“Attendance Management Tool based on Face recognition.”

Submitted in partial fulfillment of the requirements

of the degree of

Bachelor of Technology

Submitted by

Mr. ANIRUDDHA JOSHI

Mr. AASHAY MAHAJAN

Mr. ADITYA PUNDKAR

Under valuable guidance of

Prof. G.S. Malande

(Dept. of Computer Science and Eng.)

Department of Computer Science and Engineering,
Shri Guru Gobind Singhji Institute of Engineering & Technology,
Vishnupuri, Nanded, Maharashtra, India, 431606.
2019-20

DECLARATION

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

ANIRUDDHA JOSHI
(2016BCS002)

AASHAY MAHAJAN
(2016BCS156)

ADITYA PUNDKAR
(2016BCS027)

Date: 15/08/2020

CERTIFICATE

This is to certify that the report entitled **"Attendance Management Tool based on face recognition"** *being submitted by* **Aniruddha Joshi(2016bcs002), Aashay Mahajan(2016bcs156), Aditya Pundkar(2016bcs027)** *to* **Shri Guru Gobind Singhji Institute of Engineering and Technology, Vishnupuri, Nanded (M.S.), India,** *as partial fulfillment for the award of the degree of* **Bachelor of Technology** *in* **Electronics and Telecommunication Engineering**, *is a record of bonafide work carried out by him under our supervision and guidance. The matter contained in this report has not been submitted to any other university for the award of any degree or diploma.*

Prof. G.S.Malande	**Prof. S.D.Wattamwar**	**Prof. S.S.Hatkar**
(Guide)	**(Project co-ordinator)**	**(H.O.D)**

Dr.Y.V.Joshi

(Director,SGGSIE&T,Nanded)

ACKNOWLEDGEMENT

We would like to express our gratitude to all those who helped me to complete this work. We want to thank my guide **Mr. G.S. Malande** for his continuous help and generous assistance. He helped in a broad range of issues from giving me direction, helping to find the solutions, outlining the requirements and always having the time to seem.

We have furthermore to thank **Prof. S.S. Hatkar**, Head of the Department of Computer Science & Engineering, to encourage us to go ahead and for continuous guidance.

We would like to thank my colleagues who helped us time to time from preparing report and giving good suggestions. We also extend sincere thanks to all the staff members of Department of Computer Science & Engineering for helping me in various aspects.

Last but not least we are grateful to our parents for all their support and encouragement.

Aniruddha Joshi

Aashay Mahajan

Aditya Pundkar

ABSTRACT

Development of artificial intelligence in recent years, facial recognition gains more and more attention. Compared with the traditional card recognition, fingerprint recognition and iris recognition, face recognition has many advantages, including but limit to non-contact, high concurrency, and user friendly. It has high potential to be used in government, public facilities, security, e-commerce, retailing, education and many other fields.

In this project we propose an automated attendance management system. This system, which is an desktop application based on face detection and recognition algorithms, automatically detects the student when he enters the class room and marks the attendance by recognizing him. The system architecture and algorithms used in each stage are described in this project. Different real time scenarios are considered to evaluate the performance of various face recognition systems. This project also proposes the techniques to be used in order to handle the threats like spoofing. When compared to traditional attendance marking this system saves the time and also helps to monitor the students.

Recognition technique is one of the most efficient bio metric technique for identification of people. We can utilize it in the field of education for managing the attendance of students. There are a lots of colleges and schools in which thousands of students are taking the education. In every classroom there are about ninety to hundred students are studying. Also in every few days, a new school or college is opened. To maintain the attendance and records of these so many numbers of students is a very difficult task.In a classroom with large number of students, it is a very tedious and time consuming task to take the attendance manually. Therefore we can implement an effective system which will mark the attendance of students automatically by recognizing their faces.

CONTENTS

List of Figures

1 Introduction

1.1 Introduction

Recognition technique is one of the most efficient bio metric technique for identification of people. We can utilize it in the field of education for managing the attendance of students. There are a lots of colleges and schools in which thousands of students are taking the education. In every classroom there are about ninety to hundred students are studying. Also in every few days, a new school or college is opened. To maintain the attendance and records of these so many numbers of students is a very difficult task.In a classroom with large number of students, it is a very tedious and time consuming task to take the attendance manually. Therefore we can implement an effective system which will mark the attendance of students automatically by recognizing their faces.

In recent research areas point of view faces in images and videos have easily identified and localized. So to propose a fully automated system we need an efficient and robust face detection method. In given class to find position and size of object we are going for many robust techniques. There are many challenges such as size color shape and texture of human face.

1.2 Necessity

1.2.1 Increasing no. of students

In recent times, there are myriad of schools containing tens ofthousand of students. According to the estimated growth these numbers will sky rocket soon. And needless to say, managing and surveilling large numbers of students will be a lot of work for the education sector. Thus an automated attendance management at the primary level is a great way to start the revolution in a surveillance of large number of students. In this we propose a system that will guarantee that, no matter how much the pupils, handling them will be a piece of cake.

1.2.2 Limiting human contact

In this dire circumstances, where we are facing the Coronavirus pandemic, we need to limit human contact to minimum as possible. Naturally it is obvious that

fingerprint based bio metric identification machines are absolutely out of the question. Similarly, the attendance page that is passes across the classroom is also not sanitary. We may consider other option like retina scanner. But they are costly and time consuming. They can be used on a small scale but using them on a large scale its next to impossible. They can't be used extensively. Thus, the proposed Attendance Management System is a rational and wise choice.

With the rapid development of artificial intelligence in recent years, facial recognition gains more and more attention. Compared with the traditional card recognition, fingerprint recognition and iris recognition, face recognition has many advantages, including but limit to non-contact, high concurrency, and user friendly. It has high potential to be used in government, public facilities, security, e-commerce, retailing, education and many other fields.

1.2.3 Disciplinary Issues

Its been a prevalent problem across the education system regarding the attendance. The central board of education had to take particular and dire measures to compel student to take part in studies and their academic initiatives. But still this issue is been boiling with fake attendance, or tampering the attendance sheet. Teacher has to be on their toes and keep a keen eye on the students to prevent this kind of untoward behavior. This kind of applications avoid the problem the system has had for such a long time. Student will have to be present physically in the class to log the attendance. And it may be a long shot, but excitement of something new shall even help boast their morale up.

1.3 Existing Work

At present attendance marking involves manual attendance on paper sheet by professors and teachers. but it is very time consuming process and chances of proxy is also one problem that arises in such type of attendance marking. also there are attendance marking system such as RFID, Biometrics etc. but these systems are currently not so much popular in schools and classrooms for students as they have their own advantages and disadvantages. The Problems with Current System The

problem with this approach in which manually taking and maintains the attendance records is that it is very inconvenient task. Traditionally, students attendances are taken manually by using attendance sheet given by the faculty members in class, which is a time consuming event Moreover, it is very difficult to verify one by one student in a large classroom environment with distributed branches whether the authenticated students are actually responding or not the ability to compute the attendance percentage becomes a major task as manual computation produces error and also wastes a lots of time this method could easily allows for impersonation and the attendance sheet could be stolen or lost.

Another bio metric attendance system based on fingerprint recognition is widely used especially in corporate offices. Although they are more secure as compared to the current system, there too are contact based systems.

1.4 Proposed Work

Traditional student attendance marking technique is often facing a lot of trouble. The face recognition student attendance system emphasis its simplicity by eliminating classical student attendance marking technique such as calling student names or checking respective identification cards. There are not only disturbing the teaching process but also causes distraction for students during exam sessions. Apart from calling names, attendance sheet is passed around the classroom during the lecture sessions. The lecture class especially the class with a large number of students might find it difficult to have the attendance sheet being passed around the class. Thus, face recognition student attendance system is proposed in order to replace the manual signing of the presence of students which are burdensome and causes students get distracted in order to sign for their attendance. Using Local Binary Patterns Histograms (LBPH) algorithm in Open CV the student's faces are trained and recognized. Python automates the tasks by providing for the execution of the programs in Computer Vision and GUI of the system along with managing the database of the student attendance.

Furthermore, the face recognition based automated student attendance system able to overcome the problem of fraudulent approach and lecturers does not have to count the number of students several times to ensure the presence of the students.

2 Literature survey

2. Literature Survey

Ever since IBM introduced first personal computer on 1981, to the ".com" era in the early 2000s, to the online shopping trends in last 10 years, and the Internet of Things today, computers and information technologies are rapidly integrating into everyday human life. As the digital world and real world merge more and more together, how to accurately, securely and effectively capture and store users data has become an important research topic. The attendance management system is no exception to this digitalization trend. More so due to the fact that its advantages over the current system are overwhelming.

Traditional identity recognition technology mainly relied on the foreign objects (ID card, individual's signature, etc.). However, as explained in the previous section, these techniques are vulnerable to spoofing or replication. As a result, the identity can be "stolen" or more appropriately put impersonated by others. Another method proposed was bio-metrics ie. fingerprint scanning, retina scan, etc which owing to their efficiency and reliability took over the existing system quickly. Bio-metric fingerprint scanning is perhaps the most widely used. However, considering the current circumstances where the entire world is battling with the Coronavirus pandemic which spreads by contact, it is safe to say that bio-metric fingerprint scanning technique could be a little unhygienic.

Different from the traditional identity recognition technology, bio-metrics is the use of the inherent characteristics of the body for identification, such as fingerprints, irises, face and so on. Compared with the traditional identity recognition technology, biological features have many advantages, as:

1. Reproducibility, biological characteristics are born with, cannot be changed, so it is impossible to copy other people's biological characteristics.

2. Availability, biological features as part of the human body, readily available, and will never be forgotten.

3. Easy to use.

Many biological characteristics will not require individuals to corporate with the examine device. Based on the above advantages, bio-metrics has attracted the attention of major corporations and research institutes and has successfully replaced traditional recognition technologies in many fields. And with the rapid development of computer and artificial intelligence, bio-metrics technology is easy to cooperate with computers and networks to realize automation management, and is rapidly integrating into people's daily life.

In recent research areas' point of view faces in images and videos have easily identified and localized. So to propose a fully automated system we need an efficient and robust face detection method. In given class to find position and size of object we are going for many robust techniques. There are many challenges such as size color shape and texture of human face.

2.1 Approach

Face recognition can be traced back to the sixties and seventies of the last century, and after decades of twists and turns of development has matured. The traditional face detection method relies mainly on the structural features of the face and the color characteristics of the face.

Some traditional face recognition algorithms identify facial features by extracting landmarks, or features, from an image of the subject's face. For example, as shown in Figure 2.1, an algorithm may analyze the relative position, size, and/or shape of the eyes, nose, cheekbones, and jaw. These features are then used to search for other images with matching features. These kinds of algorithms can be complicated, require lots of compute power, hence could be slow in performance. And they can also be inaccurate when the faces show clear emotional expressions, since the size and position of the landmarks can be altered significantly in such circumstance.

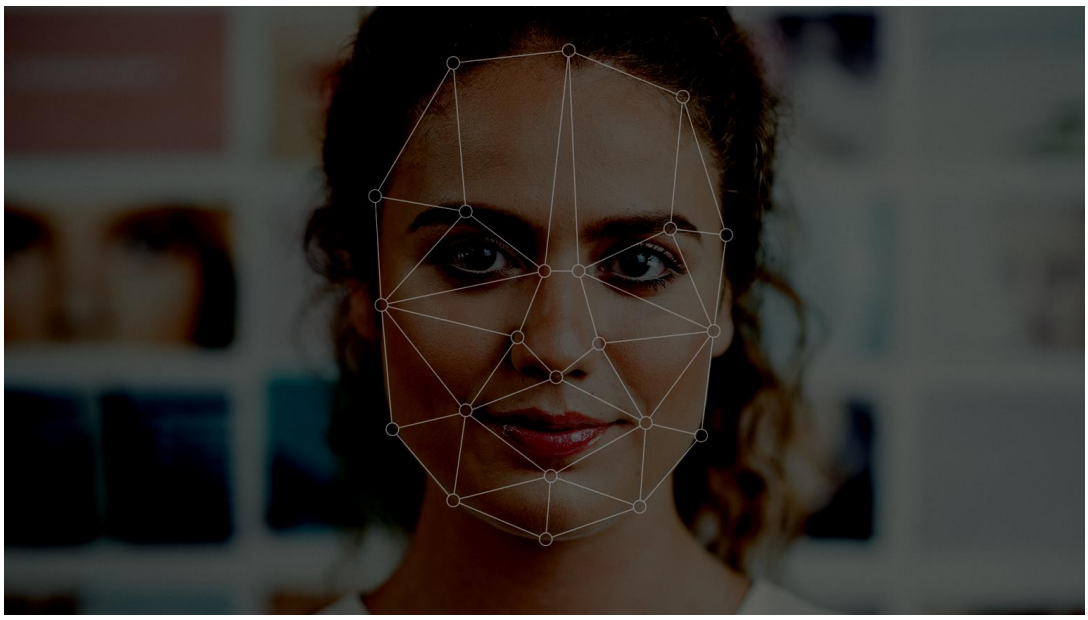


Fig2.1 Identifying facial features

A complete face recognition system includes face detection, face preprocessing and face recognition processes. Therefore, it is necessary to extract the face region from the face detection process and separate the face from the background pattern, which provides the basis for the subsequent extraction of the face difference features. The recent rise of the face based on the depth of learning detection methods, compared to the traditional method not only shorten the time, and the accuracy is effectively improved. Face recognition of the separated faces is a process of feature extraction and contrast identification of the normalized face images in order to obtain the identity of human faces in the images.

2.2 Analysis of related work

Until now we are introduced the facial recognition and we discussed the use case and bright future of this technology. A tremendous amount of research and effort from many major company and universities and been dedicated to this field. In the first part of this chapter, we will review the most significant work in the facial recognition field.

With the rapid increase of computational powers and accessibility of innovative sensing, analysis and rendering equipment and technologies, computers are becoming more and more intelligent. This along with considerable amount or research and literature available, the field of face detection has made considerable progress in the past decade.

2.2.1 Challenges

Challenges in face detection, are the reasons which reduce the accuracy and detection rate of face detection. These challenges are complex background, too many faces in images, odd expressions, illuminations, less resolution, face occlusion, skin color, distance and orientation etc. (Figure 3).

• Odd expressions Human face in an image may have odd expressions unlike normal, which is challenge for face detection.

• Face occlusion Face occlusion is hiding face by any object. It may be glasses, scarf, hand, hairs, hats and any other object etc. It also reduces the face detection rate.

• Illuminations Lighting effects may not be uniform in the image. Some part of the image may have very high illumination and other may have very low illumination.

• Complex background Complex background means a lot of objects presents in the image, which reduces the accuracy and rate of face detection.

• Too many faces in the image It means image contains too many human faces, which is challenge for face detection.

• Less resolution Resolution of image may be very poor, which is also challenging for face detection.

• Skin color Skin-color changes with geographical locations. Skin color of Chinese is different from African and skin-color of African is different from American and so on. Changing skin-color is also challenging for face detection**.**

• Distance Too much distance between camera and human face may reduce the detection rate of human faces in image.

• Orientation Face orientation is the pose of face with an angle. It also reduces the accuracy and detection rate of face detection.

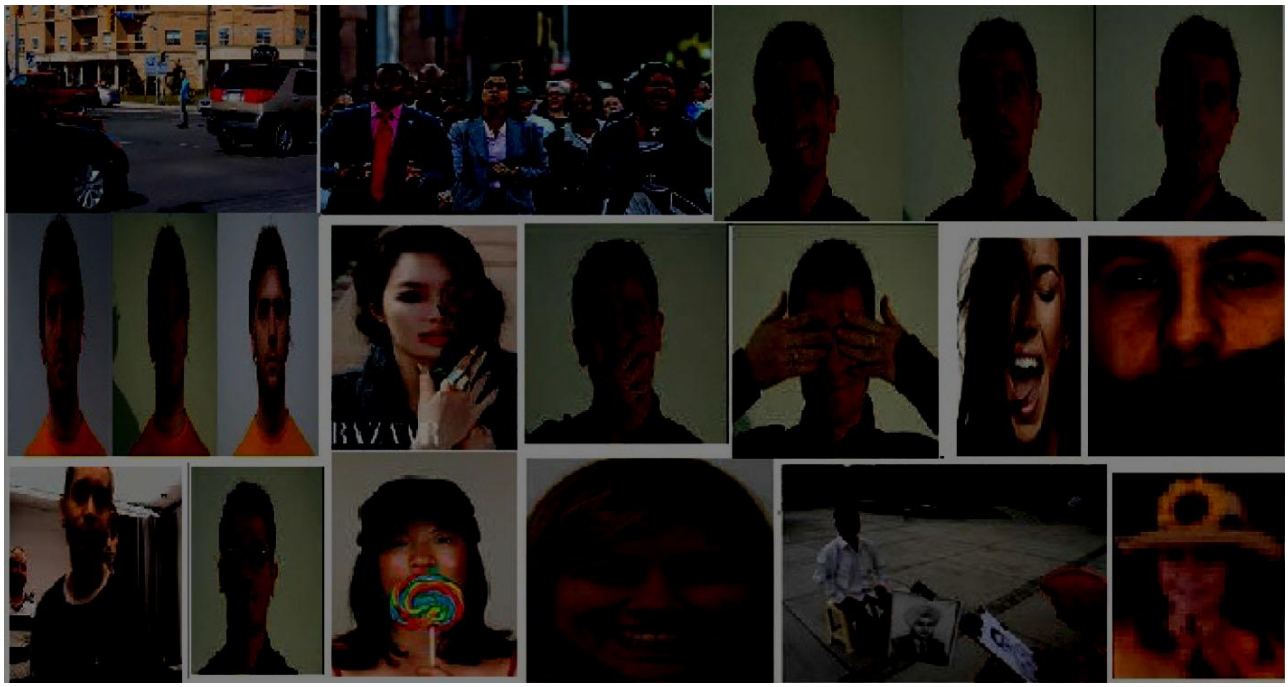


Fig 2.2 Challenges

2.3 Haar Cascade classifier

The Haar Cascade facial identification system is a technology capable of identifying a face of a person from a digital image or a video frame from a video source. It is an important part of early stages as discussed because it is capable of successfully narrow down and extract a facial from given input image which in turn will be fed to the latter stage involving face recognition.

Haar Cascade classifier is based on the Haar Wavelet technique to analyze pixels in the image into squares by function. This uses "integral image" concepts to compute the "features" detected. Haar Cascades uses the Ada-boost learning algorithm which selects a small number of important features from a large set to give an efficient result of classifiers then use cascading techniques to detect the face in an image.

2.3.1 Haar Features Extraction Phase

Haar features are similar to these convolution kernels which are used to detect the presence of that feature in the given image. Given an input image and convolution kernel, we place kernel to a corner and do convolution multiplication shifting the kernels. This method is used to detect different types of edges using different kernels.

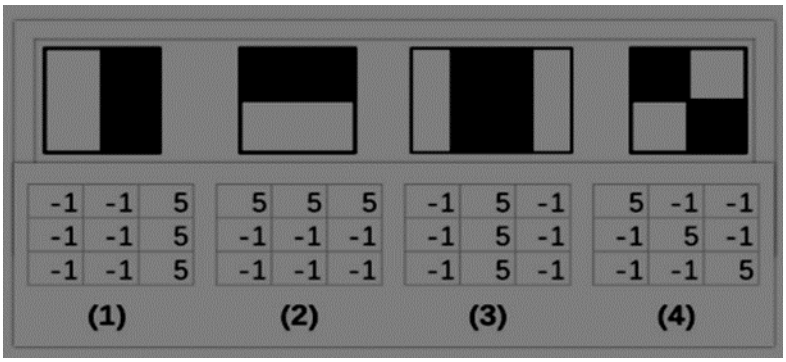


Figure 2.3 Types of kernels or Haar-features used to extract a feature

Here are some Haar-Features. The first two are “edge features”, used to detect edges. The third is a “line feature”, while the fourth is a “four rectangle feature”, most likely used to detect a slanted line. Every haar feature has some sort of resemblance to identify a part of the face.

2.3.2 AdaBoost Phase

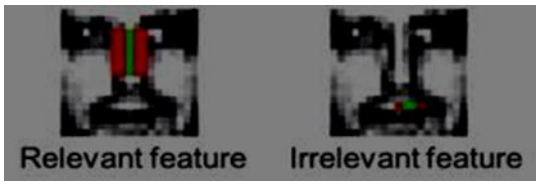


Fig 2.4 Relevant Features

There can be approximately 160,000 + feature values within a detector at 24*24 base resolution which need to be calculated. But it is to be understood that only a few set of features will be useful among all these features to identify a face. AdaBoost is used to remove redundant features and choose only relevant features. For example, feature detecting a vertical edge is useful detecting a nose but irrelevant detecting a lip.

AdaBoost is used to get the best features among all these 160,000+ features. These features are also called as weak classifiers. After these features are found a weighted combination of all these features in use in evaluating and deciding any given window has a face or not. Each of the selected features (weak classifiers) are considered okay to be included if they can at least perform better than random guessing (detects more than half the cases). Each of the weak classifiers is relevant detecting a part of the face. The output of a weak classifier is binary if it has identified a part of the face or not.

2.3.3 Cascading Phase

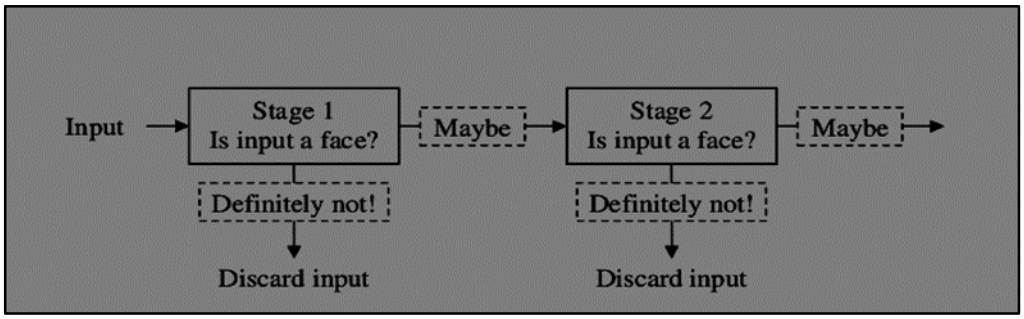


Figure 2.5 Cascading Phase

For every 24*24 window, we need to calculate 2,500 relevant features that are picked by AdaBoost from 160,000 features. Given an input image we need to move our 24*24 window all over the image and compute 2,500 features for every window and take a linear combination of all outputs and see if it exceeds a certain threshold or not. Instead of calculating 2,500 features for every window, we use the idea of cascades. We do a sampling of 2,500 features into x different cascades. Now we can detect if there is the face or not in different cascades linearly. If $cascade_i$ finds a face in an image then the image is passed to the next cascade. If no face is found is a cascade we can move to the next window. This reduces the time complexity.

- Outcome

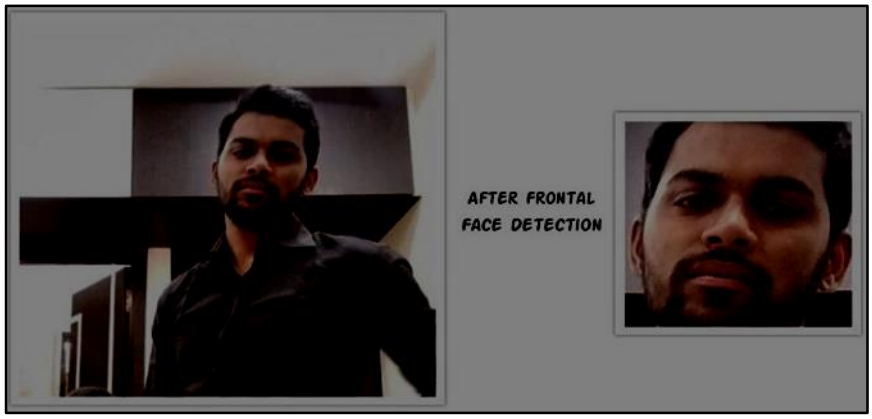


Fig 2.6 Detecting face

2.3.4 Face Detection

Face detection can consider a substantial part of face recognition operations. According to its strength to focus computational resources on the section of an image holding a face. The method of face detection in pictures is complicated because of variability present across human faces such as pose, expression, position and orientation, skin color, the presence of glasses or facial hair, differences in camera gain, lighting conditions, and image resolution.

To counter majority of the problems revolving around face recognition, we must first pre-process the image to suitable format which can be fed to our model. Pre-processing is the first and essential step for face detection. This is usually achieved through a technique called grayscaling. This also ensures that the image we feed to our model is uniform and our model does not have to make any assumptions about the image. Furthermore, to mark a person's attendance we must first identify if at all there is a face within the frame of the video capturing device. The detected face is then used by the subsequent layers for further processing.

Some types of face detection techniques are

- Knowledge Based

The knowledge-based method depends on the set of rules, and it is based on human knowledge to detect the faces. Ex- A face must have a nose, eyes, and mouth within certain distances and positions with each other. The big problem with these

methods is the difficulty in building an appropriate set of rules. There could be many false positive if the rules were too general or too detailed. This approach alone is insufficient and unable to find many faces in multiple images.

- Feature Based

The feature-based method is to locate faces by extracting structural features of the face. It is first trained as a classifier and then used to differentiate between facial and non-facial regions. The idea is to overcome the limits of our instinctive knowledge of faces. This approach divided into several steps and even photos with many faces they can report a success rate of up to 94%

- Template Matching

Template Matching method uses per-defined or parameterized face templates to locate or detect the faces by the correlation between the templates and input images. Ex- a human face can be divided into eyes, face contour, nose, and mouth. Also, a face model can be built by edges just by using edge detection method. This approach is simple to implement, but it is inadequate for face detection. However, deformable templates have been proposed to deal with these problems.

- Appearance Based

The appearance-based method depends on a set of delegate training face images to find out face models. The appearance-based approach is better than other ways of performance. In general appearance-based method rely on **techniques** from statistical analysis and machine learning to find the relevant characteristics of face images.

2.3.5 Feature Extraction

Face detection and feature extraction go hand in hand. The detected face is the mined for features by the training algorithm. There are various statistical methods widely used for face detection and feature extraction. This is usually the phase which requires a lot of computational power.

Eigen face based algorithm is a method for efficiently representing faces using Principal Component Analysis. Support Vector Machines are linear classifiers that maximize the margin between the decision hyper plane and the examples in the

training set. These are generally used to fine tune the model during training phase to reduce the margin of error after the model is deployed.

Deep Neural Networks or, to be more precise, Powerful Convolutional Neural Networks can also be used to succefully extract facial features. Such models consist of many layers where each layer is responsible for different task (one layer for low level features, another for composite features, etc.).

2.3.5 Face Localization and Recognition

The trained model is then used to search for faces in an image and if present localize them in a bounding box. This is the ultimate goal. The model is responsible for face localization which means to locate one or more faces. Bounding boxes set around the faces are considered for the next stage which is responsible for to accurately predict if a face is among the localized dataset or not. Subsequently, attendance is marked corresponding to the face identified.

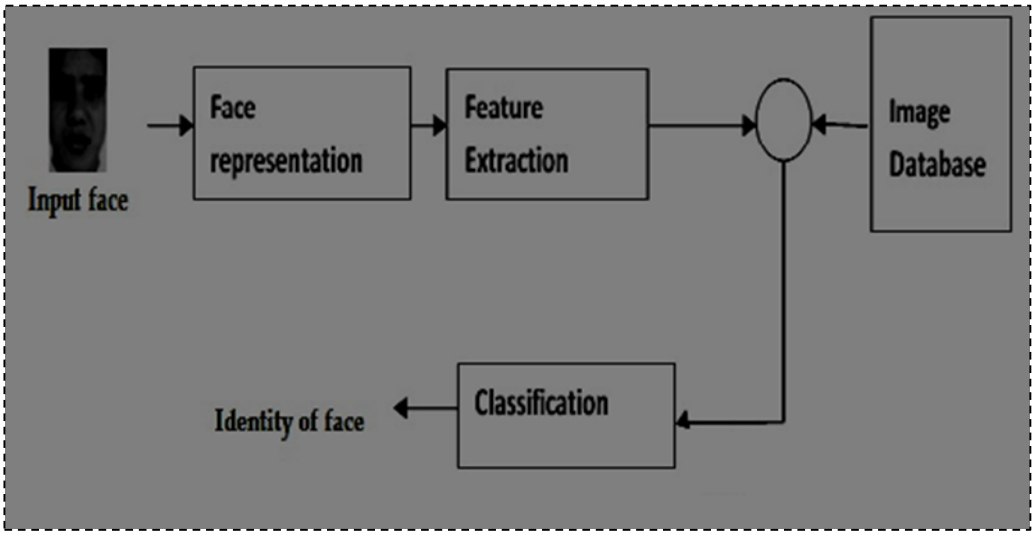


Fig.2.7 Classification process

2.4 The LBPH Algorithm

For the purpose of this application we are going to use the Local Binary Pattern Histogram(LBPH) algorithm. There exist several methods for extracting the

most useful features from (preprocessed) face images to perform face recognition. One of these feature extraction methods is the Local Binary Pattern (LBP) method. The choice of this method will be justified as we further learn more about this algorithm, its pros and cons and its reliability. This relative new approach was introduced in 1996. With LBP it is possible to describe the texture and shape of a digital image. This is done by dividing an image into several small regions from which the features are extracted (figure 1.2).

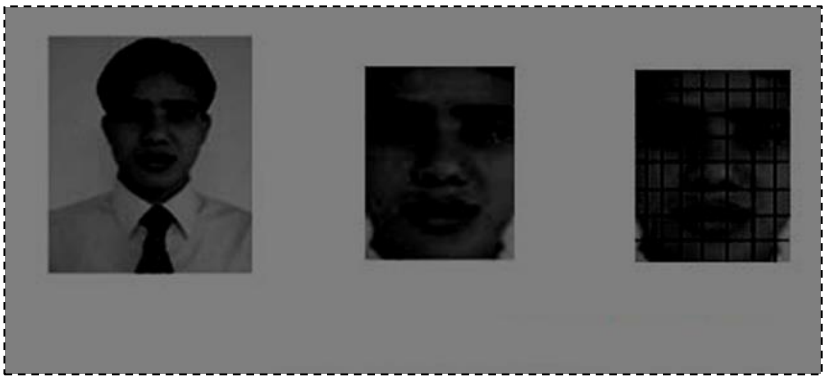


Fig2.8. Dividing faces to extract features

2.4 The LBPH Algorithm

This operator works with the eight neighbors of a pixel, using the value of this center pixel as a threshold. If a neighbor pixel has a higher gray value than the center pixel (or the the same gray value) than a one is assigned to that pixel, else it gets a zero. The LBP code for the center pixel is then produced by concatenating the eight ones or zeros to a binary code (figure 1.3).

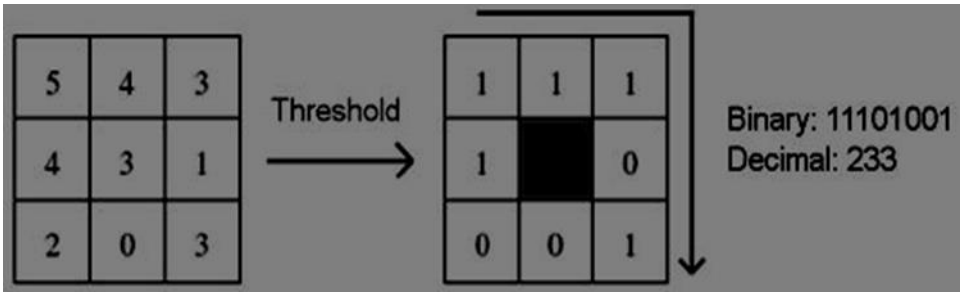


Fig.2.9 LBPH process

Later the LBP operator was extended to use neighborhoods of different sizes. In this case a circle is made with radius R from the center pixel. P sampling points on the edge of this circle are taken and compared with the value of the center pixel. To get the values of all sampling points in the neighborhood for any radius and any number of pixels, (bilinear) interpolation is necessary.

If the coordinates of the center pixel are (xc, yc) then the coordinates of his P neighbors (xp, yp). If the gray value of the center pixel is gc and the gray values of his neighbors are gp, with p = 0, ..., P − 1, then the texture T in the local neighborhood of pixel (xc, yc) can be defined as :

$$T = t(gc, g0, \ldots, gP\text{-}1)$$

Once these values of the points are obtained is it also possible do describe the texture in another way. This is done by subtracting the value of the center pixel from the values of the points on the circle.

Since t(gc) describes the overall luminance of an image, which is unrelated to the local image texture, it does not provide useful information for texture analysis. Therefore, the new texture is :

$$T\ (g_0 - g_c, \ldots, g_{P-1} - g_c\ )$$

Although invariant against gray scale shifts, the differences are affected by scaling. To achieve invariance with respect to any monotonic transformation of the gray scale, only the signs of the differences are considered. This means that in the case a point on the circle has a higher gray value than the center pixel (or the same value), a one is assigned to that point, and else it gets a zero :

$$T\ (s(g_0 - g_c), \ldots, s(g_{P-1} - g_c\ )$$

Where

$$s(x) = \begin{cases} 1, & \text{if } x>0 \\ 0, & \text{if } x<=0 \end{cases}$$

In the last step to produce the LBP for pixel (xc, yc) a binomial weight 2p is assigned to each sign s(gp − gc). These binomial weights are summed:

$$LBP_{PR}(x_c, y_c) = \Sigma_{p=0 \text{ to } P-1}\, s(g_p - g_c)\, 2^p$$

2.4.1 Uniform of Local Binary Patterns

A Local Binary Pattern is called uniform if it contains at most two bitwise transitions from 0 to 1 or vice versa. In a matter of fact this means that a uniform pattern has no transitions or two transitions. Only one transition is not possible, since the binary string needs to be considered circular. The two patterns with zero transitions, with for example eight bits, are 00000000 and 11111111. Examples of uniform patterns with eight bits and two transitions are 00011100 and 11100001. We can use this to identify patterns:

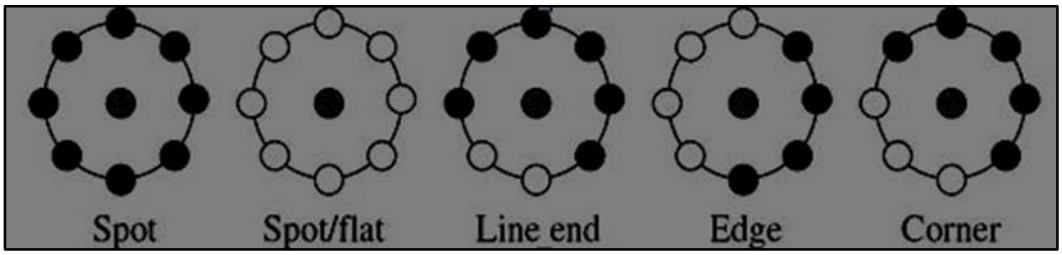


Fig 2.10 Identifying patterns

2.4.2 Face Recognition with Local Binary Patterns

We explained how the LBP-method can be applied on images (of faces) to extract features which can be used to get a measure for the similarity between these images. The main idea is that for every pixel of an image the LBP-code is calculated. The occurrence of each possible pattern in the image is kept up. The histogram of these patterns, also called labels, forms a feature vector, and is thus a representation for the texture of the image. These histograms can then be used to measure the similarity between the images, by calculating the distance between the histograms.

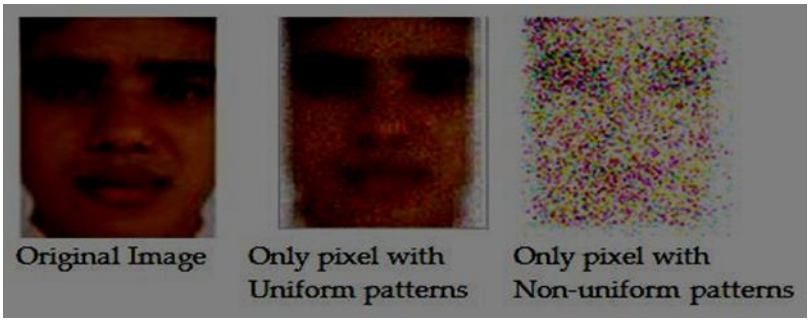


Figure 2.11 shows an image which is split in an image with only pixels with uniform patterns and in an image with only non-uniform patterns.

It occurs that the image with only pixels with uniform patterns still contains a considerable amount of pixels, namely 99 % of the original image. So, 99% of the pixels of the image have uniform patterns (with LBP this is even 99 %). Another striking thing is the fact that, by taking only the pixels with uniform patterns, the background is also preserved. This is because the background pixels all have the same color (same gray value) and thus their patterns contain zero transitions. It also seems that much of the pixels around the mouth, the noise and the eyes (especially the eyebrows) have uniform patterns.

2.4.3 Feature Vectors

Once the Local Binary Pattern for every pixel is calculated, the feature vector of the image can be constructed. For an efficient representation of the face, first the image is divided into K2 regions. In figure 1.7 a face image is divided into 82 = 64 regions. For every region a histogram with all possible labels is constructed. This means that every bin in a histogram represents a pattern and contains the number of its appearance in the region. The feature vector is then constructed by concatenating the regional histograms to one big histogram.

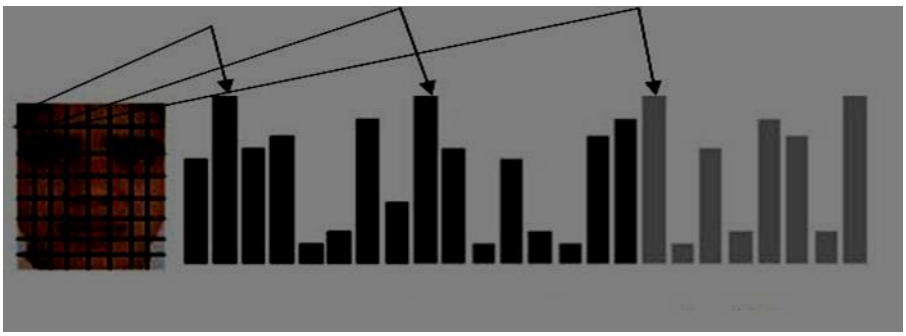


Figure 2.12 : Face image divided into 64 regions, with for every region a histogram

For every region all non-uniform patterns (more than two transitions) are labeled with one single label. This means that every regional histogram consists of P (P − 1) + 3 bins: P (P − 1) bins for the patterns with two transitions, two bins for the patterns with zero transitions and one bin for all non-uniform patterns. The total feature vector for an image contains K2 (P (P− 1) + 3) bins. So, for an image divided into 64 regions and eight sampling points on the circles.

The feature vector is effectively a description of the face on three different levels of locality: the labels contain information about the patterns on a pixel-level; the regions, in which the different labels are summed, contain information on a small regional level and the concatenated histograms give a global description of the face.

2.4.3 Comparing Feature Vectors

In this step, the algorithm is already trained. Each histogram created is used to represent each image from the existing dataset. So, given an input image, we perform the steps again for this new image and creates a histogram which represents the new image. To compare two face images, a sample (S) and a model (M), the difference between the feature vectors has to measure. This can be done with several possible dissimilarity measures for histograms. Some of the popular methods are :

- Euclidean Distance This simply measures, as the name suggests, the Euclidean distance between the two histograms of sample (S) and model (M).

$$D(S, M) = \text{SquareRoot}(\Sigma_{i=1 \text{ to } N} (hist_S - hist_M)^2)$$

- Histogram Intersection Calculated by considering the minimum of the two histograms over sum of all regions.

$$D(S, M) = \Sigma_{j=1 \text{ to } k^2} (\Sigma_{i=1 \text{ to } N} min(S_{i,j} , M_{i,j}))$$

- Log-Likelihood Stastic Somewhat similar to Histogram intersection, here we consider the log likelihood instead of the minimum of the two models.

$$L(S, M) = \Sigma_{j=1 \text{ to } k^2} (\Sigma_{i=1 \text{ to } N} S_{i,j} \, log \, M_{i,j}))$$

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. Note: don't be fooled about the 'confidence' name, as lower confidences are better because it means the distance between the two histograms is closer.

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 if the confidence is lower than the threshold defined.

Thus, the algorithm can predict similarity between two images based on their extracted feature vectors from the Local Binary Pattern Histogram with a certain

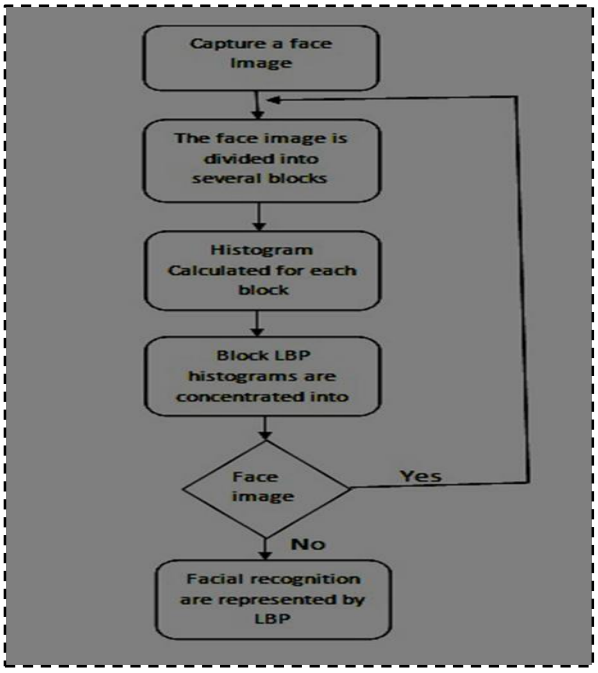


Fig 2.13 Complete LBPH flow

2.5 Security

Hashing is one of the powerful data structure and the basic idea is to use a math problem to organize data into easily searchable buckets. Because organizing and searching for data are such widespread problems in computing, hashing algorithms have become crucial to cryptography, graphics, telecommunications, and biology.

Hash table or Hash Map stores data in the form of (key, value) combination where the key (hashcode) is generated from a hash function and the resulting hash code is

used as the index at which the value is stored in the data structure. That makes accessing the data faster as the index value behaves as a key for the data value.

In most cases the hash function uses the value for generating a key ,hash code, based on some mathematical formula, so that next time when an application or a program look for the value in the data structure, the hash function generates the key using the given value and look for the value at the index location.

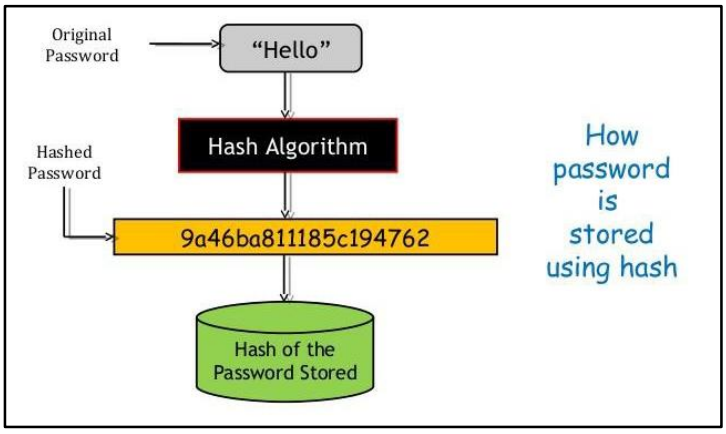


Fig2.14. How hashing works

Organisations or applications hash passwords because in the event an attacker gets read access to the database, we do not want him to retrieve the passwords plain text. Hashing passwords is to prevent this from happening, when the attacker gets his hands on your database, you want to make it as painful as possible to retrieve those passwords using a brute-force attack. Hashing passwords will not make your site/application any more secure, but it will perform damage containment in the event of a breach.

3 Software requirnment

System Development

3.1 Software requirements specification

Operating environments:

Software

1) Operating system: Windows/Linux

2) Languages: Python3 and opencv using python

3) Database: MySQL community server 5.6.10

4) Interface to network

Hardware:

1) Processor: Intel® Core™ i5-5200 CPU@2.20GHz x64-based processor

2) RAM: 8GB

3) Disk: 1TB

4) Web cam

5) Mouse and keyboard

As our project is a desktop application we need a gui interface which will be provided by python . For storing data we need a database. The database with interface available in every language is MySQL. And for securely storing private info such as password we use hashing functions provided by bcrypt.

3.2 Python

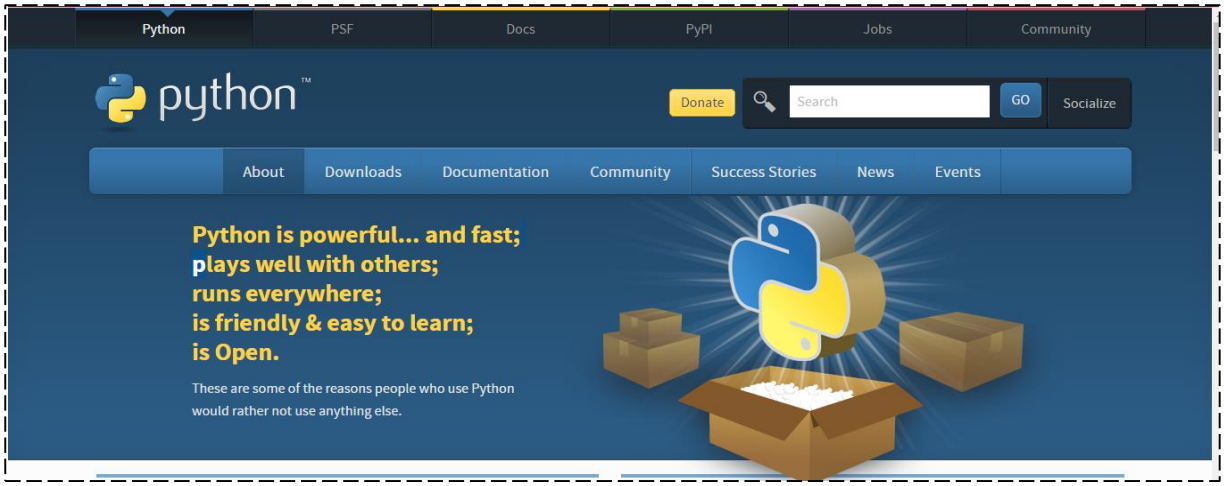


Fig3.1. Python

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

3.2.1 Reasons which make Python most popular:

1- Python is an Inerpreted language, as its the Interpreter which executes the python code line by line, which makes it easier to debug.

2- It can smoothly run on almost any type of platform including Mac OS X, Windows, All versions of Linux, Unix.

3- Databases − Python provides interfaces to all major commercial databases.

4- GUI Programming − Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.

5-It supports automatic garbage collection.

3.3 MYSQL

MySQL is an Oracle-backed open source relational database management system (RDBMS) based on Structured Query Language (SQL). MySQL runs on virtually all platforms, including Linux, UNIX and Windows. Although it can be used in a wide range of applications, MySQL is most often associated with web applications and online publishing.

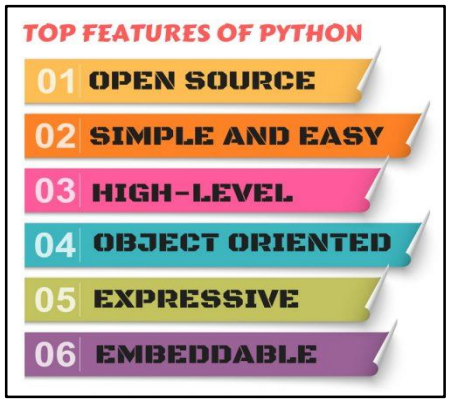


Fig 3.2 features of python

MySQL is an important component of an open source enterprise stack called LAMP. LAMP is a web development platform that uses Linux as the operating system, Apache as the web server, MySQL as the relational database management system and PHP as the object-oriented scripting language. (Sometimes Perl or Python is used instead of PHP.)

Today, MySQL is the RDBMS behind many of the top websites in the world and countless corporate and consumer-facing web-based applications, including Facebook, Twitter and YouTube.

3.3.2 How MySQL works

MySQL is based on a client-server model. The core of MySQL is MySQL server, which handles all of the database instructions (or commands). MySQL server is

available as a separate program for use in a client-server networked environment and as a library that can be embedded (or linked) into seperate applications.

3.3.3 Core MySQL features

MySQL enables data to be stored and accessed across multiple storage engines, including InnoDB, CSV, and NDB. MySQL is also capable of replicating data and partitioning tables for better performance and durability. MySQL users aren't required to learn new commands; they can access their data using standard SQL commands.

3.3.4 Getting started with MySQL

MySQL is written in C and C++ and accessible and available across over 20 platforms, including Mac, Windows, Linux and Unix. The RDBMS supports large databases with millions records and supports many data types including signed or unsigned integers 1, 2, 3, 4, and 8 bytes long; FLOAT; DOUBLE; CHAR; VARCHAR; BINARY; VARBINARY; TEXT; BLOB; DATE; TIME; DATETIME; TIMESTAMP; YEAR; SET; ENUM; and OpenGIS spatial types. Fixed- and variable-length string types are also supported.

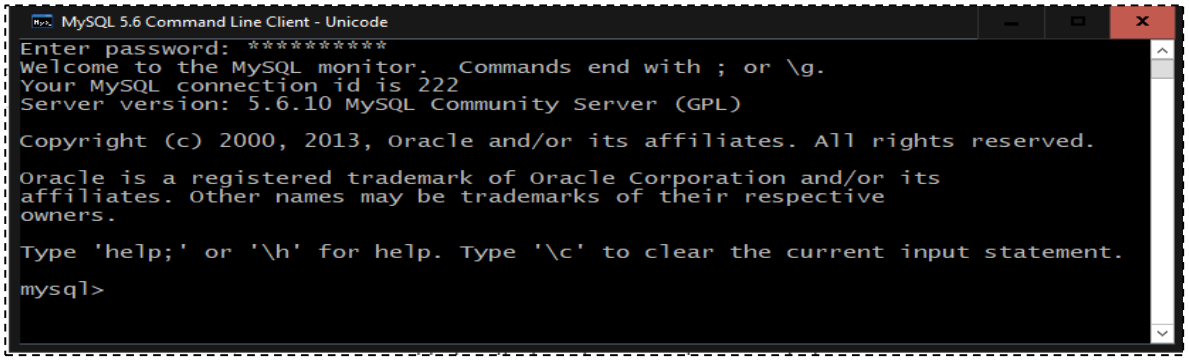


Fig3.3 MYSQL command line client

3.4 Python-OpenCV

OpenCV was started at Intel in 1999 by Gary Bradsky, and the first release came out in 2000. Vadim Pisarevsky joined Gary Bradsky to manage Intel's Russian software

OpenCV team. OpenCV now supports a multitude of algorithms related to Computer Vision and Machine Learning and is expanding day by day.

OpenCV supports a wide variety of programming languages such as C++, Python, Java, etc., and is available on different platforms including Windows, Linux, OS X, Android, and iOS. Interfaces for high-speed GPU operations based on CUDA and OpenCL are also under active development. OpenCV-Python is the Python API for OpenCV, combining the best qualities of the OpenCV C++ API and the Python language. OpenCV-Python is a library of Python bindings designed to solve computer vision problems.

Compared to languages like C/C++, Python is slower. That said, Python can be easily extended with C/C++, which allows us to write computationally intensive code in C/C++ and create Python wrappers that can be used as Python modules. This gives us two advantages: first, the code is as fast as the original C/C++ code (since it is the actual C++ code working in background) and second, it easier to code in Python than C/C++. OpenCV-Python is a Python wrapper for the original OpenCV C++ implementation.

3.4.2 Opencv library

OpenCV-Python makes use of Numpy, which is a highly optimized library for numerical operations with a MATLAB-style syntax. All the OpenCV array structures are converted to and from Numpy arrays. This also makes it easier to integrate with other libraries that use Numpy such as SciPy and Matplotlib.

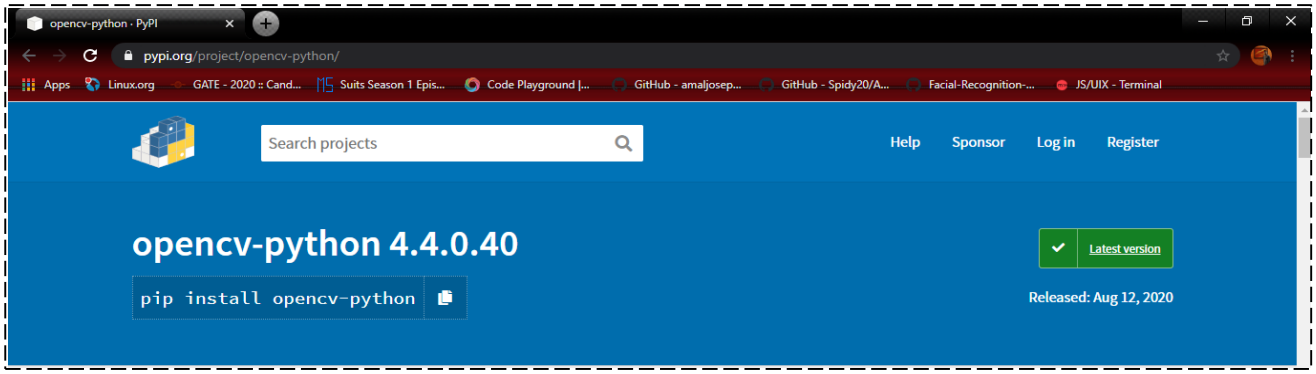


Fig3.4. Opencv-python version

3.4.2 Scope of Python-opencv

OpenC Valmost always begins with a conversion to grayscale, but it can also be a color filter, gradient, or a combination of these. From here, we can do all sorts of analysis and transformations to the source.

1) Background Subtracting
2) Color filtering
3) Edge detection
4) Feature matching for object recognition
5) General object recognition

3.5Tkinter

Python has a lot of GUI frameworks, but Tkinter is the only framework that's built into the Python standard library. Tkinter has several strengths. It's cross-platform, so the same code works on Windows, macOS, and Linux. Visual elements are rendered using native operating system elements, so applications built with Tkinter look like they belong on the platform where they're run.

Tkinter is lightweight and relatively painless to use compared to other frameworks. This makes it a compelling choice for building GUI applications in Python, especially for applications where a modern sheen is unnecessary, and the top priority is to build something that's functional and cross-platform quickly.

3.5.2 How tkinter works

Tkinter commonly comes bundled with Python, using Tk and is Python's standard GUI framework. It is famous for its simplicity and graphical user interface. It is open-source and available under the Python License.

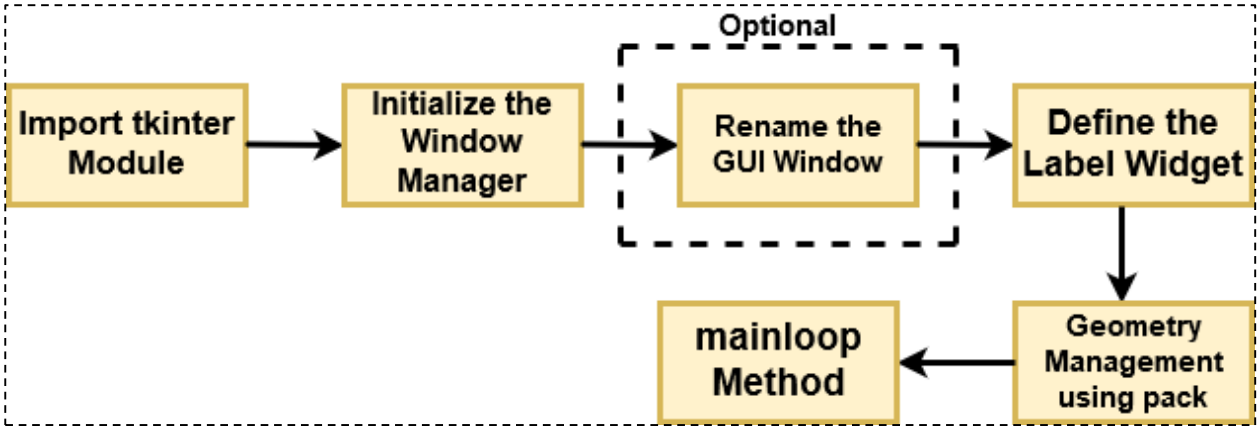


Fig3.5. TKinter window buildup process

3.6 BCRYPT

Not to be confused with encryption, hashing is the irreversible, one way process of taking a string and turning into a fixed length of seemingly random characters. While it's technically possible to reverse-hash a value, the computing power required makes it unfeasible.. Now the attacker would find something like this in the database:

AdityaRP23:
$2b$12$mdyDXtH9LGfyjGHKzvCDXe2sQbm2z.xPrQwb99c4FcTe3Ezlx2fpK

3.6.2)Installing bcrypt

Installing bcrypt in python is a simple process. For windows we have to provide below command.

pip install bcrypt

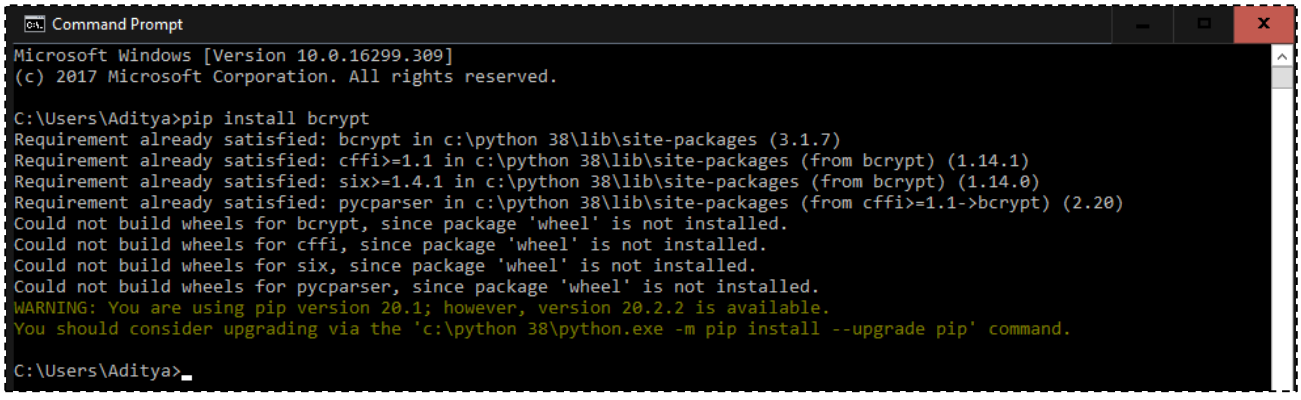


Fig3.6. bcrypt installation process

3.6.3 Hashing with bcrypt

Hashing passwords or any other string is incredibly simple using the bcrypt.hashpw() function. bcrypt.hashpw() takes 2 arguments: A string (bytes)

3.6.4 Salt

A salt is a random string of data hashed alongside a password to keep the hash result unique. Salts should be recreated each time a new password is saved, and the salt is stored alongside the hashed result so that it can be used again for comparison. Libraries like bcrypt are smart enough to store the salt IN the resulting string so that developers don't need to do the extra work.

Fortunately for us, bcrypt also provides a function to generate salt for us - bcrypt.gensalt().

3.7 smtblib

The smtplib module defines an SMTP client session object that can be used to send mail to any Internet machine with an SMTP or ESMTP listener daemon.

3.7.1 Setting up a Local SMTP Server

Rather than sending emails to the specified address, it discards them and prints their content to the console. Running a local debugging server means it's not necessary to deal with encryption of messages or use credentials to log in to an email server.

$ python -m smtpd -c DebuggingServer -n localhost:1025

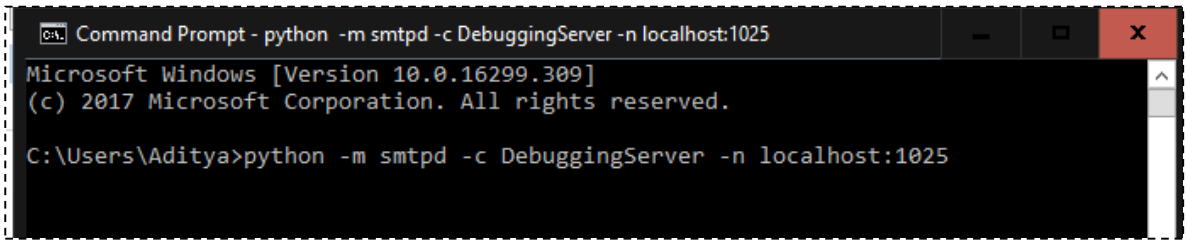


Fig 3.7 SMTP debugging server

3.7.2 Creating SMTP connection

When you send emails through Python, you should make sure that your SMTP connection is encrypted, so that your message and login credentials are not easily accessed by others. SSL (Secure Sockets Layer) and TLS (Transport Layer Security) are two protocols that can be used to encrypt an SMTP connection. It's not necessary to use either of these when using a local debugging server.

There are two ways to start a secure connection with your email server:

Start an SMTP connection that is secured from the beginning using SMTP_SSL().

Start an unsecured SMTP connection that can then be encrypted using .starttls().

4 Workflow of application

4.1 Working

The application launches with the main window which provides all the basic functionality. Faculties have the authority to register or search the students and also to generate attendance sheet or reset attendance.

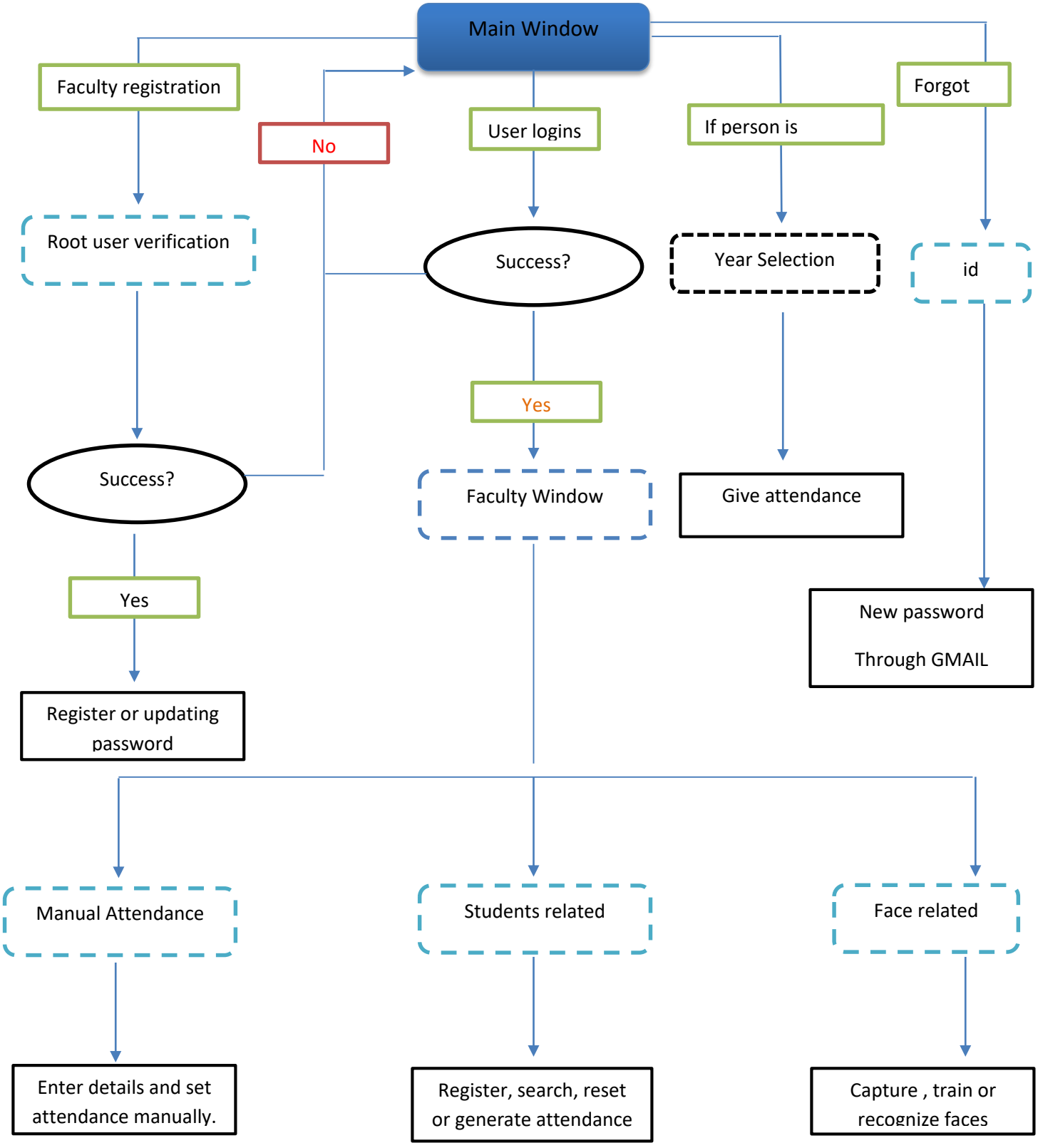
To register a student, the faculty must provide the student's info which includes Name, Roll Number and Year. This info will be associated with the student in the database. After that the student's images are captured through the webcam. Facial images are captured from a considerable number of facial angles so that the student can be recognized even if his/her face is partially obscured. These images added to the dataset and the algorithm is allowed to train on the dataset.

To mark attendance, a student must first select his year and then has to present himself in front of the webcam. The webcam captures the student's image and attempts to identify the image from the stored in the selected year's dataset. If a match is found, the attendance corresponding to the students details is marked.

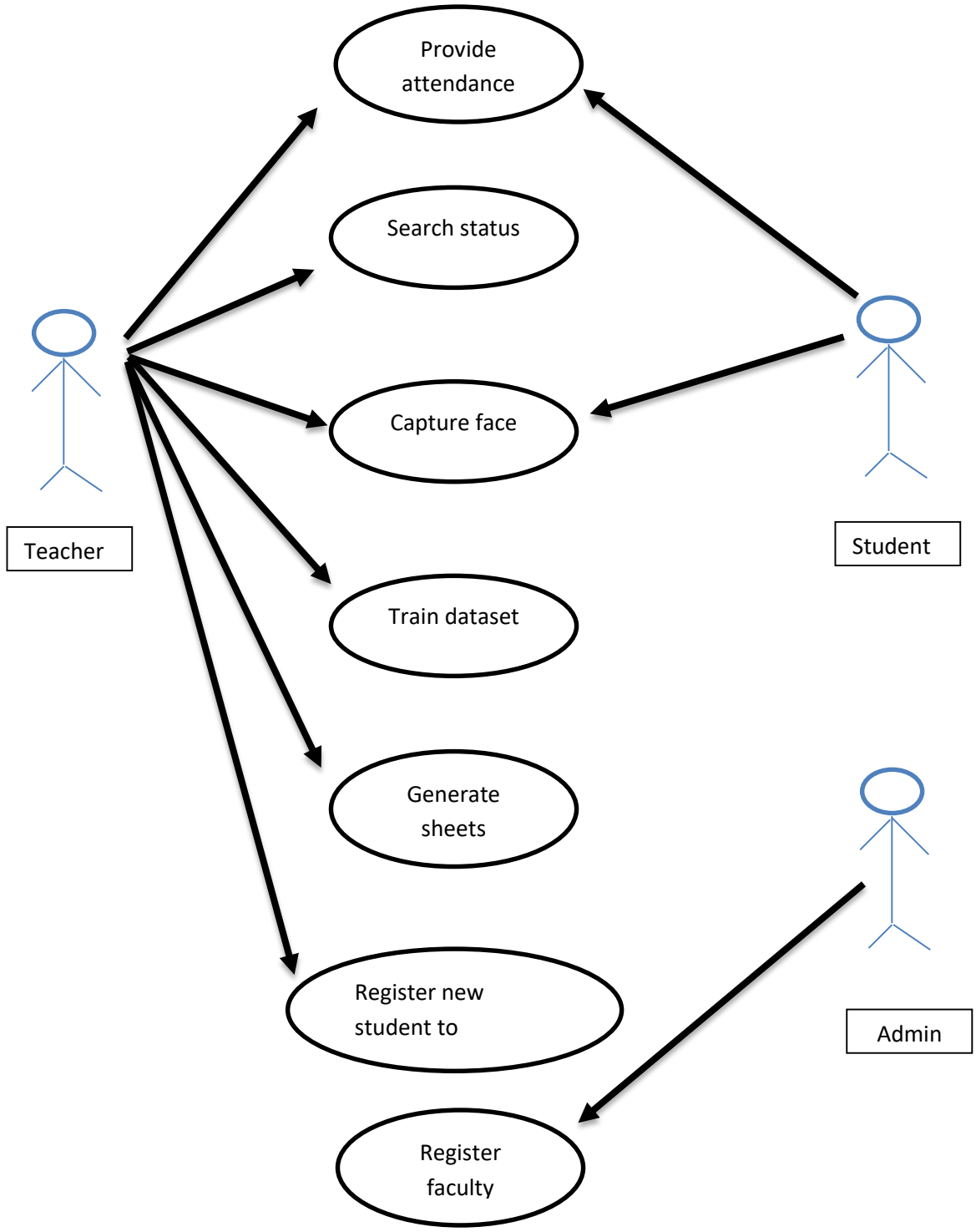
Only the root user has the authority to register a faculty. The faculty can also mark a student's attendance manually if some unforeseen situation occurs due to which student's attendance could not be marked. For that the faculty needs to enter the details of the student in question. A faculty can generate the attendance sheet for reference by choosing appropriate year. Also, faculty can void the attendance if he/she wishes to by using the provided functionality.

In case the faculty forgets the password, an auto-generated password is sent to the faculty's registered mail ID. This password is intended to be temporary. After successful login, the faculty can also change the password using the change password option. For that, knowledge of the current password is necessary. After entering the current password faculty can change his/her password.

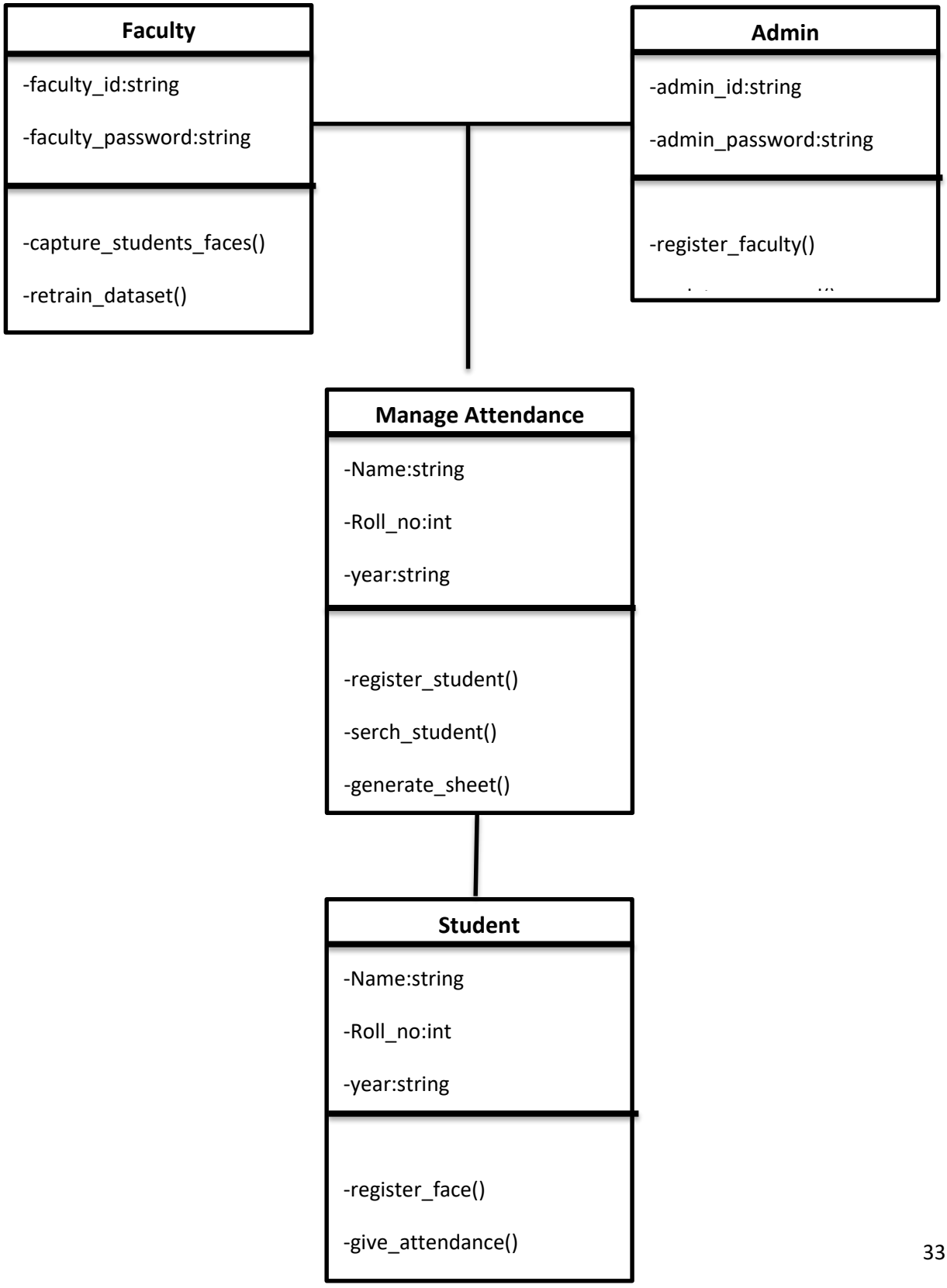
4.2 FLOW of Application



4.3 Use case Diagram



4.4 Class Daigram



5. Performance analysis

5.1 Confidence value

Performance of the LBPH algorithm can be asserted using an equation of confidence. The LBPH algorithm is trained on a relatively smaller database and not some recognized face dataset. Thus its performance metrics rely on how the model classifies the image with certain confidence. Confidence can be loosely defined as the probability with which the model classifies the image into a certain class.

This implementation is used to test the performance of the LBP-method on different kind of face images. Several parameters, like the LBP operator (P and R), non-weighted or weighted regions and the dividing of the regions, are varied to see the influence of these parameters on the performance. For this experiment we have collected lots of face images of a single person. This is done using the webcam. A total of 200 face images are captured. Also note that these images are taken from different angles or different perspective of the face to better identify a person even if the person is facing the webcam from odd angles. Based on algorithm, the face image of an unknown identity is compared with face images of known individuals from a large database.

Accuracy=100*(1-(confidence/300))

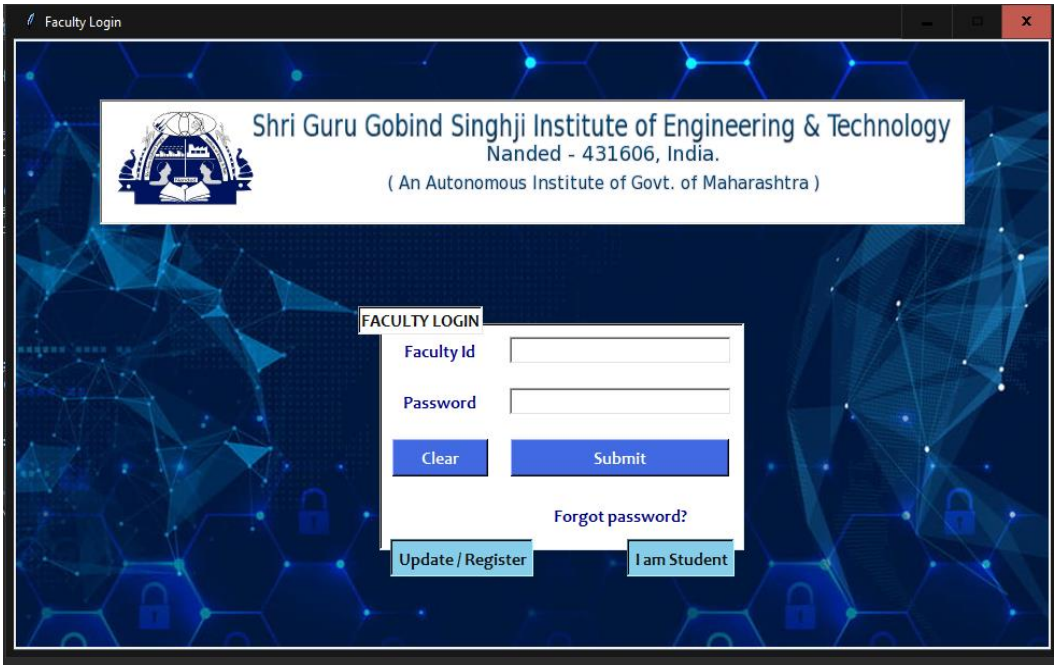
We are accepting faces whose confidence values are<70. Hence in our project using opencv we get accuracy upto 85%.

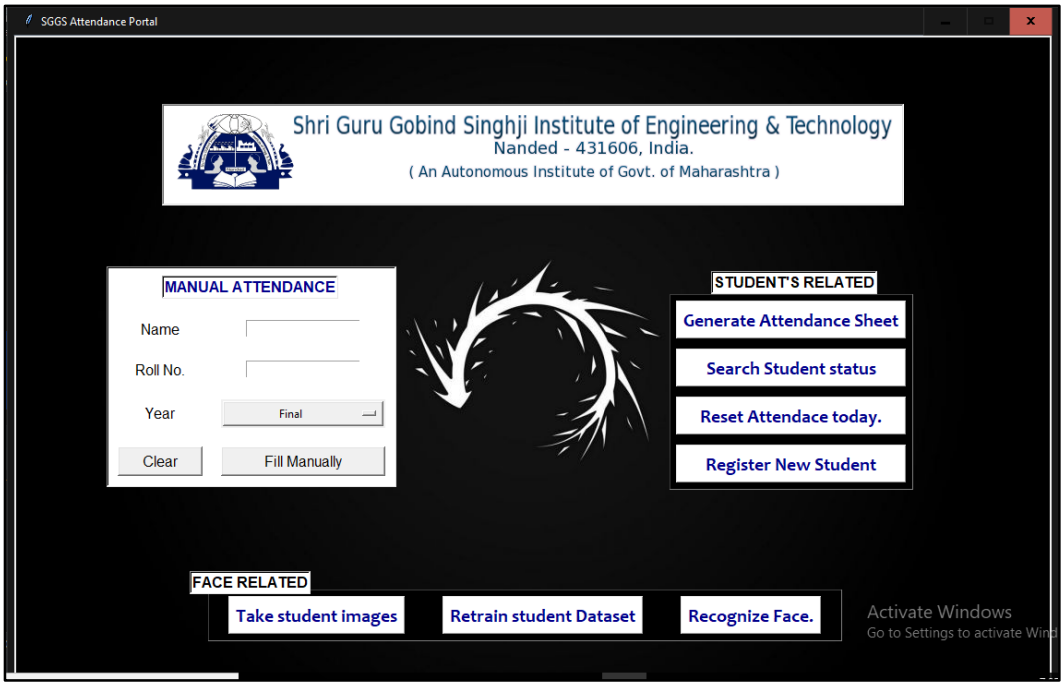
5.2 Testing bcrypt hashing

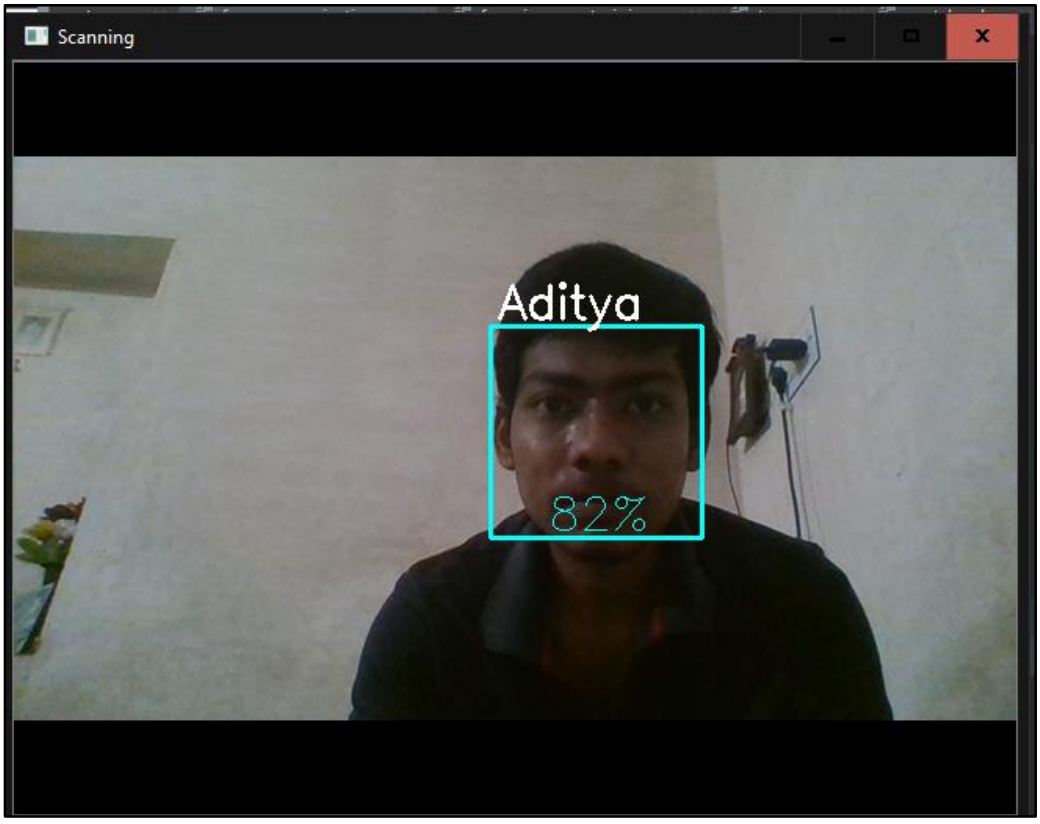
Iteration for hashing	Cost Factor
10	0.15163326263427734
14	1.4977905750274658
15	2.962327003479004
16	5.850872993469238
17	11.564289569854736

- For default salt iteration i.e. 12 it takes 0.3 sec approx. Hence bcrypt can hash at most 4 passwords per second.(0.3 s per password).
- Using a strong password and higher iteration will make it difficult to crack.
- We should be adjusting our work factor so it takes 250-300 ms to compute.
- Our implementation uses the factor 12 as a **lower** bound. And it then benchmark the system, increasing the factor if it's being computed too fast.
- And the function to check a password forces us to deal with the consequences of that. Sometimes a re-hash is needed. It takes 230ms to calculate saved hash.

- **Screen shots**




SGGS Attendance Portal
Shri Guru Gobind Singhji Institute of Engineering & Technology
Nanded - 431606, India.
(An Autonomous Institute of Govt. of Maharashtra)
MANUAL ATTENDANCE
Name
Roll No.
Year
Final
Clear
Fill Manually
STUDENT'S RELATED
Generate Attendance Sheet
Search Student status
Reset Attendace today.
Register New Student
FACE RELATED
Take student images
Retrain student Dataset
Recognize Face.
Activate Windows
Go to Settings to activate Wind


Scanning
Aditya
82%

6. Conclusion

6.1 Conclusion

Our proposed work has an extensive range of advantages over the pre existing work. While keeping the current condition in mind, the Attendance Management System might be the go to method for tracking the attendance. Among the entire sorts of bio metric, face detection and recognition system is the most accurate. In this report, we have presented a modern approach to the Attendance management issue. It is exciting to see face detection techniques be increasingly used in real-world applications and products. Applications and challenges of face detection also discussed which motivated us to do research in face detection and choose appropriately a system which subsides our necessities.

- We have developed an application to manage the various tasks related to Attendance management system with sole control with the faculty members.

- To reduce the false-positives drastically and increase the efficiency in this research, we are using haar like features and for recognition of face we are using LBPH (local binary pattern histogram).

- This reference design is already in use for authentication in corporate offices, banks and other public places in several developed countries and are quite efficiently handled.

- Thus for a safety purpose in real time we designed a face recognition system in minimum expenses using Raspberry Pi, Python, OpenCV and the LBPH Face recognition algorithm.

6.2 Disadvantages of proposed work

• It seems rather unlikely, but there is an askew chance rural parts would lack the system and the required configuration such as a personalized computer and a particular operating system with a suitable web cam. In such a case we shall not be able to provide the advantages of the proposed work. Student window is not provided. And database queries can be more optimized.

• As any other Machine Learning algorithm, this system is also prone to some margin of error which is often unavoidable. The model can only predict the output with certain confidence. However, this margin can be reduced and made as small as possible using the ever evolving techniques discovered by the Machine Learning and Deep Learning community.

6.3 Advantages of proposed work

Facial recognition software gives users a means of tracing their student's attendance, by further removing human mistakes. Among the entire sorts of bio metric, face detection and recognition system is the most accurate. Face recognition precisely reports all the features of attendance, absenteeism, and also over time. The identification procedure is spot-on every single time at a speed that is now practically possible. This scheme can match thousands of operators in less than a second, plus the software offers info that is 100% precise without you lifting a finger.

Facial recognition permits employees to waltz in and out inside seconds, instantly removing the inconvenience of swiping cards otherwise singling badges around. This saves time and effort, resulting in a satisfactory working environment.

Student should only look at the device for less than a second he will be identified.

Unlike the fingerprint hardware that employee should use his finger, the face terminals doesn't require body punch.

Facial recognition technology does not require any physical engagement.

Attendance is taken automatically by the camera placed in the classroom therefore there will be no chances of proxy attendances.

This system uses a simple algorithm and flowchart which is easy to understand as there are no complicated sections, information flow is simple as there is less hardware's components used therefore each section is clearly Understood.

Virtual Classroom: V+irtual classrooms are the class rooms without the lecturers to teach as students will be learning online. This system is very useful in virtual classrooms where there will be no lecturers to take attendances this system will automation

5.4 Future Scope

5.4.1 Extrapolating the application

Our application although in its primary phases has a great potential to be extended and used Nation wide. With right sources of technology and technical savants we can Centralize the Education System of our country. Our nation in 2020 is exhibiting great changes in the education sector. But still record of the student to is eluded from the interesting parties. Where as other nations have a centralized system to monitor their students which also keeps student on the edge. A centralized system will not only keep attendance of the student, but his grades, track of his co curricular activities, sports performance and suggest a well suited vocation for the student.

6.4.2 Technological aspects

Many difficulties has been faced when recognized face images from database such as pose and lighting variations, expression variations, age variations, and facial occlusions. In future to improve the pose correction, quality based frame selection, aging correction, and mark based matching techniques can be combined to build a unified system for video based face recognition.

In this era of ever evolving technological advancements, it would be foolish to think the current system will not be outdated by a new better system having more capabilities. Thus the system is prone to change when a new, better algorithm is in town. Simultaneously, there exist other techniques for face detection and recognition having their own advantages over the current system and limitations as compared to this system.

One such technique is using Deep Convolutional neural networks. Although the CNNs have shown to have better accuracy than its counterparts, they have been known to have very high requirements in terms of computation especially during the training phase. Naturally, these systems are not suited for very large dataset as in our case. A high demand for infrastructure might be a problem because to think that high end infrastructure would be available everywhere in the nation would be an overkill.

7 References

1) https://pythonise.com
2) https://pypi.org/project/
3) https://geeksforgeeks.com
4) https://towardsdatascience.com
5) https://mysql.com
6) https://github.com/opencv/opencv/tree/master/data/haarcascade
7) Michael Jones and Paul Viola, “Robust real-time Object detection”, international Journal of Computer
8) https://opencv.com
9) XueMei Zhao,ChengBing Wei, “A real time face recognition system based on the improved LBPH algorithm”2017 IEEE 2nd International Conference
10) V. Betcy Thanga Shoba, I. Shatheesh Sam “Face recognition using LBPH Descriptor and Convolution Neural network” 2018 Second International Conference on Intelligent Computing and Control System
11) Md. Abdur Rahim, Md. Najmul Hosssain, et al ”Face recognition using local Binary Patterns(LBP)”Global journal of computer science and Technology Graphics and Vision