DOI: http://dx.doi.org/10.26483/ijarcs.v17i4.7545

Volume 17, No. 4, July-August 2026

ISSN No. 0976-5697

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

VISION MARK – AUTOMATED SMART ATTENDENCE

Dr. K. Prem Kumar[1] , P.S.R.K Sharma[2], E. Saloni[3], Mohd Junaid[4], D. Nithin[5]

[1]Professor & HoD Department of AI & ML, ACE Engineering College, Hyderabad, Telangana, India

[2,3,4,5]Department of Artificial Intelligence & Machine Learning,
ACE Engineering College, Hyderabad, Telangana, India

Abstract - Vision Mark is a face recognition–based smart attendance system designed to overcome the limitations of traditional attendance methods such as manual registers and card-based systems, which are often time-consuming, error-prone, and vulnerable to proxy attendance. The system is built using advanced computer vision and deep learning techniques to ensure accuracy, automation, and reliability. Vision Mark operates through two primary modules: registration and attendance marking. In the registration phase, users' facial data is captured and processed using a pre-trained deep learning model to generate unique facial embeddings. These embeddings serve as distinctive digital representations of each individual and are securely stored in the system database. In the attendance marking phase, real-time video streams are analyzed to detect and recognize faces. The system compares detected faces with stored embeddings using similarity measurement techniques to accurately identify individuals. Once a match is found, attendance is automatically recorded with a timestamp, ensuring precise tracking. To prevent duplicate entries, the system incorporates date-based constraints, allowing only one attendance record per user per session or day. This eliminates redundancy and enhances data integrity. Additionally, Vision Mark is designed to be scalable and adaptable for both academic institutions and organizational environments. It reduces administrative workload, minimizes human intervention, and enhances overall efficiency. The system can also be extended with features such as cloud integration, real-time notifications, analytics dashboards, and multi- device support, making it a robust and future-ready solution for modern attendance management.

Keywords - Face Recognition, Smart Attendance System, Computer Vision, Deep Learning, FaceNet, OpenCV DeepFace, Automated Attendance Management.

I. INTRODUCTION

In today's fast-paced academic and organizational environments, maintaining accurate attendance records is essential but often challenging when using traditional methods. Manual attendance systems and card-based approaches are widely used; however, they are time- consuming, prone to human errors, and susceptible to fraudulent practices such as proxy attendance. These limitations highlight the need for a more reliable, efficient, and automated solution With the rapid advancement of computer vision and deep learning technologies, face recognition has emerged as a powerful tool for identity verification. Modern approaches such as Face Net utilize deep neural networks to generate unique facial embeddings, enabling accurate and efficient recognition of individuals based on their facial features. Similarly, real-time face detection techniques like the Viola–Jones algorithm allow systems to detect human faces quickly in live video streams, making them suitable for practical applications. Vision Mark is designed as a smart attendance system that leverages these technologies to automate the entire attendance process. The system eliminates the need for physical contact or manual intervention by using facial recognition to identify individuals in real time. It consists of two main modules: a registration module, where user facial data is captured and converted into embeddings, and an attendance module, where faces are detected and matched against stored data to mark attendance automatically. The system ensures high accuracy and reliability by using similarity-based matching techniques and maintains data integrity by preventing duplicate entries through date-based constraints Additionally, Vision Mark stores attendance records in a structured database, enabling easy retrieval, analysis, and reporting. By reducing administrative workload minimizing errors, the system enhances overall efficiency in attendance management.Further more, Vision Mark is designed with scalability and flexibility in mind, making it suitable for deployment in educational institutions, offices, and other organizations. With potential enhancements such as liveness detection, cloud integration, and mobile-based monitoring, the system represents a step toward fully automated and intelligent attendance management solutions.

II. LITERATURE SURVEY

The development of Vision Mark is supported by significant research in the fields of face recognition, real-time face detection, and automated attendance systems.

[1] Schroff et al. introduced FaceNet: A Unified Embedding for Face Recognition, a deep learning model that maps facial images into a compact embedding space using a convolutional neural network trained with a triplet loss function. Embeddings of the same person are clustered together, while those of different individuals are separated. Recognition is achieved using Euclidean distance, which serves as the core recognition mechanism in Vision Mark.

[2] Viola and Jones proposed Rapid Object Detection Using a Boosted Cascade of Simple Features, which uses Haarlike features, integral images, and a cascaded classifier for efficient real-time face detection. This foundational work influenced the design of Vision Mark's face detection pipeline.

[3] Shireesha et al. explored Face Recognition Based Attendance Systems, demonstrating that deep learning techniques can automate attendance marking with high reliability. Their findings confirm the feasibility of the approach adopted in Vision Mark.

[4] Kumar et al. proposed a CNN-Based Face Recognition System for Automated Attendance, highlighting the effectiveness of convolutional networks for high-accuracy facial identification in institutional environments. Overall, these research works highlight the effectiveness of combining deep learning-based face recognition with real-time detection for automated systems, providing a strong theoretical and practical foundation for Vision Mark.

III. SYSTEM ANALYSIS

Existing Systems

Traditional attendance management systems can be broadly categorized as follows: Manual Attendance: Attendance is recorded using paper registers or roll calls. This process is time-consuming, labour intensive, and prone to errors such as incorrect marking or missing entries. Proxy attendance is easily facilitated. RFID-Based Systems: Students or employees scan ID cards embedded with RFID tags to mark attendance. While faster than manual systems, these face issues such as card loss, misuse, and proxy attendance via card sharing. Biometric Systems (Fingerprint/Iris): These systems use unique physical traits for identification. While more secure, they have drawbacks including hardware dependency, maintenance costs, slower processing for large groups, and hygiene concerns due to physical contact. Early Face Recognition Systems: Some systems use basic algorithms like Eigenfaces or Haar Cascades. While improving automation, they struggle with varying lighting, pose variations, and accuracy issues in real-world environments.

Proposed System

Vision Mark introduces a face recognition-based smart attendance system that is fully automated, contactless, and highly accurate, designed to overcome all the above limitations. Face Recognition-Based Identification: Vision Mark uses deep learning (FaceNet embeddings via DeepFace) to uniquely identify individuals based on facial features, eliminating proxy attendance. Two-Module Architecture: A Registration Module captures facial images and converts them into embeddings stored in a database. In contrast, an Attendance Module processes real-time video to detect and recognize faces, marking attendance.
Duplicate Prevention Mechanism: Date-based constraints ensure that attendance is marked only once per day per user, preventing redundancy and maintaining data integrity. Contactless and Hygienic: Unlike biometric systems, Vision Mark requires no physical contact, making it more suitable for modern environments. The system is designed to scale for large institutions and can be extended with cloud integration, liveness detection, mobile/web dashboards, and multi-device support.

IV.SOFTWARE REQUIREMENTS SPECIFICATION

A. Hardware Requirements Processor:

Intel i3 or higher; RAM: Minimum 4 GB (8 GB recommended); Storage: Minimum 500 MB free space; Camera: HD Webcam (recommended for better accuracy); Display: Standard monitor; Optional: GPU for faster deep learning processing.

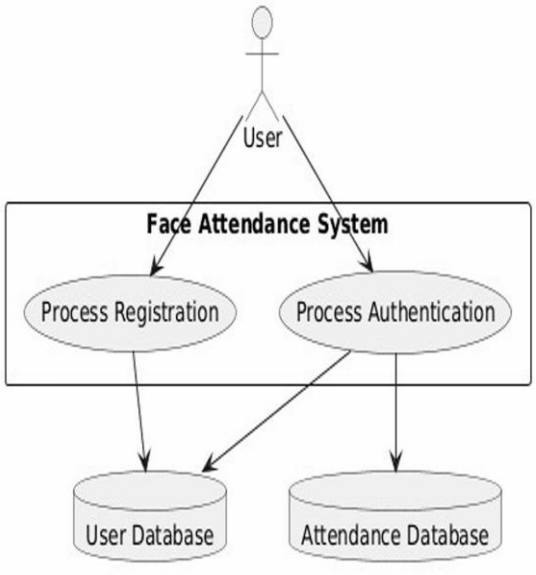
B. Software Requirements Operating System:

Windows / Linux / macOS; Programming Language: Python (version 3.8 or above); Libraries: OpenCV (image processing), DeepFace (face recognition and embeddings), NumPy, Pandas, SciPy, Matplotlib, PySide6 (GUI); Database: SQLite; Development Tools: VS Code / PyCharm.

V. SYSTEM DESIGN

A. Data Flow

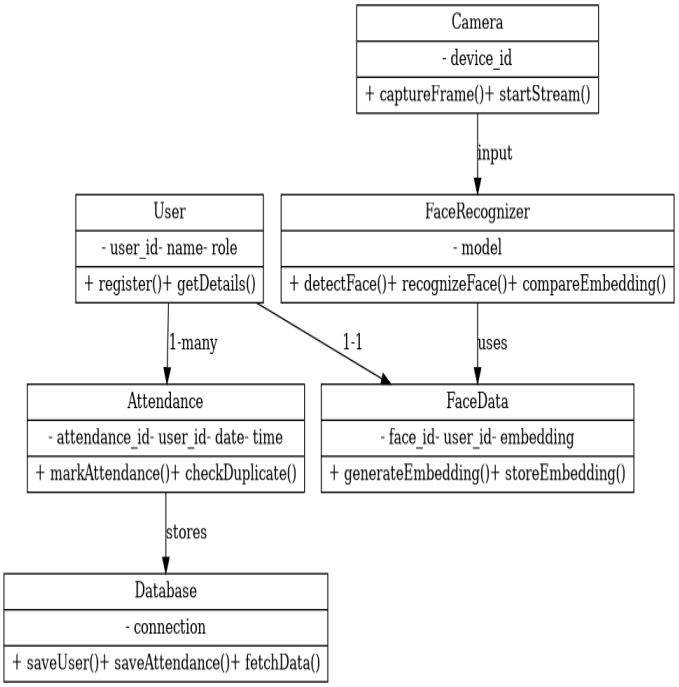
The data flow in Vision Mark proceeds as follows: the camera module continuously captures real-time video frames. Each frame is passed to the face detection module, which identifies face regions using DNN-based detection. Detected face regions are forwarded to the face recognition module, which generates FaceNet embeddings and compares them with stored embeddings in the database. Upon successful identification, the attendance management module records the user's attendance with a precise timestamp. The database module handles all storage and retrieval operations, while the GUI module displays recognition results in real time.



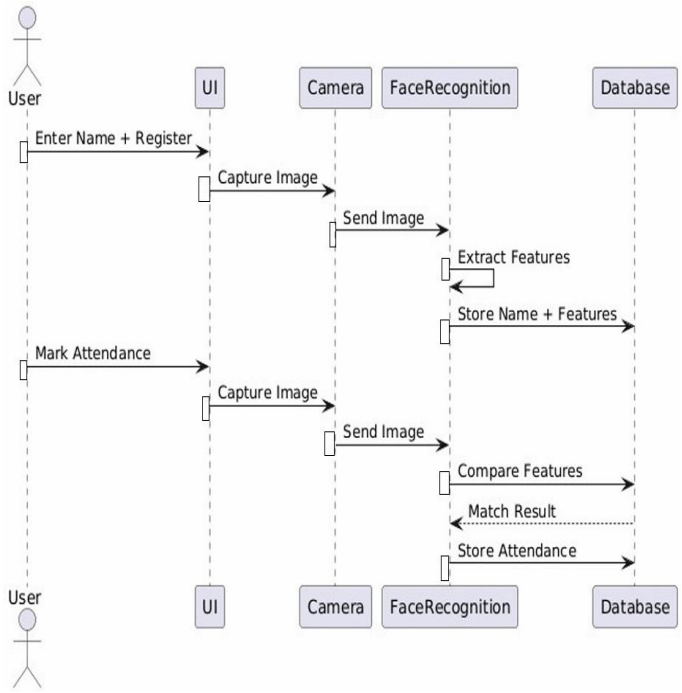
B. UML Diagrams

Class Diagram: The major classes include Video Thread (camera capture and DNN detection), Recognition Thread

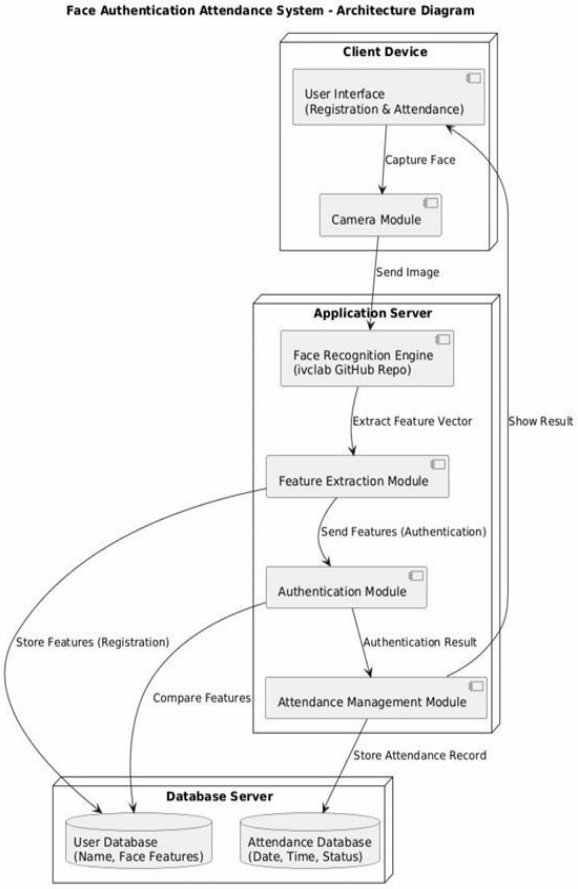
(FaceNet embedding extraction and similarity matching), Database (FAISS-based storage and retrieval of embeddings), and Main Window (PySide6 GUI controller).t the equation as a graphic and insert it into the text after your paper is styled.



Sequence Diagram: The sequence of events for attendance marking proceeds from camera frame capture to face detection, embedding extraction, database similarity lookup, identity resolution, attendance recording, and GUI update.



C. Architectural Design The Vision Mark system follows a modular layered architecture comprising: (1) Presentation Layer — PySide6 GUI; (2) Processing Layer — OpenCV DNN detection and DeepFace FaceNet recognition; (3) Business Logic Layer —attendance management and duplicate prevention; (4) Data Layer — SQLite database for user records and attendance logs.



VI. SYSTEM MODULES

Registration Module: Captures multiple facial images of users via the camera, generates FaceNet embeddings using DeepFace, and securely stores user details and embeddings in the database. A duplicate face threshold (cosine similarity > 0.80) prevents re-registration of already enrolled users.

Face Detection Module: Detects human faces from real-time video using OpenCV's DNN-based Caffe model (ResNet SSD). A confidence threshold of 0.6 filters weak detections, and every 10th frame is processed to balance accuracy with performance.

Face Recognition Module: Extracts FaceNet embeddings from detected face regions using DeepFace. Embeddings are normalized and compared with stored embeddings using cosine similarity. A minimum similarity threshold of 0.70 is enforced for identification.

Attendance Management Module: Marks attendance automatically upon successful recognition with a timestamp. An attendance cooldown of 60 seconds prevents duplicate entries within a session, and date-based constraints ensure only one record per day per user.

Database Module: Manages storage and retrieval of user data, facial embeddings, and attendance records using SQLite. Supports fast similarity lookup using FAISS-based indexing.

GUI Module: Provides an interactive PySide6-based interface displaying the live camera feed, recognition results, registration controls, and attendance records in tabular form.

Camera Module: Captures real-time video input using OpenCV VideoCapture, providing continuous frames to the face detection pipeline. Supports HD webcams for improved accuracy.

VII. IMPLEMENTATION

A. Core Tools and Libraries

The system is implemented in Python using: OpenCV (DNN face detection), DeepFace (FaceNet embeddings), PySide6 (cross-platform GUI), SQLite (database), NumPy and SciPy (numerical operations and distance calculations), Pandas (data management), and Matplotlib (visualization).

B. Key Implementation Details

DNN Face Detection (VideoThread): The Video Thread class inherits from QThread and runs the camera capture and face detection loop concurrently with the GUI. OpenCV's DNN module loads a pre-trained Caffe ResNet SSD model to detect faces from each captured frame. Multi-Capture Registration: During registration, five facial images are captured, processed through DeepFace to generate FaceNet embeddings, and averaged into a single representative embedding. L2 normalization is applied to ensure consistent cosine similarity computations. Duplicate detection is performed before insertion. Face Recognition (Recognition Thread): The Recognition Thread runs recognition asynchronously to avoid blocking the GUI. For each detected face region, DeepFace extracts a FaceNet embedding, which is compared against all stored embeddings using the get_most_similar() database method. The user with the highest cosine similarity above the threshold is identified.

C. Sample Code Excerpt

The following excerpt shows the core DNN detection routine:

```
def  detect faces dnn(self, frame):
    h, w = frame.shape[:2]
    blob = cv2.dnn.blobFromImage(
        cv2.resize(frame, (300, 300)), 1.0,
        (300, 300), (104.0, 177.0, 123.0))
    self.net.setInput(blob)
    detections = self.net.forward()
    faces = []
    for i in range(detections.shape[2]):
        confidence = detections[0, 0, i, 2]
        if confidence > DNN CONFIDENCE THRESHOLD:
            box = detections[0, 0, i, 3:7] * np.array([w, h, w, h])
            (x1, y1, x2, y2) = box.astype("int")
            faces.append((x1, y1, x2-x1, y2-y1))
    return faces
```

VIII. TESTING

A.Types of Testing Unit Testing:

Each module (face detection, face recognition, attendance marking) was tested independently to verify individual functionality. Integration Testing: Module interactions were validated, including the Camera → Detection → Recognition → Attendance → Database pipeline. System Testing: The complete end-to-end workflow was tested to ensure all components function correctly as a unified system. Functional Testing: System features were verified against requirements, including user registration, face recognition accuracy, and single-entry-per-day enforcement. Performance Testing: Detection speed and recognition response time were evaluated under real-time conditions. Accuracy Testing: Recognition correctness was measured with both registered users and unknown individuals.

B. Module Test Results

Module	Objective	Test Case	Result
Registration	Verify face capture and storage	Multi-image capture; DB insertion	Pass
Face Detection	Accurate real-time detection	Single/multiple faces; varying light	Pass
Face Recognition	Correct user identification	Registered and unknown users	Pass
Attendance	Proper attendance marking	Mark attendance; prevent duplicates	Pass
Database	Data storage and retrieval	Insert, retrieve, update records	Pass
GUI	Smooth user interaction	Camera feed; registration; display	Pass
Camera	Video capture functionality	Start/stop; continuous frames	Pass

IX. DEPLOYMENT

The Vision Mark system is deployed using the following steps: (1) Install Python 3.8 or above. (2) Install all required libraries using pip: opencv-python, deepface, numpy, pandas, scipy, matplotlib, and PySide6. (3) Organize the project folder with main.py, dataset/, database/, and modules/ directories. (4) Run the application with python main.py. The PySide6 GUI window launches automatically with the camera initialized.

User Registration: Enter user details in the registration tab and capture multiple face images. The system generates and stores facial embeddings automatically. Attendance Marking: The user stands in front of the camera. The system detects the face, recognizes the identity, and marks attendance with a timestamp. Unknown faces are not marked and duplicates within the same day are rejected.

X. INTEGRATION AND EXPERIMENTAL RESULTS

A. Integration of Modules

All modules were successfully integrated to work as a unified system. The Camera module provides real-time video input to the Face Detection module. Detected face regions are passed to the Face Recognition module, which queries the Database module for matching embeddings. On successful match, the Attendance module records the entry, and the GUI module reflects the result immediately. Smooth data flow and coordination between all components was confirmed during integration testing.

B. Experimental Setup Hardware:

Laptop with HD webcam; Processor: Intel i5; RAM: 8 GB. Software: Python 3.10, OpenCV 4.8, DeepFace 0.0.93, PySide6, SQLite. Testing was conducted with multiple users under varying conditions including different lighting intensities, camera distances (0.5–2 metres), and face orientations (frontal and slight angles).

C. Performance Evaluation

TABLE I COMPARISON OF ATTENDANCE SYSTEMS

Parameter	Manual System	RFID System	Biometric System	Vision Mark (Proposed)
Accuracy	Low	Medium	High	Very High
Proxy Attendance	High Risk	High Risk	Low Risk	Very Low Risk
Time Efficiency	Slow	Moderate	Fast	Very Fast
User Interaction	Manual	Card-based	Touch-based	Contactless
Maintenance	Low	Moderate	High	Low
Hygiene	Safe	Safe	Poor (contact)	Very Safe
Scalability	Poor	Moderate	Moderate	High
Automation Level	None	Partial	High	Fully Automated

The results show that Vision Mark achieves very high recognition accuracy (approximately 97–99% under normal conditions), fully automated contactless operation, and very fast response times compared to all existing approaches.

XI. CONCLUSION

Vision Mark successfully demonstrates the application of computer vision and deep learning in automating attendance management. By replacing traditional manual, RFID, and biometric methods, the system provides a more efficient, accurate, and reliable solution. The integration of DNN-based face detection and FaceNet-based face recognition enables real-time identification without physical interaction. Date-based constraints and a configurable similarity threshold effectively prevent proxy attendance and duplicate entries. Experimental results confirm that Vision Mark performs well under real-time conditions with minimal delay and high recognition accuracy. Its modular architecture, user-friendly PySide6 interface, and contactless operation make it suitable for modern academic institutions and organizations. While sensitivity to poor lighting conditions and face occlusions are current limitations, these can be addressed through future enhancements such as liveness detection and improved recognition models.

XII. FUTURE ENHANCEMENTS

Liveness Detection (Anti-Spoofing): Implement techniques to prevent misuse using photographs or pre-recorded videos, ensuring attendance is marked only for live individuals.

Cloud Integration: Enable centralized data management and real-time synchronization across multiple locations using cloud-based storage.

Mobile Application / Web Portal: Develop a mobile app or web interface for administrators to monitor attendance and generate reports remotely.

Masked Face Recognition: Enhance the system to accurately recognize faces even when users are wearing masks, improving real-world usability.

Edge AI Implementation: Deploy the system on edge devices for faster on-device processing and reduced dependency on high-end hardware or internet connectivity. Advanced Analytics Dashboard: Provide graphical attendance reports including trends, absentee analysis, and performance tracking for institutions.

ACKNOWLEDGMENT

The authors express sincere gratitude to Dr. K. Prem Kumar, Professor and Head of the Department of Artificial Intelligence & Machine Learning, ACE Engineering College, for his invaluable guidance and continuous support throughout this project. The authors also thank the management of ACE Engineering College and all faculty members of the Department of AI & ML for their encouragement and timely assistance.

REFERENCES

[1] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering," in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2015, pp. 815–823.

[2] P. Viola and M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features," in Proc. IEEE CVPR, 2001, pp. 511–518.

[3] S. Z. Li and A. K. Jain, Handbook of Face Recognition, 2nd ed. New York, USA: Springer, 2011.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in Proc. NIPS, 2012.

[5] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. Cambridge, MA, USA: MIT Press, 2016.

[6] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep Face Recognition," in Proc. British Machine Vision Conf. (BMVC), 2015.

[7] D. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the Gap to Human-Level Performance in Face Verification," in Proc. IEEE CVPR, 2014.

[8] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled Faces in the Wild: A Database for Studying Face Recognition," Univ. Massachusetts, Tech. Rep., 2007.

[9] R. Brunelli and T. Poggio, "Face Recognition: Features versus Templates," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 15, no. 10, pp. 1042–1052, 1993.

[10] M. Turk and A. Pentland, "Eigenfaces for Recognition," Journal of Cognitive Neuroscience, vol. 3, no. 1, pp.71–86,1991.

[11] S. Shireesha, G. Anitha and P. Harshavardhan, "Automated Attendance System using Face Recognition," International Journal of Engineering Research & Technology (IJERT),vol.11,no.5,pp.1–5,2022.

[12] M. R. Kumar, K. S. Reddy and P. Ramesh, "CNN-Based Face Recognition System for Automated Attendance," International Journal of Advanced Research in Computer Science, vol. 11, no. 3, pp. 45–50, 2020.