SMART ATTENDANCE SYSTEM v2.0

Enroll. No. (s) - 9919103096, 9919103121, 9919103150
Names of Student (s) - Rishabh Rajpurohit, P Paul Jonathan, Yash Raj Gairola

Department of CSE/IT
Jaypee Institute of Information Technology, Noida

March 2023

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, NOIDA

Table of Contents

Chapter-1 Introduction

1.1 General Introduction

Facial recognition is a powerful computer vision application that has received significant attention in recent years, owing to its potential application in diverse fields such as security, entertainment, and healthcare. In this study, we present a new version of our previous facial recognition model that leverages the K-nearest neighbors (KNN) algorithm to improve accuracy and efficiency.

The KNN algorithm is a popular machine learning technique that is frequently used for classification and regression tasks. It works by identifying the K-nearest neighbors to a particular data point in a training dataset and then predicting the label of the data point based on the labels of its nearest neighbors. In our new facial recognition model, we use KNN to classify faces based on their unique features, comparing them with a set of known faces in a training set.

1.2 Problem Statement

In our previous iteration of the project, we encountered issues with slow speed and large library sizes when running our facial recognition application on a Raspberry Pi. One of the key challenges was the slow speed of the Local Binary Pattern Histograms (LBP) algorithm, which we used to extract facial features in the earlier version. To address this issue, we have revamped our model to use the Haar Cascade classifier for feature extraction and the K-nearest neighbors (KNN) algorithm for classification. By using KNN, we were able to overcome the speed limitations of LBP and achieve faster and more accurate facial recognition. To further enhance the performance of our application, we have also decided to develop a web application that can leverage the processing power and resources of a remote server.

1.3 Significance/Novelty of the Proposed Solution

Smart Attendance systems are now easier to develop and install than ever before. They are present in most of the companies/universities, basically every institution involving employees constitutes them. But our focus is based on how we can further optimize it or make it more convenient. Our idea is to use a webcam for facial detection. We are using Haar cascade for facial detection and the KNN algorithm for facial recognition. This is much faster than the lbph algorithm. We also want to make it more user friendly as a web application portal which we are currently working on other than the command line interface which currently exists.

.

1.4 Empirical Study/Field Survey/Technology Used

1.4.1 Haar Cascade

It is an Object Detection Algorithm used to identify faces in an image or a real time video. The algorithm uses edge or line detection features proposed by Viola and Jones in their research paper “Rapid Object Detection using a Boosted Cascade of Simple Features” published in 2001. The algorithm is given a lot of positive images consisting of faces, and a lot of negative images not consisting of any face to train on them. The objective is to find out the sum of all the image pixels lying in the darker area of the haar feature and the sum of all the image pixels lying in the lighter area of the haar feature. And then find out their differences.

1.4.2 KNN

K-Nearest Neighbor (KNN) is a popular machine learning algorithm used for both classification and regression tasks. In classification, the algorithm works by identifying the K-nearest neighbors to a particular data point in a training dataset and then predicting the label of the data point based on the labels of its nearest neighbors. The K in KNN represents the number of nearest neighbors to consider when making a prediction. The algorithm can use various distance metrics, including Euclidean distance and Manhattan distance to determine the proximity between data points. One of the key advantages of KNN is that it is a non-parametric algorithm, meaning it does not make any assumptions about the underlying data distribution. Additionally, KNN is relatively simple to understand and implement, making it a popular choice for many machine learning tasks.

1.4.3 Flask

Flask is a Python framework that is used for building web applications. Flask is a back-end framework, which means that it provides the technologies, tools, and modules that can be used to build the actual functionalities of the web app rather than the design or look of it1. Flask is used for building web applications, APIs, and microservices2. Flask is simple and lightweight, which makes it easy to learn and use3. Flask is also extensible, which means that developers can add more functionality to it as needed3.

1.5 Brief description of Solution Approach

Our attendance management system involves preprocessing of faces by extracting faces from images and resizing them to 50x50 pixels. The System should be able to scan 50 or more images of a user and store them in a Database DIrectory with the user's name. After that we train the KNN model and store it on a hard drive. During face recognition and taking attendance, a user's face is captured, preprocessed and predicted using the K-Nearest Neighbors (KNN) algorithm for classification. Our project is designed to be faster, making it efficient for educational institutions and organizations to manage and store attendance records. We propose to develop a web application with an improved user interface that allows users to view attendance records, generate reports, and manage classes and schedules. The web application will also provide text-to-speech support for visually and auditorily impaired users. The Attendance folder in our project directory should store a record of CSV files named according to the current date, including details such as Subject Name, Class Time, Class Type, and Faculty Name. To avoid redundancy in records, we maintain a Python list that is checked each time we recognize a new face. If the face is not found in the list, we mark them present and store the details accordingly. Additionally, our system has a proximity requirement of 50 cm for accurate face detection.

1.6 Comparison of existing approaches with the proposed solution

Biometric-based techniques are promising options for recognizing individuals in recent years. Biometric-based technologies include identification based on physiological characteristics (such as face, fingerprints, finger geometry, hand geometry, hand veins, palm, iris, retina, ear and voice) and behavioral traits (such as gait, signature and keystroke dynamics). Face Recognition appears to offer several advantages over the above mentioned methods, a few of which are outlined here:

a. Almost all these technologies require some voluntary action by the user, i.e The user needs to place his hand on a hand-rest for fingerprinting or hand geometry detection and has to stand in a fixed position in front of a camera for iris or retina identification. However, face recognition can be done without any explicit action or participation on the part of the user since face images can be acquired from a distance by a camera. This is particularly beneficial for security and surveillance purposes.
b. Data acquisition in general is a drawback with problems for other biometrics techniques that rely on hands and fingers that can be rendered useless if the epidermis tissue is damaged in some way. Iris and retina identification require expensive equipment and are much too sensitive to any body motion. Voice recognition is susceptible to background noises in public places. However, facial images can be easily obtained with a couple of inexpensive fixed cameras. Good face recognition algorithms and appropriate pre-processing of the images can compensate for noise and slight variations in orientation, scale and illumination.

c. Finally, technologies that require multiple individuals to use the same equipment to capture their biological characteristics potentially expose the user to the transmission of germs and impurities from other users .However, face recognition is totally non-intrusive and does not carry any such health risks.

Chapter-2 Literature Survey

2.1 Literature studied

- Paper 1

Face Recognition and Identification using Deep Learning Approach, KH Teoh, RC Ismail, SZM Naziri, R Hussin, MNM Isa and MSSM Basir, School of Computing and Informatics, Albukhary International University, Kedah Malaysia, School of Microelectronic Engineering, Universiti Malaysia Perlis, Perlis, Malaysia, Department of Electrical Engineering, Politeknik Mukah, Sarawak, Malaysia.

Basically, authors here used haar cascade for face detection. They have found that there are several cases in which haar cascade does not detect faces which are a distance below 60 cm from the camera tha haar cascade do not detect faces. For the model they have used convolutional and deep neural networks for the model which have accuracy of 91.7% in recognising images and 86.7% accuracy in video feed.

- Paper 2

Face Recognition System, Shivam Singh, Prof. S. Graceline Jasmine, Department of SCSE, Vellore Institute of Technology, Chennai, Tamil Nadu, India

Paper uses haar_cascade for frontal face detection, preprocessing: face image is resized to 50x50 histogram normalization used to make image more clear, db_storage: facial features are stored in the database, post processing: names are shown in video output. This paper compares a few algorithms for facial recognition which are: Neural Networks, Principal Component Analysis, Eigenfaces, Fisherfaces, Local Binary Pattern Histograms. Basically , here they used KLT algorithm, viola jones used haar cascade classifier and PCA for feature extraction. In this project , they have also run some experiments on other facial recognition algorithms. They have found that the PCA algorithm outperforms all the other algorithms. And he has also acknowledged that poor lighting is a disadvantage here.

- Paper 3

Face Recognition with Local Binary Patterns, Timo Ahonen, Abdenour Hadid, and Matti Pietik ̈ainen, Machine Vision Group, Infotech Oulu, PO Box 4500, FIN-90014 University of Oulu, Finland.

In this paper, they have described an approach of recognising faces using local binary pattern histogram. They described how the approach of lbpf works, in lbpf there are several formulas for calculating distance between two histograms to find out the classification. In this paper, they have run some performance tests on several algorithms and distance measures, using the CSU face identification and evaluation system using the ferret test. the results of the test were that in LBP category, fb, fc and dup distance measures were outperforming the others with 79% accuracy and PCA was 65%, BIC 37% and 42% for EBGM.

2.2 Integrated Summary of the Literature Studied

Since, from all these research papers we can see that, lbph has the highest accuracy rate, across 10,000 + images and also is the easiest to implement. so, we have used lbph in our project. All the research papers have used haar cascade (viola jones algorithm) for face detection. One paper used an interesting approach by comparing different types of distances between histograms, to find out which distance is more optimized and give better results. for example, chi square distance, euclidean distance, manhattan distance etc. Chi square had the best results which was fascinating to know .

Chapter-3 Requirement Analysis and Solution Approach

3.1 Solution Approach (overall and module wise detailed description of our proposed solution)

Fig-1 Use Case Diagram

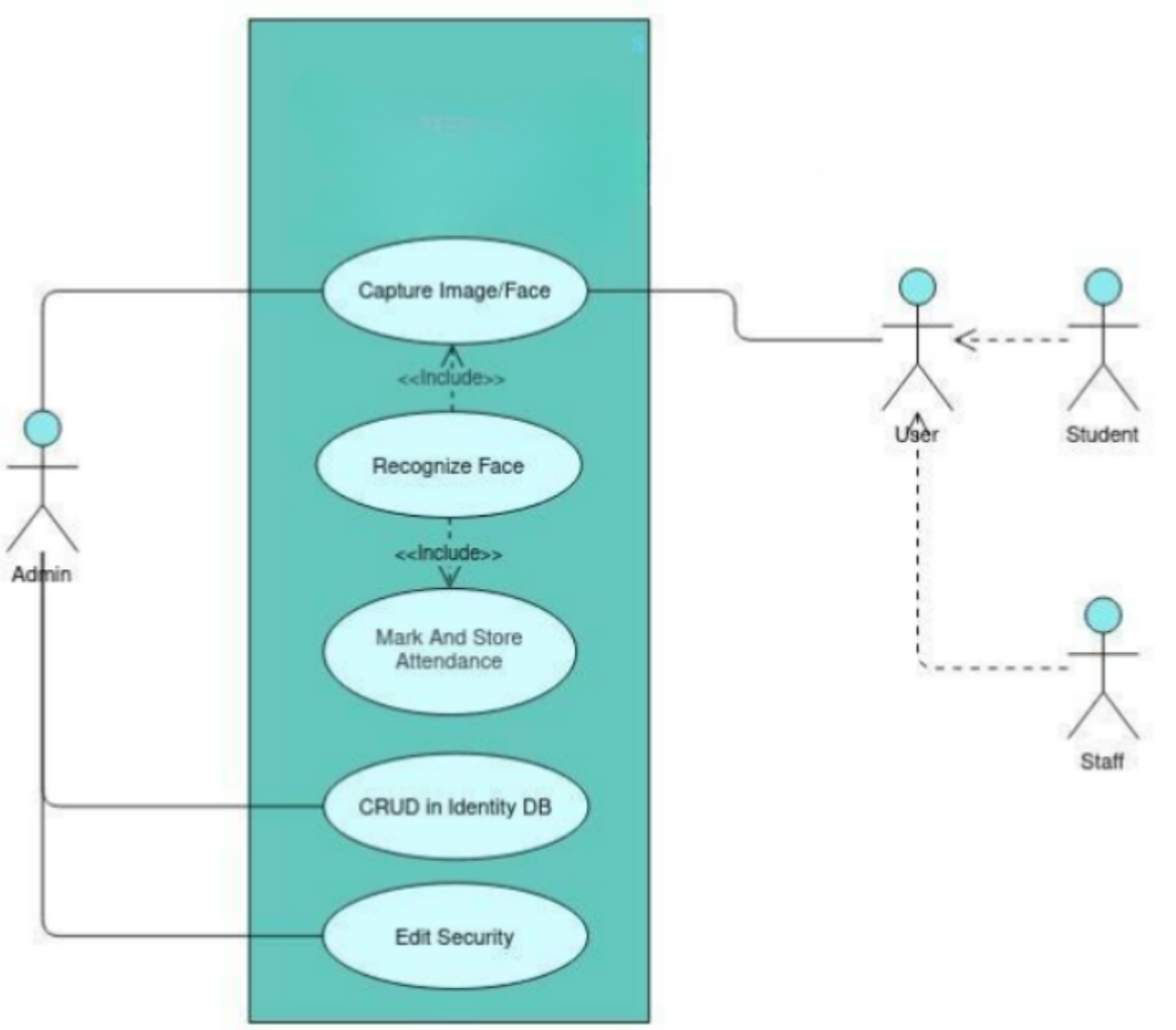


Fig-2 Activity Diagram

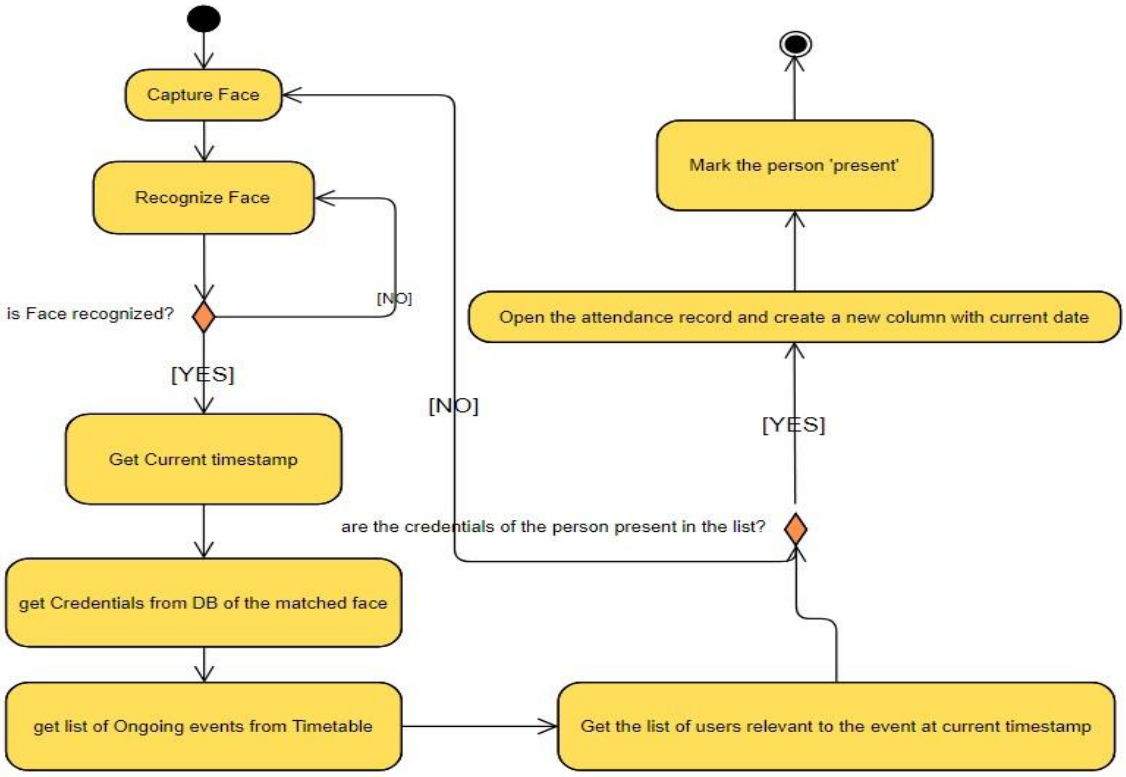
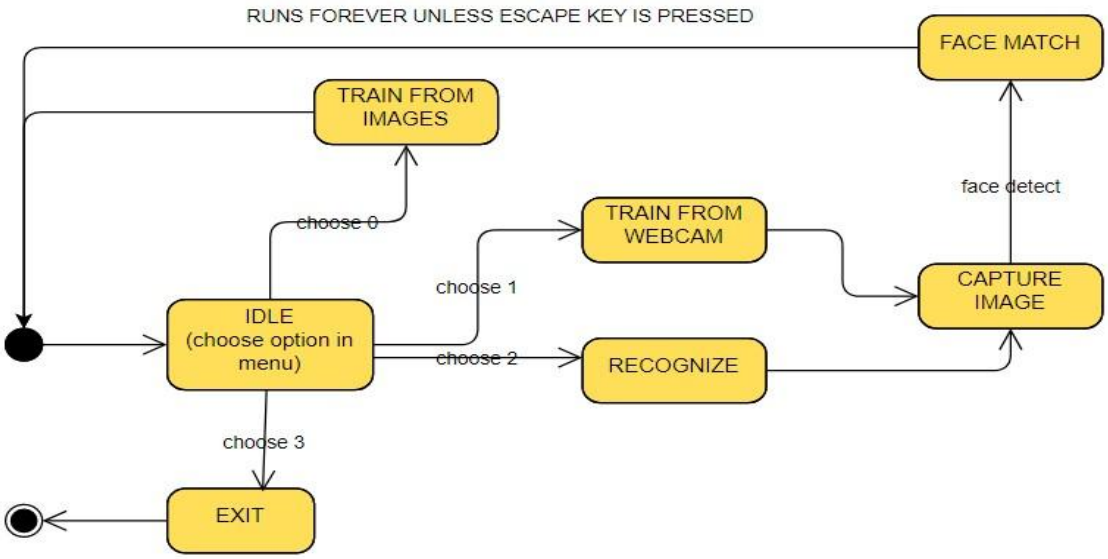


Fig-3 State Diagram



Our code is divided into 2 files described as follows:

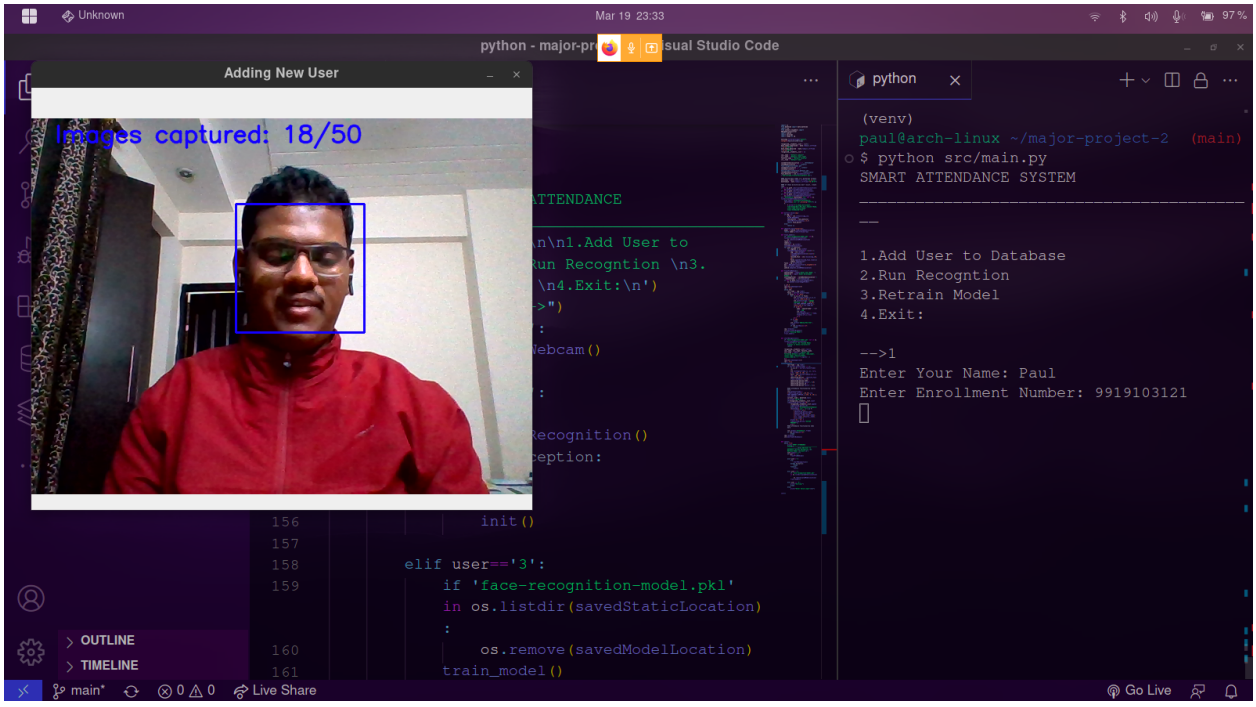
- Main.py: This contains the most important part of our project. It contains the control logic along with KNN classifier and functions like take attendance and face capture functionality.
- App.py: This is the code for web server functionality using Flask. It uses an index.htm page as template for rendering.

Chapter-4 Modeling and Implementation Details

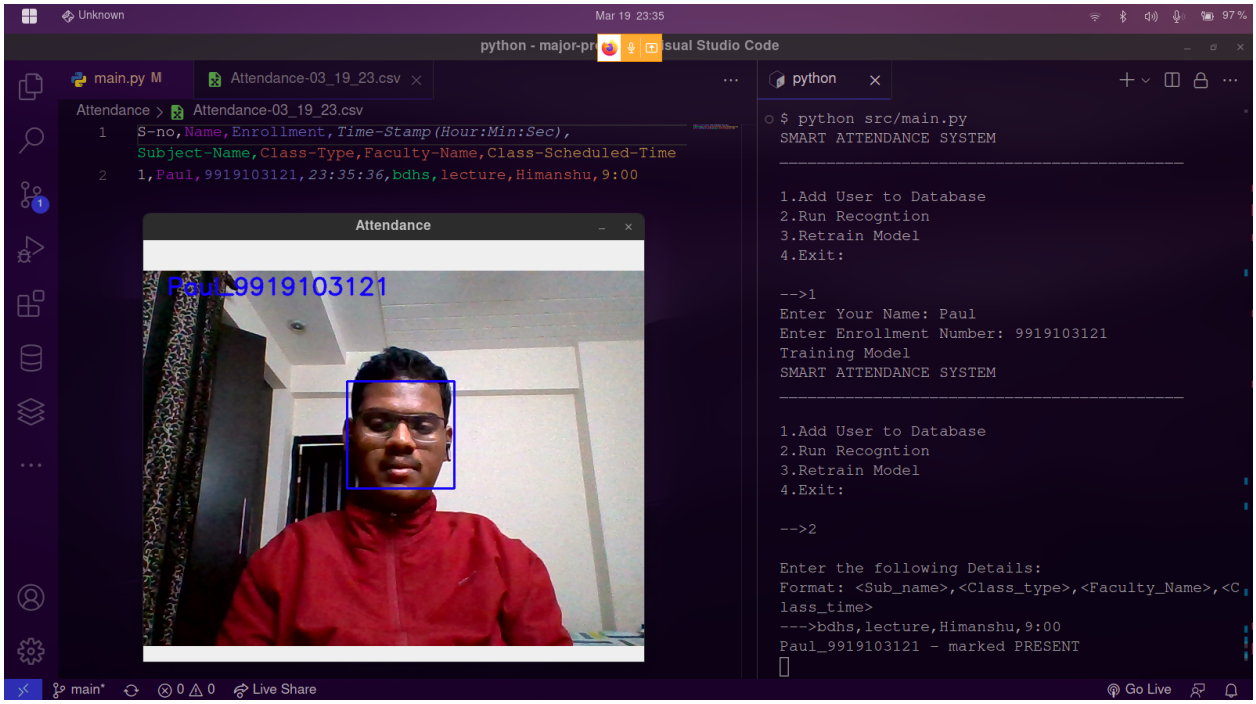
4.1 Implementation and Run-Through

In the Project code we have the Following steps:

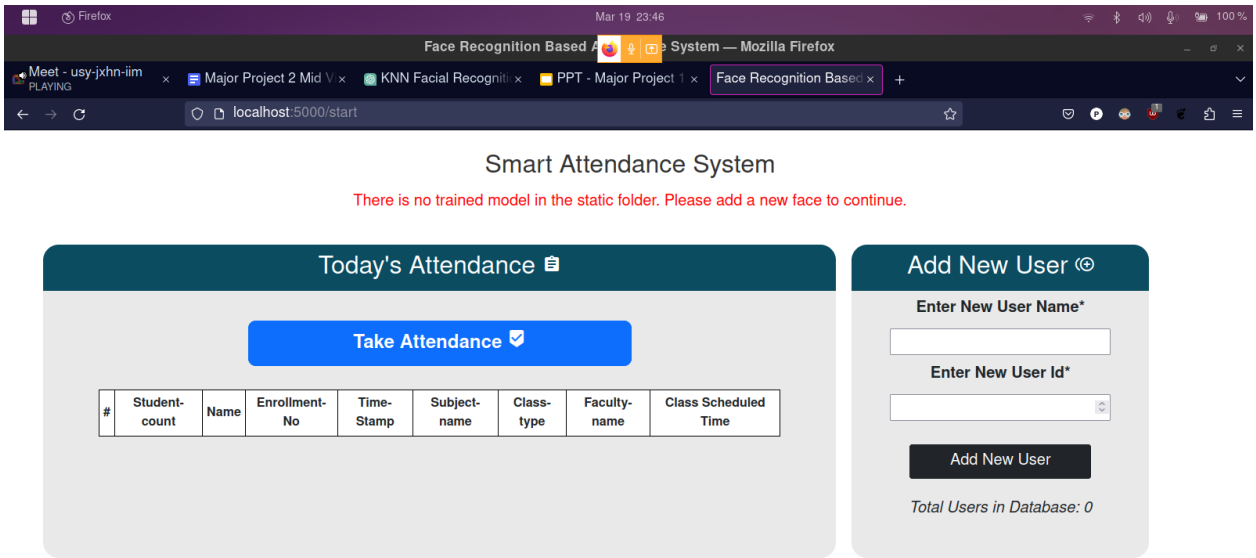
- Training from live camera feed - The program asks for a label name, The camera is turned on and the program captures images from the camera feed and detects faces using haar_cascade and stored in the data folder under the label name given



- Recognizing the person in front of the camera - The camera is turned on and the program captures images from the camera feed and detects face using haar_cascade and gives the faces to the model which finds the closest match and overlays the label on the camera feed and marks the attendance on the csv file



- Web Application - We have also planned to make a web application and have made progress in it, here is a screenshot of the main page



4.2 Issues and Remaining Work

- 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'.
- 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.

Chapter-5 Result

We successfully demonstrated that our system can detect and recognize faces present in the database and act accordingly and it can also generate structured Attendance records each day in a separate file. Also Attendance of different classes are well segregated in groups each starting from Serial No. 1. The last entry in each group can be used to deduce the total strength of class. In the future we can integrate data handling and analysis functionality like Generating Reports both weekly and monthly. We are currently working on a web Portal Interface for this system. In short the possibilities are endless.

Chapter-6 Conclusion

Smart Attendance Systems based on face recognition techniques thus proved to be time saving and secured. Earlier we wanted a better face recognition model so we switched to KNN from LBPH. KNN proved to be faster and more accurate. We have started developing a web application portal for better user interface. There is much scoop for adding new functionality to the system. Poor lighting conditions may affect image quality which indirectly degrades system performance. Our system will perform well in a well lit environment.

References

[1] K. H. Teoh, et al., “Face Recognition and Identification using Deep Learning Approach”, Journal of Physics: Conference Series, 2020. Available: https://iopscience.iop.org/article/10.1088/1742-6596/1755/1/012006/pdf

[2] T. Ahonen, A. Hadid, M. Pietikainen. "Face Recognition with Local Binary Patterns", in Conf.
European Conference on Computer Vision, Prague, Czech Republic, May 11-14, 2004, pp.470-482.
Available: https://www.researchgate.net/publication/221304831_Face_Recognition_with_Local_Binary_Patterns

[3] S. Singh, S. G. Jasmine, “Face Recognition System”, International Journal of Engineering Research & Technology (IJERT), vol. 8, issue 5, pp. 263-263, May 2019. Available: https://www.ijert.org/research/face-recognition-system-IJERTV8IS050150.pdf