International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 11 | Nov - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Smart Campus Face Recognition System for Automated Attendance Tracking and Security Enhancement

Shweta Sathe[1], Mansi Korde[2], Supriya Garje[3], Komal Bhagwat[4], Prof. J.R. Mahajan[5]

[1]*Adsul's Technical Campus, Chas*
[2]*Adsul's Technical Campus, Chas*
[3]*Adsul's Technical Campus, Chas*
[4]*Adsul's Technical Campus, Chas*
[5]*Adsul's Technical Campus, Chas*

Abstract - *This project presents a Smart Campus Face Recognition System designed to automate attendance tracking and enhance campus security using artificial intelligence and computer vision. The system automatically detects and recognizes student faces to mark attendance in real time, generate Excel reports, and send notifications to parents. It includes division-based filtering, uniform detection, and tracking of late or half-day arrivals to ensure accuracy and discipline. By integrating AI, image processing, and data analytics, the system reduces manual effort, prevents manipulation, and improves transparency, accountability, and communication within educational institutions.*

Key Words: Face Recognition, Smart Attendance System, Artificial Intelligence, Machine Learning, Computer Vision, Campus Security, Image Processing, Automation, Data Analytics, Student Monitoring.

1.INTRODUCTION

Educational institutions today require efficient, accurate, and transparent systems for managing student attendance and campus discipline. Traditional methods like manual roll calls or QR-based systems are slow, error-prone, and vulnerable to proxy attendance. To address these challenges, the proposed Smart Campus Face Recognition System leverages artificial intelligence and computer vision to automate attendance marking and enhance security.

Using live camera feeds, the system performs real-time face detection, recognition, and liveness verification to ensure authenticity. It enforces division-based attendance, identifies late arrivals or half-day presence, and checks uniform compliance for discipline. Additionally, it generates automated Excel attendance reports and shares summaries or alerts with administrators and parents. This integration of AI-driven automation significantly improves efficiency, transparency, and communication within educational institutions.

2. LITERATURE SURVEY

The literature review highlights significant advancements in artificial intelligence and computer vision applied to attendance and security systems. Parkhi et al. (2015) introduced the VGGFace model, which achieved high accuracy in facial recognition using deep learning, while Zhang et al. (2016) proposed the MTCNN framework for fast and precise real-time face detection—both of which influence the recognition component of this project. Bhattacharya and Bandyopadhyay (2019) demonstrated an automated attendance system integrating facial recognition with databases, reducing manual effort and errors. Similarly, Rahman and Hossain (2021) developed a uniform detection system using image classification to ensure dress code compliance, and Patel and Joshi (2020) presented an automated notification framework that improved communication between parents and institutions. Sharma and Gupta (2022) explored AI-driven campus surveillance, showcasing how object detection and facial analysis enhance safety. While previous works focus on isolated functions such as attendance or security, the proposed system combines these approaches into a unified, intelligent solution for real-time attendance, discipline monitoring, and campus security.

3. METHODOLOGY

The proposed Smart Campus Face Recognition System uses a modular and automated approach integrating AI, computer vision, and real-time data processing to ensure accurate and transparent attendance management. First, high-definition cameras capture live video streams, and deep learning models such as MTCNN or RetinaFace detect faces. Extracted facial features are matched with stored embeddings using recognition algorithms like FaceNet or ArcFace. Once identified, the system automatically marks attendance in a central database with a timestamp, preventing duplication or proxy entries.

To maintain accuracy, division verification ensures only students of the assigned class are recorded. A uniform detection module checks compliance through image analysis before finalizing attendance. The system then generates Excel reports with details such as timestamps, remarks, and status, which can be printed or stored.

Additionally, automated notifications are sent to parents via email or SMS, informing them of attendance, late arrivals, or absences. The system also detects half-day and late arrivals based on predefined time thresholds. All data are securely encrypted and accessible only to authorized administrators, ensuring privacy and security. This end-to-end workflow provides an intelligent, efficient, and secure solution for modern campus attendance management.

4. SYSTEM REQUIREMENTS AND INTERFACES

The proposed system is designed as a desktop-based application using Anaconda Navigator and Spyder IDE. It provides an intuitive and user-friendly interface for administrators to input data, view real-time outputs, and interact with AI modules for testing and prediction visualization. The system uses an SQLite (DBSQL3) database for data storage, retrieval, and management.

Hardware Requirements:

Processor: Intel Core i5 or higher
RAM: Minimum 8 GB (recommended for smooth AI model execution)
Storage: Minimum 500 GB HDD or 256 GB SSD
Operating System: Windows 10 / Windows 11
Input Devices: Standard keyboard and mouse
Audio Devices: Functional microphone and speakers (for voice module)

Software Requirements:

Operating System: Windows 10 / Windows 11
Programming Language: Python 3.10 or above
Development Environment: Anaconda Navigator, Spyder IDE
Database: SQLite (DBSQL3)
AI & Machine Learning Libraries: Transformers, NumPy, Pandas, Scikit-learn, TensorFlow / PyTorch
Visualization Tools: Matplotlib, Seaborn
Supporting Libraries: OpenCV, Tkinter (for GUI), OS, and Sys modules

This configuration ensures smooth execution of the AI-driven modules and efficient interaction between the user interface, database, and machine learning components.

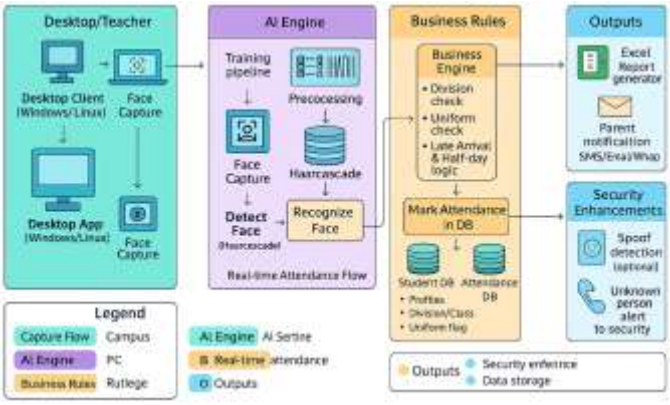

Fig 1. High Level Sytem Architecture

5. ADVANTAGES

Automation of Attendance:
Eliminates manual roll calls by using face recognition for accurate and real-time attendance marking.
Improved Accuracy and Security:
Reduces errors and prevents proxy attendance through AI-based face verification and liveness detection.
Time Efficiency:
Saves valuable class and administrative time by automating attendance and report generation.

Uniform and Division Verification:
Ensures only authorized students in proper dress code and assigned division are marked present.

Real-Time Parent Communication:
Automatically notifies parents via email or SMS about student attendance, late arrivals, or absences.

Enhanced Campus Surveillance:
Integrates with security systems to strengthen monitoring and ensure campus safety.

Data Transparency and Accessibility:
Attendance data is securely stored, easily retrievable, and exportable for analysis or reporting.

6. LIMITATIONS

Lighting and Camera Quality Dependency:
System accuracy may decrease in poor lighting or with low-resolution cameras.
Facial Obstruction Issues:
Masks, headgear, or face coverings can affect recognition accuracy.
Initial Setup Cost:
Requires investment in quality cameras, processing hardware, and storage infrastructure.
Database Scalability:
Managing large-scale image datasets may demand higher storage and computational power.
Privacy Concerns:
Storing and processing facial data raises data security and consent-related issues.
Internet/Network Dependence (for Notifications):
Automated alerts and cloud backups rely on stable internet connectivity.
Model Maintenance:
AI models may need periodic retraining to maintain accuracy as students' appearances change over time.

7. OUTPUT

Fig.2 Registration Form

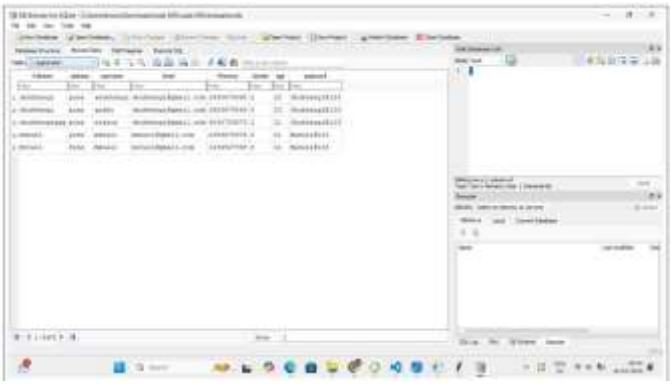

Fig.3 Database

8.CONCLUSION

The Smart Campus Face Recognition System effectively integrates Artificial Intelligence, Machine Learning, and Computer Vision to automate student attendance and enhance campus security. Using the Haarcascade algorithm, the system accurately detects and recognizes faces, marking attendance in real time through an intuitive desktop interface. It incorporates key features such as late arrival detection, half-day identification, and uniform verification, ensuring accuracy and discipline. Automated Excel report generation and parent notifications improve transparency and accountability, while reducing manual effort and proxy attendance. Overall, the system demonstrates a reliable, scalable, and technology-driven approach to modern campus management, with potential for future expansion into cloud integration, CCTV-based monitoring, and advanced deep learning models for even higher accuracy.

REFERENCES

[1] P. Viola and M. J. Jones, "Robust Real-Time Face Detection," Interna tional Journal of Computer Vision, vol. 57, no. 2, pp. 137–154, 2004. [On line]. Available: https://doi.org/10.1023/B:VISI.0000013087.49260. Fb

[2] G. Bradski, "The OpenCV Library," Dr. Dobb's Journal of Software Tools, 2000. [Online]. Available: https://opencv.org/

[3] R. Lienhart and J. Maydt, "An Extended Set of Haar-like Features for Rapid Object Detection," IEEE International Conference on Image Pro cessing (ICIP), vol. 1, pp. 900–903, 2002. [Online]. Available: https: //doi.org/10.1109/ICIP.2002.1038171

[4] M. Lutz, "Programming Python," 4th ed., O'Reilly Media, 2010. [Online]. Available: https://learning.oreilly.com/library/view/ programming-python-4th/9781449398712/

[5] I. Goodfellow, Y. Bengio, and A. Courville, "Deep Learning," MIT Press, 2016. [Online]. Available: https://www.deeplearningbook.org/

[6] S. Sharma and R. Patel, "AI-Based Smart Campus Systems: Au tomation, Security, and Learning Analytics," IEEE Access, vol. 10, pp. 12435–12450, 2022. [Online]. Available: https://doi.org/10.1109/ ACCESS.2022.3154512

[7] M. A. Turk and A. P. Pentland, "Eigenfaces for Recognition," Journal of Cognitive Neuroscience, vol. 3, no. 1, pp. 71–86, 1991. [Online]. Available: https://doi.org/10.1162/jocn.1991.3.1.71 23

[8] J. Grayson, "Python and Tkinter Programming," Manning Publica tions, 2000. [Online]. Available: https://www.manning.com/books/ python-and-tkinter-programming

[9] A. Geitgey, "Face Recognition with Python," GitHub Repository, 2017. [Online]. Available: https://github.com/ageitgey/face_recognition

[10] N. Li and T. Zhao, "Artificial Intelligence in Education: Enhancing Learning and Administration," International Journal of Smart Education Systems, vol. 5, no. 3, pp. 102–116, 2021. [Online]. Available: https: //doi.org/10.4018/IJSESD.2021070105