

Smart Attendance System Using Face Recognition

Gauri Wankhede, Maheshwari Swami, Gayatri Udawant, Dr. Kalpana Paithane

Student, Department of Electronics and Telecommunication Engineering, MGM's College of Engineering, Nanded, Maharashtra, India

Professor, Department of Electronics and Telecommunication Engineering, MGM's College of Engineering, Nanded, Maharashtra, India

Abstract -The Smart Attendance System using Face Recognition is a computer-vision-based solution developed to automate attendance marking through a laptop camera. The project addresses common difficulties associated with manual registers, identification cards, and touch-based biometric methods, including time consumption, human errors, duplicate entries, and proxy attendance. The implemented system is developed in Python and uses OpenCV and a face-recognition library to process live video. Facial images of authorized users are registered first and converted into numerical facial encodings. During operation, frames from the laptop camera are resized for faster processing, faces are detected, and the resulting encodings are compared with stored encodings using Euclidean Distance. When a valid match is obtained, the attendance module records the recognized name with the current date and time in an Excel sheet and checks previous entries to avoid duplicate marking. Testing of the working model showed successful face detection and recognition under normal operating conditions, responsive live processing after image downscaling, and immediate digital attendance updating. The prototype therefore demonstrates a practical, contactless, and low-additional-hardware approach for attendance management in educational and organizational environments.

Key Words: Face recognition, smart attendance, computer vision, facial encoding, Euclidean Distance, OpenCV.

1. INTRODUCTION

Attendance management is an important administrative activity in educational institutions and organizations. Traditional methods such as manual registers, RFID cards, and fingerprint systems require manual effort and may suffer from errors, duplicate entries, proxy attendance, and physical-contact requirements [5], [6].

To overcome these limitations, face recognition provides a contactless and automated approach for identifying individuals. The proposed Smart Attendance System using Face Recognition uses a laptop camera to capture live video and recognize registered users in real time. Facial images are converted into numerical facial encodings and compared with stored encodings using Euclidean Distance [1], [5]. Face detection is performed using computer-vision techniques [3].

After successful recognition, the system automatically records the person's name, date, and time in an Excel sheet. Frame downscaling is also applied to reduce computational load and

improve real-time performance [2], [5]. The system reduces manual workload, minimizes proxy attendance, and provides organized digital attendance records.

The proposed system can be applied in colleges, schools, offices, training centers, laboratories, and other organizations requiring automated attendance monitoring. Thus, it demonstrates a practical application of computer vision and facial recognition for efficient and contactless attendance management [1], [4].

2. LITERATURE REVIEW

Attendance management has evolved from manual registers to automated systems based on RFID, fingerprint recognition, and face recognition. Traditional methods require considerable human effort and may suffer from errors, lost records, and proxy attendance. RFID systems reduce manual work but can still allow card sharing and require additional hardware [5], [6]. Fingerprint systems provide stronger authentication but require physical contact and may be affected by dirt, damaged fingers, and hygiene concerns [5].

Face recognition has emerged as a suitable contactless solution for automated attendance. These systems capture images through a camera, detect faces, extract facial features, and compare them with stored facial data. This approach reduces manual effort, improves convenience, and supports real-time attendance processing [1], [4].

Viola and Jones introduced the Haar Cascade classifier for rapid object detection, which has been widely applied to face detection because of its relatively low computational requirements and fast processing [3]. However, its performance can be affected by poor lighting, face orientation, and partial occlusion.

Facial encoding is another important component of face recognition. Facial characteristics such as eye distance, nose structure, jawline, lip position, and overall facial geometry are converted into numerical representations. These encodings allow efficient comparison between stored and live facial data while reducing the need to process complete images [1].

For face matching, **Euclidean Distance** can be used to measure the difference between two facial encoding vectors. A smaller distance indicates greater similarity, while a larger distance indicates an unknown face. Its simplicity and computational efficiency make it suitable for small- and medium-scale real-time attendance systems [1], [6].

Real-time performance is also important in camera-based attendance systems. Frame resizing or downscaling reduces the number of pixels processed in each frame, thereby lowering computational load and improving processing speed. Such optimization is particularly useful for systems operating on computers with limited resources [4].

For attendance storage, Excel provides a simple and accessible solution for maintaining digital records. It supports record management, attendance analysis, report generation, and data sharing without requiring a complex database system [5], [6].

Overall, the reviewed literature establishes that face recognition can provide a more contactless, automated, and efficient approach to attendance management. The combination of face detection, facial encoding, mathematical matching, real-time stream optimization, and digital record storage provides the foundation for the proposed Smart Attendance System.

3. PROPOSED SYSTEM AND METHODOLOGY

The proposed Smart Attendance System uses computer vision, facial recognition, and real-time image processing to automate attendance through a laptop camera. The system consists of four main stages: face registration, face detection and recognition, attendance processing, and data storage. During operation, live camera frames are processed, facial encodings are generated, and the detected face is compared with registered facial data. After successful matching, attendance is automatically recorded in an Excel sheet.

3.1 System Architecture

The proposed system consists of a laptop camera, computer-based processing unit, face-recognition software, attendance management module, and Excel-based storage. The camera captures live frames, while the software performs face detection, feature extraction, encoding, and matching. The attendance module then verifies the recognized identity and updates the attendance record.

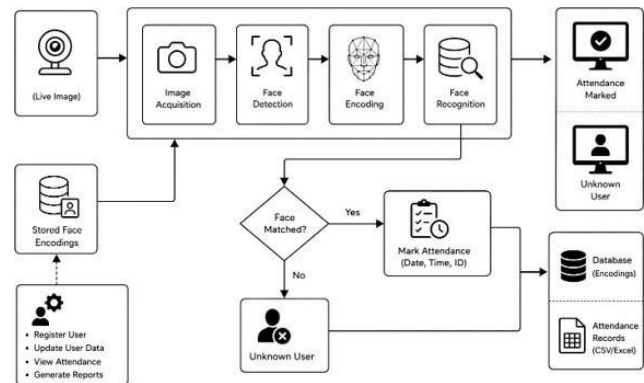


Fig. 1. System Architecture of the Proposed Smart Attendance System

Figure 1 shows the overall architecture of the proposed system. The laptop camera provides live video input to the processing system. The captured frames are processed for face detection and feature extraction. Facial encodings are compared with the registered database, and a successful match is passed to the attendance module. Finally, the person's name, date, and time are stored in the Excel attendance record.

3.2 Hardware and Software Requirements

The system primarily uses a laptop camera and computer. The camera continuously captures live facial images, while the computer performs image processing, recognition, database handling, and attendance management. The software is implemented using Python, OpenCV, Face Recognition, NumPy, Pandas, and Excel-based file handling. OpenCV is used for camera access, frame processing, resizing, and displaying recognition results, while the Face Recognition library generates and compares facial encodings. Pandas supports attendance-sheet management and duplicate-entry prevention.

3.3 Face Registration

The system primarily uses a laptop camera and computer. The camera continuously captures live facial images, while the computer performs image processing, recognition, database handling, and attendance management. The software is implemented using Python, OpenCV, Face Recognition, NumPy, Pandas, and Excel-based file handling. OpenCV is used for camera access, frame processing, resizing, and displaying recognition results, while the Face Recognition library generates and compares facial encodings. Pandas supports attendance-sheet management and duplicate-entry prevention.

3.4 Real-Time Face Detection and Recognition

During real-time operation, the laptop camera continuously captures video frames. The system detects faces within each frame and isolates the facial region for further processing. Important facial characteristics such as eye position, nose structure, jawline, lip structure, and overall facial geometry are converted into numerical facial encodings. These encodings require less processing than complete images and allow efficient comparison with stored data.

The generated live encoding is compared with registered encodings using Euclidean Distance. A smaller distance indicates greater similarity and results in recognition when the matching condition is satisfied. Otherwise, the individual is classified as unknown.

3.5 Attendance management

After successful recognition, the attendance module retrieves the person's identity and current date and time. The information is automatically entered into an Excel sheet containing attendance details. Before adding a new record, the system

Figure 1 shows the overall architecture of the proposed system. The laptop camera provides live video input to the processing system. The captured frames are processed for face detection and feature extraction. Facial encodings are compared with the registered database, and a successful match is passed to the attendance module. Finally, the person's name, date, and time are stored in the Excel attendance record.

3.6 Stream Optimization

Continuous processing of high-resolution video frames can increase computational load and cause system delay. Therefore, the proposed system uses image downscaling/frame resizing before face recognition. Smaller frames reduce the amount of data processed and improve recognition speed while retaining sufficient facial information. The system continuously follows the cycle of frame capture, resizing, face detection, encoding, matching, attendance marking, and result display.

3.7 Overall Working Process

The working sequence can be summarized as follows:

- 1) Start the application and initialize the camera. 2) Load the registered facial encodings and identity labels. 3) Capture a live frame. 4) Resize the frame for faster processing. 5) Detect the face or faces present in the frame. 6) Generate an encoding for each detected face. 7) Compare the live encoding with the stored encodings using Euclidean Distance. 8) If the matching condition is satisfied, identify the user; otherwise display the unknown-user condition. 9) Check the attendance record for an existing entry. 10) If no duplicate exists, store the name, date,

and time in the Excel file. 11) Continue processing the camera stream until the application is stopped.

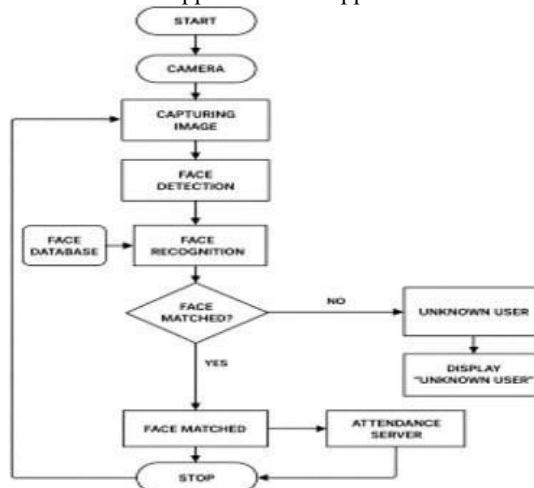


Fig. 2. Flowchart of the Proposed Smart Attendance System

Figure 2 presents the sequential working flow of the system, beginning with camera input and face detection, followed by facial encoding and comparison with stored data. After successful verification, attendance is recorded in the Excel sheet. The system then continues monitoring the camera feed for subsequent users.

4 IMPLEMENTATION

The implementation connects the individual software stages into a continuous attendance pipeline. At program start, the stored user information is loaded and the camera is initialized. The program then enters a loop in which frames are captured, resized, and examined for faces. The recognition library produces numerical encodings, and the comparison stage evaluates whether a stored identity is sufficiently close to the live observation. The successful result is passed to the attendance logic, which performs the duplicate check and updates the Excel record [1], [5].

OpenCV provides the bridge between the camera and the processing pipeline. It reads the live stream, supplies frames to the program, and supports resizing and display operations. The face-recognition library handles the identity-related processing. Python provides the control logic that coordinates these stages, while Pandas and Excel handling routines organize the attendance output [2], [7].

The implementation also distinguishes between recognition and attendance marking. A face can be detected without being recognized, and a recognized face can still be prevented from creating a second record by the duplicate check. This separation

is useful because it keeps the attendance file dependent on verified recognition rather than on simple face presence.

4.1 System Flow

The operational flow is presented in Fig. 2. The process starts with the camera and proceeds through image capture and face detection. The detected face is sent to the recognition stage, which uses the face database for comparison. A successful match leads to the attendance server/module, whereas a failed match is directed to the unknown-user path. The loop then returns to camera processing so that additional users can be evaluated during the same session.

4.2 Data Management

The attendance file acts as the final record produced by the prototype. The report demonstrates a tabular record containing identity information, date and time, and attendance-related fields. Immediate updating makes the result visible as soon as recognition succeeds. The digital format also makes it easier to review attendance history and prepare reports than a paper register [5], [6].

Because facial information is biometric data, a real deployment would require appropriate access control and protection of the stored information. The present prototype is a demonstration of the recognition and attendance workflow; it does not claim to provide a complete enterprise-grade privacy or security framework. This distinction is important when moving from an academic prototype to a production system.

4.3 Experimental Setup and Test Conditions

The working model was evaluated using the same general configuration described in the project report: a laptop with an integrated camera, the Python software environment, registered facial samples, and an Excel attendance file. The purpose of testing was to verify the complete sequence rather than to produce a laboratory-grade biometric benchmark. The main observations concerned whether the system detected faces, recognized registered users, rejected unmatched users, updated the attendance record, avoided duplicate entries, and continued to operate smoothly.

Testing was performed under normal indoor conditions and included moderate variations in facial expression, hairstyle, and lighting. The report states that registered users were recognized successfully within a short period and that the system remained responsive after frame resizing. These observations are reported qualitatively here because the supplied project report does not provide a controlled numerical dataset, confusion matrix, false-accept rate, false-reject rate, or measured frame-rate table.

A more rigorous future evaluation should define the number of registered users, number of test images per user, lighting categories, camera distance, face angle, occlusion conditions,

and repeated trials. Accuracy, precision, recall, false acceptance, false rejection, average recognition time, and processing rate could then be calculated from the recorded outcomes. Such a protocol would make it possible to compare different algorithms or parameter settings objectively.

4.4 Observed Operating Sequence

At the beginning of a session, the program loads the registered face information and starts the laptop camera. As a user enters the camera view, the system captures the current frame and prepares it for processing. A face is detected and represented numerically. The matching stage compares the representation with the stored database. If a valid match is found, the corresponding identity is passed to the attendance module. The module checks whether that identity has already been recorded for the current session and, when appropriate, writes a new row containing the required attendance information.

The same cycle continues for subsequent users. If the camera observes a person who is not represented in the database, the system follows the unknown-user path instead of creating an attendance record. If a recognized person remains in front of the camera, the duplicate-prevention mechanism stops the repeated frames from generating multiple attendance entries. This behavior is essential for a video-based system because the same person may appear in dozens of consecutive frames.

The observed sequence demonstrates why attendance automation cannot be reduced to face detection alone. Detection answers the question “Is there a face?”, while recognition addresses “Whose face is it?”. Attendance logic

then addresses a third question: “Should this recognition event create a new record?”. The prototype combines all three decisions in a single workflow.

4.5 Testing Observations

The testing observations can be grouped into five categories: detection, recognition, recording, responsiveness, and stability. In the detection category, the camera successfully supplied frames and the software identified faces under normal indoor conditions. In the recognition category, registered users were generally identified without repeated attempts, while unknown faces were not accepted for attendance. These outcomes correspond to the intended decision flow shown in Fig. 2.

For recording, the system transferred successful recognition results into the Excel attendance sheet and included the person's name, date, and time. The project report also demonstrates that the record remains in a structured tabular form. This is important because an attendance system is useful only when the recognition event is converted into a record that can be reviewed later.

For responsiveness, the key observation was the effect of image downscaling. The report describes faster frame processing, reduced visible lag, and lower hardware utilization after resizing the camera frames. These improvements are consistent with the basic principle that fewer pixels require fewer computations. The optimization therefore supports continuous operation on the available laptop hardware.

5 RESULTS AND DISCUSSION

The working model was tested to examine face detection, recognition, attendance recording, stream responsiveness, and overall operational stability. The observations reported in the project indicate that authorized users could be detected and recognized through the laptop camera under normal lighting conditions. The system then updated the Excel record with the recognized identity and current date and time [4], [5], [6].

Recognition remained effective when moderate variations such as small changes in facial expression, hairstyle, and lighting were introduced. The project did not report a numerical accuracy percentage; therefore, this paper intentionally reports the result qualitatively rather than introducing an unsupported performance value. Most registered users were recognized without repeated scanning attempts, while unmatched faces produced larger comparison distances than matched faces [1], [4].

The use of numerical facial encodings was also beneficial from a processing perspective. Instead of repeatedly comparing complete image files, the system compared compact numerical representations. The report observations indicate that the encodings were sufficiently consistent for the same person across repeated attempts, which supported stable recognition during testing.

Stream optimization produced a clear practical benefit. Full-sized frames required more processing effort, whereas resized frames reduced computational load and improved the responsiveness of the live system. The optimized pipeline operated without noticeable lag during the demonstrated tests and did not require additional camera hardware. These observations agree with the role of frame resizing described in computer-vision implementations [4], [7].

5.1 Attendance Recording Result

The attendance module successfully transferred recognition results into the Excel sheet. The record was updated after successful identification and retained the name, date, and time information required for attendance monitoring. The report also demonstrates that the digital record can be reviewed directly, making the output convenient for academic or office use.

The duplicate-prevention mechanism is especially important in a live camera environment. A recognized person may remain in front of the camera for several consecutive frames; without a record check, the same presence could generate repeated entries. By checking existing attendance information before insertion, the implementation avoids this form of redundancy [5], [6].

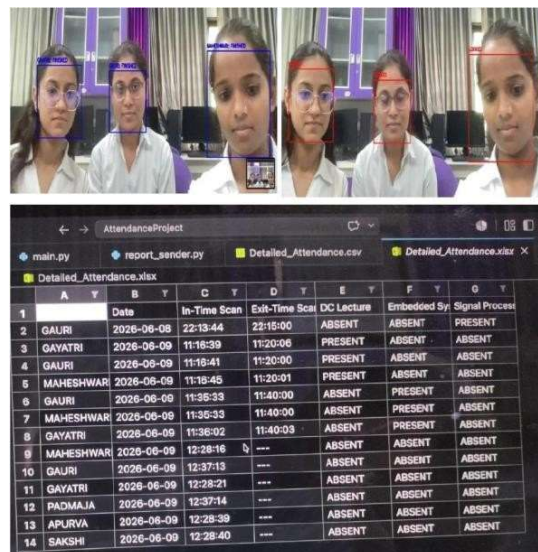


Fig. 3. Working-model output showing recognition and digital attendance record

Figure 3 demonstrates the practical output of the prototype. The upper portion shows faces identified by the recognition interface, while the lower portion shows the corresponding attendance information in an Excel-style record. The project report notes organized storage, immediate updating, and easy accessibility of the digital record.

5.2 Security and Practical Discussion

The system provides stronger resistance to ordinary proxy attendance than a system based only on cards or manually entered identifiers because the attendance decision depends on facial verification. A user cannot simply hand an RFID card to another person and obtain attendance through possession of the token. However, face recognition should not be described as completely immune to spoofing or unauthorized use. A production system would need additional safeguards such as liveness detection, protected storage, access control, and secure data handling.

Contactless operation is another practical advantage. The user does not need to touch a shared scanner, and the same camera can be used for several people. This can reduce the physical interaction required during attendance and can make

the procedure more convenient in classrooms and offices [5], [6].

5.3 Limitations

The prototype has limitations that should be considered before deployment. Recognition performance can change with poor illumination, low camera quality, extreme face angles, partial occlusion, or significant appearance changes. Large user databases can also increase computational demand. In addition, biometric information and attendance records require appropriate privacy protection [3], [4], [5].

The current model is also a prototype centered on a laptop camera and local Excel storage. It does not provide the scalability, centralized authentication, cloud synchronization, or advanced anti-spoofing controls expected in a large institutional deployment. These limitations do not prevent the model from demonstrating the core concept, but they define the boundary of the current implementation.

5.4 Result Interpretation

The results indicate that the prototype met its principal functional objective: a registered person appearing before the laptop camera could be recognized and used to trigger an attendance update. The result is meaningful because it confirms the interaction of several modules rather than the operation of one library in isolation. Camera acquisition, frame processing, facial encoding, matching, and spreadsheet handling all participated in producing the final record.

The qualitative recognition result also suggests that the selected representation was sufficiently stable for the tested users. The report notes that small changes in expression and hairstyle did not prevent recognition under the observed conditions. At the same time, the report identifies poor lighting, low-quality cameras, extreme face angles, and partial obstruction as factors that can reduce performance. Therefore, the correct interpretation is that the system works reliably within the tested operating range rather than under every possible environmental condition.

The attendance result is equally important. The Excel record was updated immediately after successful recognition and remained organized in tabular form. This reduces the administrative steps required after attendance is taken. Instead of collecting handwritten sheets and entering them later, the digital record is created as part of the recognition event itself.

The responsiveness result demonstrates the value of optimization. Processing reduced-size frames lowered the computational burden and made the live recognition cycle smoother. The report specifically observes faster frame

processing, reduced lag, better real-time behavior, and lower hardware utilization after downscaling [4], [7].

5.5 Error Sources and Mitigation

Several error sources can affect a face-recognition attendance system. Illumination changes may alter the visual appearance of the face. A camera placed too low, too high, or too far away can reduce the quality of the detected facial region. Side-facing poses and partial occlusion can remove useful facial information. Motion can also blur the captured image. These conditions can make the live encoding less similar to the registered encoding even when the person is genuine [3], [4].

At the software level, an overly strict matching condition can reject genuine users, while an overly relaxed condition can increase the chance of incorrect matches. The project therefore uses a comparison condition and an unknown-user path rather than automatically accepting every detected face. In a future implementation, the matching threshold should be selected using a representative validation set and documented together with the test conditions.

Duplicate attendance is a different type of error. It does not arise from incorrect recognition; it arises because a video stream repeatedly observes the same recognized person. The project mitigates this through an attendance-history check before writing a new entry. This illustrates why application-level logic is necessary even when the underlying recognition module is functioning correctly.

5.6 Privacy and Security Considerations

A face-recognition system processes biometric information and therefore deserves more careful protection than an ordinary attendance spreadsheet. Registered facial representations and attendance records should be accessible only to authorized personnel. In a production system, storage should be protected by authentication and appropriate permissions, and unnecessary copies of biometric information should be avoided. The project report identifies privacy and security as important limitations for face-recognition deployment [3], [4], [5].

The prototype is primarily intended to demonstrate functionality, so its local Excel storage should not be interpreted as a complete security architecture. If the system is

expanded, secure database storage, encrypted communication, user access control, audit logging, backup procedures, and retention rules should be considered. Liveness detection can also be added to reduce the risk of attempts to fool a camera using a photograph or another presentation attack.

6 APPLICATIONS AND FUTURE SCOPE

The proposed system is applicable wherever routine attendance or identity verification is required. Educational institutions can use it for classroom attendance, examination verification, laboratories, libraries, and training centers. Offices can use the same workflow for employee attendance and entry monitoring. The project report also identifies potential use in government organizations, healthcare facilities, industrial units, research laboratories, events, hostels, and access-control environments [4], [5], [7].

A major future direction is multi-face recognition. The present prototype is intended as a compact attendance solution, whereas a larger classroom system could identify several faces in a single frame and process their attendance together. This would reduce the need for users to approach the camera one by one and would make the system more suitable for larger groups.

Cloud connectivity could allow attendance information to be synchronized to a centralized service and accessed by authorized administrators from different locations. A mobile application could provide attendance history and notifications. These extensions would transform the local Excel-based prototype into a broader attendance-management platform, although they would also introduce additional requirements for authentication, privacy, networking, and data security.

Internet of Things (IoT) integration is another possible enhancement. Attendance recognition could become one input to a smart classroom or smart-office system. For example, a recognized user could trigger access-control actions or classroom automation. CCTV connectivity could also extend the system from a single laptop camera to a larger monitored environment.

Recognition robustness can be improved through stronger deep-learning models, better cameras, improved training data, and techniques such as liveness detection. The project report also identifies possible extensions including real-time analytics, mask handling, voice assistance, and integration with other smart-management functions. Such improvements should be evaluated experimentally rather than assumed to provide a fixed performance benefit.

6.1 Scalability Considerations

The present design is appropriate for a small or moderate number of registered users, but larger deployments would place greater demands on storage, computation, and administration. As the number of encodings increases, the matching stage has more reference data to compare. A larger system could use an

optimized indexing strategy or a dedicated recognition service rather than performing all processing locally on a single laptop.

A centralized database would also improve multi-class or multi-department management. Authorized administrators could maintain user registration, attendance records, and reporting from one controlled platform. The Excel-based output used by the current prototype can be viewed as a simple first-level storage mechanism that may later be replaced by a relational or cloud database while retaining the recognition logic.

6.2 Integration with Institutional Systems

An institutional deployment could connect recognition with existing student-information or human-resource systems. Instead of maintaining names only inside a local dataset, the application could retrieve authorized identities from a central record and return attendance events to the institutional platform. This would reduce duplicate data entry and allow attendance information to contribute to reports, alerts, and analytics.

Mobile or web interfaces could provide authorized users with attendance summaries. Notifications could be generated when an attendance event is recorded, while administrators could review daily or monthly trends. Such integration would extend the project from an attendance marker to an attendance-management service, but it would require stronger authentication and careful handling of personal information.

6.3 Suggested Deployment Roadmap

A practical development path can be organized into stages. The first stage is the present local prototype, where recognition and Excel recording are validated. The second stage can introduce a structured database, administrator login, and protected user registration. The third stage can add multi-face recognition, liveness detection, and improved analytics. The final stage can connect the service to cloud or institutional infrastructure and provide controlled mobile or web access.

At each stage, testing should be repeated because improving one part of the system can change the behavior of another part. For example, replacing the recognition model may alter processing time, matching thresholds, or the format of stored encodings. Similarly, adding network access introduces new failure modes that are not present in a standalone laptop. A staged approach therefore allows the project to retain its simple working core while progressively increasing capability and security.

7 CONCLUSION

The Smart Attendance System using Face Recognition demonstrates a practical method for replacing manual attendance marking with a contactless, automated workflow. The prototype combines a laptop camera, Python-based image processing, facial encoding, Euclidean Distance matching,

attendance logic, and Excel storage into a single operational system. This combination addresses several problems associated with manual registers and token-based attendance, particularly repetitive manual work, duplicate entries, and proxy attendance [4], [5], [6].

The experimental observations show that the implemented model can detect and recognize registered users under normal operating conditions, update attendance records automatically, and maintain responsive camera processing after frame downscaling. The use of numerical facial encodings provides an efficient representation for comparison, while the duplicate check improves the consistency of the attendance file [1], [7].

The project also highlights the difference between a working prototype and a production biometric system. Lighting, camera quality, face position, privacy, scalability, and anti-spoofing remain important concerns. Nevertheless, the demonstrated architecture provides a clear foundation for future improvements such as multi-face recognition, cloud storage, mobile access, IoT integration, stronger deep-learning models, and liveness detection. The work therefore establishes a useful academic implementation of computer vision for automated attendance management.

ACKNOWLEDGEMENT

The authors sincerely thank Dr. Mrs. K. P. Paithane for her guidance, technical suggestions, and support throughout the development of the mini project. The authors also acknowledge the Department of Electronics & Telecommunication Engineering, MGM's College of Engineering, Nanded, for providing the academic environment and facilities required for the work. The authors are grateful to the faculty members, staff, parents, and friends who supported the project and its documentation.

REFERENCES

- [1] M. Turk and A. Pentland, "Eigenfaces for Recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.
- [2] Python Software Foundation, "Python Documentation."
- [3] P. Viola and M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features," *Proceedings of the IEEE*

Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2001.

[4] IEEE Xplore Digital Library, "Face Recognition Based Attendance System Research Papers."

[5] IJRASET, "Smart Attendance System using Face Recognition," *International Journal for Research in Applied Science & Engineering Technology*, vol. 10, issue 5, 2022.

[6] IJERT, "Automated Attendance System Using Face Recognition," *International Journal of Engineering Research & Technology*, vol 9, issue 6, 2020.

[7] OpenCV Development Team, "Open Source Computer Library Documentation."