IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

AI Powered Thief Detection and Police Alert System

[1]Rohan Avhad, [2]Himanshu Kale, [3]Abhishek Bhange, [4]Anjali Ghorpade, [5]Prof. M. D. Patil

[1 2 3 4]UG Student, Department of Electronics and Telecommunication Sinhgad Institute of Technology and Science, Pune

[5]Assistant Professor, Department of Electronics and Telecommunication Sinhgad Institute of Technology and Science, Pune

Abstract: The growing demand for enhanced security solutions has led to the development of AI-powered theft detection systems that leverage real-time facial recognition, machine learning technologies, and advanced analytics. This review paper focuses on a system designed to automatically detect and identify criminals from live video feeds using facial recognition technology. By cross- referencing detected faces with a database of known offenders, the system pro- vides immediate alerts to law enforcement via email and SMS, facilitating faster response times and more effective resource allocation. This paper presents a detailed examination of the current technologies employed, including the integration of OpenCV for image processing, Flask for web deployment, and SQLite for storing records and detection logs. Additionally, it outlines the system objectives, methodology, and future potential in enhancing public safety and reducing false alarms. As AI and computer vision continue to evolve, this system demonstrates promising advancements in crime prevention and the automation of law enforcement processes.

***Keywords*- Facial Recognition, Machine Learning, OpenCV, Flask, SQLite, Public Safety, Crime Prevention**

Introduction

In recent years, advancements in artificial intelligence (AI) and computer vision have dramatically transformed security systems, particularly in the domain of automated theft detection and instant threat notification. Traditional surveillance systems, while effective to some degree, typically rely on manual monitoring by security personnel, which can lead to delayed responses, human error, and inefficient allocation of resources. In contrast, AI-powered theft detection systems overcome these limitations by introducing automated, real-time identification of suspicious activities and individuals through machine learning algorithms and facial recognition technologies. These intelligent systems continuously analyze live video feeds from CCTV cameras, detecting faces and comparing them against a vast database of known offenders or individuals with criminal records. Upon detecting a match, the system promptly sends alerts to law enforcement agencies and security personnel via email and SMS, accompanied by video evidence for rapid verification. This automated process accelerates response times and minimizes the need for manual oversight, thereby enhancing overall security and improving the efficiency of resource deployment in critical situations. Additionally, such systems offer the potential for continuous 24/7 monitoring without fatigue, which is a significant limitation in human-centric surveillance operations. Through a comprehensive literature survey, detailed examination of system objectives, and analysis of various implementation methodologies, this review underscores the potential of AI technologies to significantly enhance public safety and crime prevention efforts. As AI continues to evolve, these systems are expected to become more accurate, reliable, and scalable, making them vital tools in modern security infrastructure. Moreover, integration with cloud platforms and IoT devices can facilitate centralized monitoring and coordinated responses across multiple locations. Additionally, ongoing advancements in AI-driven image processing and automation will likely contribute to a substantial reduction in false alarms, further improving their effectiveness in real-world applications. These innovations not only enhance system responsiveness but also help in building public trust in automated surveillance solutions. The proposed system demonstrates a proactive approach to threat detection, shifting the paradigm from passive surveillance to active, intelligent intervention.

I. Literature Survey

The table below provides a comparative analysis of research papers focused on face detection, recognition, and crime detection systems. It highlights the authors, titles, and publication years, along with the advantages (pros) and limitations (cons) of the proposed methods and technologies in each study. This comparison aims to offer insights into the advancements and challenges in the domain of criminal identification systems using artificial intelligence and related techniques. By systematically analyzing each study, it becomes easier to identify patterns in algorithm performance, real-time feasibility, data privacy concerns, and system scalability. Furthermore, this comparison serves as a foundation for identifying research gaps, drawing conclusions about the effectiveness of various approaches, and guiding the development of more efficient, accurate, and secure AI-based security systems.

Table 2.1 Comparative Analysis of Research

Sr. No.	AUTHOR	TITLE	YEAR	PROS	CONS
[1]	Apurva Pongade, Shruti Karad, Divya Ingale, S. Mahabare	Face Detection and Recognition for Criminal Identification System	2024	Haar Cascade - classifier provides high-precision	Data Security, Scalability Challenges
[2]	Waqas Ali Manj, Zunaira Faraz, Hamza Farooq, Ali Fazal	Automatic Face Recognition of Criminals in Investigation Using Artificial Intelligence	2023	CNN- High Accuracy 99.38%, Wide Application	Lighting Sensitivity, Pose Limitation Less Security
[3]	Parth Virdhe, Anuj Nemanwar, Sairaj Shirole, A. Chouthankar	Theft Detection using Deep Learning	2023	Yolo Automated Crime Detection Accuracy: 82%	Only detects object and suspicious activity
[4]	Arjun Menon, Shivani Singh, Raushan Kumar, Ritvik Sethi	Leveraging Facial Recognition Technology in Criminal Identification	2023	MTCNN - Automation and Speed	Data Security

[1] This study focused on face detection and recognition for criminal identification systems using the Haar Cascade classifier, known for its high precision in facial detection tasks. Despite its accuracy, the system encounters significant challenges related to data security and scalability, which limit its effectiveness in large-scale deployments and raise concerns about protecting sensitive information.

[2] This study examined automatic face recognition of criminals during investigations using artificial intelligence, particularly employing CNNs. The system achieved impressive accuracy, reaching 99.38%, and demonstrated a wide range of applications. However, it was sensitive to changes in lighting and pose, which impacted performance. Additionally, the study highlighted security limitations, making the system vulnerable in certain scenarios.

[3] This study explored deep learning techniques, utilizing the Yolo algorithm for automated crime detection. With an accuracy of 82%, the system showed promise in identifying objects and suspicious activities. However, it was primarily limited to object detection, lacking the ability to identify individuals involved in the crimes, which restricted its utility in more comprehensive theft detection scenarios.

[4] This research effort leveraged facial recognition technology in criminal identification, using the MTCNN algorithm to provide speed and automation in facial recognition processes. While the system improved efficiency, it still faced data security challenges, which are critical when dealing with sensitive criminal records and identification data.

II. BLOCK DIAGRAM

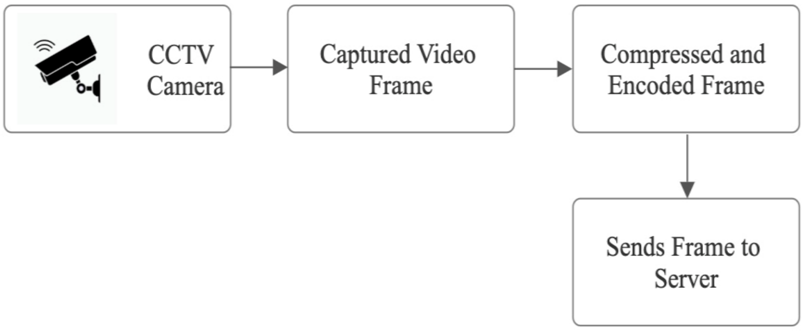

Fig. 3.1 Client Side

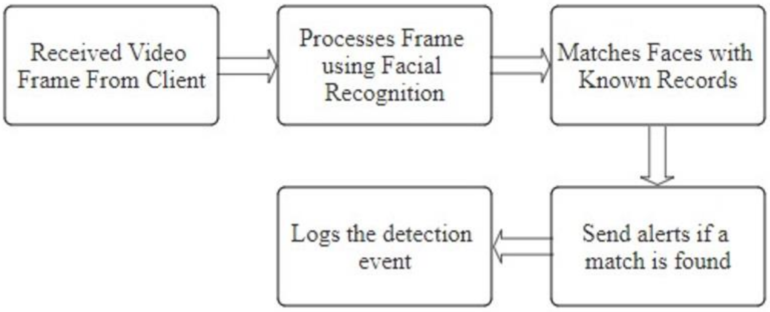
Received Video Frame From Client
Processes Frame using Facial Recognition
Matches Faces with Known Records
Send alerts if a match is found
Logs the detection event

Fig. 3.2 Server Side

III. System workflow

4.1 Captures Video Frames (CCTV): This is the starting point of the surveillance process, where video input is captured from a CCTV camera or a live feed source. The continuous video stream is broken down into individual frames—static images taken at successive time intervals. These frames act as the raw input for further facial analysis. Accurate frame capturing is critical, as it directly affects the reliability and performance of the detection process.

4.2 Sends Frames to the Server: Once the frames are captured, they are encoded (e.g., in JPEG or PNG format) and compressed to reduce their size without significant loss in quality. These compressed frames are then transmitted over the network to a central server. Efficient frame transfer is key for maintaining real-time performance, especially in bandwidth-constrained environments.

4.3 Received Video Frame from Client: The server receives the incoming video frames sent by the client-side device. This data forms the input for the server-side processing unit. Upon receipt, each frame is pre-processed—converted to the required color format (RGB or grayscale), resized, and cleaned—ensuring consistency for accurate facial recognition.

4.4 Processes Frames using Facial Recognition: This stage involves applying facial recognition algorithms to the received frames. Techniques like MTCNN or Haar Cascades detect faces in the image, and models like FaceNet extract facial embeddings numerical vectors that uniquely represent facial features. These embeddings are then passed on for comparison against stored records.

4.5 Matches Faces with Known Records: The extracted face embeddings are compared to existing embeddings stored in the criminal database. Using a similarity metric like Euclidean distance, the system determines whether the face from the frame matches any known offender. A low distance score indicates a high likelihood of a match.

4.6 Logs the Detection Event: If a match is found, the system logs the event details in the SQLite database. This includes the matched individual’s identity, timestamp, video source, and the confidence level of the match. These logs serve as official records and support retrospective investigations, audits, and reporting.

4.7 Sends Alerts if a Match is Found: Upon successful detection, the system automatically sends real-time alerts via SMS (using Twilio) and email (via smtplib). These alerts are directed to designated law enforcement or security personnel, providing them with immediate information and video evidence for rapid response and action.

IV. Software tools & frameworks

The system employs a range of tools for efficient implementation:

5.1 **Python:** Used as the core programming language due to its simplicity, versatility, and rich ecosystem of libraries for machine learning, computer vision, and data processing. It forms the backbone of AI logic and facial recognition implementation.

5.2 **HTML & CSS:** Employed for creating the frontend user interface of the web application. HTML structures the layout, while CSS enhances the visual styling, ensuring a clean and intuitive user experience for monitoring and administration.

5.3 **OpenCV:** A powerful open-source computer vision library used for capturing video frames, detecting faces, and processing images in real-time. It enables seamless integration with camera devices and supports real-time frame manipulation and analysis.

5.4 **Flask:** A lightweight Python web framework that manages the server-side of the application. It handles video frame requests from clients, processes them, and coordinates interactions between the database, facial recognition engine, and alert systems.

5.5 **SQLite:** Chosen as a lightweight, embedded database to store criminal records, face encodings, and detection logs. Its simplicity and zero-configuration setup make it ideal for rapid development and deployment in small to medium-scale systems.

5.6 **Twilio:** Integrated as a messaging API to deliver instant SMS alerts to authorities when a known criminal is detected. It ensures real-time communication and enhances the system's responsiveness in critical scenarios.

5.7 **Visual Studio Code (VS Code):** The primary development environment used for writing, testing, and debugging the codebase. Its support for Python, extensions, and Git integration makes it suitable for managing a multi-component project efficiently.

V. EXPERIMENTATION

6.1 **Database Initialization:** The experimentation begins with setting up the SQLite database, which forms the backbone of data storage. Tables are created to store details of known criminals, such as name, age, city, and facial encodings, along with a separate table for logging detection events. Indexing is applied to the facial encodings for faster search and comparison, while primary keys and data validation rules ensure data integrity and structured storage. This foundational setup ensures seamless interaction between the database and the recognition/notification components.

6.2 **Adding Known Criminal Records:** In this phase, the database is populated with criminal records. Each entry includes personal information and a unique facial encoding generated using a pre-trained model like FaceNet, which converts facial features into numerical vectors. These encodings act as reference points during real-time recognition. Scripts are designed to automate data entry while ensuring checks for duplicates and formatting consistency. The structured storage of encodings supports high-speed facial matching during detection.

6.3 **Server Implementation:** The server, built using Flask, orchestrates the core operations of video analysis and alert generation.

a. **Frame Processing:** Incoming frames from the client are received, resized, and converted (e.g., to RGB) to maintain consistency in recognition.

b. **Facial Recognition Matching:** Algorithms like MTCNN detect faces, and FaceNet extracts facial features, which are compared with existing encodings in the database using Euclidean distance for match prediction.

c. **Notification System:** On a successful match, the server immediately sends alerts via email using the smtplib library and SMS using Twilio's API to ensure quick action.

d. **Event Logging:** All detection results, including timestamps, names, and frame references, are logged in the database, maintaining a robust record for future audits and system evaluation.

6.4 **Client Implementation:** The client application is responsible for real-time video capture using OpenCV from sources such as CCTV cameras. Captured frames are compressed (e.g., JPEG) to reduce size and sent asynchronously to the server through HTTP requests. This efficient transmission ensures minimal latency, allowing real-time analysis. The client acts as the system's input unit, continuously streaming video to the server. It is scalable to support multiple feeds or can be deployed on edge devices for distributed environments.

VI. RESULTS & DISCUSSIONS

7.1 **Results:** To assess the effectiveness of our AI-powered theft detection and notification system, we conducted controlled tests in three different lighting environments: daylight, room light, and low light. The main goal was to evaluate how lighting conditions affect the accuracy and reliability of the facial recognition module used to identify known criminals from live video feeds. Each test involved presenting the system with a set of faces (both known and unknown) under each lighting condition, and measuring its ability to correctly recognize and match the known individuals. We recorded recognition accuracy (i.e., correct matches) and false positive rate (incorrect identification).

Table 7.1 Performance Comparison Across Lighting Conditions

Lighting Condition	Accuracy	False Positives	Observations
Daylight	94%	3%	Clear and natural lighting resulted in high facial clarity, enabling accurate detection with minimal errors.
Room Light	88%	6%	Artificial indoor lighting caused minor shadows and glare, slightly lowering recognition performance.
Low Light	72%	12%	Poor lighting led to less clear face captures, reducing system accuracy and increasing false positives.

7.2 Analysis:

a. **Daylight Conditions:** The system exhibited its best performance under natural daylight. Facial features were captured clearly with minimal distortion or shadow, allowing the model to accurately compare and match against the database. Most detection events resulted in correct identifications, and the notifications were promptly sent.

b. **Room Light Conditions:** The performance saw a slight drop due to varied lighting intensities, especially when the subject was not directly facing the light source. This occasionally led to partial face detections or shadow-covered features, which impacted the facial encoding quality.

c. **Low-Light Conditions:** Recognition accuracy significantly dropped in dim environments. The lack of adequate brightness affected the visibility of key facial features. Although the model still attempted recognition, the chances of mismatch and false alerts increased. In several cases, the system failed to detect a face altogether or misidentified individuals due to noise in the frames.

The results confirm that lighting is a critical factor in facial recognition accuracy. While the system performs excellently under optimal conditions like daylight, its performance degrades in poorly lit environments. These insights help identify areas for enhancement, such as integrating low-light enhancement filters, infrared cameras, or adaptive brightness correction in future development phases to improve system reliability in all scenarios.

VII. Conclusion

This AI-powered theft detection and police notification system successfully fulfills its primary objectives of real-time criminal identification, efficient record maintenance, and instant alert delivery to law enforcement agencies. The system proves to be effective in controlled environments, offering high facial recognition accuracy, secure and structured database operations, and prompt multi-channel alerts via email and SMS when known criminals are identified in live video feeds.

Despite these strengths, the system does encounter limitations most notably, in scenarios where individuals wear masks or when video is captured in low-light environments. These challenges can affect facial feature visibility, leading to reduced detection accuracy and an increased rate of false positives or missed identifications.

To overcome these limitations, future work will focus on integrating more advanced deep learning techniques, including masked face recognition models and low-light image enhancement algorithms. Additionally, incorporating infrared-enabled cameras, edge computing for local processing, and adaptive lighting adjustments can further improve system performance in diverse real-world conditions.

Overall, the project demonstrates the transformative potential of AI and computer vision in modernizing surveillance systems. It lays a solid foundation for the development of more intelligent, automated, and reliable crime prevention solutions. With continued advancements, this system can contribute meaningfully to enhancing public safety, reducing manual monitoring, and enabling faster law enforcement response in critical situations.

VIII. Future Scope

To enhance the robustness and adaptability of the AI-powered theft detection and notification system in real-world scenarios, several promising future improvements can be implemented. A primary enhancement involves integrating advanced deep learning models capable of handling complex visual conditions, such as low-light environments and facial obstructions caused by masks which are increasingly common in modern surveillance challenges. Implementing low-light image enhancement techniques, thermal imaging support, and mask-resistant facial recognition algorithms can significantly improve recognition accuracy under adverse conditions.

Another critical development area is the inclusion of motion detection and behavioral analysis. These techniques can enable the system to identify suspicious activities such as loitering or erratic movements in restricted areas even before clear facial identification occurs. Such proactive features would allow for preventive intervention and improve the system's capability to detect threats early.

Scalability is essential for deploying this system across larger and more diverse environments, such as shopping malls, railway stations, stadiums, and smart cities. Cloud integration with platforms like Google Cloud, Microsoft Azure, or AWS would enable centralized data management, real-time analytics, and remote monitoring across multiple camera sources. Cloud services would also improve system reliability, data storage, and performance, while supporting distributed AI model deployment for faster, edge-based recognition.

To enhance usability and real-time access, a mobile application interface can be developed, allowing security personnel and law enforcement to receive alerts, review detection logs, and communicate instantly through smartphones. Additional innovations such as voice-based notifications, smart speaker integration (e.g., Alexa, Google Assistant), and multi-language support would improve accessibility and responsiveness across different regions and users.

Furthermore, integration with national criminal databases or missing persons registries would expand the system's impact and societal utility, allowing it to function not only as a theft prevention tool but also as a critical asset in locating fugitives or vulnerable individuals.

References

[1] Apurva Pongade, Shruti Karad, Divya Ingale, Shravani Mahabare "Face Detection And Recognition For Criminal Identification System" IJCRT January 2024.

[2] Waqas Ali Manj, Zunaira Faraz, Hamza Farooq, Muhammad Ali Fazal "Automatic Face Recognition of Criminals in Investigation Using Artificial Intelligence" Journal of Xidan University 2023.

[3] Parth Virdhe, Anuj Nemanwar, Sairaj Shirole, Aditya Chouthankar "Theft Detection using Deep Learning" 2023.

[4] Arjun Menon, Shivani Singh, Raushan Kumar, Ritvik Sethi, Abha Kiran Rajpoot "Leveraging Facial Recognition Technology in Criminal Identification" 2023.

[5] W. Farhat, H. Faiedh, C. Souani, and K. Besbes, "Embedded system for road sign detection using MicroBlaze," in 2015 IEEE 12th International Multi-Conference on Systems, Signals Devices (SSD15), 2015, pp. 1–5.

[6] T. Wang, C. Chang, and Y. Wu, "Template-based people detection using a single downward-viewing fisheye camera," in 2017 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS), 2017, pp. 719–723.

[7] S. Issaoui, R. Ejbeli, T. Frikha, and M. Abid, "Embedded approach for edge recognition: Case study: Vehicle registration plate recognition," in 2016 13th International Multi-Conference on Systems, Signals Devices (SSD), 2016, pp. 336–341.

[8] C. Maaoui, F. Abdat, and A. Pruski, "Wavelet transform based facial feature points detection," in 2017 14th International Multi-Conference on Systems, Signals Devices (SSD), 2017, pp. 5–10.

[9] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition,2001, pp. 511–518.

[10] S. Ma and L. Bai, "A face detection algorithm based on Adaboost and new Haar- Like feature," in 2016 7th IEEE International Conference on Software Engineering and Service Science (ICSESS), 2016, pp.651–654.

[11] M. A. S. Mollah, M. A. A. Akash, M. Ahmed, and M. A. H. Akhand, “Improvement of haar feature based face detection incorporating human skin color analysis,” in 2016 International Conference on Medical Engineering, Health Informatics and Technology (MediTec), 2016, pp. 1–6.

[12] L. Cuimei, Q. Zhiliang, J. Nan, and W. Jianhua, “Human face detection algorithm via Haar cascade classifier combined with three additional classifiers,” in 2017 13th IEEE International Conference on Electronic Measurement Instruments (ICEMI), 2017, pp. 483–487.

[13] N. Dalal and B. Triggs, “Histograms of Oriented Gradients for Human Detection” in 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05), 2005, pp. 886–893.

[14] A. Dehghani, D. Moloney, and X. Xu, “Face detection speed improvement using bitmap-based Histogram of Oriented gradien,” in 2017 International Conference on Systems, Signals and Image Processing (IWSSIP), 2017, pp. 1–5.

[15] Y. Kim, H. Shahdoost, S. Jadhav, and C. S. Gloster, “Improving the Accuracy of Arctan for Face Detection,” in 2017 IEEE 25th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM), 2017, pp. 202–202.

[16] D. Huang, C. Chen, T. Chen, J. Wu, and C. Ko, “Real-Time Face Detection Using a Moving Camera,” in 2018 32nd International Conference on Advanced Information Networking and Applications Workshops (WAINA), 2018, pp. 609–614.