SMART ATTENDANCE SYSTEM USING FACE RECOGNITION

A PROECT REPORT

Submitted by

JAISURYA.S 61072011113

In partial fulfilment for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

GOVERNMENT COLLEGE OF ENGINEERING – BARGUR

(An Autonomous institution Affiliated to Anna University – Chennai)

APRIL-2023

BONAFIDE CERTIFICATE

Certified that this project report "Smart Attendance system using Face Recognition" is the Bonafide record of work done by

JAISURYA.S **61072011113**

Who carried the project work under the supervision.

SIGNATURE **SIGNATURE**

Dr. Dr. J. NAFEESA BEGUM. M.E., Ph.D. **Dr. S. SELVI. M.E., Ph.D.**

Head of the Department ,
Computer Science and Engineering,
Government College of Engineering,
Bargur– 635 104

Project guide,
Computer Science and Engineering,
Government College of Engineering,
Bargur – 635 104

Submitted for the project work and viva voce examination held on ……..

Internal Examiner **External Examiner**

ACKNOWLEDGEMENT

It is indeed a great pleasure and proud privilege to acknowledge the help and support we received from the positive minds around us in making this endeavor a successful one.

We express our hearty thanks to **Dr. S. SELVI M.E., Ph.D.** our project guide, Associate Professor, Department of Computer Science and Engineering, for his enthusiastic guidance, inspiring discussions, intellectual inspiration during the entire period of this project.

Our sincere thanks to **Dr. J. NAFEESA BEGUM. M.E., Ph.D**. of Computer Science and engineering who has given us both moral and technical support adding experience to the job we have undertaken.

We take enormous pleasure in thanking our principal **Dr. VIJAYAN.M.E., Ph.D.** for his encouragement and providing necessary facilities to carry put this project.

We also thank other staff members, a non-teaching staff member of our college, who have given their constant support and motivation in all our endeavors.

We heartily thank our family and friends for their precious support on each and every moment.

ABSTRACT

The maintenance the attendance record with day-to-day activities is a challenging task. The conventional method of calling name of each student is time consuming and there is always a chance of proxy attendance. The following system is based on face recognition to maintain the attendance record of students. Our system automatically starts taking snaps and then apply face detection and recognition technique to the given image and the recognize students are marked as present and their attendance update with corresponding time. We have used deep learning techniques to develop this system, histogram of oriented gradient method is used to detect faces in images and deep learning method is used to compute and compare feature facial of students to recognize them. Our system is capable to identify multiple faces in real time and record the entry time.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

Every organization requires a robust and stable system to record the attendance of their students. and every organization have their own method to do so, some are taking attendance manually with a sheet of paper by calling their names during lecture hours and some have adopted biometrics system such as fingerprint, Iris system to mark the attendance. The conventional method of calling the names of students manually is time consuming event. The RFID card system, each student assigns a card with their corresponding identity but there is chance of card loss or unauthorized person may misuse the card for fake attendance. While in other biometrics such as finger print, iris or voice recognition, they all have their own flaws and also, they are not 100% accurate

Use of face recognition for the purpose of attendance marking is the smart way of attendance management system. Face recognition is more accurate and faster technique among other techniques and reduces chance of proxy attendance. Face recognition provide passive identification that is a person which is to be identified does not to need to take any action for its identity

Face recognition involves two steps, first step involves the detection of faces and second step consist of identification of those detected face images with the existing database. There are number of face detection and recognition methods introduced. Face recognition works either in form of appearance based which covers the features of whole face or feature based which covers the geometric feature like eyes, nose, eye brows, and cheeks to recognize the face

1.1 RELATED WORK

In recent years, a number of faces recognition-based attendance management system have introduced in order to improve the performance of students in different organization. In Jomon Joseph, K. P. Zacharia proposed a system using image processing, PCA, Eigen faces, Microcontroller, based on MATLAB. Their system works only with front face images and there is need of a suitable method which works with the orientation of the system. Ajinkya Patil with their fellows in proposed a face recognition approach for attendance marking using Viola jones algorithm, Haar cascades are used to detect faces in images and recognition performs through Eigen face method. Another approach of making attendance system easy and secure, in the author proposed a system with the help of artificial neural networks, they used PCA to extract face images and testing and training were achieved by neural networks, their system performs in various orientation

Efficient Attendance Management system is designed with the help of PCA algorithm, the have achieved accuracy up to 83% but their system performance decreases due to slightly changes in light condition. An eigen face approach along with PCA algorithm for marking face recognition attendance system have introduced by author in, they mention comparison of different face recognition algorithm in their paper. Overall, it was good approach to maintain record of attendance.

1.2 METHODOLOGY

The proposed system is designed for automating the attendance of the different organization and reduces the flaws of existing manual system. The system calculates the attendance subject wise, that is the data of students and subjects are added manually by administrator, and whenever time for corresponding subject arrives the system automatically starts taking snaps and find whether human faces are appear in the given image or not. We have used Histogram of Oriented Gradient for face detection and deep learning techniques to calculate and compare 128-d face features for face recognition. Once faces are detected and recognize with the existing database,

system calculate attendance for the recognize students with the respective subject id in real time. And an excel sheet generated and saved by the system automatically.

Our system splits into two parts, First the front-end side which consist of GUI which is based on Electron JS that is JavaScript stack which is serving as a client and the second is the backend side which consist of logic and based on Python which is serving as a server. And we know that both the languages cannot communicate with each other directly so we have used

IPC (Inter Personal Communication) techniques with zero library as a bridge to communicate these two languages. The Electron JS call the python functions and interchange data via TCP with help of Zero PC Library

1.3 DATA ACQUISITION

Data acquisition is the process of sampling signals that measure real world physical conditions and converting the resulting samples to digital numeric values that can be manipulated by a computer.

1.3.1 IMAGE ACQUISITION

Image is acquire using a high-definition camera which is placed in the classroom. This image is given as an input to the system.

1.3.2 DATASET CREATION

Dataset of students is created before the recognition process. Dataset was created only to train this system. We have created a dataset of 5 students which involves their name, roll number, department and images of student in different poses and variations. For better accuracy minimum 15 images of each student should be captured. Whenever we register student’s data and images in our system to create dataset, deep learning applies to each face to compute 128d facial features and store in student face data file to recall that face in recognition process. This process is applying to each image taken during registration

1.3.3 STORING

We have used JSON to store the student's data. and undefined data form. If you are trying to store or exchange data in functions or dates than JSON is not right choice for you.

1.4 FACE RECOGNITION PROCESS

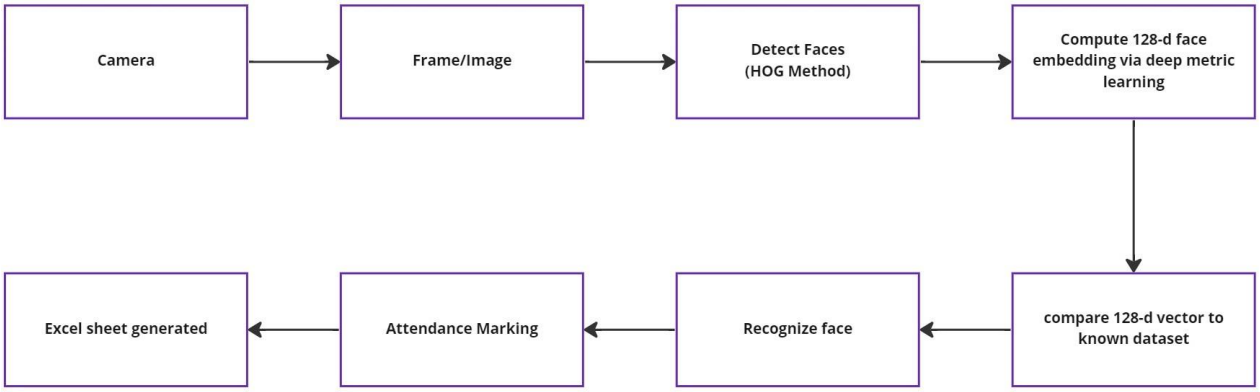


Fig:1.1 face recognition process

CHAPTER 2

LITERATURE REVIEW

2.1 SMART ATTENDANCE MANAGEMENT SYSTEM USING FACE RECOGNITION

Kaneez Laila Bhatti1 et al., has proposed a system which uses face recognition approach to reduce the flaws of existing system with the help of machine learning, it requires a good quality camera to capture the images of students, the detection process is done by histogram of oriented gradient. And recognizing perform through deep learning. The frontend side (client side) which consist of GUI which is based on electron JS and backend side consist of logic and python (server side), an IPC (Inter Personal Communication) bridge is developed to communicate these two stacks. The images capture by the camera is sent to system for further analysis, the input image is then compared with a set of reference images of each of the student and mark their attendance.

Face detection is important as the image taken through the camera given to the system, face detection algorithm applies to identify the human faces in that image, the number of image processing algorithms are introduced to detect faces in an image and also the location of that detected faces. We have used HOG method to detect human faces in given image.

Once the faces are detected in the given image, the next step is to extract the unique identifying facial feature for each image. Basically, whenever we get localization of face, the 128 key facial point are extracted for each image given input which are highly accurate and these 128-d facial points are stored in data file for face recognition.

This is last step of face recognition process. We have used the one of the best learning techniques that is deep metric learning which is highly accurate and capable of

outputting real value feature vector. Our system ratifies the faces, constructing the 128- d embedding (ratification) for each. Internally compare faces function is used to compute the Euclidean distance between face in image and all faces in the dataset. If the current image is matched with the 60% threshold with the existing dataset, it will move to attendance marking.

Once the face is identified with the image stored in JSON file, python generate roll numbers of present students and return that, when data is returned, the system generates attendance table which includes the name, roll number, date, day and time with corresponding subject id. And then passes the data to python to store the table into an excel sheet automatically.

The system performs satisfactory in different poses and variations. In future this system needs, be improved because this system sometimes fails to recognize students from some distance, also we have some processing limitation, working with a system of high processing may result even better performance of this system.

2.2 STUDY OF IMPLEMENTING AUTOMATED ATTENDANCE SYSTEM USING FACE RECOGNITION TECHNIQUE

Nirmalya Kar et al., has proposed a research paper, The present authors used the eigenface approach for face recognition which was introduced by Kirby and Sirovich in 1988 at Brown University. The method works by analysing face images and computing eigenface which are faces composed of eigenvectors. The comparison of eigenface is used to identify the presence of a face and its identity. There is a five-step process involved with the system developed by Turk and Pentland. First, the system needs to be initialized by feeding it a set of training images of faces. This is used to define the face space which is set of images that are face like. Next, when a face is encountered it calculates an eigenface for it.

By comparing it with known faces and using some statistical analysis it can be determined whether the image presented is a face at all. Then, if an image is

determined to be a face the system will determine whether it knows the identity of it or not. The optional final step is that if an unknown face is seen repeatedly, the system can learn to recognize it. The two main components used in the implementation approach are open-source computer vision library (OpenCV) and Light Tool Kit (FLTK). One of OpenCV's goals is to provide a simple-to-use computer vision infrastructure that helps people build fairly sophisticated vision applications quickly.

OpenCV library contains over 500 functions that span many areas in vision. The primary technology behind Face recognition is OpenCV; the interface is designed using

FLTK. The user stands in front of the camera keeping a minimum distance of 50cm and his image is taken as an input. The frontal face is extracted from the image then converted to grey scale and stored. The output from cvEigenDecomposite () is stored in a local variable - projectedTestFace. Because there's no need to store the projected test image, we used a C array for projectedTestFace, rather than an OpenCV matrix.

The 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.

2.3ATTENDANCE MANAGEMENT USING FACIAL RECOGNITION

Bhautik, Gondaliya et al., has proposed Automatic face recognition (AFR) technologies have made many improvements in the changing world. Smart Attendance using Real-Time Face Recognition is a real-world solution which comes with day-to-day activities of handling student attendance system. Face recognition-based attendance system is a process of recognizing the students face for taking attendance by using face biometrics based on high - definition monitor video and other information technology.

In this face recognition project, a computer system will be able to find and recognize human faces fast and precisely in images or videos that are being captured through a surveillance camera. Numerous algorithms and techniques have been developed for improving the performance of face recognition but the concept to be implemented here is Deep Learning. It helps in conversion of the frames of the video into images so that the face of the student can be easily recognized for their attendance so that the attendance database can be easily reflected automatically.

The facial recognition technology can be used in recording the attendance through a high-resolution digital camera that detects and recognizes the faces of the students and the machine compares the recognized face with students' face images stored in the database. Once the face of the student is matched with the stored image, then the attendance is marked in attendance database for further calculation.

There is no need for the teacher to manually take attendance in the class because the system records a video and through further processing steps the face is being recognized and the attendance database is updated.

2.4 SMART ATTENDANCE MARKING SYSTEM USING FACIAL RECOGNITION

Neela Ashish Kumar et al., has proposed that, a model for attendance management system. This system requires various techniques which are already existed and can use with some modifications. The first step is to design such a system on the facial recognition. The functioning of the algorithms needs to be understood so that the face detection dynamics can be understandable. This system can work most efficiently as we are using the engine's face detection algorithm. It actually divides the image into smaller parts. It will be easy to compare the smaller parts rather than covering the whole face.

By the implementation of the algorithm, we were able to track the face of the student. It can be done three ways i.e., head tracking, facial feature tracking, complete tracking. It has high accuracy and can be used multiple times. It will be very useful in the online examinations and also for marking the attendance in the classrooms automatically.

A proper and efficient algorithm always enhances the efficiency of the face recognition systems. Various algorithms have been proposed such as Face geometry methods, Feature Invariant methods, Machine learning based methods etc. Among them, we used Viola and joined detection algorithm which gives high detection rate and also fast. This algorithm is efficient for real-time application for its efficiency. Viola-Jones algorithm is applied to this frame, which detects the faces in the frame. To ensure that the detected object is facing, each detected object is cropped and further processed for eye detection and if eyes are detected they are considered as faces else are rejected. Features of all the faces are extracted using different facial features such as the nose, eyes, the distance between eyes, etc.

The detected face is extracted and subjected to pre-processing. This pre-processing step involves with histogram equalization of the extracted face image and

is resized to 100x100. Histogram Equalization is the most common Histogram Normalization technique. This improves the contrast of the image as it stretches the range of the intensities in an image by making it clearer.

This database development phase consists of image capture of every individual and extracting the biometric feature, in our case, it is facing, and later it is enhanced using pre-processing techniques and stored in the database. In our project, we have taken the images of individuals in different angles, different expressions and also in different lighting conditions.

The feature selection and extraction of the captured image is the important module of the total attendance recognition system. For a reliable, compatible and error-free face recognition system we need good feature extraction attributed and good classification and detection algorithm. The performance of a face recognition mainly depends upon good classification algorithm. Now here we used Viola-jones Algorithm for face detection as it is the most accurate face detection algorithm for real-time application.

The main characteristics of the viola-jones algorithm are its very high detection rate and very low false positive rate. Here we take the implementation of HAAR features. The few properties are very common to human faces such as eye region is darker than the upper cheeks and nose bridge is brighter than the eyes. Here we use PCA for feature extraction.

CHAPTER 3

SOFTWARE DESCRIPTION

3.1 FRONTEND

Front-end web development is the development of the graphical user interface of a website, through the use of HTML, CSS, and JavaScript, so that users can view and interact with that website.

3.1.1 HTML

HTML stands for Hypertext Markup Language. It is used to design web pages using a markup language. HTML is the combination of Hypertext and Markup language. Hypertext defines the link between web pages. A markup language is used to define the text document within the tag which defines the structure of web pages. This language is used to annotate (make notes for the computer) text so that a machine can understand it and manipulate text accordingly. Most markup languages (HTML) are human-readable. The language uses tags to define what manipulation has to be done on the text**.**

3.1.2 CSS

Cascading Style Sheets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable.CSS handles the look and feel part of a web page. Using CSS, you can control the colour of the text, the style of fonts, the spacing between paragraphs, how resolutions in display for different devices and screen sizes as well as a variety of other effects.

3.1.3 JAVASCRIPT

JavaScript is a light-weight object-oriented programming language which is used by several websites for scripting the webpages. It is an interpreted, full-fledged programming language that enables dynamic interactivity on websites when applied to an HTML document. It has been adopted by all other graphical web browsers.

3.2 LIBRARY USED FOR FACE RECOGNITION

3.2.1 PYTHON

Python is a high-level, general-purpose, self-contained programming language designed to meet the needs of computer scientists, software developers, and college students interested in coding. It is also known as an object-oriented programming (OOP) language. The main benefits of using Python instead of other languages are that it is very easy to start programming with and because of its flexible syntax, it can be used to program almost any kind of software application. Python is an open-source language that’s free to use and has a wide range of features that make it easy to customize.

3.3 WEB SERVER

Back-end Development refers to the server-side development. It focuses on databases, scripting, website architecture. It contains behind-the-scene activities that occur when performing any action on a website. It can be an account login or making a purchase from an online store. Code written by back-end developers helps browsers to communicate with database information.

3.3.1 FLASK

Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.

CHAPTER 4

PROBLEM DESCRIPTION

4.1 PROBLEM STATEMENT

Monitoring attendance is an important part of all organizations. At the beginning and ending of the work day, it is usually registered by bio-metrics sensors or RFID tags but it may pose problems like taking more time for large number of employees to register attendance, upfront cost etc. Face recognition-based attendance system is a problem of recognizing face for taking attendance by using a user's own personal device and its camera functionalities.

The concept of face recognition is to give a computer system the ability of finding and recognizing human faces fast and precisely in images or videos. Numerous algorithms and techniques have been developed for improving the performance of face recognition. Recently Deep learning has been highly explored for computer vision applications. Human brain can automatically and instantly detect and recognize multiple faces. But when it comes to computer, it is very difficult to do all the challenging tasks on the level of human brain. The face recognition is an integral part of biometrics. In biometrics, basic traits of human are matched to the existing data. Facial features are extracted and implemented through algorithms, which are efficient and some modifications are done to improve the existing algorithm models. Computers that detect and recognize faces could be applied to a wide variety of practical applications including criminal identification, security systems, identity verification etc.

4.2 OVERVIEW OF THE PROJECT

The project aims at creating a face recognition-based attendance system that is lightweight and can be operated from an employee's own device. After the one-time registration the user's data along with the facial features are stored in the system. The user is then able to mark attendance from their own device using the device camera and also able to view their attendance being marked in real-time. The web application uses facial recognition package from python and runs on a flask supported server

CHAPTER 5
WORK FLOW

5.1 NETWORK CONSTRUCTION:

Given figure below is the model of the attendance monitoring system and how it will be implemented in a particular class. As we can see that there is also a teacher's desk who will be facing the students hence, he will not be considered as a student. A camera is setup in the middle of the class room at a suitable height to get the full view of the class till the last bench. After the students have been seated.

fig.5.1 Process Flow

The camera will take an image and starts the process of face detection using the techniques and methods discussed in the methodology section. After this the program will automatically make a folder in the database having the students to be recognized. The already placed images of each student is taken and used from database for image recognition. The images will be fetched and compared with each of the entry in the database and hence will be checked whether the student is present in the class or not. If there is no match the program will move to the next picture.

5.2 MAIN ROUTE DISCOVERY

Despite the complexity of the underlying system, there are three general steps of face recognition based smart attendance marking system: -

1. Finding human faces in real-time is crucial in a smart student attendance system.
2. After the analogue facial image is captured, the captured data or vectors are transformed into data or vectors based on the person's facial characteristics.
3. For verification, the face recognition based smart attendance marking system compares the data above with the data in the database.

Then next we have to train the model. The is trained with the images extracted from the video. There is a pre-processing step, that is taken to extract features from the video and to generate variations or alignments from the source video. file. Then these extracted features are dumped to a.pkl file, with names and images mapped properly using pic

5.3 BACKUP NODE LOCALIZATION AND CONSTRUCTION

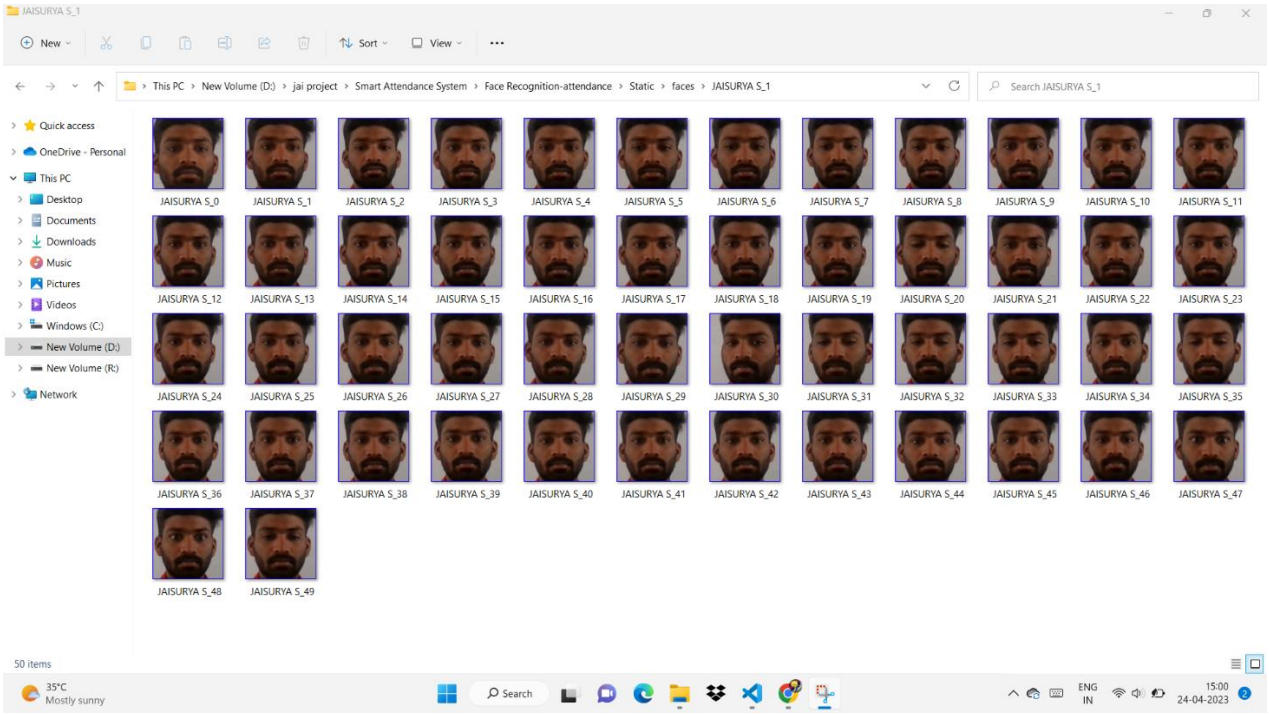


fig:5.2 local storage

Use of face recognition for the purpose of attendance marking is the smart way of attendance management system. Face recognition is more accurate and faster technique among other techniques and reduces chance of proxy attendance. Face recognition provide passive identification that is a person which is to be identified does not to need to take any action for its identity.

Face recognition involves two steps, first step involves the detection of faces and second step consist of identification of those detected face images with the existing database. There are number of face detection and recognition methods introduced. Face recognition works either in form of appearance based which covers the features of whole face or feature based which covers the geometric feature like eyes, nose, eye brows, and cheeks to recognize the face

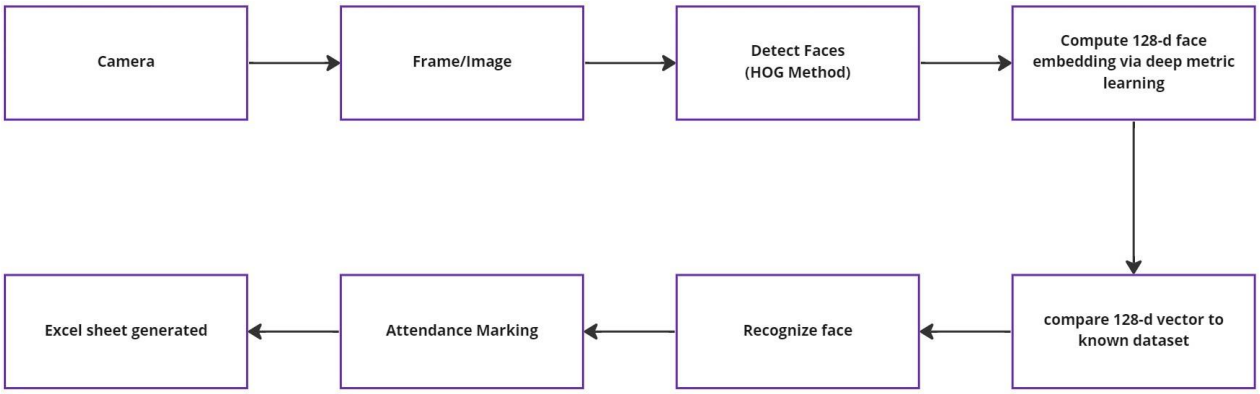


fig.5.3. Visualization Map

CHAPTER 6

SYSTEM IMPLEMENTATION

6.1 SYSTEM ARCHITECTURE

Once the face is identified with the image stored in JSON file, python generate roll numbers of present students and return that, when data is returned, the system generates attendance table which includes the name, roll number, date, day and time with corresponding subject id. And then passes the data to python to store the table into an excel sheet automatically. Each sheet is saved according to the subjects which already entered by the administrator, for example when system generates excel sheet by sending the compiled sheet in an array to python, the python first checks whether there exit any excel sheet of that date, if yes then it creates separate worksheet by subject id, so that attendance is differentiated for different subjects

6.1.1 FACE DETECTION AND EXTRACTION

Face detection is important as the image taken through the camera given to the system, face detection algorithm applies to identify the human faces in that image, the number of image processing algorithms are introduced to detect faces in an image and also the location of that detected faces. We have used HOG method to detect human faces

6.1.2 FACE POSITIONING

There are 68 specific points in a human face. In other words, we can say 68 face landmarks. The main function of this step is to detect landmarks of faces and to position the image. A python script is used to automatically detect the face landmarks and to position the face as much as possible without distorting.

6.1.3 FACE ENCODING

Once the faces are detected in the given image, the next step is to extract the unique identifying facial feature for each image. Basically, whenever we get

localization of face, the 128 key facial point are extracted for each image given input which are highly accurate and these 128-d facial points are stored in data file for face recognition.

6.1.4 FACE MATCHING

This is last step of face recognition process. We have used the one of the best learning techniques that is deep metric learning which is highly accurate and capable of outputting real value feature vector. Our system ratifies the faces, constructing the 128-d embedding (ratification) for each. Internally compare face's function is used to compute the Euclidean distance between face in image and all faces in the dataset. If the current image is matched with the 60% threshold with the existing dataset, it will move to attendance marking

6.2 DATA TRANSFER

The active device is utilized to produce an electromagnetic field of a given radius and strength. Which used to implement an attendance system. In a school setting for example, students can be given NFC tags that are uniquely programmed with their unique identification numbers. Upon attending the classes, the lecturers bring the readers and a student is required to swipe their NFC tags near the reader, say the lecturers' phone.

This information is then transmitted to the school database to mark the attendance of the student. However, this system is vulnerable to impersonation where one person can sign in for someone else. The other related systems that use biometrics (Face recognition, etc) to identify end user are time management systems used in many colleges, institutions and schools. However, these systems introduce further privacy concerns. These systems are also subject to physical damage from their users. Therefore, they need additional maintenance costs. The idea proposed by us, Removes physical access from anyone to the automated system.

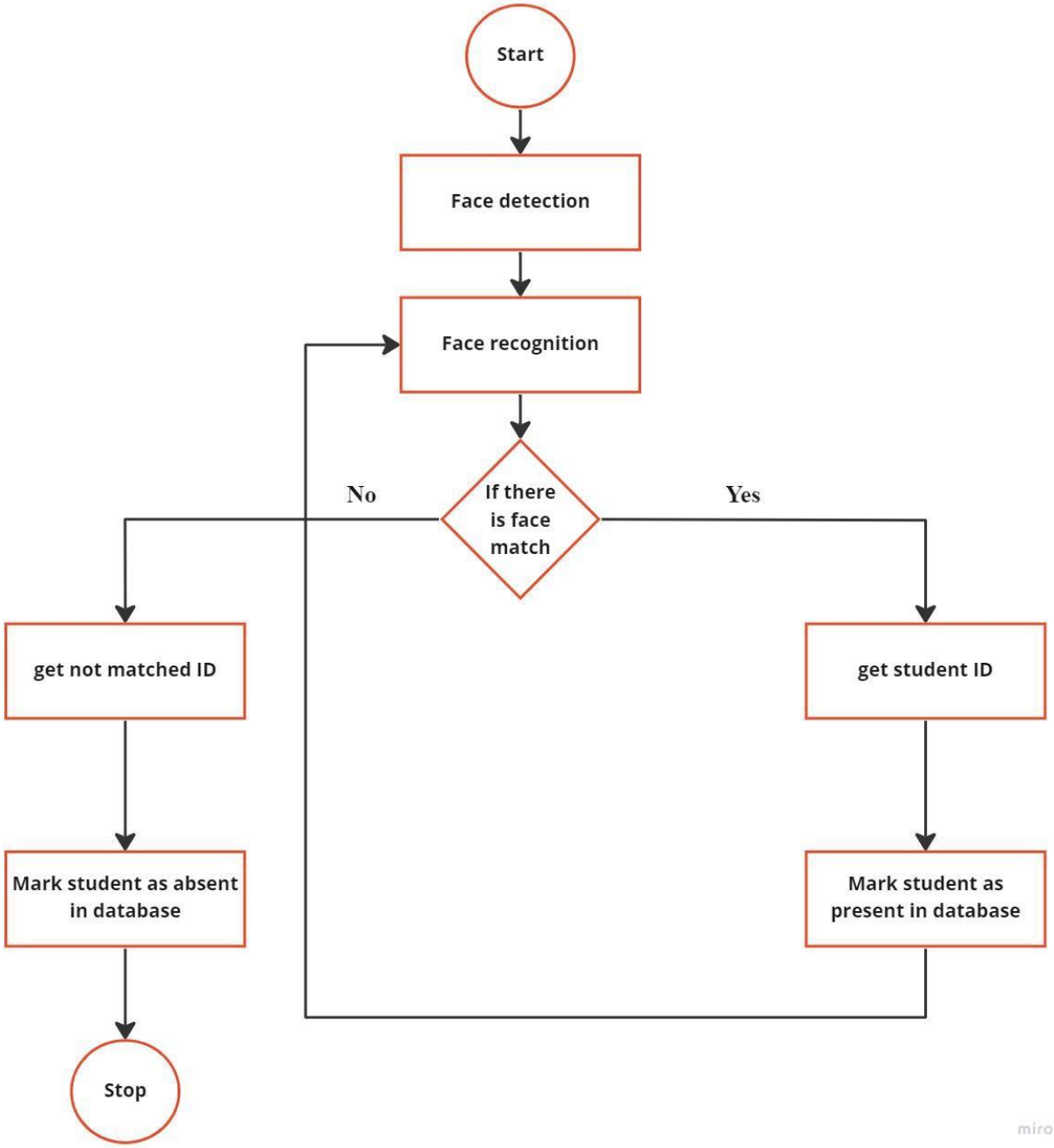


fig.6.1 Mind Map

6.3 DATA INTEGRATION

The function of the proposed system is to capture the face of every student and keep it on the website for them to attend. The face of the expert must be taken in such a way that everyone can see what the student's face is like, even the seating area and the way the students stand. There is no need for the teacher to be physically present in

the classroom because the system records the video and then with continuous face processing steps is monitored and therefore the attendance website is updated.

All Individuals in the category must register themselves by entering the information they require so that their photos can be taken and stored within the database. During each session, a face is found in the live streaming video of the class. The recovered faces are compared with the existing images in the database. Once a match is found, attendees are marked on the appropriate reader. At the top of each session, a list of absentees will be sent to the appropriate faculty in charge of the session.

Biometric detection technology will be used to record attendees with a high-definition camera that detects individual faces so the machine compares known faces with student faces stored within the website. When the face of the code is matched to a saved image, attendees are marked with the current website for further calculation. If the captured image does not match the existing student face on the website, this image is saved as a new image on the website.

6.3.1 RECOGNITION AND DETECTION

In every session, as soon as you run the program the camera starts automatically and starts detecting the face in the frame. If the face detected is registered in the database, the system recognizes the face and shows his/her name on the screen. If the face is not registered the program will quit automatically.

6.4 DATA INTEGRITY

Nowadays**,** almost all top schools use the face recognition school smart attendance management system. Face recognition school attendance is an essential system in schools or companies to be identified, verified, and recorded as being present.

Smart attendance management systems using fingerprint or smart card attendance systems have almost become the standard. Still, a recent pandemic outbreak has brought to light the problem of solutions that require physical contact. Artificial Intelligence-based attendance systems are contactless technologies that eliminate any material connection between the machine and the person because they use smart attendance systems using face recognition. We can better understand how a smart attendance management system can improve the safety and efficiency of buildings, schools, etc. when we know how the technology works.

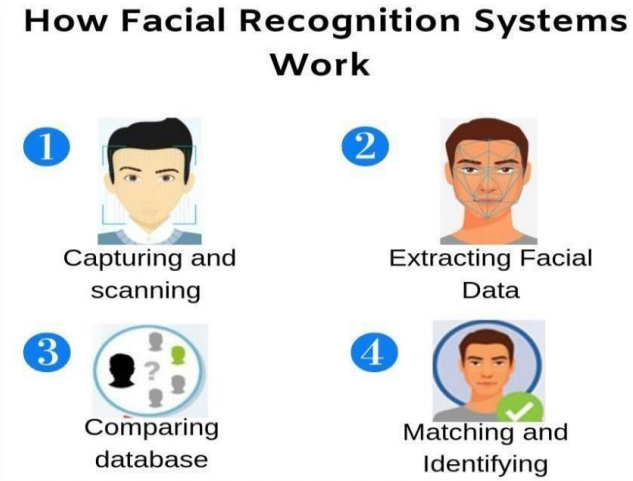


fig.6.2 Facial Recognition System

The Automated Attendance System is used in larger areas such as the seminar where it helps to feel the presence of more people. Sometimes poor classroom lighting may affect image quality that indirectly reduces system performance, this may be overcome in the later phase by improving video quality or using certain algorithm.

CHAPTER 7

SYSTEM DESIGN

7.1 INPUT DESIGN

The goal of designing input data is to make data entry as easy, logical and error free from errors as possible. In entering data, operators need to know the following: The allocated space for each field. Field sequence, which much match that in the source document. The input datasets for the application to analyse is integrated into the application in such a way that user is able to give the input in real time. The system is able to capture 50 different images of the users face as datasets and save it in the application. These 50 images are used to analyse the facial features and create data point that can be used to recognize the user in future.

7.1.1 Advantages of this input design:

- Able to give user input in real time.
- User id and name can also be entered on time.
- User can able to mark attendance by using the facial recognition.
- Image is recorded using the user's own device camera.

7.2 OUTPUT DESIGN

Output design generally refers to the results and information that are generated by the system for many end-users; it should be understandable with the enhanced format. The Output of the software is used to make the remote installation of the new software in the system and, it is awake the immediate alert to the system that should be enhanced it as the input to the system. Output is the main reason for developing the system and the basis on which they evaluate the usefulness of the application.

7.2.1 Advantages of this output design:

- The system uses the recorded user image and compares it with the image datasets for similarity, and the user is only given attendance once his image is being recognised.
- The user's attendance is being updated immediately upon matching of face.
- The users name, user id along with the entry time is being is displayed in the attendance table.
- Output of the total number person whose data have been saved also being displayed.

7.3 EXPERIMENTAL SETUP

The system is being hosted in a local host server in flask and being run on the hosts system for the experimental purposes. The user opens up the designated host port in order to launch the flask application. The user is able to Register with name register number and his face. After the registration the count of registered users is being increased.

After the completion of the registration process the user can Mark their attendance by opening the application and clicking the mark attendance button. As soon as the button is clicked the host devices camera is opened and scans for users face, After successfully recognizing the users face the attendance is marked In the table below with name register number and in time. The attendance is being recorded in a CSV file that is being saved in the hosts system and can be viewed as a excel file.

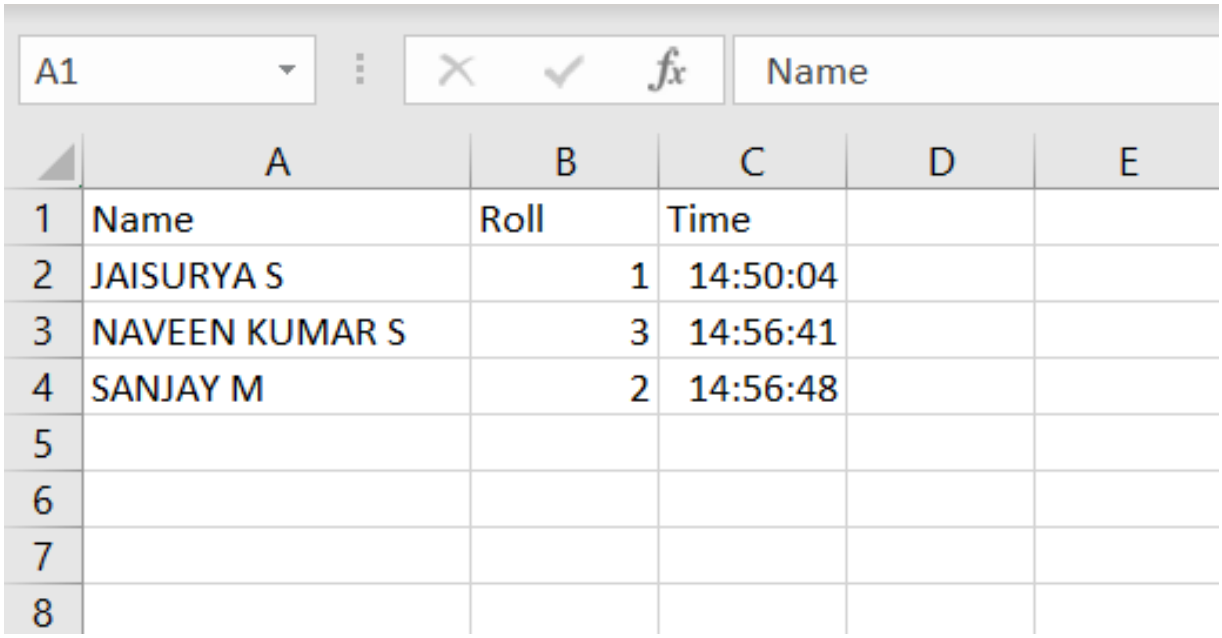


fig 7.1: Excel sheet to store data

CHAPTER 8
CONCLUSION AND FUTURE ENHANCEMENT

8.1 CONCLUSION

In this approach, a face recognition based smart attendance system is thoroughly described. The proposed approach provides a method to identify the individuals by comparing their input image obtained from recording video frame with respect to trained image. The trained images are taken when the user registers in the profile. Extraction of features from the facial image is performed by Face recognition package from flask. The algorithm is designed to stabilize the system by giving consistent results. The accuracy of this proposed approach is 80 % for high-quality images, 75 % for low-quality images and 95.76 % of face database when two images per person are trained. As a conclusion for analysis, the extraction of facial feature could be challenging especially in different lighting.

8.2 FUTURE ENHANCEMENT

There is a big scope of taking the findings of the report further. This model is developed for big organizations, schools, colleges and corporate business. This concrete idea is generated to solve the current devices i.e., biometric, id card reader since the time efficiency will be less for the employees and marking the attendance will be delayed.

For now, the proposed project provides a better accuracy for a period of time and when a better application is found it is always the common thing in the modern generation to make a way for the future generations. And we hope that we get better changes than the proposed in the future.

APPENDIX 1

SOURCE CODE

MAIN :

```
Import cv2
import os
from flask import Flask,request,render_template
from datetime import date
from datetime import datetime
import numpy as np
from sklearn.neighbors import KNeighborsClassifier
import pandas as pd
import joblib
#### Defining Flask App
app = Flask(__name__)
#### Saving Date today in 2 different formats
datetoday = date.today().strftime("%m_%d_%y")
datetoday2 = date.today().strftime("%d-%B-%Y")
#### Initializing VideoCapture object to access WebCam
face_detector = cv2.CascadeClassifier('static/haarcascade_frontalface_default.xml')
cap = cv2.VideoCapture(0)
#### If these directories don't exist, create them
if not os.path.isdir('Attendance'):
    os.makedirs('Attendance')
if not os.path.isdir('static/faces'):
    os.makedirs('static/faces')
if f'Attendance-{datetoday}.csv' not in os.listdir('Attendance'):
    with open(f'Attendance/Attendance-{datetoday}.csv','w') as f:
        f.write('Name,Roll,Time')
#### get a number of total registered users
```

```
def totalreg():
    return len(os.listdir('static/faces'))
#### extract the face from an image
def extract_faces(img):
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    face_points = face_detector.detectMultiScale(gray, 1.3, 5)
    return face_points
#### Identify face using ML model
def identify_face(facearray):
    model = joblib.load('static/face_recognition_model.pkl')
    return model.predict(facearray)
#### A function which trains the model on all the faces available in faces folder
def train_model():
    faces = []
    labels = []
    userlist = os.listdir('static/faces')
    for user in userlist:
        for imgname in os.listdir(f'static/faces/{user}'):
            img = cv2.imread(f'static/faces/{user}/{imgname}')
            resized_face = cv2.resize(img, (50, 50))
            faces.append(resized_face.ravel())
            labels.append(user)
    faces = np.array(faces)
    knn = KneighborsClassifier(n_neighbors=5)
    knn.fit(faces,labels)
    joblib.dump(knn,'static/face_recognition_model.pkl')
#### Extract info from today's attendance file in attendance folder
def extract_attendance():
    df = pd.read_csv(f'Attendance/Attendance-{datetoday}.csv')
```

```
    names = df['Name']
    rolls = df['Roll']
times = df['Time']
    l = len(df)
    return names,rolls,times,l
#### Add Attendance of a specific user
def add_attendance(name):
    username = name.split('_')[0]
    userid = name.split('_')[1]
    current_time = datetime.now().strftime("%H:%M:%S")
    df = pd.read_csv(f'Attendance/Attendance-{datetoday}.csv')
    if int(userid) not in list(df['Roll']):
        with open(f'Attendance/Attendance-{datetoday}.csv','a') as f:
            f.write(f'\n{username},{userid},{current_time}')
@app.route('/')
def home():
    names,rolls,times,l = extract_attendance()
    return
render_template('home.html',names=names,rolls=rolls,times=times,l=l,totalreg=tota
lreg(),datetoday2=datetoday2)
#### This function will run when we click on Take Attendance Button
@app.route('/start',methods=['GET'])
def start():
    if 'face_recognition_model.pkl' not in os.listdir('static'):
        return
render_template('home.html',totalreg=totalreg(),datetoday2=datetoday2,mess='Ther
e is no trained model in the static folder. Please add a new face to continue.')
    cap = cv2.VideoCapture(0)
    ret = True
```

```
    while ret:
        ret,frame = cap.read()
        if extract_faces(frame)!=():
            (x,y,w,h) = extract_faces(frame)[0]
            cv2.rectangle(frame,(x, y), (x+w, y+h), (255, 0, 20), 2)
              face = cv2.resize(frame[y:y+h,x:x+w], (50, 50))
            identified_person = identify_face(face.reshape(1,-1))[0]
            add_attendance(identified_person)
cv2.putText(frame,f'{identified_person}',(30,30),cv2.FONT_HERSHEY_SIMPLE
X,1,(255, 0, 20),2,cv2.LINE_AA)
        cv2.imshow('Attendance',frame)
        if cv2.waitKey(1)==27:
            break
    cap.release()
    cv2.destroyAllWindows()
    names,rolls,times,l = extract_attendance()
    return
render_template('home.html',names=names,rolls=rolls,times=times,l=l,totalreg=tota
lreg(),datetoday2=datetoday2)
#### This function will run when we add a new user
@app.route('/add',methods=['GET','POST'])
def add():
    newusername = request.form['newusername']
    newuserid = request.form['newuserid']
    userimagefolder = 'static/faces/'+newusername+'_'+str(newuserid)
    if not os.path.isdir(userimagefolder):
        os.makedirs(userimagefolder)
    cap = cv2.VideoCapture(0)
    I,j = 0,0
```

```
    while 1:
        _,frame = cap.read()
        faces = extract_faces(frame)
        for (x,y,w,h) in faces:
            cv2.rectangle(frame,(x, y), (x+w, y+h), (255, 0, 20), 2)
            cv2.putText(frame,f'Images Captured:
{i}/50',(30,30),cv2.FONT_HERSHEY_SIMPLEX,1,(255, 0, 20),2,cv2.LINE_AA)
            if j%10==0:
                name = newusername+'_'+str(i)+'.jpg'
                cv2.imwrite(userimagefolder+'/'+name,frame[y:y+h,x:x+w])
                i+=1
            j+=1
          if j==500:
            break
        cv2.imshow('Adding new User',frame)
        if cv2.waitKey(1)==27:
            break
    cap.release()
    cv2.destroyAllWindows()
    print('Training Model')
    train_model()
    names,rolls,times,l = extract_attendance()
    return
render_template('home.html',names=names,rolls=rolls,times=times,l=l,totalreg=tota
lreg(),datetoday2=datetoday2)
#### Our main function which runs the Flask App
if __name___== '__main__':
        app.run(debug==True)
```

HOME PAGE:

```
< !doctype html>
<html lang= »en »>
<style type=’text/css’>
   * {
      padding: 0;
      margin: 0;
      font-family: ‘Segoe UI’, Tahoma, Geneva, Verdana, sans-serif;
   }
 body {
      background-image: url(‘https://cutewallpaper.org/21/1920-x-1080-
gif/1920x1080-Wallpapercartoon-Wallpapers-Driverlayer-Search-.gif’);
      background-size: cover;
      font-family: sans-serif;
      margin-top: 40px;
      height: 100vh;
      padding: 0;
      margin: 0;
   }
   table {
      border: 1px;
      font-family: arial, sans-serif;
      border-collapse: collapse;
      width: 86%;
      margin: auto;
   }
   td,
   th {
```

```
        border: 1px solid black !important;
        padding: 5px;
    }
    tr:nth-child(even) {
        background-color: #dddddd;
    }
</style>
<head>
    <!—Required meta tags ➔
    <meta charset=”utf-8”>
    <meta name=”viewport” content=”width=device-width, initial-scale=1”>
    <link rel=”stylesheet”
href=”https://fonts.googleapis.com/icon?family=Material+Icons”>
    <!—Bootstrap CSS ➔
    <link href=”https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-
beta3/dist/css/bootstrap.min.css” rel=”stylesheet”
        integrity=”sha384-
eOJMYsd53ii+scO/bJGFsiCZc+5NDVN2yr8+0RDqr0Ql0h+Rp48ckxlpbzKgwra6”
crossorigin=”anonymous”>
    <title>Face Recognition Based Attendance System</title>
</head>
<body>
    <div class=’mt-3 text-center’>
        <h1 style=”width: auto;margin: auto;color: white;padding: 11px;font-size:
44px;”>Face Recognition Based
            Attendance System</h1>
    </div>
    <div class=’mt-3 text-center’>
```

```
        <h3 style="font-size: 22px;color:beige;">{{ datetoday2 }} | <span
id="clock"></span></h3>
    </div>
    {% if mess%}
    <p class="text-center" style="color: red;font-size: 20px;">{{ mess }}</p>
    {% endif %}
    <div class="row text-center" style="padding: 20px;margin: 20px;">
        <div class="col"
            style="border-radius: 20px;padding: 0px;background-
color:rgb(211,211,211,0.5);margin:0px 10px 10px 10px;min-height: 400px;">
            <h2 style="border-radius: 20px 20px 0px 0px;background-color:
#0b4c61;color: white;padding: 10px;">Today's
                Attendance <I class="material-icons">assignment</i></h2>
            <a style="text-decoration: none;max-width: 300px;" href="/start">
                <button
                    style="font-size: 24px;font-weight: bold;border-radius:
10px;width:490px;padding: 10px;margin-top: 30px;margin-bottom: 30px;"
                    type='submit' class='btn btn-primary'>Take Attendance <i
                        class="material-icons">beenhere</i></button>
            </a>
            <table style="background-color: white;">
                <tr>
                    <td><b>S No</b></td>
                    <td><b>Name</b></td>
                    <td><b>ID</b></td>
                    <td><b>Time</b></td>
                </tr>
                {% if l %}
    {% for I in range(l) %}
```

```
            <tr>
                <td>{{ i+1 }}</td>
                <td>{{ names[i] }}</td>
                <td>{{ rolls[i] }}</td>
                <td>{{ times[i] }}</td>
            </tr>
            {% endfor %}
            {% endif %}
        </table>
</div>
<div class="col"
        style="border-radius: 20px;padding: 0px;background-
color:rgb(211,211,211,0.5);margin:0px 10px 10px 10px;height: 400px;">
        <form action='/add' method="POST" enctype="multipart/form-data">
            <h2 style="border-radius: 20px 20px 0px 0px;background-color:
#0b4c61;color: white;padding: 10px;">Add
                New User <I class="material-icons">control_point_duplicate</i></h2>
            <label style="font-size: 20px;"><b>Enter New User Name*</b></label>
            <br>
            <input type="text" id="newusername" name='newusername'
                style="font-size: 20px;margin-top:10px;margin-bottom:10px;"
required>
            <br>
            <label style="font-size: 20px;"><b>Enter New User Id*</b></label>
            <br>
            <input type="number" id="newusereid" name='newuserid'
                style="font-size: 20px;margin-top:10px;margin-bottom:10px;"
required>
            <br>
```

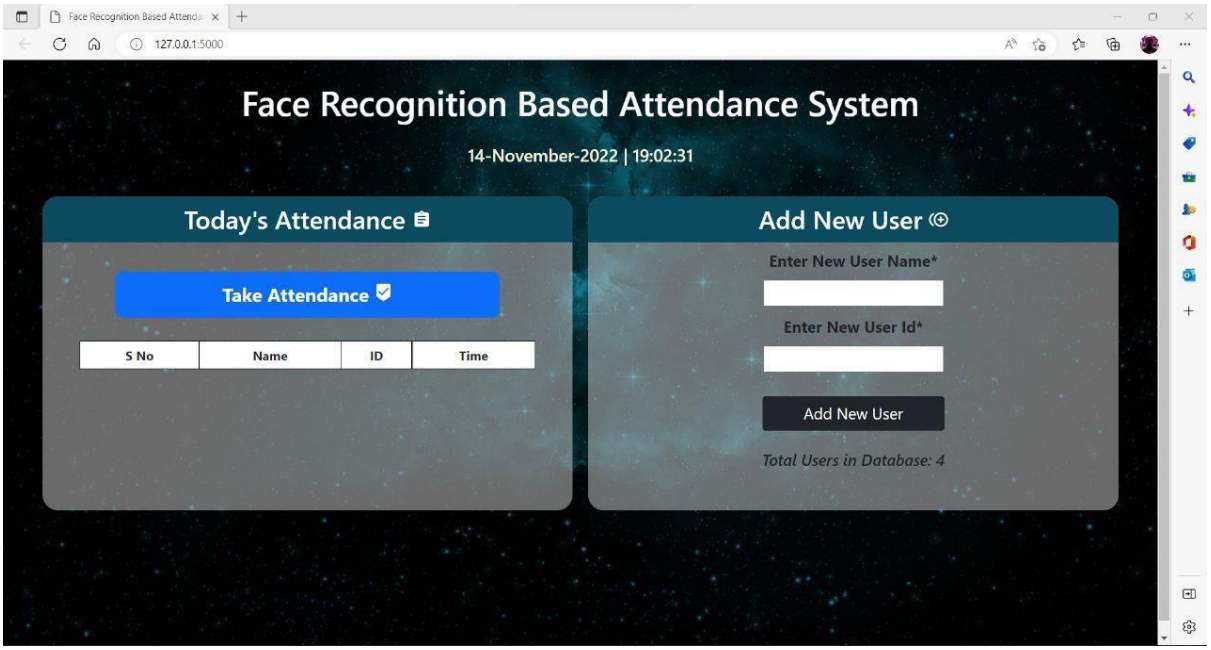
```
            <button style="width: 232px;margin-top: 20px;font-size: 20px;"
type='submit' class='btn btn-dark'>Add
                New User
            </button>
        <br>
            <h5 style="padding: 25px;"><i>Total Users in Database:
{{totalreg}}</i></h5>
          </form>
        </div>
    </div>
    <script type="text/javascript">
        var clockElement = document.getElementById('clock');

        function clock() {
            clockElement.textContent = new Date().toString().slice(15, 24);
        }
        setInterval(clock, 1000);
    </script>
</body>
</html>
```

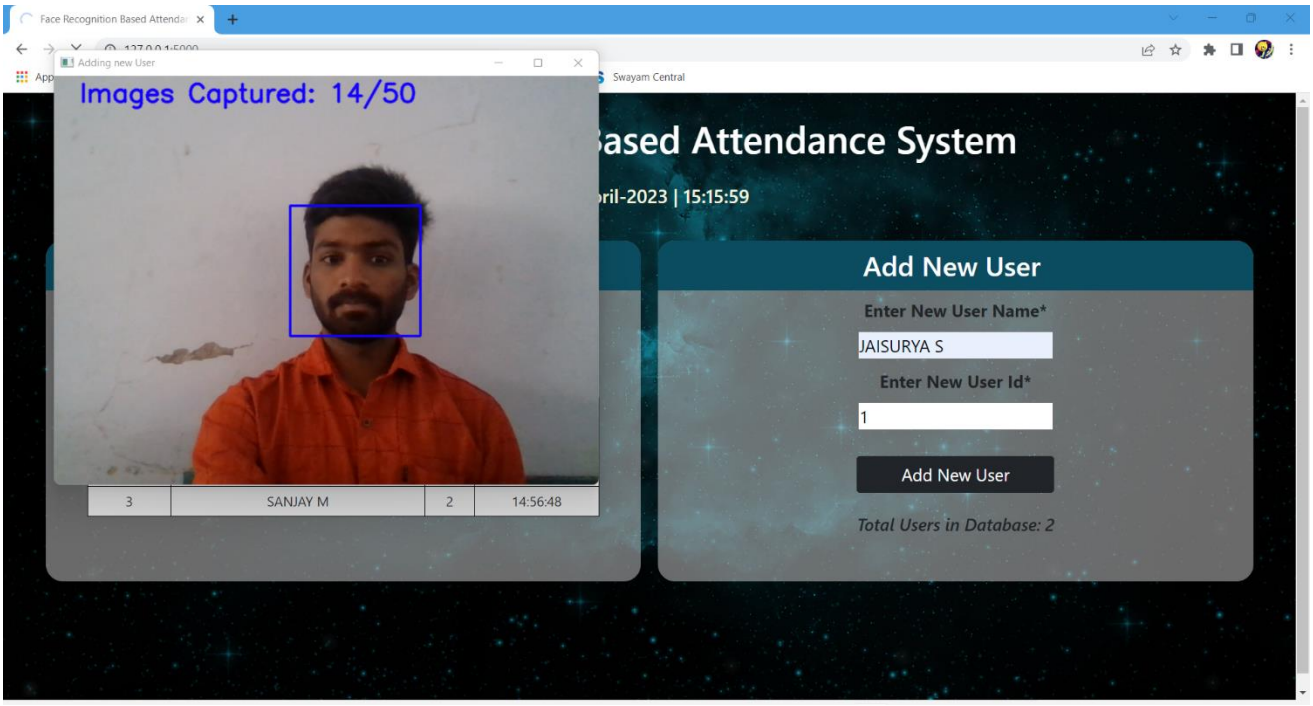
APPENDIX 2

SCREEN SHOTS

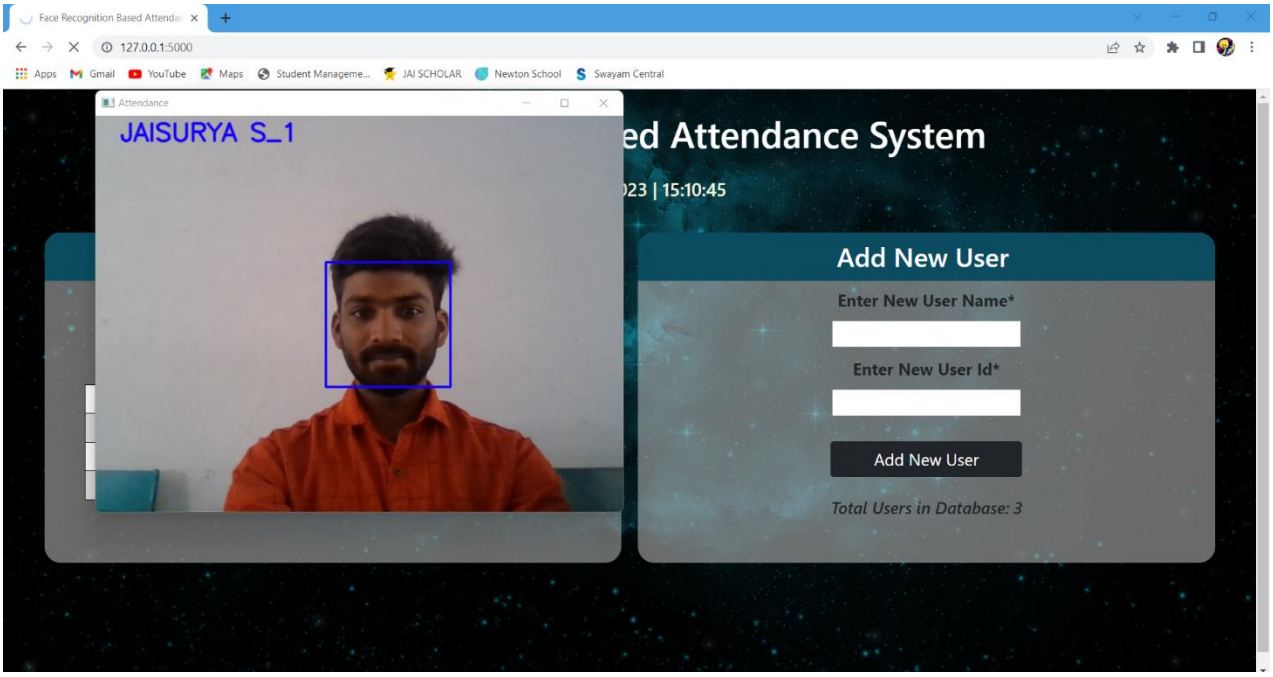
A.2.1 USER INTERFACE



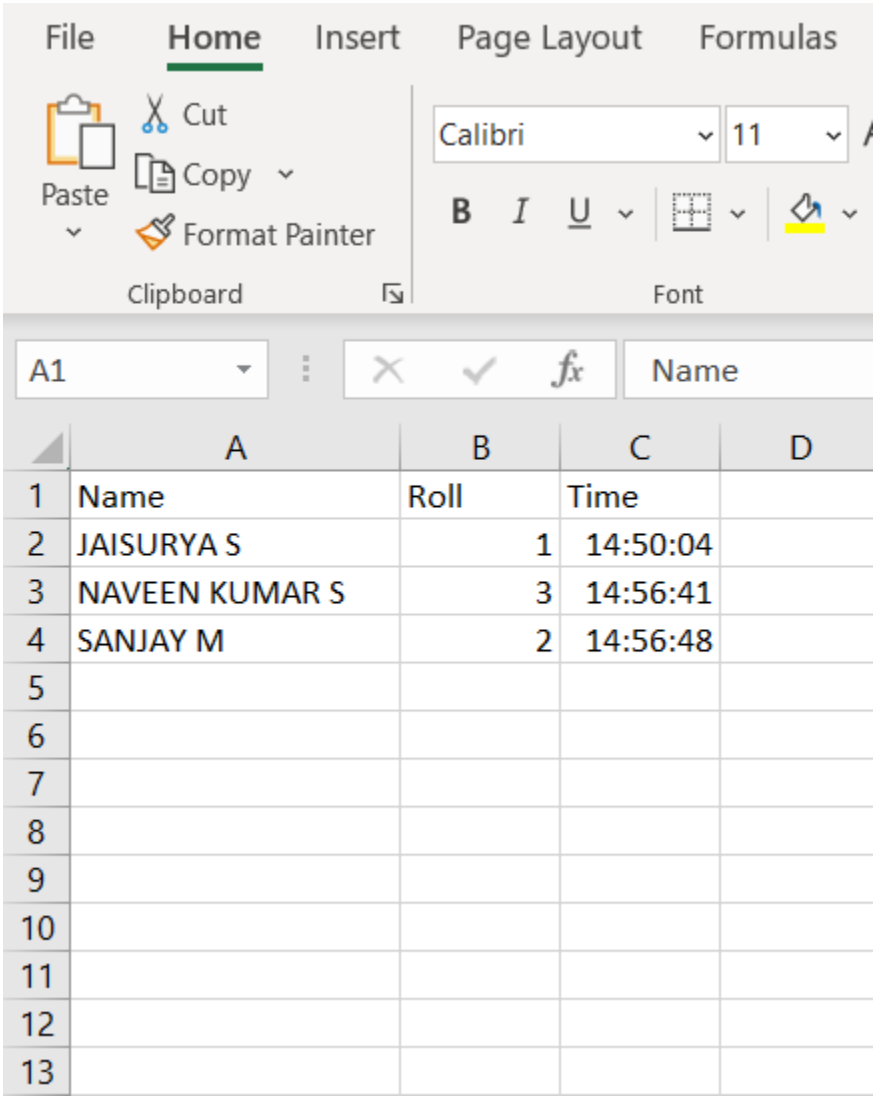
A.2.2 TRAIN MODEL



A.2.3 FACE RECOGNITION



A.2.4 DATABASE



	A	B	C	D
1	Name	Roll	Time	
2	JAISURYA S	1	14:50:04	
3	NAVEEN KUMAR S	3	14:56:41	
4	SANJAY M	2	14:56:48	
5				
6				
7				
8				
9				
10				
11				
12				
13				

REFRENCES

[1] Kaneez Laila Bhatti1, Laraib Mughal1, Faheem Yar Khuhawar1,Sheeraz Ahmed Memon1,"Smart Attendance Management System Using FaceRecognition "

https://eudl.eu/doi/10.4108/eai.13-7-2018.159713

[2] Nirmalya Kar, Mrinal Kanti Debbarma, Ashim Saha, and Dwijen Rudra Pal,

"Study of Implementing Automated Attendance System Using Face Recognition Technique"

https://scholar.google.com/scholar?hl=en&as_sdt=0%2C5&q=Nirmalya+Kar+&btnG=

[3] Bhautik Gondaliya, Shankar Sontakke, Kuldeep Suryawanshi, Tapsi Sonawane and Rupali Satpute," Attendance Management using Facial Recognition"

https://easychair.org/publications/preprint/Mfx2

[4] Neela Ashish Kumar, Swarnalatha P, Prudhvi Chowdary, Jai Krishna Naidu, Kailasa Sandeep Kumar,Undergraduate Students, School of Computer Science and Engineering, VIT," Smart Attendance Marking System using Facial Recognition"

https://rjstonline.com/HTML_Papers/Research%20Journal%20of%20Science%20and%20Technology__PID__2019-11-2-3.html

[5] V. Bhalla, T. Singla, A. Gahlot and V. Gupta, "Bluetooth based attendance management system", *International Journal of Innovations in Engineering and Technology (IJIET)*, vol. 3, no. 1, pp. 227-233, 2013.

https://scholar.google.com/scholar?as_q=Bluetooth+based+attendance+management+system&as_occt=title&hl=en&as_sdt=0%2C31

[6] P. Viola and M. J. Jones, "Robust real-time face detection", *International journal of computer vision*, vol. 57, no. 2, pp. 137-154, 2004.

https://digital-library.theiet.org/content/journals/10.1049/iet-spr.2008.0172

[7] Naveed Khan Baloch, M. HaroonYousaf, Waqar Ahmad and M. Iran Baig, "Algorithm for Efficient Attendance Management: Face Recognition based Approach", *IJCSI*, vol. 9, no. 4, No 1, July 2012.

https://scholar.google.com/scholar?as_q=Algorithm+for+Efficient+Attendance+Management%3A+Face+Recognition+based+Approach&as_occt=title&hl=en&as_sdt=0%2C31

[8] Nirmalya Kar, Mrinal Kanti Debbarma, Ashim Saha and Dwijen Rudra Pal, "Study of Implementing Automated Attendance System using Face Recognition Technique", *IJCCE (International Journal of Computer and Communication Engineering)*, vol. 1, no. 2, July 2012.

https://scholar.google.com/scholar?as_q=Study+of+Implementing+Automated+Attendance+System+using+Face+Recognition+Technique&as_occt=title&hl=en&as_sdt=0%2C31