

**TRIBHUVAN UNIVERSITY**

**INSTITUTE OF ENGINEERING**

**PULCHOWK CAMPUS**

**Project Proposal  
on  
Smart Attendance System**

**SUBMITTED BY:**

Nikesh D.C. (075BCT052)

Ravi Pandey (075BCT065)

Rohan Chhetry (075BCT066)

Yukta Bansal (075BCT096)

**SUBMITTED TO:**

Department of Electronics and Computer Engineering



**TRIBHUVAN UNIVERSITY**

**INSTITUTE OF ENGINEERING**

**PULCHOWK CAMPUS**

**A Minor Project Report**

**On**

**SMART ATTENDANCE SYSTEM**

**Submitted by:**

**Nikesh DC/075BCT052**

**Ravi Pandey/075BCT065**

**Rohan Chhetry/075BCT066**

**Yukta Bansal/075BCT096**

**Submitted to**:

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

LALITPUR, NEPAL

(January 2, 2022)

**ACKNOWLEDGEMENT**

Firstly, we want to convey our thanks to our department for giving us this opportunity to work on our project of choice. We extend our sincere gratitude to your seniors who have provided us the necessary materials and motivated us to take upon this project. All our teachers and classmates deserve humble appreciation for providing us their suggestions on the selection of this project and its enhancement ideas.

**ABSTRACT**

The project of our choice, namely Smart Attendance System will be a working product that makes the monotonous and tedious task of taking attendance a more fun, technical and easier process.

Our project aims to create a mobile application capable of identifying each student from a group and marking their attendance. There may be some difficulties while identifying faces in all the different lighting instances and so we will also be providing a manual method to edit the attendance taken.

**TABLE OF CONTENTS**

**1. INTRODUCTION**

**1.1. Background**

Records of attendance and verified authentication is crucial for many organizations and thus require an effective and efficient system for maintaining the attendance records or identifying correctly an individual. Manual book-keeping and verification have been in use since ancient times but they are cumbersome, labor-intensive, prone to record loss and false verifications. Keeping such problems in mind, various automations have been introduced in recent years. The use of biometric recognitions is one of the fields that has been gaining widespread popularity. Biometric authentication relies on use of distinctive and measurable traits of human beings such as fingerprint, palm print, palm vein, iris/retina, voice, electroencephalogram, electrocardiogram, face, etc. Biometrics seems to be a robust candidate technology for individual authentication. More conventionally, token based identification such as RFID, barcode, QR code, NFC have also been used but their limitation on enforcing validness is pretty obvious. For selection of biometrics for certain application Jain et al. [REFRENCE NO.] identified seven factors: universality, uniqueness, permanence, measurability, performance, acceptability and circumvention.

For fingerprint recognition, a portable fingerprint recognizing device is first configured with the fingerprints of attendees and then the attendees would register their fingerprint later for attendance. This is usually carried in serial fashion/queue (when there are large number of attendees at a time). Also, a problem with fingerprint or palm recognition is that its accuracy plummets when the finger is wet or dirty. Similar problems are observed with using iris or retina-based systems for attendance. In recent years, work in face recognition has been expanding rapidly and accuracy as well as adaptability have been considerably increased making it a strong contender for biometrics-based system**.** Several methods have been introduced recently for authentication through face recognition, which can be classified as feature-based and image-based. Feature-based methods rely on predefined traits on a human face and use them to distinguish one face from another. However, this approach is slowly being dominated by the image-based approach which statistically compares the actual face image for recognition.

According to Center for Strategic and International Studies (CSIS) (2020), verification algorithms used to match subjects to clear reference images can achieve accuracy scores as high as 99.97% on standard assessments like NIST’s Facial Recognition Vendor Test (FRVT) which is comparable to best results of iris scanners.

**1.2. Problem statement**

A traditional method of recording student attendance is by calling out their roll numbers and marking the attendance manually. The task of taking attendance of students manually is time-consuming and prone to proxy attendances such as marking someone present despite their physical absence. A change in the attendance marking system is long overdue and our motivation for this project.

**1.3. Objective**

To mark the attendance of a student by using facial recognition and manage attendance logs of different classes in an attendance database.

**1.4. Scope**

Though the proposed system is mainly intended for economically and efficiently automating attendance of students in educational institutions, similar or slightly modified methods can be installed in varieties of organizations for different purposes such as surveillance.

**1.5. Applications**

* Semi-autonomous and reliable attendance of students during lectures in educational institutions.
* Attendance in programs with large number of attendees such as workshops/seminars.

**2. LITERATURE REVIEW**

A brief look at recent works shows that the concept of face recognition and its variations have been incorporated into attendance management and similar systems by several groups and organizations. Kawaguchi, Yohei ,et. al.[1] illustrated a system that takes attendance of students for a classroom lecture by continuous observation of the lecture’s video feed. They highlighted difficulties in estimating attendance precisely using face recognition independently as the accuracy of face recognition is not sufficiently high leading to misrecognition of faces. To remedy this, they suggested using constraints of correspondence between sitting position of students and their faces.

An android based face recognition attendance system using linear discriminant analysis was proposed by Sunaryono et. al.[2] with 97 percent accuracy and required 0.000096s to recognize a face image in the server. Their proposed system mandated students to capture their face image and displayed QR code using their smartphone.

Jadhav et. al.[3] put forth an automated attendance system which automatically detects the student face when he/she enters the class using a camera mounted near the entrance and then marks their attendance based on face detection and recognition algorithms.

Kadry Seifedine and Smaili Khaled[4] proposed a system in which the attendance is done through biometric recognition employing iris matching. The minutiae are extracted by employing feature extracting algorithm from a digital image of the person's eye taken by the system which is then stored as template for verifying later. For verification, a person places his eye on the iris recognition sensor which then extracts the minutiae from the captured image and then sorts through a matching algorithm using the previously stored template.

The system proposed by Khuhawar et. al.[5] records daily attendance of students automatically subject-wise according to an administrator defined schedule. Their system uses a fixed camera which automatically snaps image of the classroom at scheduled time. The system then applies face detection using Histogram of Oriented Gradient and deep learning techniques to calculate-and-compare 128-d face features for face recognition. The data of successfully recognized students is recorded in an excel sheet automatically.

These approaches towards creating an attendance system using face detection and recognition have varying applications in their respective use cases. In this project, we take motivation from these systems and several other similar works and aim to create a robust system for making the attendance taking procedure semi-autonomous and hassle-free.

**3. METHODOLOGY**

The attendance system is intended to be developed in two major divisions: An attendance server and a mobile application for attendance marking. The mobile application will have two variations: one for the attendance taking entity, hereafter referred to as teacher and one for the attendee, hereafter referred to as student. The applications for teacher and student will have different features according to their usage. The design of the various parts of the system is explained along with system block diagram below:

**3.1. System Block Diagram**

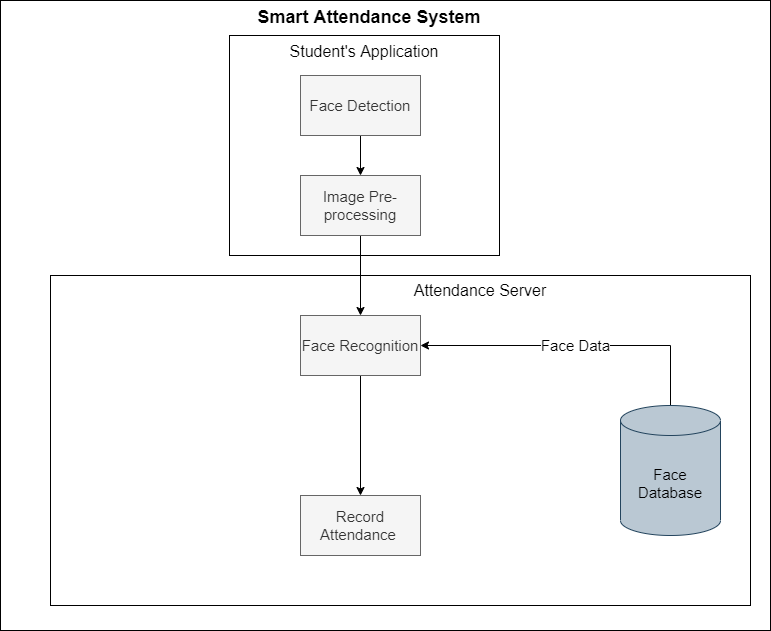
****

Figure 1:- Smart Attendance System Block Diagram

**3.2. Working Principle**

When the teacher wants to take attendance, he/she connects to the attendance server through his/her mobile application. The attendance server generates a random unique code and sends it to the teacher’s mobile application, which is then shared with the students manually to initiate the attendance for a lecture. The students then enter the code into their application and capture a self-image. The captured image is processed and sent to server for face recognition along with their unique identifier which for most case is their roll no which act as constraint for improving accuracy. Upon successful recognition of a student from the lecture’s attendee, the student is marked as present and the information is feedback to student. If the student can’t be recognized the student will be prompted to recapture their image. The teacher can turn off the attendance period when all present students have been recorded or after certain interval from start of attendance taking process so that proxy attendance of late students will be discouraged.

Each component of the block diagram is explained in detail below:

**3.2.1. Student’s Application**

A mobile application for the teachers is used to receive the unique code generated by the attendance server. The code must to entered by the student in order to start the attendance process. This is done to ensure the proper timing of the attendance irrespective to stringent routines but with respect to the event (lecture delays, changes or cancellation). After entering a valid code, the application will prompt the user to capture a self-image for face recognition and enter his roll no. The captured image is then processed as follows:

**3.2.1.1. Face Detection**

The first step of any face recognition is face detection. The presence of a face in the captured image is detected by applying a frequently used face detection algorithm proposed by Paul Viola and Michael Jones[1]; the Viola-Jones object detection algorithm. The algorithm uses Haar-like Features to localize the probable location of the face in the image, uses an integral image created from the original image for efficient computation and applies cascaded classifiers to detect the face. The portion in which the face id detected which is represented by a rectangular bounding box containing the face features is then passed over to the image pre-processor.

**3.2.1.2. Image Pre-processor**

This system first aligns face based on facial landmarks to obtain a canonical alignment based on translation, scale and rotation as it could provide higher accuracy[1]. Since the proposed system is based on the feature-based approach of face recognition, so the face image is processed to extract 128d face embedding data[2]. These feature data, here referred as face embedding, are unique to each distinct face and hence can be used for recognition. The face embedding is then sent to the attendance server. Since the face embedding is only of 128 bytes it places minimal constraint in network and server’s performance.

**3.2.2. Attendance Server**

A server is used to manage the attendance logs of all students for corresponding lectures.

**3.2.2.1. Face Recognition**

The server receives face embedding data obtained after the processing of the image along with roll no from the student application. Due to use of student’s roll no as added constraint, face recognition is simplified to face verification which makes computation immensely faster as the comparison is reduced from k-NN classification to 1x1 comparison. This should allow our attendance server to handle large number of users simultaneously. The face data is then compared against the face data of students stored in the database for similarity using Euclidean L2 distance[1]. Recognition is considered successful when the similarity score between the images exceeds a threshold value and the respective student is marked as present and notified to student in real-time. In the lower possible case of failure of recognition, the server prompts the student application to recapture image of the student.

**3.2.2.2. Record Attendance**

The attendance of all successfully recognized students is marked and the logs are stored in a central database. A list of present students and absentees is sent to the teacher application. The attendance data can be exported in the form of excel file.

**3.2.2.3. Face Database**

A central database is maintained locally in the attendance server which holds face data for all the students enrolled in the organization. The face data is generated at the time of enrollment of a new student. The face data can be individually updated through the student application with time automatically, which will increase success probability for successful recognition as student’s face naturally varies in time by process such as ageing or simply by change of appearance.

**3.2.3. Teacher’s application**

A separate mobile application is provided to the teachers. Through this app, they will start the process of attendance and control its active time or stop attendance after class is over to prevent proxy late attendance. They will also be able to view records present in the attendance server.

Requirements

Tentative Applications to be used in the development

The system will be built using the Open Source Computer Vision (OpenCV) library, which will provide the necessary functionalities for face detection and recognition, in Object Oriented programming paradigm using python programming language. We will be using Kivy, a cross-platform python framework, to build the mobile applications.

Required Resources  
 1. List of students along with pictures for creating Face database

2. A computer for attendance server

3. Lan connection for server and smartphones of teachers and students

We will be referencing various materials such as OpenCV documentation, Python documentation and Kivy manuals along with the papers cited in the references section to gather the necessary information to initiate the development process. We will be taking help from some tutorials and articles on face recognition systems online in order to code the various modules of the system and after proper unit testing of each modules, we will be assembling them into a complete attendance system.