

ATTENDANCE SYSTEM USING FACE RECOGNIZATION

Submitted in partial fulfillment of the requirements for the award of
Bachelor of science in computer science

By

MS.P.PONKOUSALYA(39290078)

MS.V.UMA MAHESWARI(39290116)



DEPARTMENT OF COMPUTER SCIENCE

SCHOOL OF COMPUTER SCIENCE

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

JEPPIAR NAGAR , RAJIV GANDHI SALAI , CHENNAI-600 119

MARCH 2022



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

(Established under Section 3 of UGC Act, 1956)

JEPPIAAR NAGAR, RAJIV GANDHI SALAI CHENNAI- 600119

www.sathyabama.ac.in



DEPARTMENT OF COMPUTER SCIENCE

BONAFIDE CERTIFICATE

This is to certify that the Project Report is the bonafide work of **P.PONKOUSALYA(39290078)** and **V.UMA MAHESWARI(39290116)** who carried out the project entitled "**ATTENDANCE SYSTEM USING FACE RECOGNIZATION** " under our supervision from November 2021 to May 2022

Internal Guide

(Dr. Kalaiarsai ME., PH.D.,)

Head of the department

(Dr. L. LAKSHMANAN M.E., Ph.D.,)

Submitted for viva voce examination held on

Internal Examiner

External Examiner

DECLARATION

I **V.UMA MAHESWARI** here by declare that the Project Report entitled “**ATTENDANCE SYSTEM USING FACE RECOGANIZATION**” done by me under the guidance of Mrs **KALAIARASI (M.E., Ph.D)**is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering / Technology degree in Computer science.

DATE:

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. SASIKALA, M.E., Ph.D., Dean**, School of Computing and **Dr. S . VIGNESHWARI M.E., Ph.D., and Dr. L. LAKSHMANAN M.E., Ph.D.**, Head of the Department for providing me necessary support and details at the right time during the progressive reviews

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. KALAIRASI M.E., PH.D.**, for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department Of Computer Science** who were helpful in many ways for the completion of the project.

ABSTRACT

This project details a real - time automated attendance system which will mark the attendance of the student and employees. The daily attendance marking is common in schools, colleges and companies. But manual attendance marking is a difficult process, especially for a large number of students/ employees. There are many drawbacks in the attendance system like fake attendance, cost efficient for attendance record and extra. To overcome these drawbacks, there is a need for a smart automated attendance system. Proposed projects for face recognition systems method are not usually accurate as desired. So we aim to solve the exceeding older system, we introduce a new approach to identify the student/employees using face recognition system by deploying with registered id and name.

This system has to be implemented with 4 phases such as

Image capturing

Segmentation of group images and face detection

Face comparison and recognition

Updating of attendance in database

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	v
	LIST OF FIGURE	viii
1	INTRODUCTION	1
	1.1 TECHNIQUES FOR FACE RECOGNITION	2
	1.2 AIM	5
	1.3 SYNOPSIS	5
2	LITERATURE SURVEY	6
	2.1 REVIEW ON LITERATURE WORK	6
	2.2 EXISTING SYSTEM	8
	2.3 PROPOSED SYSTEM	9
3	METHODOLOGY	10
	3.1 REQUIREMENT SPECIFICATION	10
	3.1.1 HARDWARE SPECIFICATION	10
	3.1.2 SOFTWARE SPECIFICATION	10
	3.1.3 TECHNOLOGY USED	10

3.2	APPLICATIONS	11
3.2.1	INSTITUTIONS	11
3.2.2	COMPANIES	11
3.2.3	PRISON	11
3.2.4	MOBILE PLATFORM	12
3.2.5	ADDITIONAL USES	12
3.3	ADVANTAGES AND DISADVANTAGES	13
3.3.1	ADVANTAGES	13
3.3.2	DISADVANTAGES	14
3.4	SYSTEM DESIGN	14
3.4.1	ACTIVITY DIAGRAM	14
3.4.2	DATA FLOW DIAGRAM	15
3.5	SYSTEM ALGORITHM	16
3.5.1	LBPH ALGORITHM	16
3.5.2	THE FACE RECOGNITION ALGORITHM	19
3.6	MODULES	20
3.6.1	IMAGE ACQUISTION	20
3.6.2	PRE - PROCESSING SYSTEM	25
3.6.3	FACE DETECTION	29

	3.6.4 FACE RECOGNITION	34
4	RESULT AND DISCUSSION	38
	4.1 ATTENDANCE	39
	4.2 DATABASE	42
	4.3 TEST CASE	43
5	CONCLUSION AND FUTURE WORK	44
	REFERENCES	46
	APPENDIX	49
	A. SOURCE CODE	49
	B. SCREENSHOT	59

LIST OF FIGURES

FIGURE NO.	FIGURE NAMES
2.1	EXAMPLE OF EIGEN FACES
3.1	ACTIVITY DIAGRAM
3.2	DATA FLOW DIAGRAM
3.3	LBPH
3.4	LBPH PROCEDURE
3.5	EXTRACTING THE HISTOGRAM

3.6	IMAGE ACQUISITION
3.7	HAAR CASCADE FEATURES
3.8	FACE FEATURES
3.9	GRAY SCALE IMAGE
3.10	HAAR LIKE FEATURES ALGORITHM
3.11	FACE DETECTED
4.1	NAME LIST OF 11 STUDENTS
4.2	ATTENDANCE MARKED
4.3	SECOND RAW CELL ATTENDANCE
4.4	DATABASE STORED DATA

CHAPTER 1

INTRODUCTION

A face recognition system may be a technology capable of identifying or verifying a individual from a digital image or a video frame from a video source. There are multiple methods during which face recognition systems work, but generally, they work by comparing selected countenance from given image with faces within a database. It's also described as Biometric AI based application which will uniquely identify an individual by analyzing patterns supported the person's facial texture and shape.

While initially a sort of computer application, it's seen wider uses in recent times on mobile platforms and in other sort of technology, like robotics. It's typically used as access control in security systems and may be compared to other biometrics like fingerprint or eye iris recognition systems. Although the accuracy of face recognition system as a biometric technology is less than iris recognition and fingerprint recognition, it's widely adopted due to its contactless and non-invasive process. Recently, it's also become popular as a billboard identification and marketing tool. Other applications include advanced human-computer interaction, video surveillance, automatic indexing of images, video database, among others.

During 1964 and 1965, Bledsoe, alongside Helen Chan and Charles Bission, worked on using the pc to acknowledge human faces (Bledsoe 1966a, 1966b; Bledsoe and Chan 1965). He was pleased with this work, but because the funding was provided by an unnamed intelligence that didn't allow much publicity, little of the work was published. Support the available references, it had been reveled that the Bledsoe's initial approach involved the manual marking of varied landmarks on the face like the attention centers, mouth, etc., and these were mathematically rotated by computer to catch up on pose variation. The distances between landmarks were also automatically computed and compared between images to work out identity.

This project was labeled man-machine because the human extracted the coordinates of a group of features from the images, which were then employed by the pc for recognition. Using a graphics table (GRAFACON or RAND TABLET), the operator would extract the coordinates of features like the middle of pupils, the within corner of eyes, the surface corner of eye, point of window peak, and so on. From these coordinates, an incensory of 20 distances, like the width of mouth and width of eyes, pupil to pupil, were computed. These operators could process about 40 pictures an hour. Low- resolution images of faces are often enhanced using face hallucination.

1.1 TECHNIQUES FOR FACE RECOGNITION

The method of face recognition is performed in two steps. The first involves features extraction and selection and therefore the second is that the classification of objects. Later developments introduced varying technologies to the procedure. Some of the foremost notable include the subsequent techniques:

- Traditional
- 3-Dimensional recognition
- Skin texture analysis
- Facial recognition combining
- Thermal cameras

Traditional

Some face recognition algorithms identify countenance by extracting landmarks, or features, from a picture of the subject's face. For example, an algorithm may analyze the relative position, size and shape of the eyes, nose, cheekbones and jaw. These features are then won't look for other images with matching features.

Other algorithms normalize a gallery of face images then compress the face data, only saving the info within the image that's useful for face recognition. A probe image is then compared with the face data. One among the earliest successful systems is predicated

on template matching techniques applied to a group of salient facial features, providing a sort of compressed face representation.

Popular face recognition algorithms include principle component analysis using Eigen faces, linear discriminated analysis, elastic bunch graph matching using the Fisher face algorithm, the hidden Markov model, the multi linear subspace learning using tensor representation, and therefore the neuronal motivated dynamic link matching.

3- Dimensional recognition

Three- dimensional face recognition technique uses 3D sensors to capture information about the form of a face. This information is then won't identify distinctive features on the surface of a face, like the contour of the attention sockets, nose and chin.

One advantage of 3D face recognition is that it's not suffering from changes in lighting like other techniques. It also can identify a face from a variety of viewing angles, including a profile view. Three-dimensional data points from a face vastly improve the presion of face recognition.

3D research is enhanced by the event of sophisticated sensors that do a far better job of capturing 3D faces imagery. The sensors work by projecting structured light onto the face. Up to dozen or more of those image sensors are often placed on an equivalent COMS chip, each sensor captures a special a part of the spectrum.

New method is to introduce how to capture a 3D picture by using three tracking cameras that time at different angles; one camera are going to be pointing at the front of the topic, other to the side, and third one at an angle. All these cameras will work together so it can track a subject's face in real time and be ready to face detect and recognize.

Skin texture analysis

Another emerging trend uses the visual details of the skin, as captured in standard digital or scanned images. This technique was called as Skin Texture Analysis, turns

the unique lines, patterns, and spots apparent during a person's skin into a topological space.

Surface Texture Analysis works an equivalent way face recognition does. A picture is taken of a patch of skin to distinguish any lines, pores and therefore the actual skin texture. It can identify the contrast between identical pairs, which aren't yet possible using face recognition software alone.

Tests have shown that with the addition of skin texture analysis, performance in recognizing faces can increase 20 to 25 percent.

Facial recognition combining

As every method has its advantages and drawbacks, technology companies have amalgamated the normal, 3D recognition and Skin Textual Analysis, to make recognition systems that have higher rates of success.

Combined techniques have an advantage over other systems. It is relatively insensitive to changes in expression, including blinking, frowning or smiling and has the power to catch up on mustache or beard growth and therefore the appearance of eyeglasses. The system is additionally uniform the reference to race and gender.

Thermal cameras

A different sort of taking input for face recognition is by using thermal cameras, by this procedure the cameras will only detect the form of the top and it will ignore the topic

Accessories like glasses, hats or makeup. Unlike conventional cameras, thermal cameras can capture facial imagery even in low light and nighttime conditions without employing a flash and exposing the position of the camera. However, a drawback with using thermal pictures for face recognition is restricted. Diego Socolinsky and Andrea Selinger research the utilization of thermal face recognition in real world and operation scenarios, and at an equivalent time build a replacement database of thermal face

images. The research uses low-sensitive, low-resolution ferroelectric electric sensors that are capable of acquiring long-wave thermal infrared (LWIR).

The result show that a fusion of LWIR and regular visual cameras has greater leads to outdoor probes. Indoor results show that visual feature 97.05% accuracy, while LWIR has 93.93% and therefore the fusion has 98.40%, however on the outdoor proves visual has 67.06%, LWIR 83.03% and fusion has 89.02%. The study used 240 subjects over a period of 10 weeks to make a replacement database. The data was collected on sunny, rainy and cloudy days.

1.2 AIM

The main aim of this project is to make an attendance registration when a user scans their face within a specified location.

1.3 SYNOPSIS

The attendance system using face scanning applications are available nowadays, Here we are implementing in an official environment, when an employee enters into the zone, the application automatically scales the current location with the static company's zone. When they near the zone then our application automatically triggers a notification. Then the employee needs to do face verification. When the face is matched then the attendance will be added into the database.

CHAPTER 2

LITERATURE SURVEY

Face detection may be a technology that determines the situation and size of human face in arbitrary image. The countenance is detected and the other objects like trees, buildings and bodies etc are ignored from the digital image.

2.1 REVIEWS ON LITERATURE WORK

Jireh Robert Jam proposed a system of Face Detection and Recognition Student Attendance System. He implements algorithms for face detection and recognition in image processing to create a system which will detect and recognize frontal faces of scholars during a classroom. In human interactions, the face is the most important factor as it contains important information about a person. So he developed a working prototype of a system that will facilitate class control during a classroom by detecting the frontal faces of scholar. But the main drawback is that the evaluation showed that recognition part can achieve approximately 60% recognition rate.

Cheng, et al. developed the system to manage the context of the scholars for the classroom lecture by using note PCs are all the students. Because this system uses the more of every student, the attendance and therefore the position of the students are obtained. However, it is difficult to understand the detailed situation of the lecture. His system takes images of faces. In recent decade, variety of algorithms for face recognition has been proposed, but most of these works affect only single image of a face at a time. By continuously observing of face information, his approach can solve the matter of the face detection and improve the accuracy of face recognition. The drawback was detection time was taking more to detect more faces and it can't able to detect some faces on the screen.

Yohei Kawaguchi developed the system in order to obtain the attendance, positions and face images in classroom lecture; they proposed the attendance management system based on face recognition in the classroom lecture. The system estimates the attendance and the position of each student by continuous observation and recording. The result of their preliminary experiment shows continuous observation improved the performance for estimation of the attendance. Their current work is focused on the method to obtain the different weights of each focused seat according to its location. Their drawback was to improve face detection effectiveness by using the interaction among our systems.

Rajath S Bharadwaj proposed the system that allots attendance to the recognized faces in the database. The basis of developing an automatic attendance management system is to computerize the standard method of taking attendance. His proposed system strives to outgrow the constraints of the existing systems and provides features such as detection of faces, extraction of features, and detection of extracted features and analysis of student's attendance. The system's correctness in detecting and recognizing faces will be more due to use of large number of features(shape, color, LBP, wavelet, auto correlation, etc) of the face. The main drawback this system was attendance is updated only for a single face thou it scans many faces, and more than 5 faces detection was failed.

The[4] Authors had developed a system that capturing the images from camera or cc camera and applying techniques face detection and recognition can decrease the manual work from human and increase the security safety, taking the decision from this recognition result. Based in this face detection and recognition can used in implement so many application like automatic attendances system based on face recognition, worker attendances, security, safety, police application like finding thief in image that help to catching thief. In this system they have implemented an attendance system for a lecture section or laboratory by which lecturer in teaching assistant a record student's attendance. It saves time and effort, especially if it's a lecture with huge number of scholars. The complete system is implemented in MATLAB.

Sunil M P developed the system to implement an attendance system for a lectures or laboratory in which lecturer or teaching assistant can record student's attendance. It

saves time and effort. The complete system is implemented with OpenCV and raspberry pi. This attendance system shows the use of facial recognition techniques for the purpose of student attendance. The result of the experiment shows the detection and recognition part. This method can also detect multiple faces and can be easily used in a classroom. The precision of face recognition is almost more than 90%, only the limitation of the system is if the image is out of the database it produces the faulty result.

Hari Prasanth proposed the solution for the present problem is thru automation of attendance system using face recognition. This project describes the tactic of detection and recognition the face in real time. Here, the camera is fixed with in the classroom and which can capture the image. The faces with the image are detected then recognized with the database after which the attendance is marked. If the attendance is marked as absent, the message about the student's absence is shipped to their parents through lot. This method requires only simple hardware for installation. The management of attendance in this technique is simpler and more accuracy. One difficult task in this system is face testing is considered to be unknown; otherwise it is known and belongs to the person in the database.

2.2 EXISTING SYSTEM

Face is widely used for identification and resolution of humans. The face identification process refers to generating face identifiers and associating personal identity information with the identifier. The face resolution process refers to searching personal identity information or obtaining related personalized service by face identifier. Different from ID code which mainly consists of numbers and alphabets, face identifier needs to be generated by performing some facial image processing algorithms, including face detection, preprocessing, feature extraction and face identifier generation. In the face resolution process, face identifier matching algorithms need to be performed. In addition, the size of face identifier is larger and its structure is more complex than ID code. Therefore, the processes of face identification and resolution need more computation, communication and storage capabilities. Some researchers have

adopted cloud computing to improve processing and storage capacity. Image processing deals with CNN and RNN algorithms with large dataset of trained images. Every image represents an Eigen vector, the data set helps produce variety for the system. A representation of these eigenvectors are called as Eigen faces [Fig 1].



Figure 2.1: Example of Eigen Faces

2.3 PROPOSED SYSTEM

We implement this application in an official environment, when an employee enters into the zone, the application automatically scales the current location with the static company's zone. Then the employee needs to do face verification. When the face is matched then the attendance will be added into the database. Thus the attendance count will be added to the server. This project was created to detect multi-faces at a time. The accuracy level of detection was increased from the existing systems.

CHAPTER 3

METHODOLOGY

The system consists of a camera that captures the images of the employee and sends it to the image enhancement module. After enhancement the image comes within the Face Detection and Recognition modules then the attendance is marked on the database server. At the time of enrollment, templates of face images of individual employees are stored in the Face database. Here all the faces are detected from the input image and therefore the algorithm compares them one by one with the face database. If any face is recognized the attendance is marked on the server from where anyone can access and use it for different purposes. In this way a lot of time is saved and this is highly secure process no one can mark the attendance of other. Attendance is maintained on the server so anyone can access it for purposes like administration, employees themselves.

3.1 REQUIREMENTS SPECIFICATION

3.1.1 HARDWARE SPECIFICATION

- Hard Disk: 500GB and above
- RAM: 4GB and above
- Processor: I3 and above

3.1.2 SOFTWARE SPECIFICATION

- Operating system: Windows 10 (64 bit)
- Software: Python

- Tool: Python 3.7 IDLE, jupyter notebook, Spyder

3.1.3 TECHNOLOGY USED

- LBPH Algorithm
- The Face Recognition Algorithm

3.2 APPLICATIONS

3.2.1 INSTITUTIONS

In institutions they are using the old tradition way for marking attendance, which was called out each one of the student names to check if they are present or absent. This method takes more time. By using facial recognition, the process of marking attendance was easy and time saving. The number of students in an institution was more, using this method of attendance can increase the standard of the institutions.

3.2.2 COMPANIES

In many companies, workers/employees have to use biometrics or ID cards to mark their time of entry and exit. When there was a peak hour the number of entry was more, and this makes more rush. So the man has to wait in a queue until his/her turn has to come. This will cause waste of time. Facial recognition systems provide a convenient way to manage the attendance system. So employees don't want to wait in a queue for marking their entry. This automatic process will mark the time and date of the employees when they enter the company.

3.2.3 PRISON

In prisons facial recognition is useful. Everyday a head count of the prisoners is done, to check all are present. Using this method, the process of the head count was

automated and the efficiency and reliability is improved. So the security was also increased on prisoners.

3.2.4 MOBILE PLATFORMS

Deep Face may be a deep learning face recognition system created by a search group at face book. It identifies human faces in digital images. It employs a nine layer neural net with over 120 million connection weights, and was trained on four million images uploaded by Face book users. The system is claimed to be 97% accurate, compared to 85% for the FBI's Next Generation Identification system.

Apple introduced Face ID on the flagship iphone X as a biometric identification successor to the Touch ID, a finger print based system. Face ID features a face recognition sensor that consists of two parts: a "Romeo" module that projects quite 30,000 infrared dots onto the user's face, and a "Juliet" module that reads the pattern. The pattern is shipped to an area "Secure Enclave" within the device's central processing unit (CPU) to verify a match with the one owner's face.

3.2.5 ADDITIONAL USES

In addition to getting used for security systems, authorities have found variety of other applications for face recognition systems. While earlier post-9/11 deployments were well- publicized trials, newer deployments are rarely written about thanks to their covert nature. Face recognition has been leveraged as a sort of biometric identification for various computing platforms and devices; Android 4.0 "Ice Cream Sandwich" added face recognition employing a Smartphone's front camera as a way of unlocking devices, while Microsoft introduced face recognition login to its Xbox 360 computer game console through its Kinect accessory, also as Windows 10 via its "Windows Hello" platform (which requires an infrared-illuminated camera).

Apple's iPhone X Smartphone introduced face recognition to the merchandise line with its "Face ID" platform, which uses an infrared illumination system.

3.3 ADVANTAGES AND DISADVANTAGES

3.3.1 ADVANTAGES

One key advantage of a face recognition system that it's ready to person mass identification because it doesn't require the cooperation of the test subject to figure. Properly designed systems installed in airports, multiplexes, and other public places can identify individuals among the gang, without passers-by even being conscious of the system.

However, as compared to other biometric techniques, face recognition might not be most reliable and efficient. Quality measures are vital in face recognition systems as large degrees of variations are possible in face images. Factors like illumination, expression, pose and noise during face capture can affect the performance of face recognition systems. Among all biometric systems, face recognition has the very best false acceptance and rejection rates, thus questions are raised on the effectiveness of face recognition software in cases of railway and airport security.

The process of recognizing a face takes a second or less — and this is incredibly beneficial for the companies. In the era of constant cyber attacks and advanced hacking tools, companies need a technology that might be both secure and fast. Considering that face recognition is nearly instant, it grants a fast and efficient verification of an individual. In addition, it's hard to fool this technology so this is often another bonus.

Before, security guards had to perform manual identification of an individual that took an excessive amount of time and didn't boast high accuracy. But today, face recognition is totally independent within the identification process and not only takes

seconds but is additionally incredibly accurate. The 3D face recognition technology and therefore the use of infrared cameras significantly boosted the extent of accuracy of face recognition and made it really hard to fool.

3.3.2 DISADVANTAGES

Ralph Gross, a researcher at the Carnegie Mellon Robotics Institute in 2008, describes one obstacle related to the viewing angle of the face: "Face recognition has been getting pretty good at full frontal faces and 20 degrees off, but as soon as you go towards profile, there've been problems." Besides the pose variations, low-resolution face images are also very hard to recognize. This is one among the most obstacles of face recognition in surveillance systems.

Face recognition is a smaller amount effective if facial expressions vary. A big smile can render the system less effective. For instance: Canada, in 2009, allowed only neutral facial expressions in passport photos.

There is also inconstancy within the datasets employed by researchers. Researchers may use anywhere from several subjects to many subjects and a couple of hundred images to thousands of images. It is important for researchers to form available the datasets they want to one another, or have a minimum of a typical dataset.

Data privacy is that the main concern when it involves storing biometrics data in companies. Data stores about face or biometrics are often accessed by the third party if not stored properly or hacked. In the Tech world, Parris adds (2017), "Hackers will already be looking to duplicate people's faces to trick face recognition systems, but the technology has proved harder to hack than fingerprint or voice recognition technology in the past."

3.4 SYSTEM DESIGN

3.4.1 ACTIVITY DIAGRAM

The system module was also separated into three working modules. They are facial representation, features extraction and classification. The first task was modeling the face. The way the face is represented determines the other two steps. The captured was transformed to match with the positions of the image already present. They are stored as a binary value. The last and final step is recognizing a familiar face. The system will compare the face capture with the face already stored in the database.

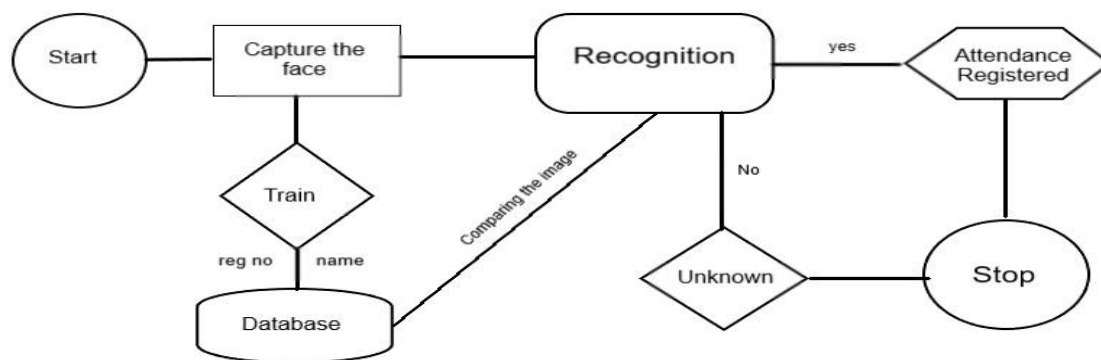


Figure 3.1: Activity diagram

3.4.2 DATA FLOW DIAGRAM

The above diagram shows how the path of the system flows. Images from the live camera are sent into the system. The captured images are converted into a format of grayscale as LBPH works. Through the grayscale images features of the faces are extracted. That feature refers to the gradient in the face. Then the features are compared with the images which are stored to check whether the images match. If the captured face matches, the name and registered id will be shown below and the attendance will be marked for the person.

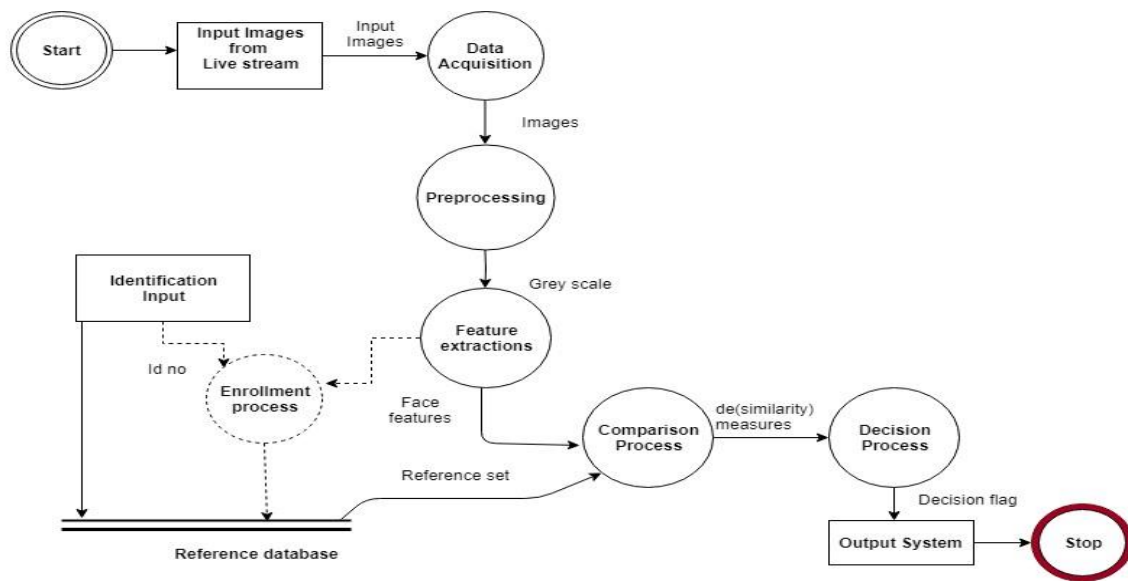


Figure 3.2: Data flow Diagram

3.5 SYSTEM ALGORITHM

This section describes the software algorithm technology used for the system.

- LBPH Algorithm.
- Face Recognition Algorithm.

3.5.1 LBPH ALGORITHM

LBPH stands for Local Binary Pattern Histogram, a basic algorithm that's used to detect faces from the front side. Local Binary Patterns is a type of visual descriptor used for classification in computer vision. The LBP operator helps to get local features by Local Binary Pattern acts. The LBP operator will divide the face which was shown in the camera into pixels. It will divide the examined window into cells. Concatenate (normalized) histograms of all cells. This gives a feature vector for the entire window.

Applying the LBP operation:

The first computational step of the LBPH is to make an intermediate image that describes the first image during a better way, by highlighting the facial characteristics. To do so, the algorithm uses an idea of a window, supported the parameters radius and neighbors.

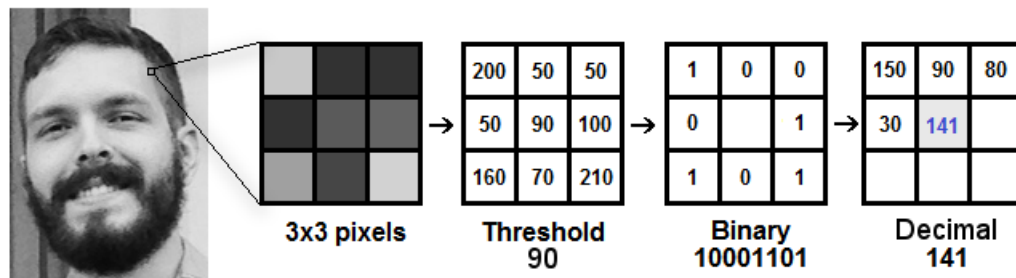


Figure 3.3: LBPH

Based on the image above, let's break it into several small steps so we can understand it easily:

- Suppose we have a facial image in grayscale.
- We can get part of this image as a window of 3x3 pixels.
- It also can be represented as a 3x3 matrix containing the intensity of every pixel (0~255).
- Then, we'd like to require the central value of the matrix to be used because the threshold.
- This value will be used to define the new values from the 8 neighbors.
- For each neighbor of the central value (threshold), we set a replacement binary value.

We set 1 for values equal or above the edge and 0 for values less than the edge.

- Now, the matrix will contain only binary values (ignoring the central value). We need to concatenate each binary value from each position from the matrix line

by line into a replacement binary value (e.g. 10001101). Note: some authors use other approaches to concatenate the binary values (e.g. clockwise direction), but the final result will be the same.

- Then, we convert this binary value to a decimal value and set it to the central value of the matrix, which is really a pixel from the first image.
- At the top of this procedure (LBP procedure), we've a replacement image which represents better the characteristics of the first image.

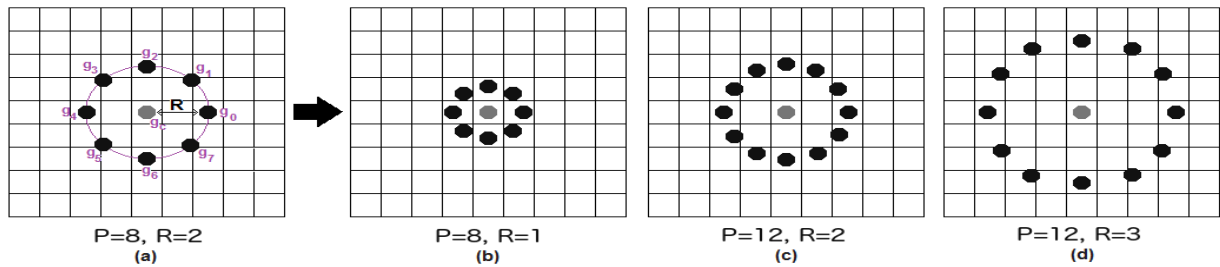


Figure 3.4: LBPH Procedure

It can be done by using bilinear interpolation. If some data point is between the pixels, it uses the values from the 4 nearest pixels (2x2) to estimate the value of the new data point.

Extracting the Histograms:

Now, using the image generated within the last step, we will use the Grid X and Grid Y parameters to divide the image into multiple grids, as are often seen in the following image:

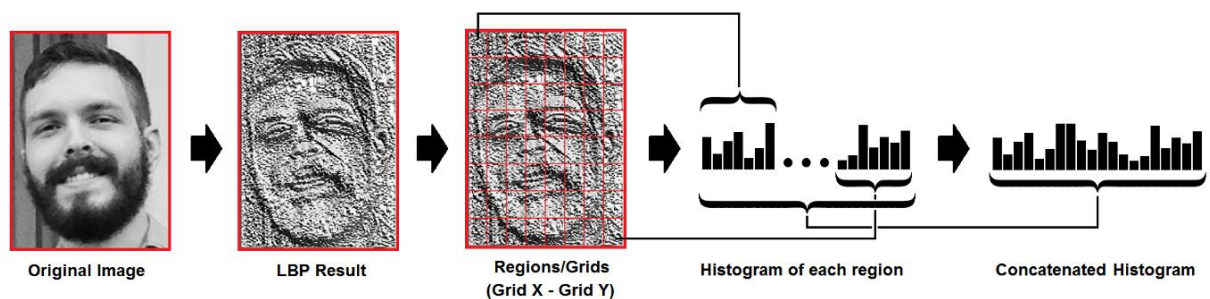


Figure 3.5: Extracting the Histograms

Based on the image above, we can extract the histogram of each region as follows:

- As we've a picture in grayscale, each histogram (from each grid) will contain only 256 positions (0~255) representing the occurrences of every pixel intensity.
- Then, we'd like to concatenate each histogram to make a replacement and larger histogram. Supposing we have 8x8 grids, we will have $8 \times 8 \times 256 = 16.384$ positions in the final histogram. The final histogram represents the characteristics of the image original image.

3.5.2 THE FACE RECOGNITION ALGORITHM

Input: A person's face.

Output: Attendance is marked for recognized faces.

Initialization:

1. Open the camera `cv2.VideoCapture(0)`
2. Import the face classifier `LBPHFaceRecognizer_create()`
3. Read the trained data from the file `recognizer.read('trainer.yml')`

GPU Part:

1. Image is captured frame by frame `img = cap.read()`
2. Face in the frame is converted to greyscale `gray = (img, cv2.COLOR_BGR2GRAY)`
3. All faces are detected in the frame `faces = face_cas.detectMultiScale()`

4. for coordinates of face:

- values inside frame are read `roi_gray = gray[y:y + h, x:x + w]`
- values are compared with stored values `confidence=recognizer.predict(roi_gray)`

5. if `confidence<85`: face matching with the values is recognized attendance is marked
`dataEntry(name,class_count)`

6. else: face is not recognized `id = 'Unknown'`

3.6 MODULES

This section describes the modules of the work.

- Image acquisition
- Pre-Processing image
- Face detection
- Face recognition

These steps of process are takes place during the face recognition.

In the initiative image is captured from the camera. There are illumination effects in the captured image due to different lighting conditions and a few noise which is to be removed before going to the next steps. Median filter is employed for removal of noise within the image. There are other techniques like FFT and low pass filter for noise removal and smoothing of the images but median filter gives good results.

Let's see about the modules work clearly.

3.6.1 IMAGE ACQUISITION

Image acquisition are often accomplished by digitally scanning an existing photograph or by using an electro-optical camera to accumulate a live picture of a topic . Video also can be used as a source of facial images. The most existing face recognition systems contain one camera.

The recognition rate is comparatively low when face images are of varied pose and expression and different illumination. With increasing of the pose angle, the popularity rate decreases. The recognition rate decreases greatly when the pose angle is larger than 30 degrees. Different illumination isn't a drag for a few algorithms like LDA which will still recognize faces with different illumination, but this is often not true for PCA. To overcome this problem, we will generate the face images with frontal view (or little rotation), moderate countenance, and same illumination if PCA algorithm is getting used.

Before any recognition can even be attempted, they system must acquire an image of the subject with sufficient quality and resolution to detect and recognize the face. The issues examined in this section are the sensor-issues in lighting, image/sensor resolution issues, the field-of view, the depth of field, and effects of motion blur.

To recognize a face one needs an image with visible features, which requires that we collect an image with sufficient light levels and quality. Understanding the impact of illumination variation, or normalizing to reduce it, is by far the most well studied of the issues associated with lighting and face-recognition. While this type of work is very important, it is more focused on algorithms and not acquisition, and hence not covered in this chapter. This section will focus on illumination aspects associated with acquisition, in particular, collecting and measuring light. When working at close range in daylight conditions, the issue of sufficient lighting is not a critical concern. However, as one starts looking at long-range face-based recognition, especially for 24 hour “surveillance”, assuring sufficient light level is critical. Addressing this raises 2 unique issues: how to measure those light levels, and what sensors to use to collect in lower light and/or long range settings. Long-range face needs very long focal lengths, often in the range 800-3200mm. Combining distance with the inherent limits on optics results in high F-Numbers levels. For example the Questar Ranger 3.5 which is a portable telescope used in long range surveillance, provides 1275-3500mm focal lengths, but it

comes at a cost of light, with the 89mm(3.5inch) providing F13.2 at 1175mm and F35 at 3500mm.¹ Recalling that each F-stop is a 50% loss of light, this telescope will measure intensity that is orders of magnitude smaller than that measured with a more traditional F4 lens used for close/moderate range face recognition. This need for light is even more exasperated by the need for faster shutter speeds to avoid motion blur issues that will be described later in this chapter. Understanding the available lighting for long-range settings thus is far more important than for standard face recognition. A question then is how to report light levels for long range experiments, especially for low-light conditions. This is important not just for scientific experimentation, but for practical concerns if one wants to determine if conditions are sufficient for a particular system to operate. The most common measure for low-light imaging is in terms of lux. Lux is a measure of illuminance (the accumulated light energy reaching a surface), and measures how much light is in the scene. Given the lux reaching a surface, and the bidirectional reluctance function of the material/subject, one can estimate the luminous lux (the light leaving the surface in a particular direction). Luminous lux is measured in lumens. One can also compute the luminous remittance which is the luminous lux per unit area emitted by a source. Luminous remittance, like luminance, is measured in lux. Given the Luminous lux one can use field-of-view of the lens and its F-stop to estimate the amount of light reaching the sensor from the targets. When the models are done right, this can be effective in predicting the light reaching the camera and hence the response of the sensor. Unfortunately, to use this approach for long-range low-light imaging there are a number of difficulties. First, the reluctance function of the face varies considerably across the population. More significantly, the reflection is directional and is impacted significantly by self-shadowing, so measuring the scene irradiance with a traditional lux meter is not very effective without accounting for rectangle, shading and shadowing which requires a detailed calculation after measurement, making it difficult to use without advanced computer models. Finally, an issue especially important for low light settings is that, even higher end hand-held light sensors are only effective down to .01lux. These sensors use a light-to-voltage conversion that makes them good for bright scenes. But even though their accuracy is officially rated at plus or minus .01lux, in practice, it is quite tenuous below .1lux. In many of our field experiments, the available lux sensors report under low or zero (it is

too dark for them to operate). There are higher end NVIS lux meters, such as the ANV-410 and TSP-410, but these are significantly more expensive and still have the issues of not providing sufficiently directional measurements to measure light that will reach the sensor. There is an alternative, which is to directly measure the light leaving the face in the direction of the sensor: luminance. The candela per square meter (cdm²) is the SI unit of luminance; nit is a non-SI name also used for this unit.

A candela is a lumen per steroidal (solid-angle), so a cdm²(nit) is equivalent to a lumen/m²sr), where as a lux is a lumen/m² there is no simple conversion between lux and nits without using knowledge of the view subtended by the source (face), which varies with distance. Luminance is valuable because it quantities describe the “brightness” of the source and does not vary with distance, whereas illuminance in lux (the “light” falling on a surface) must be manipulated to estimate how much light there is to measure. Putting it another way, illuminance is a good measure to use when asking how well people or cameras can function anywhere within a dimly lit environment, but luminance is the better measure to use for how well they can view a particular target. The question then is how to effectively measure luminance for long-range face, especially if experimenting in low-light conditions. To address these problems, and provide for a simple infield measurement, we have adapted a different type of measurement sensor.

Using a sensor originally de-signed for “sky quality” measurements or “sky darkness” measurements, we have a device that can operate at much lower light levels and can measure a narrow enough FOV to capture just the data of the face. The sensor being used is the SQM-L2, based on the TAOS TSL237S sensor, which is a light to frequency converter. The SQM-L has an added lens so that the Full Width Half Maximum (FWHM) of the sensor is ~20°. The sensitivity to a point source ~19° off-axis is a factor of 10 lower than on-axis and fall off faster beyond that. We will be experimenting with adding a component for further restriction of the field of view. The SQM-L sensor reports Magnitudes/arc second, which is an astronomical unit of measurement, but which is easily converted into cdm². If we let see the SQM-L value reported, then Luminance cdm² = $108000 \times 10^{-0.4 \times s}$. We use the SQM-L for long-range face experiments by having the subject look toward the camera, so it has appropriate

lighting falling on the face, and then aiming the sensor on the center of their face while holding the sensor about 18 inches away. At this range a face subtends approximately 18° , i.e. the sensor is measuring the light leaving the face and tiny else (but some care has got to be used if there are distant lights behind the topic, and not shadow the face from any light sources with the hand/sensor). We call this measurement the “face luminance” and consider it the most useful overall lighting measurement for estimating performance of a long-range face-system in low-light conditions.

This is really a measurement of luminance but it can be converted to luminous lux using the area of a face and the solid angle subtended by a face from the target range, which is simple scaling.



Figure 3.6: Image Acquisition

(Example of low-light long-range EMCCD imagery. The measured scene illuminance for the left image was .01lux, and illuminance was not measurable for the other two images. The measured Face Luminance left to right, was 0.089 nits, .0768 units and 0.015 nits respectively).

For example in figure, we have long range images in low-light conditions. The images were obtained at approximately 100m with an F5.6 Sigma 300mm-800mm. Capture occurs under star-light conditions 60, 90 and 120 minutes after sunset, with a street light 100m off on the subject's left. We prefer the face luminance measurement approach because works in the low- light setting where we want to operate and it already accounts for the complex lighting/face shape interactions and is easily converted to a direct measure of the luminous lux heading in the direction of the camera. It is also very easy to “measure” in the field: hold the sensor, face the direction of the camera and push the button on the sensor and hold (maybe up to 60 seconds if it's really dark), then read the measurements from the unit's LEDs. These

measurements are more repeatable and reliable than using simple lux-based estimations of overall illumination and then trying to convert it to lux at the sensor. The second major issue impacting light levels for acquisition is the inherent imaging system sensitivity. This is significantly impacted by the sensor. Again, since long-range face is generally for surveillance, there is a general need to consider low-light conditions. One approach often suggested for dealing with low-light settings is the use of Infra-Red sensors for face recognition. For long wave IR (8-14 microns), the human body is a light source and such images could be collected in total darkness. While there has been some significant progress in the area of LWIR face, we believe LWIR is too limited for long range face for several reasons. First, the need for long focal distance lenses and high-resolution sensors for long-range face- the mixture of which are simply not available for LWIR. The resolution issues will be discussed in the next section. The second limitation of LWIR is that since long-range face is usually for non-cooperative subjects; LWIR requires specialize enrollment whereas visible recognition can use standard intelligence photos.

3.6.2 PRE – PROCESSING SYSTEMS

The system captures around 50 images of every individuals face. The images are converted into grey scale as LBPH operates using images in grey scale and the images are stored in a folder. The stored images will be saved with a name and ID unique to that person. In Image pre-processing HAAR CASCADE algorithm was used to train the image. HAAR CASCADE features are more useful in image processing.

Haar Cascade

Haar-like features are digital image features utilized in visual perception. A Haar-like feature considers adjacent rectangular regions at a selected location during a detection window, sums up the pixel intensities in each region and calculates the difference between these sums. A human face, it's a standard observation that among all faces

the region of the eyes is darker than the region of the cheeks. Therefore, a standard Haar feature for face detection may be a set of two adjacent rectangles that lie above the attention and therefore the cheek region. The position of those rectangles is defined relative to a detection window that acts sort of a bounding box to the target object. The key advantage of a Haar-like feature over most other features is its calculation speed. Due to the utilization of integral images, a Haar-like feature of any size are often calculated in constant time (approximately 60 microprocessor instructions for a 2-rectangle feature).

It is a machine learning based approach where a cascade function is trained from tons of positive and negative images. It is then wont to detect objects in other images.

The algorithm has four stages:

- Haar Feature Selection
- Creating Integral Images
- Adaboost Training
- Cascading Classifiers

It is documented for having the ability to detect faces and body parts in a picture, but are often trained to spot almost any object. Here we will work with face detection. Initially, the algorithm needs tons of positive images

(images of faces) and negative images (images without faces) to coach the classifier.

Then we need to extract features from it. For this, Haar features shown within the below image are used.

They are just like our convolution kernel. Each feature may be a single value obtained by subtracting sum of pixels under the white rectangle from sum of pixels under the black rectangle.

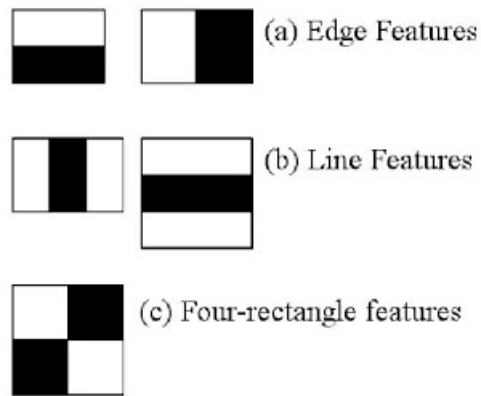


Figure 3.7: Haar Cascade Features

Now, all possible sizes and locations of every kernel are went to calculate many features. For each feature calculation, we'd like to seek out the sum of the pixels under white and black rectangles. To solve this, we introduced the integral image. However large your image, it reduces the calculations for a given pixel to an operation involving just four pixels. It makes things super-fast.

But among of these features we calculated, most of them are irrelevant. For example, consider the image below. The top row shows two good features. The first feature selected seems to specialize in the property that the region of the eyes is usually darker than the region of the nose and cheeks. The second feature selected relies on the property that the eyes are darker than the bridge of the nose. But an equivalent window applied to cheeks or the other place is irrelevant. It is achieved by Adaboost.

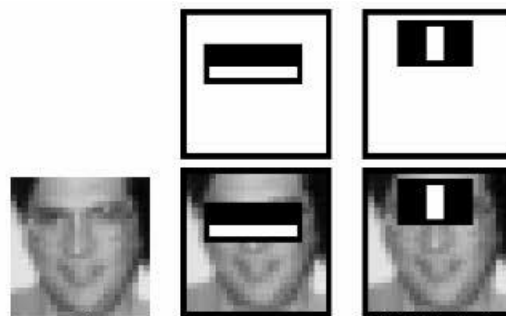


Figure 3.8: Face Features

For this, we apply each and each feature on all the training images. For each feature, it finds the simplest threshold which can classify the faces to positive and negative. Obviously, there will be errors or misclassifications. We select the features with minimum error rate, which suggests they're the features that the majority accurately classify the face and non-face images. (The process is not as simple as this. Each image is given an equal weight within the beginning. After each classification, weights of misclassified images are increased. Then the same process is done. New error rates are calculated. Also new weights. The process is sustained until the specified accuracy or error rate is achieved or the specified numbers of features are found).

The final classifier may be a weighted sum of those weak classifiers. It is called weak because it alone can't classify the image, but alongside others forms a robust classifier. The paper says even 200 features provide detection with 95% accuracy. Their final setup had around 6000 features.

So now you take an image. Take each 24x24 window. In an image, most of the image is non-face region. So it's a far better idea to possess an easy method to see if a window isn't a face region. If it is not, discard it in a single shot, and don't process it again. Instead, specialize in regions where there are often faces. This way, we spend more time checking possible face regions.

For this we used the concept of Cascade of Classifiers. Instead of applying all 6000 features on a window, the features are grouped into different stages of classifiers and applied one-by-one.

(Normally the primary few stages will contain very many fewer features). If a window fails the primary stage, discard it. We don't consider the remaining features on it. If it passes, apply the second stage of features and continue the method. The window which passes all stages may be a face region.

OpenCV provides a training method (see Cascade Classifier Training) or pretrained models, which will be read using the `cv::CascadeClassifier::load` method. First, a `cv::CascadeClassifier` is made and therefore the necessary XML file is loaded using the `cv::CascadeClassifier::load` method. Afterwards, the detection is completed using the `cv::CascadeClassifier::detectMultiScale` method, which returns boundary

rectangles for the detected faces or eyes. Be sure the program will find the trail of files `haarcascade_frontalface_alt.xml` and `haarcascade_eye_tree_eyeglasses.xml`. They are located in `opencv/data/haarcascades`.

3.6.3 FACE DETECTION

Face detection may be a technology getting used during a sort of applications that identifies human faces in digital images. Face detection also refers to the psychological process by which humans locate and attend to faces during a visual scene.

Face detection are often considered a selected case of object-class detection. In object-class detection, the task is to seek out the locations and sizes of all objects in a picture that belong to a given class. Examples include upper torsos, pedestrians, and cars.

Face-detection algorithms specialize in the detection of frontal human faces. It is analogous to image detection during which the image of an individual is matched bit by bit. Image matches with the image stores in database. Any facial feature changes within the database will invalidate the matching process.

Firstly, the possible human eye regions are detected by testing all the valley regions within the gray-level image. Then the genetic algorithm is employed to get all the possible face regions which include the eyebrows, the iris, the nostril and therefore the mouth corners.

Each possible face candidate is normalized to scale back both the lighting effect, which is caused by uneven illumination; and therefore the shirring effect, which is thanks to head movement. The fitness value of every candidate is measured supported its projection on the Eigen-faces. After variety of iterations, all the face candidates with a high fitness value are selected for further verification. At this stage, the face symmetry is measured and therefore the existence of the various countenances is verified for every face candidate.

Facial motion capture

Facial motion capture is that the process of electronically converting the movements of an individual's face into a digital database using cameras or laser scanners. This database may then be used to produce CG (computer graphics) computer animation for movies, games, or real-time avatars. Because the motion of CG characters springs from the movements of real people, it leads to a more realistic and nuanced computer character animation than if the animation were created manually.

A facial motion capture database describes the coordinates or relative positions of reference points on the actor's face. The capture could also be in two dimensions, during which case the capture process is usually called "expression tracking", or in three dimensions. Two dimensional captures are often achieved employing a single camera and capture software. This produces less sophisticated tracking, and is unable to completely capture three-dimensional motions like head rotation. Three-dimensional capture is accomplished using multi-camera rigs or laser marker system. Such systems are typically much more expensive, complicated, and time-consuming to use. Two predominate technologies exist; marker and marker less tracking systems.

Technology

Digital video-based methods are getting increasingly preferred, as mechanical systems tend to be cumbersome and difficult to use. Using digital cameras, the input user's expressions are processed to supply the top pose, which allows the software to then find the eyes, nose and mouth. The face is initially calibrated employing a neutral expression. Then counting on the architecture, the eyebrows, eyelids, cheeks, and mouth are often processed as differences from the neutral expression. This is done by trying to find the sides of the lips as an example and recognizing it as a singular object. Often contrast enhancing makeup or markers are worn, or another method to form the processing faster. Like voice recognition, the simplest techniques are only good 90 percent of the time, requiring an excellent deal of tweaking by hand, or tolerance for errors.

Since computer generated characters don't actually have muscles, different techniques are used to achieve the same results. Some animators create bones or objects that are controlled by the capture software, and move them accordingly, which when the character is rigged correctly gives an honest approximation. Since faces are very elastic this system is usually mixed with others, adjusting the weights differently for the skin elasticity and other factors counting on the specified expressions.

Photography

Some recent digital cameras use face detection for autofocus. Face detection is additionally useful for choosing regions of interest in photo slideshows that use a pan-and-scale Ken Burns effect. Modern appliances also use smile detection to require a photograph at an appropriate time.

Emotional Inference

Face detection are often used as a part of a software implementation of emotional inference. Emotional inference is often wont to help people with autism understand the emotions of individuals around them.

Lip Reading

Face detection is important for the method of language inference from visual queues. Lip reading has applications in help computers determine who is speaking which is required when security is vital.

Gray Scale Base

Gray information within a face can also be treating as important features. Facial features such as eyebrows, pupils, and lips appear generally darker than their surrounding facial regions. Various recent feature extraction algorithms search for local

gray minima within segmented facial regions. In these algorithms, the input images are first enhanced by contrast-stretching and gray-scale morphological routines to improve the quality of local dark patches and thereby make detection easier.

Real – time face detection

Real-time face detection involves detection of a face from a series of frames from a video- capturing device. While the hardware requirements for such a system are much more stringent, from a computer vision stand point, real-time face detection is really a far simpler process than detecting a face in a static image. This is because unlike most of our surrounding environment, people are continually moving. We walk around, blink, fidget, wave our hands about, etc.

The feature-based method is to locate faces by extracting structural features of the face. It is first trained as a classifier then wont 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 report a hit rate of 94%.

The Face Detection working process

The face detection work on detect multiple faces in a picture . Here we work on Open CV for Face Detection, and there are some steps that how face detection operates, which are as follows:-

- 1 Firstly the image is imported by providing the location of the image. Then the image is transformed from RGB to Grayscale because it's easy to detect faces within the grayscale.



Figure 3.9: Gray scale image

2 After that, the image manipulation used, during which the resizing, cropping, blurring and sharpening of the pictures done if needed. The next step is image segmentation, which is employed for contour detection or segments the multiple objects during a single image in order that the classifier can quickly detect the objects and faces in the picture.

3 The next step is to use Haar-Like features algorithm, which is proposed by Viola and Jones for face detection. This algorithm used for locating the situation of the human faces during a frame or image. All face s shares some universal properties of the human face just like the eyes region is darker than its neighbor pixels and nose region is brighter than eye region.

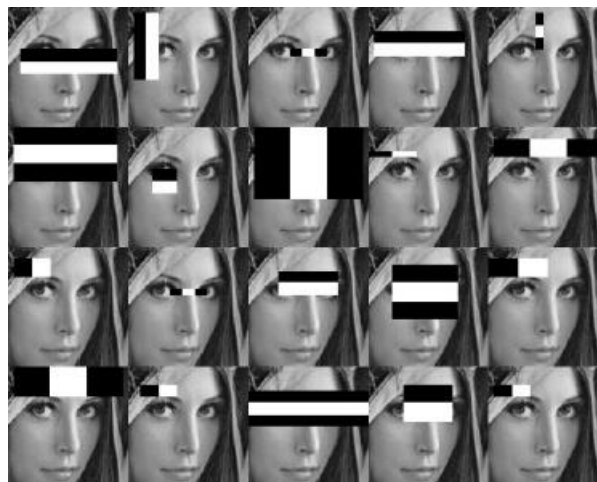


Figure 3.10: Haar like features algorithm

4 The haar-like algorithm is additionally used for feature selection or feature extraction for an object in a picture, with the assistance of edge detection, line detection, centre detection for detecting eyes, nose, mouth, etc. in the picture. It is wont to select the essential features in a picture and extract these features for face detection.

5 The next step is to offer the coordinates of x , y , w , h which makes a rectangle box up the image to point out the situation of the face or we can say that to show the region of interest in the image. After this, it can make a rectangle box up the world of interest where it detects the face. There also are many other detection techniques that are used together for detection like smile detection, eye detection, blink detection, etc.

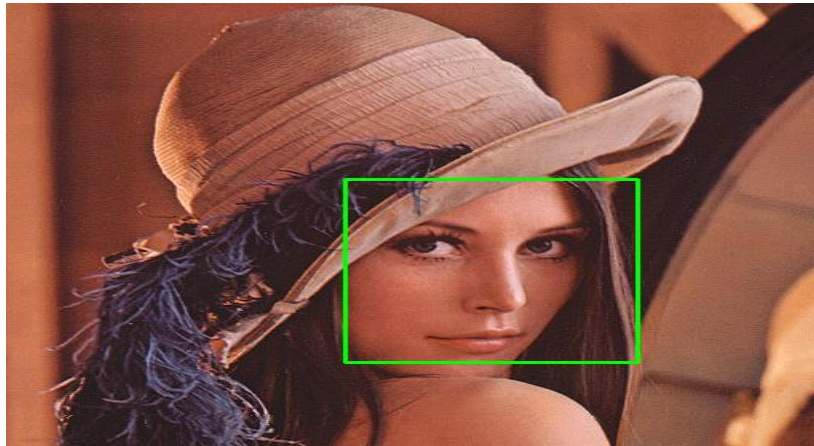


Figure 3.11: Face detected

3.6.4 FACE RECOGNITION

Facial recognition may be a way of recognizing a person's face through technology. A face recognition system uses biometrics to map countenance from a photograph or video. It compares the knowledge with a database of known faces to seek out a match. Facial recognition can help verify identity, but it also raises privacy issues.

Biometrics can add safety and convenience, but what about the question of knowledge security? But that's where it gets complicated. If privacy is vital to you, you almost

certainly want some control over how your personal information — your data — is employed. And here's the thing: your "face print" is data.

You might be good at recognizing faces. You probably find it a cinch to spot the face of a friend and classmates. You're conversant in their countenance — their eyes, nose, mouth — and the way they are available together.

That's how a face recognition system works, but on a grand, algorithmic scale. Where you see a face, recognition technology sees data. That data can be stored and accessed.

So how does facial recognition work?

Technologies vary, but here are the basic steps:

Step 1: A picture of your face is captured from a photo or video. Your face might appear alone or during a crowd. Your image may show you looking straight ahead or nearly in profile.

Step 2: Facial recognition software reads the geometry of your face. Key factors include the distance between your eyes and the distance from forehead to chin. The software identifies facial landmarks — one system identifies 68 of them — that are key to distinguishing your face.

The result: your facial signature.

Step 3: Your facial signature — a mathematical formula — is compared to a database of known faces.

Step 4: A determination is made. Your face print may match that of a picture during a face recognition system database.

Steps for face recognition working

First, we'd like to load the specified XML classifiers.

Import cv2

```
harcascadePath= "haarcascade_frontalface_default.xml"
```

```
facecascade= cv2.CascadeClassifier(harcascadePath)
```

This loads the face cascade into memory so it's ready to be used . Remember, the cascade is simply an XML file that contains the info to detect faces.

```
Cam = cv2.VideoCapture(0)
```

This line sets the video source to the default webcam, which OpenCV can easily capture.

```
while True:
```

```
#Capture frame-by-frame
```

```
ret, im = cam.read()
```

In here, We're capturing the photo. The read () method reads one frame from the video source which is that the Webcam and it returns: The actual video frame read (one frame on each loop).

A return code:

If we run out of frames by any chance, The return code will tell us but in our scenario, We are not reading from a file it's our own Webcam so there's no possibility for us to run out of frames.

```
gray = cv2.cvtColor(im, cv2.COLOR_BGR2GRAY)
```

We just converted our Webcam feed to Grayscale. (Most of the operations in OpenCV are done in grayscale.)

```
faces = faceCascade.detectMultiScale(gray, scaleFactor=1.2, minNeighbors=5,  
minSize=(30, 30),
```

```
flags=cv2.CASCADE_SCALE_IMAGE)
```

Here's when our code detects faces in our frame. Let's drill down from each of the above. The detect Multi Scale function may be a general function that detects objects. It detects faces since we're using it on the face cascade. The first option is the grayscale image. The second is the scale Factor. Since some faces could also be closer to the camera, they might appear bigger than the faces within the back. The scale factor compensates for this. (You can change it to 1.5)

The detection algorithm uses a moving window to detect objects. Min Neighbors defines what percentage objects are detected near the present one before it declares the face found. Min Size, meanwhile, gives the size of each window.

The function returns an inventory of rectangles during which it believes it found a face. Next, we'll loop over where it thinks it found something.

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

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

This function returns 4 values: the x and y location of the rectangle, and therefore the rectangle's width and height (w , h). We use these values to draw a rectangle using the built-in rectangle() function.

```
cv2.imshow('im', im)
```

This is the frame name (Kind of comparable to the title of the Webcam popup) img_counter is a variable to count and increment after each photo was taken. We should declare this before declaring our While loop on the beginning

```
(While=True)
```

To Clean everything up;

```
# When everything is done, release the capture
```

```
cam.release()
```

```
cv2.destroyAllWindows()
```

CHAPTER 4

RESULTS AND DISCUSSION

Given a still image or video of a scene, identify or verify one or more persons in this scene using a stored database of facial images.

Interclass similarity:

Different persons may have very similar appearance

... Twins

... Relatives

... Strangers may look like those persons images are looks same, so the device can't able to recognize who was who. Due to these attendance was falling for wrong person. And also device may fails to detect the face looks similar.

Difference between two images of the same subject under different view angles is greater than differences between images of two different subjects under the same view. This device clearly identifies the images and recognition was perfectly done. Facial expressions caused by facial muscle movements may significantly deform the face surface. Due to facial expression, the mouth and nose angles are differs, so camera takes more time to detect the persons. It takes 20% more time compares with non-facial expression.

... Males are easier to recognize than females (5-9%)

... Younger peoples are easiest to acknowledge than older people

Performance measurements:

... False Acceptance Rate (FAR): Percentage of instances that the system accepts a claimed identity when it shouldn't. FAR for this project has 60% above.

... False Rejection Rate (FRR): Percentage of instances that the system rejects a claimed identity when it shouldn't. FRR rates more than anything else. False Rejection are proving 80% - 90%.

... Recognition Rate (RR): Percentage of instances that an individual is correctly recognized by the system. Based on RR the recognition rate percent was 80%. As per the target it was 10% less than what we thought.

4.1 ATTENDANCE

Class attendance is very important aspects for the students studying in the colleges or schools. For an organization to be successful, it needs precise and quick method for recording the performance of the individuals inside this organization. Attendance gives the data of the individual whether that particular person is physically present or absent. The traditional method of calling roll number or name of the student for marking attendance is time consuming or wastage of time during the class hour. In this project an automated student attendance system is made, which is based on face recognition. The recognized face of the individual is used for marking the attendance.

Creating Attendance Sheet

As the database contain images of students registered, so the attendance sheet should contain name of the student.

For example, the 11×2 cell array is created which contain the name of the 11 student as shown in the figure.

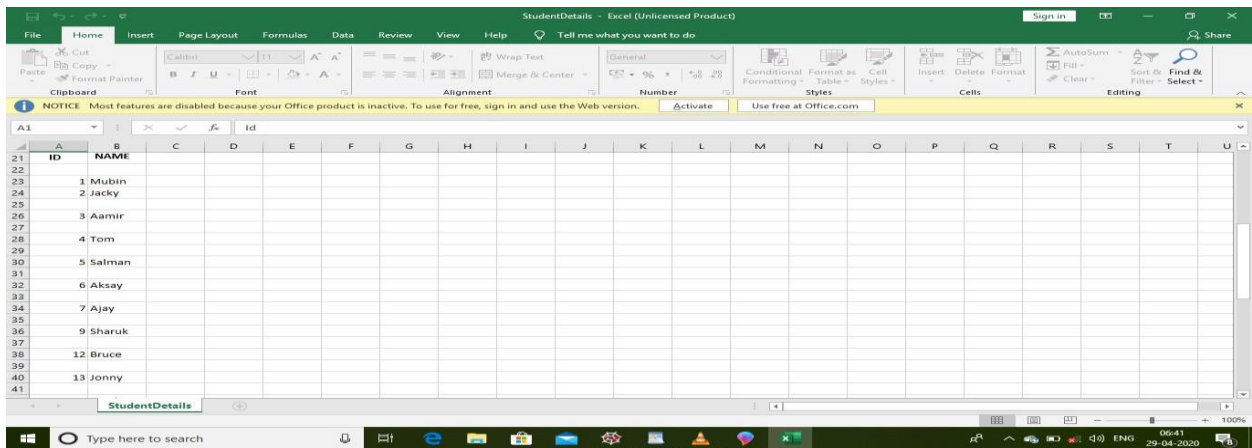


Figure 4.1: Name list of 11 students

Marking Attendance of the Student

When the face of the particular student is recognized, attendance is marked for that student for that particular day. If some students are absent then no attendance is marked for them. In the 11x2 cell each cell contain name of the student, and attendance is marked for the student shown in below. The attendance sheet is stored in the variable name data. Consider the second name stored in the cell array. i.e., Jacky. When Jacky is recognized for let's say five times i.e, one in each day then attendance is marked five times for that particular student. Like this other student attendance is marked according to their presence in that particular day.

In Figure below, first row showing 1×4 cell, which means the particular student is present for 1 day with time. The first column is used for storing the id of the student and second row is used to storing the name of the student and so the next two columns is used for marking the attendance by date and time. Similarly the same is applicable for other raw of the 11×2 cell.

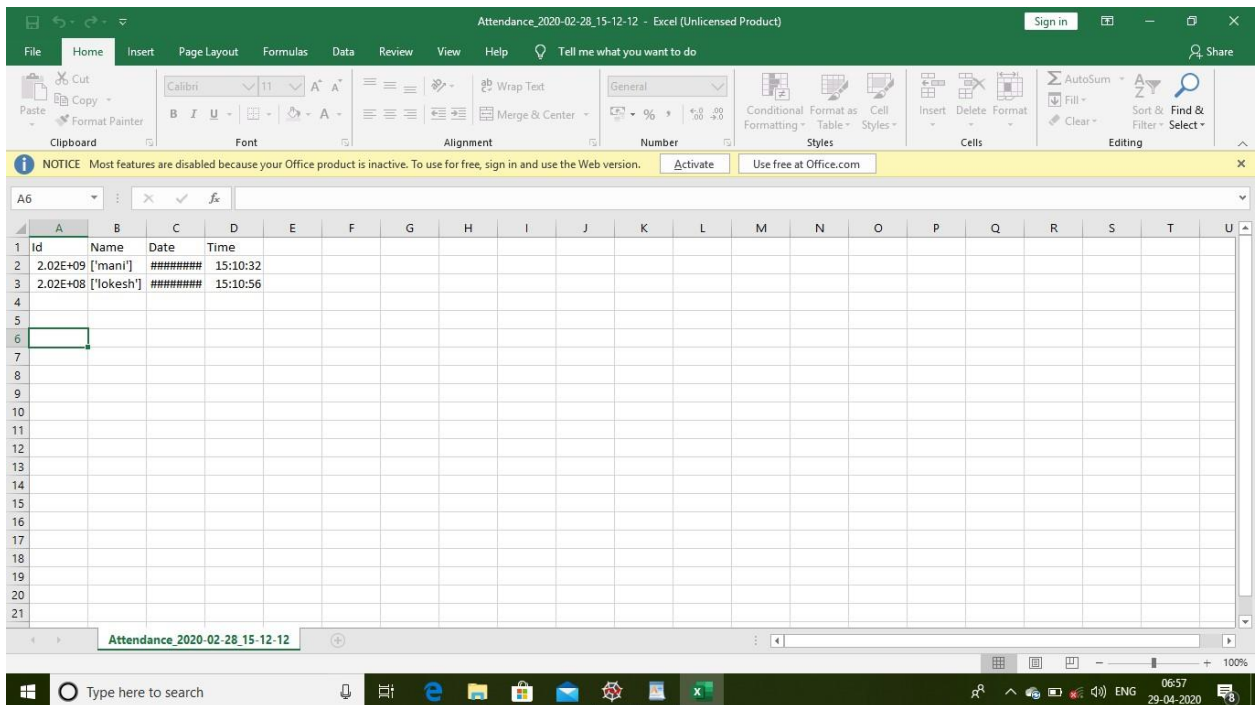


Figure 4.2: Attendance Marked

Take another example of no 2, which is showing 2×4 cell, which means that particular student is present for a days. The marked attendance of the student in the second row cell is shown in Figure below.

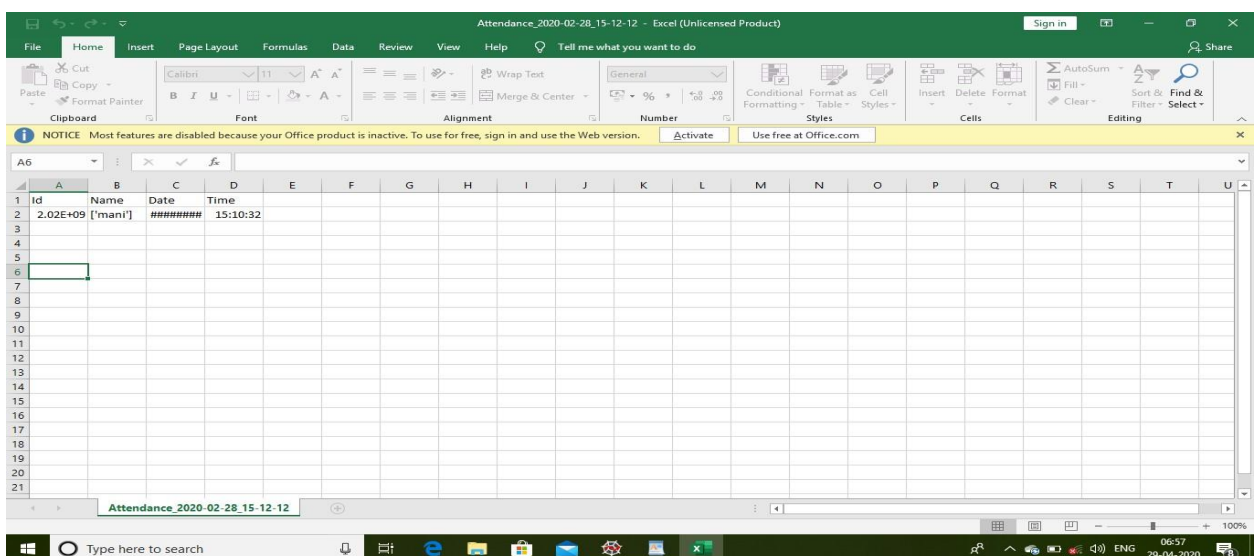


Figure 4.3: Second row cell attendance

4.2 DATABASE

Database holds the name and ID of all people whose attendance will have to be marked. As and when a face is detected and matched with the existing records, the attendance is automatically updated in the database. Every time a face is recognized, the number of times that person is present increases by a count of 1. There is also a column which shows when the attendance was last updated. And also the data base holds the image of the person who as registered in this system. When the face recognition occurs the faces which all are detected by the camera using the rectangle frames are stored inside the database. Detected faces are marked as a present with name, time and date. The unknown faces which are tried to detected are stored in the database as an unknown images. Later it can be used for other purpose. The unknown image storing process is more useful for police and management to know that unverified person was available inside the place/ classroom. Database stored data are shown in the give figure below.

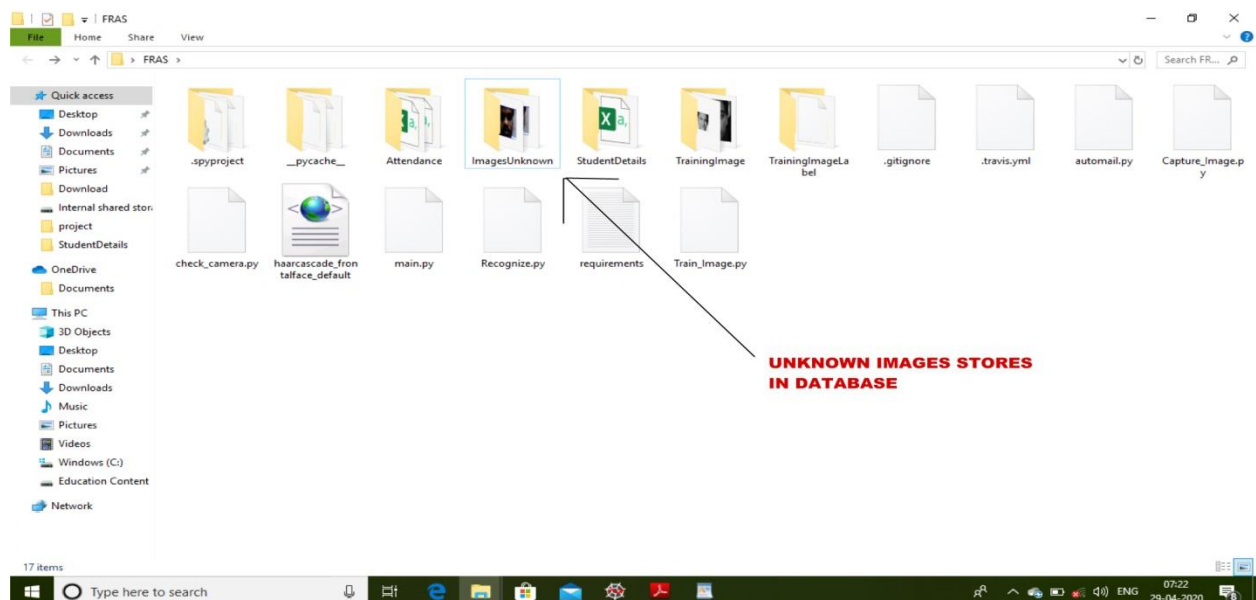


Figure 4.4: Database stored data

4.3 TEST CASE

Testcase are mentioned below in table are used to test the facial recognition.

S No.	Action	Inputs	Expected Output	Actual Output	Test Result
1	Capture Images	A person's face	Images are captured and stored	Images are captured and stored	Pass
2	Train the image dataset	Stored images of a face	Create histogram and store values	Histogram in created and values are stored	Pass
3	Face Recognition	A live stream of a person's face	Name of detected person is displayed on the screen	Name of detected person is displayed on the screen	Pass
4	Update attendance for multiple people at once	Multiple faces from a Live video stream	Update Attendance for all faces detected	Updated attendance for all faces detected	Pass
5	Detect more than 7 faces	7 people facing present in front of camera	Detect all 7 faces present in front of camera	Only 5 faces are detected at a time	Fail

CHAPTER 5

CONCLUSION AND FUTURE WORK

The Aim of this paper is to make an attendance registration when a user scans their face within the specified location. To do this we capture the image of the students, convert it into grayscale frames, relate it with the database to know their present and absent, marking attendance for the particular student to maintain the records. The Attendance system using Face Authentication helps to increase the accuracy and speed to achieve the real-time attendance to meet the need for these days.

The detection and recognition algorithms were studied thoroughly taking number of the test from different varying condition images. For face detection combination of RGB to grayscale and LBPH algorithm is used. For face recognition principal component analysis method is used. Attendance of the student are marked using the recognized face of every individual student and the data is stored in an attendance spreadsheet. The attendance of every student marked automatically by recognizing their face with the face present in the database.

The system estimates the attendance of each student by continuous observation and recording. The result of our experiment shows continuous observation improved the performance for estimation of the attendance. Automated Attendance System has been envisioned for the use of reducing the mistakes that happens in the traditional(manual) attendance taking system. This is to automate and make a system that is useful to the organization such as an institute. The efficient and accurate method of attendance in the institute environment that can replace the old manual works of method. This Method is secured enough, reliable and available for use.

FUTURE WORK

The future work can be done on this project it to alert the students by sending SMS or mail about the Attendance. For this work GSM module can be used. SMS alert can be given to the parents also. So the students can't able to cheat their parents also. Then the accuracy level also need to increase for multi face detection, detecting the 5 and above images are required to improve more.

REFERENCES

1. Shireesha Chintalapati, M.V. Raghunadh, "Automated Attendance Management System Based On Face Recognition Algorithms", IEEE International Conference on Computational Intelligence and Computing Research, 2013.
2. Kosov S., Scherbaum K., Faber K., Thormahlen T. and Seidel H.-P., "Rapid stereo-vision enhanced face detection", Image Processing (ICIP), 2009 16th IEEE International Conference on, IEEE, 1221-1224 (2009).
3. Levada A., Correa D., Salvadeo D., Saito J. and Mascarenhas N., "Novel approaches for face recognition: templatematching using dynamic time warping and LSTM NeuralNetwork Supervised Classification", Systems, Signals and Image Processing, 2008. IWSSIP 2008. 15th International Conference on, IEEE, 241-244 (2008).
4. R. Bruneli and T. Poggio, "Face recognition: features versus templates," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 15, pp. 1042-1052, 1993.
5. V. Blanz and T. Vetter, "Face recognition based on fitting a 3D morphable model," IEEE Transactions On Pattern Analysis and Machine Intelligence, vol. 25, no. 9, September 2003.

6. Y. Gao and K.H. Leung, "Face recognition using line edge map," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 6, June 2002.
7. Michal Dolecki, Pawel Karczmarek, Adam Kiersztyn, Witold Pedrycz , "Face recognition by humans performed on basis of linguistic and neural network", Neural Networks (IJCNN)2016 International Joint Conference on , pp.5135-5140,2016,ISSN 2161-4407.
8. Ming Du, Aswin C.Sankaranarayanan, and Rama Chellappa , Robust Face Recognition from Multi-View Videos, IEEE "Transactions on Image Processing", Vol.23, No. 3,March 2014.
9. Kyunghnam Kim "Face Recognition using Principle Component Analysis", Department of Computer Science, University of Maryland, College Park, MD 20742, USA.
10. Javier Ruiz Del Solar, Rodrigo Verschae, and Mauricio Correa. Face recognition in unconstrained environments: A comparative study. In ECCV Workshop on Faces in RealLife Images: Detection, Alignment, and Recognition, Marseille, France, October 2008
11. H.K.Ekenel and R.Stiefelhagen,Analysis of local appearance based face recognition: Effects of feature selection and feature normalization. In CVPR Biometrics Workshop, New York, USA, 2006.
12. Goldstein, A.J., Harmon, L.D., and Lesk,A.B.(1971). Identification of human faces. In Proc. IEEE, Vol.59, page 748.
13. A.F. Abate, M. Nappi, D. Riccio, and G. Sabatino, "2D and 3D face recognition: A survey", Pattern Recognition Letters, vol.28, issue 15, pp.1885-1906, Oct 2007.
14. K. Wong, H. Law, and P. Tsang, "A System for Recognising Human Faces," Proc. ICASSP, pp. 1,638-1,642, 1989.

15. N. Dalal, B. Triggs "Histograms of oriented gradients for Human Detection", IEEE Computer Society Conference on Computer Vision and Pattern Recognition , Vol. 1, 2005, pp. 886 – 893.
16. S.Edelman, D.Reisfeld, and Y. Yeshurun, "A System for Face Recognition that Learns from Examples," Proc. European Conf. Computer Vision, S. Sandini, ed., pp. 787-791. Springer- Verlag, 1992.
17. P. Hallinan, "A Low-Dimensional Representation of Human Faces for Arbitrary Lighting Conditions," Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 995-999, 1994.
18. W.Zhao, R.Chellappa, P. J. Phillips, A. Rosenfeld, "Face Recognition: A Literature Survey." ACM Computing Surveys, 2003, vol. 35, no. 4, pp. 399-458.
19. L. Zhao and X. Xu, "Distance function for face recognition based on 2D PCA," 2011 Int. Conf. Multimed. Technol., pp. 5814-5817, 2011.
20. Nasser, M. A. et al. (2011). "Automated facial recognition attendance system." Graduation Project, Petra University, January 2011
21. RoshanTharanga, J. G., et al. "Smart Attendance Using Real Time Face Recognition (SMAR-FR)." Department of Electronic and Computer Engineering, Sri Lanka Institute of Information Technology (SLIIT), Malabe, Sri Lanka.
22. D. Cahi and K.N. Ngan, "Face segmentation using skin color map", IEEE Trans. On Circuit and Systems for Video Technology, vol 9, no 4, pp. 551-564, Jun. 1999.

APPENDIX

A.SOURCE CODE

```

import os # accessing the os functions
import check_camera
import Capture_Image
import Train_Image
import Recognize

# creating the title bar function

def title_bar():
    os.system('cls') # for windows

    # title of the program

    print("\t*****")
    print("\t***** Face Recognition Attendance System *****")
    print("\t*****")

# creating the user main menu function

def mainMenu():
    title_bar()
    print()
    print(10 * "*", "WELCOME MENU", 10 * "*")
    print("[1] Check Camera")
    print("[2] Capture Faces")
    print("[3] Train Images")
    print("[4] Recognize & Attendance")
    print("[5] Auto Mail")
    print("[6] Quit")

    while True:
        try:
            choice = int(input("Enter Choice: "))

            if choice == 1:
                checkCamera()
                break
            elif choice == 2:
                CaptureFaces()
                break
            elif choice == 3:
                Trainimages()
                break

```

```

        elif choice == 4:
            RecognizeFaces()
            break
        elif choice == 5:
            os.system("py automail.py")
            break
            mainMenu()
        elif choice == 6:
            print("Thank You")
            break
        else:
            print("Invalid Choice. Enter 1-4")
            mainMenu()
    except ValueError:
        print("Invalid Choice. Enter 1-4\n Try Again")
    exit

# -----
# calling the camera test function from check camera.py file

def checkCamera():
    check_camera.camera()
    key = input("Enter any key to return main menu")
    mainMenu()

# -----
# calling the take image function form capture image.py file

def CaptureFaces():
    Capture_Image.takeImages()
    key = input("Enter any key to return main menu")
    mainMenu()

# -----
# calling the train images from train_images.py file

def Trainimages():
    Train_Image.TrainImages()
    key = input("Enter any key to return main menu")
    mainMenu()

# -----
# calling the recognize_attendance from recognize.py file

```

```
def RecognizeFaces():  
    Recognize.recognize_attendance()  
    key = input("Enter any key to return main menu")  
    mainMenu()
```

```
# -----main driver -----  
mainMenu()
```

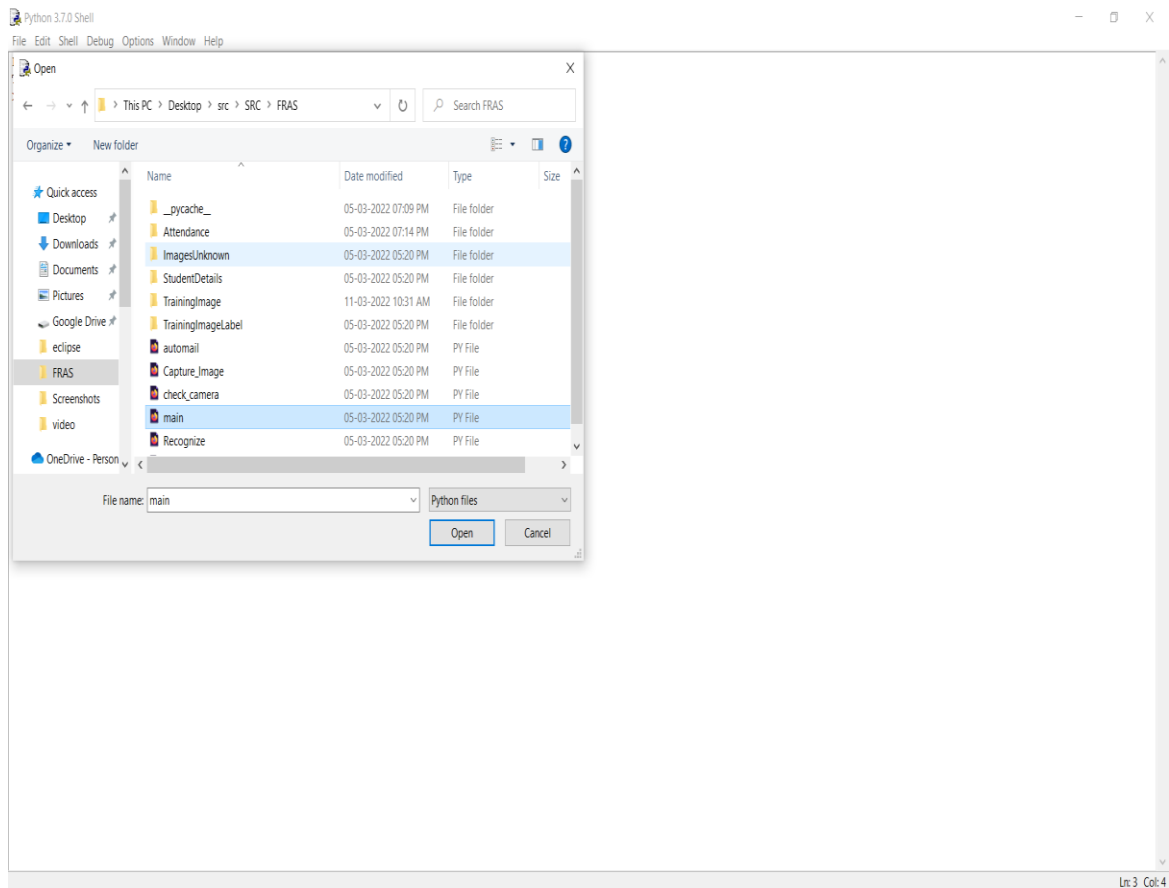
B.SCREENSHOT

PYTHON IDLE

IMPORT PACKAGES

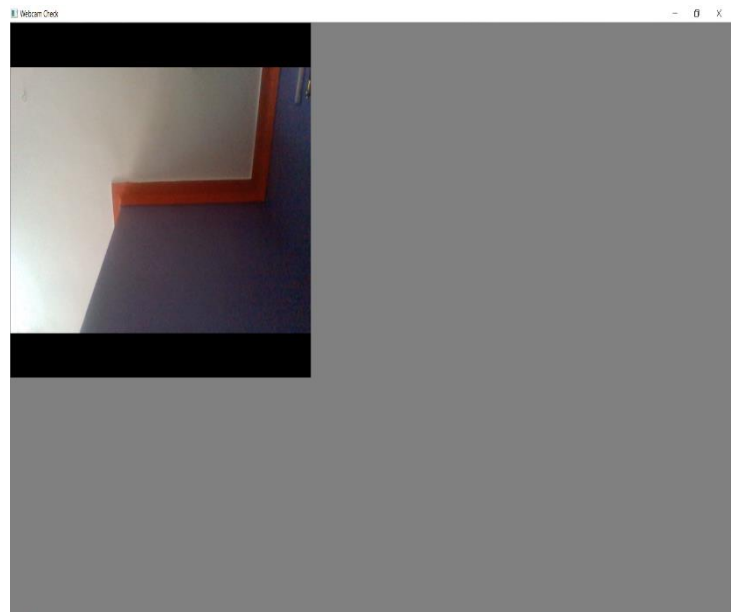


TO IMPORT THE PACKAGES



CHECK CAMERA

```
Python370Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (tags/v3.7.0:1b00f50b1, Jun 27 2019, 04:59:51) [MSC v.1914 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\VTISHA\OneDrive\OneDrive\Python\main.py =====
***** Face Recognition Attendance System *****
***** WELCOME MENU *****
[1] Check Camera
[2] Capture Faces
[3] Train Images
[4] Recognize & Attendance
[5] Auto Mail
[6] Quit
Enter Choice: 1
```



CAPTURE IMAGES

```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\VISHAL\Desktop\src\SRC\FRAS\main.py =====
***** Face Recognition Attendance System *****

***** WELCOME MENU *****
[1] Check Camera
[2] Capture Faces
[3] Train Images
[4] Recognize & Attendance
[5] Auto Mail
[6] Quit
Enter Choice: 1

===== RESTART: C:\Users\VISHAL\Desktop\src\SRC\FRAS\main.py =====
***** Face Recognition Attendance System *****

***** WELCOME MENU *****
[1] Check Camera
[2] Capture Faces
[3] Train Images
[4] Recognize & Attendance
[5] Auto Mail
[6] Quit
Enter Choice: 2
Enter Your Id: 11
Enter Your Name: |
```

TRAIN IMAGES

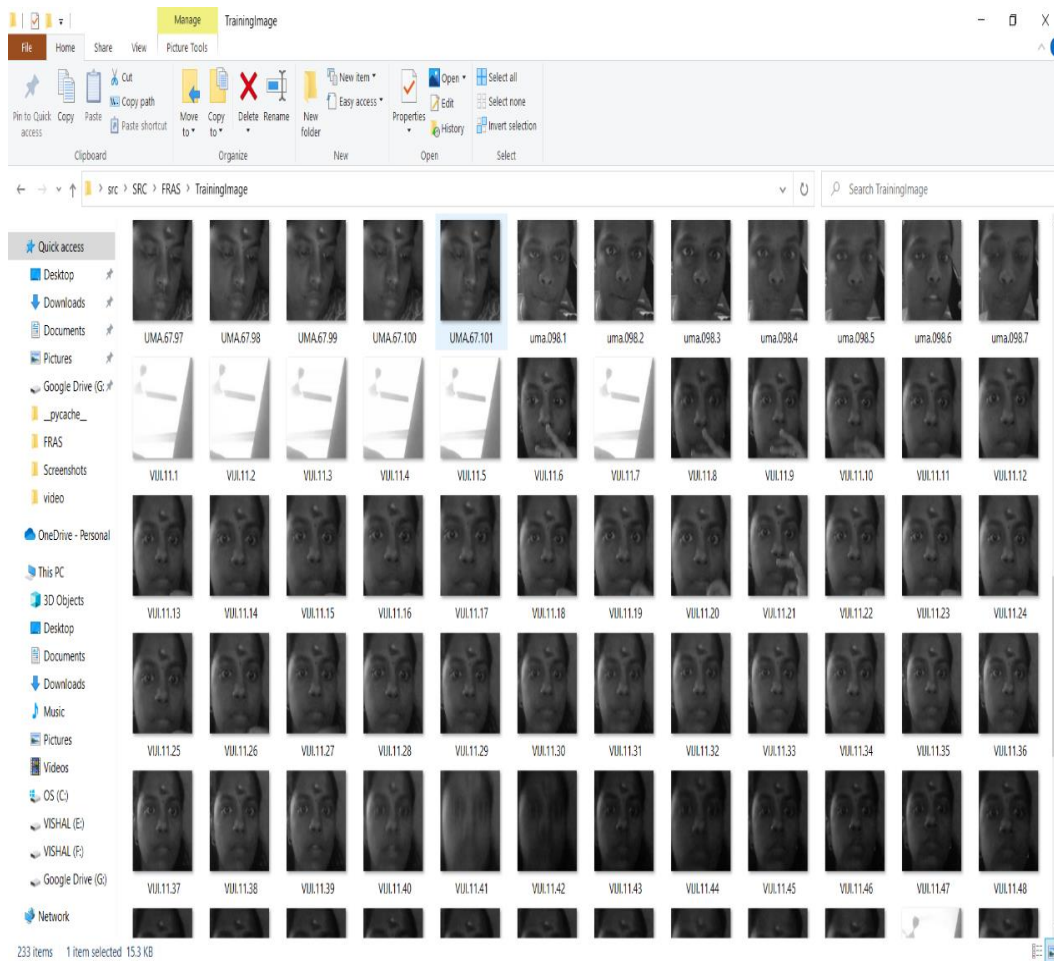
```
*Python 3.7.0 Shell*
File Edit Shell Debug Options Window Help
C:\Users\VISHAL\Desktop\src\SRC\FRAS\main.py
Images Trained44 Images Trained45 Images Trained46 Images Trained47 Images Trained48 Images Trained49 Images Trained50 Images Trained51 Images Trained52 Images Trained53 Images Trained54 Images Trained55 Images Trained56 Images Trained57 Images Trained58 Images Trained59 Images Trained60 Images Trained61 Images Trained62 Images Trained63 Images Trained64 Images Trained65 Images Trained66 Images Trained67 Images Trained68 Images Trained69 Images Trained70 Images Trained71 Images Trained72 Images Trained73 Images Trained74 Images Trained75 Images Trained76 Images Trained77 Images Trained78 Images Trained79 Images Trained80 Images Trained81 Images Trained82 Images Trained83 Images Trained84 Images Trained85 Images Trained86 Images Trained87 Images Trained88 Images Trained89 Images Trained90 Images Trained91 Images Trained92 Images Trained93 Images Trained94 Images Trained95 Images Trained96 Images Trained97 Images Trained98 Images Trained99 Images Trained100 Images Trained101 Images Trained102 Images Trained103 Images Trained104 Images Trained105 Images Trained106 Images Trained107 Images Trained108 Images Trained109 Images Trained110 Images Trained111 Images Trained112 Images Trained113 Images Trained114 Images Trained115 Images Trained116 Images Trained117 Images Trained118 Images Trained119 Images Trained120 Images Trained121 Images Trained122 Images Trained123 Images Trained124 Images Trained125 Images Trained126 Images Trained127 Images Trained128 Images Trained129 Images Trained130 Images Trained131 Images Trained132 Images Trained133 Images Trained134 Images Trained135 Images Trained136 Images Trained137 Images Trained138 Images Trained139 Images Trained140 Images Trained141 Images Trained142 Images Trained143 Images Trained144 Images Trained145 Images Trained146 Images Trained147 Images Trained148 Images Trained149 Images Trained150 Images Trained151 Images Trained152 Images Trained153 Images Trained154 Images Trained155 Images Trained156 Images Trained157 Images Trained158 Images Trained159 Images Trained160 Images Trained161 Images Trained162 Images Trained163 Images Trained164 Images Trained165 Images Trained166 Images Trained167 Images Trained168 Images Trained169 Images Trained170 Images Trained171 Images Trained172 Images Trained173 Images Trained174 Images Trained175 Images Trained176 Images Trained177 Images Trained178 Images Trained179 Images Trained180 Images Trained181 Images Trained182 Images Trained183 Images Trained184 Images Trained185 Images Trained186 Images Trained187 Images Trained188 Images Trained189 Images Trained190 Images Trained191 Images Trained192 Images Trained193 Images Trained194 Images Trained195 Images Trained196 Images Trained197 Images Trained198 Images Trained199 Images Trained200 Images Trained201 Images Trained202 Images Trained203 Images Trained204 Images Trained205 Images Trained206 Images Trained207 Images Trained208 Images Trained209 Images Trained210 Images Trained211 Images Trained212 Images Trained213 Images Trained214 Images Trained215 Images Trained216 Images Trained217 Images Trained218 Images Trained219 Images Trained220 Images Trained221 Images Trained222 Images Trained223 Images Trained224 Images Trained225 Images Trained226 Images Trained227 Images Trained228 Images Trained229 Images Trained230 Images Trained231 Images Trained232 Images Trained233 Images TrainedAll Images
Enter any key to return main menu
***** Face Recognition Attendance System *****

***** WELCOME MENU *****
[1] Check Camera
[2] Capture Faces
[3] Train Images
[4] Recognize & Attendance
[5] Auto Mail
[6] Quit
Enter Choice: 4

===== RESTART: C:\Users\VISHAL\Desktop\src\SRC\FRAS\main.py =====
***** Face Recognition Attendance System *****

***** WELCOME MENU *****
[1] Check Camera
[2] Capture Faces
[3] Train Images
[4] Recognize & Attendance
[5] Auto Mail
[6] Quit
Enter Choice:
```

Ln: 70 Col: 26



TRAINED IMAGES

RECOGINZE AND ATTENDANCE

Attendance_2019-08-24_13-34-43 - Excel

VISHAL LOGANATHAN

File Home Insert Page Layout Formulas Data Review View Help Tell me what you want to do

Clipboard Font Alignment Number Styles Cells Editing

POSSIBLE DATA LOSS Some features might be lost if you save this workbook in the comma-delimited (.csv) format. To preserve these features, save it in an Excel file format. Don't show again Save As...

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	Id	Name	Date	Time																			
2	2	['Jacky']	8/24/2019	13:31:55																			
3	3	['Amir']	8/24/2019	13:32:11																			
4	4	['Tom']	8/24/2019	13:32:19																			
5	1	['Mubin']	8/24/2019	13:32:38																			
6	6	['Aksay']	8/24/2019	13:33:05																			
7	12	['Bruce']	8/24/2019	13:34:09																			
8	13	['Jonny']	8/24/2019	13:34:17																			
9	11	['Kasfia']	8/24/2019	13:34:23																			
10																							
11																							
12																							
13																							
14																							
15																							
16																							
17																							
18																							
19																							
20																							
21																							
22																							
23																							
24																							
25																							
26																							
27																							
28																							

Attendance_2019-08-24_13-34-43

Ready Accessibility: Unavailable