
**ARTIFICIAL INTELLIGENCE LAB**

**PROJECT REPORT**

# SMART ATTENDENCE SYSTEM

**SUBMITTED BY:**

HAFIZ MOIZ ALI (202101009)

AMEER ADAM SATTI (202101027)

M. DANYAL ABBAS (202101034)

**SUBMITTED TO:** ENGR. ZIA-UR-REHMAN

**SEMESTER:** BSCE-VII

---

**DEPARTMENT OF COMPUTER ENGINEERING**

**INSTITUTE OF SPACE TECHNOLOGY**

**KICSIT, KAHUTA CAMPUS**

## 1. Introduction

Attendance is a mandatory practice in educational institutes as well as in organizations to record the performance of the attendees. There are two major types of attendance systems which include manual attendance system and automated attendance system. A manual system refers to the practice in which the instructor manually marks the attendance of students on attendance sheets. In the case of large number of students, instructors often face enormous pressure while maintaining the attendance record. Automated attendance system such as facial attendance system is based on facial recognition technology which detects faces that are captured by the camera. The image is captured and matched with the database. The facial attendance system uses this technique to capture images of the person and match them with the database. Whenever an image is recognized, the attendance and the arrival time of the person will be recorded in the database. The attendance system also maintains records of the intruders by capturing their image and storing it in an additional folder with the tag of 'Unknown'. The system will be deployed in educational institutes and organizations. A paperless record will also decrease paper records and ultimately save time that is spent during the maintenance of paper records. The main advantage of the system is that it automatically marks the attendance of students and employees without any human interface which states that the system is fast and reliable. Secondly, the system is also accurate as it eliminates errors and fake attendance calls by students. To achieve accuracy, we have deployed a lock system that first confirms the person in the camera feed three times before labeling and marking attendance.

## 2. Objectives

The main objective was to design a fully functional application that helps and improves the traditional systems of attendance. The facial recognition technique is applied in the system to detect the faces of people and match them with the dataset. The goal was to make a face recognized if it is present in the database. Once it is recognized then the system creates a database that contains the name and arrival time of the person. Furthermore, if the images are not recognized then the system also captures the images of the intruders with their entry time. The purpose of this application is advanced and can be used in terms of security.

## 3. Requirement Analysis

Here is a list of the software and hardware tools that were used in the design of the system.

### 3.1 Pycharm Community Edition 2021.3.1

Pycharm is an integrated development environment used for python programming. We used pycharm community edition to write our code as it is open-source software.

### 3.2 MS Excel 2016

MS Excel is a Microsoft Office application that is used to record and analyze numerical data. In the proposed system, it was used to maintain and record the attendance of students/employees.

### 3.3 Python

Python is a general-purpose, high-level programming language. Python is the major programming language for AI and ML because it is easy to use, extensible, and has a vast number of libraries and a large community of python developers. As our system is also an application of Artificial Intelligence so we used python.

### 3.4 OpenCV

OpenCV (Open Source Computer Vision) Library is an open-source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in commercial products. In our system, we used OpenCV for image processing and accessing the camera.

## 4. Working

Our Proposed System is capable of marking and maintaining attendance records by recognizing the faces of the students. The working of this system is mainly divided into three phases. In the first phase, we provided the system with the dataset and the label associated with it. In the next phase, we implemented face recognition. And in the last phase, we marked the attendance and identify unknowns.

## 4.1 Phase 1

For implementing the first phase, we used the operating system (OS) OS function of python. It allows us to run a command in a python script, just like it was running in shell. The system got the images of the students from the dataset path that is provided. The dataset contains one clear image for each student saved with his or her name. Next, we saved the images and their corresponding labels (names of the students) in a list. We also created a copy of the label that helped us in preventing the multiple attendance of the students.

## 4.2 Phase 2

The second phase is the recognition part. Face recognition consists of four steps. In the first step, the system identifies all the faces present in the image.

1. In the first step, the system identifies all the faces present in the image.
2. In the second step, the system focuses on each face and understands that even if a face is turned in a weird direction or in bad lighting, it is still the same person.
3. Third step is to pick out unique features of the face that will be used to differentiate the features like how big the eyes are, how long the face is, etc.
4. Finally the system compares the unique features of that face to all the people in the dataset to determine the person's name.

We build separate modules where we solve each step of face recognition separately and then move on to the next step. In other words, we will merge different algorithms together.

### 4.2.1 Step 1-Finding all the Faces

The first module was face detection. We need to detect the faces in an image before we can try to tell them apart. For this, we used an algorithm called Histogram Orientation of Gradients (HOG). To detect faces in an image, we converted the image into greyscale. Then we analyzed every single pixel in our image one at a time. We looked at the pixels that are directly surrounding it. Our goal was to find out how dark the current pixel is as compared to the pixels directly surrounding it. Then we drew an arrow towards the direction in which the image was getting darker. This step was repeated continuously for every pixel and we ended up with an arrow with every pixel. These arrows are known as gradients and they show the flow of light to dark across the entire image.

### 4.2.2 Step 2-Posing and Projecting Faces

Now we had to deal with the problem that faces when turned into different positions they are not recognized by the computer. We attempted to warp each image so that the eyes and lips were always at the sample location. This helped us to compare faces in the next steps. To do this, we used an algorithm of face landmark estimation. The main concept of this algorithm is to identify 68 distinct spots on every face, such as the top of the chin, the outside corner of each eye, the inner corner of each eyebrow, etc. Then, using machine learning, we trained an algorithm that was able to locate these 68 precise locations on any face. We just rotated, scaled, and sheared the image so that we knew where the lips and eyes were.

### 4.2.3 Step 3-Encoding Faces

We reached the core of the problem of differentiating between different faces. We require a method to extract basic measurements from each face. Then we may take the same measurements of our unidentified face and compare them to those of the nearest known face. For instance, we could count the size of the ears, count the distance between the eyes, count the length of the nose, etc. Training a Deep Convolutional Neural Network is the answer. We are going to train the network to generate 128 measurements for each face rather than train it to recognize images. We're going to make use of the OpenFace tool, which will produce the measurements for us and save them in a NumPy array. OpenFace is an open-source tool used in computer vision and machine learning, and by the effective computing community and people who are interested in building interactive applications based on facial behavior analysis.

### 4.2.4 Step 4-Finding the person's name from the encoding

This last step is actually the easiest step in the whole process. All we have to do is find the person in our database of known people who has the closest measurements to our test image. We'll use a simple linear SVM classifier for this purpose . We train a classifier that takes in the measurements from a new face in the webcam and tells which known person is the closest match. Running this classifier takes milliseconds. The result of the classifier is the name of the person.


**Fig 1. Block Diagram of Attendance System**

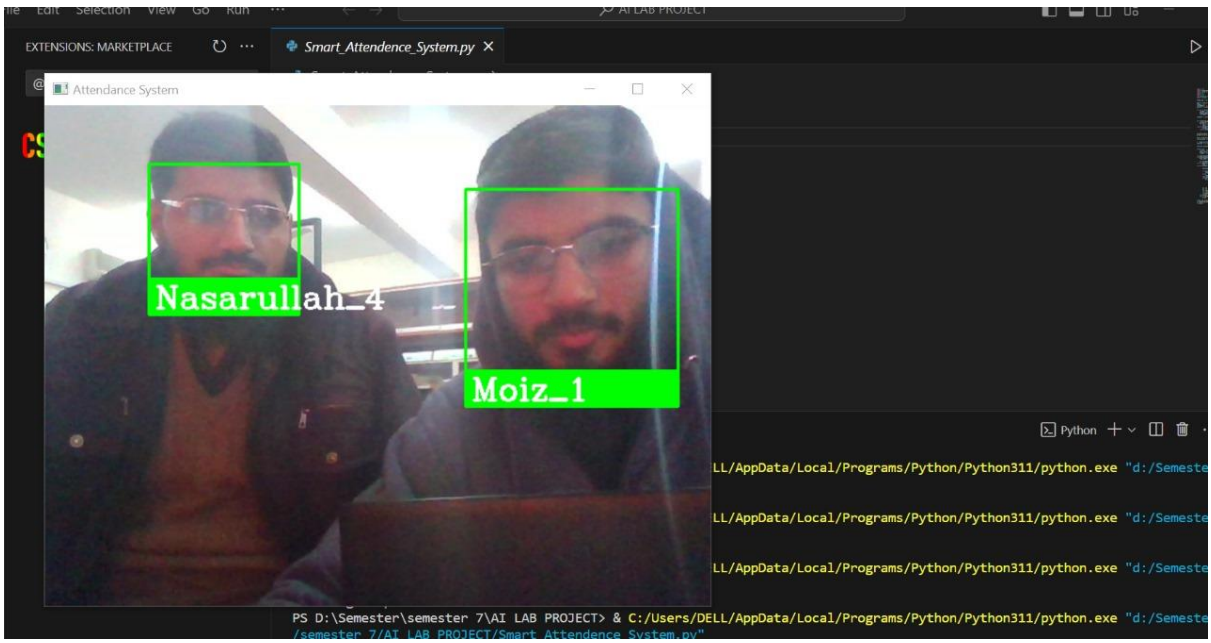
### 4.3 Phase 3

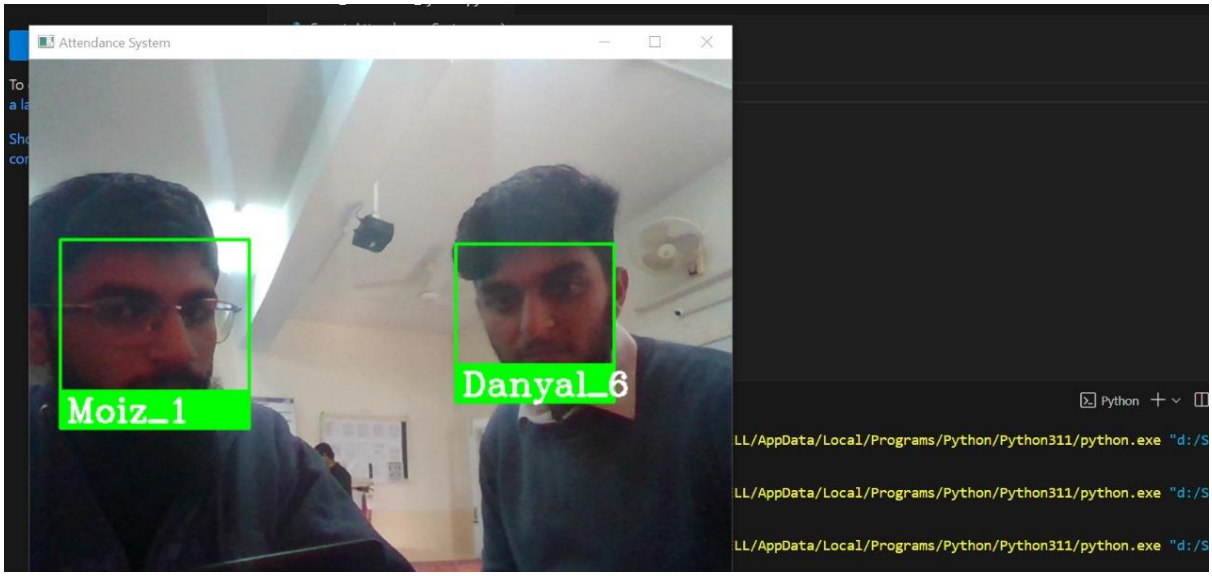
In the third and last phase, we assigned the label to the faces in the frame and created a CSV file to maintain the record of attendance in the database. We used the csv library of python. The file was named as the current date and it was write-enabled. After the student was recognized, a lock system was implemented. Here the system first confirms that the person is recognized correctly. After confirmation it assigned a label to the face and the attendance was marked in the excel sheet with the name and current time. In addition to that the system was also able to label an unknown person and maintain the record in the excel sheet. A snap of the unknown person was also saved.

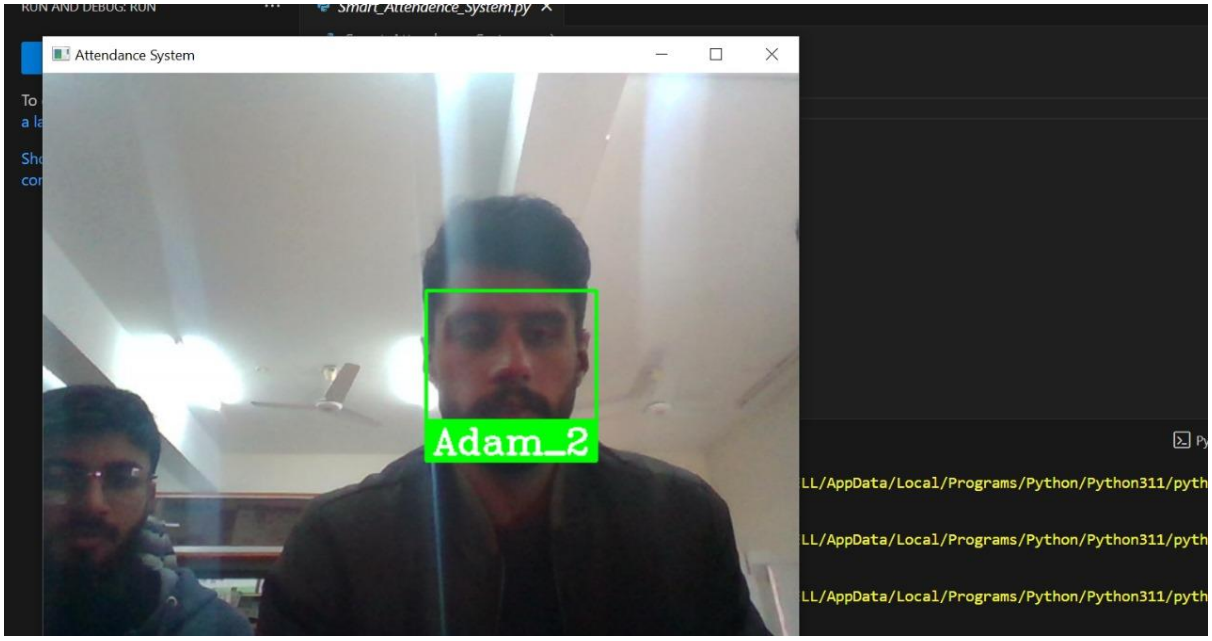
## 5. Conclusion:

The system aims in building an effective system that will be able to take autonomous attendance of students simultaneously. The system is built on the basis of facial recognition technique in which the system is trained with a data, which is basically the collection of images of the students. When the system is functional. It will start capturing frames from the webcam and whenever a face with familiar faces is detected by the system, the attendance is marked of the face that best matches with the dataset, along with the time of entry. Moreover, the system is also capable of storing the images of unknown persons in a separate folder, which means that the system can be deployed for security purposes also. The attendance and the record of unknown persons are stored in the database and can be accessed by the administration or authorized persons. This system has wide applications in schools and organizations and can be used as a modern tool for attendance and security purposes.

## 6. Snaps of Project:






7. **GitHub Links:**
   **https://github.com/moiz381/AI-LAB-PROJECT.git**

# The End