

DOI: 10.17725/rensit.2020.12.297

Automated Attendance Machine Using Face Detection and Recognition System

Muhanned AL-Rawi

Ibb University, <https://www.ibbuniv.edu.ye/>

Ibb, Yemen

E-mail: mubrawi@yahoo.com

Received February 27, 2020, reviewed March 02, 2020, accepted March 23, 2020

Abstract. This paper serves to automate the prevalent traditional tedious and time wasting methods of marking student attendance in classrooms. The use of automatic attendance through face detection and recognition increases the effectiveness of attendance monitoring and management. This method could also be extended for use in examination halls to curb cases of impersonation as the system will be able to single out the imposters who won't have been captured during the enrollment process. Applications of face recognition are widely spreading in areas such as criminal identification, security systems, image and film processing. The system could also find applications in all authorized access facilities.

Keywords: automated attendance machine; face; detection and recognition

UDC 004.931

For citation: Muhanned AL-Rawi. Automated Attendance Machine Using Face Detection and Recognition System. *RENSIT*, 2020, 12(2)297-304. DOI: 10.17725/rensit.2020.12.297.

CONTENTS

1. INTRODUCTION (297)
2. METHODOLOGY AND DESIGN (298)
 - 2.1. SYSTEM DESIGN (298)
 - 2.2. GENERAL OVERVIEW (298)
 - 2.3. TRAINING SET MANAGER SUBSYSTEM (298)
 - 2.4. FACE RECOGNIZER SUBSYSTEM (298)
 - 2.5. FULL MOBILE MODULE LOGICAL DESIGN (298)
 - 2.6. SYSTEM ARCHITECTURE (299)
 - 2.7. FUNCTIONS OF THE TWO SUBSYSTEMS (299)
 - 2.8. FULL SYSTEMS LOGICAL DESIGN (299)
 - 2.9. TOOLS (299)
3. RESULTS AND ANALYSIS (299)
 - 3.1. USER INTERFACE OF THE SYSTEM (299)
 - 3.1.1. FACES DATABASE EDITOR (299)
 - 3.1.2. THE FACE RECOGNIZER (300)
 - 3.2. FACE DETECTION (300)
 - 3.3. FACE RECOGNITION (301)
4. CONCLUSION (303)
- REFERENCES (304)

1. INTRODUCTION

Maintaining attendance is very important in all learning institutes for checking the performance of students. In most learning institutions, student

attendances are manually taken by the use of attendance sheets issued by the department heads as part of regulation. The students sign in these sheets which are then filled or manually logged in to a computer for future analysis. This method is tedious, time consuming and inaccurate as some students often sign for their absent colleagues. This method also makes it difficult to track the attendance of individual students in a large classroom environment [1,2]. In this paper, we propose the design and use of a face detection and recognition system to automatically detect students attending a lecture in a classroom and mark their attendance by recognizing their faces.

While other biometric methods of identification (such as iris scans or fingerprints) can be more accurate, students usually have to queue for long at the time they enter the classroom [2,3]. Face recognition is chosen owing to its non-intrusive nature and familiarity as people primarily recognize other people based on their facial features. This (facial) biometric system consists of an enrollment process in which the unique features of a persons' face is stored in a

database and then the processes of identification and verification. In these, the detected face in an image (obtained from the camera) is compared with the previously stored faces captured at the time of enrollment [4,5,6].

In this paper, we are setting up to design a system comprising of two modules. The first module (face detector) is a mobile component, which is basically a camera application that captures student faces and stores them in a file using computer vision face detection algorithms and face extraction techniques. The second module is a desktop application that does face recognition of the captured images (faces) in the file, marks the students register and then stores the results in a database for future analysis.

2. METHODOLOGY AND DESIGN

2.1. SYSTEM DESIGN

In this design, several related components in terms of functionality are grouped to form subsystems which then combine to make up the whole system. Breaking the system down to components and subsystems informs the logical design of the class attendance system.

2.2. GENERAL OVERVIEW

The flow diagram of Fig. 1 depicts the systems operation. From Fig.1, it can be observed that most of the components utilized are similar;(the

image acquisition component for browsing for input images, the face detector and the faces database for storing the face label pairs) only that they are employed at the different stages of the face recognition process.

2.3. TRAINING SET MANAGER SUBSYSTEM

The logical design of the training set management subsystem is going to consist of an image acquisition component, a face detection component and a training set management component. Together, these components interact with the faces database in order to manage the training set. These are going to be implemented in a windows application form.

2.4. FACE RECOGNIZER SUBSYSTEM

The logical design of the face recognizer consists of the image acquisition component, face recognizer and face detection component all working with the faces database. In this, the image acquisition, and face detection component are the same as those in the training set manager sub system as the functionality is the same. The only difference is the face recognizer component and its user interface controls. This will load the training set again so that it trains the recognizer on the faces added and show the calculated Eigen faces and average face. It should then show the recognized face in a picture box.

2.5. FULL MOBILE MODULE LOGICAL DESIGN

This android application module which is shown in Fig. 2 consists of a camera component, android face detector component and a SQLite database component to store the detected images. The android face detector and camera components work to detect a face from the camera input image. The image is then captured and saved in the SQLite database. This is retrieved by the image acquisition component of the desktop module.

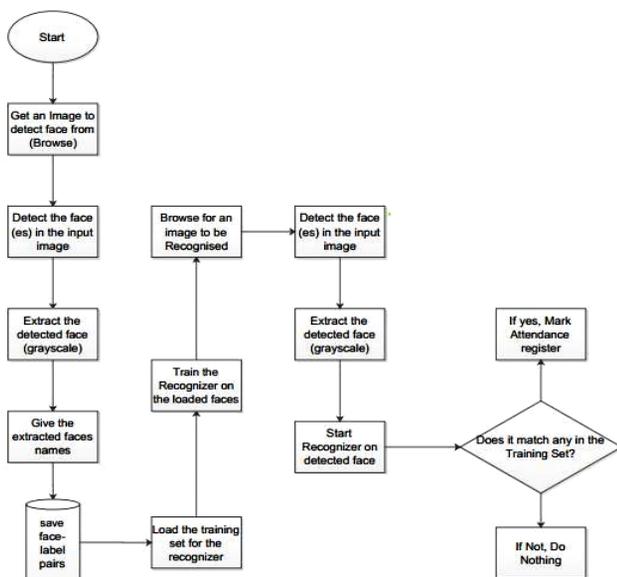


Fig. 1. Sequence of events in the class attendance system.



Fig. 2. Logical design of the mobile module.

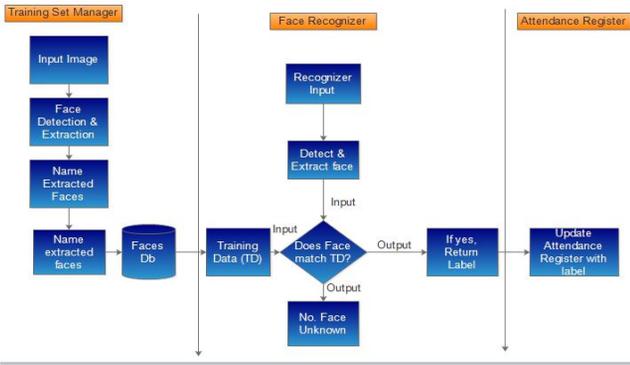


Fig. 3. The logical design of the desktop module subsystems.

2.6. SYSTEM ARCHITECTURE

Fig. 3 below shows the logical design and implementation of the three desktop subsystems

2.7 FUNCTIONS OF THE TWO SUBSYSTEMS

The functionalities of the components are depicted in the block diagrams of Fig. 4. The face recognizer system consists of two major components i.e. the training set manager and the face recognizer. These two components share the faces database, the image acquisition and the face detector components; as they are common in their functionality.

We therefore partition the system in to two subsystems and have their detailed logical designs to be implemented.

2.8. FULL SYSTEMS LOGICAL DESIGN

The logical design of the whole system is shown in Fig. 5.

2.9. TOOLS

These tools are used in the implementation of the designed system. They’ve been divided in to two categories; Mobile and Desktop tools.

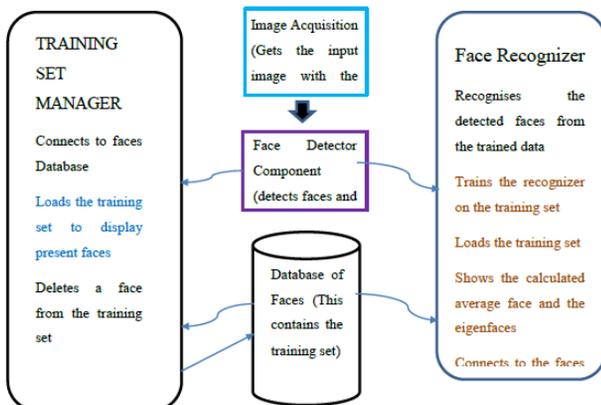


Fig. 4. Block diagram showing functions of the components.

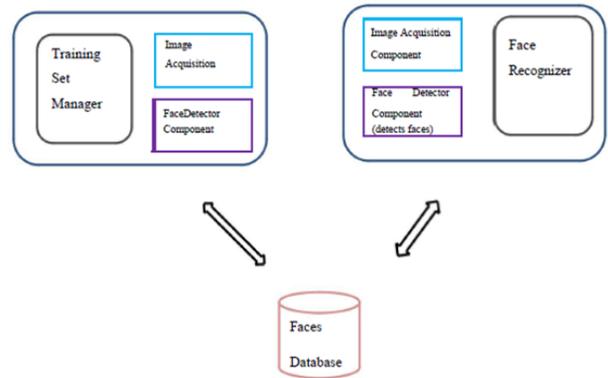


Fig. 5. Logical design of the whole system.

The mobile tools are the components that aid in the implementation of the mobile module. This module is responsible for capturing the students’ images in a classroom environment and then storing them for further processing by the desktop module. The desktop tools are components; hardware or software that are utilized in the actual development of the desktop module. The desktop module also connects to the class attendance register which is implemented as a database management system.

3. RESULTS AND ANALYSIS

3.1. USER INTERFACE OF THE SYSTEM

3.1.1. FACES DATABASE EDITOR

The faces database editor adds faces in the training set. The image is acquired from the highlighted box number 1 as shown in Fig. 6 and displayed as is on step 2 on a picture box. The Regions of Interest (ROI) i.e. faces in the image is then automatically detected by drawing a light green rectangular box. In step 3, we give the extracted grayscale face from the image a face label and then add them to the training

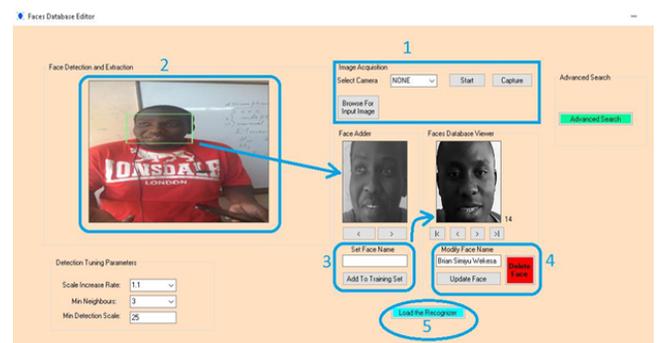


Fig. 6. The training set editor.

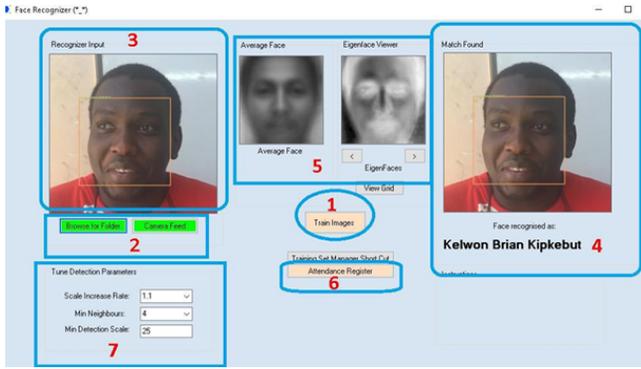


Fig. 7. The face recognizer.

set. In step 4, we can then modify the face label pairs in the event they are wrongly captured or even delete the faces if they are not as per the standards. Finally step 5 prepares us for the recognition stage.

3.1.2. THE FACE RECOGNIZER

The face recognizer compares the input face in the image captured with the faces captured during enrollment. If it is a match, it then retrieves the name associated with the input face.

Step 1 is to train the recognizer to be able to identify a face as either known or unknown. Step 2 selects the source of the image with the face to be recognized. This could be from a live camera feed or a folder with captured images. The input image with the face is then displayed in the recognizer picture box 3 as shown in Fig. 7. The name of the input face in the image is then displayed as shown in Step 4. The returned name of the input face, date and time are then utilized in populating the records in the attendance

Table 1

Attendance register.

StudentID	StudentName	Time
148	subject07.noglasses.	4/1/2016 12:45 PM
149	subject08_2	4/1/2016 4:50 PM
150	subject04_7	4/1/2016 4:51 PM
151	subject05.wink.	4/1/2016 4:51 PM
152	subject01.normal.	4/1/2016 4:51 PM
153	subject10_6	4/1/2016 4:51 PM
154	subject02.centerlight.	4/1/2016 4:51 PM
155	subject02_8	4/1/2016 6:48 PM
156	subject08_2	4/1/2016 6:49 PM
157	subject04_7	4/1/2016 6:49 PM
158	subject07.noglasses.	4/1/2016 6:49 PM
159	Brian Kelwon Kipkebut	4/2/2016 10:10 AM
160	Owuor Oloo 6	4/2/2016 10:28 AM
161	subject01.normal.	4/2/2016 10:28 AM
162	subject02_8	4/2/2016 10:28 AM
163	subject08_2	4/2/2016 10:28 AM

register database. Clicking the button of step 6 displays the register as shown in Table 1. The highlighted step 5 displays the computed average and Eigen faces. The arrows are used to navigate through the Eigen faces. The “View Grid” button displays the Eigen faces/vectors that had been computed from the covariance matrix in a grid form.

Selecting camera feed as the source of the input image pops up the window of Fig. 8. The images in the video feed are automatically detected, tracked and recognized. Images can also be added to the database from the live camera feed.

From Fig. 8, the highlighted box 1 shows the current camera view/scene. The faces and eyes in the images are automatically detected as indicated by the rectangular boxes around them. The detected face is extracted and compared with those in the database. Upon a successful match, the name associated with the face is then displayed on the upper edge of the rectangular box. The number of faces in the scene as well as their corresponding names are also shown on the highlighted box number 2. The face adder box 3 can also be used to add faces to the database.

3.2. FACE DETECTION

For group photos, a minimum neighbors’ detection tuning parameter of 3 yields the best overall performance as indicated in Fig. 9 where the physical count is 53.

The face marked by a red hexagon is not detected in the minimum neighbors’ setting of 4. This is because the face is not fully displayed. Four is the highest setting which strictly returns

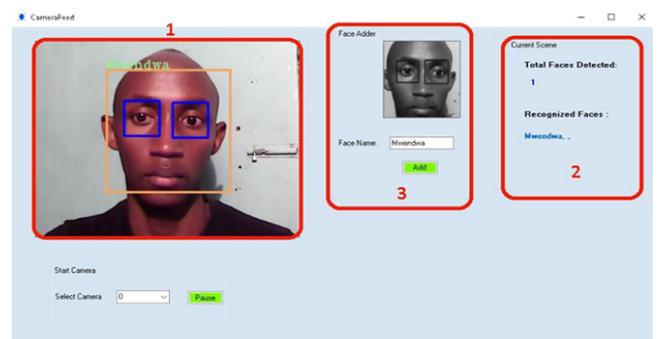


Fig. 8. The live camera feed window.

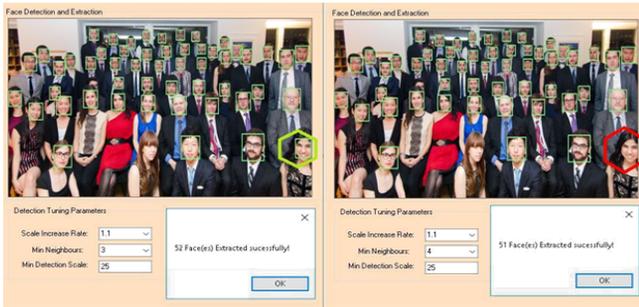


Fig. 9. Comparison between minimum neighbors setting of three and four.

frontal images. The second lady on the first row is not detected in either of the settings because her face is skewed to the right. The face detector only works with frontal images. 52 out of 53 images are successfully detected.

Fig. 10 shows a group photo with a minimum neighbors setting of 1 and 2. Tuning the minimum neighbors setting to 1 returns the number of faces in the images as 8; different from the physical count of 5. This is because the detector returned the slightest resemblance to a face as an actual face and hence the three face detections marked in red circles as shown in Fig. 10. Using the same image from class and incrementing the setting to 2 returned the number of detected faces as 5 which corresponded with the physical count. Increasing the setting further to 4 reduced

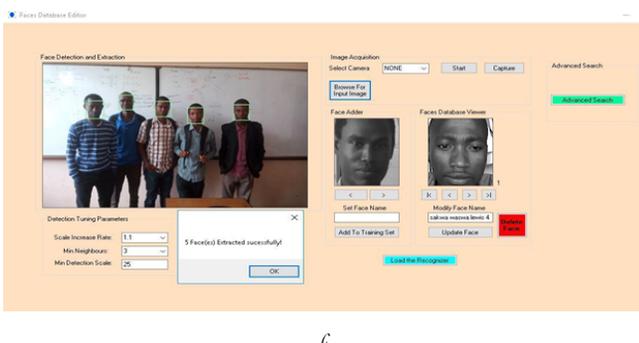
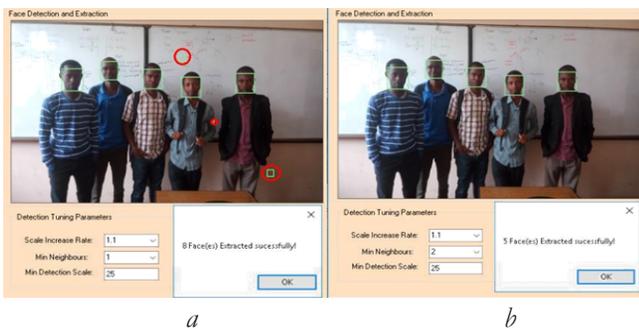


Fig. 10. Minimum neighbors setting of 1, 2 and 3 respectively on an image from class.

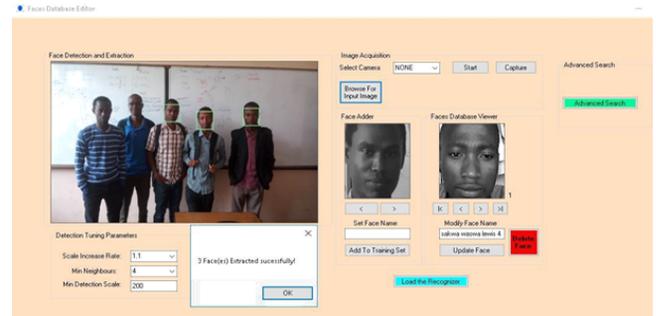


Fig. 11. Minimum detection scale of 200.

the number of detected faces to three.

A minimum detection scale of 25 had the best overall performance for very large group photos in terms of speed. Increasing the scale to 200 as shown in Fig. 11 tremendously reduces the time taken to return the number of faces in an image. The minimum detection scale also makes it possible to be able to detect and recognize faces over longer and shorter distances of recognition by decreasing and increasing the scale respectively. Low detection scales waste central processing unit cycles if the size of the faces in the image is large.

The system had 100% face detection rate for different frontal faces; local as well as faces from standard faces databases like the Yale faces. The system is also able to detect bearded faces as well as faces with glasses.

3.3. FACE RECOGNITION

In order to improve the recognition efficiency of the system, nine photos for each person from the standard Yale faces database are chosen for training, the remaining two photos are chosen for the testing set. Out of the fifteen subjects from the Yale faces database, twelve faces are correctly recognized. This is proportional to 80% accuracy. The faces of Fig. 12 are not properly recognized.

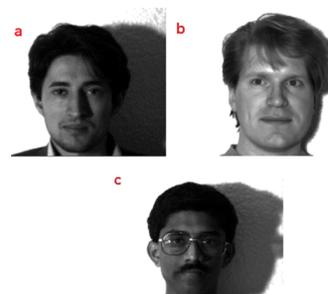


Fig. 12. The Yale database faces that were not properly recognized.

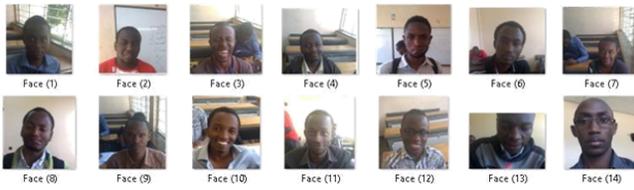


Fig. 13. Images from class.

Out of the fourteen faces of Fig. 13, ten are successfully recognized corresponding to a recognition accuracy of about 71.43%. The main cause of false recognition is the strength of the trained data and the illumination of the image. Face recognition is a form of machine learning and thus the larger and diverse the faces in the training set, the stronger the trained data used in recognizing faces.

Having several diverse faces of the same person with different facial expressions possible at the time of recognition creates strong training data and increases the accuracy of recognition. The lighting conditions present at the time of capturing the image to be recognized also affects the recognition results as is the case in Fig. 12 (a) and (c). Two closely identical people could also be recognized as one person unless the training data is strong.

Out of 60 faces in the database, tests are done for several subsets exclusive of the Yale dataset. The results obtained are tabulated as in Table 2. The percentage recognition rate is computed as the average of the percentages for the different subsets. Faces with or without glasses had no effect on the recognition rates. The mean percentage recognition rate is obtained to be 80.22%.

Table 2

Recognition results for various datasets

Dataset	No of Face	Successfully detected Faces	Successfully Recognized Faces	%Correct recognition
Center light	10	10	9	90
Left light	15	15	11	73.3
Right light	15	15	12	80
Veiled Faces	10	10	7	70
Bearded Faces	10	10	8	80
Unveiled A	20	20	17	85
Unveiled B	30	30	25	83.3

Center light faces had the best overall recognition rate at 90%. The primary issues facing most of the face detection and recognition systems that are in use today are rotation, pose, distance of recognition and illumination. These reduce the efficiency of the system unless performed under some necessary constraints. These constraints would involve positioning the subjects at specific positions, which in a real world classroom environment would be very hard and not to mention time consuming where the number of subjects involved is large.

With the help of a divergent combination of techniques and algorithms, this system helps us to achieve desired results with better accuracy. The provision of variable minimum detection scale eliminates the issue of distance for detection and recognition for both up close and group images. This has improved the face detection accuracy for upright frontal faces to 100% and consequently improved the face recognition accuracy from the typical efficiencies of 70%. Similarly, the minimum neighbors' setting has tremendously improved face detection accuracy.

Extracting and converting the rectangular part of the detected face instead of the whole image eliminates the effects of background noise on face detection improving the accuracy of the system. The camera in the system is used such that it only captures the frontal images so the problem of pose is not an issue. Histogram equalization is applied to the input images, this ensures that the output images are of uniform distribution of intensities through the reassignment of the intensity pixels. The

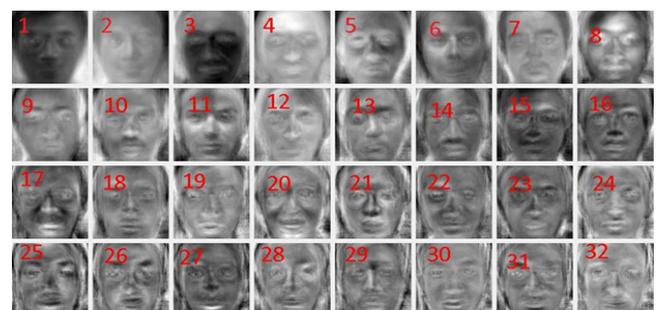


Fig. 14. The first 32 Eigen faces.

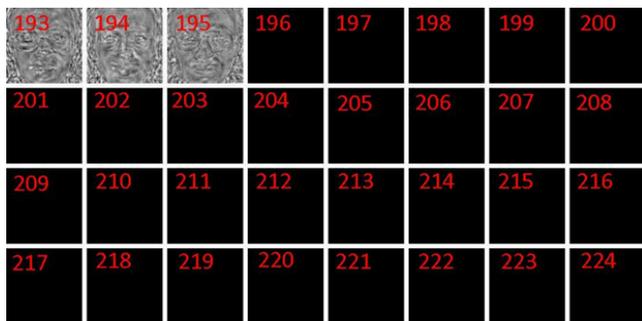


Fig. 15. *The last Eigen faces in the training set.*

input images of varying illumination are thus all enhanced in detail, this contributes into better face recognition results.

Fig. 14 shows the first 32 Eigen faces generated from a collection of 50 faces each of five people. The first few Eigen faces show dominant features of faces and the last Eigen faces from 196 to 247 are mainly image noise as shown in **Fig. 15** and are therefore discarded. The average face of **Fig. 16** obtained shows the smooth face structure of a generic human being.

From Figs. 14 and 15, it's seen that the first Eigen face shows the most dominant facial features of the training set images. The succeeding Eigen faces (principal components) in turn show the next highly probable facial features and more noise. Out of the 247 training images, 195 principal components together with the average face are enough to fully reconstruct the complete training set. We were therefore able to convert a set of correlated face variables (M) in to a set of values of K uncorrelated variables called principle components (eigenvectors). The number of Eigen faces is noted to be less than

the original face images i.e. $K < M$.

From the Eigen faces obtained in the face recognition stage, it is interesting to discover that the principal components analysis can be used for image compression as evidenced by the dominant number of Eigen faces that can comfortably represent all images in the training set. Out of 247 images in the training set, only 195 faces together with the average faces are required to fully reconstruct the 247 faces in the set.

4. CONCLUSION

It can be concluded that a reliable, secure, fast and an efficient class attendance management system has been developed replacing a manual and unreliable system. This face detection and recognition system save time, reduce the amount of work done by the administration and replace the stationery material currently in use with already existent electronic equipment.

There is no need for specialized hardware for installing the system as it only uses a computer and a camera. The camera plays a crucial role in the working of the system hence the image quality and performance of the camera in real time scenario must be tested especially if the system is operated from a live camera feed.

The system can also be used in permission based systems and secure access authentication (restricted facilities) for access management, home video surveillance systems for personal security or law enforcement.

The major threat to the system is Spoofing. For future enhancements, anti- spoofing techniques like eye blink detection could be utilized to differentiate live from static images in the case where face detection is made from captured images from the classroom. From the overall efficiency of the system i.e. 83.1% human intervention could be called upon to make the system foolproof. A module could thus be included which lists all the unidentified faces and the lecturer is able to manually correct them.



Fig. 16. *The average face.*

REFERENCES

1. Shehu V, Dika A. Using real time computer vision algorithms in automatic attendance management systems. *32nd International Conference on Information Technology Interfaces*, Cavtat, Croatia, 2010.
2. Gopala M et al. Implementation of automated attendance system using face recognition. *International Journal of Scientific & Engineering Research*, 2015, 6(3).
3. Varadharajan E. Automatic attendance management system using face detection. *Online International Conference on Green Engineering and Technologies*, India, 2016.
4. Jadhav A, Jadhav A, Ladhe T, Yeolekar K. Automated attendance system using face recognition. *International Research Journal of Engineering and Technology*, 2017, 4(1):1467-1471.
5. Paharekari S, Jadhav C. Automated attendance system in college using face recognition and NFC. *International Journal of Computer Science and Mobile Computing*, 2017, 6(6):14-21.
6. Godswill O. Automated student attendance management system using face recognition. *International Journal of Educational Research and Information Science*, 2018, 5(4):31-37.