## Automated Attendance System using RFID and Face Recognition

#### BACHELOR OF TECHNOLOGY

#### IN

**INFORMATION TECHNOLOGY ENGINEERING**

Submitted by

###### NAVEEN KUMAR SINGH (1614313048)

###### PRAFULL DEEP SAHU (1614313057)

###### SATYENDRA KUSHWAHA (1614313069)

Under the Guidance of

###### PROF. UPDESH JAISWAL



**DEPARTMENT OF INFORMATION TECHNOLOGY**

**IMS ENGINEERING COLLEGE, GHAZIABAD**

**DR. ABDUL KALAM TECHNICAL UNIVERSITY, LUCKNOW MAY, 2020**

**CERTIFICATE**

This is to certify that project entitled **“Automated Attendance System Using RFID and Face Recognition”** submitted by NAVEEN KUMAR SINGH, PRAFULL DEEP SAHU, SATYENDRA KUSHWAHA in partial fulfillment of the requirement for the award of degree of Bachelor of Technology in Mechanical Engineering contains the bonafide work of above students done under our supervision and that the same work has not been submitted elsewhere for the award of any degree to the best of our knowledge.

**Project Supervisor PROF. (Dr.) S.N. Rajan**

**PROF. UPDESH JAISWAL HOD, IT**

# 

# 

# ACKNOWLEDGEMENT

We would like to express our gratitude to all people who have helped and guided us to accomplish our project efficiently and successfully. Small but an important and timely help can prove be a milestone in one’s life. Every human being has such kind of experience. Being human, we also have the same feeling of gratefulness of today we also have achieved an important milestone in our life.

This project is dedicated to all the people, whom we met, took guidance, talked interviewed and learned something from them, at this occasion we sincerely thank all of them while submitting this project report.

We would like to heartily thank **Prof. Ms. Updesh jaiswal** who has provided us necessary information and guidance for the success of the project. Last, we would like to extend thank to all our classmates also. We would like to thank **Prof. Dr . S.N. Rajan (HOD, INFORMATION TECHNOLOGY ENGG. DEPARTMENT)**, librarian without whom this would have not been possible. We always have felt the invisible help from the almighty, without the blessing of almighty, we could not have succeeded.

###### Your’s Sincerely

Naveen Kumar singh(1614313048)

Prafull Deep Sahu(1614313057)

Satyendra Kushwaha(1614313069)

### ABSTRACT

The goal of this project is to create the automated attendance system using face recognition. When a user takes a picture of a human, our application searches related information in a database using image recognition. Since a user of the application can take a picture under different circumstances, the used image recognition algorithm had to be invariant to changes in illumination and view point. The execution of the algorithm had to run on the mobile phone, so there was need for a lightweight image recognition algorithm. A couple of these invariants can be grouped as a feature vector, identifying an image uniquely. By computing the distances between the vectors of an unknown image and known database images, a best match can be selected.

Android is flexible and provides many tools for developing applications. This allowed to develop our museum guide application in a limited amount of time. We explored, evaluated and used many of Android’s possibilities.

My SQL used for the application database. Insertions, Deletions and changes of the data in the system can do straightforward via the designed GUI without interacting with the tables. Different presentation of information is obtainable from the system.

# Contents

1. Introduction 1
   1. Introduction to Project 1
   2. [Problem Statement 2](#_TOC_250030)
   3. Objective 3
   4. [Methodology 3](#_TOC_250029)
2. Literature Survey 5
   1. [Face Detection 6](#_TOC_250028)
      1. [Haar-cascade classifier 7](#_TOC_250027)
      2. Local Binary Pattern(LBP) 9
   2. [Face Pre-processing 12](#_TOC_250026)
   3. [Face Recognition 14](#_TOC_250025)
      1. [Face recognition from still images 15](#_TOC_250023)
      2. Face recognition from intensity sequences 17
      3. [Face recognition from image sequences 18](#_TOC_250022)
      4. [Evaluation of face recognition systems. 20](#_TOC_250021)
      5. Two issues in face recognition 20
   4. Face Recognition Algorithms 22
      1. [Eigen faces 22](#_TOC_250020)
      2. [Fisher faces 25](#_TOC_250019)
      3. [Local Binary Patterns Histograms (LBPH) 26](#_TOC_250018)
   5. [Proposed Model 32](#_TOC_250014)
   6. [Image Acquisition 33](#_TOC_250013)
   7. [Face Detection 33](#_TOC_250012)
      1. [LBP Detection Algorithm 36](#_TOC_250011)
   8. [Face Pre- Processing 37](#_TOC_250010)
   9. [Face Recognition using Local Binary Pattern Histogram 38](#_TOC_250009)
      1. [Working of Local Binary Pattern Histogram 38](#_TOC_250008)
      2. Extracting the Histogram 40
      3. [Performing the face recognition 41](#_TOC_250007)
   10. [User Interface Design 43](#_TOC_250005)
3. Performance Analysis 45
   1. [Recognizing Faces 49](#_TOC_250003)
4. Conclusions 50
   1. Future Scope………………………………………………………………… 51

References………………………………………………………………………….52

**List of Figures**

1.4.1 System model for face detection & recognition 4

* + 1. The 5 Haar-like features used for detecting faces 7
    2. LBP calculation example 10
    3. Face image converted to grayscale and cropped 12
    4. Histogram of the image equalized 12
    5. Filter applied on the face 13
    6. Elliptical mask applied on the image 13

2.3.5 Same face appears differently under different illuminations 21

2.4.3 Local Binary Pattern can detected Texture primitives 27

2.5.3 Cascade classifier concept 28

3.1.1 Advantages of using Android 29

* + 1. Working of face detector 35
    2. Green rectangular frame on the detected face 37
    3. Cropped image of face detected 37
    4. Gray scaled image of cropped image 38
    5. Operation of local binary pattern histogram 39
    6. Histogram of the gray scaled image 40
    7. Use Case Diagram of the user 42
    8. Use Case Diagram of the Admin 43
    9. Capturing the face which is inside green rectangle 46
    10. Image inside green rectangle is captured and stored 47
    11. Landing page of face recognition application 49
    12. Face recognition of two student 50
    13. Face recognition of three student 51
    14. Face recognition of four student 52
    15. Face recognition of five student 53
    16. Face recognition of seven student 53
    17. Marking of attendance and reviewing it 54

# Chapter 1 Introduction

* 1. **Introduction**

Attendance systems of old practices are not quite efficient. now a days for keeping track on student’s attendance. Student enrollment in schools and colleges increasing every year and taking each student attendance plays a very important role. So, it is necessary to discuss the effective system which records the attendance of a student automatically.

Maintaining the attendance is very important in all the colleges for checking the performance of students. Every college has its own method in this regard. Some are taking attendance of students manually using attendance registers or marking attendance sheets or file based approach and some have adopted the methods of automatic attendance using some biometric techniques. But in these methods, students have to wait for a long time in making a queue at the time they enter inside the classroom.

Many biometric systems are available in the market but the key authentications are same in all of the techniques. Every biometric system consists of enrollment process in which the unique features of a person is stored in the database and after that, there are some processes of identification and verification of the person. These two processes compare the biometric feature of a person with previously stored template captured at the time of enrolment of a student. Biometric templates can be of many types like Fingerprints , Eye Iris, voice etc. Our system uses the face recognition approach for the automatic attendance of the students in the classroom environment without student intervention. The purpose ofideveloping the new attendance management system is to computerize the traditional methods of taking the attendance. Therefore, in order to drag the attention of students and make them interactive in observing technologies, we

try to move on to the latest upcoming trends on developing attendance systems. This is the reason for college attendance management system to come up with an approach that ensures a strong contribution of students in classrooms.

To track the attendance of the students, we have introduced the attendance management system. With the introduction of this attendance system, skipping classes for students without the staff’s knowledge have become difficult. Attendance management system is to count the number of students and urge students to attend the classes on time, so as to improve the quality of teaching.

Usually, a roll no. call is taken to determine whether the student is present in the class or not, which usually wastes a lot of time. In recent years, with the emerging technology and with the development of deep learning, face recognition has made great achievements, which leads us to a new way ofpthinking to solve the problem of student’s enrollment. So, in order to save time, the idea to count the number of students in a class automatically based on face recognition is incorporated. This system is developed by using face recognition technique which is used to detect the face of an individual. There are many different face recognition algorithms introduced to increase the efficiency of the system. The system provides an increased accuracy due to the use of a large number of features like Shape, color, LBP[11], Auto- Correlation etc. of the face. However, the face recognition still remains a challenging problem for us because of its fundamental difficulties regarding various factor like illumination changes, face rotation, facial expression etc.

# Problem Statement

When there are so many students in a college, it becomes more and more difficult to mark attendance for each student and it is time consuming too. The Existing system of any institute is a manual entry for the students. This system faces the issue ofwwastage of time and also becomes complicated when the strength is more than the usual. Here, the attendance is being carried out in the hand written registers. It is very tedious job for us to maintain the record of the user.

Whenever we have to measure the performance of students, finding and calculating the average of the attendance of each enrolled student is also a very complicated task for us. The human effort is more here. The retrieval of the information is not a piece of cake as the records are maintained in the hand written registers. This existing system requires correct feed on input into the respective field. Therefore we are in a need of an automated system for marking and maintaining attendance of the students. Let us suppose that the wrong inputs are entered, the application resist to work. So, the user finds it difficult to use the existing system.

# Objectives

Our objective is to detect unique faces with the help of mobile camera amidst the other natural components like walls, backgrounds etc. and to extract the unique features faces amongst other face characteristics such as beard, spectacles etc. of a face useful for face detection and recognition.

# Methodology

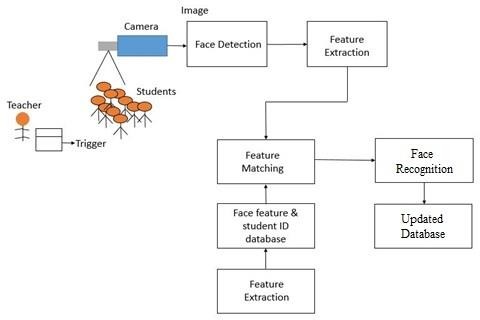
In our proposed system, the system is instantiated by the mobile .After it triggers then the system starts processing the image of the students for which we want to mark the attendance.

Image Capturing phase is one in which we capture the image of the students. This is the very basic phase from which we start initializing our system. We capture an image from our camera which predominantly checks for certain constraints like lightning, spacing, density, facial expressions etc. The captured image is resolute according to our requirements. Once it is resolute, we make sure it is either in .png or .jpeg format.

We take different frontal postures of an individual so that the accuracy can be attained to the maximum extent. This is the training database in which we classify every individual based on labels. For the captured image, from an every object we detect only frontal faces. This detects only face and removes every other parts since we are exploring the features of

faces only. These detected faces are stored somewhere in the database for further enquiry. Features are extracted in the extraction phase.

The detected bounding boxes are further queried to look for features extraction and the extracted features are stored in a matrix. For every detected phase, this feature extraction is done. Features that we look here are shape, edge, color, auto-correlation and LBP. Face is recognized once we completed the extracting features. The features which is already trained with every individual is compared with the detected faces features and if both features match, then it is recognized. Once it recognizes, it is going to update in the student attendance database. Once the process is completed, the testing images remains.



System Model of face detection & recognition

**Chapter 2 Literature Survey**

One of the most successful applications of image analysis and understanding, face recognition has recently received a significant attention, especially during the past few years. In addition to this, the problem of machine recognition of human faces continues to attract researchers from disciplines such as image processing, pattern recognition, neural networks, computer vision, computer graphics and psychology. The strong need for user-friendly systems that can secure our assets and protect our privacy without losing our identity in a sea of numbers is obvious.

We as humans use faces to recognize and identify our friends and family. Computers can now also identify people automatically using stored information such as figure, iris or face to identify a particular person. Earlier many face recognition algorithms were used to achieve fully automated face identification process. It was not fully automated and it required anual inputs of the location of the eyes, ears, nose and mouth on the images then it calculates a distance to some common point then it compares it to the stored data. The still image problem has several inherent advantages and disadvantages. However, if only a static picture of an airport scene is available, automatic location and segmentation of a face could pose serious challenges to any segmentation algorithm.

On the other hand, if a video sequence is available, segmentation of a moving person can be more easily accomplished using motion as a cue. But the small size and low image quality of faces captured from video can significantly increase the difficulty in recognition. Face recognition and sometime is called face identifying is simply putting a label to known faces just like human as mentioned above, we learn the faces of our family

and celebrities just by looking at their faces. Most of the recent techniques involve at least three steps:

* Face detection:
* Face preprocessing:
* Face recognition

# Face Detection

Face detection is a process of locating a face inside an image frame, regardless of the identity of that face. Before recognizing a face, it is first essential to detect and extract the faces from the original pictures. Face Detection target on finding the faces in an image and probably extract them to be used by the face recognition algorithm. In recent years, many methods are proposed for detecting the face.

In face detection methods, those who are depending on training sets to capture the huge unevenness in facial features have enticed much attention and given the best results. Generally these methods scan the input picture at all potential area and scales then as the sub windows either as non-face or face. Further there is an effective detection technique using *Harr-like features* and anoter as a quick training algorithm. For recognizing a face, the algorithms compare only faces. Any other element in the picture that is not part of a face deteriorates the recognition.

There are several existing algorithms for detecting faces. Prior to year 2000 there were many techniques for face detection, however they were mostly unreliable, slow and require manual inputs. In 2001 the invention of Haar-based-cascade Classifier that revolutionize the face detection method. It can detect objects in real time with an accuracy of 95%. It works not only for frontal face view but it can detect faces from side view as well.

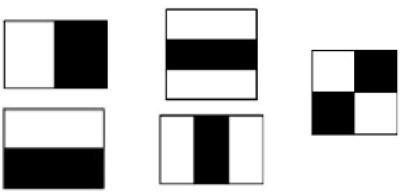
## Haar-cascade classifier

Haar-cascade is a method, in which it trains a machine learning for detecting objects in a picture. it can be used to detect faces. The basic idea of the Haar-base face detector is that if you look at most frontal faces, the region with the eyes should be darker than the forehead and cheeks, and the region with the mouth should be darkerithan cheeks, and so on.

It typically performs about 20 stages ofocomparisons like this to decide if it is a face or not, but it must do this at each possible position in the image and for each possible size of the face, so in fact it often does thousands ofachecks per image.. The name of this method is composed of two important words, Haar and Cascade. Haar belongs to Haar-like features which is a weak classifier and will be used for the face recognition.

A Haar-like feature is a rectangle which is split into two, three or four rectangles. Each rectangle is black or white. This shows the different possible features. A Haar- cascade needs to be trained with various positive and negative pictures. The objective is to extract the combination of these features that represents a face. While a positive picture contains the object which has to be recognized, a negative picture represents a picture without the object.

In the context of face detection, a positive picture possesses a face, and a negative picture does not. This machine learning requires grayscale pictures. The intensity of gray will be used to detect which feature is represented. These features can be found by calculating the sum of the dark pixels in an area subtracted by the sum of the bright pixels.



The 5 Haar-like features used for detecting faces

The basic principles in this method is based on are as follows:

* Images used in the integral representation that allows a machine to calculate the necessary object features.
* Using Haar-like features, the desired feature of the face can be found.
* Adaptive Boosting used to select the most suitable characteristics for the desired object to this part of the image.
* All the features are input to the classifier, which gives the result true or false.

The extracted combination of features from the training part will be used for detecting faces in a picture. To detect a face in an unknown picture is the combination of the features will be researched. The features are tried to be matched only in a block of pixels defined by a scale. Each feature of the combination will be tried to be matched one by one in the block. If one of the features does not appear in the block, the research in it will be stopped. The remaining features will not be tested because the machine concludes that there is no face in this block. Then, a new block is taken, and the process will be repeated.

The 5 Haar-like features used for detecting faces pixels with the researched combination in cascade which explains the second word in the name of the method. This method is efficient to detect an image without faces because only a few tests need to be run to infer that the image does not contain a face. A face is consequently detected when each feature of the combination has been recognized correctly in a block. We can see that the eyes are darker than the cheeks and the middle of the nose is brighter. All these features which were extracted from the training are used to find a pattern to represent a face. The process will proceed block by block until the last one. After checking the last block, the scale is increased, and the detection process starts again. The process is repeated several times with different scales to detect faces of different size. Only few pixels are different between two neighbor blocks. Therefore, each time a face is detected in a picture, the same face is detected in different blocks.

All the detected faces that concern the same person are merged and are considered as one at the end of the entire process. The accumulation of these weak classifiers builds a face detector able to detect faces very fast with a suitable accuracy. A Haar- cascade classifier has to be trained only once. Thus, it is possible to create one’s own Haar-cascade or use one which has already been trained.

## Local Binary Patterns (LBP)

The Local Binary Pattern operator is also known as LBP . The Local Binary Pattern (LBP) is a basic, effective and dominant texture operator, which labels the pixels of a picture by thresholding the neighbouring of each pixel, resulting in a binary number as shown in Figure 3.5.2. For each pixel, the algorithm considers the eight (8) neighbouring pixels. Then based on the gray-scale value of the selected pixel, it allocates the neighboring pixels the value of 0 or 1. Therefore, every pixel will have a string of binary values. Figures 3.5.3 shows an example of the calculation.

LBP(xp,yp)= ∑7 𝑠(𝑖𝑛 − 𝑖𝑝)2𝑛

𝑛=0

Where (xp, yp) is the pixel of an image, n signifies the neighboring pixel, (in) and (ip) the individual gray level of neighboring and central pixel, and s(x) can be described by:

S(x) = 1, x ≥ 0

S(x) = 0, x < 0

The bit is attained for every neighbouring pixel and is used in a pre-defined order to create a certain result. The final result will be in between 0 and 255, using 8 neighbouring pixels. the whole Local Binary Pattern operator extraction process.

it has proven to be a complex task for a computer, as it has many variables that can impair the accuracy of the methods, for example: illumination variation, low resolution, occlusion, amongst other.

In computer science, face recognition is basically the task of recognizing a person based on its facial image. It has become very popular in the last two decades, mainly because of the new methods developed and the high quality of the current videos/cameras.

# Face Pre-processing

Any of the previous methods can be used for extracting faces from input pictures. The next step is to pre-process these faces in order to make the training phase easier and improve the probability to recognize a person correctly. The training data will be standardized. Not all the pictures have the same zoom on the face and have maybe not all the same size. Most of the algorithms for facial recognition require the same size for the entire training set. Pre-processing includes different modifications. First of all, the faces need to be centered in the picture in the same way. The location of the two eyes and the nose is often used as a landmark for centering faces. The aim is to have the eyes at the same level and the nose at the same position for all images. To apply these modifications, the coordinates of the landmarks are needed. For that, it is possible to use a Haar-cascade classifier for detecting nose and eyes.

After detecting a face in the frame, we can now process the face inside the green rectangle. Face recognition is susceptible to changes in lighting conditions, face orientation, face expression, so it is paramount to diminish these differences as much as possible. There are numerous techniques to eliminate those issues. Some of these techniques are:-

* Geometrical transformation and cropping: This procedure includes resizing of the image and rotating the image as well as removing background.

Face image converted to gray scale and cropped

•Histogram equalization: This process standardizes the brightness and contrast of the image.



Histogram of the image equalized

* Smoothing:In this process, the image noise is eliminated by applying some filters.



Filter applied on the face

* Elliptical mask: The elliptical mask removes some remaining background.



Elliptical mask applied on the image

Processing images can be computationally expensive in a PC . It requires higher processing power. Therefore, minimalizing the image processing in the mobile device is a must to achieve a real-time face recognition system.

# Face Recognition

We will discuss current developments in face recognition in upcoming various sections.

* we briefly review issues that are relevant from a psychophysical point of view.
* we will provide a detailed review of recent developments in face recognition techniques using images.
* face recognition techniques based on video are reviewed. Data collection and performance evaluation of face recognition algorithms are addressed.
* we discuss two important problems in face recognition that can be mathematically studied, lack of robustness to illumination and pose variations, and we review proposed methods of overcoming these limitations.

.

## Face recognition from still images

Face recognition involves three key steps:

(1)Detection and rough normalization of faces. (2)Feature extraction and accurate normalization of faces. (3)Identification and/or verification.

Sometimes, different subtasks are not totally separated. For example, the facial features (eyes, nose, mouth) used for face recognition are often used in face detection.

Though fully automatic face recognition systems must perform all three subtasks, research on each subtask is critical. This is not only because the techniques used for the individual subtasks need to be improved, but also because they are critical in many different applications. For example, face detection is needed to initialize.

### Key Steps Prior to Recognition: Face Detection and Feature Extraction

The first step in any automatic face recognition systems is the detection of faces in images. Depending on the type of classification system, features can be local features such as lines or fiducial points, or facial features such as eyes, nose, and mouth. Face detection may also employ features, in which case features are extracted simultaneously with face detection. Feature extraction is also a key to animation and recognition of facial expressions. Without considering feature locations, face detection is declared successful if the presence and rough location of a face has been correctly identified.

Three types of feature extraction methods can be distinguished:

1. Generic methods based on edges, lines, and curves.
2. Feature template based methods that are used to detect facial features such as eyes.
3. Structural matching methods that take into consideration geometrical constraints on the features.

These methods have difficulty when the appearances of the features change significantly, for example, closed eyes, eyes with glasses, open mouth. A template-based approach in detecting the eyes and mouth in real images was introduced. This method is based on matching a predefined parameterized template to an image that contains a face region. Two templates are used for matching the eyes and mouth respectively.

The shape model is generated by representing each set ofwlandmarks as a vector and by applying

Principal Component Analysis (PCA) to the data. Then, after each sample image is warped so that its landmarks match the mean shape, texture information can be sampled from this shape-free face patch. PCA to this data leads to a shape free texture model. To match a given image and the model, an optimal vector of parameters are searched by minimizing the difference between the synthetic image and the given one. After matching, a best-fitting model is constructed that gives the locations of all the facial features and can be used to reconstruct the original images.

### Recognition from Intensity Images

Face recognition is such a challenging yet interesting problem that it has attracted researchers who have different backgrounds: psychology, pattern recognition, neural networks, computer vision & computer graphics. It is due to this fact that the literature on face recognition is vast and diverse. The usage of a mixture of techniques makes it difficult to classify these systems based purely on what types of techniques they use for feature representation or classification. To have a clear and high-level categorization, we instead follow a guideline suggested by the psychological study of how humans use holistic and local features. Specifically, we have the following categorization:-

1. **Holistic matching methods**. These methods use the whole face region as the raw input to a recognition system. One of the most widely used representations of the face region is eigenpictures , which are based on principal component analysis.
2. **Feature-based matching methods.** Typically, in these methods, local features such as the eyes, nose, and mouth are first extracted and their locations and local statistical are fed into a structural classifier.
3. **Hybrid methods.** Just as the human perception system uses both local features and the whole face region to recognize a face, a machine recognition system should use both. One can argue that these methods could potentially offer the best of the two types of methods.

## Face recognition from image sequences

A typical video-based face recognition system automatically detects face regions, extracts features from the video, and recognizes facial identity if a face is present. In surveillance, information security, and access control applications, face recognition and identification from a video sequence is an important problem. Face recognition based on video is preferable over using images, motion helps in recognition of familiar faces when the images are negated, inverted or threshold. It was also demonstrated that humans can recognize animated faces better than randomly rearranged images from the same set. Though recognition of faces from video sequence is a direct extension of image-based recognition, in our opinion, true video based face recognition techniques that coherently use both spatial and temporal information started only a few years ago and need further investigation. Significant challenges for video-based recognition still exist; we list several of them here.

1. **The quality of video is low**. Usually, video acquisition occurs outdoors (or indoors but with bad conditions for video capture) and the subjects are not cooperative; hence there may be large illumination and pose variations in the face images. In addition, partial occlusion and disguise are possible.
2. **Face images are small.** Again, due to the acquisition conditions, the face image sizes are smaller (sometimes much smaller) than the assumed sizes in most

image-based face recognition systems. Small-size images not only make the recognition task more difficult, but also affect the accuracy of face segmentation, as well as the accurate detection of the fiducial points that are often needed in recognition methods.

1. **The characteristics of faces/human body parts.** During the past eight years, research on human behavior recognition from video has been very active and fruitful. Generic description of human behavior not particular to an individual is an interesting and useful concept. One of the main reasons for the feasibility of generic descriptions of human behavior is that the intraclass variations of human bodies, and in particular faces, is much smaller than the difference between the objects inside and outside the class. For the same reason, recognition of individuals within the class is difficult.
2. **Face and Feature Tracking.** After faces are located, the faces and their features can be tracked. Face tracking and feature tracking are critical for reconstructing a face model and feature tracking is essential for facial expression recognition and gaze recognition. Tracking also plays a key role in spatiotemporal based recognition methods which directly use the tracking information. In its most general form, tracking is essentially motion estimation. For images like faces, some regions are too smooth to estimate flow accurately, and sometimes the change in local appearances is too large to give reliable flow.
3. **Head tracking**, which involves tracking the motion of a rigid object that is performing rotations and translations.
4. **Facial feature tracking**, which involves tracking non-rigid deformations that are limited by the anatomy of the head, that is, articulated motion due to speech or facial expressions and deformable motion due to muscle contractions and relaxations.
5. **Complete tracking**, which involves tracking both the head and the facial features.

## Evaluation of face recognition systems.

Given the numerous theories and techniques that are applicable to face recognition, it is clear that evaluation and benchmarking of these algorithms is crucial. One of the most important facts learned in these evaluations is that large sets of test images are essential for adequate evaluation. It is also extremely important that the samples be statistically as similar as possible to the images that arise in the application being considered. Scoring should be done in a way that reflects the costs of errors in recognition. In planning an evaluation, it is important to keep in mind that the operation of a pattern recognition system is statistical, with measurable distributions of success and failure.

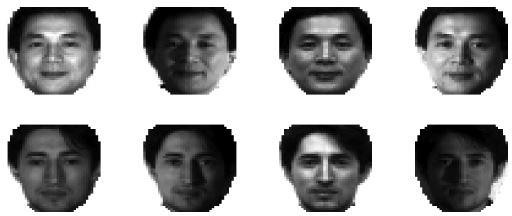
## Two issues in face recognition: Illumination and Pose variation

In this section, we discuss two important issues that are related to face recognition. However, face recognition in an uncontrolled environment is still very challenging. Though many existing systems build in some sort ofiperformance invariance by applying preprocessing methods such as histogram equalization or pose learning, significant illumination or pose change can cause serious performance degradation. In addition, face images can be partially occluded, or the system may need to recognize a person from an

image in the database that was acquired some time ago. These problems are unavoidable when face images are acquired in an uncontrolled, environment, as in surveillance video clips.

### The Illumination Problem in Face Recognition

The illumination problem is illustrated in this Figure, where the same face appears different due to a change in lighting.The changes induced by illumination are often largerithan the differences between individuals, causing systems based on comparing images to misclassify input images.



In each row, the same face appears differently under different illuminations

### The Pose Problem in Face Recognition

It is not surprising that the performance of face recognition systems drops significantly when large pose variations are present, in the input images. When illumination variation is also present, the task of face recognition becomes even more difficult. Here we focus on the out of plane rotation problem, since in-plane rotation is a pure 2D problem and can be solved much more easily.

# Facial recognition’s algorithms

After collecting enough images for the person , we can now use one of many algorithms for training the system to learn the face. Like most algorithm in machine learning, training of system must be completed first. There are several approaches for recognizing a face. The algorithm can use statistics, try to find a pattern which represents a specific person or use a convolutional neural network. These different approaches can be observed through the explanations of different algorithms. Some of the face recognition algorithms are the flowing:

* Eigenfaces.
* Fisherfaces.
* Local Binary Pattern (LBP) histograms.

## Eigenfaces

Eigenfaces is a method for performing facial recognition based on a statistical approach. The aim of this method is to extract the principal components which affect the most the variation of the images. This is a holistic approach, the treatment for predicting a face is based on the entire training set. There is no specific treatment between images from two different classes.

A class represents a person. Pre-processed pictures with grayscale are required for training the machine learning. The training part of Eigenfaces is to calculate the eigen vectors and the related eigen values of the covariance matrix of the training set.

## Fisherfaces

Since Eigenfaces maximization of both in-class and between-class scatter is undesirable, we wish to implement a method that maximizes the scatter between classes, while minimizing it within a class. This algorithm is a modification of Eigenfaces, thus also uses Principal Components Analysis (PCA). The main modification is that Fisherfaces takes into consideration classes. As it has been said previously,

Eigenfaces does not make the difference between two pictures from different classes during the training part. Each picture was affected by the total average.

The Fisherfaces approach is based on linear sub-spaces. It also uses a holistic approach. If we can train a specific face using different images of the same face under arbitrary lighting conditions, we can extract the underlying static face and represent it as a 3D linear subspace, indifferent to lighting directions, in a Lambertian surface. This highlights one of the most significant differences between the Eigenfaces method and the Fisherfaces method, specifically that the latter uses a labeled training set in order to gain a more accurate classification.

Since the training set is labeled, we can use class specific linear methods to reduce the dimensionality of the feature space. In Fisherspaces, a specific method of LDA (linear discriminant analysis) -Fisher's Linear Discriminant (FLD) is one of said class specific method, in that it reforms the scatter in order to make it more viable for classification.

# Proposed Model

Face recognition model which propose for the detection and recognition of the student faces for marking their attendance. The main modules used are:

### Dataset Generation form training:

This is first stage in which face dataset of the user is created, in which 10-20 images of each user are taken and the features used are user ID and username.

### Face Detection

For the face detection LBP-based face detector is used.

### Pre-processing:

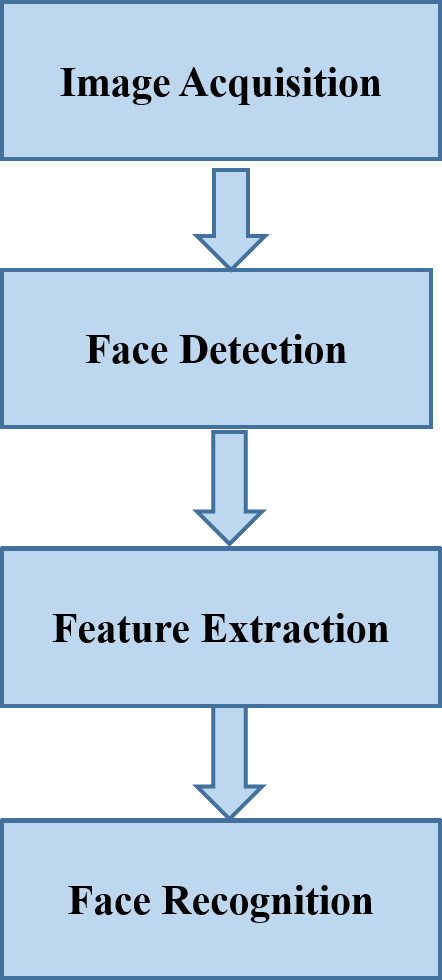
Steps involve in pre-processing are:

* + - 1. **Resizing**: Face is resized to a fixed pixel resolution after the face is detected
      2. **Cropping**: Background is removed from the image.
      3. **Grayscale Conversion:** The conversion of a colour image into a grayscale image.

### Feature Extraction and Recognition

Histogram principle based algorithm is used for feature extraction and recognition. The simple LBPH algorithm is selected for correct real-time processing of data as it is computational complexity is less and is more effective

compared to the other face recognition algorithms.



# Image Acquisition

Face recognition technology can be obtain from almost any camera or video system that creates the image of ample quality and determination. We are using our mobile phones powered by android for the image acquisition.

# Face Detection

Face detection has been enhanced as far as speed with the application of Haar-features with the involvement of the object detection method. One of the key confinements in early boosting-based methodologies is the strength to brightening and incomplete impediment of the face. To adapt to these restrictions, we propose to utilize

Local Binary Pattern (LBP) features to detect the face on our android application.

The simple plan of the LBP-based face detection is comparable to the Haar-based one, however, it uses histograms of picture for intensity comparisons, like boundaries, angles, and plane regions. The Haar-based classifier was terribly slow to observe the face within the frame. First the face image is distributed into smaller regions from that the Local Binary Pattern (LBP) area unit taken out and concatenated into one feature histogram expeditiously expressing the face in the image. The textures in the facial areas are domestically converted by the LBP patterns whereas the full form of the face is recuperated by the development of the face feature histogram.

# Steps for face Detection:

**Step 1**: Load the LBP cascade

**Step 2**: Instantiate the Open CV Cascade Classifier

**Step 3**: For each video frame received, call the cascade classifier

# Code for Face Detection:-

import numpy as np

import cv2,os

from PIL import Image

import pickle

import sqlite3

recognizer = cv2.createLBPHFaceRecognizer()

recognizer.load("recognizer\\trainingData.yml")

cascadePath = 'haarcascade\_frontalface\_default.xml'

faceCascade = cv2.CascadeClassifier(cascadePath)

path = 'dataSet'

def getProfile(id):

conn =sqlite3.connect("FaceBase.db")

cmd = "SELECT \* FROM People WHERE ID="+str(id)

cursor = conn.execute(cmd)

profile=None

for row in cursor:

profile=row

conn.close()

return profile

cam = cv2.VideoCapture(0)

font = cv2.cv.InitFont(cv2.cv.CV\_FONT\_HERSHEY\_SIMPLEX, 1, 1, 0, 1, 1) #create a font

while True:

ret, img = cam.read()

img2 = cv2.flip(img, 1)

gray = cv2.cvtColor(img2,cv2.COLOR\_BGR2GRAY)

faces = faceCascade.detectMultiScale(gray,

scaleFactor=1.2,minNeighbors=5,minSize=(100,

100),flags=cv2.CASCADE\_SCALE\_IMAGE)

for(x,y,w,h) in faces:

id, conf = recognizer.predict(gray[y:y+h,x:x+w])

cv2.rectangle(img2,(x,y), (x+w,y+h), (255,0,0),2)

profile=getProfile(id)

if(profile!=None):

cv2.cv.PutText(cv2.cv.fromarray(img2),"ID "+str(profile[0]),(x,y+h+30),font,200) cv2.cv.PutText(cv2.cv.fromarray(img2),"Name "+str(profile[1]),(x,y+h+60),font,200)

# Draw the Text

cv2.cv.PutText(cv2.cv.fromarray(img2),"Age "+str(profile[2]),(x,y+h+90),font,255) cv2.cv.PutText(cv2.cv.fromarray(img2),"Gender "+str(profile[3]),(x,y+h+120),font,255)

else:

cv2.cv.PutText(cv2.cv.fromarray(img2),"ID Unknown",(x,y+h+30),font,200) cv2.cv.PutText(cv2.cv.fromarray(img2),"Name Unknown",(x,y+h+60),font,200) cv2.cv.PutText(cv2.cv.fromarray(img2),"Age Unknown",(x,y+h+90),font,200) cv2.cv.PutText(cv2.cv.fromarray(img2),"Gender Unknown",(x,y+h+120),font,200)

cv2.imshow('im',img2)

if(cv2.waitKey(1) == ord('q')):

break

cam.release()

cv2.destroyAllWindows()

# Code for dataset Creator:-

import numpy as np

import sqlite3

import cv2

detector= cv2.CascadeClassifier('haarcascade\_frontalface\_default.xml')

cap = cv2.VideoCapture(0)

def insertOrUpdate(Id,Name):

conn = sqlite3.connect("FaceBase.db")

cmd = "SELECT \* FROM people WHERE ID="+str(Id)

cursor=conn.execute(cmd)

isRecordExist = 0

for row in cursor:

isRecordExist = 1

if(isRecordExist == 1):

cmd="UPDATE people SET Name="+str(Name)+" WHERE ID="+str(Id)

else:

cmd="INSERT INTO people(ID,Name) Values("+str(Id)+","+str(Name)+")

" conn.execute(cmd)

conn.commit()

conn.close()

Id=raw\_input('Enter user id')

name = raw\_input('Enter user name')

insertOrUpdate(Id,name)

sampleNumber=0

while(True):

ret, img = cap.read()

img2 = cv2.flip(img, 1)

gray = cv2.cvtColor(img2, cv2.COLOR\_BGR2GRAY)

faces = detector.detectMultiScale(gray, 1.3, 5)

for (x,y,w,h) in faces:

cv2.rectangle(img2,(x,y),(x+w,y+h),(10,0,130),2)

sampleNumber=sampleNumber+1

cv2.imwrite("dataSet/User."+Id+"."+str(sampleNumber)+".jpg",gray[y:y+h,x:x+w])

cv2.imshow('frame',img2)

if cv2.waitKey(300) & 0xFF == ord('q'):

break

elif sampleNumber > 20:

break

cap.release()

cv2.destroyAllWindows()

# Code for FACE DETECTION:-

import numpy as np

import cv2

detector= cv2.CascadeClassifier('haarcascade\_frontalface\_default.xml')

cap = cv2.VideoCapture(0)

while(True):

ret, img = cap.read()

gray = cv2.cvtColor(img, cv2.COLOR\_BGR2GRAY)

faces = detector.detectMultiScale(gray, 1.3, 5)

for (x,y,w,h) in faces:

cv2.rectangle(img,(x,y),(x+w,y+h),(10,0,130),2)

img2 = cv2.flip(img, 1)

cv2.imshow("Original", img2)

if cv2.waitKey(1) & 0xFF == ord('q'):

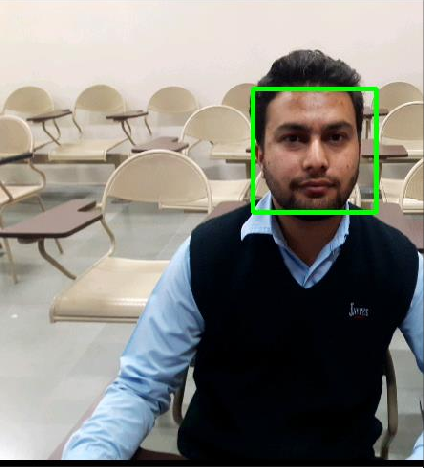
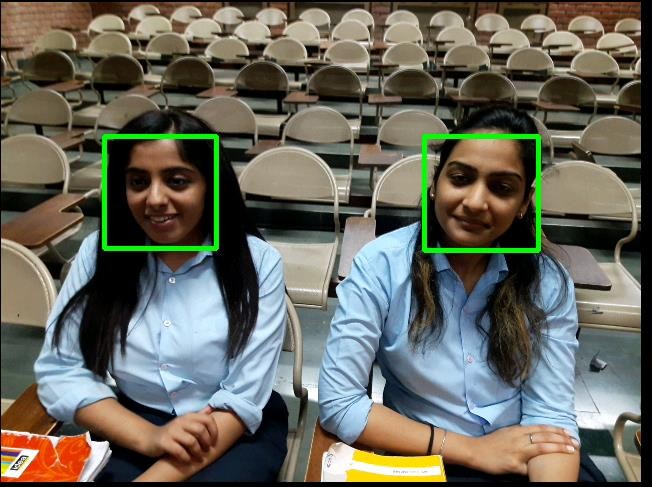
break

cap.release()

cv2.destroyAllWindows()

# Working of Face Detector-

In order to avoid the redundancy of computing the integral of rectangles, the integral pictures are calculated to speed up the calculation of the feature Figure shows the Face Detection of a Student in a classroom when we took the photo of the students. A green rectangular frame appear in the face of the students.

Green rectangular frame on the detected faces.

# Face Pre-Processing

Face pre-processing is used after detecting a face in the frame, as face is detected in figure, now the face which is inside the green rectangle is processed. Face recognition is prone to changes in lighting situations, face alignment, and face appearance, so it is supreme to reduce these changes as much as possible.

**Geometrical cropping:** These functions modify the geometry of a picture by resizing or cropping the image. Figure shows the cropped image which is detected in figure.



Cropped image of face detected.

This range is holding on in one computer memory unit. Zero represents the colour black and 255 is white. As it is going to be shown, grayscale is usually employed in computer vision and makes treatment of pictures easier once color is not essential. Figure shows the gray scale of the mentioned figure.



Gray scaled image of cropped image

# .1 Face Recognition using Local Binary Pattern Histogram

The face recognition systems operate basically in two modes:

**Verification or authentication of a facial image**: Compares the given facial picture with the facial picture identified with the client which is requiring the confirmation. It is a 1x1 comparison.

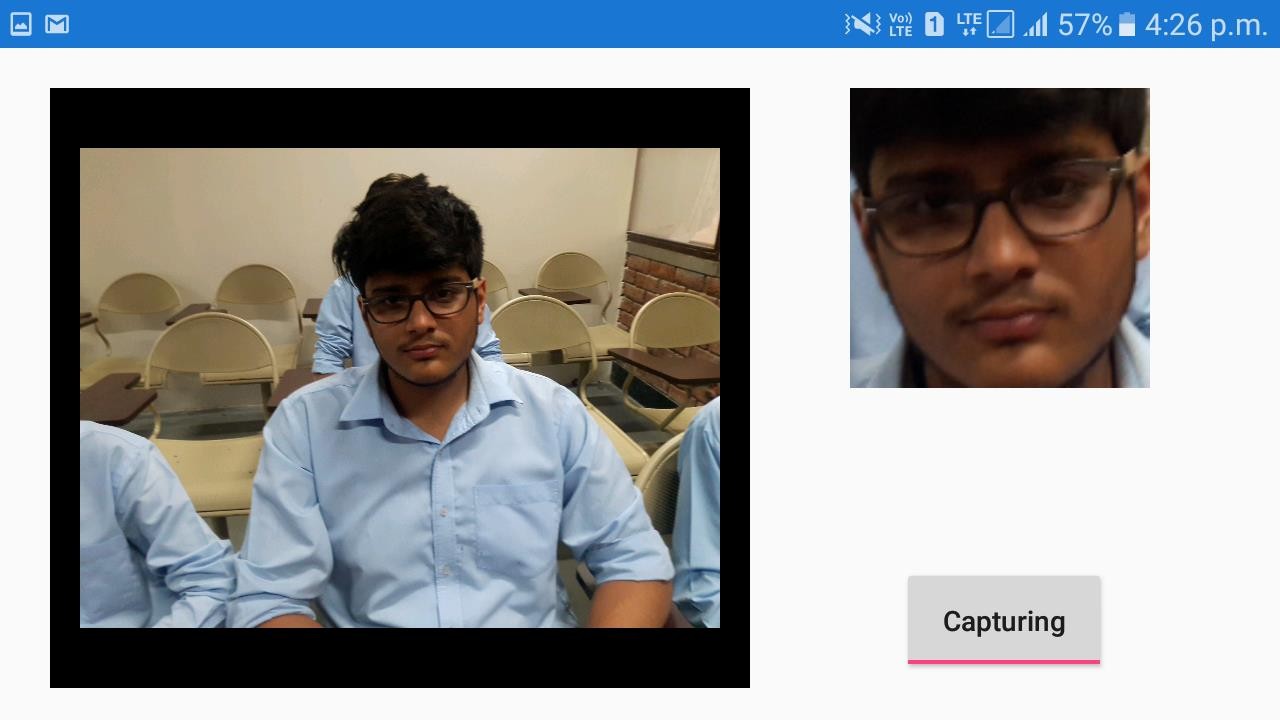
**Identification or facial recognition**: Compares the given facial picture with all facial picture from a dataset with the objective to discover the client that matches that face. It is a 1xN comparison.

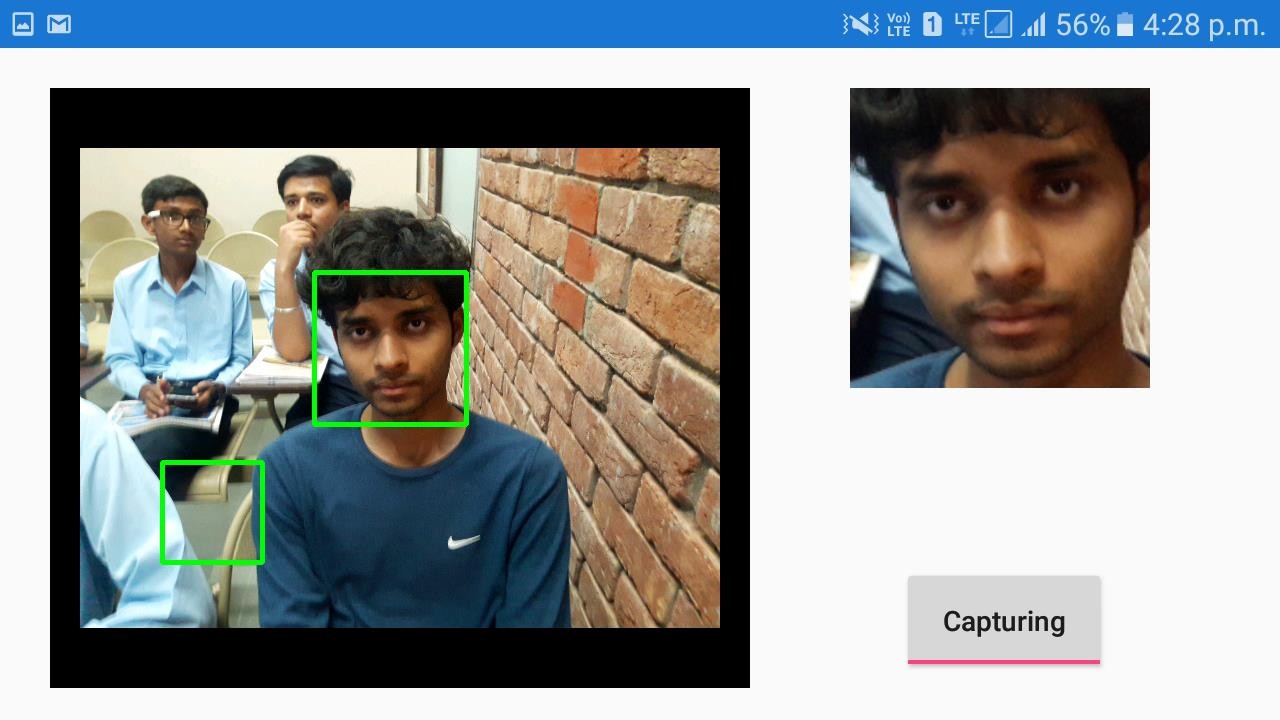
### Performing the face recognition:

In this step, the algorithm is already trained. Each histogram created is used to represent each image from the training dataset. So, given an input picture, we perform the steps again for this new image and creates a histogram of this image.

1. So to find the image that matches the given image. We just have to compare two histograms and return the face with the closest histogram.
2. We can use several methods to compare the two histograms (calculate the distance between two histograms).
3. So the algorithm output is the ID from the image with the closest histogram. The algorithm should also return the calculated distance, which can be used as a ‘confidence’ measurement, Lower confidences are superior because it means the distance between the two histograms is closer.
4. We can then use a threshold and the ‘confidence’ to automatically estimate if the algorithm has correctly recognized the image. We can assume that the algorithm has successfully recognized the face if the confidence is lower than the threshold defined

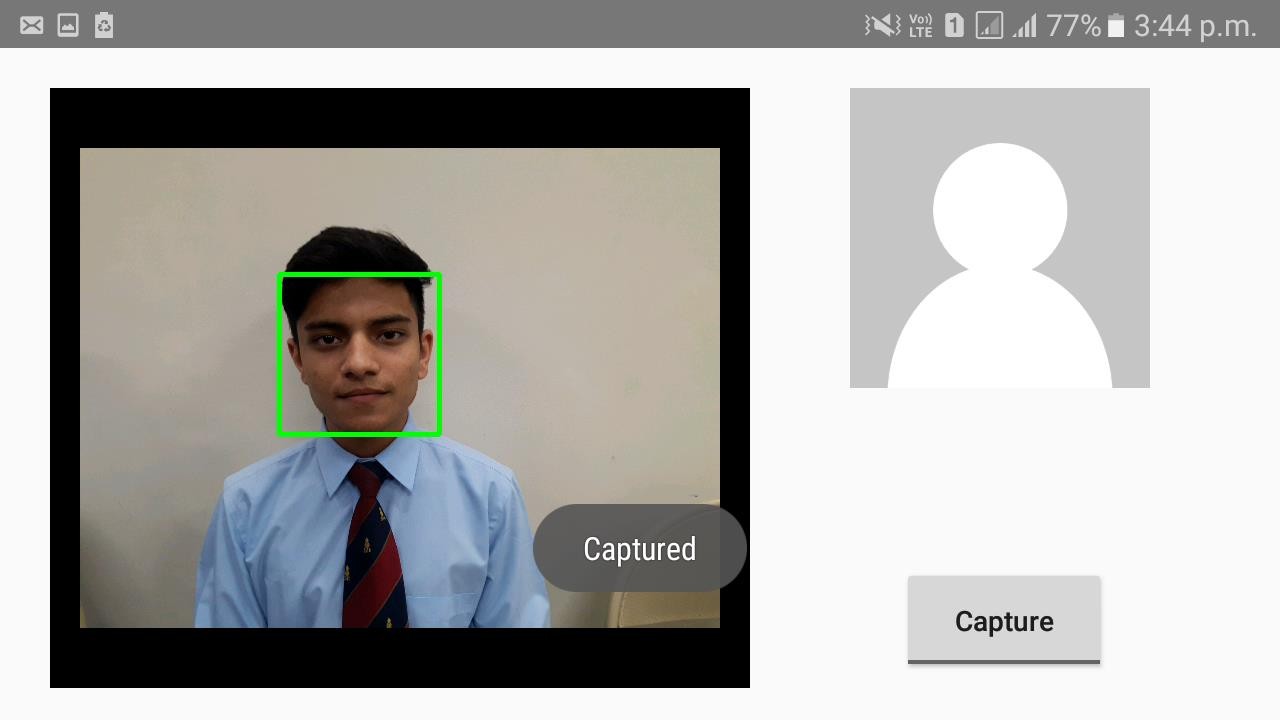
Figure Shows the process of capturing of the face after clicking on the capture button. Capturing the face which is inside green Rectangle.





Capturing the face which is inside green rectangle.

Figure Shows that the face is captured which is inside the green rectangle and the image is stored.



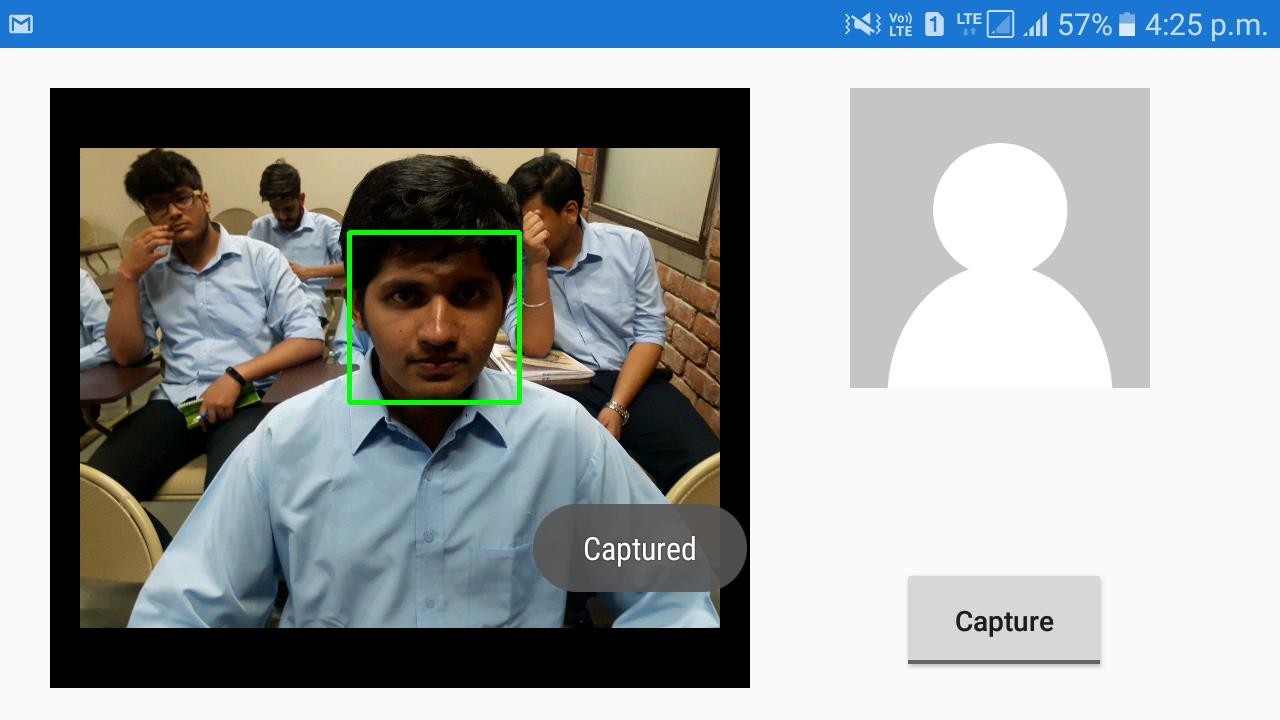


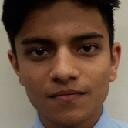
Image inside green rectangle is captured and stored.

Generation of Test Image’s for the training the system.





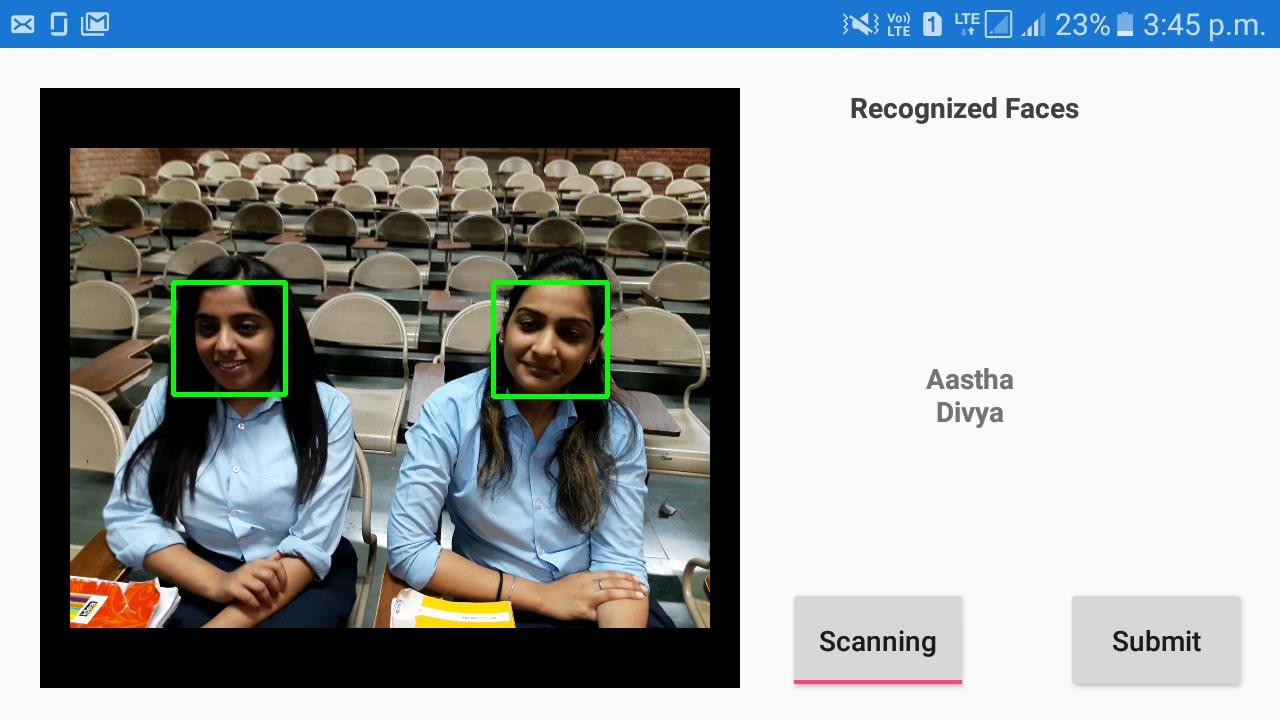
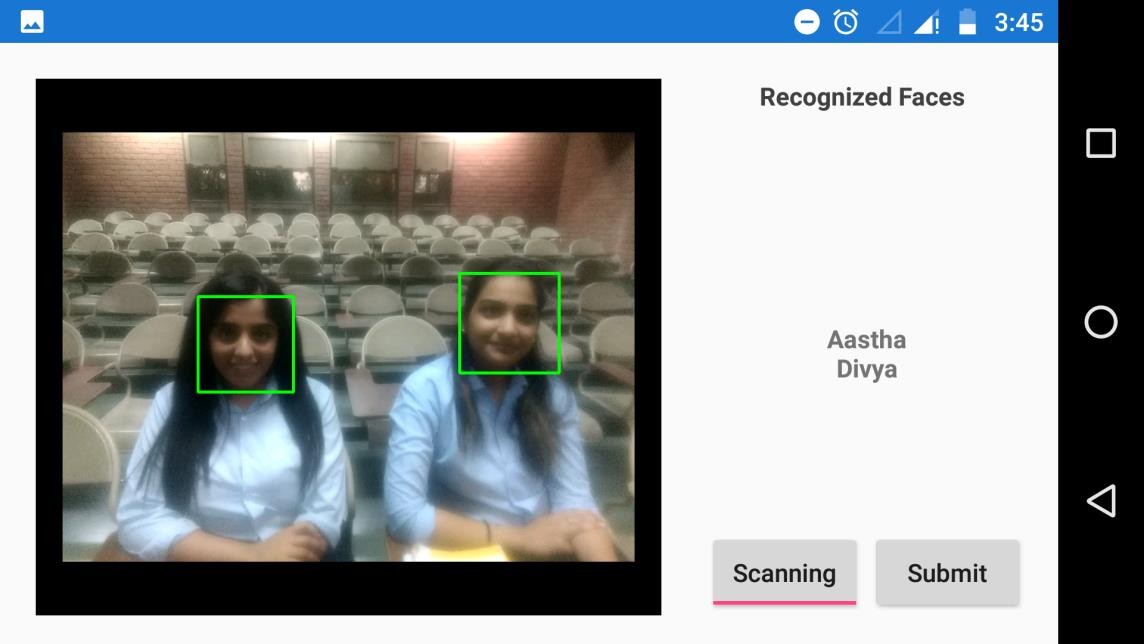




Training the System with the test image.

# Recognizing Faces

Figure shows the face recognition of two students when they are looking in the camera and when they are not looking into the camera.



Face Recognition of two Students

Figure shows the face recognition of three students when all of them are in a single row (Front row) and when they are in different row (two at front and one at second row).

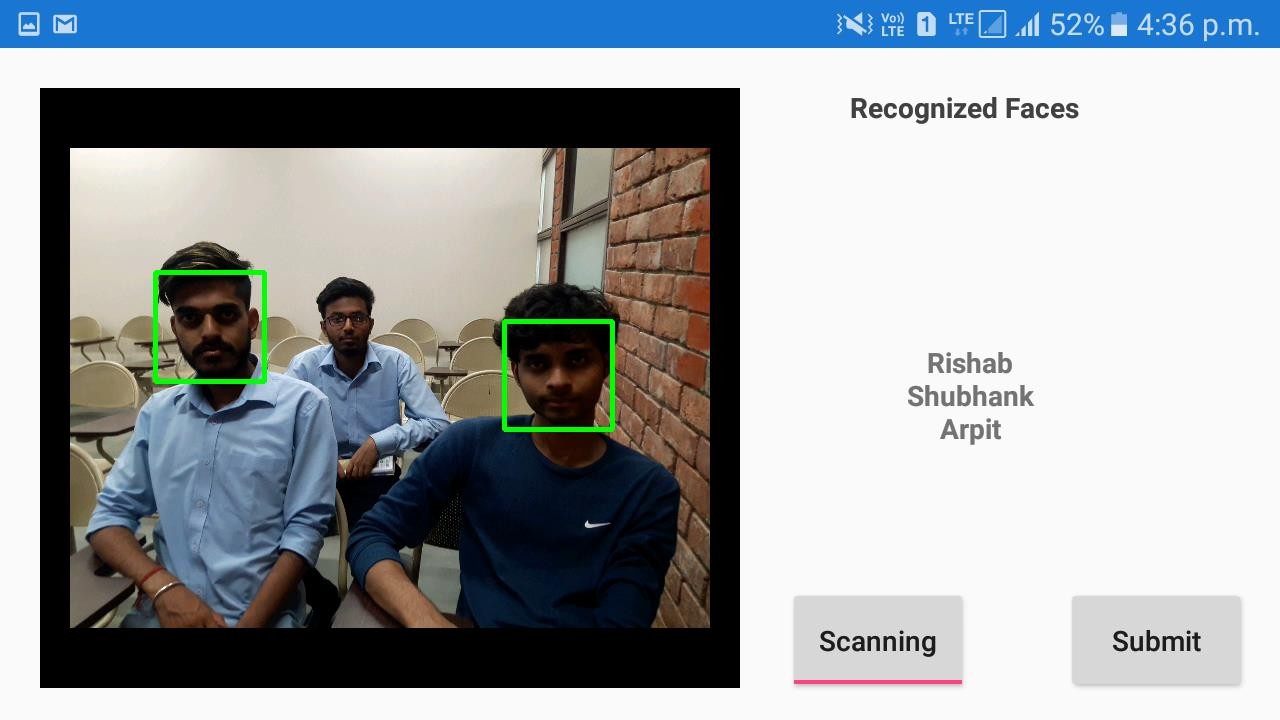
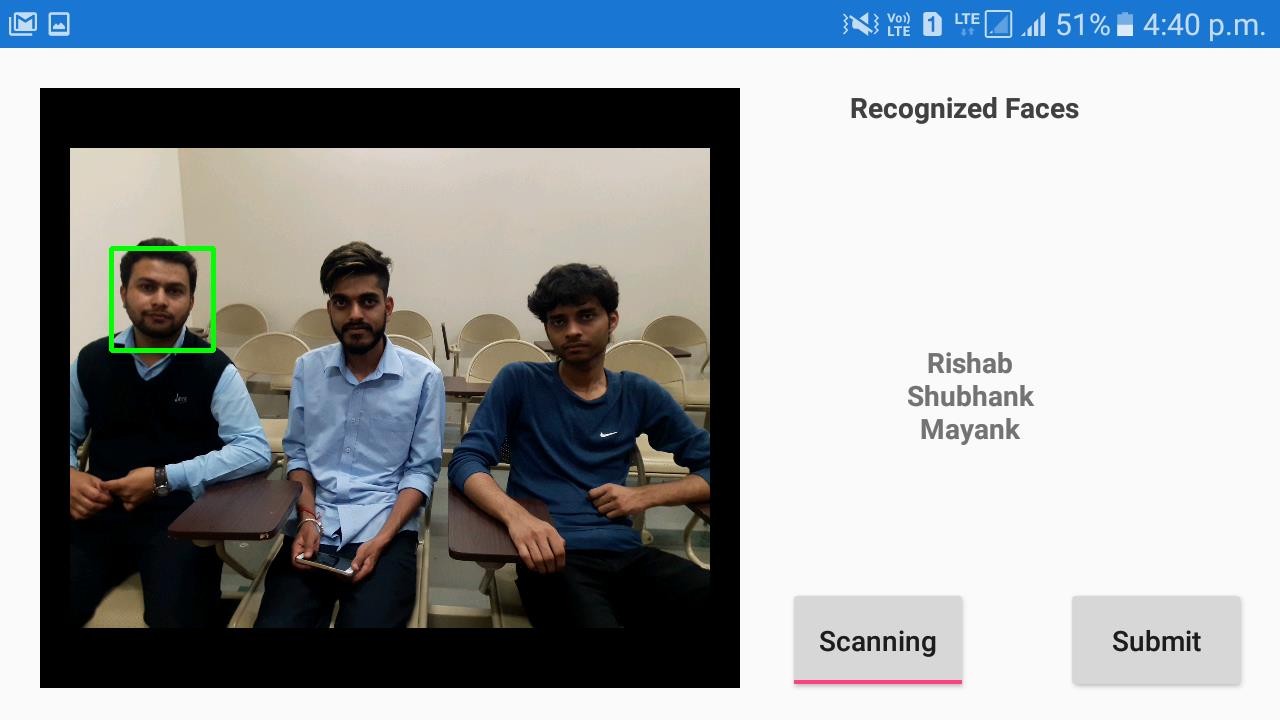


Figure 4.2.3: Face Recognition of Three Students.

Successfully detected and recognize the faces of three students.

Figure shows the face recognition of four students when all of them are in a single row (Front row) and when they are in different row (two at front and two at second row) but under different lighting conditions.

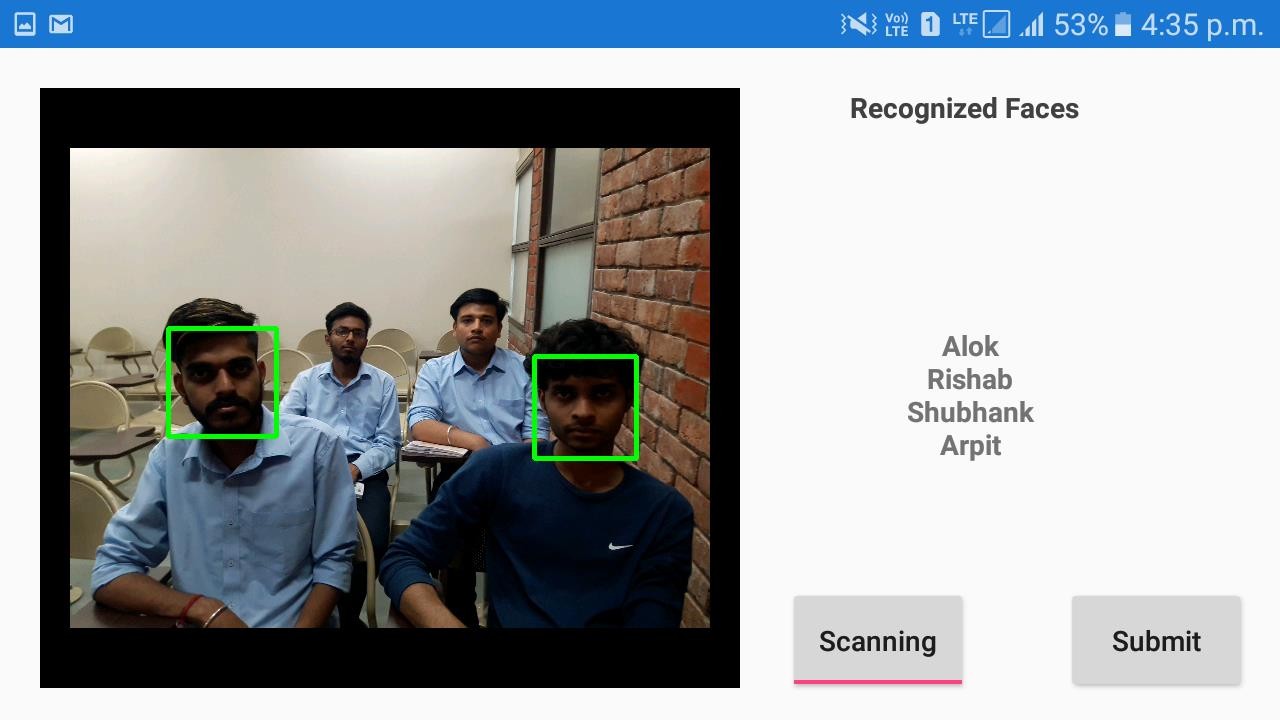
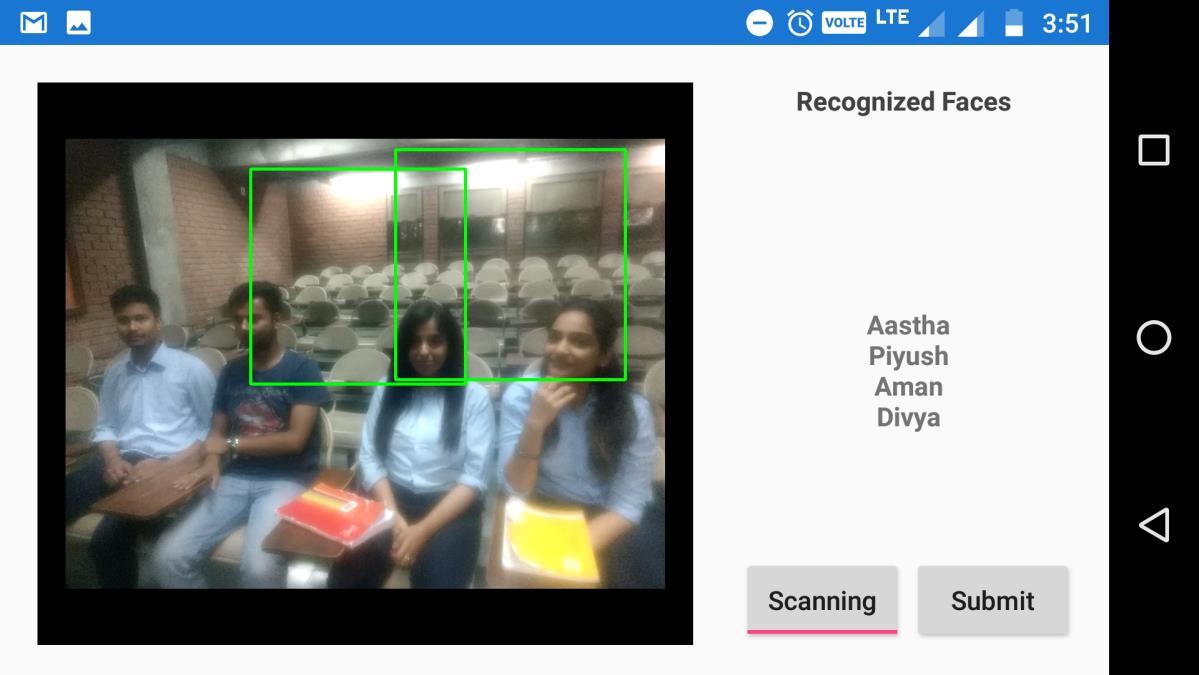


Figure 4.2.4: Face Recognition of four Students.

Successfully detected and recognize the faces of four students.

Figures shows the face recognition of five students and seven students when few seats are vacant (in front and second row) and when all seats are occupied vacant (in front and second row).

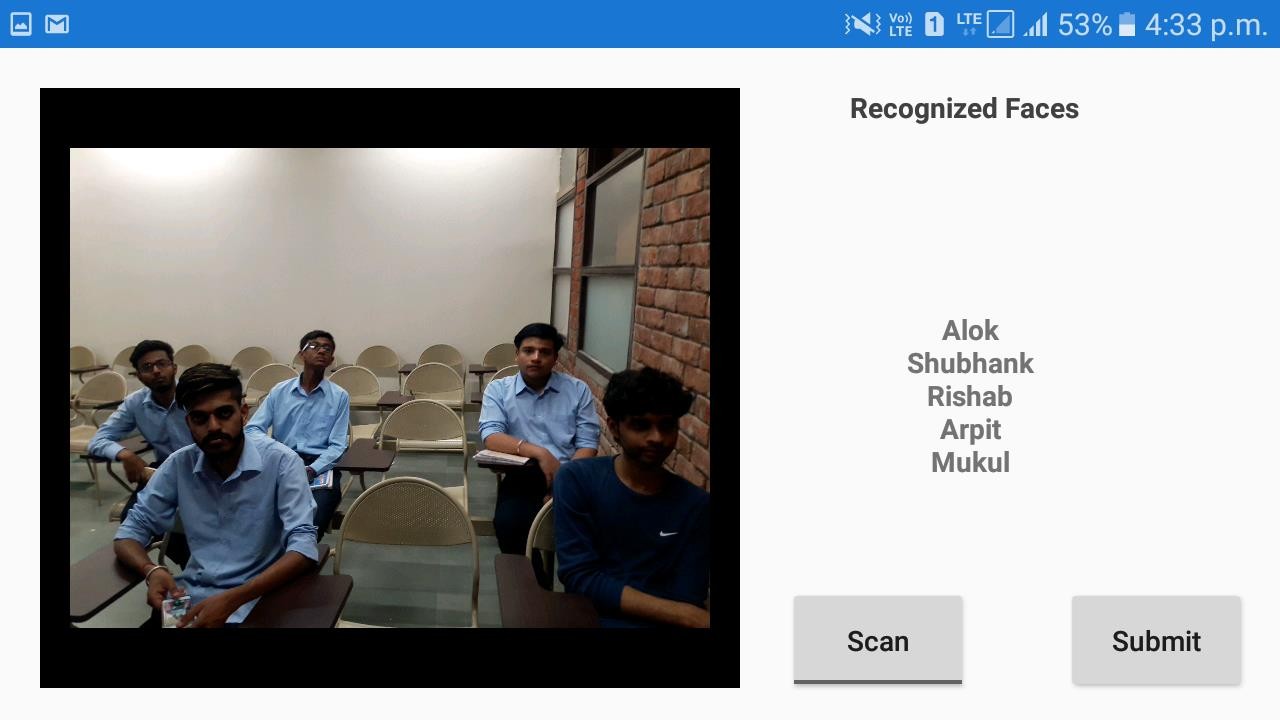


Figure 4.2.5: Face Recognition of five Students.

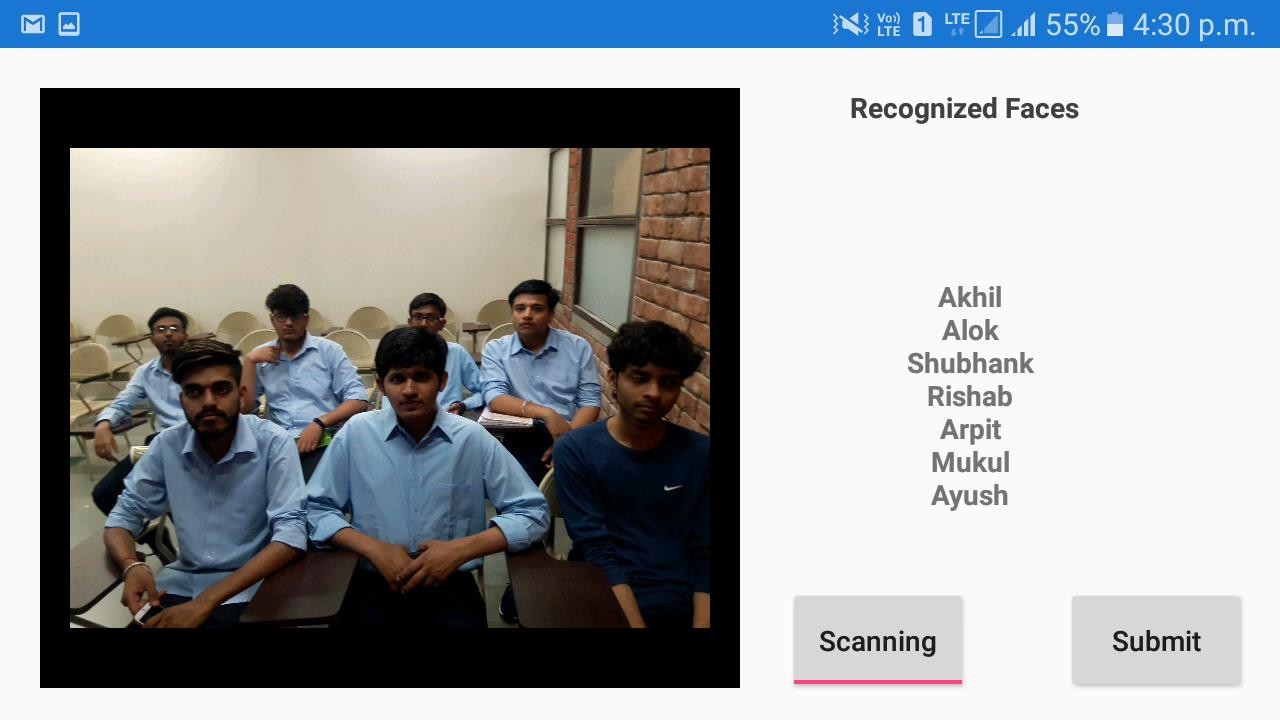
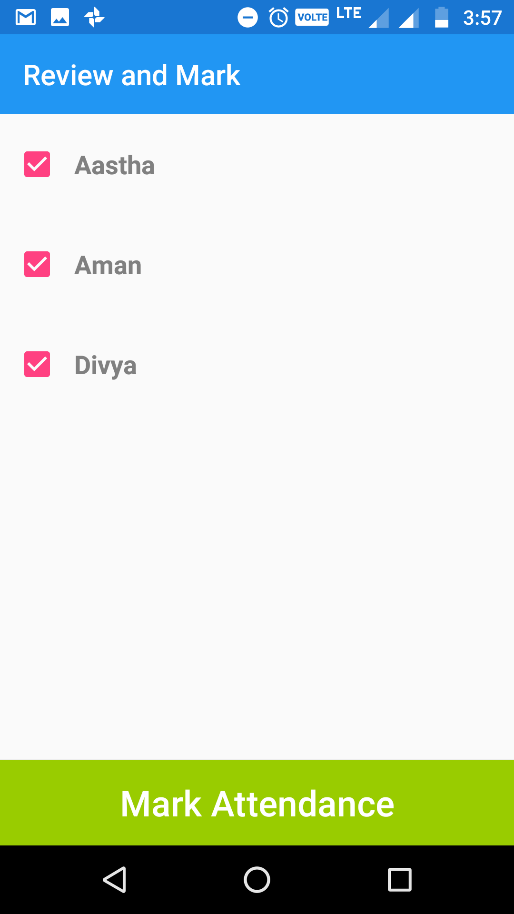
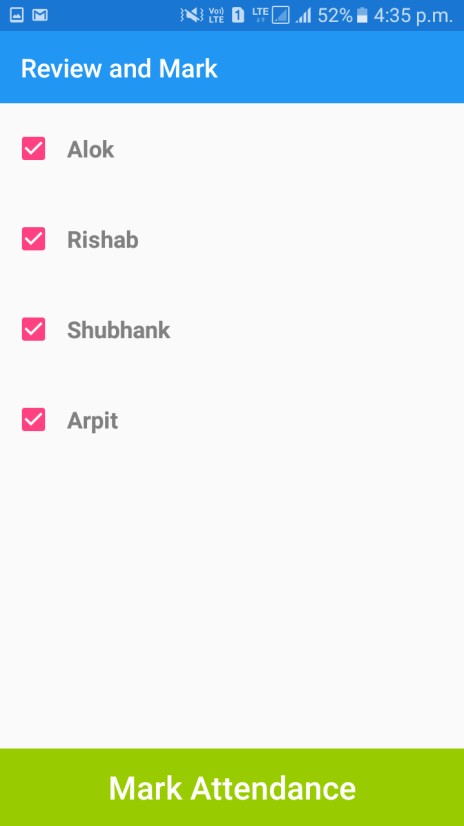
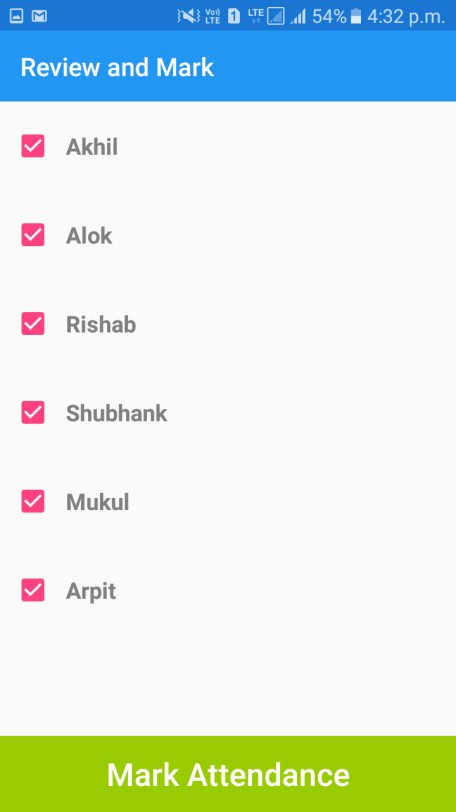
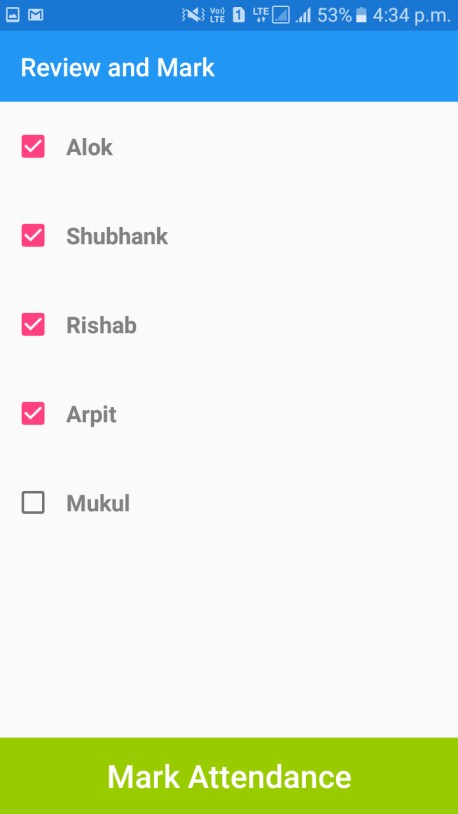


Figure 4.2.6: Face Recognition of Seven Students.

Successfully detected and recognize the faces of five and seven students.



shows the Marking of attendance and reviewing it.

Table shows the number of face detected and recognized ant their detection and recognition rate.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. of students | No of face  detected | Detection Rate | No. of face  Recognize | Recognition  rate |
| 1 | 1 | 100% | 1 | 100% |
| 2 | 2 | 100% | 2 | 100% |
| 3 | 3 | 100% | 3 | 100% |
| 4 | 4 | 100% | 4 | 100% |
| 5 | 5 | 100% | 5 | 100% |
| 6 | 6 | 100% | 6 | 100% |
| 7 | 7 | 100% | 7 | 100% |

No of face detected and recognized

# Chapter 5 Conclusion

# Conclusions

In this project, we studied various algorithms for the implemention of face recognition system in mobile phones. Eigenfaces machine-learning algorithm was the engine of training the system after applying some filters on the image. Furthermore, the Eigenfaces algorithm allows the application to recognize the face realtime. Eigenfaces was not very sensitive to a change in the number of subjects during the first phase, however, an increase in size of the training set helped the algorithm to correct its wrong prediction. An increase in the data set did not help to recognize more subjects, but it turned correct predictions into wrong ones. Eigenface was not accurate in the second phase.

Fisherfaces had better results in the first phase with a larger amount of data. However, its behavior was different in the second phase. Sometimes with 20 pictures, the results were better, but with 40 pictures the results were the same or worse. This algorithm was most of the time the worst in the two phases. It also had a very low accuracy in the second phase.

A face is detected using the Local binary pattern cascade classifier. After testing the Haar-like cascade classifier, the speed of the detection was very low as compared to LBP that always had at most 96 % in the first phase and is comparatively better algorithm. In the second phase, this algorithm had a consequent drop in its accuracy compared to the first phase. An increase in the number ofesubjects dramatically changed its prediction. In each phase, an increase in the training data had a positive effect or no effect.

We have identified various issues in our face recognition systems:-

* + - Illumination Problem: - The illumination problem is that where the same face appears different due to a change in lighting. The changes induced by illuminatio are often larger than the differences between individuals, causing systems based on comparing images to misclassify input images.
    - The Pose Problem in Face Recognition: - It is not surprising that the performance of face recognition systems drops significantly when large pose variations are present in the input images. When illumination variation is also present, the task of face recognition becomes even more difficult. Here we focus on the out-of- plane rotation problem, since in-plane rotation is a pure 2D problem and can be solved much more easily.
    - Distance Problem: - Our face recognition system fails if the distance between the camera and the face exceeds more.
    - Camera Problem: - Resolution of a camera plays a very important role in recognizing the image. Better the camera, more accurately it will recognize the face. For example, we have tested our face recognition system with two phones and we have found that the phone with high resolution camera does the better than the other.

# Future Scope

As part of the future work, we would like to develop an application that would allow the user to add or delete face classes in the training set. This would give users the freedom to define their own user groups rather than a pre-defined set on the server. We would also like to explore better algorithms for face detection and face recognition to increase the number of student to be detected and recognize.

References

1. Viola, Paul, and Michael J. Jones. "Robust real-time face detection." International journal of computer vision 57.2 (2004): 137-154.
2. Ahonen, T., Hadid, A. and Pietikainen, M., 2006. Face description with local binary patterns: Application to face recognition. IEEE transactions on pattern analysis and machine intelligence, 28(12), pp.2037-2041.
3. Alrashed, H.H., 2016. Detecting live person for the face recognition problem: submitted in partial fulfilment of the requirements for the degree of Master of Information Sciences, Massey University (Doctoral dissertation, Massey University).
4. Baggio, D. L., Emami, S., Escriva, D. M., Ievgen, K., Mahmood, N.,Saragih, J., et al. (2012). Mastering openCV with practical computer vision projects: Step-by-step tutorials to solve common real-world computer vision problems for desktop or mobile, from augmented reality and number plate recognition to face recognition and 3D
5. https://en.wikipedia.org/wiki/Viola%E2%80%93Jones\_object\_detection\_framework
6. T. Ojala, M. Pietika¨inen, and D. Harwood, “A Comparative Study of Texture Measures with Classification Based on Feature Distributions,” Pattern Recognition, vol. 29, no. 1, pp. 51-59, 1996.
7. T. Ojala, M. Pietika¨inen, and T. Ma¨enpa¨a¨, “Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns,” IEEE Trans. Patt
8. L. Wang and D. He, Texture classification using texture spectrum, Pattern Recognition, 23(8)905-910, 1990.
9. Chang-Yeon, J., 2008. Face Detection using LBP features. Final Project Report, 77.
10. Abrah&atildeo, W., Oliveira, G., Salgueiro, L., Diaz, D.H., Gomez, M.A. and Barbosa, J., A comparison of Haar-like, LBP and HOG approaches to concrete and asphalt runway detection in high resolution imagery.
11. docs.opencv.org/2.4/modules/contrib/doc/fa cerec/facerec\_tutorial.html
12. Sarabjit Singh, AmritpalKaur, Taqdir, A Face Recognition Technique using Local Binary Pattern Method, International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 3, March 2015.
13. https:/[/www.tutorialspoint.com/android/android\_overview.htm](http://www.tutorialspoint.com/android/android_overview.htm)
14. Ortega-Garcia, J., Bigun, J., Reynolds, D. and Gonzalez-Rodriguez, J., 2004. Authentication gets personal with biometrics. *IEEE signal processing magazine*, *21*(2), pp.50-62.
15. Delbiaggio, N., 2017. A comparison of facial recognition’s algorithms.
16. Zhao, W., Chellappa, R., Phillips, P.J. and Rosenfeld, A., 2003. Face recognition: A literature survey. *ACM computing surveys (CSUR)*, *35*(4), pp.399-458.
17. Pandey, S., Singh, K., Wadkar, M. and Vadalkar, H., SMART APPLICATION FOR ATTENDANCE MARKING SYSTEM USING FACIAL RECOGNITION.
18. Kar, N., Debbarma, M.K., Saha, A. and Pal, D.R., 2012. Study of implementing automated attendance system using face recognition technique. *International Journal of computer and communication engineering*, *1*(2), p.100.
19. Joseph, J. and Zacharia, K.P., 2013. Automatic attendance management system using face recognition. *International Journal of Science and research*, *2*(11), pp.328-330.