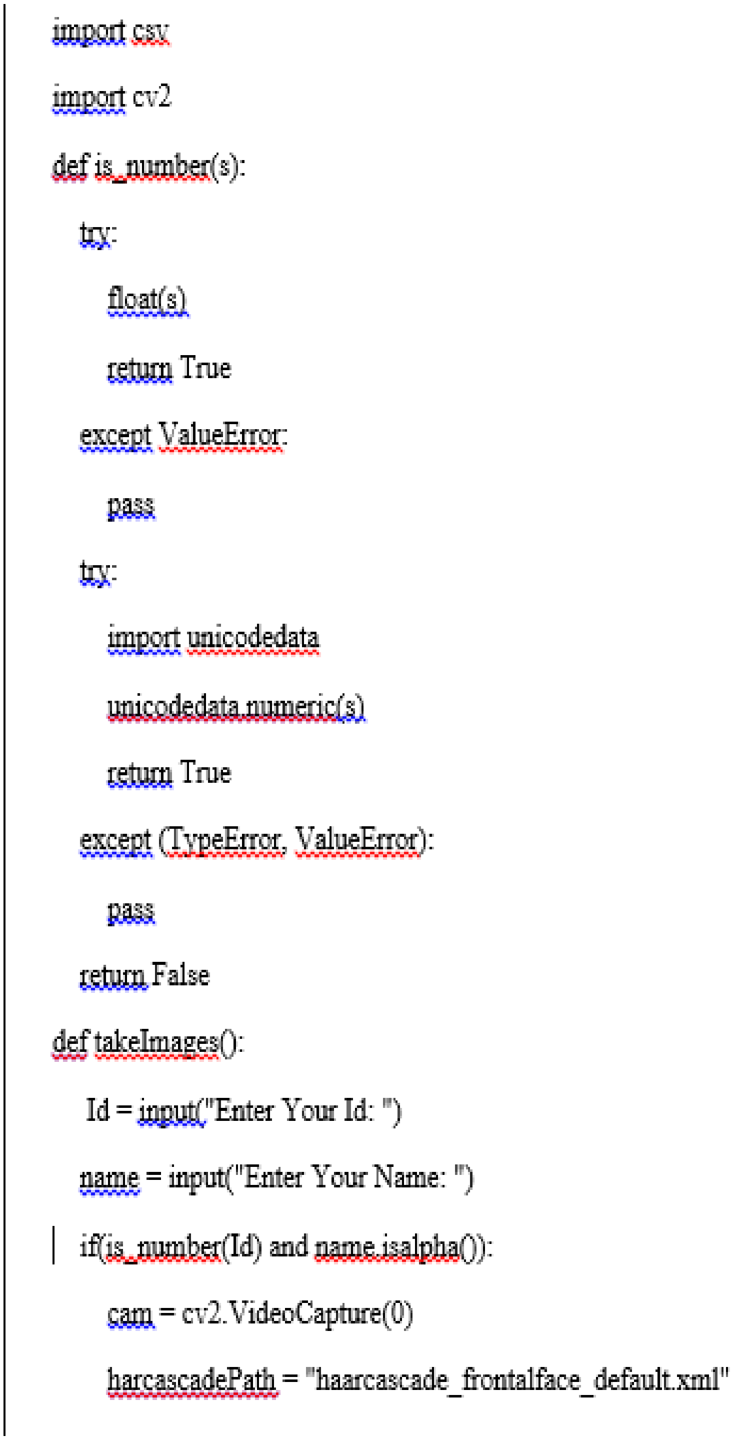
Fig. 4.1 Output of check camera.py

4.3 Capture_image.py

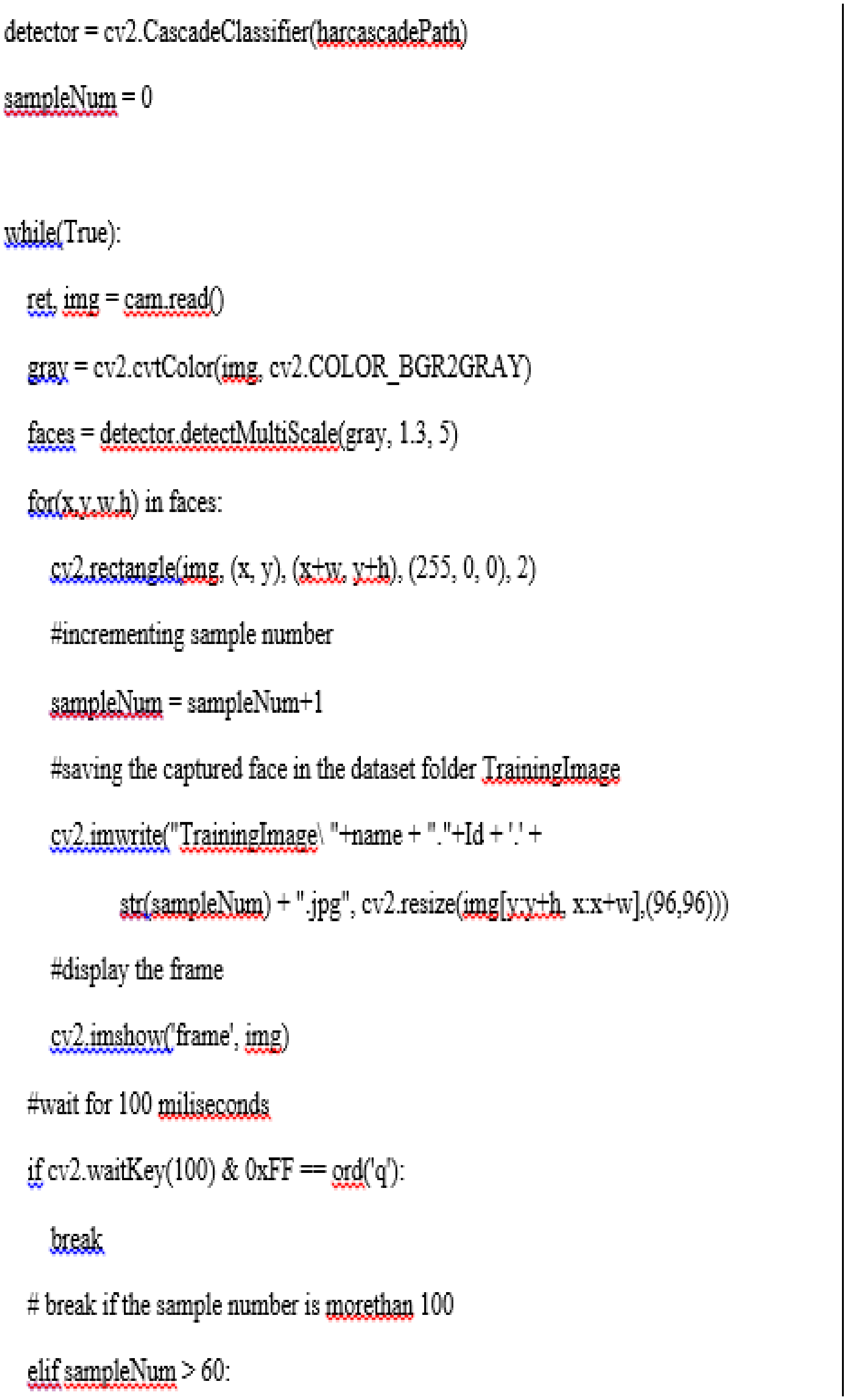
This file is used to capture the images so that model can make predictions on the images. First of all the program asks for the name and roll number of the user whose images are to be clicked? Then roll number is checked whether is only contains integers. Then an object of OpenCV is used to read images. Haar Cascade filter is loaded which is used to detect the frontal face of the user. This filter only works in grey scale so the images are captured in grey scale format. These images are saved in a jpg format. Also names and roll number of user is saved in a csv file so that it can be retrieved after.

This process automatically stops when 60 different images of user have been

Code:-

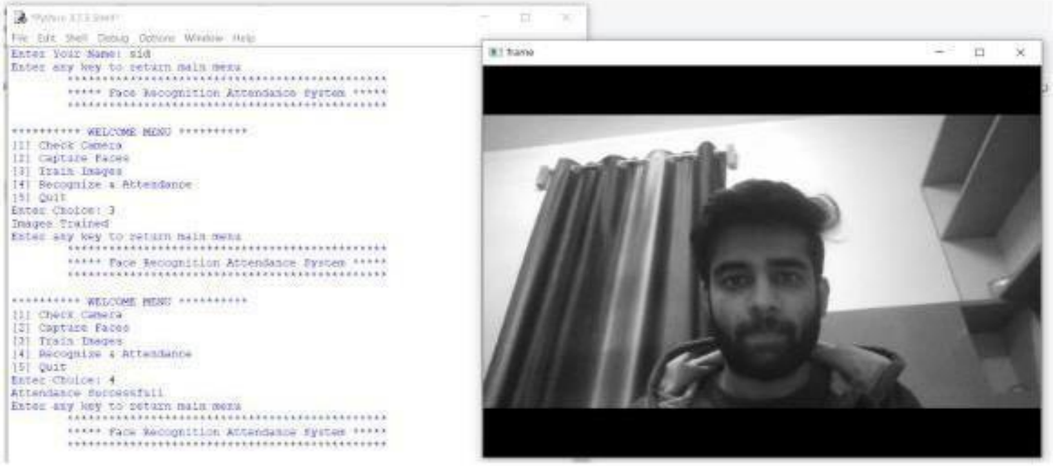


```
import csv

import cv2

def is_number(s):

    try:

        float(s)

        return True

    except ValueError:

        pass

    try:

        import unicodedata

        unicodedata.numeric(s)

        return True

    except (TypeError, ValueError):

        pass

    return False

def takeImages():

     Id = input("Enter Your Id: ")

    name = input("Enter Your Name: ")

    if(is_number(Id) and name.isalpha()):

        cam = cv2.VideoCapture(0)

        harcascadePath = "haarcascade_frontalface_default.xml"
```



```
detector = cv2.CascadeClassifier(harcascadePath)
sampleNum = 0


while(True):
   ret, img = cam.read()
   gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
   faces = detector.detectMultiScale(gray, 1.3, 5)
   for(x,y,w,h) in faces:
      cv2.rectangle(img, (x, y), (x+w, y+h), (255, 0, 0), 2)
      #incrementing sample number
      sampleNum = sampleNum+1
      #saving the captured face in the dataset folder TrainingImage
      cv2.imwrite("TrainingImage\ "+name + "."+Id + '.' +
               str(sampleNum) + ".jpg", cv2.resize(img[y:y+h, x:x+w],(96,96)))
      #display the frame
      cv2.imshow('frame', img)
   #wait for 100 miliseconds
   if cv2.waitKey(100) & 0xFF == ord('q'):
      break
   # break if the sample number is morethan 100
   elif sampleNum > 60:
```

```
                    break
            cam.release()
            cv2.destroyAllWindows()
            res = "Images Saved for ID : " + Id + " Name : " + name
            row = [Id, name]
            with open('StudentDetails\StudentDetails.csv', 'a+') as csvFile:
                writer = csv.writer(csvFile)
                writer.writerow(row)
            csvFile.close()
        else:
            if(is_number(Id)):
                print("Enter Alphabetical Name")
            if(name.isalpha()):
                print("Enter Numeric ID")
```



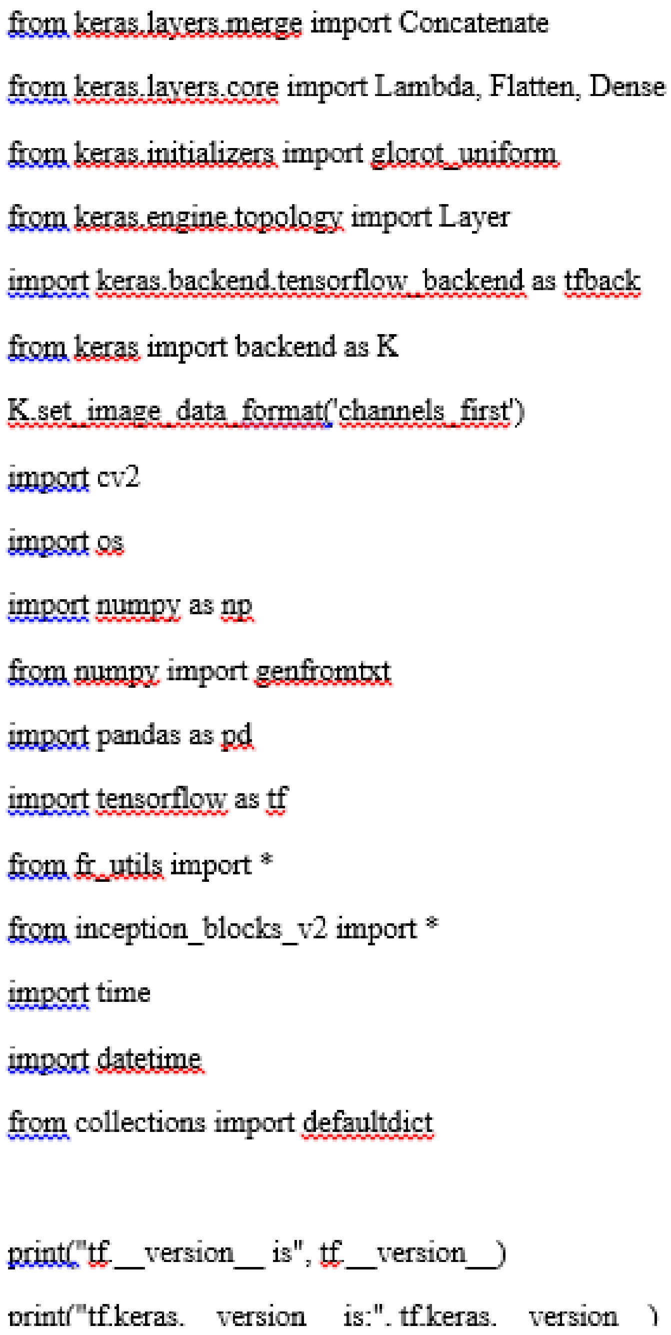
```
Enter Your Name: sid
Enter any key to return main menu
        ***********************************************
        ***** Face Recognition Attendance System *****
        ***********************************************

********** WELCOME MENU **********
[1] Check Camera
[2] Capture Faces
[3] Train Images
[4] Recognize & Attendance
[5] Quit
Enter Choice: 3
Images Trained
Enter any key to return main menu
        ***********************************************
        ***** Face Recognition Attendance System *****
        ***********************************************

********** WELCOME MENU **********
[1] Check Camera
[2] Capture Faces
[3] Train Images
[4] Recognize & Attendance
[5] Quit
Enter Choice: 4
Attendance Successfull
Enter any key to return main menu
        ***********************************************
        ***** Face Recognition Attendance System *****
        ***********************************************
```

Fig 4.3 Output of capture_image.py

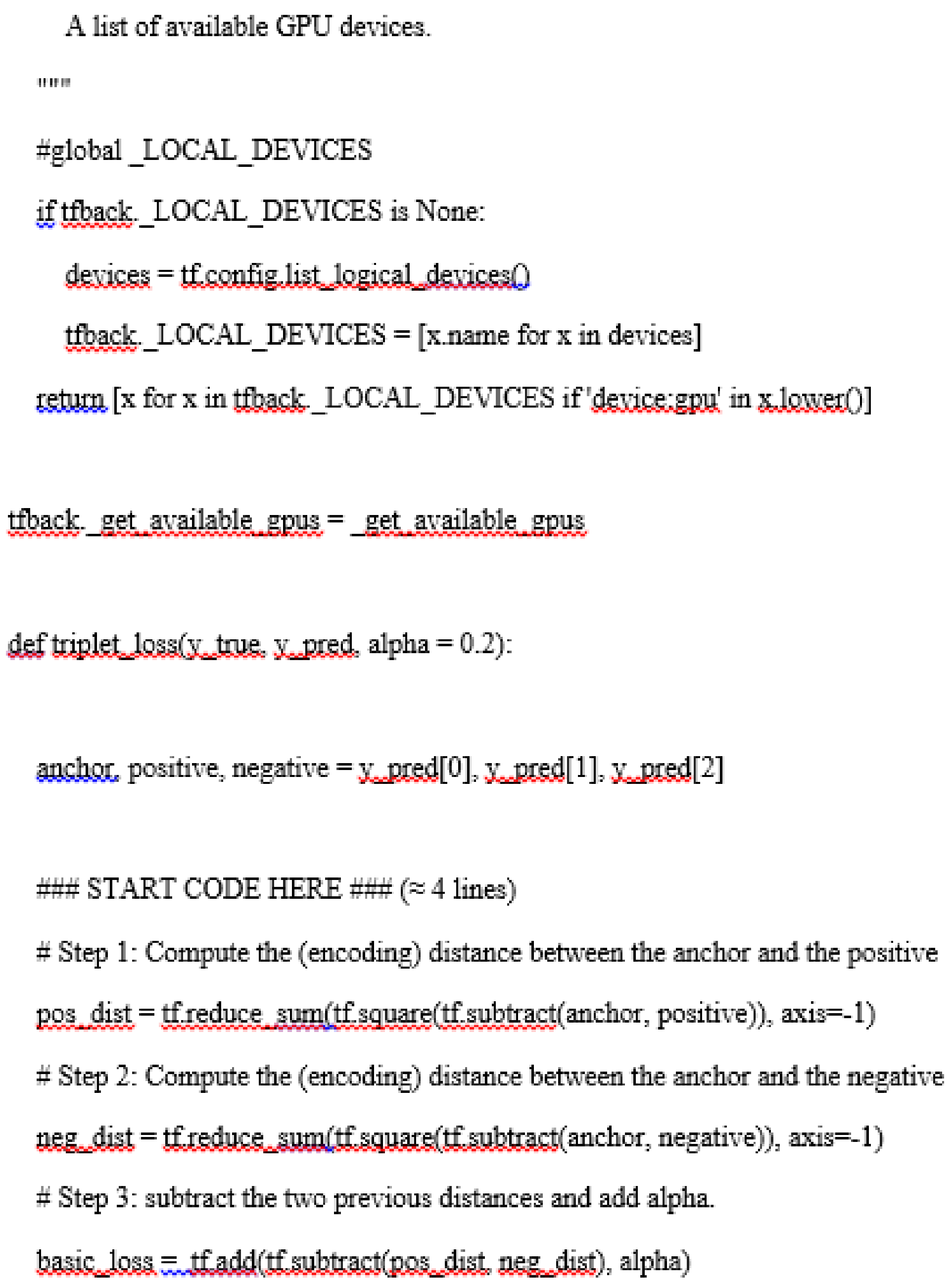
4.4 Train_image.py

This file is used to load the weights of a pre trained inception network and make predictions. First of all the pre trained model is loaded into an object called FRmodel. Then haar cascade and open CV both are used to detect the front face of a user in real time. After the detection of face, the earlier captured images of user are passed into the inception network. Inception network which contains several convolution and pooling layers, converts these images into an (m*128) dimensional vector where m is the size of dataset per user. Then the newly captured image is passed so that it can also be converted into 128 dimensional vector. After that, triplet loss function which computes the distance between two vectors. If this distance is less than or equal to a threshold value, then user is matched with database. In our project the threshold value is 0.7. Higher the value, more linnet is the system. Tensor flow and Keras are used to implement all this. When user is matched then, its name and roll number is displayed below the image in real time. If a user is not recognized then unknown is displayed below the image of the user in real time. Also an excel file is created, which contains all the names and roll numbers of students which are recognized by the system. This system can recognize multiple users in single time. As we see in figure that out of two it is recognizing only one and indicates unknown on the other face that is not trained. Hence attendance of the only one with the name is being marked Present

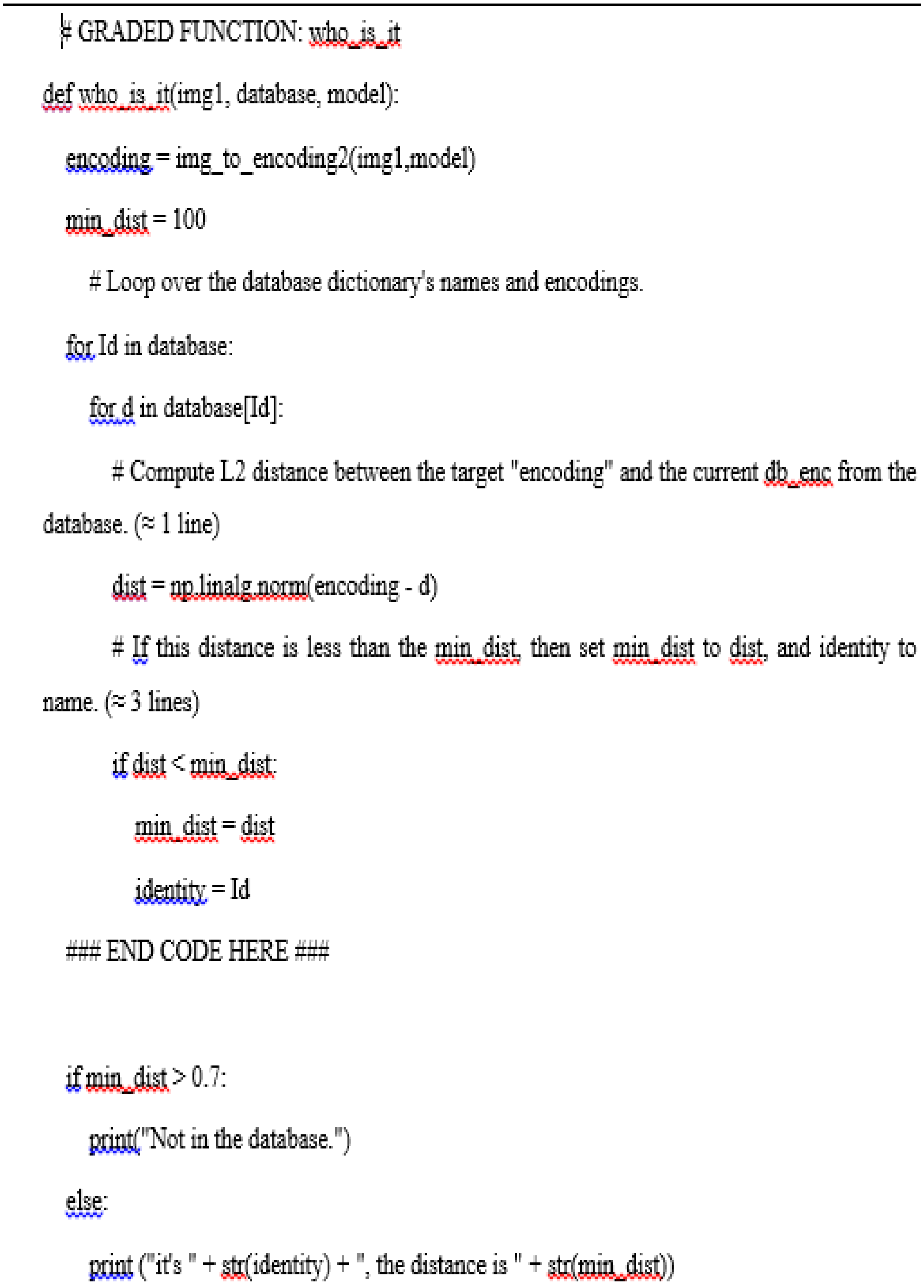
Code:-



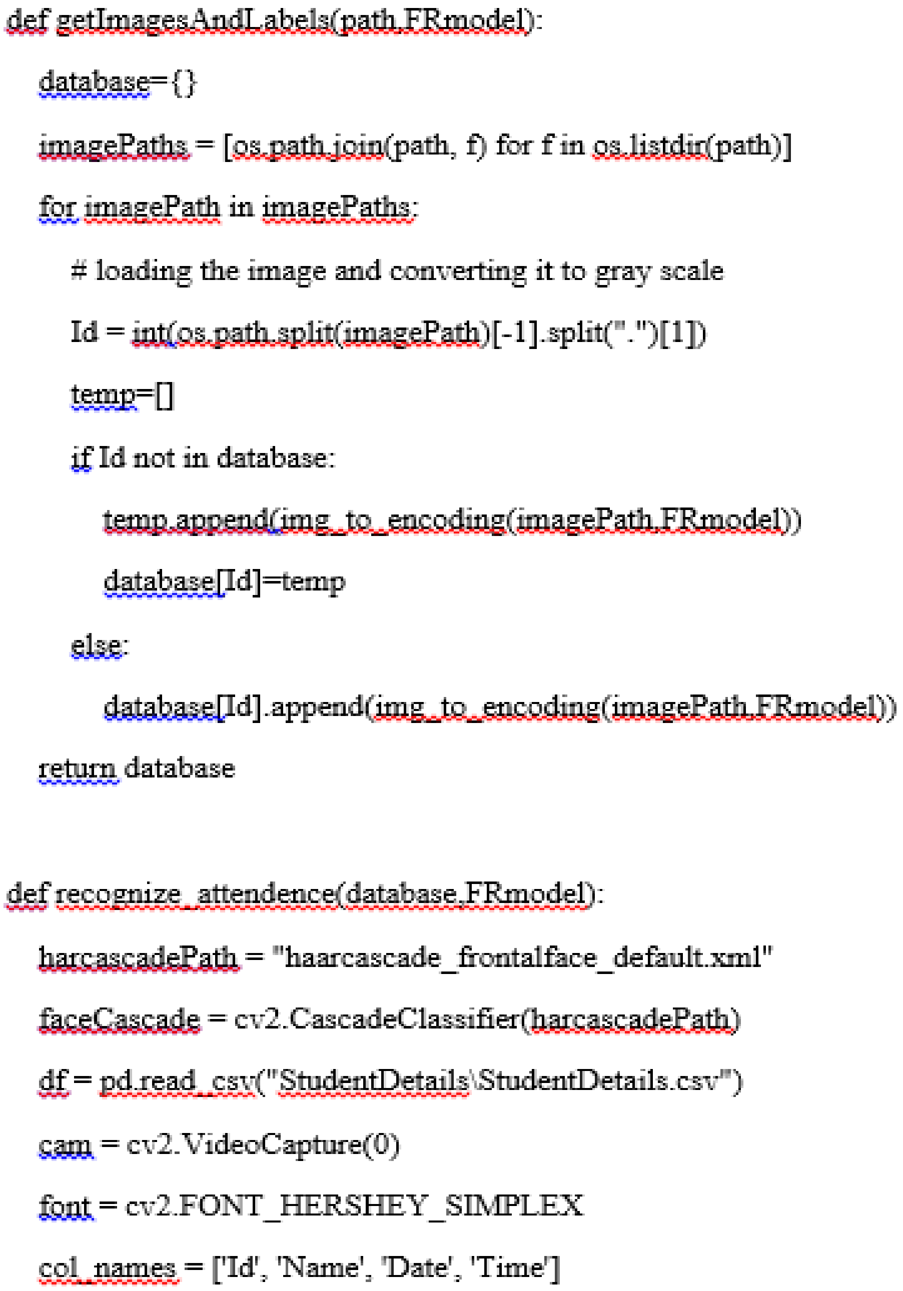
```
from keras.layers.merge import Concatenate

from keras.layers.core import Lambda, Flatten, Dense

from keras.initializers import glorot_uniform

from keras.engine.topology import Layer

import keras.backend.tensorflow_backend as tfback

from keras import backend as K

K.set_image_data_format('channels_first')

import cv2

import os

import numpy as np

from numpy import genfromtxt

import pandas as pd

import tensorflow as tf

from fr_utils import *

from inception_blocks_v2 import *

import time

import datetime

from collections import defaultdict


print("tf.__version__ is", tf.__version__)

print("tf.keras.__version__ is:", tf.keras.__version__)
```



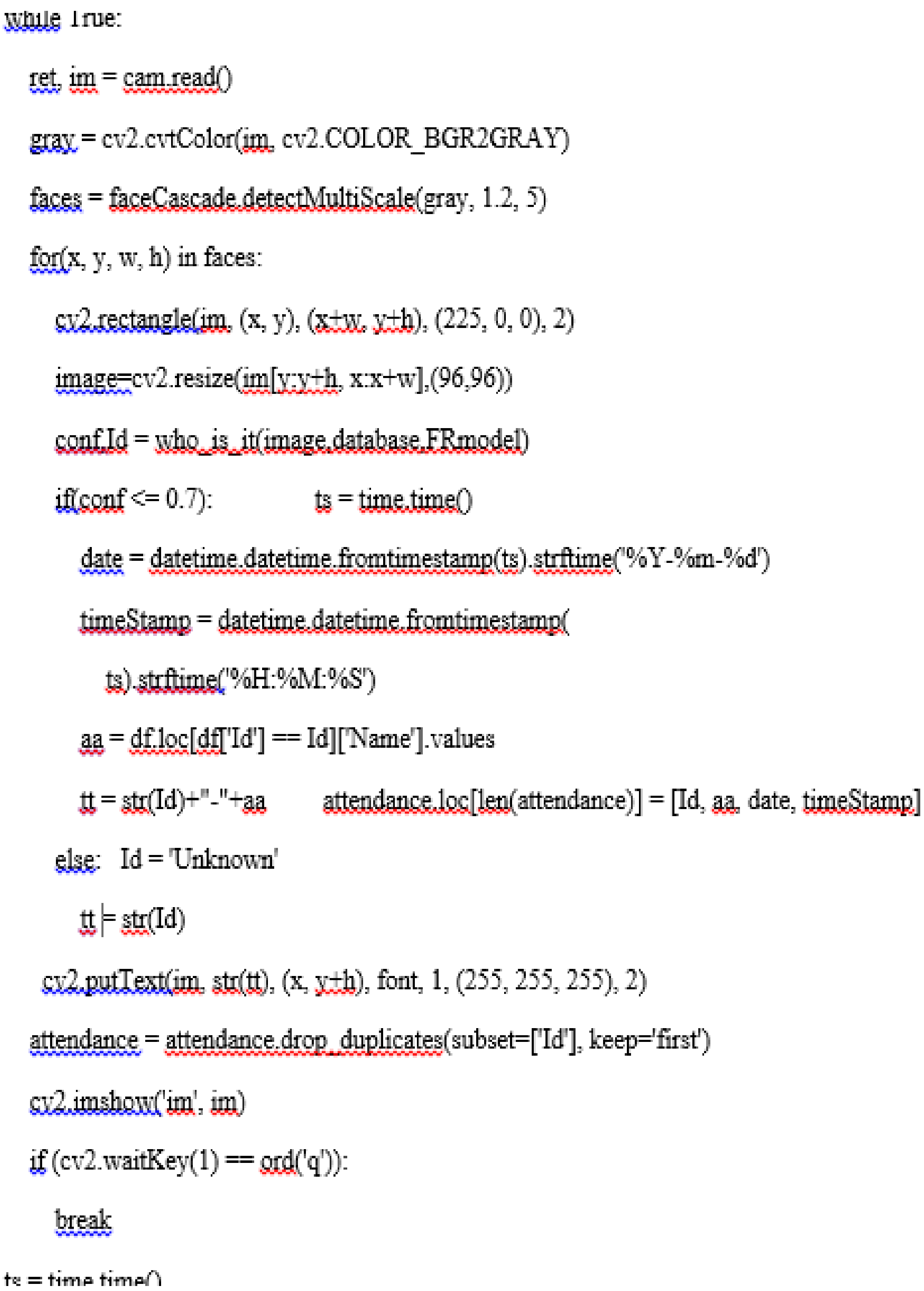
```
        A list of available GPU devices.
    """
    #global _LOCAL_DEVICES
    if tfback._LOCAL_DEVICES is None:
        devices = tf.config.list_logical_devices()
        tfback._LOCAL_DEVICES = [x.name for x in devices]
    return [x for x in tfback._LOCAL_DEVICES if 'device:gpu' in x.lower()]

tfback._get_available_gpus = _get_available_gpus

def triplet_loss(y_true, y_pred, alpha = 0.2):

    anchor, positive, negative = y_pred[0], y_pred[1], y_pred[2]

    ### START CODE HERE ### (≈ 4 lines)
    # Step 1: Compute the (encoding) distance between the anchor and the positive
    pos_dist = tf.reduce_sum(tf.square(tf.subtract(anchor, positive)), axis=-1)
    # Step 2: Compute the (encoding) distance between the anchor and the negative
    neg_dist = tf.reduce_sum(tf.square(tf.subtract(anchor, negative)), axis=-1)
    # Step 3: subtract the two previous distances and add alpha.
    basic_loss = tf.add(tf.subtract(pos_dist, neg_dist), alpha)
```



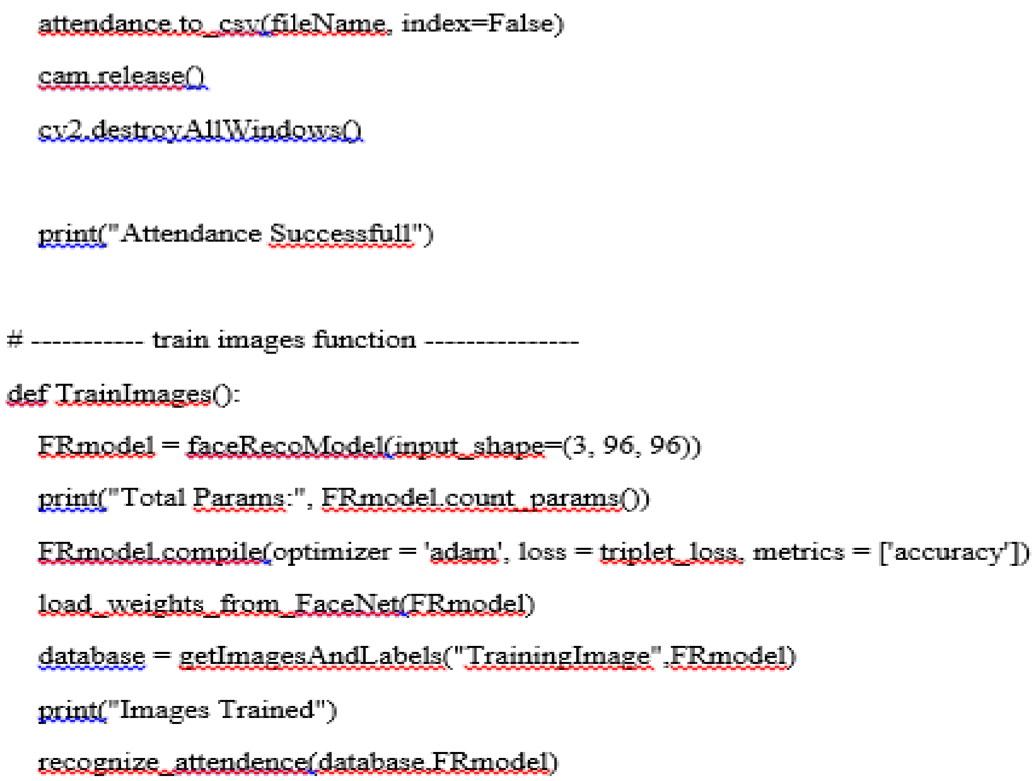
```
  # GRADED FUNCTION: who_is_it
def who_is_it(img1, database, model):
    encoding = img_to_encoding2(img1,model)
    min_dist = 100
        # Loop over the database dictionary's names and encodings.
    for Id in database:
        for d in database[Id]:
            # Compute L2 distance between the target "encoding" and the current db_enc from the
database. (≈ 1 line)
            dist = np.linalg.norm(encoding - d)
            # If this distance is less than the min_dist, then set min_dist to dist, and identity to
name. (≈ 3 lines)
            if dist < min_dist:
                min_dist = dist
                identity = Id
    ### END CODE HERE ###


    if min_dist > 0.7:
        print("Not in the database.")
    else:
        print ("it's " + str(identity) + ", the distance is " + str(min_dist))
```



```
def getImagesAndLabels(path,FRmodel):
    database={}
    imagePaths = [os.path.join(path, f) for f in os.listdir(path)]
    for imagePath in imagePaths:
        # loading the image and converting it to gray scale
        Id = int(os.path.split(imagePath)[-1].split(".")[1])
        temp=[]
        if Id not in database:
            temp.append(img_to_encoding(imagePath,FRmodel))
            database[Id]=temp
        else:
            database[Id].append(img_to_encoding(imagePath,FRmodel))
    return database

def recognize_attendence(database,FRmodel):
    harcascadePath = "haarcascade_frontalface_default.xml"
    faceCascade = cv2.CascadeClassifier(harcascadePath)
    df = pd.read_csv("StudentDetails\StudentDetails.csv")
    cam = cv2.VideoCapture(0)
    font = cv2.FONT_HERSHEY_SIMPLEX
    col_names = ['Id', 'Name', 'Date', 'Time']
```

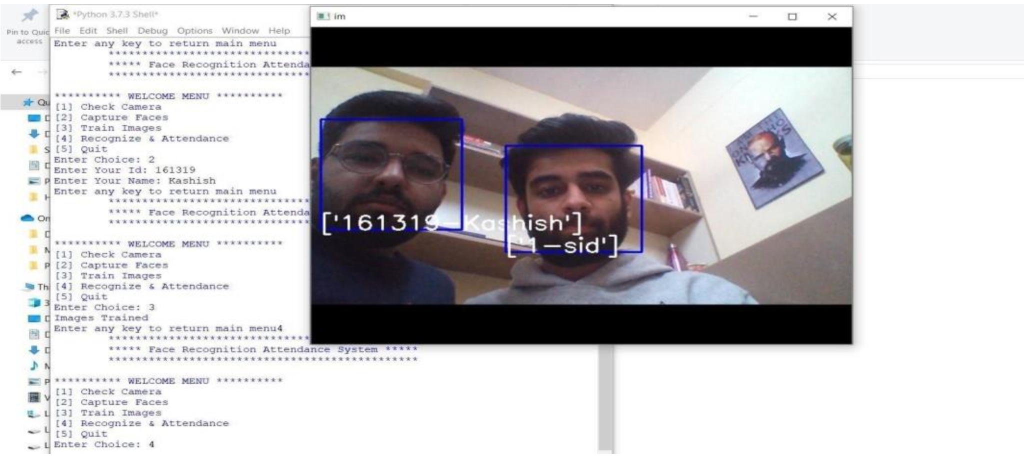


```
while True:
    ret, im = cam.read()
    gray = cv2.cvtColor(im, cv2.COLOR_BGR2GRAY)
    faces = faceCascade.detectMultiScale(gray, 1.2, 5)
    for(x, y, w, h) in faces:
        cv2.rectangle(im, (x, y), (x+w, y+h), (225, 0, 0), 2)
        image=cv2.resize(im[y:y+h, x:x+w],(96,96))
        conf,Id = who_is_it(image,database,FRmodel)
        if(conf <= 0.7):                ts = time.time()
            date = datetime.datetime.fromtimestamp(ts).strftime('%Y-%m-%d')
            timeStamp = datetime.datetime.fromtimestamp(
                ts).strftime('%H:%M:%S')
            aa = df.loc[df['Id'] == Id]['Name'].values
            tt = str(Id)+"-"+aa         attendance.loc[len(attendance)] = [Id, aa, date, timeStamp]
        else:   Id = 'Unknown'
            tt = str(Id)
      cv2.putText(im, str(tt), (x, y+h), font, 1, (255, 255, 255), 2)
    attendance = attendance.drop_duplicates(subset=['Id'], keep='first')
    cv2.imshow('im', im)
    if (cv2.waitKey(1) == ord('q')):
        break
ts = time.time()
```



```
    attendance.to_csv(fileName, index=False)
    cam.release()
    cv2.destroyAllWindows()

    print("Attendance Successfull")

# ----------- train images function ---------------
def TrainImages():
    FRmodel = faceRecoModel(input_shape=(3, 96, 96))
    print("Total Params:", FRmodel.count_params())
    FRmodel.compile(optimizer = 'adam', loss = triplet_loss, metrics = ['accuracy'])
    load_weights_from_FaceNet(FRmodel)
    database = getImagesAndLabels("TrainingImage",FRmodel)
    print("Images Trained")
    recognize_attendence(database,FRmodel)
```

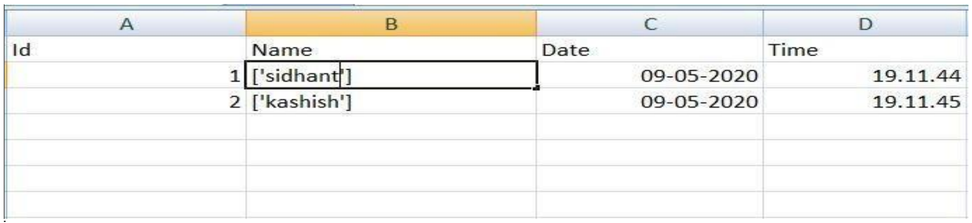
As in the given figure we can see that the model is trained so that it can recognize more than one face at a single time for decreasing the attendance marking time as in figure the attendance of both the students marked present in the same time.



```
Enter any key to return main menu
        ******************************
        ***** Face Recognition Attenda
        ******************************

********** WELCOME MENU **********
[1] Check Camera
[2] Capture Faces
[3] Train Images
[4] Recognize & Attendance
[5] Quit
Enter Choice: 2
Enter Your Id: 161319
Enter Your Name: Kashish
Enter any key to return main menu
        ******************************
        ***** Face Recognition Attenda
        ******************************

********** WELCOME MENU **********
[1] Check Camera
[2] Capture Faces
[3] Train Images
[4] Recognize & Attendance
[5] Quit
Enter Choice: 3
Images Trained
Enter any key to return main menu4
        ******************************************
        ***** Face Recognition Attendance System *****
        ******************************************

********** WELCOME MENU **********
[1] Check Camera
[2] Capture Faces
[3] Train Images
[4] Recognize & Attendance
[5] Quit
Enter Choice: 4
```

Fig. 4.4 Output of train_image.py

Here you can see the excel file which marked only the attendance of the recognized students



A	B	C	D
Id	Name	Date	Time
1	['sidhant']	09-05-2020	19.11.44
2	['kashish']	09-05-2020	19.11.45

Fig. 4.5 Output of MS Excel

4.5 Inception_block_v2.py

This file contains the implementation of inception network. In inception network, many layers of convolution and max pooling are followed by followed by a relu functions.

Convolution layers include convolutions of 1X1, 3X3 and 5X5. End part of each layer is 1X1 convolution, which is used for dimension reduction. At the end of the network, all the vectors are converted into a single dimension and passed into a dense 128 neural network. Thus it converts the images into 128 dimensional vector.

Chapter 5
Conclusion

In this chapter, we observe how neural networks can be used to implement face detection and face recognition. It will speed up the attendance marking system.

To conclude, we have discussed the whole process of development of this system and we can rely on this system.

Future Scope

This chapter discusses the future scope or the implementation of this robot. To increase the scope of this device we can add some new features. As technology is becoming more advance it will be mandatory to change the structure some day with better replacement and sometimes based on customer requirements. The results that we achieved are as follows:

- The system can be administered by a non-IT technician.
- The system is market ready for commercial use.
- The system has the capacity to carry up to a thousand faces to recognize.
- The system can serve as much people as they want within an organization. There are so many future scope on this project. Some of them are
- Can improve security
- Can used in big factory or employee attendance
- Can build on fully web base system.

Taking this on more advance level, this system can be integrated with web based projects and can be used from all over the world even on other devices like mobile phones or tablets.

References

i. R. Samet and M. Tanriverdi, "Face Recognition-Based Mobile Automatic Classroom Attendance Management System," *2017 International Conference on Cyber worlds (CW)*, Chester, 2017, pp.253-256.

   Doi: 10.1109/CW.2017.34

ii. S. Sawhney, K. Kacker, S. Jain, S. N. Singh and R. Garg, "Real-Time Smart Attendance System using Face Recognition Techniques," *2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, Noida, India, 2019, pp. 522-525.

   Doi: 10.1109/CONFLUENCE.2019.8776934

iii. S. Bhattacharya, G. S. Nainala, P. Das and A. Routray, "Smart Attendance Monitoring System (SAMS): A Face Recognition Based Attendance System for Classroom Environment," *2018 IEEE 18th International Conference on Advanced Learning Technologies, (ICALT)*, Mumbai, 2018, pp.358-360.

   Doi: 10.1109/ICALT.2018.00090

iv. P. Wagh, R. Thakare, J. Chaudhari and S. Patil, "Attendance system based on face recognition using Eigen face and PCA algorithms," *2015 International Conference on Green Computing and Internet of Things (ICGCIoT)*, Noida, 2015, pp. 303-308.

   Doi: 10.1109/ICGCIoT.2015.7380478

v. L. Fung-Lung, M. Nycander-Barúa and P. Shiguihara-Juárez, "An Image Acquisition Method for Face Recognition and Implementation of an Automatic Attendance System for Events," *2019 IEEE XXVI International Conference on Electronics, Electrical Engineering andComputing (INTERCON)*, Lima, Peru, 2019, pp. 1-4.

   Doi: 10.1109/INTERCON.2019.8853603

Major_kashish_sidhant

ORIGINALITY REPORT

10%	5%	1%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	eie.uonbi.ac.ke Internet Source	2%
2	www.irjet.net Internet Source	2%
3	Submitted to Federal University of Technology Student Paper	1%
4	opencv.org Internet Source	1%
5	Submitted to Universiti Teknologi MARA Student Paper	1%
6	Kazi Toyebul Haque, M. K. Robin, Mohammad Ashraful Anam, Hasan U. Zaman. "MediPro — A cost effective and user-friendly medical information system", 2017 International Conference on Intelligent Computing and Control Systems (ICICCS), 2017 Publication	<1%
7	Submitted to University of Wales Institute, Cardiff	<1%

	Student Paper	
8	Submitted to University of Greenwich Student Paper	<1%
9	Submitted to Coventry University Student Paper	<1%
10	Y. R. Azeez, C. Rajapakse. "An Application of Transfer Learning Techniques in Identifying Herbal Plants in Sri Lanka", 2019 International Research Conference on Smart Computing and Systems Engineering (SCSE), 2019 Publication	<1%
11	Submitted to Visvesvaraya Technological University Student Paper	<1%
12	Submitted to Caledonian College of Engineering Student Paper	<1%
13	Submitted to Texas A & M University, Kingville Student Paper	<1%
14	Submitted to The Hong Kong Institute of Education Student Paper	<1%
15	Submitted to University of Ulster Student Paper	<1%
16	www.pantechsolutions.net Internet Source	<1%

17	Praveenbalaji D, Srinivas R, Roopa S, Suresh M, Gayathri A. "ID Photo Verification by Face Recognition", 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), 2020 Publication	<1%
18	Submitted to Universiti Teknologi Malaysia Student Paper	<1%
19	Submitted to University of Leicester Student Paper	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

PLAGIARISM VERIFICATION REPORT

Date: 14/07/2020

Type of Document (Tick): PhD Thesis | M.Tech Dissertation/ Report | [✓ B.Tech Project Report] | Paper

Name: Sidhant Sharma , Kashish Bajaj **Department:** Computer Science **Enrolment No**- 161293,161319

Contact No. 7780926339, 7018317641 **E-mail.** sidhantsharma293@gmail.com , bajajkashish10@gmail.com

Name of the Supervisor: Dr. Rajinder Sandhu

Title of the Thesis/Dissertation/Project Report/Paper (In Capital letters: ATTENDANCE SYSTEM USING FACE RECOGNITION

UNDERTAKING

I undertake that I am aware of the plagiarism related norms/ regulations, if I found guilty of any plagiarism and copyright violations in the above thesis/report even after award of degree, the University reserves the rights to withdraw/revoke my degree/report. Kindly allow me to avail Plagiarism verification report for the document mentioned above.

- Total No. of Pages = 51
- Total No. of Preliminary pages = 8
- Total No. of pages accommodate bibliography/references = 1
-

Sidhant

(Signature of Student)

FOR DEPARTMENT USE

We have checked the thesis/report as per norms and found **Similarity Index** at ………**10%**... Therefore, we are forwarding the complete thesis/report for final plagiarism check. The plagiarism verification report may be handed over to the candidate.

(Signature of Guide/Supervisor) **Signature of HOD**

FOR LRC USE

The above document was scanned for plagiarism check. The outcome of the same is reported below:

<table>
<tr><th>Copy Received on</th><th>Excluded</th><th>Similarity Index (%)</th><th colspan="2">Abstract & Chapters Details</th></tr>
<tr><td rowspan="2"></td><td rowspan="4">• All Preliminary Pages
• Bibliography/ Images/Quotes
• 14 Words String</td><td rowspan="2"></td><td>Word Counts</td><td></td></tr>
<tr><td>Character Counts</td><td></td></tr>
<tr><td><b>Report Generated on</b></td><td><b>Submission ID</b></td><td>Page counts</td><td></td></tr>
<tr><td></td><td></td><td>File Size</td><td></td></tr>
</table>

Please send your complete Thesis/Report in (PDF) & DOC (Word File) through your Supervisor/Guide at plagcheck.juit@gmail.com

Checked by
Name & Signature **Librarian**

……

Please send your complete Thesis/Report in (PDF) & DOC (Word File) through your Supervisor/Guide at plagcheck.juit@gmail.com