SMART ATTENDANCE SYSTEM v2.0

Submitted in partial fulfillment of the degree of Bachelor of Technology
In
Computer Science and Engineering

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING & INFORMATION TECHNOLOGY
JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY

Introduction

- Human Beings can distinguish and recognize a lot of things including faces. This ability helps us in remembering other people and identify emotions and empathize.
- Computer Vision helps machines achieve human like abilities like image recognition, etc.
- It is easy to install face recognition system in a large scale setting like Hospitals, Schools, Colleges, Airport, etc but the actual implementation is a bit challenging because the proposed system has to account for all possible appearance variations caused by change in illumination, facial features, variations in pose, image resolution, viewing distance, etc.
- Our attendance system is an upgrade to the existing system which involved maintaining a register (like in most of the schools) and / or centralised server but without any digital assistance (like the webkiosk in our college).

Problem Statement

The traditional System of maintaining attendance in some schools still uses Registers and is a slow procedure as compared to our proposed system. In most of the universities too, the webkiosk system is manual, i.e. the faculty announces the names of students one by one but as compared to schools attendance is still stored in a centralized database because of a huge database of student data.

Significance / Novelty of Proposed Solution

- Our Proposed system is robust, stable and Modular. It is highly manageable. New Functionality can be added easily. Code is readable so any developer in future can contribute to it if we open sourced the system.
- Our format of storing attendance is highly manageable and can be easily understood. The proposed system will store the attendance in the form of a table with columns like Serial No, Enrollment No, TimeStamp of Attendance, Subject Name, Class Type [Lecture,Tutorial, Lab…], Class Time [9am Friday, 11am Tuesday…] and Faculty Name.

Working Environment Conditions

Our Proposed System is suitable for a certain environment with certain conditions for it to work in a suitable manner. The environment shall be similar to what described below:

- The system's camera shall be mounted at a certain height so it is accessible to most of the people. Ideally the camera should be right at the same height of the face.
- For better model accuracy and getting a well trained model the distance between the camera and face should be less than 50cm.
- The room/environment should be well lit. The camera should be placed in such a way that it is not directly facing a light source or the Sun itself.
- People should remove Spectacles, caps, any other accessory while using the camera/system for better accuracy and precision.

Field Study and Tech Stack

- HAAR CASCADE
- K-NEAREST NEIGHBOUR CLASSIFIER
- FLASK (Python)
- SCIKIT-LEARN (Python)
- OPENCV (Python)
- NUMPY (Python)

# Brief Description of Proposed Solution

- Our cam will capture the image of the student and convert it into 50 *50 dimension image, along with it will be changed to a grayscale image.
- KNN algorithm will be used by us to find the neighboring points in the image.
- Highly accurate and Fast
- Attendance folder is carefully designed for better comprehension of the system.

Comparison with Existing Approaches

- Face recognition can be done without any explicit action on the part of user.
- Other techniques are either expensive or can be foiled easily whereas facial recognition can be easily achieved using inexpensive cameras.
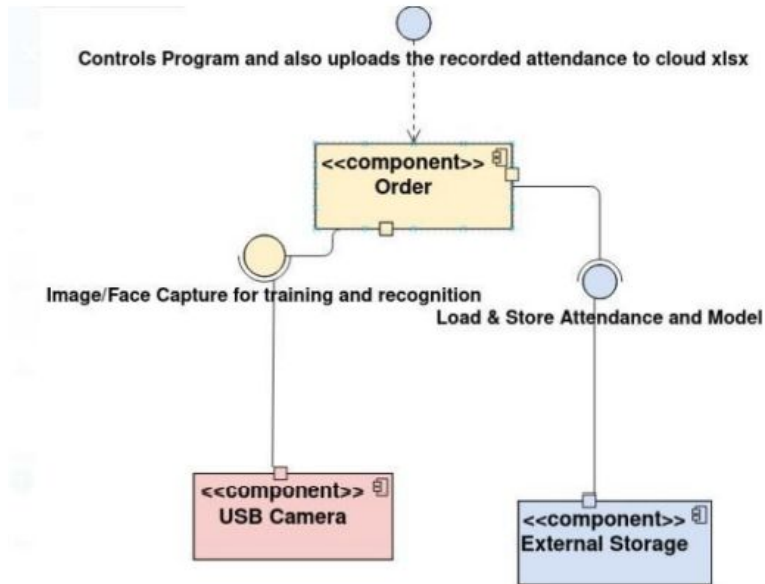- It is non-intrusive and does not carry any health risk.
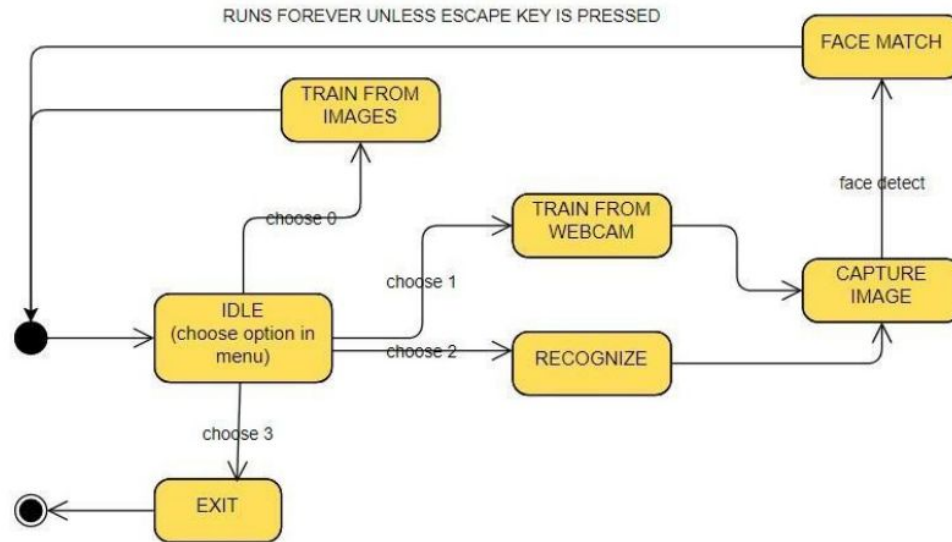
Solution Approach - Use Case Diagram
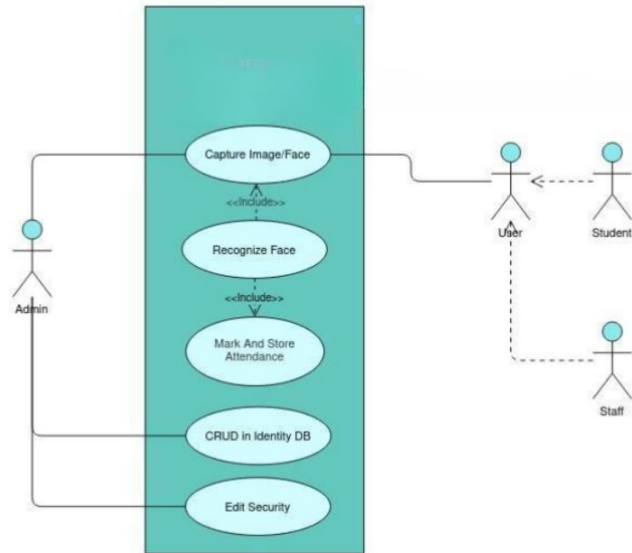
Solution Approach - Activity Diagram
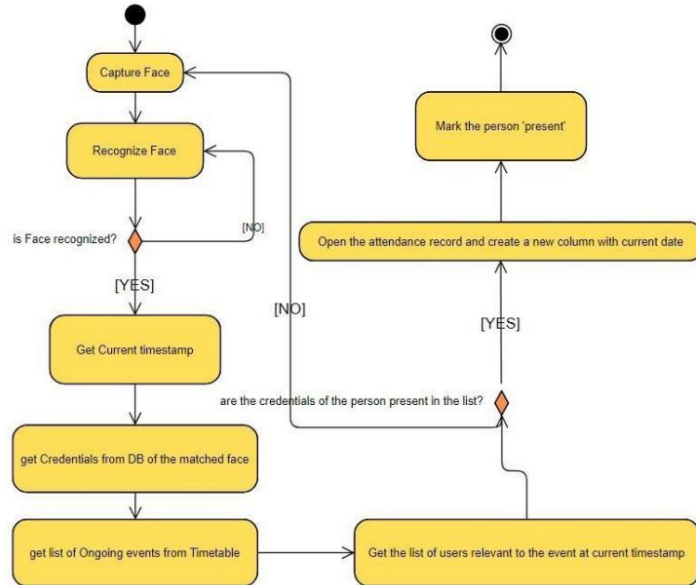
Solution Approach - Component Diagram

Solution Approach - State Diagram

Solution Approach - Modules

MAIN.PY - controller code and also contains functions for our model's training

      Contains the following modules:

            TRAIN_FROM_WEBCAM

            START_RECOGNITON

APP.PY- contains code for web application

Time for Demonstration

Our code is also available on Github:

https://github.com/Major-Project-2-0/major-project-2

Issues

- We had firstly aimed to implement this project on a Raspberry Pi, but due to low memory, hardware, and voltage problems, the library opencv-contrib could not be installed in a feasible amount of time.
- Since the accuracy and the speed of the LBPH Model was not up to par with the needs of this project, we decided to use KNN Model instead, which works faster in our case and also gives more accurate results. Specifically KNN is faster than LBPH in recognition.

Results and Future Scope

- The Command Line Version of our system is working as expected. But there is scope for improvement as we can divide the respective date's attendance record into multiple attendance records separated by different classes, i.e, each attendance record of a particular class.
- Our web application on the other hand is not that stable. And its functionality is not yet complete. We have to work upon its stability and other modules like 'list all users' and 'delete model and retrain model functionality'.

Results and Future Scope

- Attendance search and filter functionality can be added later inside the web Application. The Admin should be able to mark a missing attendance in case of an error raised by a student with proper permission.
- Authorization and Session support for users to see their attendance and weekly/monthly reports along with other details like change in attendance and percentage of classes attended.

References

1. https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning
2. https://www.tutorialspoint.com/flask/index.htm
3. User guide: contents — scikit-learn 1.2.2 documentation

Thank You

Group-17

Team Details:

- Rishabh Rajpurohit 9919103096 F4
- P Paul Jonathan   9919103121 F4
- Yash Raj Gairola   9919103150 F5