CHAPTER 1: INTRODUCTION

For every organization, today attendance is the most important thing to record the presence of employee. The presence of employee in an organization is a sign that the person is carrying out their obligations to come to the agency or organization. Usually, attendance is done manually. It can be signed or called one by one. We can use face recognition to record attendance from everyone. In this face recognition, many algorithms are performed to detect face of employee. The system can recognize a person's face and record attendance from that person so that attendance activities are more efficient and faster.

A face recognition system is a perfect way to solve these problems. Using face detection, employee's faces will be taken in real time while the employee is entering in office and entry process will be smooth. As for the admin, there is no hassle to keep all the employee's attendance registers for reports later because the attendance is automatically generated by the system. The data generated for the report will be accurate and there will be no more misleading data since the chances for employee to forge the documents are eliminated

Face recognition is a part of biometric identification that extracts the facial features of a face, and then stores it as a unique face print to uniquely recognize a person. Face recognition technology has gained the attention of many researchers because of its wide application. Face recognition technology is better than other biometric based recognition techniques like finger-print, palm-print, iris because of its non-contact process. Recognition techniques using face recognition can also recognize a person from a distance, without any contact or interaction with person. The face recognition techniques are currently implemented in social media websites like Facebook, at the airports, railway stations. The, at crime investigations. Facebook uses the facial recognition technique for automating the process of tagging people. For face recognition we require large dataset and complex features to identify a person in all conditions like change of illumination, age, pose, etc. Recent researches show there is a betterment in facial recognition systems. In the last ten years there is huge development in recognition techniques.

But currently most of the facial recognition techniques is able to work fine. During the recent few years, a good improvement has been made in facial recognition systems. In comparison to the last decade, one can observe an enormous development in the world of face recognition.

1.1 Existing System and Need for System

Speaking about manual attendance systems, these are traditional systems that require employees to fill in their attendance sheets manually. These are generally used by small scale companies where there is less number of employees. However, such attendance systems require fair and consistent execution. Besides, HR managers face enormous pressure when it comes to collecting details about employees' working hours with these systems. For better understanding, let's have a look at the pros and cons of such a system.

1.2 Company Profile: -

For my project I visited a nearby company. It was started at 2019 in Sangli. In this company there are around 20 employees work. They need a face recognition based attendance system.

1.2 Operating Environment Hardware and Software

Hardware:

- Processor: Core I3
- RAM: 4GB ram
- Hard disk driver: 1TB
- HDD Keyboard: 104 keys
- Mouse: hp Mouse
- Monitor: Laptop
- Display Type: LCD

Software:

- Operating System: Windows10(Home Basic)
- Designing Tool: Pycharm
- Frontend: Streamlit (python)
- Backend: Python

1.3 Brief Description of Technology Used

steps involved in a face recognition model are Data Gathering from provided image by encoding it, encode face data cached from webcam, compare both encodings if matches then attendance will be recorded.

In this project we need libraries called **OpenCV** and **face_recognition.**

To make face recognition work, we need to have a dataset of photos also composed of a single image per character named wit id and employee name. for e.g. image with 4digit id followed by staff name like: 0003Vrushabh.jpg",

First of all, With the usual OpenCV procedure, we extract the image, in this case, 0003Vrushabh.jpg**,** and convert it into RGB color format. Then we do the "face encoding" with the functions of the Face recognition library.

Now we have to encode all the images in our database, so that through the webcam video stream if it finds the match it shows the name

With a simple Opencv function, we will take the webcam stream and loop it until user stop program

Now we identify the face passing the frame of the webcam. If it matches face with any of the encoded face it will give us the name of the person and an array with the position at each moment of the movement and record the attendance of the employee.

1.4 Database

1.5.1 Data Table

Name	Data type	Key	Description
Id	Integer	Not Null	Date of trading day.
Name	Text	Not Null	Highest value of that day.
Date	Date	Not Null	Date of Present.
Time	Time	Not Null	Time of Present.

1.5.2 Data Dictionary:

1.Id

ID is identification number of Employee which is taken by slicing first four-digits of employee image data provided by user. Datatype of date is INTEGER.

2. Name

Name is name of Employee which is taken by slicing from fifth character from name of employee image data provided by user. Datatype of date is TEXT.

3. Date

Date is date on which presence of employee recorded. Datatype of date is DATE

4.Time

Time is time on which presence of employee recorded. Datatype of date is TIME.

CHAPTER 2: PROPOSED SYSTEM

2.1 Study of System

- **steps involved in a face recognition based attendance system:**
- **Data Gathering:** Extract unique characteristics of Kirill’s face that it can use to differentiate him from another person, like eyes, mouth, nose, etc.
- **Data Comparison:** Despite variations in light or expression, it will compare those unique features to all the features of all the people you know.
- **Face Detection:** Look at the picture and find a face in it.
- **Face Recognition:** It will determine “Hey, that’s my boy Kirill!”

2.2 Feasibility Study

An important objective of conducting the System Analysis is the determination of the feasibility. Feasibility study is carried out to select the best system that meets the performance requirements.

Feasibility study is necessary to evaluate the feasibility of a project at the earliest possible time. It is used to investigate the project and examines whether the designed system will be useful to the organization. Feasibility and risk analysis are related in many ways. If project risk is great, the feasibility of producing quality software is reduced.

2.3 Objectives of Proposed System

The main objective of face recognition based attendance system is to manage the details of attendance of employees. The project is totally built to get easier attendance system and easier access to attendance record. The purpose of this site is to reduce manual work, it tracks all the details about the attendance.

Features of website:

- Easy to operate.
- Quick access to records.
- Display records of selected date.
- Less time consuming.
- Managing Image data.
- Increase the efficiency of managing Records.

2.4 Module Specification

In Face recognition based attendance system there are 4 main modules are as follows:

□ Image Upload (Gathering data):

The first module in Face recognition based attendance system is ‘image upload’. In this module user upload image data with name as first four letter as identity number and followed by his name like “0001Vrushabh”. User can also able to see what data he or she is uploaded on the website.

□ Image Encoding (Process data):

The second module in Face recognition based attendance system is ‘image encoding’. In this module, system extract image data and store for comparison with image data getting from camera while detecting and marking attendance.

□ Face Detection and Recognition:

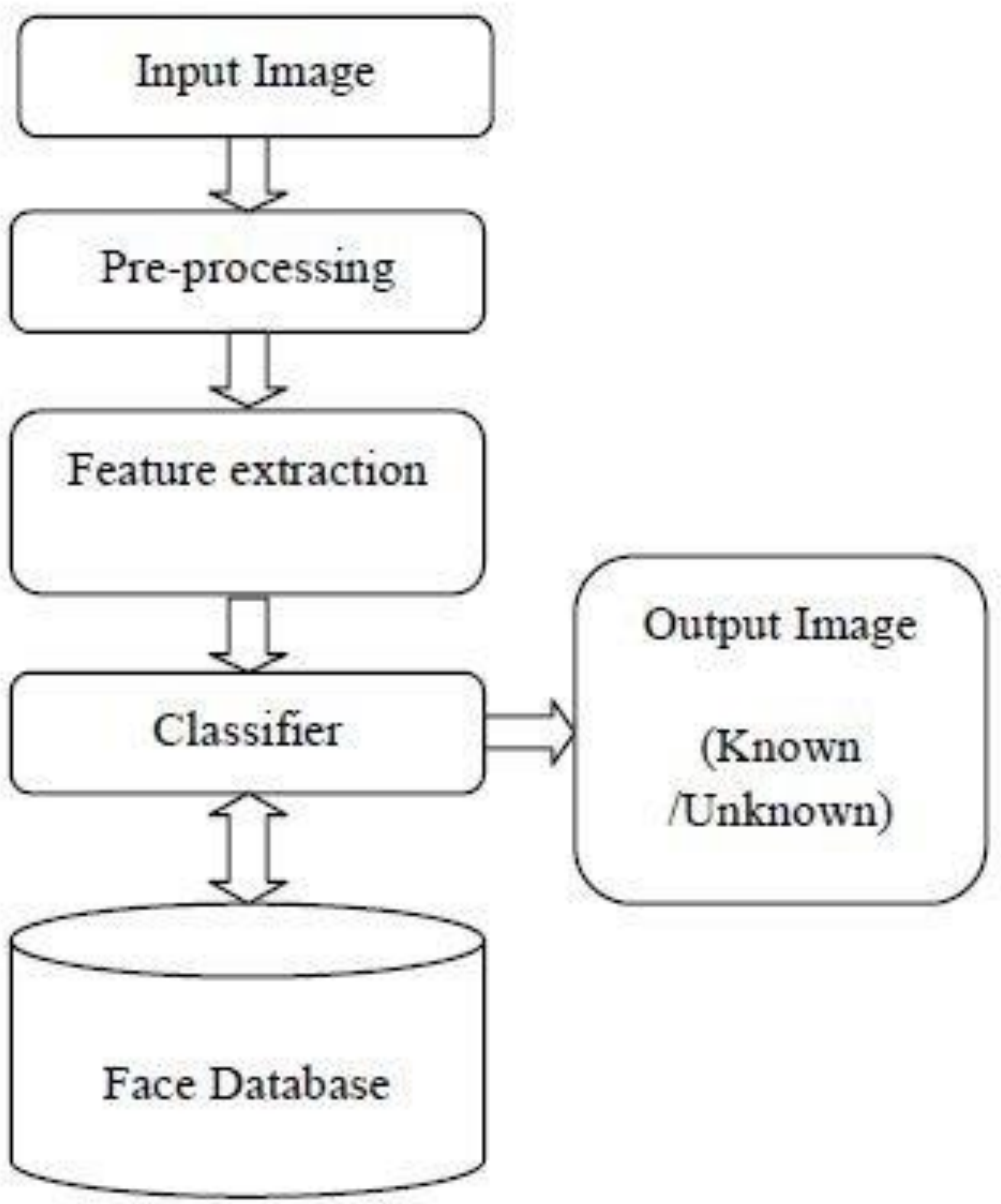
The third module in Face recognition based attendance system is ‘Face Detection and Recognition’. In this module system detect the face from camera input and extract the features rom it and compare it with available image data which is uploaded in images folder and if it matches with one of them then system will recognize the face on camera and show message and mark the attendance with current time stamp.

□ Attendance Report:

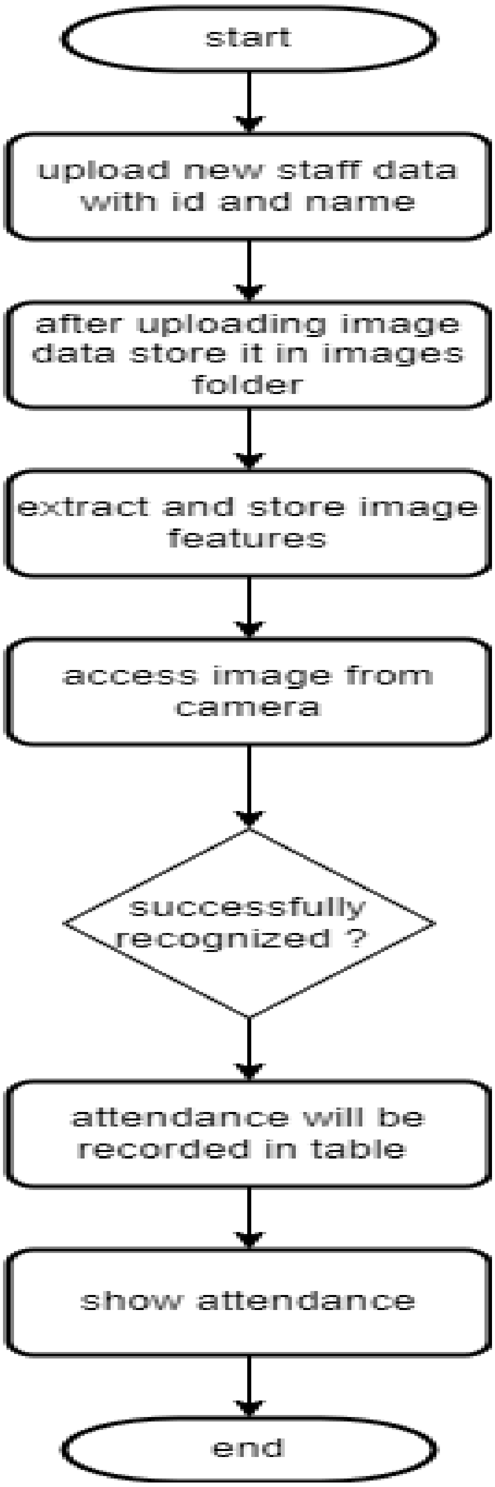
The first module in Face recognition based attendance is ‘Attendance Report’. In this module user can view attendance in tabular format of selected date in given checkbox.

CHAPTER 3: ANALYSIS & DESIGN

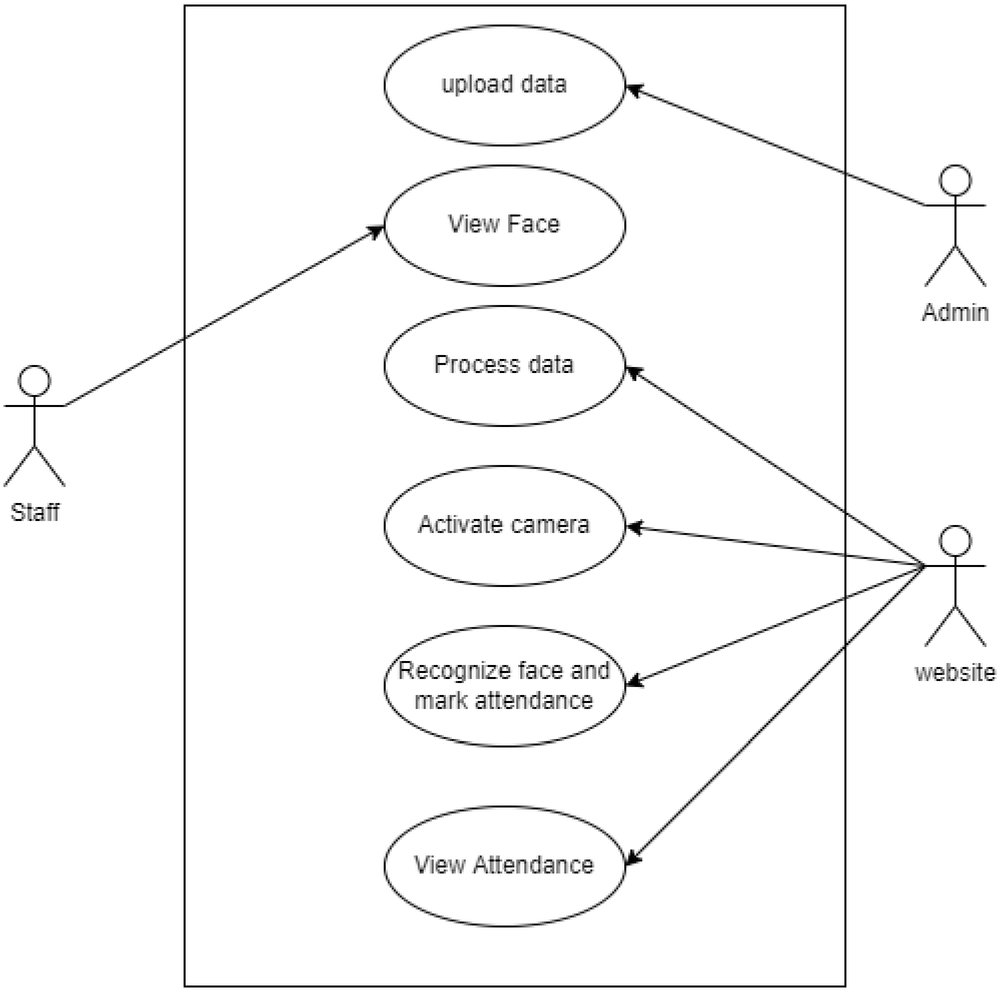
3.1 Block diagram



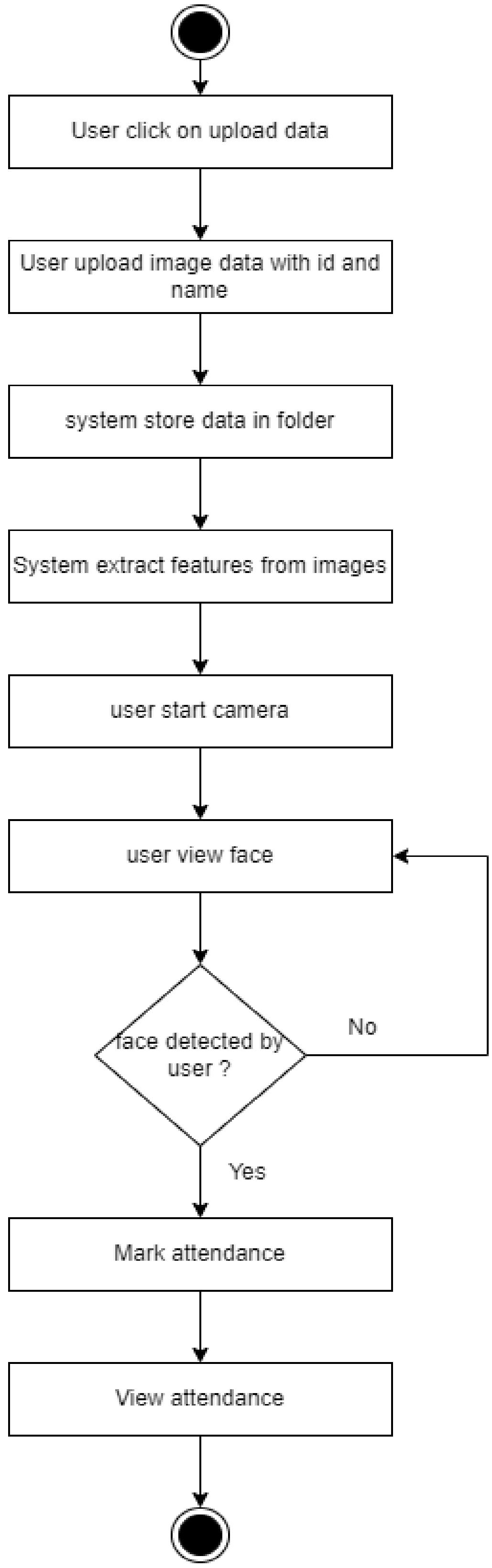
3.2 Flowchart


start
upload new staff data with id and name
after uploading image data store it in images folder
extract and store image features
access image from camera
successfully recognized ?
attendance will be recorded in table
show attendance
end

3.3 Use case Diagram



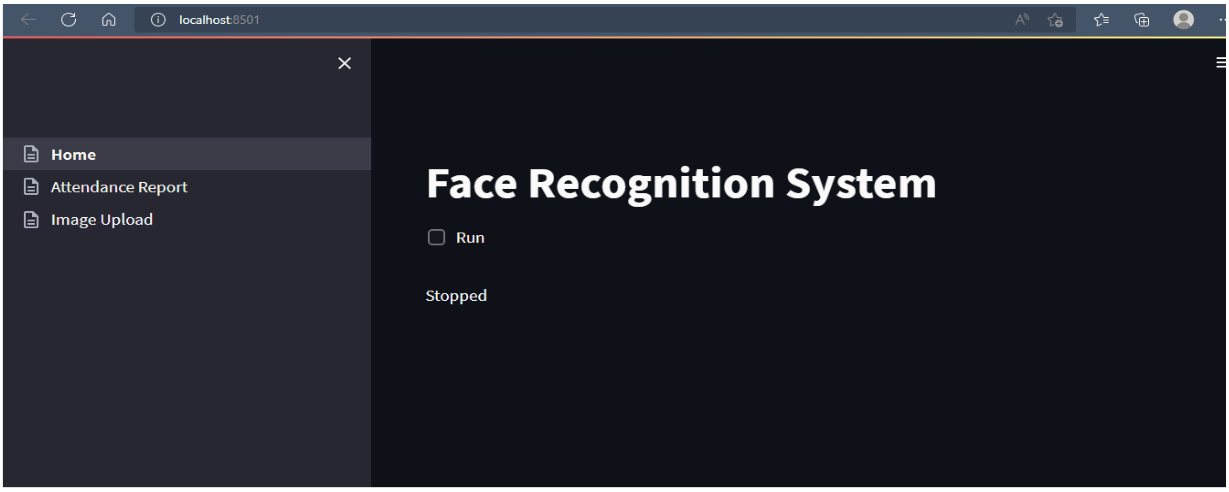
3.4 Activity Diagram


User click on upload data
User upload image data with id and name
system store data in folder
System extract features from images
user start camera
user view face
face detected by user ?
No
Yes
Mark attendance
View attendance

CHAPTER 4: USER MANNUEL

4.1 User Input /Output screens

4.1.1 Home Page

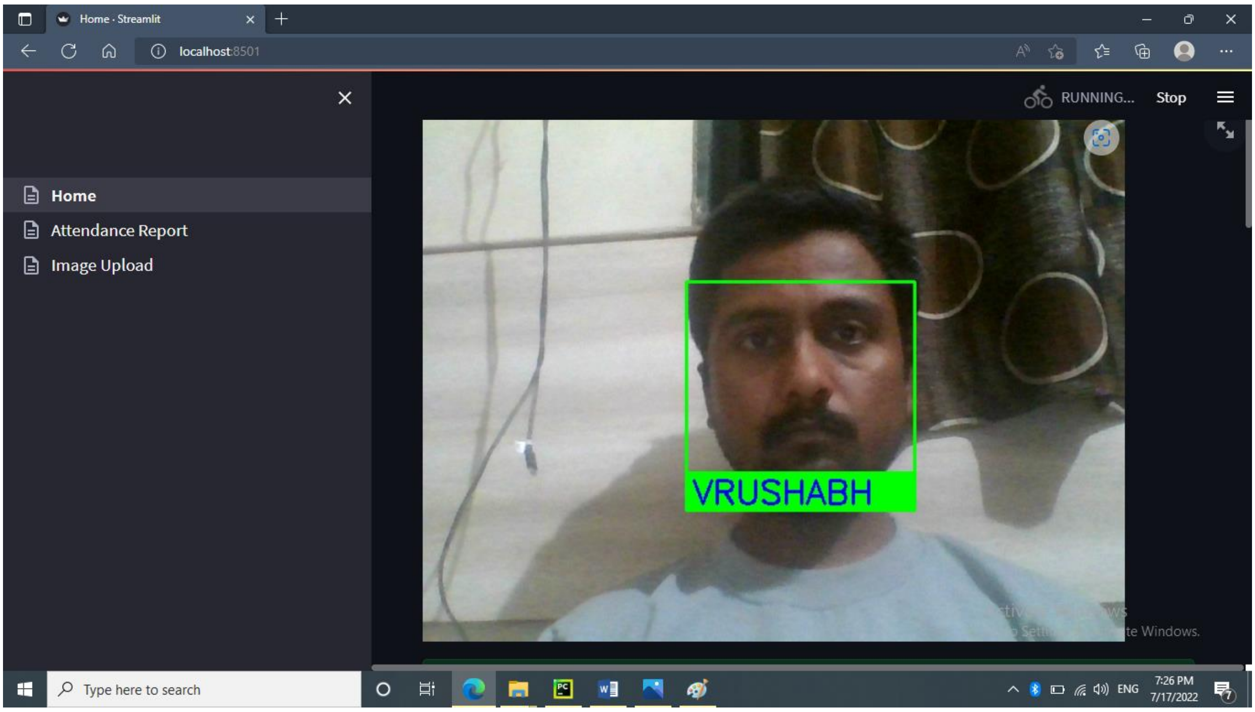


Face Recognition System

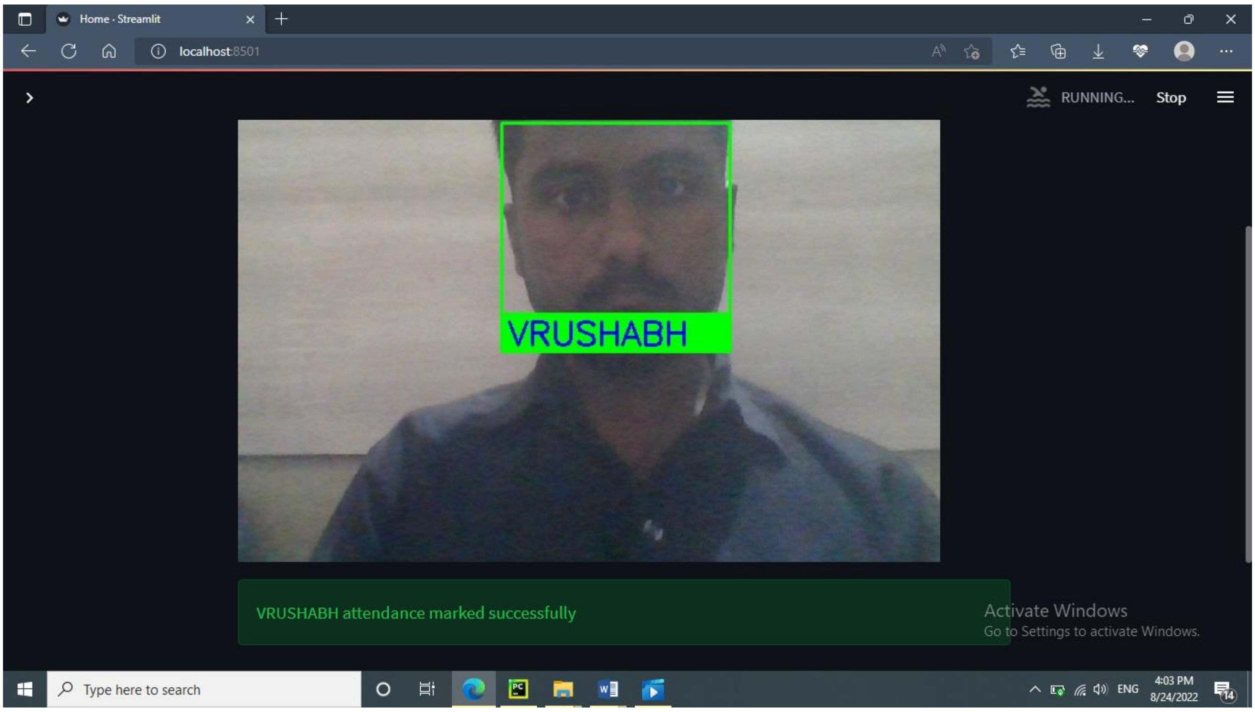
□ Run

Stopped

4.1.2 Face Detected

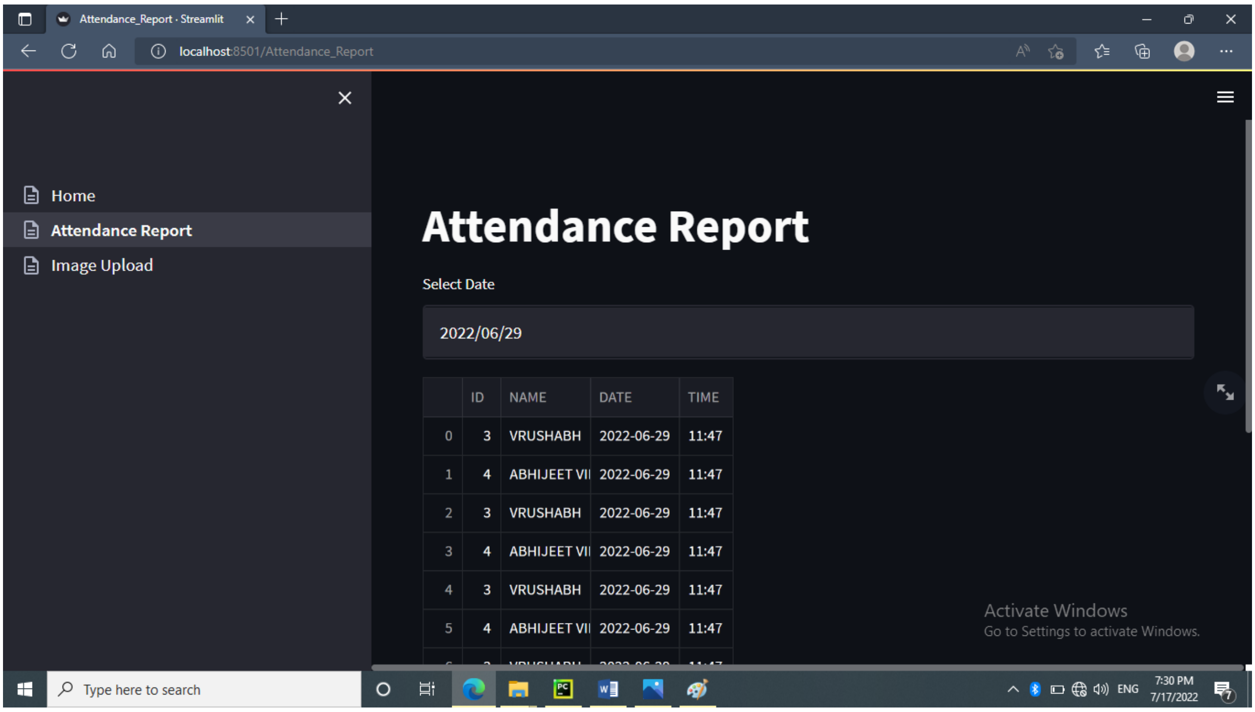


4.1.3 Attendance Mark



VRUSHABH attendance marked successfully

4.1.3 Attendance Report


Attendance_Report · Streamlit
localhost:8501/Attendance_Report
Home
Attendance Report
Image Upload
Activate Windows
Go to Settings to activate Windows.
Type here to search
ENG
7:30 PM
7/17/2022

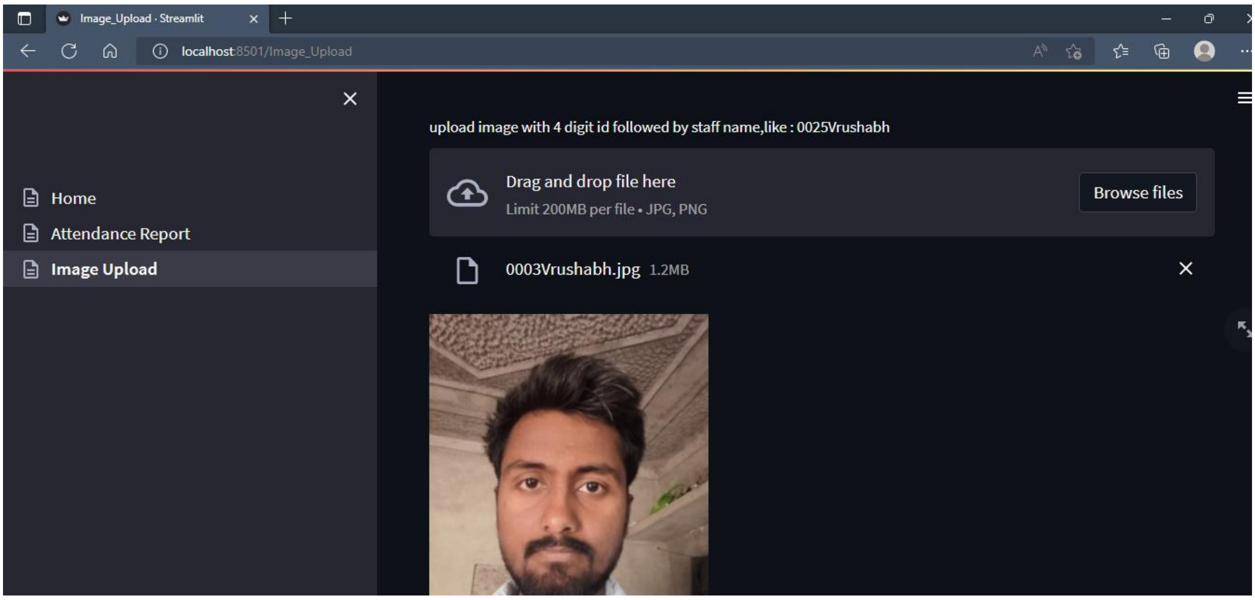
Attendance Report

Select Date

2022/06/29

	ID	NAME	DATE	TIME
0	3	VRUSHABH	2022-06-29	11:47
1	4	ABHIJEET VII	2022-06-29	11:47
2	3	VRUSHABH	2022-06-29	11:47
3	4	ABHIJEET VII	2022-06-29	11:47
4	3	VRUSHABH	2022-06-29	11:47
5	4	ABHIJEET VII	2022-06-29	11:47
6	3	VRUSHABH	2022-06-29	11:47

4.1.4 Image Upload



upload image with 4 digit id followed by staff name,like : 0025Vrushabh

Drag and drop file here

Limit 200MB per file • JPG, PNG

0003Vrushabh.jpg 1.2MB

4.1 Source Code

Home.py

```
import cv2

import numpy as np

import face_recognition

import os

import streamlit as st

from datetime import datetime

import sqlite3

import pandas as pd


now = datetime.now()


nowtime = now.strftime("%H:%M")


nowdate = now.date()


conn = sqlite3.connect('database.db', check_same_thread=False)


c = conn.cursor()

#

# # cnx = sqlite3.connect('database.db')

# df = pd.read_sql_query('SELECT * FROM attendance WHERE dt = ?', nowdate)

#
```

```
# id_list = list(df['id'])
#
# print(id_list)


#
# c.execute('SELECT id FROM attendance WHERE dt = ?', (nowdate,))
#
# id_list = (c.fetchall())
#
# print(id_list)
# print(type(id_list))
#
# outputlist = [item for t in id_list for item in t]
#
#
# print(outputlist)
# print(type(outputlist))
#


def function_attendance(i, x, y, z):
    c.execute('CREATE TABLE IF NOT EXISTS attendance(id INTEGER,name TEXT,dt
DATE,tm TIME);')
    c.execute('INSERT INTO attendance VALUES(?,?,?,?)', (i, x, y, z))
    conn.commit()
    st.success(separated_name + " attendance marked successfully ")
st.title("Face Recognition System")

run = st.checkbox('Run')

FRAME_WINDOW = st.image([])

path = 'images'

images = []

personName = []

myList = os.listdir(path)

# print(myList)


for cu_img in myList:

    current_img = cv2.imread(f'{path}/{cu_img}')

    images.append(current_img)

    personName.append(os.path.splitext(cu_img)[0])

# print(personName)



def faceEncodings(images):

    encodelist = []

    for img in images:

        img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)

        encode = face_recognition.face_encodings(img)[0]

        encodelist.append(encode)

    return encodelist
```

```
encodeListKnown = faceEncodings(images)
print("All Encodings Completed!!!")


camera = cv2.VideoCapture(0)


while run:
    ret, frame = camera.read()
    frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
    faces = cv2.resize(frame, (0, 0), None, 0.25, 0.25)
    faces = cv2.cvtColor(faces, cv2.COLOR_BGR2RGB)

    facesCurrentFrame = face_recognition.face_locations(faces)
    encodeCurrentFrame = face_recognition.face_encodings(faces, facesCurrentFrame)

    for encodeFace, faceLoc in zip(encodeCurrentFrame, facesCurrentFrame):
        matches = face_recognition.compare_faces(encodeListKnown, encodeFace)
        faceDis = face_recognition.face_distance(encodeListKnown, encodeFace)

        matchIndex = np.argmin(faceDis)

        if matches[matchIndex]:
            name = personName[matchIndex].upper()
            y1, x2, y2, x1 = faceLoc
            y1, x2, y2, x1 = y1*4, x2*4, y2*4, x1*4
```

```
cv2.rectangle(frame, (x1, y1), (x2, y2), (0, 255, 0), 2)
cv2.rectangle(frame, (x1, y2-35), (x2, y2), (0, 255, 0), cv2.FILLED)

# separate name and id

s_id = name[0:4]

separated_id = int(s_id)

separated_name = name[4:]

cv2.putText(frame, separated_name, (x1 + 6, y2 - 6), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 0, 255), 2)

c.execute('SELECT id FROM attendance WHERE dt = ?', (nowdate,))

id_list = (c.fetchall())

# print(id_list)
# print(type(id_list))

outputlist = [item for t in id_list for item in t]

# print(outputlist)
# print(type(outputlist))
```

```
            if separated_id not in outputlist:
                function_attendance(separated_id, separated_name, nowdate, nowtime)


    FRAME_WINDOW.image(frame)


else:
    st.write('Stopped')
```

Attendance Report.py

```
import streamlit as st

import sqlite3

import pandas as pd

import datetime


conn = sqlite3.connect('database.db', check_same_thread=False)


c = conn.cursor()


st.title("Attendance Report")


enter_date = st.date_input("Select Date", datetime.date.today())


#

# def function_delete_duplicate():

#       # c.execute('DELETE FROM attendance WHERE id NOT IN (SELECT MAX(id) id FROM attendance GROUP BY name, dt, tm)')

#     # c.execute('SELECT DISTINCT id,name,dt,tm FROM attendance;')

#     c.execute('DELETE FROM attendance WHERE id = 5')

#        # DELETE FROM DETAILS WHERE SN NOT IN(SELECT MAX(SN) FROM DETAILS GROUP BY EMPNAME, DEPT, CONTACTNO, CITY)

#     conn.commit()
```

```
def function_view_data(enter_date):
    c.execute('SELECT * FROM attendance WHERE dt = ?', (enter_date,))
    data = c.fetchall()
    # st.write(data)
    df = pd.DataFrame(data, columns=['ID', 'NAME', 'DATE', 'TIME'])
    st.dataframe(df)


if enter_date is not None:
    function_delete_duplicate()
    function_view_data(enter_date)
```

Image Upload.py

```
import streamlit as st


import os


from PIL import Image


@st.cache
def load_image(img_input):
    img = Image.open(img_input)
    return img



img_input = st.file_uploader("upload image with 4 digit id followed by staff name,like : 0025Vrushabh", type=["jpg", "png"])


if img_input is not None:
    file_details = {"file_name": img_input.name, "file_type": img_input.type}
    img = load_image(img_input)

    st.image(load_image(img_input), width=250)

    with open(os.path.join("images", img_input.name), "wb") as f:
        f.write(img_input.getbuffer())
```

```
if st.button("upload"):
    # function_input(id_input)


    st.success("Record uploaded successfully ")
```

Conclusion:

Building a simple automated attendance system using facial recognition in Python that detects faces and records live attendance with the time in an SQLite table is really helpful for an organization.

An attendance system that records live attendance with the time in an SQLite table using a webcam. The main package used here is the face_recognition to locate the facial features of an images.

The face_distance() is used to calculate the distance between the facial features of live image and provided image. Lowest distance will be the best match. The face_locations() is used to detect the x and y coordinates of the rectangle that surrounds the face. The name corresponding to the encoded face is stored and recorded along with the current time in table.

Limitations

1. This application can’t differ between photo and actual person.

2. There is no any alternative method provided when it fails.

Bibliography

1. www.stackoverflow.com

2. www.w3schools.com

3. https://docs.python.org

4. https://docs.streamlit.io

5.https://www.youtube.com