ISSN: 1934--9955 www.ijise.net
Vol-20 Issue-01 April 2025

# Smart Facial Attendance System with Raspberry pi using OpenCV and Python

**[1]Ch.Sasivardhan Rao, [2]B.Pavan Kalyan,[3]D.Sai Milind, [4]B.Sai Kiran,[5]B.Chandu**
**[*]Mr.P.Mahesh, [**]Dr.R.S.R.Krishnam Naidu**
[12345]UG student, Electrical and Electronics Engineering, Nadimpalli Satyanarayana Raju Institute of Technology (Autonomous), Sontyam, Visakhapatnam, Andhra Pradesh, India – 531173.
[*]Assistant Professor, Electrical and Electronics Engineering, Nadimpalli Satyanarayana Raju Institute of Technology (Autonomous), Sontyam, Visakhapatnam, Andhra Pradesh, India – 531173.
[**]Hod & Professor, Electrical and Electronics Engineering, Nadimpalli Satyanarayana Raju Institute of Technology (Autonomous), Sontyam, Visakhapatnam, Andhra Pradesh, India – 531173.

## Abstract

Facial recognition technology has become a powerful tool for uses, including managing attendance. Older methods, like paper forms or RFID systems, tend to be inaccurate and difficult to handle. To mitigate these drawbacks, this smart facial recognition attendance system (SFRAS) is aimed at offering an efficient and easy-to-use solution. The implementation of this system involves the integration of a Raspberry Pi single-board computer with a camera module that captures real-time photos of individuals as they arrive in a designated location. These images are processed with OpenCV's facial recognition algorithms to extract features. The extracted data is then matched against a database of pre-registered faces to verify individuals' identities. When identification is successful, the attendance record is automatically updated in a central database. Python scripts facilitate the communication among the Raspberry Pi, OpenCV, and the database. This SFRAS, utilizing Raspberry Pi as its hardware foundation, guarantees both cost-effectiveness and scalability, making it apt for automating attendance management in varied settings such as schools, offices, and organizations of any size. It also provides real-time monitoring and reporting functions, allowing administrators to effectively oversee attendance data. The proposed Facial Attendance System showcases the possibilities of merging Raspberry Pi, OpenCV, and Python to create an innovative and effective approach to attendance management. With additional improvements and optimizations, this system could transform the way organizations monitor and handle attendance in the digital era.

**Keywords:** Facial detection and recognition, database, Open CV, Raspberry Pi, camera Python Scripts.

### 1. Introduction

In today's educational environment, colleges constantly grapple with the challenge of monitoring student attendance efficiently while ensuring accuracy, security, and user convenience. Traditional methods like roll calls and sign-in sheets are still in use, but they are susceptible to mistakes and often lack real-time insights into attendance trends. Acknowledging the need for a more advanced solution, numerous institutions are adopting biometric technologies, particularly facial recognition systems, to transform the management of attendance on campus. This study aims to investigate the implementation of facial attendance systems in colleges, exploring the opportunities, challenges, and implications of this cutting-edge approach to tracking attendance. By examining the technical features, practical applications, and prospective advantages of facial recognition technology in a college setting, this report seeks to provide valuable insights for stakeholders to guidedecision-making and support successful integration. Facial attendance systems present various benefits over conventional methods, such as improved accuracy, automation, and real-time monitoring capabilities. By employing facial recognition algorithms to identify and verify students through their distinct facial features, these systems do away with labor-intensive data entry tasks and lessen the chances of fraudulent attendance logs. Furthermore, the smooth integration of facial attendance systems with existing campus technologies, including student databases and access control systems, helps streamline administrative functions and boost overall operational effectiveness. Nevertheless, introducing facial attendance systems in colleges comes with its own set of challenges and considerations. Ethical issues surrounding privacy, consent, and data security need careful attention to comply with legal guidelines and safeguard students' rights. Moreover, technical aspects such as system reliability, accuracy, and scalability require thorough assessment to ensure smooth operation and user acceptance. By analyzing case studies, best practices, and viewpoints from relevant stakeholders, this report aims to offer colleges actionable insights on the effective deployment of facial attendance systems. By emphasizing potential use cases, implementation strategies, and ways to navigate common obstacles, this report will equip colleges with essential knowledge and resources to utilize facial recognition technology effectively in their attendance management processes. As colleges seek to adopt innovative solutions and enhance efficiency, facial attendance systems stand out as a promising answer to the complex issues surrounding attendance tracking. By tapping into biometric technology, colleges can improve accountability, nurture a culture of punctuality, and furnish faculty and administrators with valuable insights into student engagement and attendance behaviour. This report will help colleges gain a clearer understanding of the transformative potential of facial attendance systems and the critical factors needed for successful implementation and integration into campus life.

### 1.1 Literature Survey

An Automatic Attendance Management System Using Face Recognition employs Eigen Faces and an Eigen Weight system for face detection. In this setup, a camera captures images and crops the faces of students to link them with a pupil database (Vardharajan, E et al., 2016). Prof. Arun Katara and Sudesh introduced an Attendance System Using Face Recognition and Class Monitoring, utilizing Raspberry Pi and the OpenCV library. The USB camera connects to the Raspberry Pi, alongside a connected database (Arun Katara et al., 2017). A Face Recognition Attendance Systemfeaturing GSM announcements used the Viola-Jones algorithm for face detection. The Fisher Faces algorithm generated patterns from the captured faces, creating templates stored in a database, paired with OpenCV and Software Development Kit (SDK) functionalities for the graphical user interface (Kennedy Okokpujie et al., 2017). Jenif developed an automated attendance marking system via facial recognition, where a camera captures images of students to mark attendance automatically. This system leverages a histogram algorithm for face identification, converting face images into matrix forms and using histograms to recognize specific faces. It addresses time consumption issues (Jenif D Souza, et al., 2019). An automated attendance system based on face recognition and gender classification was proposed, utilizing Haar-Cascade, LBPH Algorithm, and the LDA Model (Kritika Shrivastava et al., 2018). Another attendance system recorded video of students, converting it into frames to store in a database, improving accuracy and speed through complex neural network (CNN) algorithms (Nandhini R., 2019). Shreyak Karan and Samyak Jain introduced aReal-Time Smart Attendance Management System Using Face Recognition Techniques, utilizing CNN for face detection and recognition at the classroom entrance, along with an additional camera inside the classroom to verify attendance (Shreyak Sawhney et al., 2019).

### 1.2 Method

A cost-effective and innovative approach to tracking attendance is by using Raspberry Pi combined with face detection technology. Figure 1 illustrates the block diagram of the proposed model, which incorporates a webcam connected to a Raspberry Pi. By using Python programming with the OpenCV (Open Source Computer Vision Library), face detection can be achieved. OpenCV is an advanced tool ideal for real-time image processing, making it suitable for face detection applications. The process begins with capturing images of individuals present in a specific area. These images are then processed through the OpenCV library to detect and recognize faces. Using Python, attendance can automatically be recorded based on recognized individuals. This approach not only simplifies the attendance tracking process but also eliminates manual entry, reducing errors. Additionally, it creates a more secure monitoring method linking individuals directly to their facial identities, thus ensuring accuracy. Testing is vital for the successful implementation of this methodology. Comprehensive testing is necessary to confirm the system functions correctly and captures attendance accurately. Through diligent testing, any potential issues can be identified and rectified before a full rollout. The flow model outlining the steps involved in executing the proposed work is depicted in Fig 2.

### 2.1 System Setup and Configuration

The initial process involved installing Raspberry Pi OS on an SD card and setting up the Raspberry Pi 3B+ with basic configurations, including network connectivity. This step set the stage for crafting and executing the facial recognition software. The USB webcam was connected to one of the USB ports on the Raspberry Pi, and the system was configured to ensure it could recognize the webcam for capturing images needed for facial recognition processing. An LCD 16x 2 displays was connected to the Raspberry Pi via GPIO pins for data transmission and power supply. An I2C interface facilitated easier connections for communication with the display. A buzzer was linked to a designated GPIO pin for output and another pin to ground, providing auditory feedback upon successful or unsuccessful recognition.

### 2.2 Software Development

The setup required the installation of Python 3, followed by the use of pip to install necessary libraries, including OpenCV for handling images, “Dlib” for facial detection, and the “face recognition” library to identify faces. For GPIO control and I2C communication, RPi.GPIO was installed for hardware interactions with the LCD display and the buzzer. Python scripts were then developed to capture images, process facial recognition, and manage hardware interactions, including sending messages to the LCD and controlling the buzzer.

### 2.3 System Integration and Testing

Fig 3 depicts the complete system integration, showcasing all hardware components connected and software scripts designed. Integration testing was performed to ensure synchronized system operation, which included verifying the camera's image capture capabilities, assessing the accuracy of the facial recognition algorithm, and confirming the functionality of the LCD display and buzzer for user feedback. Based on testing outcomes, enhancements to thefacial recognition algorithm were made, potentially adjusting parameters, improving image pre-processing, or refining the training dataset.

### 2.4 User Interface and Experience Testing

Testing was conducted on the system's user interface and feedback mechanisms to ensure the LCD effectively communicated clear messages and the buzzer delivered appropriate cues during the attendance recording process.

### 2.5 Deployment and Real-World Application

The system was installed in a real-world context, such as in a classroom or office entry point, to commence the attendance tracking using facial recognition.

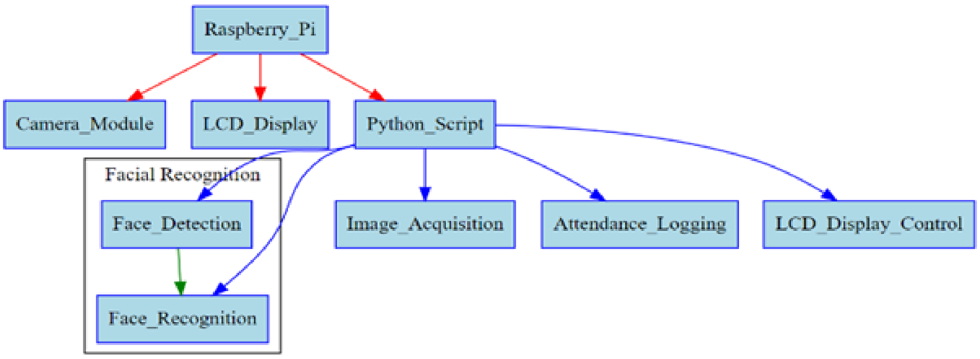


**Figure1:** BlockDiagram of the ModulesandTheirInterconnections

**Block diagram explanation:**

**Raspberry Pi:** The Heart of the System acts as the central processing unit. It's a small, low-cost computer that will manage all the components and execute the software.

**Input:** Camera Module is connected to the Raspberry Pi and captures images or video frames. This is the source of visual data for the facial recognition system.

**Facial Recognition Process (Within the Python Script)**

**Face Detection**: The captured image is first processed to detect the presence of faces. This involves identifying potential regions in the image that contain a face.

**Face Recognition:** Once a face is detected, the system analyzes it to identify who it is. This usually involves comparing facial features to a database of known faces.

**Attendance Logging:** If a recognized face matches an entry in the database (e.g., an employee), the Python script logs the attendance (time of recognition) potentially into a file or database.

**LCD Display Control:** The Python script also sends instructions to the LCD display to show relevant information, such as "Person X Present" or any other message related to attendance.

## 2.6 Monitoring and maintenance

Post-deployment, the system's performance was monitored, collecting user feedback to identify potential issues or areas for improvement. Regular updates to both software and hardware maintained optimal performance.

## 2.7 Data Management and Privacy Secure Storage

Implement robust storage solutions for safeguarding facial recognition data and attendance records, ensuring adherence to privacy regulations. Data Analysis: Utilize attendance data for further examination to identify trends or irregularities to boost organizational efficiency. Overall, leveraging Raspberry Pi, webcams, OpenCV, Python, and face detection technology presents a modern and effective method for tracking attendance. Integrating these tools allows organizations to enhance their attendance management systems and improve overall efficiency.

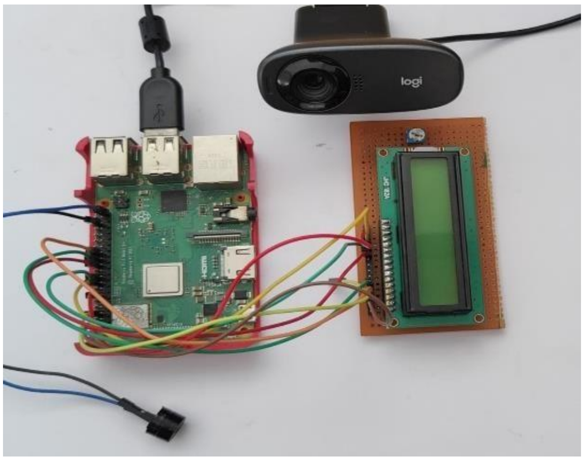

**Figure2:** Hardware Installation of the Model.

Fig 1 illustrates the block diagram of the proposed model, which integrates a camera module and an LCD display connected to a Raspberry Pi. In Fig 2, the flow diagram shows that a lightweight version of Linux, tailored for the Raspberry Pi, operates as the operating system, enabling the execution of Python scripts and interaction with hardware components. The flexibility of Python, coupled with its broad library support, makes it exceptionally suited for this project. The Open Source Computer Vision Library is employed for image processing and facial recognition, providing a variety of algorithms capable of detecting and recognizing faces in real-time. The hardware setup for this project is depicted in Fig 3, which identifies the trained face and displays a message on the LCD screen showing the person's name, while also recognizing untrained faces as unknown.

International journal of imaging science and engineering
ISSN: 1934--9955 www.ijise.net
Vol-20 Issue-01 April 2025

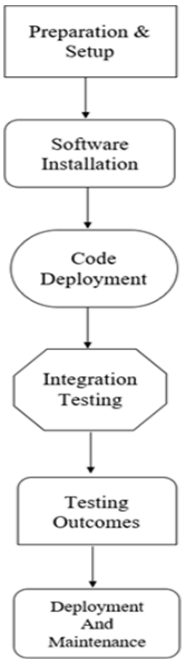


**Figure 3**: Flow Model Representing Various Steps Involved in Proposed Work

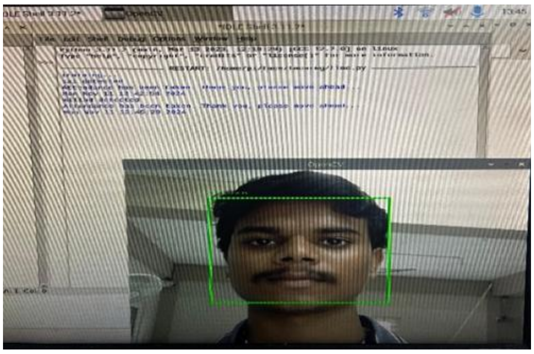

**Figure 4**: Screenshot of Various Database Images of Trained Face for Identification

## 3 Results

Upon system start-up, the Raspberry Pi activates the camera and the LCD display. The LCD instructs the user to stand in front of the camera to take images, as illustrated in Fig 4 and 5. The camera captures the user's images and transmits them to the Raspberry Pi for analysis.

**Figure 5**: Testing withFace2of TrainedFace

**Figure6:**Testing withFace1of TrainedSet

The collected images are processed with OpenCV and various face recognition libraries to detect and identify the user's face. The system matches the detected faces against a database containing 40 images of pre-registered individuals, as shown in Fig 6. When a match is identified, the system logs the user's attendance, as noted in the following Figure.

This is noted in database as shown in Figure7 along with a timestamp.

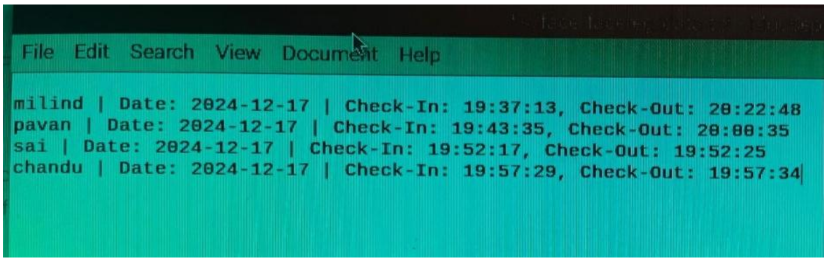


**Figure 7:** Screenshot of Data Base Displaying the Details of the Detected

ItindicatesthenameofthedetectedfaceontheLCDdisplay as"NameDetected"showninFigure8.Uponsuccessfulidentification,thesystemsignalsitsachievement by displaying "Person Detected" on theLCDscreen.


**Figure 8:** Display of Detected Message of theTrainedFace on theLCDScreen.

The identified trained faces data will be recorded in a register automatically with name, Present, with time and dateasshownin Fig 9.Incaseofanuntrained face, it checks for a match and if no match is found, the LCD display provides feedback to the user confirming an unsuccessful face as "Unknown"asshown in Fig 10.

International journal of imaging science and engineering

ISSN: 1934--9955 www.ijise.net
Vol-20 Issue-01 April 2025

**Discussion:**

This rapid communication indicates that the recognition process has completed successfully. It not only validates the system's ability to recognize persons but also enhances the engagement by offering clear ,instant feedback. This functionality is critical for applications that require speedy verification, increasing user trust and engagement with the technology.

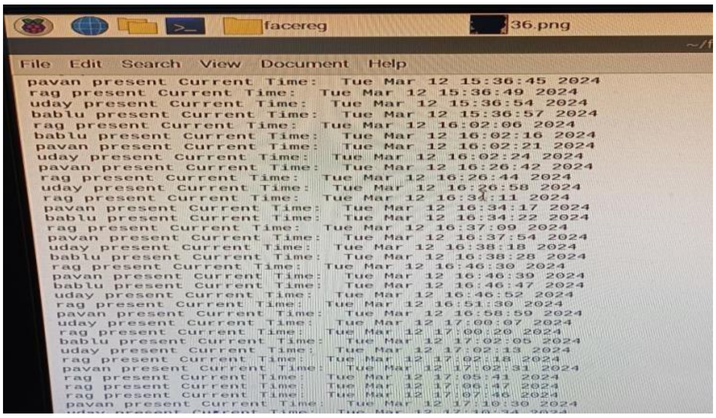


**Figure 9**: Screenshot of Identified Faces Data with Name, Time, Date

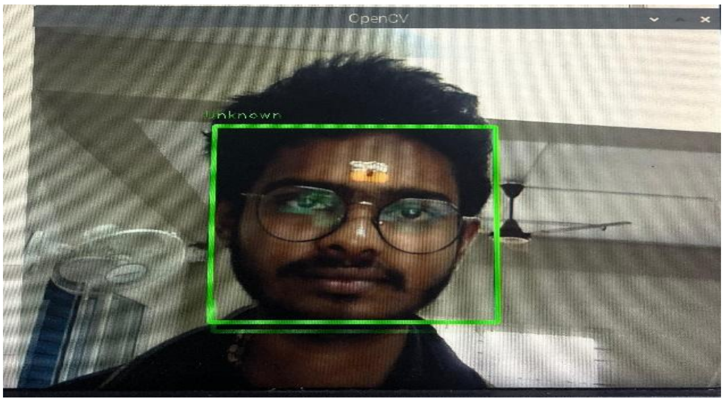


**Figure 10**: Screenshot of Unidentified FaceIndicatingas "Unknown"

**Conclusion**

This portable attendance system created with Python, OpenCV, and Raspberry Pi turned out to be an incredibly effective, inexpensive, and simple-to- install method of automating the attendance process. With the use of OpenCV's sophisticated image processing tools and Raspberry Pi's computing capability, this system offers a dependable, accurate, and user-friendly facial recognition-based attendance verification solution.

**References**

[1].E Varadharajan , R Dharani , S.Jeevitha, B. Kavinmathi, S. Hemalatha (2016). Automatic Attendance Management system using face detection at online International Conference on Green Engineering and Technologies (IC- GET), Coimbatore, India. pp. 1 doi:10.1109/GET.2016.7916753.

[2].Prof. Arun Katara, Mr. Sudesh V. Kolhe, Mr. Amar P. Zilpe, Mr. Nikhil D. Bhele, Mr. Chetan J. Bele (2017) Attendance System Using Face Recognition and Class Monitoring System. International Journal on Recent and Innovation Trends in Computing and Communication, 5(2), pp 273–276. https://doi.org/10.17762/ijritcc.v5i2.214

[3].K.Okokpujie, E. Noma-Osaghae, S. John, K.-A. Grace and I. Okokpujie. (2017) A face Recognition Attendance System with GSM Notification.IEEE 3rd International Conference on Electro-Technology for National Development (NIGERCON), Owerri, Nigeria, pp. 239-244, doi: 10.1109/NIGERCON.2017.8281895.

[4]S. Sawhney, K. Kacker, S. Jain, S. N. Singh and R. Garg, "Real-Time Smart Attendance System using Face Recognition Techniques," 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 2019, pp. 522-525, doi: 10.1109/CONFLUENCE.2019.8776934.

[5]A. Arjun Raj, M. Shoheb, K. Arvind and K. S. Chethan, "Face Recognition Based Smart Attendance System," 2020 International Conference on Intelligent Engineering and Management (ICIEM), London, UK, 2020, pp. 354-357, doi: 10.1109/ICIEM48762.2020.9160184.

[6]R. M. M and P. S. Hency Jose, "Raspberry Pi based Contactless Attendance Monitoring System for Hospitals," 2022 International Conference on Augmented Intelligence and Sustainable Systems (ICAISS), Trichy, India, 2022, pp. 715-720, doi: 10.1109/ICAISS55157.2022.10010964.

[7]M. Owais, A. A. Jalal, M. M. Hassan and A. Shaikh, "Facial Recognition based Attendance System Using CNN and Raspberry Pi," 2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), Istanbul, Turkey, 2020, pp. 1-5, doi: 10.1109/ISMSIT50672.2020.9254300.

[8]A. S. Hasban et al., "Face recognition for Student Attendance using Raspberry Pi," 2019 IEEE Asia-Pacific Conference on Applied Electromagnetics (APACE), Melacca, Malaysia, 2019, pp. 1-5, doi: 10.1109/APACE47377.2019.9020758.

[9]J. Harikrishnan, A. Sudarsan, A. Sadashiv and R. A. S. Ajai, "Vision-Face Recognition Attendance Monitoring System for Surveillance using Deep Learning Technology and Computer Vision," 2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN), Vellore, India, 2019, pp. 1-5, doi: 10.1109/ViTECoN.2019.8899418.

[10]I. ben Abdel Ouahab, L. Elaachak, M. Bouhorma and Y. A. Alluhaidan, "Real-time Facemask Detector using Deep Learning and Raspberry Pi," 2021 International Conference on Digital Age & Technological Advances for Sustainable Development (ICDATA), Marrakech, Morocco, 2021, pp. 23-30, doi: 10.1109/ICDATA52997.2021.00014.

[11]Mohammad Salah Uddin, S. M. Allayear, N. C. Das, and F. A. “A Location Based Time and Attendance System”, International Journal of Computer Theory and Engineering, Vol. 6, No. 1, February 2014.

[12]Arun Katara1, Mr.Sudesh V. Kolhe, Mr. Amar P. Zilpe, Mr. Nikhil D.Bhele, Mr.Chetan J. Bele, “Attendance System Using Face Recognition and Class Monitoring System”.

[13]Iris recognition by Neha Kak, Rishi Gupta Computer Science Engineering Lingayas Institute of Technology and Management Faridabad, Haryana, India.

[13] Okokpujie K., Etinosa NO., John S., Joy E. (2018) Comparative Analysis of Fingerprint Preprocessing Algorithms for Electronic Voting Processes. In: Kim K., Kim H., Baek N. (eds) IT Convergence and Security 2017. ICITS 2017. Lecture Notes in Electrical Engineering, vol 450. Springer, Singapore.

[14] Student Management through Iris Recognition System Monika Agrawal Computer Science Engineering, GLA University, Mathura, India.

[15]Priya Pasumarti, P. PurnaSekhar, “Classroom Attendance Using Face Detection and Raspberry-Pi”, International Research Journal of Engineering and Technology (IRJET), Volume: 05 Issue: 03 | Mar-2018, 167-171.