INVERTIS INSTITUTE OF COMPUTER APPLICATION

A Project Report
on
Smart Attendance

Submitted in Partial Fulfillment of the Requirement for the
Degree of Master of Computer Applications
Invertis University, Bareilly

Project Guide:
Mohd. Shakeel
Astt. Professor
Department of
Computer Applications

Submitted By:
Shubham Sharma
1850204029

CERTIFICATE

This is to certify that **Shubham Sharma,** student of MCA course of Invertis **University, Bareilly**, has undergone a mandatory **"FIFTH SEMESTER PROJECT"** in our organization.

During the course of his project development he has completed a project on **Smart Attendance.**

This project is in partial fulfillment for the requirement of M.C.A. curriculum as per university norms.

I wish him/her all the best in his/her carrier.

JITENDRA KR CHAUDHARY

H.O.D.

Invertis Institute of Computer Applications

MOHD SHAKEEL

Project Guide

ACKNOWLEGDEMENT

I extend my Thanks to Invertis Institute of Computer Applications, Invertis University, Bareilly who has given me opportunity to work on this Project.

Today I am feeling a great sense of Excitement on my way to successfully complete my project on "**Smart Attendance"** under the guidance of **"Mr. Kushal Johari"**.

I sincerely thank him for responding great confidence and faith in my work and being with me to encourage and guide me to successful project completion.

I should also like to thank **Mr. Jitendra Kr. Chaudhary, HOD, Department of Computer Applications** for their support and all our friends and colleagues who have created an atmosphere to encourage me from time to time making our work easy.

Thank You.

Shubham Sharma

1850204029

INDEX

LIST OF TABLES

LIST OF FIGURES

1. Introduction of project

Smart Attendance Management System is an application developed for daily student attendance in colleges or schools. This project attempts to record attendance through face detection. The aim of the project is to create software for attendance management system that can be used by all teachers for their respective courses. The teacher will start the process of taking attendance by clicking a photograph of the class. The algorithm used to detect face is Voila Jones. Histogram of Gradients is used to recognize feature. And for classification the algorithm used is Error Correcting Output Coding which uses Support Vector Machine as a binary classifier.

2. Problem definition

Maintaining attendance is very important in all learning institutes for checking the performance of students. In most learning institutions, student attendances are manually taken by the use of attendance sheets issued by the department heads as part of regulation. The students sign in these sheets which are then filled or manually logged in to a computer for future analysis.

This method is tedious, time consuming and inaccurate as some students often sign for their absent colleagues. This method also makes it difficult to track the attendance of individual students in a large classroom environment. In this project, we propose the design and use of a face detection and recognition system to automatically detect students attending a lecture in a classroom and mark their attendance by recognizing their faces.

While other biometric methods of identification (such as iris scans or fingerprints) can be more accurate, students usually have to queue for long at the time they enter the classroom, but this is not the case with face recognition.

3. System Analysis & Feasibility Study

a. *Planned approach towards working*: **-** The working in the organization will be well planned and organized. The data i.e. Image will be stored properly in database stores which will help in retrieval of information as well as its storage.

b. *Accuracy*: - The level of accuracy in the proposed system will be higher. All operation would be done correctly and it ensures that whatever information is coming from the center is accurate.

c. *Reliability*: - The reliability of the proposed system will be high due to the above stated reasons. The reason for the increased reliability of the system is that now there would be proper storage of information.

d. *No Redundancy*: **-** In the proposed system utmost care would be that no information is repeated anywhere, in storage or otherwise. This would assure economic use of storage space and consistency in the data stored.

e. *Immediate retrieval of information***: -** The main objective of proposed system is to provide for a quick and efficient detection of required information. Any type of detection would be available whenever the user requires.

f. *Immediate storage of information*: - In manual system there are many problems to store the largest amount of information for processing.

g. *Easy to Operate***: -** The system should be easy to operate and should be such that it can be developed within a short period of time and fit in the limited budget of the user.

Economic Feasibility: - Among the most important information contained in feasibility study is the cost-benefit analysis. That is, an assessment of economic justification for computer-based system. Cost-benefit analysis delineates cost for development and weights them against tangible and intangible benefits in the system.

Technical Feasibility: - Technical analysis evaluates technical merits of the system at the same time collecting additional information about performance, reliability, maintainability and productivity. In some cases, this system analysis step also includes a limited amount of research and design.

Operational Feasibility: - Operational feasibility measures how well the solution will work in the organization and how will end-user & management feel about the system. Proposed system is helpful for all the users associated with the organization. It will allow the administrator to have up-to-date information regarding all the aspects of their users. The decision-making process will also become faster with the use of data integration, consolidation. So, it is feasible to implement the system.

4. Scope of the Proposed System

- Provides facility for the automated attendance of students.
- Uses live face recognition to recognize each individual and mark their attendance automatically.
- Utilizes video and image processing to provide inputs to the system.
- Facility of marking manual attendance.
- Notification via email if there is a lack of attendance.

5. System Requirements

a. Hardware Requirements

- Windows or Linux PC
- Webcam
- 512 MB RAM
- Hard Disk: 40 GB minimum.

b. Software Requirements

i. Front End

- Python 3.3+ or Python 2.7

- Linux or Windows OS

ii. Back End

- Installed Libraries- NumPy, Tkinter, PlaySound, xlwrite, OpenCV, eigen-faces, eigen-vectors , os

6. System Logical Design

a. Data Flow Diagram

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system without going into great detail. DFDs can also be used for the visualization of data processing (structured design).

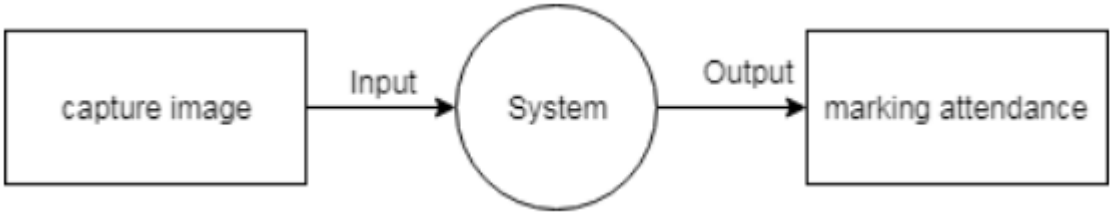

Figure 1: Level 0 - DFD

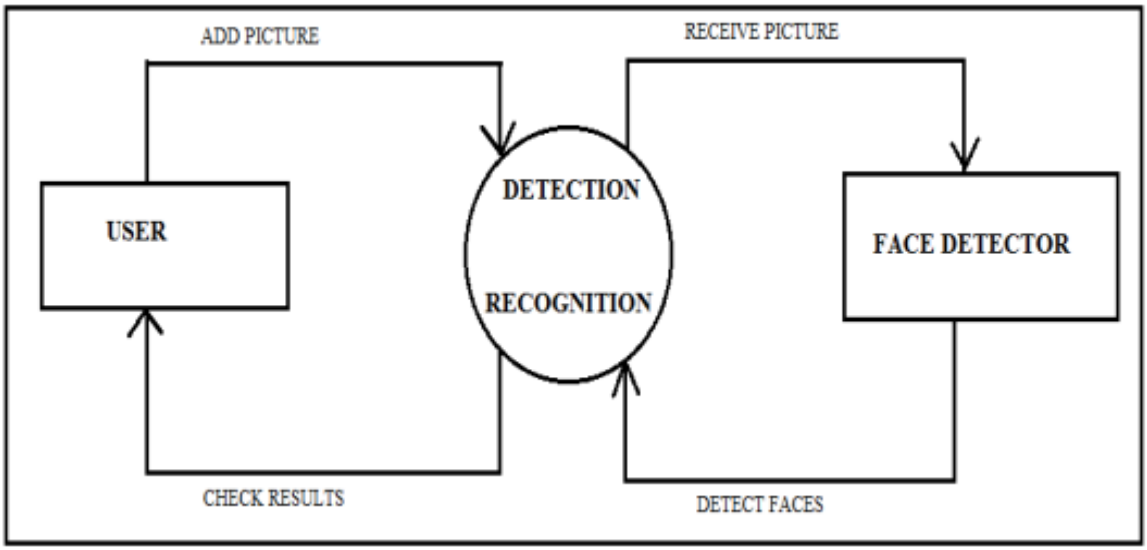

Figure 2: Level 1 - DFD

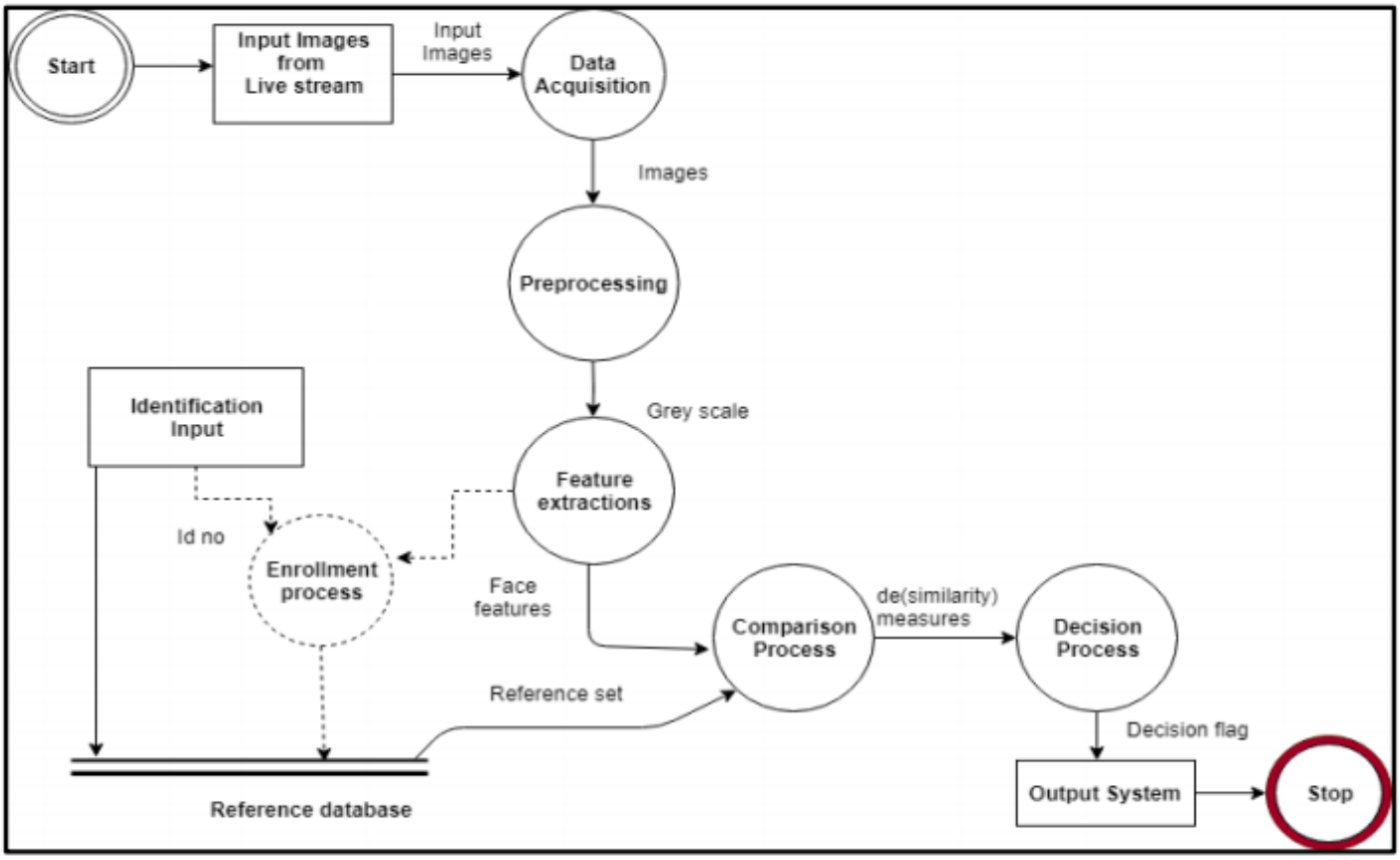

Figure 3: Level 2 - DFD

b. Data Dictionary

Field Name	Data Type	Length	Constraint	Description
ID	Int	4	Primary Key	Student Roll no.
Name	VarChar	20	Not Null	Student Name
Date	Date	10	Not Null	Attendance Date
Present	VarChar	3	Yes/No	Student Attendance

Table 1: Data Dictionary

c. ER Diagram

An entity–relationship model describes interrelated things of interest in a specific domain of knowledge. A basic ER model is composed of entity types and specifies relationships that can exist between entities. Fig 4 Shows the ERD for our project.

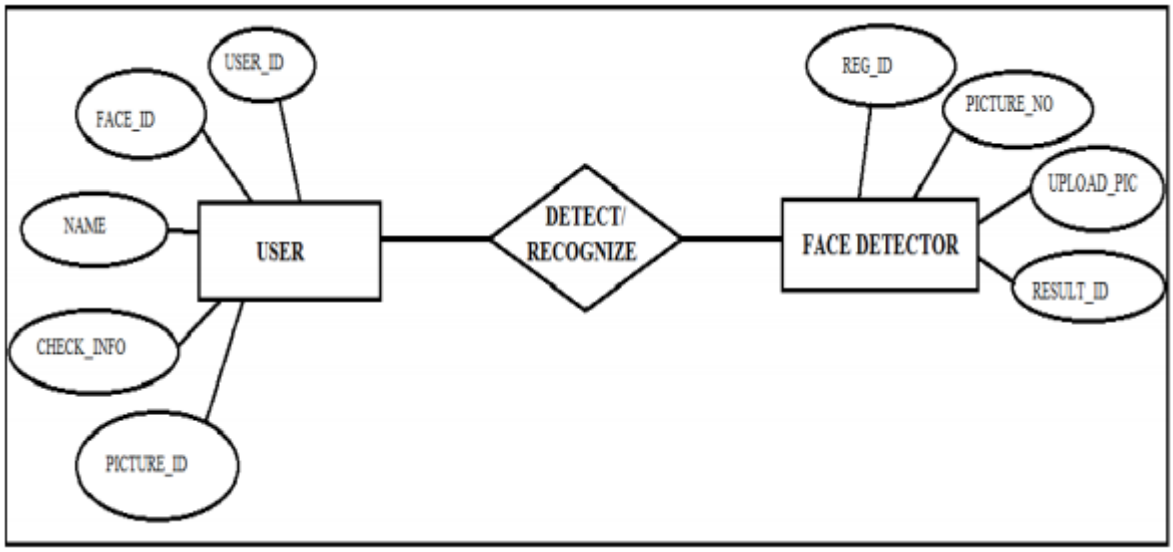

Fig 3: ER Diagram

7. System Physical Design (Coding)

Firstpage.py

```
from tkinter import *
from playsound import playsound
import os
from datetime import datetime;
root=Tk()

root.configure(background="white")

def function1():

    os.system("py dataset_capture.py")

def function2():

    os.system("py training_dataset.py")

def function3():

    os.system("py recognizer.py")
    playsound('sound.mp3')

def function5():
   os.startfile(os.getcwd()+"/developers/index.html");

def function6():
```

```
    root.destroy()

def function4():
    os.startfile(os.getcwd()+"/firebase1/attendence_files/attendence"+str(datetime.now().date())+'.xls')

root.title("AUTOMATIC ATTENDANCE MANAGEMENT USING FACE RECOGNITION")

Label(root, text="FACE RECOGNITION ATTENDANCE SYSTEM",font=("times new
roman",20),fg="white",bg="maroon",height=2).grid(row=0,rowspan=2,columnspan=2,sticky=N+E+W+S,
padx=5,pady=5)

Button(root,text="Create Dataset",font=("times new
roman",20),bg="#0D47A1",fg='white',command=function1).grid(row=3,columnspan=2,sticky=W+E+N+
S,padx=5,pady=5)

Button(root,text="Train Dataset",font=("times new
roman",20),bg="#0D47A1",fg='white',command=function2).grid(row=4,columnspan=2,sticky=N+E+W+
S,padx=5,pady=5)

Button(root,text="Recognize + Attendance",font=('times new
roman',20),bg="#0D47A1",fg="white",command=function3).grid(row=5,columnspan=2,sticky=N+E+W+
S,padx=5,pady=5)

Button(root,text="Attendance Sheet",font=('times new
roman',20),bg="#0D47A1",fg="white",command=function4).grid(row=6,columnspan=2,sticky=N+E+W+
S,padx=5,pady=5)

Button(root,text="Developers",font=('times new
roman',20),bg="#0D47A1",fg="white",command=function5).grid(row=8,columnspan=2,sticky=N+E+W+
S,padx=5,pady=5)

Button(root,text="Exit",font=('times new
roman',20),bg="maroon",fg="white",command=function6).grid(row=9,columnspan=2,sticky=N+E+W+S,
padx=5,pady=5)

root.mainloop()
```

dataset_capture.py

```
# Import OpenCV2 for image processing

import cv2
import os
```

```
def assure_path_exists(path):
    dir = os.path.dirname(path)
    if not os.path.exists(dir):
        os.makedirs(dir)
face_id=input('enter your id')
# Start capturing video
vid_cam = cv2.VideoCapture(0)

# Detect object in video stream using Haarcascade Frontal Face
face_detector = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')

# Initialize sample face image
count = 0

assure_path_exists("dataset/")

# Start looping
while(True):

    # Capture video frame
    _, image_frame = vid_cam.read()

    # Convert frame to grayscale
    gray = cv2.cvtColor(image_frame, cv2.COLOR_BGR2GRAY)

    # Detect frames of different sizes, list of faces rectangles
    faces = face_detector.detectMultiScale(gray, 1.3, 5)

    # Loops for each faces
    for (x,y,w,h) in faces:

        # Crop the image frame into rectangle
        cv2.rectangle(image_frame, (x,y), (x+w,y+h), (255,0,0), 2)

        # Increment sample face image
        count += 1

        # Save the captured image into the datasets folder
        cv2.imwrite("dataset/User." + str(face_id) + '.' + str(count) + ".jpg", gray[y:y+h,x:x+w])

        # Display the video frame, with bounded rectangle on the person's face
        cv2.imshow('frame', image_frame)

    # To stop taking video, press 'q' for at least 100ms
    if cv2.waitKey(100) & 0xFF == ord('q'):
        break
```

```
    # If image taken reach 100, stop taking video
    elif count>=30:
        print("Successfully Captured")
        break

# Stop video
vid_cam.release()

# Close all started windows
cv2.destroyAllWindows()
```

recognizer.py

```
import cv2, numpy as np;
import xlwrite1 as xlwr;
import firebase1.firebase_ini as fire;
import time
import sys
from playsound import playsound

start=time.time()
period=8
face_cas = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')
cap = cv2.VideoCapture(0);
recognizer = cv2.face.LBPHFaceRecognizer_create();
recognizer.read('trainer/trainer.yml');

flag = 0;
id=0;
filename='filename';
dict = {
            'item1': 1
}
#font = cv2.InitFont(cv2.cv.CV_FONT_HERSHEY_SIMPLEX, 5, 1, 0, 1, 1)
font = cv2.FONT_HERSHEY_SIMPLEX
while True:
    ret, img = cap.read();
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY);
    faces = face_cas.detectMultiScale(gray, 1.3, 7);
    for (x,y,w,h) in faces:
        roi_gray = gray[y:y + h, x:x + w]
        cv2.rectangle(img, (x,y), (x+w, y+h), (255,0,0),2);
        id,conf=recognizer.predict(roi_gray)
        if(conf < 70):
         if(id==1):
```

```
            id='Shubham'
            if((str(id)) not in dict):
                filename=xlwr.output('attendence','class1',1,id,'yes');
                dict[str(id)]=str(id);

        elif(id==2):
            id = 'Shubhneet'
            if ((str(id)) not in dict):
                filename =xlwr.output('attendence', 'class1', 2, id, 'yes');
                dict[str(id)] = str(id);

        elif(id==3):
            id = 'Prince'
            if ((str(id)) not in dict):
                filename =xlwr.output('attendence', 'class1', 3, id, 'yes');
                dict[str(id)] = str(id);

        elif(id==4):
            id = '_name_'
            if ((str(id)) not in dict):
                filename =xlwr.output('attendence', 'class1', 4, id, 'yes');
                dict[str(id)] = str(id);

        else:
            id = 'Unknown(name), can not recognize'
            flag=flag+1
            break

        cv2.putText(img,str(id)+" "+str(conf),(x,y-10),font,0.55,(120,255,120),1)
        #cv2.cv.PutText(cv2.cv.fromarray(img),str(id),(x,y+h),font,(0,0,255));
    cv2.imshow('frame',img);
    #cv2.imshow('gray',gray);
    if flag == 10:
        playsound('transactionSound.mp3')
        print("Transaction Blocked")
        break;
    if time.time()>start+period:
        break;
    if cv2.waitKey(100) & 0xFF == ord('q'):
        break;

cap.release();

cv2.destroyAllWindows();
```

training_dataSet.py

```
import os,cv2;
import numpy as np
from PIL import Image;

recognizer = cv2.face.LBPHFaceRecognizer_create()
detector= cv2.CascadeClassifier("haarcascade_frontalface_default.xml");

def getImagesAndLabels(path):
    #get the path of all the files in the folder

    imagePaths=[os.path.join(path,f) for f in os.listdir(path)]
    #create empth face list
    faceSamples=[]
    #create empty ID list
    Ids=[]
    #now looping through all the image paths and loading the Ids and the images

    for imagePath in imagePaths:
        #loading the image and converting it to gray scale
        pilImage=Image.open(imagePath).convert('L')
        #Now we are converting the PIL image into numpy array
        imageNp=np.array(pilImage,'uint8')
        #getting the Id from the image
        Id=int(os.path.split(imagePath)[-1].split(".")[1])
        # extract the face from the training image sample
        faces=detector.detectMultiScale(imageNp)
        #If a face is there then append that in the list as well as Id of it
        for (x,y,w,h) in faces:
            faceSamples.append(imageNp[y:y+h,x:x+w])
            Ids.append(Id)
    return faceSamples,Ids

faces,Ids = getImagesAndLabels('dataSet')

s = recognizer.train(faces, np.array(Ids))

print("Successfully trained")

recognizer.write('trainer/trainer.yml')
```

8. Input / Output Screen

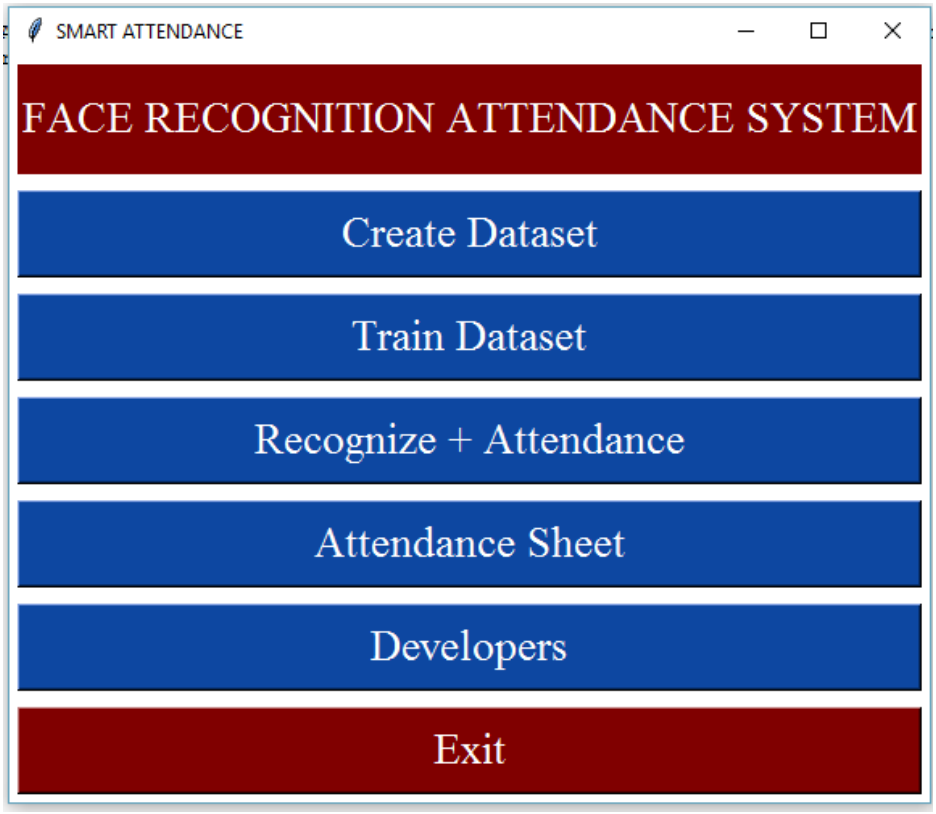

Figure 4: FirstPage

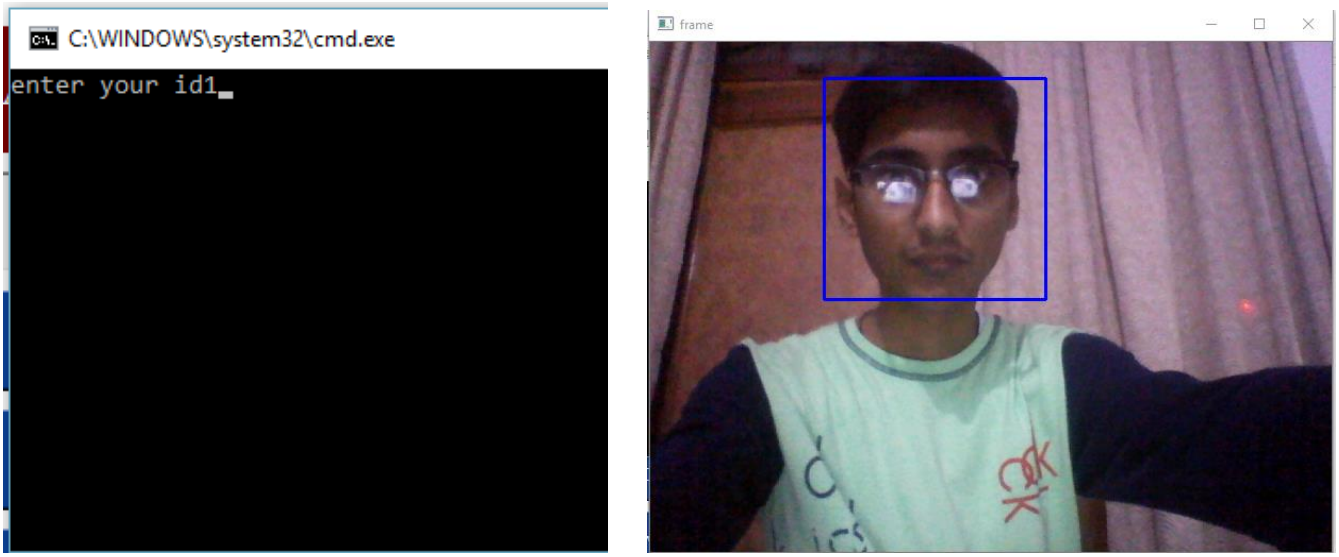

Figure 5: Create Dataset

Figure 6: train Dataset

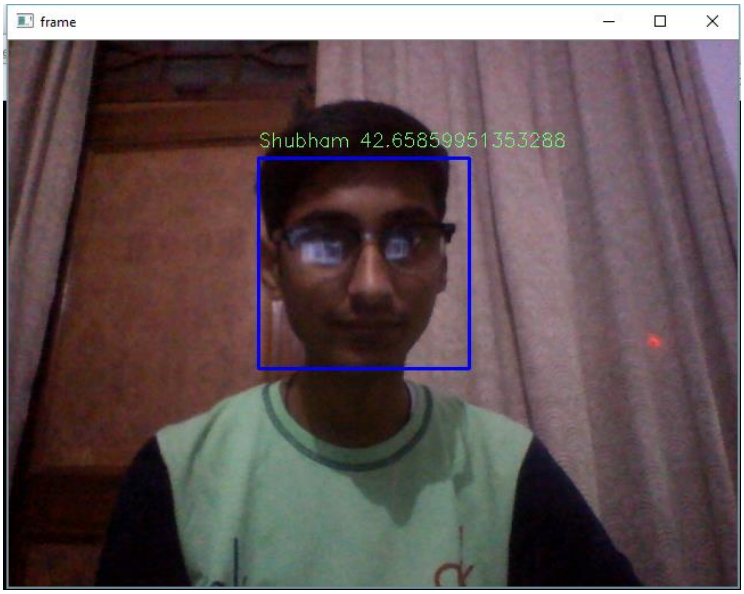

Figure 7: Recognize + Attendance

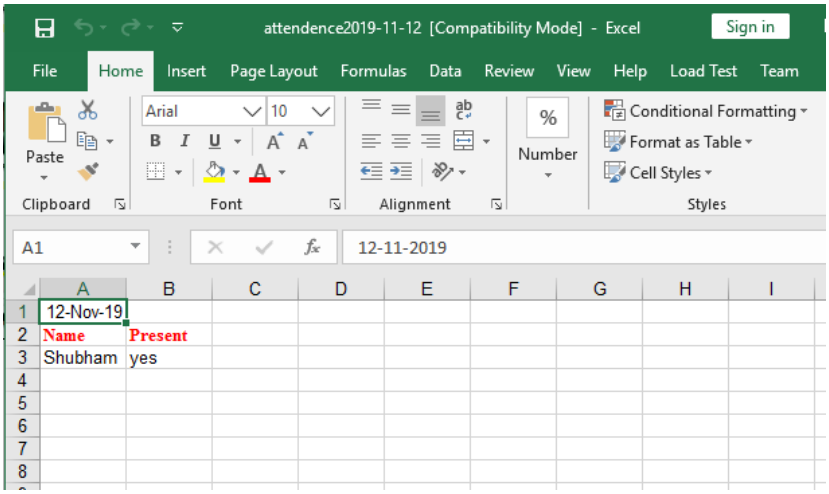

Figure 8: Attendance Report

Developers:-

Figure 9: Developers Page

9. Testing

The testing plan includes planning for several functions like:

- Registration of students
- Face detection
- Capturing 30 images automatically
- Face recognition
- Unknown face detection
- Attendance entry in Excel sheet
- Reports generation

a. Test Cases and Result

ID	A001
TITLE	Face Detection and storage in Training set
PREREQUISITE	Faces are detected from the captured image, cropped and stored in Training Set.
TEST ACTION	1. Start the application 2. Click Create Data Set 3. Enter Student ID
EXPECTED RESULT	On entering Student ID, images are captured, this detected face is cropped and in Training set database, Images are saved by the ID number which was inputted by the student and stored in .jpg format
ACTUAL RESULT	Faces are captured and stored in .jpg format with ID Number.
TEST RESULT	PASS

Table 2: Test Case 1

ID	A002
TITLE	Face Recognition and generation of attendance in an Excel sheet
PREREQUISITE	Faces within the Testing set are recognized with the Training set and accordingly entry is generated in Excel sheet
TEST ACTION	1. Start the application 2. Click Recognize + Attendance
EXPECTED RESULT	Faces are recognized if matched entry is made in Excel sheet of that particular roll number as present if not then attendance is not marked.
ACTUAL RESULT	Performed as per expectation.
TEST RESULT	PASS

Table 3: Test Case 2

ID	A003
TITLE	Face Training
PREREQUISITE	Data set has already been prepared
TEST ACTION	1. Start the application 2. Click Create Data Set 3. Enter Student ID 4. Click Train Dataset
EXPECTED RESULT	Successfully Trained
ACTUAL RESULT	Data is trained Successfully.
TEST RESULT	PASS

Table 4: Test Case 3

b. Test techniques

Considering the scope of the project and the time limitations, we will be performing following tests.

a) Unit Test – This test verifies the program logic and is based on the knowledge of the program structure.
b) Integration Test – This test verifies the entire system's functionality according to the design specification.
c) Acceptance Testing – This test verifies whether the system needs to meet the initial objectives and customer's expectations.

10. Implementation

In order to maintain the attendance this system has been proposed. It replaces the manual system with an automated system which is fast, efficient, cost and time saving as replaces the stationary material and the paper work. Hence this system is expected to give desired results and in future could be implemented for logout. Also, the efficiency could be improved by integrating other techniques with it in near future.

11. Limitations and Future Scope

Limitations

1. Expensive
2. Difficulties with big data processing and storing
3. Strong influence of the camera angle, lighting and image quality
4. Fooled by identical twins

Future Scope

1. A method could be proposed to illustrate robustness against the variations that is, in near future, we could build a system which would be robust and would work in undesirable conditions too.
2. Here it is proposed for an institute to take the attendance of the students but in future it can be used to do the same work at entry as well as exit points.
3. In further work, authors intend to improve face recognition effectiveness by using the interaction among our system, the users and the administrators. On the other hand, our system can be used in a completely new dimension of face recognition application, mobile based face recognition, which can be an aid for common people to know about any person being photographed by cell phone camera including proper authorization for accessing a centralized database.

12. Bibliography

[1] Rekha A , Dr. Chethan H, Automated Attendance System Using Face Recognition Through Video Surveillance, International Journal For Technological Research In Engineering, Volume 1, Issue 11, July-2014.

[2] Nirmalya Kar, Mrinal Kanti Debbarma, Ashim Saha, and Dwijen Rudra Pal, Study Of Implementing Automated Attendance System Using Face Recognition Technique, International Journal Of Computer And Communication Engineering, Vol. 1, No. 2, July 2012.

[3] Naveed Khan Balcoh, M. HaroonYousaf, Waqar Ahmad And M. Iram Baig, Algorithm For Efficient Attendance Management: Face Recognition Based Approach, International Journal Of Computer Science Issues, Vol. 9, Issue 4, No 1, July 2012.

[4] Kwok-Wai Wong, Kin-Man Lam, Wan-Chi Siu, An efficient algorithm for human face detection and facial feature extraction under different conditions, The Journal Of the Pattern Recognition Society, 25 Aug-2000.

[5] Sarabjit Singh, Amritpal Kaur, Taqdir, A Face Recognition Technique using Local Binary Pattern Method, International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 3, March 2015.

[6] Learning OpenCV: Computer Vision with the OpenCV Library by Gary Rost Bradski O'REILLY