ÓPTICA PURA Y APLICADA
www.sedoptica.es

Type: RESEARCH PAPER
Section: IMAGING TECHNIQUES

Automatic Class Attendance System Using Biometric Facial Recognition Technique Based on Raspberry Pi

A. A. Shabaneh[1*], S. Qaddomi[1], M. Hamdan[1], A. Abu Sneineh[2], T. Punithavathi[3]

1. *Department of Telecommunication Technology Engineering, Faculty of Engineering and Technology, Palestine Technical University–Kadoorie (PTUK), P.O. Box: 7, Yafa Street, Tulkarm, Palestine*

2. *Department of Electrical Engineering, Faculty of Engineering and Technology, Palestine Technical University–Kadoorie (PTUK), P.O. Box: 7, Yafa Street, Tulkarm, Palestine*

3. *Communication Technology Section. British Malaysia Institute, Universiti Kuala Lumpur, Malaysia*

(*) E-mail: a.shabaneh@ptuk.edu.ps

Received: 20/05/2023 Accepted: 21/09/2023
DOI: 10.7149/OPA.56.3.51152

ABSTRACT:

This study presents a novel automatic class attendance system that uses a biometric facial recognition technique based on Raspberry Pi and a camera module. Furthermore, the operation of the detection algorithm is accomplished by calculating the distance between the eyes; distance from the forehead to the chin; distance between the nose and mouth; depth of the eye sockets; shape of the cheekbones; and contour of the lips, ears, and chin. The developed face recognition system works with four major steps: capturing and scanning, extracting, comparing, and matching the captured image. The developed system recognizes students' faces in real time to handle the daily task of tracking students' attendance instead of using a manual method. The Raspberry Pi 3 model B+ microprocessor combined with the open-source libraries of Python, such as OpenCV, and a camera module are implemented in the facial biometric identification and recognition algorithm for recording attendance. The performance of the developed recognition system exhibits a high accuracy of 100% with a fast detection time of 1 s. In addition, the developed system has the ability to capture a student's face at a distance of 1.25 cm from the camera and alongside student's face orientations of 0° and 45°. The developed system performs fast and authentically, and thus, it is worthy of integration into any application.

Keywords: Face Recognition, Raspberry Pi, Authentication, Automatic, Database, OpenCV.

REFERENCES AND LINKS

[1] B. M. H. Alhafidh, R. M. Hagem, and A. I. Daood, "Face Detection and Recognition Techniques Analysis," in *2022 International Conference on Computer Science and Software Engineering (CSASE)*, 2022, pp. 265-270.DOI:

[2] E .Al Hajri, F. Hafeez, and A. A. NV, "Fully Automated Classroom Attendance System," *Int. J. Interact. Mob. Technol.,* vol. 13, pp. 95-106, 2019.

[3] R. Jayaswal and M. Dixit, "Comparative analysis of human face recognition by traditional methods and deep learning in real-time environment," in *2020 IEEE 9th International Conference on Communication Systems and Network Technologies (CSNT)*, 2020, pp. 66-71.

[4] C. Gomes, S. Chanchal, T .Desai, and D. Jadhav, "Class attendance management system using facial recognition," in *ITM Web of Conferences*, 2020, p. 02001.

[5] H. TEMİZ, "Rapid Marking Attendance with Face Recognition," *Avrupa Bilim ve Teknoloji Dergisi,* pp. 78-86, 2022.

[6] V. Bhandiwad and B. Tekwani, "Face recognition and detection using neural networks," in *2017 International Conference on Trends in Electronics and Informatics (ICEI)*, 2017, pp. 879-882.
[7] Y. Kortli, M. Jridi, A. Al Falou, and M. Atri, "Face recognition systems: A survey," *Sensors,* vol. 20, p. 342, 2020.
[8] R. Jafri and H. R. Arabnia, "A survey of face recognition techniques," *journal of information processing systems,* vol. 5, pp. 41-68, 2009.
[9] M. Hassaballah and S. Aly, "Face recognition: challenges, achievements and future directions," *IET Computer Vision,* vol. 9 ,pp. 614-626, 2015.
[10] X. Tan, S. Chen, Z.-H. Zhou, and F. Zhang, "Face recognition from a single image per person: A survey," *Pattern recognition,* vol. 39, pp. 1725-1745, 2006.
[11] R. Ibrahim and Z. M. Zin, "Study of automated face recognition system for office door access control application," in *2011 IEEE 3rd International Conference on Communication Software and Networks*, 2011, pp. 132-136.
[12] N. A. Othman and I. Aydin, "A face recognition method in the Internet of Things for security applications in smart homes and cities," in *2018 6th International Istanbul Smart Grids and Cities Congress and Fair (ICSG)*, 2018, pp. 20-24.
[13] M. G. Krishnan and S. B. Balaji, "Implementation of automated attendance system using face recognition," *International Journal of Scientific & Engineering Research,* vol. 6, pp. 30-33.2015 ,
[14] A. Jadhav, A. Jadhav, T. Ladhe, and K. Yeolekar, "Automated attendance system using face recognition," *International Research Journal of Engineering and Technology (IRJET),* vol. 4, pp. 1467-1471, 2017.
[15] A. Muqeet, "M.: Face recognition based attendance management system using Raspberry Pi," *J. Inf. Comput. Sci,* vol. 9, pp. 86-90, 2019.
[16] M. Arsenovic, S. Sladojevic, A. Anderla, and D. Stefanovic, "FaceTime—Deep learning based face recognition attendance system," in *2017 IEEE 15th International symposium on intelligent systems and informatics (SISY)*, 2017, pp. 000053-000058.
[17] I. Gupta, V. Patil, C. Kadam, and S. Dumbre, "Face detection and recognition using Raspberry Pi," in *2016 IEEE international WIE conference on electrical and computer engineering (WIECON-ECE)*, 2016, pp. 83-86.
[18] H. Daryanavard and A. Harifi, "Implementing face detection system on uav using raspberry pi platform," in *Electrical Engineering (ICEE), Iranian Conference on*, 2018, pp. 1720-1723.
[19] A. A. Wazwaz, A. O. Herbawi, M. J. Teeti, and S. Y. Hmeed, "Raspberry Pi and computers-based face detection and recognition system," in *2018 4th International Conference on Computer and Technology Applications (ICCTA)*, 2018, pp. 171-174.
[20] N. R. Borkar and S. Kuwelkar, "Real-time implementation of face recognition system," in *2017 international conference on computing methodologies and communication (ICCMC)*, 2017, pp. 249-255.
[21] https://www.raspberrypi.org, "Access date 15-07-2023," *Access time 9:00 PM*.
[22] P. Pasumarti and P. P. Sekhar, "Classroom attendance using face detection and Raspberry-Pi," *International Research Journal of Engineering and Technology (IRJET),* vol. 5, pp. 167-171, 2018.
[23] N. Mustakim, N. Hossain, M. M. Rahman, N. Islam, Z. H. Sayem, and M. A. Z. Mamun, "Face recognition system based on raspberry Pi platform," in *2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)*, 2019, pp. 1-4.
[24] M. D. Iffandi, R. N. Adiningrat, J. R. Pandapota, J. F. Ramadhan, B. Kanigoro, and E. Irwansyah, "Attendance System with Face Recognition," in *Software Engineering Perspectives in Intelligent Systems: Proceedings of 4th Computational Methods in Systems and Software 2020, Vol. 2 4*, 2020, pp. 751-757.
[25] N. M. Ara, N. S. Simul, and M. S. Islam, "Convolutional neural network approach for vision based student recognition system," in *2017 20th International Conference of Computer and Information Technology (ICCIT)*, 2017, pp. 1-6.
[26] A. Hasban, N .Hasif, Z. Khan, M. Husin, N. Rashid, K. M. Sharif*, et al.*, "Face recognition for student attendance using raspberry pi," in *2019 IEEE Asia-Pacific Conference on Applied Electromagnetics (APACE)*, 2019, pp. 1-5.

1. Introduction

Verification is one of the important tasks in the information technology era. Face recognition technology is a popular technique used by operators in authentication. Biometric recognition systems are widely used, creating many improvements in the changing world of technology [1, 2]. Four decades ago, face recognition technology was considered one of the most difficult fields in artificial intelligence [3, 4]. Classical attendance verification systems take considerable time and paper attendance sheets can be easily manipulated [5, 6]. However, a smart attendance system that uses real-time face recognition is a realistic solution that provides a means to handle the daily task of checking students' attendance. The conventional sign-up recognition method is utilized to identify a student. However, this approach is slow, time-consuming, and inconsistent, because some students often sign for their absent classmates. A person's face provides his/her true identity [7, 8]. Hence, facial recognition is one of the most accurate identification systems that can be used on a person. This recognition system can be used in educational institutions, hospitals, and companies. Methods that utilize this physical feature are regarded as a huge breakthrough in image processing performance [9, 10].

Authentication is an issue in computer-based communication. Face recognition is widely used in many applications, such as system security and door control [11, 12]. G. Krishnan et al. [13] proposed taking students' attendance by using face recognition. They implemented a face recognition technique that used the principal component analysis (PCA) algorithm. Their system can recognize the faces of students and automatically save the response in a database. It can also retrieve the list of students who are absent on a particular day.

In another early study, A. Jha [14] designed and implemented an approach for student identification by applying a face recognition technology during a lecture. A teaching assistant recorded the students' attendance. They defined the operation of this face recognition system, which was deployed as an automated attendance method in a classroom environment. The whole scheme was implemented on MATLAB software.

A face recognition-based attendance management system that uses Raspberry Pi was proposed by M. Muqeet [15]. The major contribution of his work was the development of an attendance management system with Raspberry Pi and a webcam that used the face of students as input data. A webcam was connected to the Raspberry Pi module, while the images of students' faces were recorded with the webcam. Various face extraction stages were applied using the OpenCV platform through Qt Creator Integrated Development Environment (IDE). In addition, attendance was taken by recognizing the face of each student in front of it while entering the classroom or close to the webcam. This automatic attendance system achieved an accuracy of 92%.

Numerous ideas have been proposed by investigators for face recognition attendance systems. M. Arsenovic et al. [16] developed a new deep learning-based face recognition attendance system by using convolutional neural networks (CNNs). CNN cascade was applied to face detection and generated face embeddings, because CNNs accomplished the best results for larger datasets. In a real-time environment, their developed system achieved a total accuracy of 95.02% on a few datasets of the original face images of workers.

Recording attendance imposes a huge burden on lecturers if it is accomplished manually [17, 18]. To solve this problem, a smart attendance management system should be implemented in classrooms. The major contribution of the current study is to design and fabricate a new approach for an automatic class attendance system by using a biometric facial recognition technique based on Raspberry Pi 3 Model B+ and Python programming language. In addition, the Raspberry Pi 3 Model B+, as the microprocessor, is integrated with the OpenCV library and a camera module, which are used for facial detection and recognition. The developed system is unaffected by surrounding factors, such as overlighting, the expression of faces, and image background. The developed system focuses on precise and different features, such as a person's face, eyes, ears, and nose. Furthermore, the applied algorithm for face landmark detection operation is based on calculating the distance between the eyes; distance from the forehead to the chin; distance between the nose and mouth; depth of the eye sockets; shape of the cheekbones; and contours of the lips, ears, and chin. Lastly, this study suggests a novel technique for dealing with students at universities to avoid plagiarism and student agents.

2. Proposed System Design

2.1 Architecture

Figure 1 shows the proposed classroom setup provided with the developed attendance system. The developed system comprises two databases: a student database and an attendance database. Figure 2 illustrates the architecture of the automatic class attendance system based on a biometric facial recognition technique that uses the Raspberry Pi 3 model B+. During the initial stage, the camera module is turned on to capture an image of a student's face and the captured image is saved for further processing. Then, the face recognition process begins with its fastest and most authentic operation. Subsequently, the developed system compares the captured image of the student with the pictures stored in the database for matching. Lastly, the file with the names of the attending students is saved and generated.

Fig. 1. Classroom arrangement.

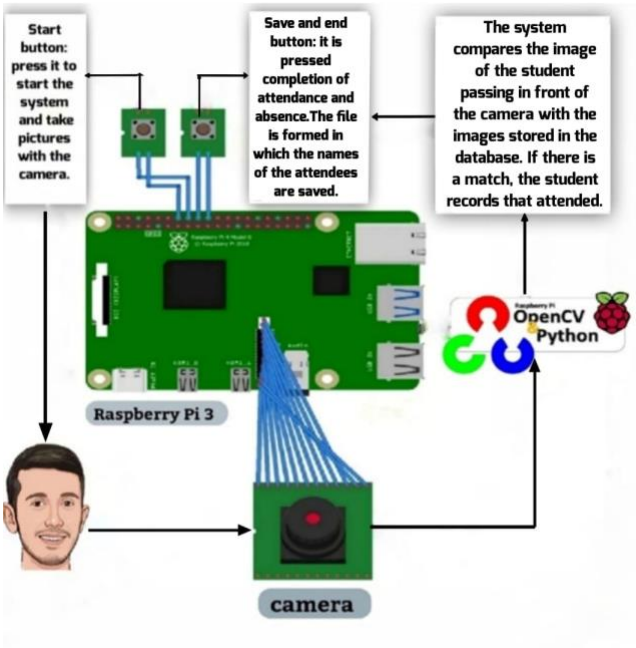

Fig. 2. System architecture.

2.2 Methodology and Algorithm

The developed system adopted a face detection and recognition technology to record students' attendance in classroom or hall meetings. This process is accomplished without any human intervention. The face detection and recognition scheme is based on the calculations of facial features. Furthermore, facial features, such as eyes, are extracted from the student areas by using the geometric relations between the eyes. The positions and angles between the eyes can determine the exact location of the face. In addition, nostril position and pattern can determine the exact location of the face. The proposed major steps of the attendance system are presented in a block diagram (Figure 3). Meanwhile, Figure 4 displays the detailed flowchart of the developed automatic class attendance system that uses a biometric facial recognition

system. The initial segment involves switching on the Python file to run the code pending the appearance of the message on the LCD screen as “save the attendance in a file.”

The Raspberry Pi, as a microprocessor with the open-source libraries of Python, such as OpenCV, and a Raspberry Pi camera module, is connected with the facial biometric identification and recognition algorithm [19, 20]. Data are stored on a 32 GB memory card connected to the Raspberry Pi. Furthermore, the Python programming language is used for coding the developed system. The principles of the developed design for the facial biometric recognition system are described as follows.

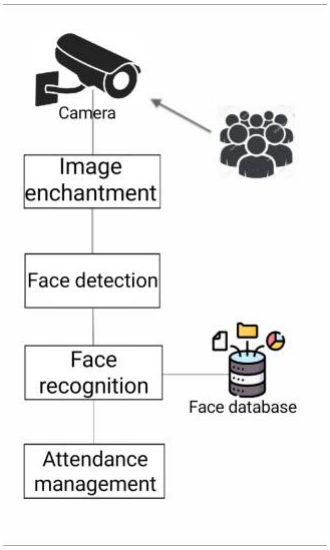

Fig. 3. Block diagram of the proposed attendance system.

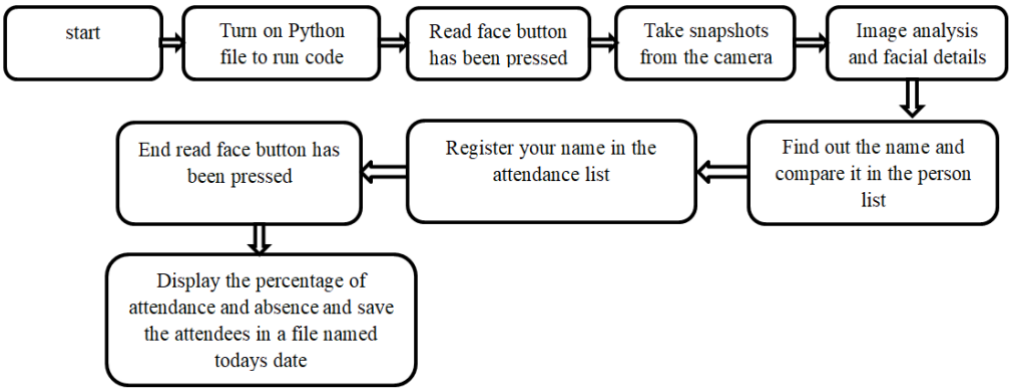

Fig. 4. Flowchart of the face recognition system.

First, the camera module captures the picture of all registered students and stores the information in the database. Subsequently, the developed system stores an image by mapping it into a face coordinate structure. Thereafter, when a registered student enters the building or the classroom, the developed system recognizes the student and records his/her attendance. In addition, the student’s information appears on the LCD screen, which is connected to the Raspberry Pi. Figure 5 shows the flow diagram of the data segment’s working process, including the camera module, database, and LCD screen.

Second, one of the most accessible features in facial recognition is the human eyes. Once the eyes are detected, detection of the eyebrows, nose, mouth, and ears follows. The developed system maps and reads face geometry and facial expressions. It recognizes facial landmarks, which are the key to distinguishing a face from other objects. Furthermore, the detection’s algorithm operation is accomplished by calculating the distance between the eyes; distance from the forehead to the chin; distance between the nose and mouth; depth of the eye sockets; shape of the cheekbones; and contour of the lips, ears, and chin (Figure 6). Once the face region is found, multiple algorithms are trained on a large dataset to detect faces. Then, the system converts the face recognition data into a string of numbers or points, called a faceprint. Each person has a unique faceprint, similar to a fingerprint. The information used by the facial recognition

system can also be used in reverse to reconstruct a person's face digitally. In addition, the background is not required to be static, and thus, the difference in background does not affect the operation of the developed system. However, lighting is an important aspect of facial recognition. Nevertheless, an environment with sufficient light provides more accurate results than a dark environment.

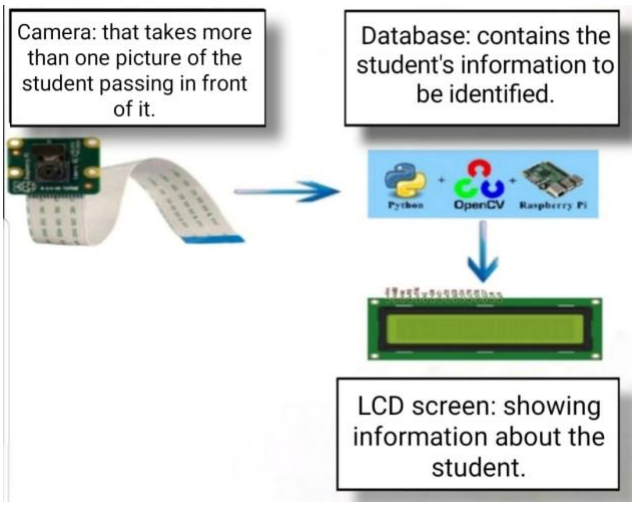

Fig. 5. Data segment.

Lastly, the developed system records the attendance in real time. In summary, the developed face recognition system works with four major steps, which are as follows:

1- Capturing and scanning: capture the picture of a student who attends the class by using the camera.

2- Extracting: unique biometric facial information is extracted from the sample.

3- Comparing: the collected information is compared with the stored database.

4- Matching: the system decides whether the collected picture is a match.

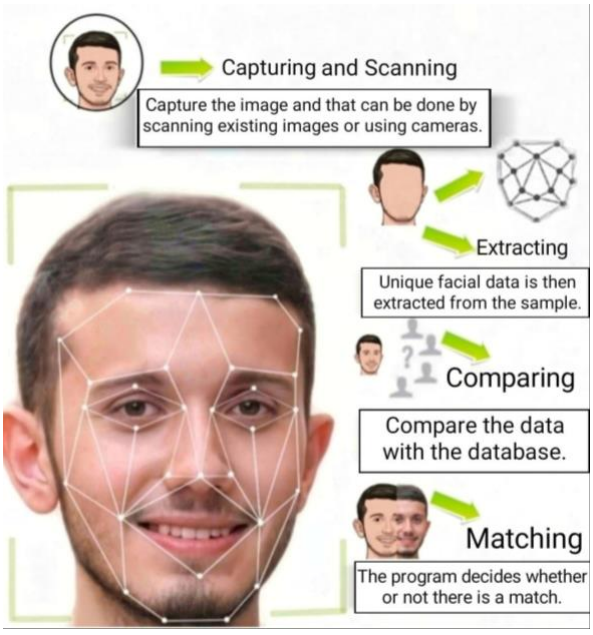

Fig. 6. Face recognition system.

3. Hardware Components

The primary objective of this project is to develop an automatic class attendance system that uses a biometric facial recognition technique based on Raspberry Pi with the help of a camera module. The developed system can be implemented in various applications where faces require authentication. The major hardware devices used in this project are the Raspberry Pi 3 Model B+ and a camera module.

3.1 Raspberry Pi 3 Model B+

Raspberry Pi is a low-priced, credit card-sized computer that can be plugged into a computer screen or television. It utilizes a standard keyboard and mouse. Raspberry Pi is used to design and improve applied Internet of things devices. The Raspberry Pi 3 Model B+ was launched with a faster 1.4 GHz microprocessor, 3× faster gigabit Ethernet, and 2.4/5 GHz dual-band 802.11ac Wi-Fi (100 Mbit/s). Other additional features are power over Ethernet, USB, and network boot [15, 21].

3.2 Camera Module

A camera is an optical tool for recording or capturing images. The Raspberry Pi camera module v1.3 is an authorized invention of the Raspberry Pi foundation. Thus, it has software and hardware support from the original designers of Raspberry Pi. It packs a 5 MP sensor that is capable of 2592 × 1944-pixel static images. The Raspberry Pi is connected using a 15 cm flex cable to the Raspberry Pi's CSI port. Moreover, the board of Raspberry Pi's camera is small, i.e., around 25 mm × 20 mm × 9 mm, and it weighs just over 3 g; thus, it is suitable for mobile or other applications where size and weight play significant roles [15, 21].

4. Software Components

4.1 Raspbian Operating System

The Raspbian operating system is an open-source operating technique based on Debian optimized for Raspberry Pi hardware. An employed scheme is the set of basic programs that run the Raspberry Pi component. Raspbian offers more than a regular operating system: it has over 35,000 packages, pre-compiled software bundled in a tidy and agreeable structure for easy set up on Raspberry Pi [15, 22, 23].

4.2 OpenCV

Open-source computer vision library (OpenCV) is an open-source image processing library generated through Intel and maintained by Willow Garage obtainable for C, C++, and Python. OpenCV requires a compiler, such as DevC++, code blocks, and visual C++ [15]. Moreover, the OpenCV library has more than 2500 optimized algorithms. These algorithms are appropriate to be employed for face detection and object identification [22]. In addition, OpenCV has four modules: main OpenCV functions, image processing algorithms, vision algorithms, and high GUI: GUI functions, Image, and Video I/O. In this project, the captured image by the camera module is loaded using OpenCV library. The captured images are in three formats: binary, gray-scale, and colored images [15].

4.3 Python

Python is a powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. In many fields and platforms, Python has a great sentence structure, dynamic typing, and interpreted nature, making it an ideal language for scripting and rapid application development. Furthermore, the Python interpreter is simply extended with new functions and data categories fulfilled in C or C++ [22]. Figure 7 shows the interface of the Python code of the developed system.

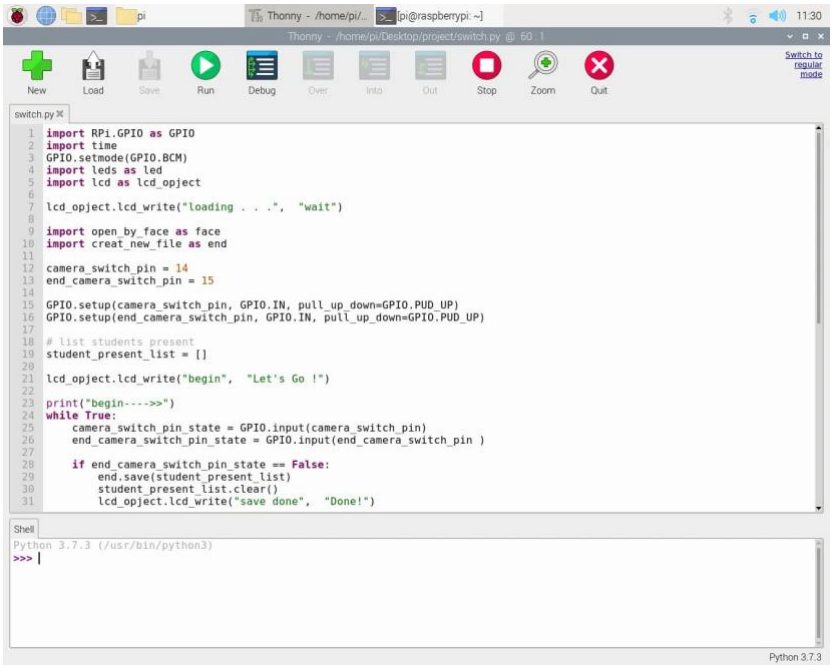

Fig. 7. Python code for interfacing Raspberry Pi with a camera module.

5. Experimental Performance and Results

The local database is built and loaded on the developed system. In this system, Raspberry Pi displays and recalls the OpenCV library, as shown in Figure 8, where a Raspberry Pi camera module is installed to capture students' faces. First, data were accessible and stored on a memory card connected to Raspberry Pi. Subsequently, a student's photo is compared with the photos stored on the device, as shown in Figure 9.

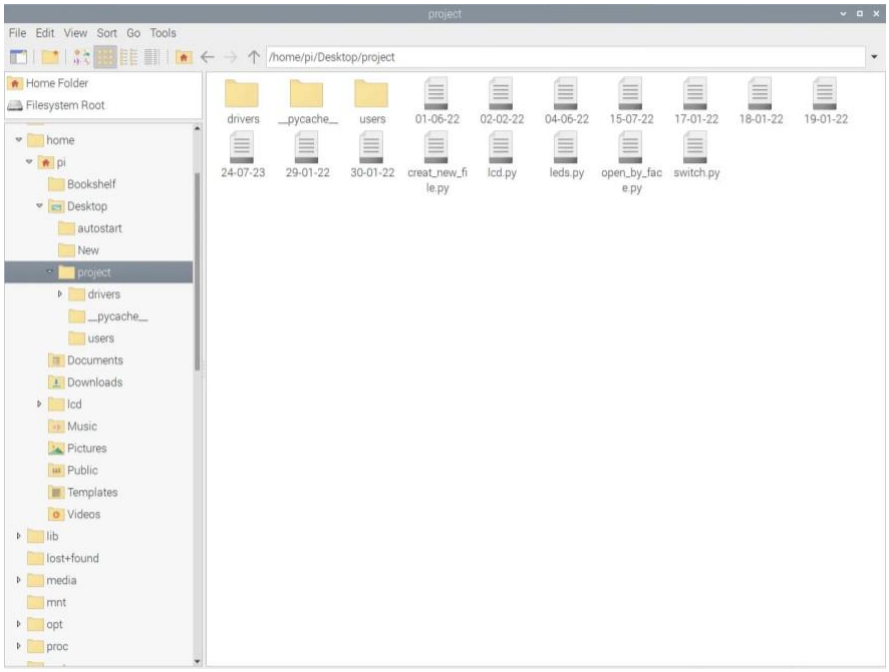

Fig. 8. Library of the developed system.

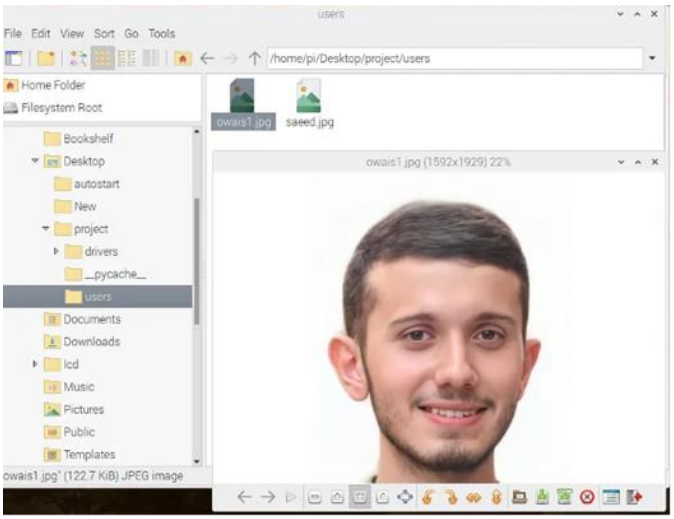

Fig. 9. Image stored inside the system.

In general, face recognizers normally take face images and find critical points, such as the corner of the mouth, an eyebrow, eyes, nose, and lips. Therefore, when the developed face recognition system identifies a student who attended the class and recorded him/her as attended, particular information about him/her will appear on the LCD. The experimental results of the proposed system indicate that students' faces are detected in color images irrespective of size, expression, lighting situations, and orientation. The advantages of the developed face recognition system are its functionality with diverse picture backgrounds. A difference in light reflection will not affect the captured picture of a person passing in front of the camera. Therefore, the developed system accepts different backgrounds of captured images. Figure 10 illustrates the altered backgrounds of the images for the same student to be recognized and identified. Nonetheless, a comparison is made of the detected faces by checking the database of students' faces.

The steps of the developed face recognition and identification system are shown in Figures 11 (a) to (F). Figure 11 (a) displays the start (begin "Let's Go"), which appears on the LCD screen when the code is run. Figure 11 (b) shows what appears on the screen when the system downloads the code and compares the image stored in the device with a person's face with a message of (loading…wait). Figure 11 (c) shows the identification message of the developed system (welcome Owais…add name done). Figure 11 (d) shows when the system does not recognize the person with a message of (unknown person…please try again). Figure 11 (e) shows what happens when the red button is pressed, and the recording of the attendance and absence is completed with a message of (save….done). The completed prototype of the developed system is shown in Figure 12.

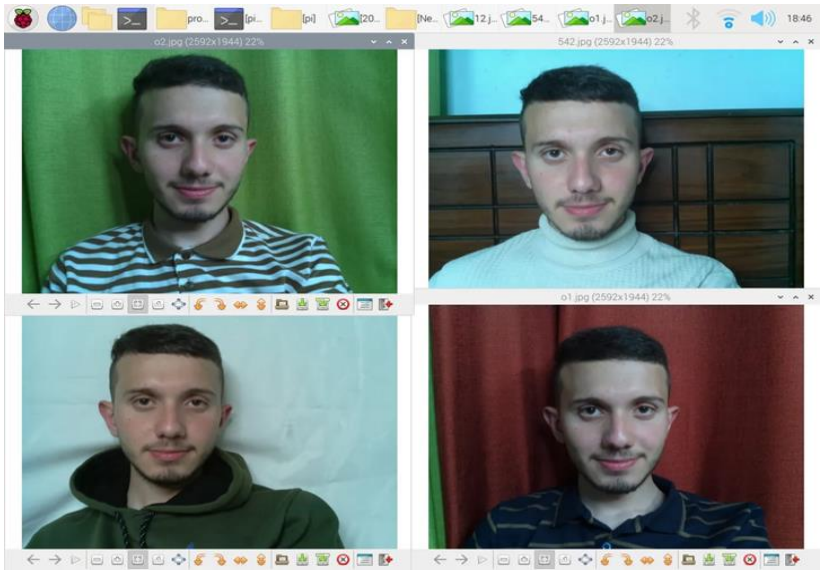

Fig. 10. Face recognized in different backgrounds.

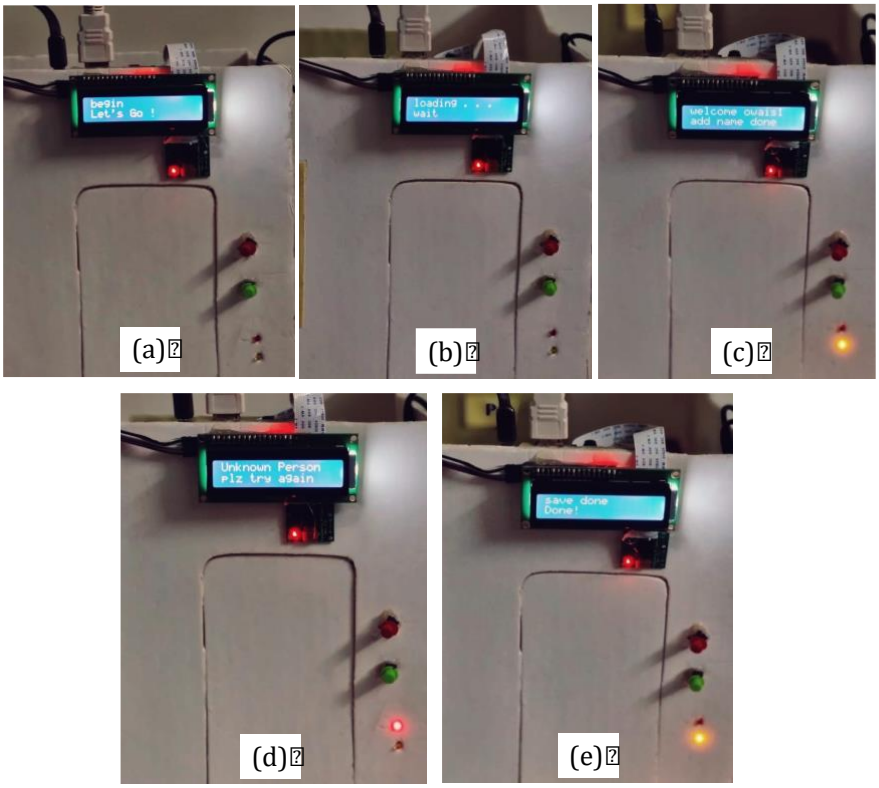

Fig. 11. Steps of the developed face recognition and identification system include the following: (a) start of the system, (b) loading the code, (c) the system identifies the student, (d) the system does not recognize the student, and (e) recording the attenda nce and absences.

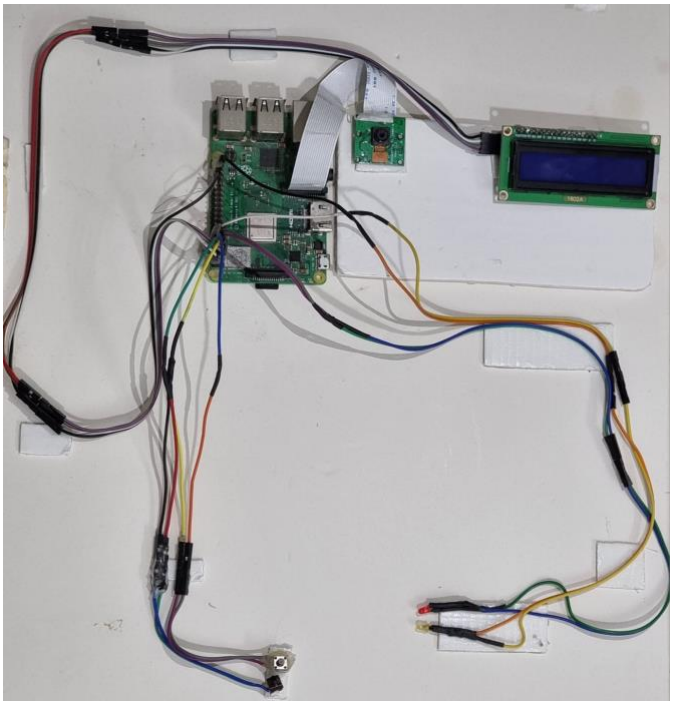

Fig. 12. Prototype of the project.

The developed system automatically detects a student when he or she enters the classroom while facing the camera module. This system helps track students compared with the traditional method, saving time. M. D. Iffandi et al. [24] designed an attendance system based on facial recognition. However, the background must be stable and sufficient lighting is required for more accurate results. By contrast, the

process of taking attendance and absence in this project is unaffected by background and illumination conditions.

Furthermore, in N. Mustakim et al., [23] gray scale was used to convert the image to the gray-scale format where the color is noise. In contrast, in this paper, face recognition does not require a change in the color of the image. Lastly, the developed system exhibits a high level of accuracy, and the whole process of recording attendance takes about 1 s.

For further investigation, different distances between student's face and the camera module of 45, 85, and 1.25 cm with 0°, 45°, and 90° angles are applied and tested for the developed automatic class attendance system. Similarly, Table 1 shows that the distance between a student's face and the Raspberry pi camera module with 0° and 45° angles does not affect the accuracy of recognition as distance increases. Nonetheless, if a student stands in front of the camera at an angle 90° with different distances, then the developed system will not recognize the student's face.

TABLE 1. Range distance and angle of face detection.

Distance between the person and the camera module (cm)	Face angle (°)		
	0	45	90
45	Recognize	Recognize	Does not recognize
85	Recognize	Recognize	Does not recognize
1.25	Recognize	Recognize	Does not recognize

Finally, the developed system will be an effective method for maintaining attendance and tracking students' records. The attendance system with face recognition performs the daily activities of attendance analysis, which is an important aspect of face recognition. In addition, the activity is done in an automated manner, saving time and effort in classrooms and hall meetings. Table 2 presents a comparison of state-of-the-art automatic class attendance systems using landmark facial recognition techniques. References [16] and [25] reported a novel method for face recognition attendance systems by using Python language-based API for face recognition and a CNN microprocessor. Although they obtained the same accuracy of 95%, they did not measure the detection time of the process. The references [26], [15], [22], and [23] reported innovating face recognition systems based on the Raspberry Pi platform. In addition, the accuracy of their systems is not high but still acceptable. However, they used different types of face recognition algorithms. Conversely, the current work achieves a high accuracy of 100% and fast detection time of 1 s compared with some previous studies.

TABLE 2. Comparison table among different face recognition techniques.

Reference	Microprocessor	Programming language and library	Accuracy	Detection time (s)	Algorithm
[16]	CNNs	Python/OpenCV	95.02%	-	Face recognition API
[25]	CNNs	Comma-separated values	95%	-	python based API for face recognition
[26]	Raspberry Pi	Python/OpenCV	Acceptable	-	Haar Cascade
[15]	Raspberry Pi	OpenCV platform using the Qt creator IDE	92%	-	Viola–Jones algorithm implemented in Python
[22]	Raspberry Pi	Python/OpenCV	Acceptable	-	LBPH
[23]	Raspberry Pi	Python/OpenCV and numPy	Acceptable	1	SQLite
Present work	Raspberry Pi 3 B+	Python/OpenCV	100%	1	Python-based API for face recognition

6. Conclusions

In this research, a new approach for an automatic class attendance system that uses a biometric facial recognition technique based on Raspberry Pi 3 Model B+ is successfully fabricated and tested. The developed system is created based on facial landmark detection that tracks the attendance status of students. In addition, the developed system uses a Raspberry Pi 3 Model B+ microprocessor integrated with OpenCV library and a camera module for facial detection and recognition. Data are stored in a memory card connected to the Raspberry Pi, where data can be accessed. The developed system's performance is unaffected by overlighting, expression of faces, and image background. The novel design of the face recognition system demonstrates fast detection time and highly accurate tracking of captured images. Consequently, the developed system can be implemented accurately in universities, schools, factories, companies, and hospitals. Subsequently, the system is user-friendly, easy to use, and consistently offers more security, privacy, and well-organized data onboard.

Acknowledgments

The authors thank Palestine Technical University–Kadoorie for supporting and funding this work.

