Smart Internz Externships Project Report

Title: IOT Based Automated Attendance System Based On Face Recognition

Introduction

Overview:

IoT(internet of things) is the process refers to scenarios where network connectivity and computing capability extends to objects, sensors and everyday items not normally considered computing device with unique identifiers object, animal and people. There is no single, universal definition. IoT implementations use different technical communications models, each with its own characteristics are device to cloud, device to gateway, back end data sharing. These models highlight the flexibility in the ways that iot devices can connect and provide value to the user.

Face recognition in video streaming is used to take attendance on the smart device. A camera is included into the device. The image is captured whenever the camera recognises a face. To recognise the person, the collected image will be fed into a facial recognition model. When a match is discovered, that person's attendance is recorded. Every hour or class, attendance is taken on a time basis. If a student is absent, the parents and the class Incharge will be notified. The administration/principal will have access to the attendance data via a web app.

Face recognition is a computer application capable of identifying or verifying a person from digital image or a video frame from video source by comparing selected facial features from the image and a face database and biometric method of identifying an individual by comparing live capture or digital image data with stored record for that person. Face recognition systems based on faceprints can quickly accurately identify target individuals when the conditions are favorable.

Purpose:

Every organization requires an attendance system to maintain record of presence of students. They have their own method to do the same. Some do manually and some use automated attendance system. Manual method includes pen and papers which consumes lot of time and wastage of resources. Also, it has risk of proxies and human error. In this project we used Face Recognition.

LITERATURE REVIEW:

1.Face and Bio-Metric Based Attendance and Security System using RFID and Arduino

In this paper they had proposed an employee attendance & security system using face and Biometric integrated with Smart RFID cards using Arduino. The project was about implementation of RFID based attendance system in integration of face recognition of the particular employee with his unique employee number. With

integration of finger print authentication (Bio-metric) into the system, security has been enhanced. Audio welcome message on the valid Employee attendance registration had introduced & for unauthorized entry Audio alert had introduced along with Sound Alarm.

2.Attendance System based on face recognition

This paper had focused to improve attendance system in school and colleges. Because there were many disadvantages in taking manual attendance like cost, fake attendance and it may be not accurate always. So biometric and face recognition techniques come into picture. Traditional face recognition methods were not more accurate. This paper had represented the idea that to save the time as well as enhance the security Attendance System using face recognition technique was very useful. In this idea images were capturing through camera or CCTV camera in school/college for attendance purpose. After that it first detected the face using features of eyes, nose, mouth, hairs and also different pose of face in image. There were different methods for face detection like Ada-Boast, SQMT, LBP and SNOW classifier method.

3. Abhishek Jha, "Class Room Attendance System Using Facial Recognition System",IEEE The International Journal of Mathematics, Science, Technology and Management (ISSN : 2319-8125) Vol. 2 Issue 3,2015-proposed the face is the identity of a person. The methods to exploit this physical feature have seen a great chance of image processing techniques. The accurate recognition of a person is the aim of a face recognition system and this identification maybe used for coming processing. The methods can be facial recognition are: International Conference on Audio and (AVBPA) and(AFGR). The facial recognition process can be divided into two stages: processing before detection where face detection and alignment and recognition occur through feature extraction are face detection, face alignment, feature extraction, face matching so on its providing an automated attendance system for all the students that attend a certain lecture, section, laboratory or exam at its specific time, thus saving time, effort and reduce distractions and disturbances.

4. Divyaharitha p, Gayathri B, Safiya Parvin A "Automated Of Attendance and Student Tracking With Face Recognition And Ultrasonic Sensor" IEEE transaction 2013- proposed the Gabor wavelet responses at single locations of facial images are collected into Gabor jets, which are extracted at several offset positions and assembled into a Gabor graph G. Often, the identity of a probe image, it is compared with several gallery images and assigned the identity of the most similar gallery image. Image comparison is traced back to the comparison of the two Gabor graphs extracted from these images. Elastic bunch graph matching (EBGM), the correspondence problem is solved locally by computing offset position corrections, so-called

disparities. The CAS-PEAL images are partitioned into a gallery of 1040 images with ambient illumination and neutral facial expression, and different probe sets. We here process the probe sets Expression and Lighting. The Expression probe set contains 1570 images with ambient illumination, showing one of five facial expressions. The Lighting probe set consists of images with neutral expression, but strong fluorescent illumination from fifteen different directions and one frontal incandescent illumination.

5. Michihiko Minon, Weijane Cin, Tetsuo Shoji "Face Recognition Based Lecture Attendance System" IEEE Transaction June 2013.- proposed if the attendance of a student of classroom lecture is attached to the video streaming service, It is important to take the attendance of the students in the classroom automatically. The attendance value can be define at particular value at the shooting plan of the sequence number of the data fusion of the lecture attendance,seat by using the background subtraction and inter-frame subtraction of the image from the sensing camera on the ceiling. They have few methods are shooting plan, architecture, existence value our system selects one seat from the estimated sitting area obtained by ASD, directs the camera to the seat and captures images. the face images are detected from the captured image, archived and recognized face detection data and face recognition data are recorded into the database.so the many product can be define at particular value of an student attendance marking.

6. Naresh Bbu N.T, Vaidehi V,Vasuhi S " Person Authenciation Using Multiple Sensor Data Fusion" ICTACT Journal On Soft Computing, April 2011, Volume: 01, Issue: 04- Naresh Babu N.T, Vaidehi V, Vasuhi s [10] proposed the Human Verification rapidly growing research area due to increasing demands for security in commercial and law enforcement applications. Person authentication involves verification of a person's identity based on his/her physiological or behavioral characteristics. Most image-processing techniques involve the image.although there exist several holistic and feature based face recognition algorithms such as Principal Component Analysis (PCA), Fisher Linear Discriminant analysis, Image Principal Component Analysis (IMPCA), Independent Component Analysis(ICA), Orthogonal Locality Preserving Projections (OLPP) and various other methods, the theoretical concept of face recognition is not satisfied by the existing systems.

Block Diagram:

Theoretical Analysis

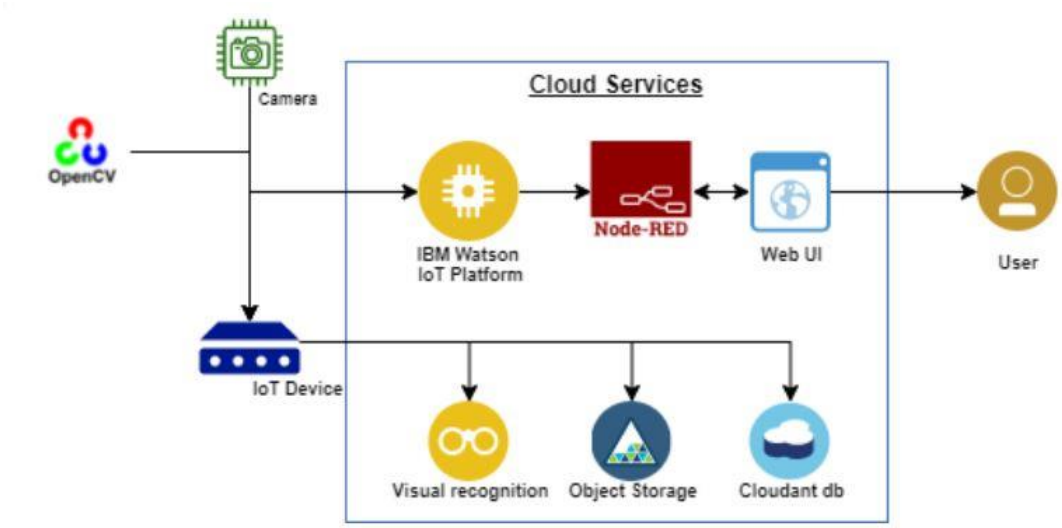


Fig 1: Overall Block Diagram of the system

Software Designing:

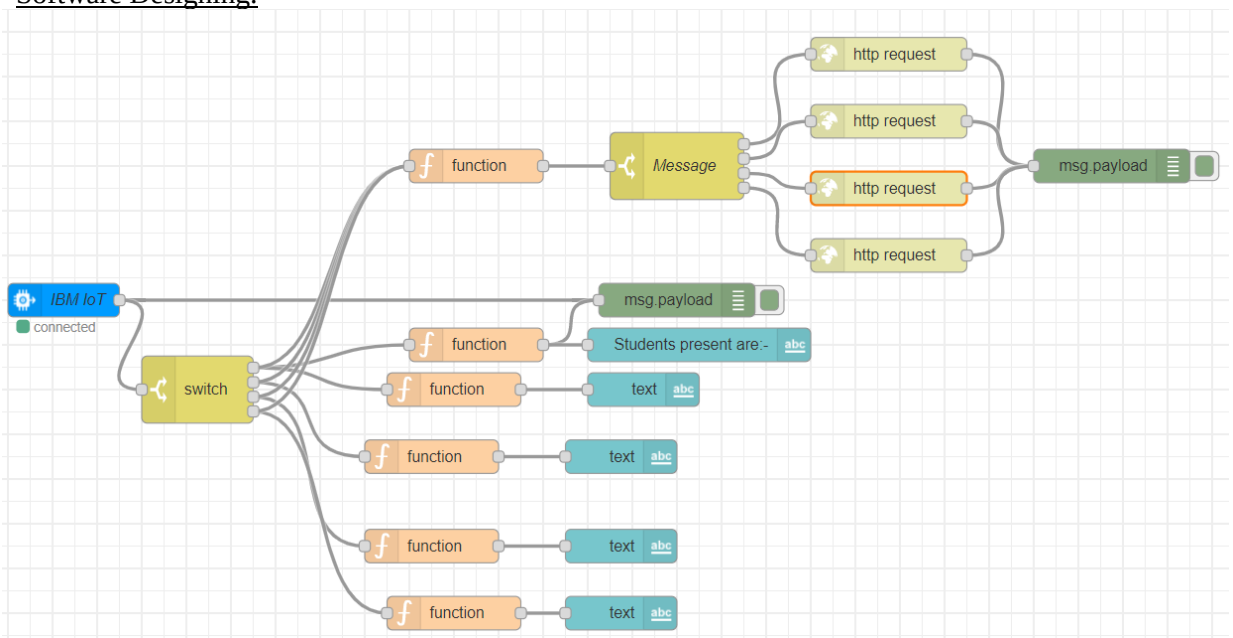


Fig 2: NODE-RED flow

We use the function node to send the message.

The switch is used to detect the person and send the information to the node which determines the status of the person i.e absent or present.

Existing problem: Maintaining attendance register daily is a difficult and time consuming task. For this we are using pen and paper which also leads to the wastage of resources . There's a chance of proxies and human mistake.

Proposed solution: To resolve this issue, This is the website where you can find a solution to your problem. Here we introduce Face recognition to take attendance. Face recognition system is to identify a person using his face image. Face recognition module that recognizes the individual student's face and update the student attendance database automatically.

Tools used:

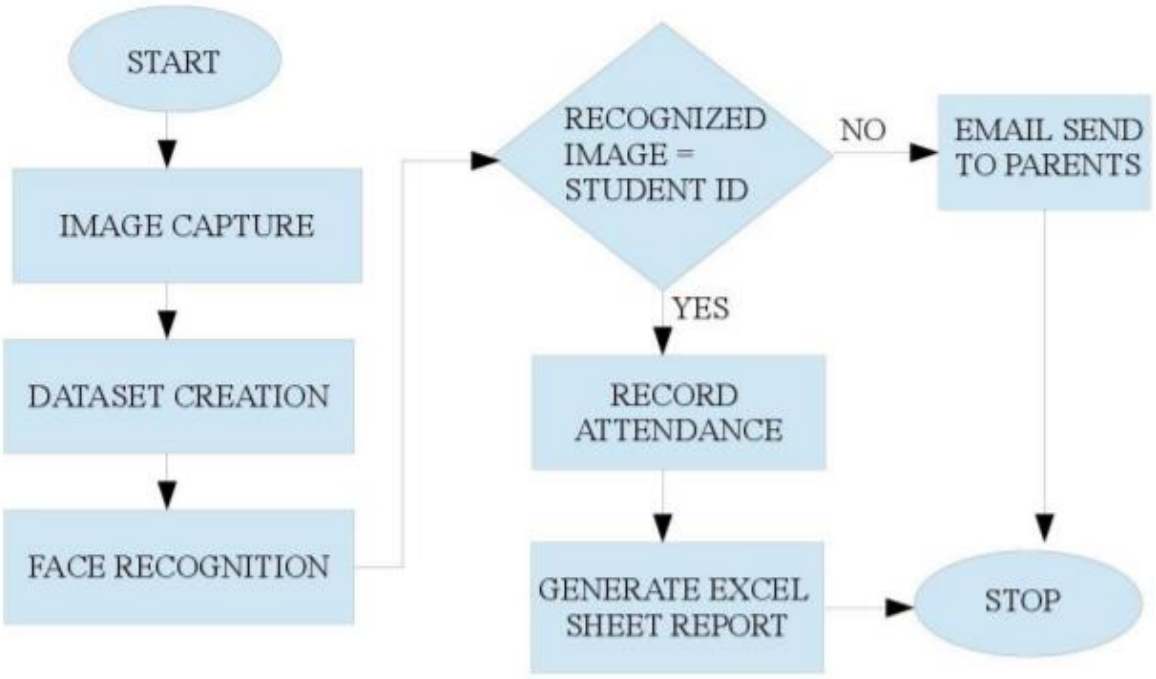
Python
IBM Watson IoT platform
Node Red to fetch the data and design the web UI
Fast2SMS service to notify the concerned authority about the status.

PROCEDURE:

- Attendance is taken by the smart device using face recognition in video streaming.
- The device is equipped with a camera. Whenever the camera detects a face it captures the image.
- The captured image will be given to the face recognition model to recognize the person.
- When a match is found the attendance of that particular person is stored.
- The attendance is taken on a time basis for every hour or class.
- If the student was absent an intimation will be sent to the parents and class Incharge.
- The admin/principal can view the attendance details through the Web UI.

Experimental Investigation

FLOW CHART:

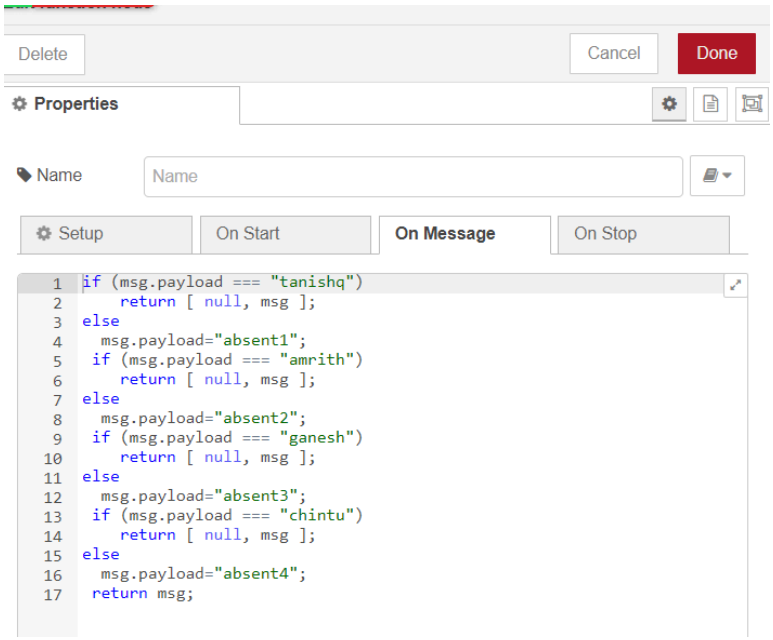


RESULTS:

Resultant python output:

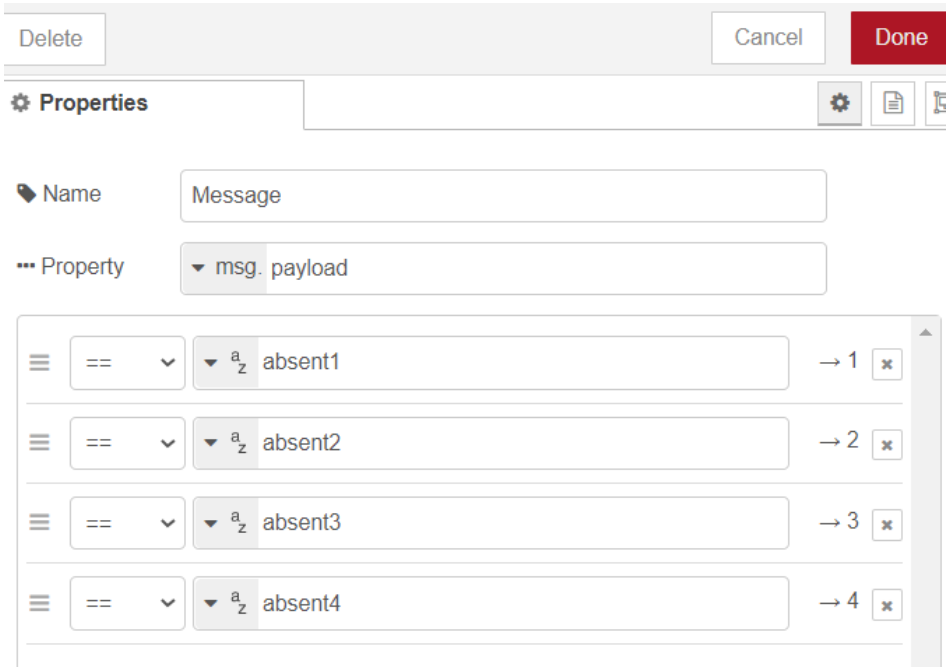
```
========= RESTART: C:\Users\DELL\Desktop\Attendance\FinalAttendance.py =========
2021-08-02 13:36:28,448   wiotp.sdk.device.client.DeviceClient   INFO    Connected successfully: d:6teby0:attendance:98765
[]
Published data Successfully: %s []
['chintu']
Published data Successfully: %s ['chintu']
['chintu']
Published data Successfully: %s ['chintu']
['chintu', 'tanishq']
Published data Successfully: %s ['chintu', 'tanishq']
['chintu', 'tanishq']
Published data Successfully: %s ['chintu', 'tanishq']
['chintu', 'tanishq']
Published data Successfully: %s ['chintu', 'tanishq']
>>>
```


The required data is successfully published in the IBM IoT platform.

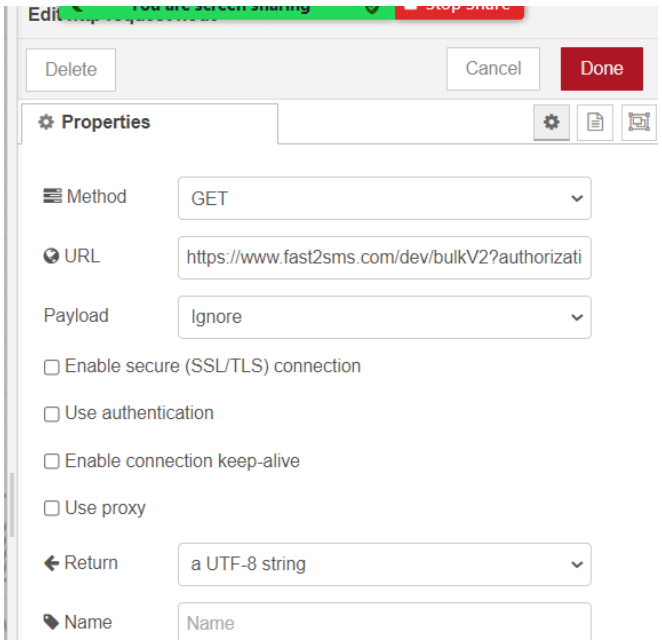


Here, if the node detects Tanishq is absent, it sets the payload as absent1 as shown in the above code. Similarly, if Amrith is not detected, it sends the payload absent2 and so on.

MESSAGE NODE:

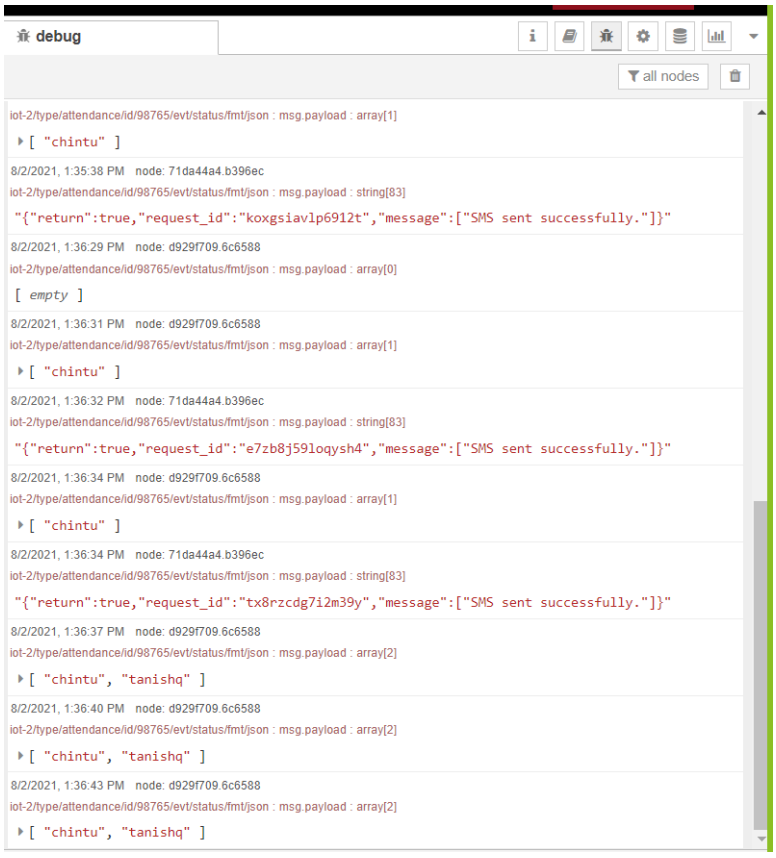


It goes to the first path if it detects the payload as absent1



Where the link for a SMS service is attached. It sends a message to that particular person's parents and the authorities

DEBUG NODE SENDING INFO:



ADVANTAGES AND DIS-ADVANTAGES:

1. Breach of privacy

Using Face Recognition system enables the government authorities to sabotage the privacy rights of other people, especially in countries where privacy rights are not of great importance. Authorities can use the technology as they want and use it to spy on others. They can collect the data without people knowing about it and use it as they wish without the consent of the people involved. Similarly, the police department can also use this technology to track criminals. But at the same time, they can use the same technology to track anyone at any place and at any time. They can detect the person from CCTV footage, smartphones, videos, social media posts, and other activities online. Indeed, it can be difficult to say if any data privacy can be maintained in the wake of the facial recognition system.

2. Biased performance

There is the possibility that facial recognition systems might not be able to identify women or people of color. Although many think it is just a misconception, some reports have emerged complaining about the same thing happening. This might occur as the problem after the first implementations, where the dataset is not very large and contains information with limited characteristics.

3. No so reliable

As per a study conducted by the Massachusetts Institute of Technology (MIT), facial recognition systems have done a lot of misidentifications. Some factors could lead the system in the wrong way, such as poor light, wrong camera angle, bad image or video quality, and more.

APPLICATIONS:

1. To verify identities in Government organizations.
2. Enterprises.
3. Attendance in Schools and colleges.
4. To detect fake entries at international borders.
5. Industries.

CONCLUSION:

This project introduces the efficient method of attendance management system in the classroom environment that can replace the old manual methods. This method is secure enough, reliable, accurate and efficient. There is no need for specialized hardware for installing the system in the classroom. It can be constructed using a camera and computer.

Future Scope

Almost all academic institutions require attendance record of students and maintaining attendance manually can be hectic as well as time consuming task. Hence maintaining attendance automatically with the help of face recognition will be very helpful and less prone to errors as compared to manual process. This will also reduce manipulation of attendance record done by students and it will save time as well. The future scope of the proposed work can be, capturing multiple detailed images of the students and using any cloud technology to store these images. The system can be configured and used in Atm machines to detect frauds.Also, the system can be used at the time of elections where the voter can be identified by recognizing the face.

Bibliography

[1] Mamtha G, Rajeshwari M, Srivardhini P "Automated Attendance Marking" International Journal For Research in Applied Science and Engineering

[2] Hteik Htar Lwin, Aung Soe Khaing, Hla Myo Tun "Automatic Door Access System Using Face Recognition" International Journal Of Scientific & Technology Research Volume 4, Issue 06, June 2015

[3] Eu Jun Chin, Wei Jen Chen and Florence Choong "Automatic Attendance Capture and Tracking System" Journal Of Engineering Science and Technology EURECA 2014 Special Issue January (2015) 45 – 59.

[4] Divyaharitha p, Gayathri B, Safiya Parvin A "Automated Of Attendance and Student Tracking With Face Recognition And Ultrasonic Sensor" IEEE transaction 2013.

[5] Mahesh Sutar, Mahesh Patil , SachinWaghmare, "Smart Attendance System Using RFID In IOT", International Journal of Advanced Research in Computer Engineering & Technology(IJARCET)

[6] Park, Jh., Salim, M.M., Jo, J.H. et al. CIoT-Net: a scalable cognitive IoT based smart city network architecture. Hum. Cent. Comput. Inf. Sci. 9, 29 (2019).

[7]Beymer, D. and Poggio, T. (1995) Face Recognition From One Example View, A.I. Memo No. 1536, C.B.C.L. Paper No. 121. MIT

[8]Menezes, P., Barreto, J.C. and Dias, J. Face tracking based on Haar-like features and eigenfaces. 5th IFAC Symposium on Intelligent Autonomous Vehicles, Lisbon, Portugal, July 5-7, 2004.

[9].Rohit Chavan, Baburao Phad, Sankalp Sawant, Vinayak Futak, Asha Rawat, "Attendance Management Using Face Recognition" , IJIRST, Vol: 1, Issue: 11, April 2015

[10] Ali Akbar, Chowdhary Obaid, Choudhary Yasir, "Automated Attendance Management System Using Face Recognition" IJARCET, Vol: 6, Issue: 8, August 2017

Team Members

Member 1:

- **Name:** CHITTARANJAN TADIKONDA
- **Registration No:** 18BIS0129
- Mail ID: crchintu07@gmail.com
- GitHub Project Link: https://github.com/smartinternz02/SI-GuidedProject-4744-1627031121

Member 2:

- **Name:** Thota sai ganesh
- **Registration No:** 18BEC0046
- Mail ID: thontasaiganesh117@gmail.com
- GitHub Project Link: https://github.com/Saiganesh7

Member 3:

- **Name:** Amrith Sriram
- **Registration No:** 18BEC0896
- Mail ID: amrith.sriram2018@vitstudent.ac.in
- GitHub Project Link: https://github.com/smartinternz02/SI-GuidedProject-4743-1627031118
-

Member 4

- **Name:** TANISHQ ANUP MEHTA
- **Registration No: 1**9BBS0178
- Mail ID: tanishqanup.mehta2019@vitstudent.ac.in
- GitHub Project Link:

Appendix

Source code:

Python Codes:

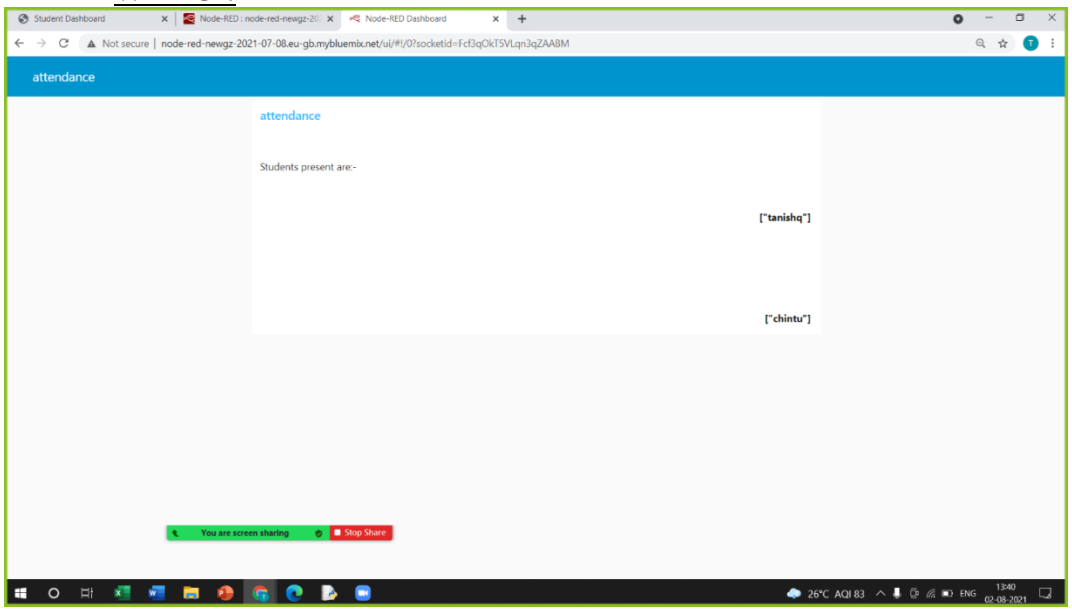
```
import face_recognition
import cv2
import datetime
import wiotp.sdk.device
import time
import random
myConfig = {
   "identity": {
      "orgId": "6teby0",
      "typeId": "attendance",
      "deviceId":"98765"
   },
   "auth": {
      "token": "W-027b0eRPv+Hg1@@f"
   }
}
def myCommandCallback(cmd):
   print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
   print()
# Get a reference to webcam #0 (the default one)
video_capture = cv2.VideoCapture(0)

# Load a sample picture and learn how to recognize it.
chintu_image =
face_recognition.load_image_file(r"C:\Users\DELL\Desktop\Attendance\chintu1.jpeg")
chintu_face_encoding = face_recognition.face_encodings(chintu_image)[0]

# Load a second sample picture and learn how to recognize it.
amrith_image =
face_recognition.load_image_file(r"C:\Users\DELL\Desktop\Attendance\amrith1.jpeg")
amrith_face_encoding = face_recognition.face_encodings(amrith_image)[0]

ganesh_image =
face_recognition.load_image_file(r"C:\Users\DELL\Desktop\Attendance\ganesh1.jpeg")
ganesh_face_encoding = face_recognition.face_encodings(ganesh_image)[0]

tanishq_image =
face_recognition.load_image_file(r"C:\Users\DELL\Desktop\Attendance\tanishq1.jpg")
tanishq_face_encoding = face_recognition.face_encodings(tanishq_image)[0]

'"biden_image1 = face_recognition.load_image_file("nikhil.jpg")
biden_face_encoding1 = face_recognition.face_encodings(biden_image1)[0]"'

# Create arrays of known face encodings and their names
known_face_encodings = [
   chintu_face_encoding,
   amrith_face_encoding,
   ganesh_face_encoding,
   tanishq_face_encoding
```

```
]
known_face_names = [
    "chintu",
    "amrith",
    "ganesh",
    "tanishq"
]

# Initialize some variables
face_locations = []
face_encodings = []
face_names = []
process_this_frame = True
client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()
while True:
    # Grab a single frame of video
    cnt=0
    ret, frame = video_capture.read()

    # Resize frame of video to 1/4 size for faster face recognition processing
    small_frame = cv2.resize(frame, (0, 0), fx=0.25, fy=0.25)

    # Convert the image from BGR color (which OpenCV uses) to RGB color (which face_recognition
uses)
    rgb_small_frame = small_frame[:, :, ::-1]

    # Only process every other frame of video to save time
    if process_this_frame:
        # Find all the faces and face encodings in the current frame of video
        face_locations = face_recognition.face_locations(rgb_small_frame)
        face_encodings = face_recognition.face_encodings(rgb_small_frame, face_locations)


        for face_encoding in face_encodings:
            # See if the face is a match for the known face(s)
            matches = face_recognition.compare_faces(known_face_encodings, face_encoding)
            name = "Unknown"

            # If a match was found in known_face_encodings, just use the first one.
            if True in matches:
                first_match_index = matches.index(True)
                name = known_face_names[first_match_index]
                #print(name)
            if name not in face_names:
                face_names.append(name)
        print(face_names)
        client.publishEvent(eventId="status", msgFormat="json", data=face_names, qos=0,
onPublish=None)
        print("Published data Successfully: %s", face_names)
        time.sleep(2)

    process_this_frame = not process_this_frame
```

```
    # Display the results
    for (top, right, bottom, left), name in zip(face_locations, face_names):
        # Scale back up face locations since the frame we detected in was scaled to 1/4 size
        top *= 4
        right *= 4
        bottom *= 4
        left *= 4

        # Draw a box around the face
        cv2.rectangle(frame, (left, top), (right, bottom), (0, 0, 255), 2)

        # Draw a label with a name below the face
        cv2.rectangle(frame, (left, bottom - 35), (right, bottom), (0, 0, 255), cv2.FILLED)
        font = cv2.FONT_HERSHEY_DUPLEX
        cv2.putText(frame, name, (left + 6, bottom - 6), font, 1.0, (255, 255, 255), 1)

    # Display the resulting image
    cv2.imshow('Video', frame)
    cnt+=1
    if cnt >=3:
        break
    # Hit 'q' on the keyboard to quit!
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
# Release handle to the webcam
video_capture.release()
cv2.destroyAllWindows()
```

UI output Screenshot:

WEB UI:



MESSAGE DISPLAYED ON USERS PHONE:

The below message was received by Amrith as he was not present.