Automated Attendance System Using Facial Recognition

By:

Maaz Ansari J002

Riddhi Mehta J030

Sidh Satam J042

Index

Introduction

Face Recognition

Technology capable of identifying or verifying a person from a digital image or a video frame from a video source. There are multiple ways in which facial recognition occurs, but in general, it is done by comparing different facial features of the given image and from the images in the database. It is also called as biometric artificial intelligence. Face recognition has recently grown its importance, especially during the past several years as one of the most successful applications of image analysis.

Our Project and its Significance

In our current scenario, the faculty manually take attendance by calling out roll numbers after the lecture ends. This involves a lot of human interaction and hard work as the faculty have to enter the attendance in the excel file on their own.

On an average, every teacher takes 4-5 classes per day for different subjects in different batches. Every class has a strength of 60 students, hence that would mean the teacher has to manually mark attendance for 300 students per day.

This is a monotonous and a tiresome task which can be eliminated by the use of **Automated Attendance System Using Facial Recognition**.

Advantages

Compared to other biometric systems, it does not require the cooperation of the test subjects to be a part of the recognition. It is capable to do a mass identification and the properly designed systems installed at airports, government restricted areas, or public places can identify individuals among the crowd without the people being aware of the system

Disadvantages

The main problem with the current algorithms used in facial recognition is that, the recognition works almost perfectly well at full frontal faces and 20 degrees off. But, as soon as the face is a profile face, the problem arises and the facial recognition algorithm finds it difficult to classify it. This is one of the limitations faced in our current project. Other limitations are properly defined and discussed in the section of **limitations** further along the report.

Other Applications of Facial Recognition[1]:

Areas	Specific Applications
Security	Office access, building access control, airports, flight boarding system, email authentication on multimedia workstations, and trading terminals security.
Surveillance	Monitoring and searching for drug offenders and know criminals, power grid surveillance, CCTV control, and portal control.
Criminal Justice System	Post-event analysis and forensics.
Image Database Investigation	Searching image database of licensed drivers, benefit recipient, missing children, immigration checkpoints, national ID, welfare registration, and police bookings.
Human-Computer Interactions	Proactive computing and interactive gaming.
Smart Card Applications	In lieu of maintaining a database of face images, face prints can be stored in a smart card, bar code, or magnetic strip and authenticated by matching the live image with stored template.
Video Indexing	Labelling faces in video.
Civilian Applications	E-booking and E-commerce.
Multimedia Environments with Adaptive Human-Computer Interface	Part of ubiquitous or context-aware systems, behaviour monitoring at childcare or senior citizen centres, and recognizing a customer and assessing the customer's needs.

Table 1: Face Recognition Applications.

Literature Survey

For our project we got motivation by the research carried out by the following people and their published papers:

"Eigenfaces for recognition" [2], here they have developed a near-real time computer system that can locate and track a subject's head, and then recognize the person by comparing characteristics of the face to those of known individuals. The computational approach taken in this system is motivated by both physiology and information theory, as well as by the practical requirements of near-real time performance and accuracy. This approach treats the face recognition problem as an intrinsically two-dimensional recognition problem rather than requiring recovery of three-dimensional geometry, taking advantage of the fact that these faces are normally up right and thus may be described by a small set of two dimensional characteristic views. Their experiments show that the eigenface technique can be made to perform at very high accuracy, although with a substantial "unknown "rejection rate and thus potentially well suited to these applications. The future scope of this project was-in addition to recognizing face, to use eigenface analysis to determine the gender of the subject and to interpret facial expressions.

"Fast face recognition using Eigen faces" [3], their approach signifies face recognition as a two-dimensional problem. In this approach, face reorganisation is done by Principal Component Analysis (PCA). Face images are faced onto a space that encodes best difference among known face images. The face space is created by Eigen face methods which are eigenvectors of the set of faces, which may not link to general facial features such as eyes, nose, and lips. The Eigen face method uses the PCA for recognition of the images. The system performs by facing pre-extracted face image on to a set of face space that shows significant difference among known face images. Face will be categorized as known or unknown face after imitating it with the present database. From the obtained results, it was concluded that, for recognition, it is sufficient to take about 10% Eigen faces with the highest eigenvalues. It is also clear that the recognition rate increases with the number of training images.

"Face recognition using Eigen face approach" [4], this paper is a step towards developing a face recognition system which can recognize static images. It can be modified to work with dynamic images. In that case the dynamic images received from the camera can first be converted into the static ones and then the same procedure can be applied on them. The scheme is based on an information theory approach that decomposes face images into a small set of characteristic feature images called 'Eigenfaces', which are actually the principal components of the initial training set of face images. Recognition is performed by projecting a new image into the subspace spanned by the Eigenfaces ('face space') and then classifying the face by comparing its position in the face space with the positions of the known individuals. The Eigen face approach gives us efficient way to find this lower dimensional space. Eigenfaces are the Eigenvectors which are representative of each of the dimensions of this face space and they can be considered as various face features. Any face can be expressed as linear combinations of the singular vectors of the set of faces, and these singular vectors are eigenvectors of the covariance matrices. The Eigenface approach for Face Recognition process is fast and simple which works well under constrained environment. It is one of the best practical solutions for the problem of face recognition. Many applications which require face recognition do not require perfect identification but just low error rate. So instead of searching large database of faces, it is better to give small set of likely matches. By using Eigenface approach, this small set of likely matches for given images can be easily obtained.

"Face recognition using eigenfaces and artificial neural networks" [5], this paper presents a methodology for face recognition based on information theory approach of coding and decoding the face image. Proposed methodology is connection of two stages:

Feature extraction using principle component analysis and recognition using the feed forward backpropagation Neural Network. The algorithm has been tested on 400 images (40 classes). A recognition score for test lot is calculated by considering almost all the variants of feature extraction. The proposed methods were tested on Olivetti and Oracle Research Laboratory (ORL) face database. Test results gave a recognition rate of 97.018%

“Performance Showdown of Publicly Available Face Detection Model” [6], here, they conducted an experiment on 5 publicly available face detection models which we can try on your own in This GitHub. Additionally, they also provided an example of how to use the face detection module on your own project. They are:

1. OpenCV Haar Cascades Classifier
2. DLib Histogram of Oriented Gradients (HOG)
3. DLib Convolutional Neural Network (CNN)
4. Multi-task Cascaded CNN (MTCNN) — Tensorflow
5. Mobilenet-SSD Face Detector — Tensorflow

The device used to benchmark these models is Dell Inspiron 15 7577 with hardware specification:

- CPU = Intel Core i7–7700HQ Quad Core Processor
- GPU = Nvidia GeForce GTX 1060
- RAM = 16GB

The results concluded that, both on the setup when using the best hardware configuration or using only CPU, Mobile Net SSD gives the outstanding performance not only in term of accuracy (the higher the better) but also in term of inference time speed (the lower the better). This shows that from these publicly available model, using Mobile Net SSD implementation may provide you with the best performance when you want to deploy it for the real-time system.

In the end, it’s up to us which face detector we want to use in your own project.

Problem Statement

Developing a complete facial recognition based attendance system (that can be used as a prototype to sell in the market) that eliminates the human interaction between the professors and the students, and notifies the students their attendance as well as sends the excel file with the attendance of all students for that lecture to that faculty.

Current Scenario

The faculty manually take attendance by calling out roll numbers after the lecture ends. This involves a lot of human interaction and hard work as the faculty have to enter the attendance in the excel file on their own.

On an average, every teacher takes 4-5 classes per day for different subjects in different batches. Every class has a strength of 60 students, hence that would mean the teacher has to manually mark attendance for 300 students per day.

This is a monotonous and a tiresome task which can be eliminated by the use of Automated Attendance System Using Facial Recognition.

Future Scenario

Using this system, we aim to automatically mail the students their attendance as well as mail the excel file with the attendance of all the students for that lecture to the faculty.

This system ensures minimal human intervention, less time consumption, proxy elimination etc.

Objectives

Creating the Dataset

It was the first phase of the project. Collecting the pictures was an important step as we would use them to train our model. A lot of brain storming was required to decide the number of pictures required, the angle at which the pictures should be taken, the situation (light effect) was crucial, as well as the quality of the pictures mattered. Moreover, we had to make sure that all these pictures do not occupy a large storage space which would lead to a heavy load on the code to read them.

After discussion, we concluded that we needed a total of 20 pictures per person which were evenly distributed amongst full frontal face, 20 degrees off, 40 degrees off and top view. A few were even taken at profile face but, as discussed in literature review, we knew that we will encounter recognition problems if we use them. We have taken all the pictures in natural light, because we were going to augment them in the later phases of the project anyway.

Algorithm and Design

The project has been divided into various components which use different algorithm to achieve results. The results of one algorithm in each phase act as an input for another algorithm in the next phase. The components and the algorithms are sequentially mentioned as follows whose details are further given in methodology:

1. **Phase 1 – Creating Training Model**
    a. Detection of the face in the given photo – FaceNet algorithm
    b. Data Augmentation on the detected faces – Salt and Pepper, Gaussian Filter, Rotations and Flips, Lighten and Darken algorithms
    c. Generate embedding – MTCNN algorithm
    d. Training the model using classifier – MLP Classifier algorithm

2. **Phase 2 – Real Time Face Detection**
    a. Interface
    b. Initialise
    c. Detection
    d. Tracking
    e. Classification
    f. Attendance (Final Step)

Graphical User Interface

It is essential to create a simple graphical user interface for professors who can select the programme, stream, year, and the duration of the class. Once the selection is done, the results are chained to initialise the attendance system for that particular class.

Methodology

Block Diagram

The below image depicts the entire project block diagram right from collection of raw data, pre-processing and using the augmented data to train the model. It is followed by real time running of the project and the final step of emailing the professors and students with the attendance.

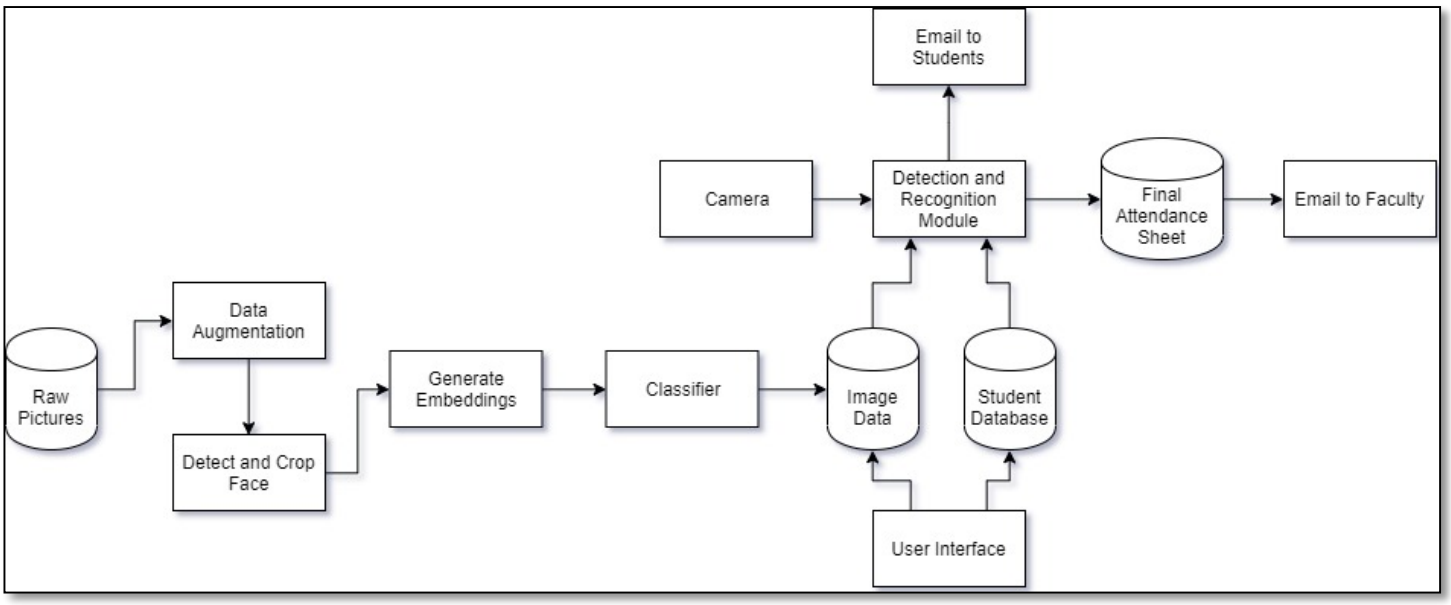


Figure 1: Methodology – Block Diagram

Explanation

The architecture of the project is such that it has been divided into specifically two phases, where the first phase does the pre-processing and training whereas the second phases focuses on the real time detection and tracking.

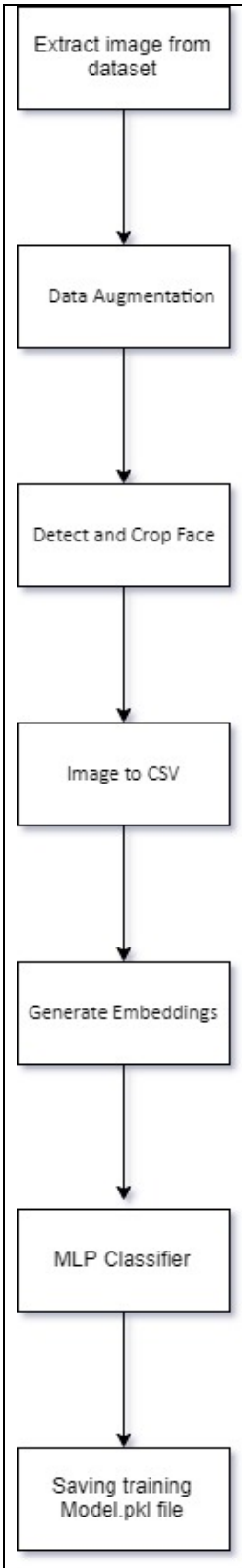
The **raw pictures** are the collected pictures of the test subjects (students) which are pre-processed using **detect, crop and data augmentation** algorithms. The augmented pictures are converted from image to csv in pixel format, which are in turn used to **generate embedding**.

These embedding are an input to the **classifier** which will learn, classify and map specific and unique face features to a particular person.

The **user interface** is where the professor enters the details of the class, and based on the details, the **student database** is retrieved. The process is now activated which is followed by **activation of the camera**; and the **image data of** the ongoing class is continuously compared with the student database by fast tracking of MobileNetSSD.

The **detection and recognition module** runs and marks the attendance of the student who have been captured by the camera for more than 60%. This **notification** is sent to the **student** by the end of the lecture and the **final attendance sheet** is sent to the **professor** of that class, both in the **form of an email**.

Algorithm and Flow Chart – Phase 1



Extract Image from Dataset

The image is extracted from the raw pictures' dataset.

Data Augmentation

Image is augmented by performing Data Augmentation Techniques which would help in recognition of faces at different angles or in different lighting conditions. We would be changing the orientation of the image by performing Rotations and flips and adding noise to the image as Salt and Pepper Noise or adding a Gaussian Filter to Lighten or Darken the image.

Detect and Crop Face

Face is detected in it using FaceNet. FaceNet develops a deep convolutional network to learn a mapping from face images to a compact Euclidean space where distances directly correspond to a measure of face similarity. It uses triplet-loss as its loss function. The image is cropped to a bounding box surrounding the face.

Image to CSV

These cropped faces are now converted from the image matrix to a one dimensional array of pixels by reshaping it to save space and make the code efficient. Reading of full images take a lot of time and memory, as compared to reading a simple CSV which has each image in one row.

Generate Embedding

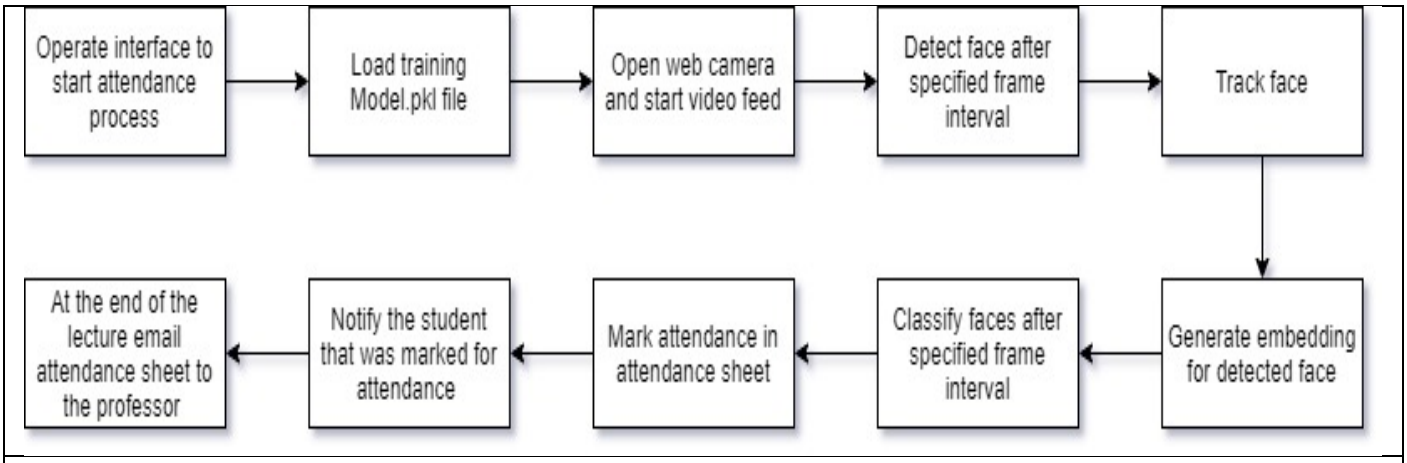
Using transfer learning, we will create embeddings for the new dataset using FaceNet. FaceNet aligns the faces and generates MTCNN (Multi-Task Cascaded Convolutional Neural Networks) embeddings for each face.

MLP Classifier and save training as Model.pkl file

Using MLP, we create a classifier by tweaking the hyper parameters for best accuracy and then save the model to a pickle file.

Table 2: Phase 1 of the Program

Algorithm and Flow Chart – Phase 2



Interface and Loading of Model.pkl file

There is no specific algorithm used in making or operating the interface. The basic logic is that, the professor selects one programme, and then the streams of that programme are filtered accordingly. The professor is then asked to select the duration of the class which can be 1 hour, 2 hours, 3 hours etc. Once these options are selected, the values are passed in such a way to select the model.pkl file of that programme, stream and year.

Camera and Video Feed

This is the initialisation step where the camera is activated to capture the live video feed. The model.pkl file has the details of the embedding of the students which is saved once the professor enters the details of the class.

Detect, Track and Generate Embedding for the face

Once the camera is activated, the faces in the frame are detected, cropped for the face, and its embedding is generated to compare it with the embedding saved in the student database. If there is a match found, then the face is tracked for some time to make sure that the match is right, otherwise the face is tagged as unknown.

Classify Face

Once the face has been tracked for a significant time over the entire period of the lecture, the person has now been classified as present. Every person has a unique SAP Id through which he or she is mapped in the database. Once the image is matched with the person, the SAP Id is marked as 1 next to it which signifies that he or she has been marked as present. Till the class is over, this process goes on and accordingly we have a proportion of the SAP Id's marked as 1 and others are left as unmarked.

Final Step

The final step is to send a notification to those students who are marked as 1 that their attendance is registered for the day for that particular lecture. The database also contains the emails of these students. The emailing part of the code looks up the email of those students who have been marked as 1, and email is sent to them. An excel sheet is automatically generated with all the names of the students in the class as one column, and other columns being – date, lecture timings, duration of the class, attendance. This excel file which was first manually made by the professor is now automatic and this is mailed to the professor whose email details have been registered in the start of the entire process.

Table 3: Phase 2 of the Program

Implementation and Results

Since the project is divided into two different phases, we have shown the implementation in two different parts. The first part focuses on the steps till training the model file, and the second part is the real time face recognition. The implementation is shown as follows:

Data Augmentation

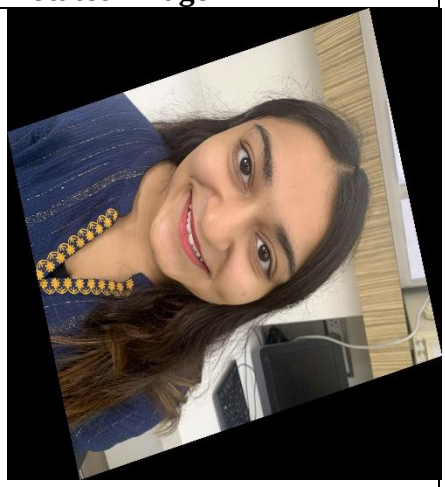
Original Image	Flipped Image	Rotated Image
		
Salt and Pepper Image	**Lighten Image**	**Darken Image**
		

Table 4: Data Augmentation Examples

Detect and Crop Face

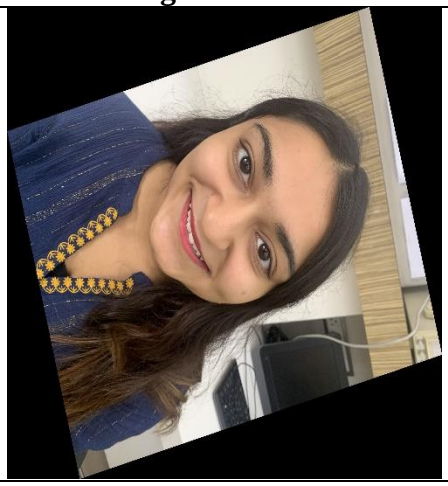
Augmented Image	Cropped Image (after Detection)
	

Table 5: Detect and Crop Face

Graphical User Interface

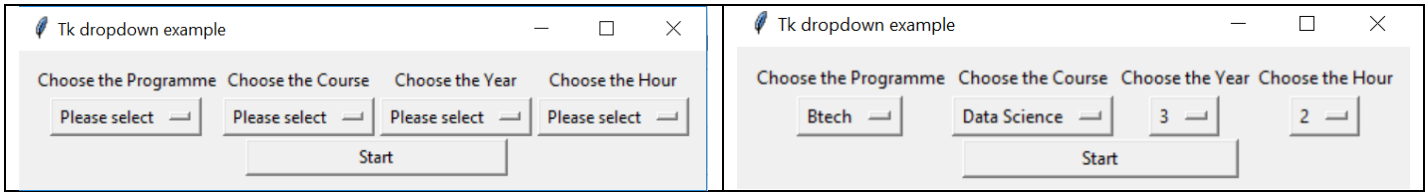


Table 6: Graphical User Interface

Recognition of Faces

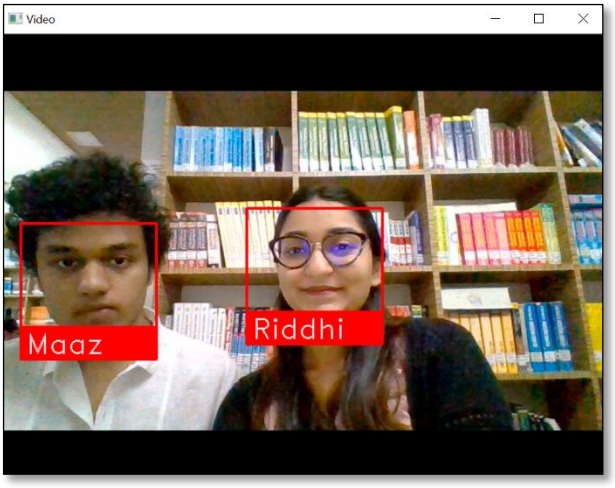


Figure 2: Recognition of Faces

Mark Attendance

	Name	Email	15-10-19
0	Riddhi	riddhi.mehta1721@gmail.com	
1	Samarth	sarin.samarth07@gmail.com	
2	Maaz	maazsansari846@gmail.com	
3	Sidh	sidhsatam@gmail.com	
4	Bhavesh	mbhavesh21@gmail.com	

	Name	Email	15-10-19
0	Riddhi	riddhi.mehta1721@gmail.com	3
1	Samarth	sarin.samarth07@gmail.com	0
2	Maaz	maazsansari846@gmail.com	3
3	Sidh	sidhsatam@gmail.com	0
4	Bhavesh	mbhavesh21@gmail.com	0

Figure 3: Marking of Attendance in Database

Email to Student

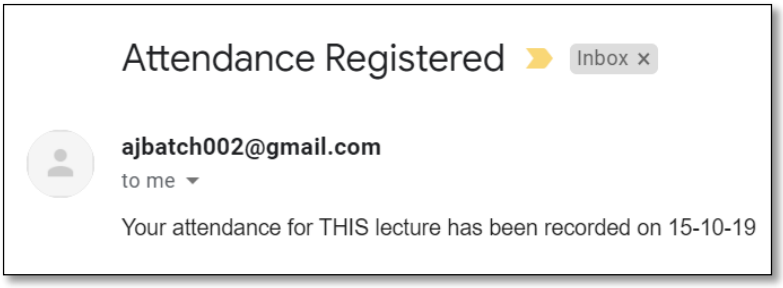


Figure 4: Email to Student for particular lecture

Email to Professor

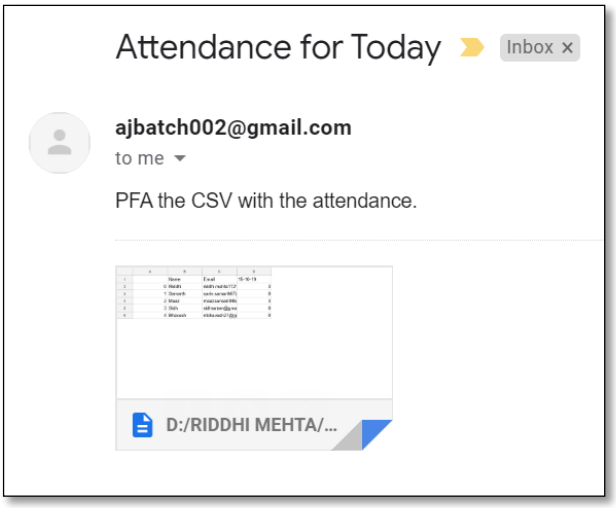


Figure 5: Email to Professor with Final Attendance

Accuracy

Since, the program runs in real time, we have computed the accuracy in different light situations, different backgrounds, and with different permutations and combinations of test subjects. We have used two different metrics to computer accuracy. First is **Validation Loss** and second is the **Confusion Matrix**.

As shown in Figure 2: Recognition of Faces, we ran the program several times with different test subjects to calculate the accuracy.

Our accuracy was 86% when we carried out the testing phase of our program.

The validation loss of the model was 0.02.

Conclusion, Limitations and Future Recommendations

Conclusion and Limitations

We believe that working on this project has helped us a lot in building our domain knowledge conceptually, technical (hardware and software) skills as well as deriving business insights from the same. An artificial intelligence project of this level has not only given us an opportunity to learn and implement different algorithms, but has also given us a chance to convert the same as an application used in businesses.

Facial recognition based attendance system, when installed in various colleges all over Mumbai attempts to solve a problem being faced by our professors since years now. There is a lot of scope in this project and it shows a high scalability because it can used in a simple classroom for taking attendance, or it can be scaled to be used in the entire college to keep a track of unknown, or unregistered teachers or students. Although, with the usage of a technology like this, we have faced many questions about privacy violations. Many express concern that privacy is being compromised by the use of surveillance technologies and that it is now easier to keep a track a people and their whereabouts. **Since, it was a concern of major intensity, our next goal is integrating security aspects to our project and make sure that the database and personal information are not revealed to the outside world, and are encrypted in a way that only the core members of the project can access or edit the changes**.

In conclusion, we believe that we have successfully implemented the project, by incorporating accurate facial recognition algorithms, faster face tracking, notifications to the students, attendance sheet automatically emailed to the professor and an easy graphical user interface to be used by the professors to initiate the entire procedure of **Facial Recognition based Attendance System**. We have also successfully identified the challenges that we have faced in this project, and are planning to take it as our next goal or phase of the project.

Future Recommendations

Every project has a scope for improvement and with the growing technology comes better, faster and accurate algorithms that nullifies the use of previous algorithms. Since, this is a project which needs to be constantly updated because we cannot compromise with the security as well as the accuracy of all the test subjects involved, so the recommendation that we have received during the course of our project is to keep working on it, and make sure that every aspect of the project has been given proper attention.

It is very important to always keep the pace and flow of the progress of the project. We have decided (as mentioned in the previous sections of the report) that our next goal is to integrate **security** and **encryption** to protect sensitive data to the outside world. Apart from the things that can be done to improve the project and the challenges that can be solved, it is equally **essential to discuss the future prospects** of this project.

Facial recognition has a very unique feature unlike other biometric systems – capability of mass identification. Schools and colleges are often targeted by people of bad faith to harass, or bully students, of which many times can lead to unfortunate events. This project can also be scaled to enhance the security system of these schools or colleges. People studying at these places as well as trusted visitors should be allowed to enter, but an instant notification needs to be sent to the security when there is a trespasser. In this way, people are more aware of their surroundings and can feel relieved to know that there is someone watching out for strangers and people with the objective of causing harm.

References

[1] “Improve Face Recognition Rate Using Different Image Pre-Processing Techniques” (Nawaf Hazim Barnouti)

[2] “Eigenfaces for recognition’’ (Mathew Turk and Alex Pentland)

[3] “Fast face recognition using eigenfaces” (Arun Vyas and Rajbala Tokas)

[4] “Face recognition using eigenface approach” (Vinay Hiremath and Ashwini Mayakar)

[5] “Face recognition using eigenfaces and artificial neural networks” (Mayank Agarwal, Nikunj Jain, Mr. Manish Kumar and Himanshu Agrawal)

[6] “Performance Showdown of Publicly Available Face Detection Model” (Alvin Prayuda Juniarta Dwiyantoro), April 30, 2018**.**