Face Recognition Attendance System using Support Vector Machine

Devi R.[1], Logeshwaran[2], Sedhu Ram[3], Sridhar[4], Tamizharasu[5]

[1-5]Department of Information Technology, Government College of Technology, Coimbatore, India

E-mail: [1]r.devi@gct.ac.in, [2]logeshvvaran@gmail.com, [3]sedhuram135@gmail.com, [4]sridhareswar3@gmail.com, [5]tamizharasu040@gmail.com

Abstract

The Face Recognition Attendance System takes attendance marking to the next level by automating the process through face recognition. It addresses the issues found in older methods like manual registers and RFID. By harnessing the power of machine learning and computer vision tools like Dlib and SVM, this system crafts unique face embeddings for each person at the moment of capture, enabling real-time identification. It's built with Python, using the Flask framework and a MySQL database, and comes packed with features like new user registration, real-time monitoring, and the option to export data in CSV format. A custom dataset was put together, featuring 50 images for each user, all created using the BlazeFace model right in the browser. This system achieves an accuracy rate of 98%, along with macro precision and recall scores of 0.99 and 0.98, respectively. This represents a major advancement in recognizing individuals already stored in the database. This model is a contactless and scalable design, that ensures better accuracy, security, and efficiency, making it a preferable choice for schools and workplaces, while also reducing manual errors and preventing fake attendance.

Keywords: Face Recognition, Attendance System, Machine Learning, Computer Vision, Support Vector Machine (SVM), Dlib, Real-Time Attendance, Automated Attendance Tracking, Contactless Biometrics.

1. Introduction

Organizations and institutions have to maintain attendance on a daily basis, but conventional methods like manual registers, telephones, and RFID systems are not efficient as they are time-consuming, error-prone, and vulnerable to proxy attendance. We propose a

DOI: https://doi.org/10.36548/jscp.2025.3.005
Received: 11.07.2025, received in revised form: 13.08.2025, accepted: 30.08.2025, published: 10.09.2025

computer vision and machine learning-based intelligent wireless Face Recognition Attendance System that has the potential to completely automate the process with the aim of eliminating all the above issues. This project utilized BlazeFace for face detection directly in the web browser to take images post-registration, without needing any third-party applications or specialized hardware. Coupled with an SVM classifier and Dlib for facial embedding extraction, the system performs precise and efficient real-time operations. Face data collection, model training, real-time recognition via webcam or CCTV, and automatic attendance marking are all included in the system, which is built on top of a MySQL database and Python's Flask framework. After training on an edited dataset of 50 images per user that were taken via the browser, the proposed SVM model outperformed traditional CNN and KNN models in terms of speed and dependability, achieving an astounding 98% accuracy, 0.99 precision, and 0.98 recall. We also developed a limited website that allows administrators to add users, monitor attendance, and export data to make the system user-friendly and convenient. This work provides a lightweight, scalable, and highly accurate solution that combines machine learning with the easy access of browser applications for organizations aiming to improve their attendance systems with the least amount of infrastructure and maximum efficiency.

2. Related Work

In the past, numerous studies have been conducted on face recognition-based attendance systems to address the disadvantages of traditional systems such as roll call and RFID cards. One such system [1] suggested an internet-based real-time face-recognition and mask-detection system with an 81.8% accuracy rate in recognizing faces and an 80% accuracy rate in recognizing masks. Though executed for the purposes of convenience and security, it was affected by issues like reliance on internet access and privacy. In another study [2], a group image-based personal identification system was tested using Histogram of Oriented Gradients (HOG) and an SVM with an RBF kernel, achieving 95% accuracy with reduced proxy attendance but was susceptible to light and camera positioning. In the research [3], comparison experiments were attempted with CNN and LBPH algorithms, finding that CNN was more accurate (99%) and less susceptible to outside influences than LBPH (92%), at higher computation costs. Live attendance using the K-Nearest Neighbors (KNN) algorithm [4] also yielded promising results but was susceptible to environmental influences as well as compromised biometric data privacy. Hybrid approaches, such as CNNs with LSTM networks [5] achieving 99.82% accuracy and CNNs with Genetic Algorithms (GA) [6], which increased recognition speed and accuracy to 96.49%, were also said to have made further progress. According to both of the two papers, the dataset's range was extremely low, which is concerning for generalizability. Another study [7] employed YOLOv7 for 100% effective face detection, exceeding techniques such as HOG + Dlib and LBPH, but it had scalability problems due to its computationally demanding nature. Systems based on Haar Cascade, LBPH, Eigenfaces, and Fisherfaces [8] [9] showed low costs and real-time performance, but most of the time, they lacked explicit information on training procedures, dataset variability, and performance metrics. An Android-based attendance system with OTP and GPS [10] verifies user presence within a geofence using a time-sensitive code, providing additional security and automation but relying on smartphone and network availability. Overall, data-dependent solutions for automated attendance systems have so far been validated in the literature, yet they are frequently undermined by limitations of dataset generalizability, real-world feasibility, and biometric privacy areas in which this proposed system aims to improve.

3. Proposed Methodology

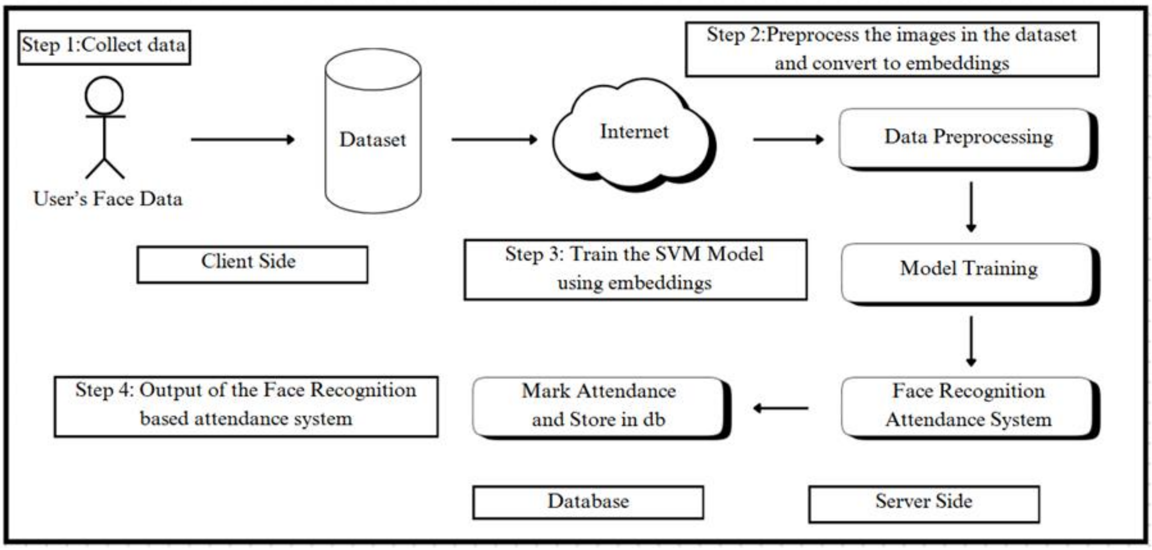


Figure 1. System Architecture

The Face Recognition Attendance System [11][12] is designed to automate the attendance process by using facial recognition technology as shown in Fig.1. By combining machine learning and computer vision techniques, the system ensures accurate, secure, and real-time attendance tracking. The complete workflow is divided into six key stages: dataset creation, image preprocessing, model training, real-time recognition, attendance logging, and system management via a web interface. It is depicted in Fig.2.

3.1 Dataset Creation

The initial step is to develop a facial image dataset for every user at the time of registration. A user (student or staff) inputs their name and roll number (or employee number), and then several facial images are captured via a regular webcam. For consistent and real-time facial detection, BlazeFace is applied directly in the browser via TensorFlow.js. 50 photos per user are collected and saved in structured folders identified by the user's ID. The pictures are stored on the server in PNG format, and the corresponding user data are saved within a MySQL database, which also keeps track of where every user's image folder is for simple access and management.

3.2 Image Preprocessing and Embedding Generation

Once the images are captured, they are preprocessed to extract meaningful facial features. This is done using a Dlib pretrained model. First, the shape_predictor_68_face_landmarks.dat model detects 68 key facial landmarks such as the jawline, eyes, nose, and mouth to properly align the face. The aligned face is then passed on to the dlib_face_recognition_resnet_model_v1.dat model, which uses a ResNet-34 model to generate a 128-dimensional facial embedding. The embeddings are used to uniquely represent every face and are normalized and stored in a pickle file along with their respective labels (roll number and name). This preprocessing ensures that the facial information is normalized and ready for classification.

3.3 Model Training with SVM

The system must then be trained for face identification using an SVM classifier. The facial embeddings and their corresponding user labels are read from the pickle file. User names are converted to numerical labels by the LabelEncoder of scikit-learn. The system is trained with a linear-kernel SVM so that it can distinguish between users based on facial embeddings. The model is saved after training to be utilized in future recognition. To measure its performance quantitatively, the system calculates parameters such as accuracy, precision, recall, and F1-score, and graphs these through a confusion matrix that identifies any misclassifications or areas for improvement.

3.4 Real-Time Face Recognition and Attendance Logging

After training the system, it is ready for user recognition in real time. Using a live webcam or CCTV footage, the system is constantly detecting faces using Dlib's face detector. The system extracts embeddings for every face detected as it extracted embeddings during preprocessing and then passes the extracted embeddings through the trained SVM model to predict the user identity. Before taking note of attendance, it checks whether the user has already logged in today in an effort to avoid redundancy. Otherwise, the user's name, ID, and time are logged in the MySQL database. For enhanced user experience, the identified name and status of attendance are updated on the screen in real time, as is shown in Fig.7. and Fig.8.

3.5 Web-Based User Interface

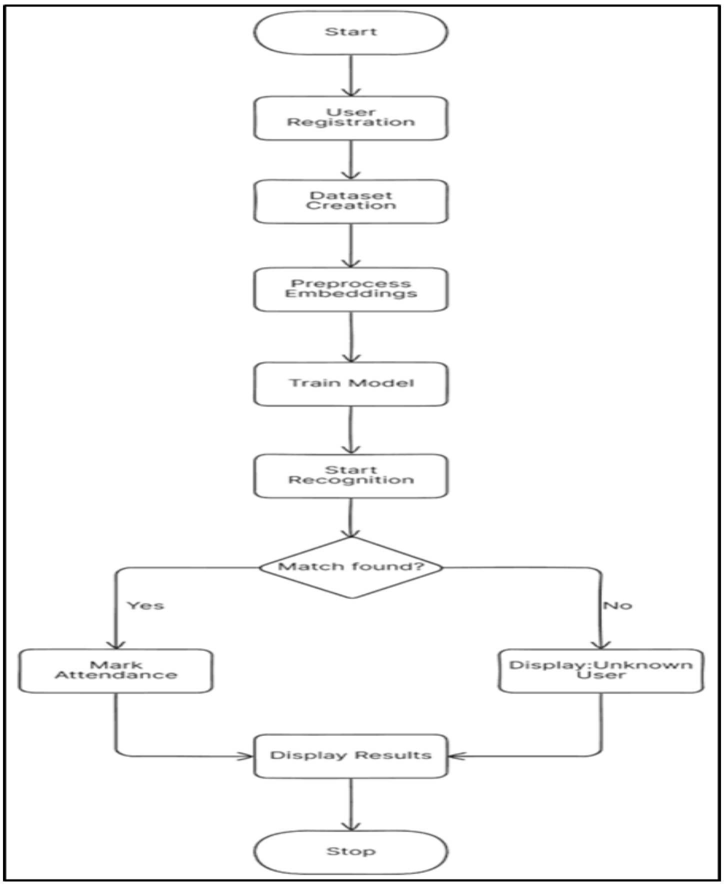


Figure 2. Overall Process Flow

To manage the entire system easily, a web-based interface was developed using Flask (for backend) and Bootstrap (for frontend styling), with added JavaScript to improve user interaction. The interface allows administrators to perform several functions: register new users, capture images, preprocess embeddings, train the model, start face recognition, and view or export attendance logs. Attendance records can be downloaded as CSV files, and registered user data can be viewed in tabular format. The interface is responsive, ensuring it works smoothly on both desktops and mobile devices, making it accessible and administrator-friendly, as shown in Fig.6.

3.6 Single Image Embedding Conversion and Matching

Single-image recognition is also supported by the system for easy use. It begins with face detection through dlib.get_frontal_face_detector() and landmark detection through the shape_predictor_68_face_landmarks.dat model. Such landmarks help with alignment before passing the face to ResNet-based recognition, generating a 128-dimensional embedding vector. During recognition, these embeddings are compared with each other based on Euclidean distance, with a smaller distance indicating a better match. This enables effective and robust matching under varying conditions, such as lighting or slight variations in facial angles.

3.7 Discussion on the Support System

The Face Recognition Attendance System takes advantage of the integration of machine learning libraries and frameworks with database tools that support real-time efficient identification and data management. Dlib provides the primary computer vision capabilities in the form of facial embeddings. The Support Vector Machine (SVM) is the classifier used to effectively identify separate identities. The Flask web framework is employed in the application to offer a lightweight scalable backend for real-time communication. MySQL is employed for storing attendance and face data to ensure secure management of user and attendance data. The BlazeFace model facilitates development of an individualized dataset by means of face image retrieval directly through a browser. All these characteristics combine to create a feasible system that is precise in its identification and simple to operate in schools and workplaces.

4. Results and Discussion

The performance of the Face Recognition Attendance System is evaluated using key evaluation metrics: accuracy, precision, recall, F1-score, and a confusion matrix. These metrics provide a comprehensive understanding of the model's effectiveness and reliability, as shown in Fig.3.

4.1 Accuracy

Accuracy reflects the overall correctness of the model's predictions, calculated as the ratio of correctly classified samples to the total samples:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{1}$$

The SVM model achieved an impressive 98% accuracy, demonstrating strong overall performance in recognizing registered users.

4.2 Precision

Precision measures the accuracy of positive predictions, representing the proportion of correctly identified positive cases among all predicted positives:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

The model maintained a macro average precision of 0.99, reflecting its ability to minimize false positives across multiple classes.

4.3 Recall (Sensitivity)

Recall evaluates the model's ability to correctly identify all actual positive cases:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

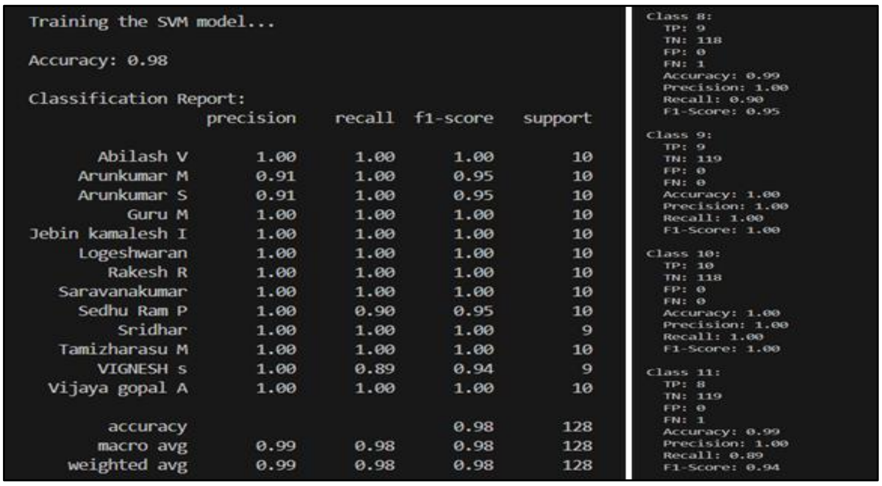
The system achieved an average recall of 0.98. While most classes performed optimally, Classes 8 and 11 exhibited slightly lower recall values (0.90 and 0.89, respectively) due to false negatives.

4.4 F1-Score

The F1-score is the harmonic mean of precision and recall, providing a balanced measure when the dataset is imbalanced:

$$\mathrm{F1-score} = 2 \times \frac{\mathrm{Precision} \times \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}} \tag{4}$$

The system achieved an average F1-score of 0.98, highlighting a strong balance between precision and recall, ensuring reliable performance. Notably, Classes 8 and 11 experienced minor drops (0.95 and 0.94) due to false negatives, though the scores remain high and indicative of a robust system. The bar chart representation of precision, recall, and F1-score is showed in Fig.4



```
Training the SVM model...

Accuracy: 0.98

Classification Report:
                    precision    recall  f1-score   support

        Abilash V       1.00      1.00      1.00        10
      Arunkumar M       0.91      1.00      0.95        10
      Arunkumar S       0.91      1.00      0.95        10
           Guru M       1.00      1.00      1.00        10
Jebin kamalesh I        1.00      1.00      1.00        10
      Logeshwaran       1.00      1.00      1.00        10
         Rakesh R       1.00      1.00      1.00        10
   Saravanakumar        1.00      1.00      1.00        10
      Sedhu Ram P       1.00      0.90      0.95        10
          Sridhar       1.00      1.00      1.00         9
   Tamizharasu M        1.00      1.00      1.00        10
        VIGNESH s       1.00      0.89      0.94         9
   Vijaya gopal A       1.00      1.00      1.00        10

         accuracy                           0.98       128
        macro avg       0.99      0.98      0.98       128
     weighted avg       0.99      0.98      0.98       128
```

```
Class 8:
  TP: 9
  TN: 118
  FP: 0
  FN: 1
  Accuracy: 0.99
  Precision: 1.00
  Recall: 0.90
  F1-Score: 0.95

Class 9:
  TP: 9
  TN: 119
  FP: 0
  FN: 0
  Accuracy: 1.00
  Precision: 1.00
  Recall: 1.00
  F1-Score: 1.00

Class 10:
  TP: 10
  TN: 118
  FP: 0
  FN: 0
  Accuracy: 1.00
  Precision: 1.00
  Recall: 1.00
  F1-Score: 1.00

Class 11:
  TP: 8
  TN: 119
  FP: 0
  FN: 1
  Accuracy: 0.99
  Precision: 1.00
  Recall: 0.89
  F1-Score: 0.94
```

Figure 3. Evaluation Metrics

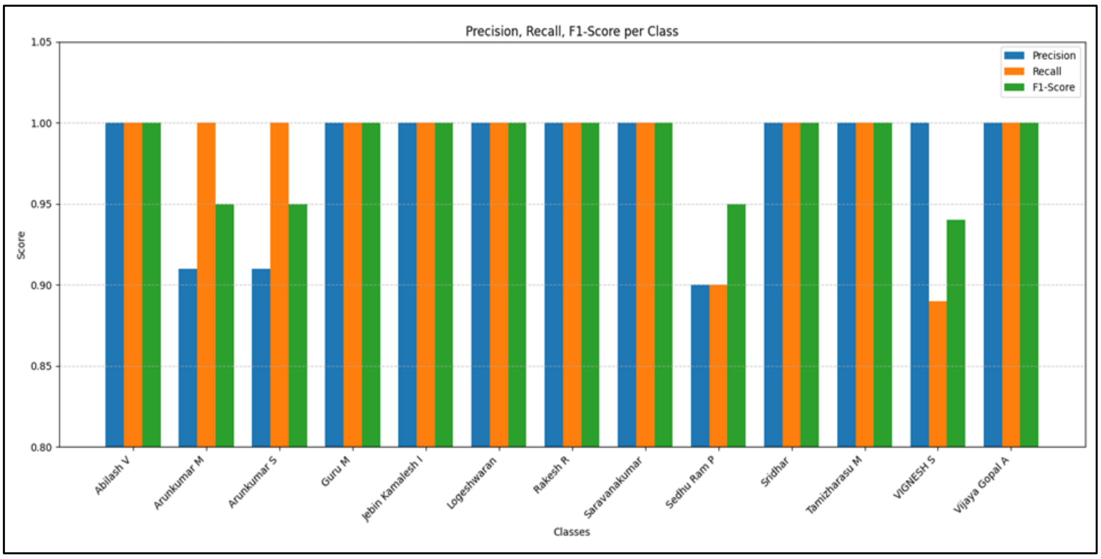


Figure 4. Barchart Representing the Precision, Recall, and F1-score for Each Class

4.5 Confusion Matrix

The confusion matrix provides an insightful breakdown of the model's predictions, categorizing results into:

- True Positives (TP): Correctly identified positive samples.
- True Negatives (TN): Correctly identified negative samples.
- False Positives (FP): Incorrectly predicted positive samples (Type I Error).
- False Negatives (FN): Incorrectly predicted negative samples (Type II Error).

The confusion matrix analysis confirms the model's high reliability, with most classes achieving perfect classification (precision, recall, F1-score of 1.00). Minor misclassifications resulted in false negatives for Classes 8 and 11, leading to slightly reduced recall values. Despite these small deviations, no false positives were observed in any class - a critical factor in preventing incorrect attendance logging. It is shown in Fig.5.

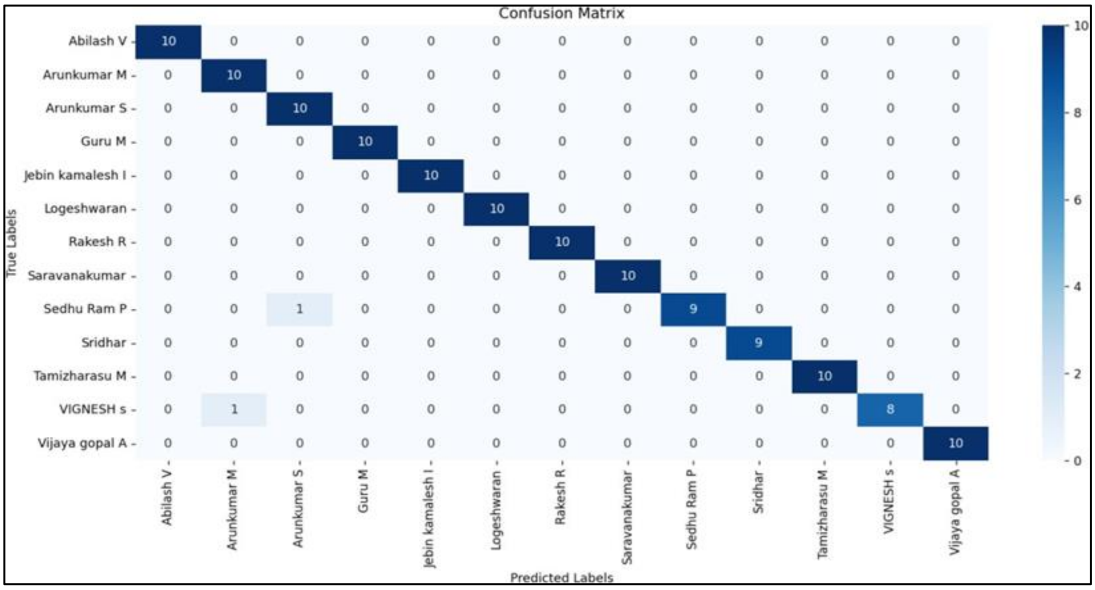


Figure 5. Confusion Matrix

4.6 False Negative Analysis and Misclassification Insights

Despite the high performance of the Support Vector Machine (SVM) model, which achieved an overall accuracy of 98%, analyzing the errors is crucial to further improving reliability. Specifically, false negatives (FN) are of concern, as they represent instances where the model failed to correctly identify a true class label. In real-world systems like attendance tracking this can lead to missing records for valid entries [24] [25]. The table below highlights the false negatives observed for each misclassified class and the corresponding percentage:

Table 1. False Negative Rate (FNR) Per Class

Class Name	False Negatives (FN)	Total Samples	False Negative %
Sedhu Ram P	1	10	10.00%
Vignesh S	1	9	11.11%
Overall Total	2	128	1.56%

Table.1. indicates that only 2 out of 128 instances were falsely classified as another class, resulting in a low overall false negative rate of 1.56%, which is acceptable, though still a candidate for optimization.

Analysis of Misclassification Causes

The misclassifications observed in the confusion matrix and classification report can be attributed to the following potential reasons:

1. **Visual Similarity Between Classes:** Individuals such as Sedhu Ram P and Arunkumar S might have visually similar facial characteristics, leading the model to incorrectly predict the class. This is a common challenge in face recognition tasks.
2. **Insufficient Data Samples:** Certain classes (e.g., Vignesh S) have fewer training samples (only 9), which may have affected the classifier's ability to generalize well for these cases.
3. **Image Quality Variations:** Variations in image resolution, lighting, facial orientation, and background can introduce noise in feature extraction, resulting in decreased model performance for certain images.
4. **Model Sensitivity:** Although the SVM model is robust, it can be sensitive to overlapping features between classes, especially when operating in high-dimensional space with limited separation margins.

Recommendations for Improvement

To reduce false negatives and enhance classification reliability, the following steps are recommended:

1. **Data Augmentation:** Increases the diversity and volume of training data using techniques such as rotation, flipping, and brightness adjustment.
2. **Balanced Class Distribution:** Ensures uniform sample counts across all classes to prevent bias.
3. **Feature Enhancement:** It utilizes advanced feature descriptors like Histogram of Oriented Gradients (HOG) [12] or apply Principal Component Analysis (PCA) for dimensionality reduction before classification.
4. **Hyperparameter Tuning:** It optimizes SVM parameters (e.g., kernel type, C value, gamma) through grid search or cross-validation.

4.7 Overall Performance

The classification report, that provides averages with weights and overall scores for precision, recall, and F1-scores of 0.99, 0.98, and 0.98, respectively, further demonstrates the model's durability. These indicators illustrate how the system functions for various user types. Reliable attendance tracking is ensured by the system's high accuracy, near-perfect precision, and recall, which also reduce errors and prevent proxy attendance. A few minor adjustments could make the model even more helpful, particularly when it comes to handling false negatives in specific classes, where it yields good outcomes.

4.8 Comparison of CNN, SVM, And KNN for Face Recognition

To evaluate the performance of the proposed Support Vector Machine (SVM)-sensors-based face recognition system, a comparison was conducted with two other commonly used classifiers: Convolutional Neural Networks (CNN) and k-Nearest Neighbors (KNN). The comparison indicates the relative merits and deficiencies of each model in face recognition and confirms the adoption of SVM as the most suitable algorithm for real-time attendance systems.

SVM Performance Overview (Proposed System)

For facial recognition under the same test environment, the Support Vector Machine (SVM) achieved 98% accuracy. SVM is well-suited to high-dimensional space, learns optimal decision boundaries to prevent overfitting, has a fast prediction time for real-time tracking, and is not affected by noise; but is slower to train and requires accurate hyperparameter adjustment. On the other hand, K-Nearest Neighbors (KNN), while simple to implement, is heavily impacted by the curse of dimensionality in high-dimensional embedding spaces, leading to slow inference times for large distance calculations, sensitivity to noise, and general lower accuracy and robustness than SVM. The comparative results are shown in Table.2

Table 2. Comparative Results of Different Machine Learning Algorithms

Model	Accuracy	Training Time	Inference Speed	Robustness	Scalability
CNN	82%	High	Medium	Medium	Medium
SVM	98%	Medium	High	High	High
KNN	85%	Low	Low	Low	Low

4.9 Comparison of the Existing System with SVM Face Recognition

A Support Vector Machine (SVM)-based face recognition attendance system offers a modern, contactless, and highly accurate (e.g., 98%) solution, automating processes [13] [14] and reducing "buddy punching." While effective and clean, it can be environmentally sensitive, and SVM training can be computationally demanding with scalability problems for very large sets of data. In comparison to other technologies, Retina scanning [15][16] is more accurate but much more intrusive; fingerprint systems [17][18] are in contact with the user, and have hygiene concerns and are prone to finger condition vulnerabilities; manual attendance [19] can be subject to human mistake, forgery, and inefficiency, and RFID systems[20][21], though convenient, are token-based and hence reusable, making them less secure against proxy attendance compared to biometric facial recognition. An attendance system based on QR code[22][23] tracks presence electronically by scanning individual, usually dynamic, QR codes with a smartphone, offering an efficient, contactless, and usually low-cost substitute for paper-based records. The comparative results with the existing systems are shown in Table.3

Table 3. Comparative Results with Existing Systems

Model	Accuracy	Training Time	Inference Speed	Robustness	Scalability	Cost
Retina	100%	High	Medium	High	Medium	Very high
Finger print	95%	High	Medium	High	Medium	High
RFID	85%	Medium	Low	Low	Low	Medium

QR code	82%	NA	Based on classroom strength	Medium	High	Low
Manual	75%	NA	Based on classroom strength	Medium	Low	Low
SVM-based Face Recognition	98%	Medium	Low	Medium	High	Medium

4.10Results of Face Recognition Attendance System

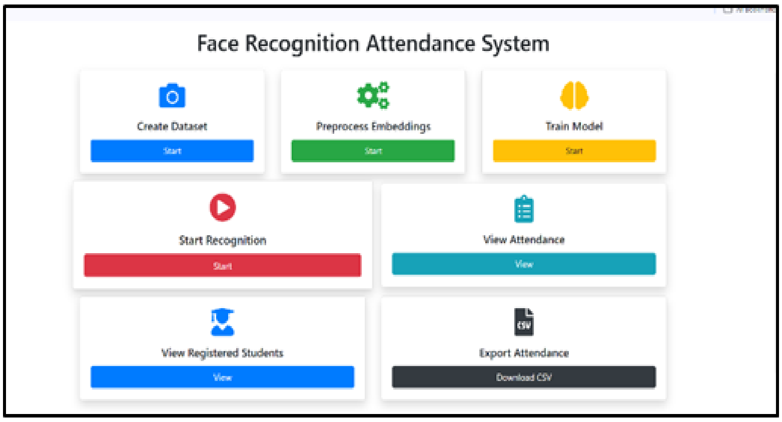


Figure 6. Dashboard of Face Recognition System

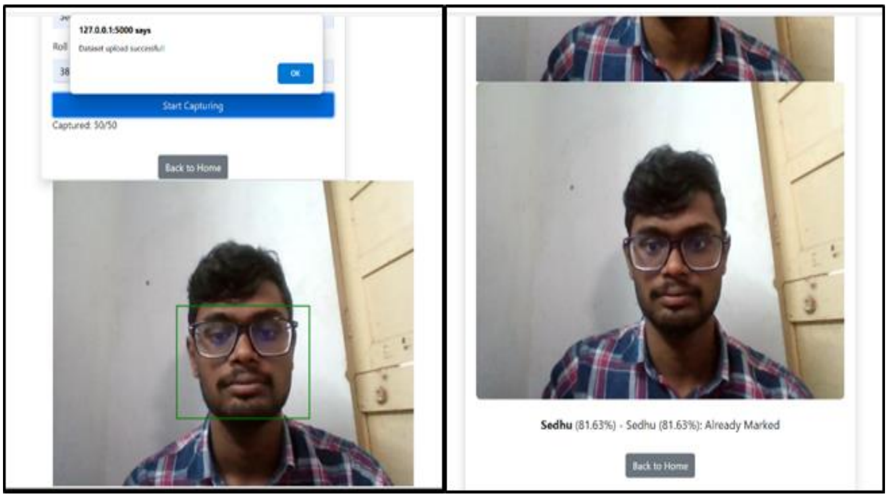


Figure 7. Face Recognition Attendance

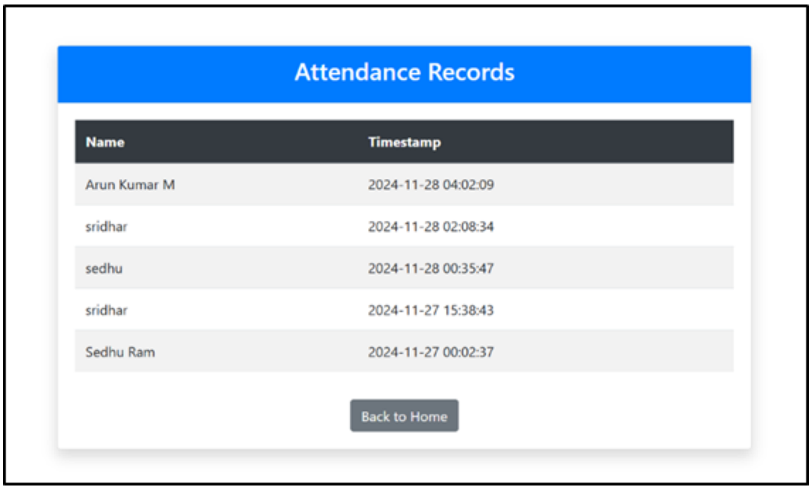


Figure 8. Attendance Records

5. Conclusion

The Face Recognition Attendance System offers a contactless, secure, and effective substitute for conventional attendance techniques by utilizing an SVM classifier and contemporary computer vision. It eliminates proxy attendance and minimizes the need for human intervention by automating every step of the process, from data processing to real-time recognition and logging. It exceeds CNN and KNN in important metrics and attains 98% accuracy. A scalable design for numerous deployments, strong backend/frontend integration, necessary features like tracking, data export, and registration, and precise, real-time attendance are some of the main contributions. Considering its good performance, the system has a false negative rate of 1.56%, requires more preprocessing for images that were noisy or poorly lit, and demands significant computing power for SVM training. Scalability to very large datasets is another disadvantage; this may call for optimization models or deep learning. Future improvements will include cloud-based storage, multi-camera support, liveness detection, mobile/web extensions, and deep learning models (e.g., CNN, FaceNet) for large-scale accuracy. In the end, this AI-powered project offers a useful, dependable, and secure virtual attendance solution that is very beneficial for contemporary work and learning settings.

References

[1] Kamil, Muhammad Haikal Mohd, Norliza Zaini, Lucyantie Mazalan, and Afiq Harith Ahamad. "Online attendance system based on facial recognition with face mask detection." Multimedia tools and applications 82, no. 22 (2023): 34437-34457.

[2] Sanap, D. S., Ms Sakshi Narwade, Mr Sahil Goge, and Mr Krishna Pandit. "Face Recognition Based Attendance System Using Histogram of Oriented Gradients and Linear Support Vector Machine." European Journal of Theoretical and Applied Sciences 1, no. 6 (2023): 904-915.

[3] Budiman, Andre, Ricky Aryatama Yaputera, Said Achmad, and Aditya Kurniawan. "Student attendance with face recognition (LBPH or CNN): Systematic literature review." Procedia Computer Science 216 (2023): 31-38.

[4] A. Umalkar, S. S. Manhas, I. Chandiwala, and N. Bhagat, “Face Recognition Based Attendance System Using Real Time Data,” 2023. [Online]. Available: www.ijcrt.org

[5] Shukla, Ashish Kumar, Archana Shukla, and Raghvendra Singh. "Automatic attendance system based on CNN–LSTM and face recognition." International Journal of Information Technology 16, no. 3 (2024): 1293-1301.

[6] Ojo, Olufemi S., and Mayowa O. Oyediran. "Development of an Improved Convolutional Neural Network for an Automated Face-Based University Attendance System." (2023).

[7] Lateef, Ahmad S., and M. Kamil. "Face Recognition-Based Automatic Attendance System in a Smart Classroom." Iraqi Journal for Electrical and Electronic Engineering 20, no. 1 (2023): 37-47.

[8] Mekala, V., Vibin Mammen Vinod, M. Manimegalai, and K. Nandhini. "Face recognition based attendance system." International Journal of Innovative Technology and Exploring Engineering 8, no. 12 (2019): 520-525.

[9] Vyshnavi, Ms K., Suhel Shaik, Mylavarapu Santosh Kumar, M. S. V. Praveen, and Swetha Potti. "Facial Recognition Based Attendance System Using Opencv."

[10] Sakthivel, & Janakiraman, Sreerambabu & Kalidasan, & Riyaz, Mohammed. (2023). Attendance Information of Student by Android based using OTP and GPS. International

Journal for Research in Applied Science and Engineering Technology. 11. 610-613. 10.22214/ijraset.2023.55034.

[11] Khan, Er & Vadanagara, Saad & Chamadiya, Mohammed & Nafees, Khan & Shah, Ashmiza. (2023). Face Recognition Based Attendance System. International Journal of Advanced Research in Science, Communication and Technology. 392-396. 10.48175/IJARSCT-9225.

[12] Prasad, D., C. Sirisha, G. Lakshmi Tanuja, and M. Durga Harinadh. "Face recognition attendance system using hog algorithm." Dogo Rangsang Research Journal (2023): 225-261.

[13] Jha, Phul Babu, Arjun Basnet, Biraj Pokhrel, Bishnu Pokhrel, Gopal Kumar Thakur, and Surya Chhetri. "An automated attendance system using facial detection and recognition technology." Apex Journal of Business and Management 1, no. 1 (2023): 103-120.

[14] Sain, Avoy, Samrat Dutta, Rimpi Saha, and Unmesh Mandal. "Automated Facial Recognition based Attendance System using OpenCV in Python." International Journal of Scientific Research in Computer Science, Engineering and Information Technology (2023): 105.

[15] S. By et al., "RETINA BASE ATTENDANCE SYSTEM," International Journal of Scientific Research in Engineering and Management (IJSREM) International Journal of Scientific Research in Engineering and Management, 2022, [Online]. Available: www.ijsrem.com.

[16] Chempavathy, B., M. Dhanalakshmi, M. G. Varun, Vishvash Bohra, and Vrushil Ashokkumar Kalsariya. "An improved attendance monitoring system through facial recognition using RetinaFace algorithm." In 2024 3rd International Conference for Innovation in Technology (INOCON), IEEE, (2024): 1-5.

[17] K. Bhojwani, K. Lohar, A. Awasare, S. Golatkar, and V. bodhale, "Fingerprint Based Attendance System," 2024. [Online]. Available: www.ijrpr.com.

[18] Adesoba, Olayiwola, and Israel Joseph, "A A Fingerprint-Based Attendance System for Improved Efficiency," 2025, ITEGAM-JETIA 11 (51), 9-19. doi: 10.5935/jetia.v11i51.1305.

[19] S. Dassanayake, G. Sir John, D. Dassanayake, and W. Wanniarachchi, “Challenges of Manual Attendance System Towards Student Motivation Ashen Wanniarachchi Challenges of Manual Attendance System Towards Student Motivation,” 2021. [Online]. Available: https://www.researchgate.net/publication/356202082.

[20] H. U. Zaman, J. S. Hossain, T. T. Anika, and D. Choudhury, “RFID based attendance system,” in 2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT), IEEE, Jul. (2017): 1–5. doi: 10.1109/ICCCNT.2017.8204180.

[21] Krupali Panchal, Parthvi Jingar, Om Vataliya, and Harshul Rathod, “RFID Based Attendance System for Department,” 2024, IJISAE, vol.12.no.14, pp. 505–514. Available: https://www.ijisae.org/index.php/IJISAE/article/view/4687.

[22] K. J. Liew and T. H. Tan, “QR Code-Based Student Attendance System,” in Proceedings - 2021 2nd Asia Conference on Computers and Communications, ACCC 2021, Institute of Electrical and Electronics Engineers Inc., (2021): 10–14. doi: 10.1109/ACCC54619.2021.00009.

[23] R. D. Bacuna and B. G. Dadiz, “eSAM : Attendance System Using QR Codes in Romblon State University-Cajidiocan Campus,” in 2022 International Conference for Advancement in Technology (ICONAT), IEEE. doi: 10.1109/ICONAT53423.2022.9725959.

[24] M. Ali, A. Diwan, and D. Kumar, “Attendance System Optimization through Deep Learning Face Recognition,” International Journal of Computing and Digital Systems, 2024, vol. 15, no. 1: 1527–1540. doi: 10.12785/ijcds/1501108.

[25] Kumar, S. Jain, and M. Kumar, “Face and gait biometrics authentication system based on simplified deep neural networks,” International Journal of Information Technology, 2023, vol. 15: 1005–1014.