A Seminar Report

Automated Attendance System Using Facial Recognition

Submitted in partial fulfillment of the requirements for the award of degree

BACHELOR OF ENGINEERING

In

COMPUTER SCIENCE AND ENGINEERING

By

KOMALI BEERAM (160118733067)

Department of Computer Science and Engineering,

Chaitanya Bharathi Institute of Technology (Autonomous),

(Affiliated to Osmania University, Hyderabad)
Hyderabad, TELANGANA (INDIA) –500 075
[2021-2022]

CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY (A)
Kokapet (Village), Gandipet, Hyderabad, Telangana-500075. www.cbit.ac.in

COMMITTED TO RESEARCH, INNOVATION AND EDUCATION 43 years

CERTIFICATE

Certified that seminar work entitled **"*Automated Attendance System Using Facial Recognition*"** is the bonafide work carried out by, **Komali Beeram (160118733067)**, in partial fulfillment for the award of Bachelor of Engineering in Computer Science and Engineering from **Chaitanya Bharathi Institute of Technology(A)**, Gandipet during the academic year **2021-2022**.

SIGNATURE
Dr. Sangeeta Gupta

MENTOR
Smt. I Srujana

SIGNATURE
Smt. K. Mary Sudha

SIGNATURE
Smt. G Kavita

DECLARATION

I, here by, declare that the project report entitled "Automated Attendance System Using Facial Recognition" submitted to Department of Computer Science and Engineering, Chaitanya Bharathi Institute of Technology, in partial fulfillment of the requirements for the B.E VIII Semester course titled Technical Seminar, is a bonafide record of work carried out by Komali Beeram (1601 18 733 067) during the academic year 2021-2022 under the guidance and supervision of Smt. I Srujana. I further declare that the work reported has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Signature of the Student
Komali Beeram

ABSTRACT

Daily attendance marking is a common and important activity in schools and colleges for checking the performance of students. Manual attendance maintenance is a difficult process, especially for a large group of students. Some automated systems which were developed to overcome these difficulties, have drawbacks like cost, fake attendance, accuracy, and intrusiveness. Traditional face recognition systems employ methods to identify a face from the given input, but the results are not usually accurate and precise as desired and are time-consuming. To overcome these drawbacks, there is a need for a smart and automated attendance system.

The aim is to deviate from such traditional systems and introduce a new approach to identify a student using a face recognition system by Multi-Task Cascaded Convolutional Neural Networks, the generation of a facial mode which can further be deployed as an Automated Attendance System in a classroom environment which will be more accurate and bring out faster results when compared to the other competitive machine learning models

List of Tables

Name	Page No.
Table 3.1 Data Dictionary	**16**

List of Figures

Name **Page No.**

Fig 2.1 Attendance System using QR Code 10
Fig 2.2 Attendance Monitoring System 12
Fig 3.1 Proposed Model Flowchart 15
Fig 3.2 Data Flow Diagram Context Level 17
Fig 3.3 Data Flow Diagram Level 1 18
Fig 3.4 Implementation categories 19
Fig 3.5 Haar Cascade 20
Fig 3.6 DLIB(CNN) 20
Fig 3.7 DLIB(HOG) 20
Fig 3.8 MTCNN 20
Fig 3.9 Face Detection Code 21
Fig 3.10 128 Face Embeddings 22
Fig 3.11 Feature Extraction using FaceNet Keras Code 23
Fig 3.12 Feature Classification Code 24
Fig 3.13 Attendance Marking Code 24
Fig 4.1 Plot of 15 photos of a student detected from the training dataset 25
Fig 4.2 Face Embeddings 25
Fig 4.3 Accuracy of the model 26
Fig 4.4 Detected and correctly identified by Model 26
Fig 4.5 Faces detected and wrongly identified by model 26
Fig 4.6 Attendance Marked in the sheet 26

Table of Contents

	Title Page	i
	Certificate of the Guide	ii
	Declaration of the Student	iii
	Abstract	iv
	List of Tables	v
	List of Figures	vi
1.	**INTRODUCTION**	**1**
	1.1 Overview	1
	1.2 Motivation	2
2.	**LITERATURE REVIEW**	**3**
	2.1 Identified Gaps	14
3	**DESIGN OF PROPOSED SYSTEM**	**15**
	3.1 Proposed Methodology	15
	3.2 System Design	16
	3.2.1 Data Dictionary	
	3.2.2 Data Flow Diagram	
	3.3 Implementation	19
	3.3.1 Database Creation	
	3.3.2 Face Detection	
	3.3.3 Face Extraction	
	3.3.4 Face Classification	
	3.3.5 Attendance Marking	
4	**RESULTS**	**25**
5	**CONCLUSION and FUTURE WORK**	**27**
	REFERENCES	**28**

1. INTRODUCTION

1.1 Overview

Attendance maintenance is a significant function in all the institutions to monitor the performance of students. Every institute does this in its own way. Some of the institutes use the old paper or file-based systems and some have adopted strategies of automatic attendance using some biometric techniques.

In this modern era of automation many scientific advancements and inventions have taken place to save labor, increase the accuracy and to ameliorate our lives. Automated Attendance System is the advancement that has taken place in the field of automation replacing traditional attendance marking activity. Automated Attendance Systems are generally bio-metric based, smart-card based, and web based. These systems are widely used in different organizations. Traditional method of attendance marking is very time consuming and becomes complicated when the strength is more. Automation of Attendance System has an edge over traditional methods as it saves time and can be used for security purposes. This also helps to prevent fake attendance.

A facial recognition system is a biometric software which is suited for determining or validating a person by performing comparison on patterns based on their facial appearances. Face recognition systems have upgraded appreciably in their management over the recent years and this technology is now vastly used for various objectives like security and in commercial operations. Face recognition is a powerful field of research which is a computer based digital technology. Face recognition for the intent of marking attendance is a resourceful application of the attendance system. It is widely used in security systems and it can be compared with other biometrics such as fingerprint or eye iris recognition systems. As the number of students in an educational institute or employees at an organization increases, the needs for lecturers or to the organization also increase the complication of attendance control. This work may be helpful for the explanation of these types of problems. The number of students present in a lecture hall is observed, each person is identified and then the information about the number of students who are present is maintained.

1.2 Motivation

According to the previous attendance management system, the accuracy of the data collected is the biggest issue. This is because the attendance might not be recorded personally by the original person, in another word, the attendance of a particular person can be taken by a third party without the realization of the institution which violates the accuracy of the data. For example, student A is too lazy to attend a particular class, so student B helped him/her to sign for the attendance which in fact student A did not attend the class, but the system overlooked this matter due to no enforcement practice. Supposing the institution establishes enforcement, it might need to waste a lot of human resource and time which in turn will not be practical at all. Thus, all the recorded attendance in the previous system is not reliable for analysis usage. The second problem of the previous system is where it is too time consuming. Assuming the time taken for a student to sign his/her attendance on a 3-4 paged name list is approximately 1 minute. In 1 hour, only approximately 60 students can sign their attendance which is obviously inefficient and time consuming

Thus, a face recognition student attendance system is proposed to replace the manual signing of the presence of students which is burdensome and causes students to get distracted in order to sign for their attendance. Furthermore, the face recognition based automated student attendance system is able to overcome the problem of fraudulent approach and lecturers do not have to count the number of students several times to ensure the presence of the students. Many attendance management systems that already exist lack efficiency and information sharing. Therefore, in this current proposed system, those limitations are overcome and also further improved and made much more efficient.

2. LITERATURE REVIEW

This section documents the available relevant literature concerning the problem domain. The implication was that we devoted sufficient time to reviewing research already taken on related problems. This was done to find out what data and other materials are already available from earlier research and identify gaps that the present research may fill.

a. Smart Attendance System using QR code[1]

The proposed system here, generated a smart attendance system which uses Quick Response (QR) code to track and record the attendance. Students and professors are given a unique QR code, at the beginning of the course, they are required to scan their QR code using a QR reading device. Attendance of students whose QR code is scanned will be recorded. This system is responsive to mobile phones and different computer systems.

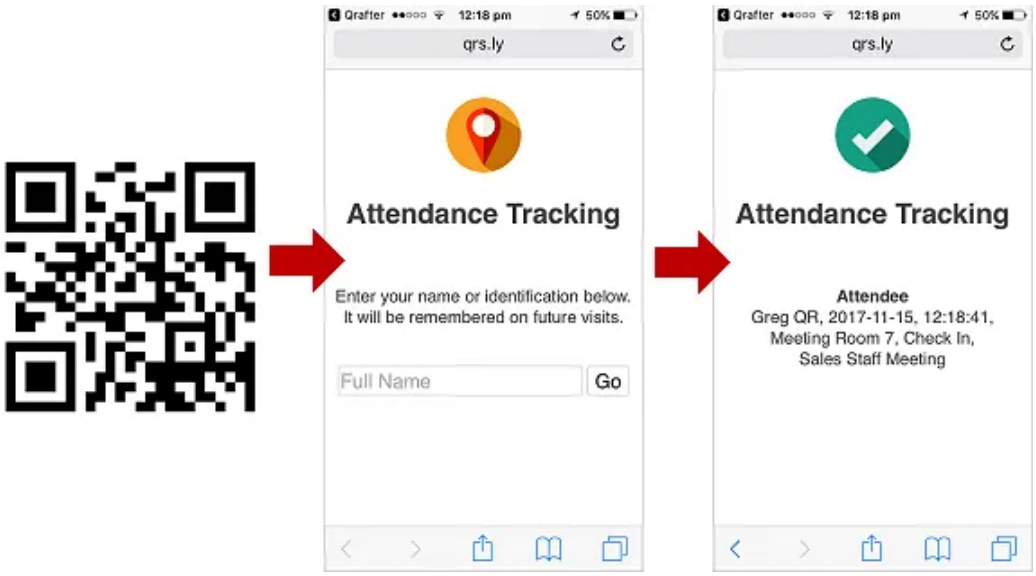

Fig 2.1 Attendance System using QR Code

b. Face Recognition Attendance System Based on Real-Time Video Processing[2]

Here they considered accuracy rate, stability of system in actual time video processing, truancy of system and interface setting of the face recognition system. Face detection and recognition are two main parts of face recognition. Feature extraction is done by the LDA (Linear Discriminant Analysis) method. This model takes help of methods such as geometric Feature method, Subspace analysis method, Neural Network methods, Support Vector Machine (SVM) method to develop their

face recognition algorithm. Experimentally this model of video face recognition system gives an accuracy rate up to 82%.

c. Face Recognition System Based on LBPH Algorithm[3]

Abhishek Pratap Singh et al. proposed a face recognition system using Local Binary Histogram algorithm and Haar Cascade classifier. Local Binary Pattern (LBP) is a simple yet very efficient texture operator which labels the pixels of an image by thresholding the neighborhood of each pixel and considers the result as a binary number. It was used for identifying a face and also it recognized both front and side faces irrespective of picture quality, with better recognition rate in real time. The system was able to recognize a known and unknown person. When the rate of change in the frame was very high, occlusions occurred and the proposed method was not able to robustly recognize the faces.

d. Automatic Attendance System using Face Recognition[4]

In this work, facial recognition is applied into an attendance checking system that uses faces of registered people to check their attendance. This system has a GUI, which allows user-to-system interaction and attendance marking will be easy through recognizing the face of the student with the help of the recognition algorithm and mark the attendance. Cascade classifier is used to detect faces. The Local Binary Pattern Histogram algorithms for this technology using face recognition, to monitor students and they can verify their attendance status with the help of the Register Number.

e. Automatic Face Detection and Recognition for Attendance Maintenance[5]

This paper focuses on building a deep learning based attendance capturing system. They developed a system architectural solution using YOLO(You Look Only Once) embeddings by applying multiple augmentations, picture quality check and denoise methods to get a better attendance system. Their proposed solution uses a Deep Learning approach that involves reading data from live-video from a camera installed in the classroom, converting them to necessary frames, perform face detection using YOLOv3 on these frames and apply face recognition with Linear Support Vector machine classifier(SVC) on detected faces and mark the attendance. During testing, their system was detecting non-facial images and face reflections as actual faces.

f. Face Recognition for Attendance System Detection[6]

Rudy and Marcus developed a face recognition system that consists of four stage processes. The processes were face detection process using Haar Cascade Algorithm and skin color detection where images were converted from RGB format to YCrCb format, alignment process that applied face features normalization, feature extraction process, and classification process using LBPH algorithm. The face recognition accuracy was 98.2% at a face distance 40 cm from the camera with lighting condition of 24 lux and for lighting condition of 7 lux accuracy was 94.7%. Testing was performed for camera distance between 40cm-90cm, accuracy decreased for farther distance face recognition.

g. Automatic Attendance System using Deep Learning[7]

Sunil Aryal et al. proposed an attendance system with a combination of facial recognition algorithm and machine learning algorithm. Single Shot Multibox Detect (SSD) was used for face detection from an image capturing in real time and recognizing the detected face using pre-trained FaceNet Model, which are optimized based upon triplet loss. From the experiment analysis, the accuracy obtained of the proposed system was 97%. The approach solved the problems of face recognition but cannot identify each and every student present in a class.

h. Attendance Monitoring System using Face Detection & Face Recognition[8]

An automatic attendance management system was proposed using face recognition algorithms. A camera at the doorway captures a student's image while entering the class. But, that system faced limitations as it could not define two persons at the same time.

Fig 2.2 Attendance Monitoring System

i. Real-Time Smart Attendance System using Face Recognition Techniques[9]

In this proposed solution, two databases (face database and attendance database) are used. During enrolment, facial images of students are stored into the face database. The camera captures the images of the classroom, the images get enhanced and the attendance is marked in the attendance database after face detection and recognition. AdaBoost algorithm and Principal Component Analysis (PCA) are used for face detection and face recognition respectively.

j. IAAS: IoT-Based Automatic Attendance System with Photo Face Recognition in Smart Campus[10]

This paper proposes an IoT-based Automatic Attendance System (IAAS) which is an attendance checking system using a face recognition technology. The image data of students are collected by a capturing device, for example a smartphone and tablet PC, and then processed by the face recognition system. This system checks who are attending the class and sends their attendance to an attendance database system through an email notification. For face detection, MTCNN with ESRGAN is used, ESRGAN is a technique to increase the image resolution along with Haar-Cascade to extract user's face data. To construct a notification service, an email protocol that is Simple Mail Transfer Protocol is used. Message Queuing Telemetry Transport(MQTT) for IoT Message exchange is used.

k. Going deeper with convolutions[11]

A GoogleNet technique is used to make a Face Verification model. Szegedy et al. proposed a GoogleNet, which is a deep convolutional neural network architecture named Inception. The main advantage of this method is a significant quality gain at a modest increase in computational requirements when its neural network is compared to shallower and less wide networks.

In addition, the data are trained using the vggface2 data set. However, vggface2 provides a huge amount of data, but because it is a picture of Westerners, it has the disadvantage of poor learning efficiency in Asians.

2.1 Identified Gaps

- Lack of efficiency
- Lack of robust and error free systems
- Lack of efficiency and accuracy of previous existing algorithms
- Proxies
- Cost
- Fail to recognize faces when illuminated, rotated, pose, occlusions etc.
- Detection process is slow, and computation is complex in terms of neural networks
- Long training time
- Large database is required to achieve high accuracy

3. DESIGN OF THE PROPOSED SYSTEM

3.1 Proposed Methodology

The proposed system will provide login credentials to faculty who are in the university with which they can interact with the system. The student details and face images will be stored in the database.

The proposed system will firstly capture a picture of a face and discern all the faces in it. It then concentrates on one face at a time and understands that even if the face is turned in a strange direction or in bad lighting, it is still the same person. The shadows on the face due to low lighting can affect the image but it wont change the person. The next step is to determine various features of the face that can help in distinguishing it from another person. These characteristics could be size of eyes, nose, length of face, skin color etc. Comparing these distinctive features of that face to all the faces of people we have in the database; we will find out the person's name. Our brain, as a human, is made to do all these things automatically and instantaneously. Computers are incapable to generalize this kind of high-level techniques, so we need to teach the computer by programming each step of facial recognition separately. Face recognition has three categories: detection, verification, and identification. Face verification is a one-to-one match that compares a face image against a template face image, whose identity is being claimed. On the contrary, face identification is a one-to-many problem that compares a query face image.

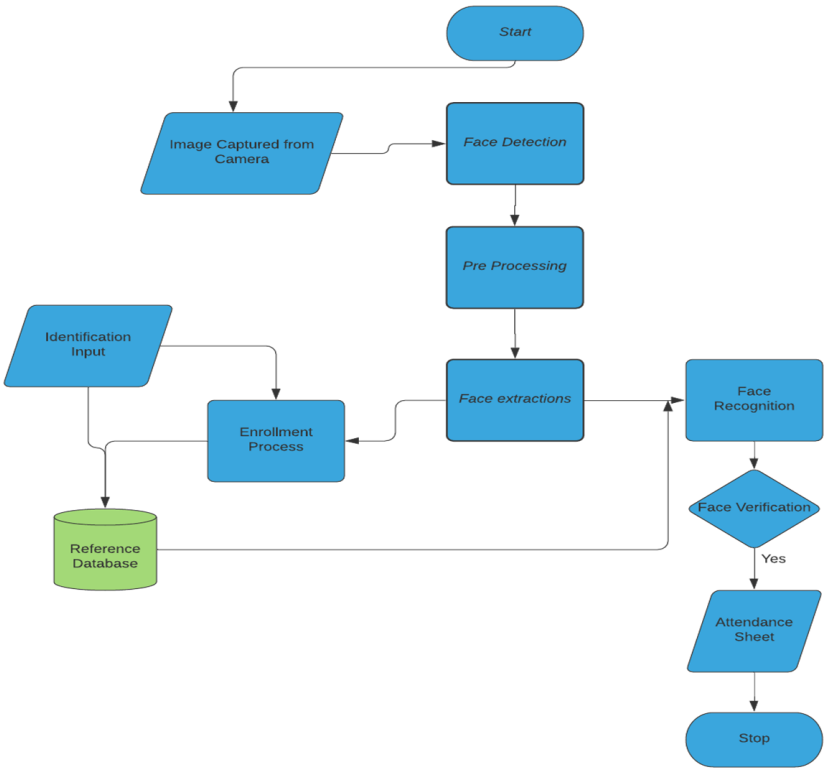

Fig 3.1 Proposed Model Flowchart

3.2 System Design

Design documents, personas, user case scenarios etc. to run comprehensive tests including components testing and on the finished applications.

3.2.1 Data Dictionary

A data dictionary contains metadata i.e., data about the database. The data dictionary is very important as it contains information such as what is in the database, who can access it, where is the database physically stored, when and where database is created, database table constraints such as primary key attributes, foreign key information etc. The users of the database normally do not interact with the data dictionary, it is only handled by the database administrators.

FIELD NAME	DATA TYPE	NULL	CONSTRAINT	DESCRIPTION
student_id	int(auto_increment)	NO	PRIMARY KEY	Unique ID of student.
student_name	varchar(50)	NO	LENGTH>0	Full name of the student.
student_email	varchar(50)	NO	UNIQUE	Email address of the student.
student_photo	image	NO	UNIQUE	A photograph of student face.
Date	date	NO	LENGTH>0	Date of the attendance.
Attendance	varchar(10)	NO	Present or Absent	Attendance of the student.
faculty_id	int(auto_increment)	NO	PRIMARY KEY	Unique ID of faculty.
faculty_name	varchar(50)	NO	LENGTH>0	Full name of the faculty.

Table 3.1 Data Dictionary

3.2.2 Data Flow Diagram

It is the graphical representation of the flow of data between various processes through an information system. It differs from flowchart as it shows the data flow instead of control of the program. It can also be used for visualization of data programming.

DFDs are drawn at different levels. Level 0 shows the main process in the system. Further levels describe the subprocesses and the development in the system. All the processes must have at least 1 data flow in and out; should modify the incoming data producing new forms of outgoing data. Each data stored must be involved with at least one data flow. Each external entity must be involved with at least one data flow. The data flow must be attached to at least one process.

Level 0

This is the fundamental system model or context diagram that represents the entire software requirement as a single bubble with input and output data denoted by incoming and outgoing arrows. Then the system is decomposed and described as a DFD with multiple bubbles. Parts of the system represented by each of these bubbles are then decomposed and documented as more and more detailed DFDs. The below diagram depicts how students, faculty can access the system and how a student can give attendance through a camera and both faculty and students can view the attendance report.

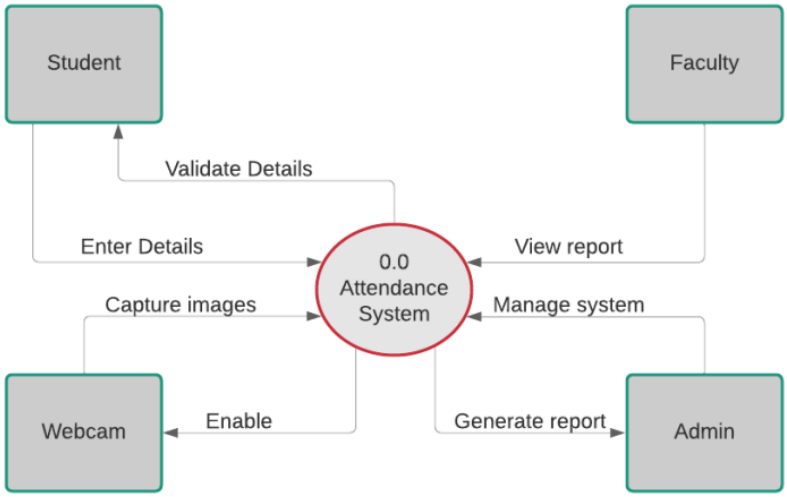

Fig 3.2 Data Flow Diagram Context Level

Level 1

In Level 1 DFD, a context diagram is decomposed into multiple bubbles/processes. In this level, the main objectives of the system are broken down to the high -level process of Level 0 DFD into subprocesses. It also projects or records the specific/necessary detail about the system's functioning. The below DFD shows how a student logs into the student, and how the web camera is enabled, and the student pictures are captured and then the image acquisition, detection, pre-processing and recognition is done with the help of the student database. The diagram also illustrates the activities or functions, or modules managed by faculty and admin.

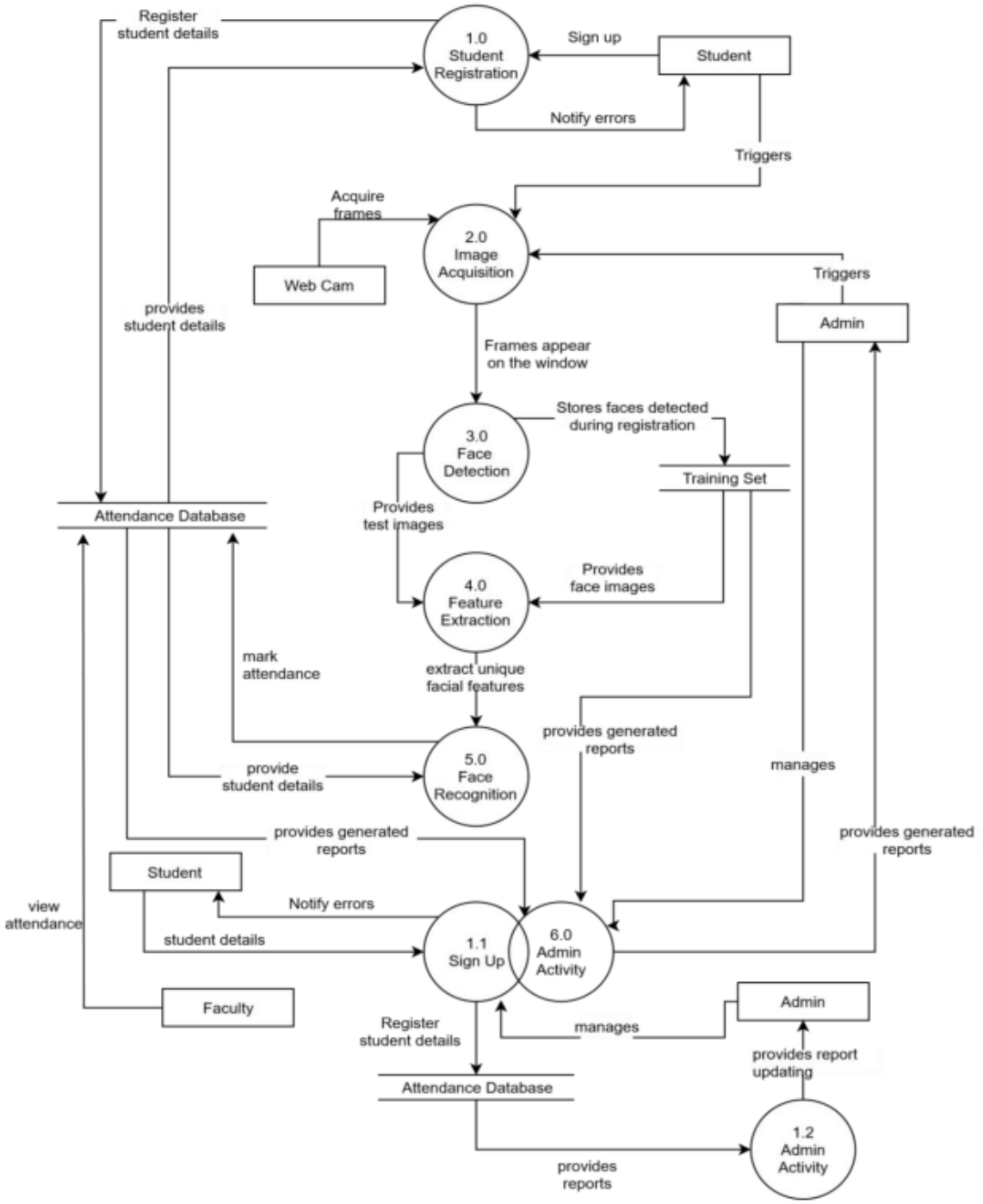

Fig 3.3 Data Flow Diagram Level 1

3.3 Implementation

In this section, we go through the model's implementation specifics as well as the performance of our approach and other relevant ideas in terms of prediction. Once the designs are deemed to be viable, technical implementation begins. Implementing the project was the toughest part as all the coding was done in this phase. The complete automated attendance system can be divided into four categories: face detection, feature extraction, feature matching and marking attendance.

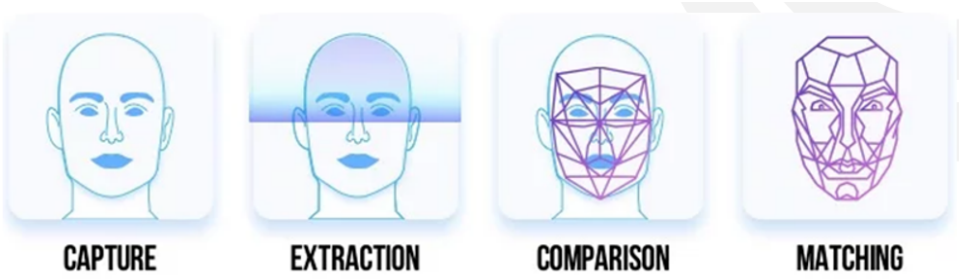

Fig 3.4 Implementation categories

3.3.1 Database Creation

For the database, a set of at least 10 pictures with various sizes, color, lighting, angles are taken and stored in a folder under their name.

3.3.2 Face Detection

The face detection method is used to find the faces present in the image, extract the faces, display, and create a compressed file to use it further for feature extraction.
Methods used in Face Detection:

- **HAAR CASCADE FACE DETECTION:**
  This method has a simple architecture that works nearly real-time on CPU. Also it can detect images at different scales. But the major drawback is that it gives false results as well as it does not work on non-frontal images.
- **DLIB (HOG) FACE DETECTION:**
  It is the fastest method on CPU which can work on frontal and slightly no-frontal images. But it is incapable of detecting small images and handling occlusions. Also, it often excludes some part of the chin and forehead while detection.
- **DLIB (CNN) FACE DETECTION:**
  It works very fast on GPU and is capable of working for various face orientations in images. It can also handle occlusions. But the major disadvantage is that it is trained on a minimum face size of 80*80 so it can not detect small faces in images. It is also very slow on the CPU.
- **MTCNN:**
  This method gives the most accurate results out of all the four methods. It works for faces having various orientations in images and can detect faces across various scales. It can even handle occlusions. It does not hold any major drawback as such but is

comparatively slower than HOG and Haar Cascade Method.

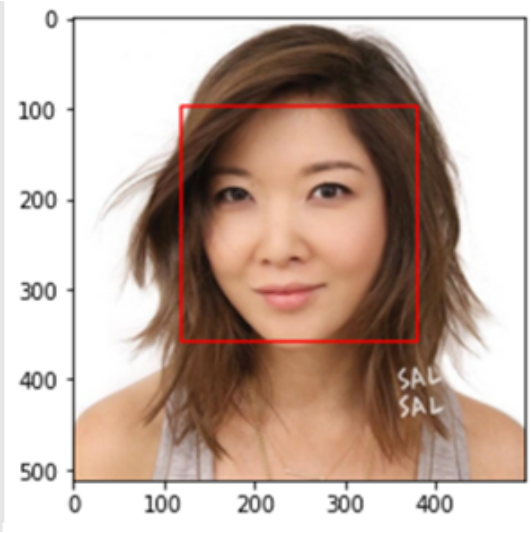

Fig 3.5 Haar Cascade

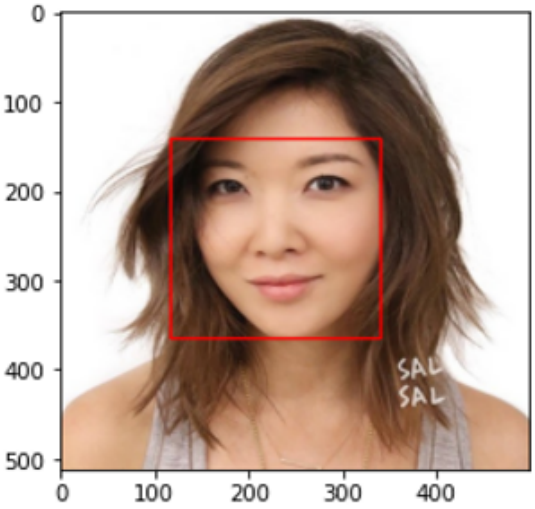

Fig 3.6 DLIB(CNN)

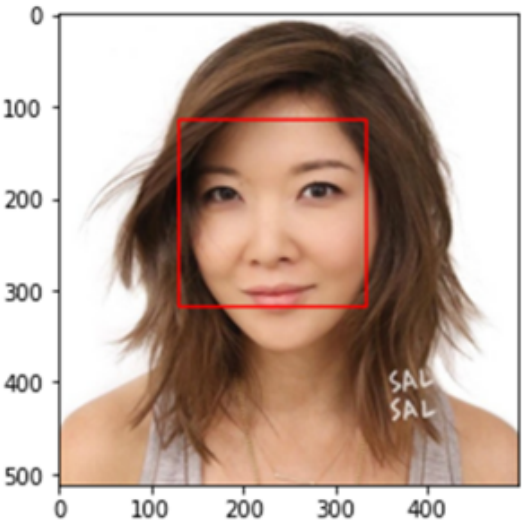

Fig 3.7 DLIB(HOG)

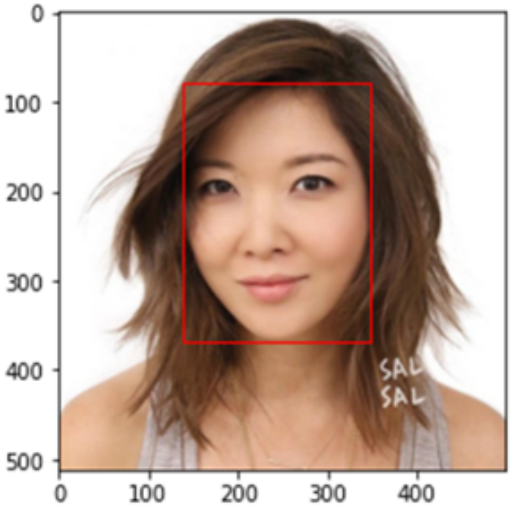

Fig 3.8 MTCNN

We used the MTCNN library to create a face detector and extract faces for our use with FaceNet face detector models in subsequent sections.

```
def extract_face(filename, required_size=(160, 160)):
  image = Image.open(filename)
  image = image.convert('RGB')
  pixels = asarray(image)
  detector = MTCNN()
  results = detector.detect_faces(pixels)
  x1, y1, width, height = results[0]['box']
  x1, y1 = abs(x1), abs(y1)
  x2, y2 = x1 + width, y1 + height
  face = pixels[y1:y2, x1:x2]
  image = Image.fromarray(face)
  image = image.resize(required_size)
  face_array = asarray(image)
  return face_array
```

Fig 3.9 Face Detection Code

3.3.3 Face Extraction

Face extraction is the basic and most important initializing step for face recognition. It extracts the biological components of the face. These biological components are the features of the face which differ from person to person. There are various methods which extract various combinations of features, commonly known as nodal points. No two people can have all the nodal points like each other except for identical twins. Each person's face has about 80 nodal points which, when measured by facial recognition software, create a faceprint based on a numeric code representing a face in a database. Some features measured by the software are depth of the sockets, distance between the eyes, shape of the cheekbones, width of nose and length of the jawline

Facial feature extraction has two approaches:

- **Shallow Approach**

  1. PCA

  PCA is used to reduce the dimensionality of the data. In PCA, the original features of the dataset will be converted into a linear combination of uncorrelated variables (features). These combinations are known as Principal Components. PCA increases algorithm performance and improves visualization. PCA results in loss of information if the number of Principal Components is not selected wisely.

  2. LDA

  It is a dimensionality reduction technique used to classify different classes based on the features of the supervised data. The major drawback of LDA is the so-called Small Sample Size Problem and non-linearity problem. SSS problem occurs when the sample is quite small as compared to the dimension of the data.

  3. Cosine Similarity

  The measure of cosine angle between two vectors is known as the cosine

similarity between two vectors, the closer the cosine value to 1 and greater will be the possibility of a match. One vector among the two vectors is the test data (detected face) and the other is the vector of the training dataset. But it gives false results for sparse numeric data.

4. HOG

It only uses magnitude values of pixels without including the neighboring values which lead to the extraction of improper features during image rotation.

5. SIFT

SIFT is relatively similar to a sparse descriptor. Sparse Descriptor is a technique that initially detects the key points in the image and then generates descriptors at these points. SIFT consists of scale rotation and affine transformation properties as well. But it requires a long-running time as compared to other systems.

- **Deep Approach**

1. VGG

VGG uses various architectures such as VGGFace1, VGGFace2 by Keras. The basic difference among these models is the number of layers included in its architecture that varies from model to model. These models have quite good accuracy.

2. Face Recognition API

FaceRecognition API is easier to use. It has a much easier architecture to implement with some inbuilt libraries required for feature recognition. You need to upload a picture and call the FaceRecognition API. The API then simulates the browser using the user's information to call the recognition points. It works well in real-time and holds good accuracy.

3. FaceNet Keras

FaceNet Keras is a one-shot learning model. It fetches 128 vector embeddings as a feature extractor. It is even preferable in cases where we have a scarcity of datasets. It consists of good accuracy even for such situations.

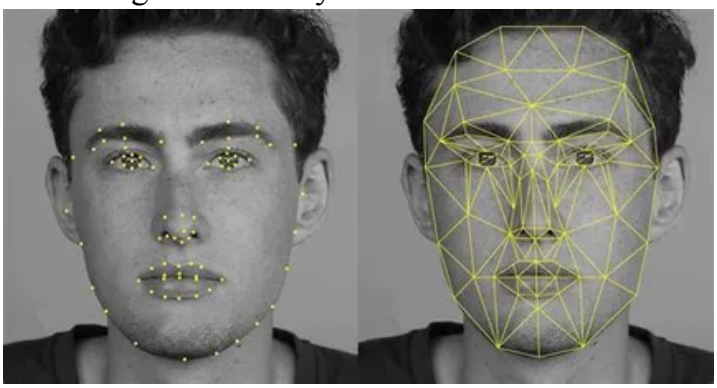

Fig 3.10 128 Face Embeddings

The FaceNet model is used as part of the classifier itself, in other words, we use it to pre-process a face to create a face embedding that is stored and used as input to the classifier model. The below function transforms the dataset into 93 face embeddings, each composed of a 128-element vector.

```
def get_embedding(model, face_pixels):
  face_pixels = face_pixels.astype('float32')
  mean, std = face_pixels.mean(), face_pixels.std()
  face_pixels = (face_pixels - mean) / std
  samples = expand_dims(face_pixels, axis=0)
  yhat = model.predict(samples)
  return yhat[0]
```

Fig 3.11 Feature Extraction using FaceNet Keras Code

3.3.4 Face Classification

Face classification is a geometry-based or template-based algorithm used to classify the features of the test data among different classes of facial features in the training data. These template-based classifications are possible using various statistical approaches.
The well-known methods used in feature classification can be given as:

- ***Euclidean distance***
  It is a distance-based feature classification method that calculates the distance between the facial nodes and the face which has the minimum difference between these distance values is the match. But it is suitable for the datasets having a smaller number of classes and lower dimensionality features.
- ***Cosine Similarity***
  In cosine similarity, the solution that we obtain after calculating the cosine of an angle is brought into concern. Here, we would compare the differences between these results. The more the value is closer to 1, the greater is the probability of the match. But it may give a false result if the test data features are incomplete (i.e., if the resultant value is 0 then the features do not match, and if nearly all the features match, then the value is 1).
- ***SVM***
  SVM (Support vector machine) creates an optimal hyperplane to classify the classes of training dataset based on the different features of the face. The dimensionality of the hyperplane is one less than the number of features. Different kernels can be applied to see what features are used by the classifier to remove the features if required. This can help to improve speed.
- ***KNN***
  KNN (K-Nearest Neighbor) is all about the number of neighbors i.e., the k value. In KNN, if k=3 then we check that the data is close to which 3 data points. Thereafter, it is decided that closest data points belong to which class. Now, the test data is predicted to be in this class KNN has a curse of dimensionality problem which can be solved by applying PCA before using KNN classifier.
- ***ANN***

KNN (K-Nearest Neighbor) is all about the number of neighbors i.e., the k value. In KNN, if k=3 then we check that the data is close to which 3 data points. Thereafter, it is decided that closest data points belong to which class. Now, the test data is predicted to be in this class KNN has a curse of dimensionality problem which can be solved by applying PCA before using KNN classifier.

Since we were working on normalized face embedding inputs, we have used the Linear Support Vector Machine. This is because the method is very effective at separating the face embedding vectors.

```
data = load('/content/test-embeddings.npz')
trainX, trainy, testX, testy = data['arr_0'], data['arr_1'], data['arr_2'], data['arr_3']
print('Dataset: train=%d, test=%d' % (trainX.shape[0], testX.shape[0]))
in_encoder = Normalizer(norm='l2')
trainX = in_encoder.transform(trainX)
testX = in_encoder.transform(testX)
out_encoder = LabelEncoder()
out_encoder.fit(trainy)
trainy = out_encoder.transform(trainy)
testy = out_encoder.transform(testy)
model = SVC(kernel='linear', probability=True)
model.fit(trainX, trainy)
yhat_train = model.predict(trainX)
yhat_test = model.predict(testX)
```

Fig 3.12 Feature Classification Code

3.3.5 Attendance Marking

The attendance is marked in an excel sheet which updates automatically with a different date added to a new column in the sheet.

```
wb=openpyxl.Workbook()
sheet=wb.active
c1=sheet.cell(row=1,column=1)
c1.value='Name'
c2=sheet.cell(row=1,column=2)
d=date.today()
val=d.strftime("%d/%m/%Y")
c2.value=val
for i in range(0,9):
  a=sheet.cell(row=i+2,column=1)
  a.value=names[i]
  b=sheet.cell(row=i+2,column=2)
  b.value='Absent'
def attendance(name):
  for i in range(1,10):
    if str(sheet.cell(row=i+1,column=1).value)==str(name):
      b=sheet.cell(row=i+1,column=2)
      b.value='Present'

wb.save("demo.xlsx")
```

Fig 3.13 Attendance Marking Code

4. RESULTS

This section discusses the result of the automated attendance system. The system is tested using a computer with the specification as follows, a processor of Intel® Core™ i5-8250U CPU @1.60GHz 1.80GHz and a ram size of minimum 4GB.

Facial Recognition Test Result

To find out the accuracy level of face recognition used in this system, a photo of a student whose face has been trained is taken. The size of the photo taken is 160 X 160. This is done to make the face recognition process accomplish faster, because the more significant the image size, the time needed to recognise the student's face becomes longer. Running the face detection function, a figure is created with the faces detected in one of the student's directory/folder. Below we can observe that each face was correctly detected and that we have a range of lighting, skin tones, and orientations in the detected faces.

Fig 4.1 Plot of 15 photos of a student detected from the training dataset

To get face embeddings, FaceNet Keras model is loaded correctly. The train dataset was then transformed into 76 face embeddings, each composed of a 128-element vector. The 28 examples in the test dataset were also suitably converted to face embeddings.

```
Loaded:  (76, 160, 160, 3) (76,) (23, 160, 160, 3) (23,)
WARNING:tensorflow:No training configuration found in the save file, so the model was
Loaded Model
WARNING:tensorflow:7 out of the last 11 calls to <function Model.make_predict_function
(76, 128)
(23, 128)
```

Fig 4.2 Face Embeddings

The model is then evaluated on the train and test dataset, showing perfect classification accuracy of 91.304 on the test data set. This is not surprising given the size of the dataset and the power of the face detection and face recognition models used. The accuracy may be affected by illumination of light leading to the results.

```
Dataset: train=76, test=23
Accuracy: train=100.000, test=91.304
```

Fig 4.3 Accuracy of the model

Below are the results of the predicted values, there are few cases where the prediction has been wrong. This might be because of a small training dataset and difference in the illumination of the light, and certain facial features.

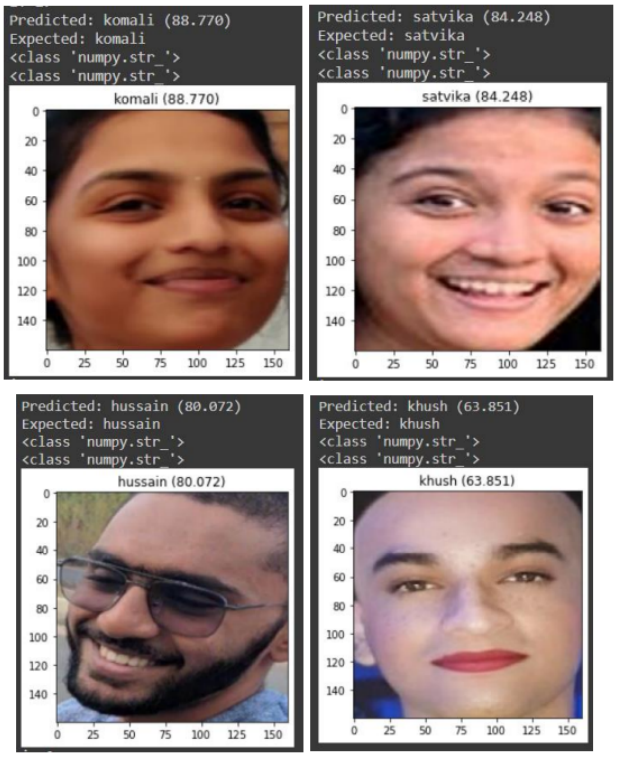

Fig 4.4 Detected and correctly identified by Model

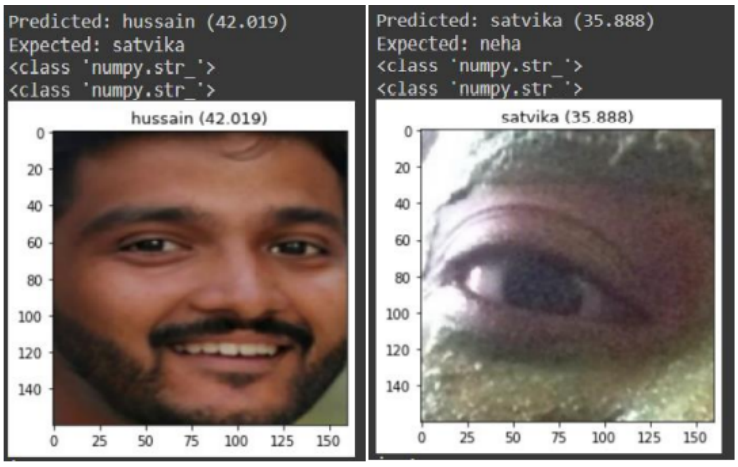

Fig 4.5 Faces detected and wrongly identified by model

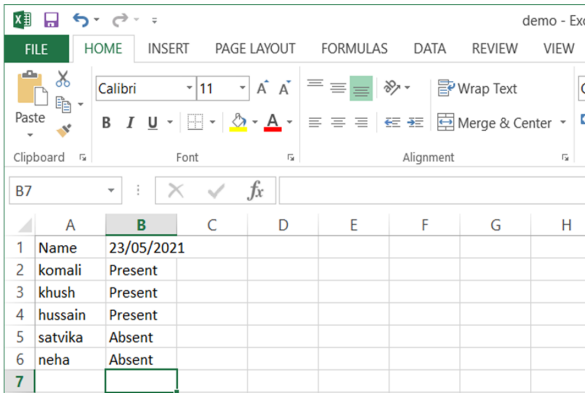

Fig 4.6 Attendance Marked in the sheet

5. CONCLUSION

Before the development of this project, there were many loopholes in the process of taking attendance using the traditional methods which caused troubles to most of the institutions. Automated Attendance System using MTCNN and FaceNet is expected to be able to replace the old manual attendance process, which is currently used. The system has also proved to be time saving, securing and conquering the defects by merely saving resources but also reducing human intervention in the whole process. This standalone system detects the person which was already given in the dataset to track and an embedding being created was successfully detected with an accuracy of 91.304%. In real time scenarios, MTCNN and FaceNet algorithms outperforms other algorithms with better recognition rate and low false positive rate. SVM has also proved to be a better classifier when compared to others.

Future Work

For future work, the plan is to use cloud-based face recognition to speed up the face recognition process. The future work is to also improve the recognition rate of algorithms when there are unintentional changes in a person like having a mask or change in features due to surgery. The system should also be able to detect identical twins distinctly. Currently, the system is not curated to detect multiple faces from a single image which is yet to be achieved soon. The system should be able to take input from the camera and be developed into a web application. At the same time, there are many places worth further research and discussion in terms of improving detection accuracy and recognition speed by implementing image quality check and denoise methods with less maintenance and low cost hardware. It can also further be developed as a smart attendance system based on real-time video processing by implementing the actual sign-in accuracy rate, stability analysis, skipping rate analysis and interface settings.

REFERENCES

1. Asri Nuhi, Agon Memeti, Florinda Imeri, Betim Cico, “Smart Attendance System using QR code”, 9th Mediterranean conference Embedded Computing, Budva, Montenegro, 2020
2. Hao Yang, Xiaofeng Han “Face Recognition Attendance System Based on Real-Time Video Processing” is supported in part by the Basic Public Welfare Research Project of Zhejiang Province under Grant LGF20H180001, 2020
3. Abhishek Pratap Singh, SunilKumar S Manvi, Pratik Nimbal, Gopal Krishna Shyam, “Face Recognition System Based on LBPH Algorithm”, International Journal of Engineering and Advanced Technology(IJEAT), ISSN: 2249-8958, Volume-8, Issue-5S, May 2019
4. Sathiyanarayanan P, Karthick R, Gokulnath R, Yogesh Kumar S, “Automatic Attendance System using Face Recognition”, Journal of Emerging Technologies and Innovative Research, Issues 7, Volume 7, May 2020
5. Narayana Darapaneni, Aruna Kumari Evoori, Vijaya Babu Vemuri, Thanga Selvi Arichandrapandian, “Automatic Face Detection and Recognition for Attendance Maintenance”, IEEE 15th International Conference on Industrial and Information Systems(ICIIS), DOI: 10.1109/ICIIS51140.2020.9342670, November 2020
6. R Hartanto, M N Adji, “Face Recognition for Attendance System Detection”,10th IEEE International Conference on Information Technology and Electrical Engineering (ICITEE), 2018, pp. 376-381
7. S Aryal, R Singh, A Sood, G Thapa, “Automatic Attendance System using Deep Learning”, SSRN Electron J, 2019
8. P Kowsalya, J Pavithra, G Sowmiya, C K Shankar, “Attendance Monitoring System using Face Detection & Face Recognition”, International Research Journal of Engineering and Technology(IRJET), volume 06, no. 03, pp. 6629-6632, March 2019, ISSN: 2395-0072
9. Shreyank Sawhney, Karan Kacker, Samyak Jain, Shailendra Narayan Singh, Rakesh Garg, “Real-Time Smart Attendance System using Face Recognition Techniques”, 9th International Conference on Cloud Computing, Data Science and Engineering(Confluence), 2019
10. Jeong J P, Kim M, Lee Y, Lingga P , “IAAS: IoT-Based Automatic Attendance System with Photo Face Recognition in Smart Campus”, International Conference on Information and Communication Technology Convergence (ICTC), 2020, DOI:10.1109/ICICTC70.2020.928927
11. C. Szegedy, Wei Liu, Yangqing Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 1–9