An attendance system for exams using face recognition based on MTCNN and OpenCV algorithms

Khaled Elabbani[1], Ahmed Jafar[2] and Waleed Khalafullah[3]

Collage of Computer Technology Benghazi

elabani1980@yahoo.com[1] , ahmed.s.jaffar@gmail.com[2]

waleed.younus.kh@gmail.com[3]

الملخص

ساهمت تقنيات الذكاء الاصطناعي على تحسين أداء المؤسسات ورفع من كفاءتها، حيث تعتبر المصادقة والتحقق من الهوية من الاعتبارات الرئيسية للمؤسسات التعليمية ومراكز الامتحانات، في هذه الورقة تم اقتراح نظامًا آليًا قائمًا للتعرف على الوجوه عوضا عن الطرق التقليدية التي تستغرق وقتا وجهدا لحصر حضور الطلبة والتحقق من عدم انتحال الشخصية، حيث يستخدم النظام المقترح خوارزمية LBP للتعرف على الوجه وخوارزميات Haar-Cascade و MTCNN لاكتشاف الوجه. استخدمت لغة البايثون في تنفيد المقترح مع مكتبة OpenCV في تطبيق خوارزمية التعرف على الوجه .بالإضافة إلى ذلك، يقوم النظام بتوليد جداول بيانات حصر الطلبة بشكل آلي.

Abstract:

In access systems, authentication and identity verification are major considerations. Face recognition has been widely used in numerous applications and is helpful in validating biometrics. Most educational institutions and examination centers find it challenging to manage attendance using the traditional system. Institutions have been very concerned about student attendance, which takes a lot of time and effort to determine and is a laborious job with a high chance of error during the data entry into the computer. This paper proposed a face recognition-based automated system for storing records of attendance. The proposed system uses the LBP algorithm for facial recognition and the Haar-Cascade and MTCNN algorithms for face detection. Additionally, the system creates a data table to track students' attendance along with a request for access from the student.

Keywords: Face detection, face recognition, LBP, MTCNN, OpenCV, Haar- Cascade

1. Introduction

Recently, the field of artificial intelligence has taken on great importance in scientific research, especially machine learning and deep learning, and the development of this field has rapidly been associated with computer vision. One of the main areas in the field of computer vision is object detection and face recognition, and it attracted many studies and used in various applications, especially mobile applications [1]. Face recognition technology can be used in many areas of protection systems such as monitoring in airports and banks, also in the field of health by monitoring patients, in addition to education in the student attendance system and identification instead of using traditional methods. Face recognition still exists and has remained a major focus due to the use of many different algorithms and the testing of their efficiency [2].

Face recognition applications still help greatly, accurately and efficiently, due to the recent development of machine learning and neural networks. The attendance monitoring system is a very important process in all organizations and institutions, several facial recognition attendance systems developed, each with its strengths and weaknesses, and used in different fields. The attendance system generally consists of steps such as image acquisition, database development, pre-processing, and face detection [3]. The main goal of this paper is to present a proposal for the use of facial recognition technology in the system for entering the exam and verifying the identity of the student. Many examination centers carry out a package of traditional procedures to verify identity through identification documents such as the student card, which results in wasting time and the difficulty of detecting impersonate. In this paper, the problems of impersonation will be avoided when entering the exams by proposing to implement the identity verification application before entering the exam. This paper is organized as follows: Section 2 and 3 Related Work and Proposed Methodology; Section 4 Experiments and Result and finally the conclusion.

1.1 Face detect:

In 2001 Paul Viola and Michael Jones published "Rapid Object Detection using a Boosted Cascade of Simple Features" Presented an efficient technique for object identification using cascading classifiers by a feature called Haar cascade based on face detection. The sequencing function is developed using machine learning utilizing a huge number of both positive and negative images, as illustrated in Figure 1. Utilizing it to locate the components in the other images is the following step [4].

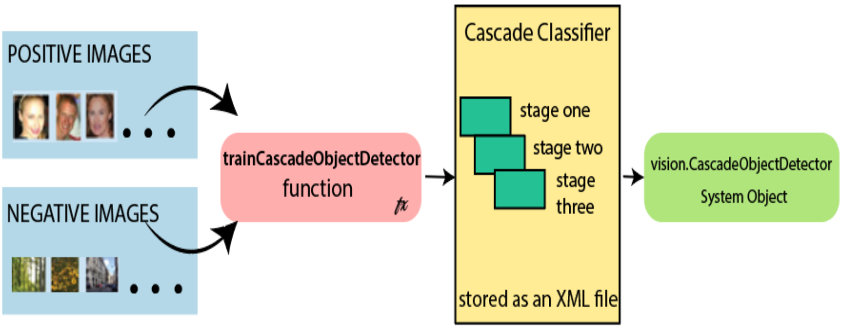


Figure 1 Cascade Classifier

The method needs a huge number of both images with faces in them and images without them in order to train the classifier. Then we need to extract features from it.

For instance, as seen in figure 2, the top row contains two exceptional attributes. The first attribute picked seems to place the most focus on the fact that the area around the eyes is typically darker than the area surrounding the nose and cheekbones. The second characteristic was picked because the eyes are darker than the bridge of the nose [5].

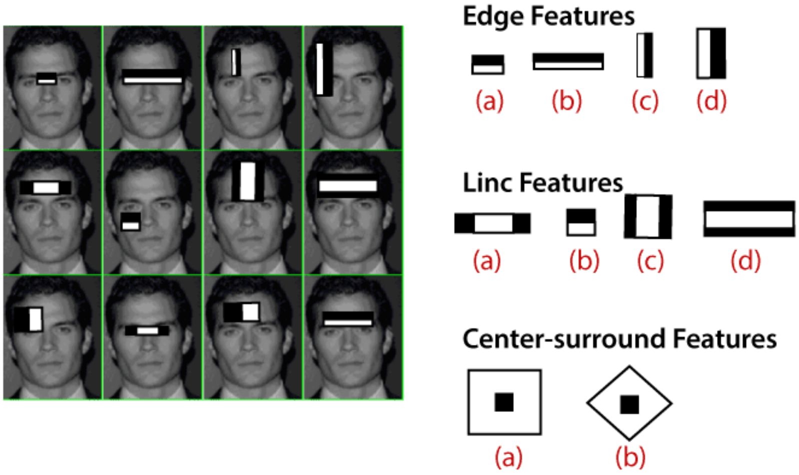


Figure 2 explain face detect using Haar cascade technique

Haar-like properties serve as the building block for Haar-classified object detection. These properties use the difference in contrast values between successive rectangular sets of pixels rather than pixel density values [6].

- The algorithm works in four stages to detect an object as follows:
- Haar Feature Selection
  - Create integral image
  - Adaboost Training
  - Cascading Classifiers

Another way to detect faces is a multi-task cascade convolutional neural network (MTCNN) used to obtain precise face coordinates as an additional method of face detection. The left and right corners of the mouth, the center of the nose, and the centers of the left and right eyes are five critical locations that may be detected using the MTCNN network [7]. The eye and mouth are mostly located by critical places in the follow-up. Three sub-networks make up the core of the MTCNN network structure: the proposal network (P-Net), refine network (R-Net), and output network (O-Net), which receives input from the images. Face detection is achieved using a coarse-to-fine procedure using these three cascaded sub-networks. The precise bbox coordinates are then output [8]. In figure 3 depicts the architecture of the MTCNN.

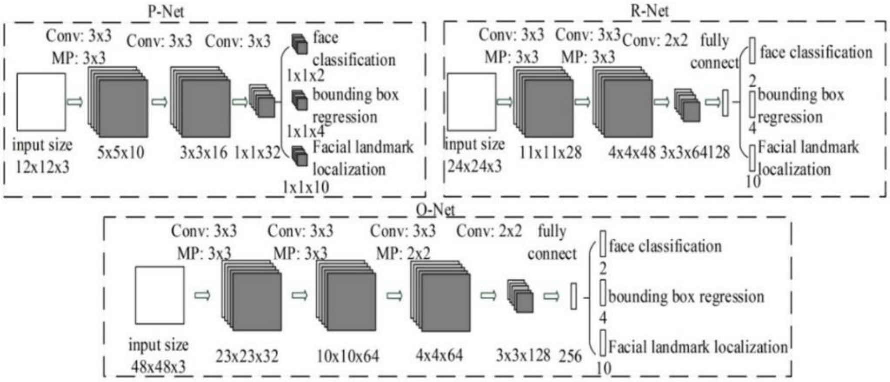


Figure 3 MTCNN architecture

1.2 Face recognition:

Face recognition is a significant research issue that crosses many areas and fields. A computer program that can recognize, track, identify, or authenticate human faces from an image or video taken with a digital camera is also available. Issues that prevent it from approaching or exceeding human-level accuracy are still limiting progress. These difficulties include variations in lighting, noise, scale, standing, etc. that affect how people's faces seem. Francis Galton suggested a formal system for classifying faces and assembling face profiles as curves, determining their average, and then classifying additional profiles according to how far they deviated from the average. This classification produces a vector of independent measures that can be compared to other vectors in a database since it is multi-modal [9]. Serign Modou Bah and Fang Ming presented a new method using Local Binary Pattern (LBP) algorithm combined with advanced image processing techniques like Contrast Adjustment, Bilateral Filter, Histogram Equalization, and Image Blending. The results of our tests demonstrate that our approach is very accurate, dependable, and robust for facial recognition systems that can really be used in a real-life setting as an automatic attendance system [10].

1.3 Attendance management problem

Most educational institutions and international test centers, they use traditional methods in the entrance examination system, by presenting an identity document or a passport. The entry procedures take a long time and constitute an obstacle in identifying the identity by increasing the number of students. Moreover, the difficulties you face in identifying impersonators. Also, most companies take a long time to record the attendance of employees in traditional ways using the most common fingerprint. Several institutions use different systems to manage attendance, such as fingerprint and magnetic cards, which depend on the method of use, cost, reliability and security. The traditional methods have become useless with the increase in the number of students and the increase in cases of identity theft [11].

Through research on many scientific papers on the subject of attendance management, made several proposals about automating student attendance verification systems using facial recognition techniques to avoid fake attendance and improve the issue of time consumption in entry, which is a concern for institutions. The face recognition system is better than the Traditional methods, including aspects of processing time, providing access to student information and recording the time of entry and exit. Through all the problems we decided to present a proposal to develop solutions to the problem of attendance management by recognizing faces in examination centers.

2. Related Work:

An attendance system was proposed and provided an accurate and high result in face detection and counting in [12]. The system is designed efficiently using graphical interfaces. The proposal relied on the YOLO V3 algorithm to detect the face and identify it through a camera installed in the classroom, and a photo is taken at the beginning and the other at the end to ensure the presence of the attendees. The algorithm counts the attendance followed by recognizing the faces after going through the database of faces, then generates spreadsheets and sends them via email to students and parents at the end of the month.

Another study developed an automated attendance system using the Python language, by representing two different algorithms for detecting faces, the first is Haar-Cascade and the second is Histogram of Oriented Gradients (HOG) in [13]. Students were identified in the classroom using convolutional neural networks (CNNs), also with mask-wear in mind for students as a precautionary measure for COVID-19 and facial recognition in the presence of a mask. A graphical user interface (GUI) was used in the proposal to ensure use ability. The study proved that the algorithm used (HOG) provided high accuracy and efficiency in performance in several conditions of changing lighting.

In [14] the author proposes a hospital-based system that can transcribe patient forms with the help of facial recognition. The study suggested an algorithm based on the face vector to avoid injuries that occur in the faces of patients coming to a hospital to avoid delays in starting treatment. The system also gives short text messages that are sent to the police and the patient's relatives. The study relied on patient information from the central databases of different hospitals over the Internet, where the algorithm tested a large number of patient images from different databases, and the accuracy of the results reached more than 98%.

In paper [15], the author suggested an automated system for the students to attend the exam using the eigenface and Haar cascade algorithms that were programmed in Python and the OpenCV library to recognize faces. In order to avoid the noise that may occur to the students' image, and to improve the brightness of the facial images and the orientation of the eyes and nose, the Principal Component Analysis algorithm (PCA) was used. In addition, the system works to support the video streaming service to give a great opportunity for continuous recognition of the students.

3. Proposed Methodology

In this study of the attendance registration system for students applying to take the exam, the system was divided into two stages. In the first stage, a database of students' faces will be prepared. The second stage is face recognition and attendance registration. In figure (4) the first stage is capturing an image of the faces of students using a camera, and then identifying the face after comparing the images obtained by using the Cascade Classifier algorithm and the Multi-task Cascaded Convolutional Networks algorithm (MTCCN), then extracting the facial features. Then trained by using the following Eigenfaces, Local Binary Patterns Histograms (LBPH) and Fisherfaces algorithms, and finally stored in the database.

First step in the second stage is capturing an image of each student when entering the exam, Secondly the step of face detection, and then the face recognition process using the training algorithms, which is above-mentioned. And matching them with the database, finally recording attendance and sending a notification to the student and the exam center.

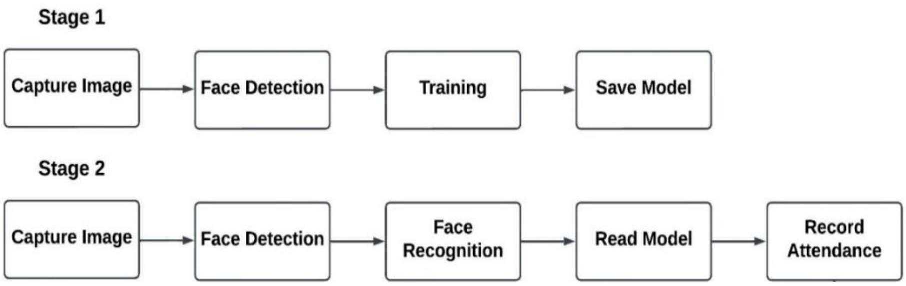


Figure 4: Stages of proposed methodology

3.1 Haar cascade vs MTCNN for Face Detection

According to our study to determine the face zone, which is required when the student enters the exam to stand in front of the camera to identify the face, we applied both Haar cascade and MTCNN. We have applied both algorithms to get the best results and to choose the right one for the proposal in terms of accuracy and efficiency. In this study, Haar-cascade method is implemented using OpenCV version 4.5.5 library which worked with python and their performances in terms of accuracy and required time are compared. Implement the proposal involves by calling the Extract Face() as following steps as shown in figure 5 :

1- Opening the camera by executing *VideoCapture( )*

2- The faces will be extracted from the video frame according to the Har-Cascade method. The classifier type must be selected, which is defined front face using the *OpenCV* library by loading *haarcascade_ frontalface_alt_tree.xml*, the faces in the image will be detected and then cropped into grayscale images.

```
def ExtractFace ():
face_cascade = cv2.CascadeClassifier_

('./haarcascade/haarcascade_frontalface_default.xml')
cap = cv2.VideoCapture(0)
if not cap.isOpened():
raise IOError("Cannot open webcam")
no_img=1
while(True):
ret, img = cap.read()
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
faces = face_cascade.detectMultiScale(gray, 1.3, 5)
for (x,y,w,h) in faces:
img = cv2.rectangle(img,(x,y),(x+w,y+h),(0,255,0),2)
cv2.imshow('Face Detect',img)
Return img[y:y+h, x:x+w]
```

Figure 5: Source code of Face detection using Haar cascade in python

In figure 6 there is a difference in results in A and B. it can be observed that the results of the Haar Cascade algorithm determined the face in (B), but in (A) some faces are not detected.

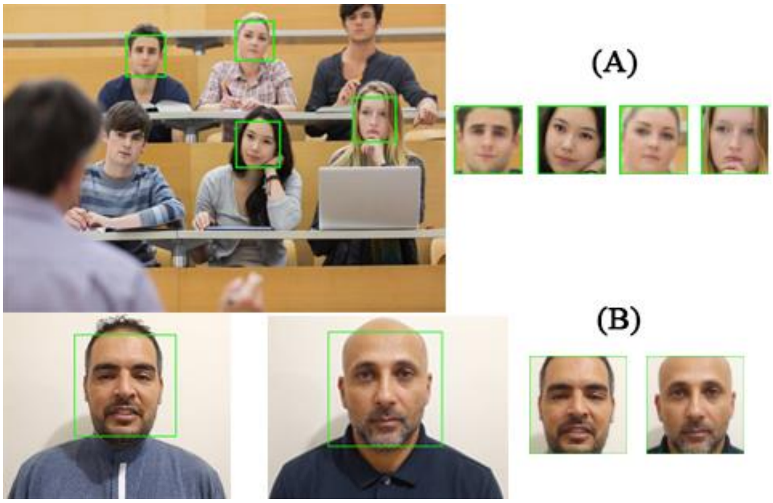


Figure 6: Extract face using Haar cascade

After implementing Haar Cascade for face detection, we followed another method in this study: to identify the face using the MTCNN algorithm, which includes its three-stage cascaded architecture (P-Net, R-Net, and O-Net) of the designed deep convolutional networks carefully predict the face and characteristics. In the same way, following the previous steps in opening the camera and obtaining the image, to start an algorithm MTCNN in its first phase, which produces facial areas from the image, then it goes through the second phase to filter the squares surrounding the face, and finally in the third phase, obtaining a proposal for the landmarks of the face [16]. In figure 7 presented the source code of face detection using python based on MTCNN.

```
:def detect_face(image)
()detector = MTCNN
bounding_boxes = detector.detect_faces(image)
return bounding_boxes
:def draw_bounding_boxes(image, bboxes)
:for box in bboxes
x1, y1, w, h = box['box']
cv2.rectangle(image, (x1,y1),(x1+w,y1+h),(0,255,0),2)
return image, image[y1:y1+h, x1:x1+w]
:def mark_key_point(image, keypoint)
cv2.circle(image, (keypoint), 1, (0,255,0), 2)
image = cv2.imread(image_path)
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
bboxes = detect_face(image)
: for i  in  bboxes
mark_key_point(image, i['keypoints']['left_eye'])
mark_key_point(image, i['keypoints']['right_eye'])
mark_key_point(image, i['keypoints']['nose'])
mark_key_point(image, i['keypoints']['mouth_left'])
mark_key_point(image, i['keypoints']['mouth_right'])
img1,img2= draw_bounding_boxes(image, bboxes)
cv2.imshow('Face Detect',img1)
```

Figure 7: Source code of Face detection using MTCNN in python

In Figure 8, we observed that the results were more accurate in the performance of the MTCNN algorithm that determines the facial landmarks.

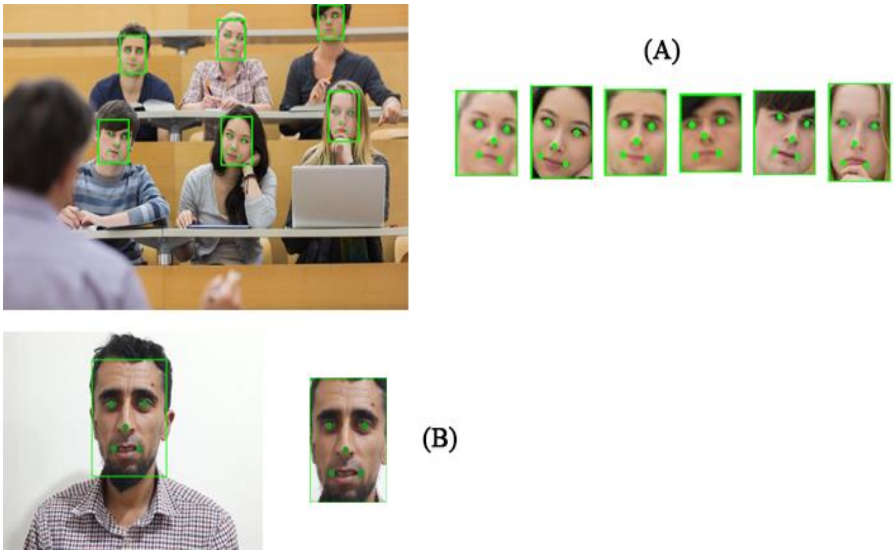


Figure 8: Source code of Face detection using MTCNN in python

Summarized up the above, regarding the results, we have decided to use the MTCNN algorithm for face detection which provides higher accuracy.

3.2 LBP for Face Recognition

In this study, we use Local Binary Pattern (LBP) for face recognition stage in the exam attendance system. LBP addresses the pixels in an image by specifying the surroundings of each pixel and displaying the result as binary [17][18]. The algorithm steps are as following:

- Parameters: LBPH utilizes parameters: Radius, Neighbors, Grid x and Grid y.
- Training: A dataset containing the faces of the persons we wish to recognize is required. For the algorithm to identify an input image and provide you with an output, we also need to set an ID for each image. The same ID must appear on all images of the same person.

- Using the LBP process: The application procedure begins with the computational phase of LBPH to produce an intermediate image that, by emphasizing facial traits, accurately represents the original image. The approach, which is based on the parameters radius and neighbors, leverages the idea of a sliding window. This procedure is presented below in figure 9.

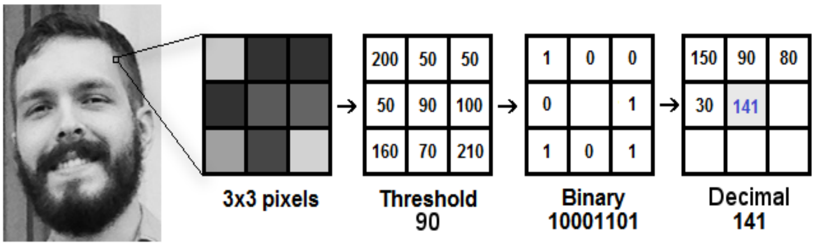


Figure 9: LBP process

- Extracting the Histograms: At this phase, the image is divided into numerous grids using the Grid X and Grid Y parameters.
- Face recognition performing: Each histogram that is produced in this phase is then utilized to represent each image from the training dataset. With an input image, it repeats the processes for the new image and generates a histogram that serves as an image representation.

4. Experiments and Result

This section discusses and shows the mechanism of the proposed system to confirm the identity of the examinee students, which was previously divided into several stages as already mentioned. The application was implemented on the Python3 environment with the use of the OpenCV4.5 Library to implement face detection and face recognition algorithms. The implementation of the proposed steps is as follows:

We have already mentioned in this paper, the MTCNN algorithm will be used to detect the face, and the LBP algorithm will be used to recognize faces through the technique of training and recognition. Firstly, students are registered by preparing 20 images for each student through the camera at different angles and lighting for the dataset to ensure accurate results. After that, in the testing and implementation phase, the face detection algorithm is applied to the dataset to crop the faces after detection, and then the faces are trained and stored in a database. Secondly, a captured image of the student in front of the camera before entering the exam, and also captured another image while exiting the exam to ensure the confirmation of the student's presence during the exam. As shown in Figure 10, there is a restriction on the system that rejects "NOT Allowed" the image when more than one student is in front of the camera.

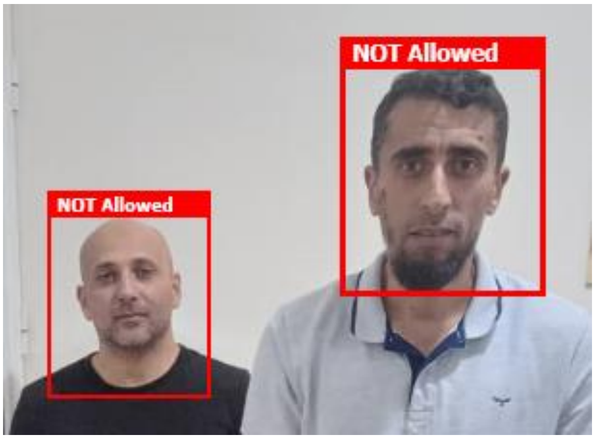


Figure 10: The system displays a "Not Allowed" message when more than one student is in front camera

Then the face recognition process by the LBP algorithm will predict the comparison of the captured image with the student's face database to ensure that the student's identity is confirmed to avoid identity theft as shown in Figure 11.

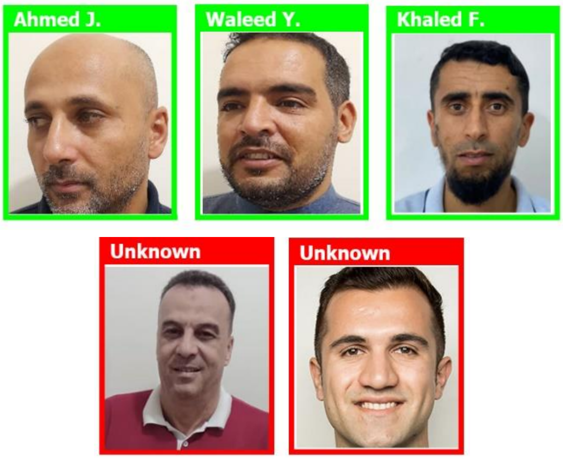
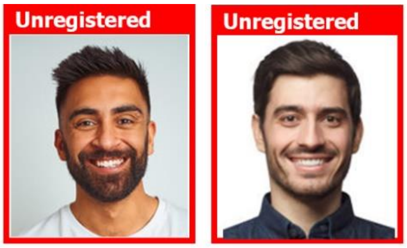
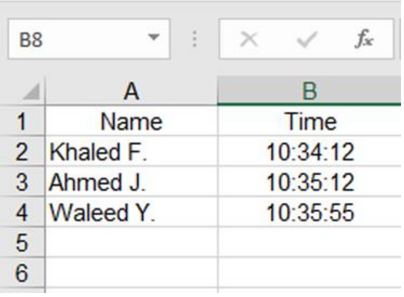


Figure 11: Face recognition process by the LBP algorithm

After face recognition of the student, the student’s registration in the course is verified and matched by searching in database courses. The system refuses entry to students who are not registered in the course as shown in figure 12. Finally, data is generated in a spreadsheet with a notification sent by Email or SMS to the student to confirm entry.





	A	B
1	Name	Time
2	Khaled F.	10:34:12
3	Ahmed J.	10:35:12
4	Waleed Y.	10:35:55
5		
6		

Figure 12: System detects unregistered students and generate spreadsheet

Conclusion

There are several methods to record the attendance of students and prevent impersonation in exams, which may take a long time in the case of a large number of students. In this paper, we proposed an attendance management system using face recognition techniques by taking pictures from cameras and comparing them to the student's photo database after training the system on students' faces in advance. The system has shown an accuracy in identifying students in most cases with different lighting and changes in the face such as hairstyle, beard or glasses. The greater the number of images the student has to train the system, the higher the accuracy of the results. In the implementation phase, we used the Python environment, which showed effectiveness in the use of artificial intelligence algorithms, with easy access to many open-source libraries, also in this paper, we concluded that the proposed system is effective in terms of reliability, material cost and reliability to replace the traditional methods of attendance management.

Reference

[1] Teoh, K.H., Ismail, R.C., Naziri, S.Z.M., Hussin, R., Isa, M.N.M. and Basir, M.S.S.M., 2021, February. Face recognition and identification using deep learning approach. In *Journal of Physics: Conference Series* (Vol. 1755, No. 1, p. 012006). IOP Publishing.

[2] Salim, O.A.R., Olanrewaju, R.F. and Balogun, W.A., 2018, September. Class attendance management system using face recognition. In *2018 7th International conference on computer and communication engineering (ICCCE)* (pp. 93-98). IEEE.

[3] Kumar, N.A., Swarnalatha, P., Chowdary, P., Naidu, J.K. and Kumar, K.S., 2019. Smart attendance marking system using facial recognition. *Research Journal of Science and Technology*, *11*(2), pp.101-108.

[4] Gupta, I., Patil, V., Kadam, C. and Dumbre, S., 2016, December. Face detection and recognition using Raspberry Pi. In *2016 IEEE international WIE conference on electrical and computer engineering (WIECON-ECE)* (pp. 83-86). IEEE.

[5] Khan, M., Chakraborty, S., Astya, R. and Khepra, S., 2019, October. Face detection and recognition using OpenCV. In *2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)* (pp. 116-119). IEEE.

[6] Gangopadhyay, I., Chatterjee, A. and Das, I., 2019. Face detection and expression recognition using Haar cascade classifier and Fisherface algorithm. In *Recent Trends in Signal and Image Processing* (pp. 1-11). Springer, Singapore.

[7] Ji, Y., Wang, S., Zhao, Y., Wei, J. and Lu, Y., 2019. Fatigue state detection based on multi-index fusion and state recognition network. *IEEE Access*, *7*, pp.64136-64147.

[8] Jin, R., Li, H., Pan, J., Ma, W. and Lin, J., 2021. Face recognition based on MTCNN and Facenet.

[9] Tolba, A.S., El-Baz, A.H. and El-Harby, A.A., 2006. Face recognition: A literature review. *International Journal of Signal Processing*, *2*(2), pp.88-103.

[10] Bah, S.M. and Ming, F., 2020. An improved face recognition algorithm and its application in attendance management system. Array, 5, p.100014.

[11] Arsenovic, M., Sladojevic, S., Anderla, A. and Stefanovic, D., 2017, September. FaceTime—Deep learning based face recognition attendance system. In *2017 IEEE 15th International symposium on intelligent systems and informatics (SISY)* (pp. 000053-000058). IEEE.

[12] Khan, S., Akram, A. and Usman, N., 2020. Real time automatic attendance system for face recognition using face

API and OpenCV. *Wireless Personal Communications*, *113*(1), pp.469-480.

[13] Akay, E.O., Canbek, K.O. and Oniz, Y., 2020, October. Automated student attendance system using face recognition. In 2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT) (pp. 1-5). IEEE.

[14] Parab, M. and Mehendale, N., 2021. Face Recognition-Based Automatic Hospital Admission with SMS Alerts. *SN Computer Science*, *2*(2), pp.1-11.

[15] Puthea, K., Hartanto, R. and Hidayat, R., 2020, May. The Attendance Marking System based on Eigenface Recognition using OpenCV and Python. In Journal of Physics: Conference Series (Vol. 1551, No. 1, p. 012012). IOP Publishing.

[16] Karmakar, R., 2020. FACIAL ATTENDANCE SYSTEM USING MTCNN AND FEATURE MAPPING. *International Journal of Engineering Applied Sciences and Technology*, *5*, pp.546-550.

[17] Elias, S.J., Hatim, S.M., Hassan, N.A., Abd Latif, L.M., Ahmad, R.B., Darus, M.Y. and Shahuddin, A.Z., 2019. Face recognition attendance system using Local Binary Pattern (LBP). *Bulletin of Electrical Engineering and Informatics*, *8*(1), pp.239-245.

[18] Khan, M., Chakraborty, S., Astya, R. and Khepra, S., 2019, October. Face detection and recognition using OpenCV. In *2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)* (pp. 116-119). IEEE.