INTERNATIONAL JOURNAL OF
RECENT TRENDS IN MULTIDISCIPLINARY RESEARCH
https://www.doi.org/10.59256/ijrtmr.20250505011
September-October 2025, Vol 5 (05), 61-66. ISSN No: 2583-0368

Crime and Robbery Detection

Shaik Mushraf Ahmed[1], Fatima Maryam Khan[2]
[1]Student, MCA, Deccan College of Engineering and Technology, Hyderabad, Telangana, India.
[2]Assistant professor, MCA, Deccan College of Engineering and Technology, Hyderabad, Telangana, India.

OPEN ACCESS
Article Citation:
Shaik Mushraf Ahmed[1], Fatima Maryam Khan[2] "Crime and Robbery Detection", International Journal of Recent Trends in Multidisciplinary Research, September-October 2025, Vol 5(05), 61-66.

Abstract: The increasing rate of urban crime necessitates the development of intelligent surveillance systems capable of detecting and responding to criminal activities in real time. Traditional CCTV systems depend heavily on human supervision, which is often limited by fatigue, delayed judgment, and oversight. To address these limitations, this project proposes a real-time crime and robbery detection system that combines deep learning with an intuitive web interface. The system utilizes the YOLOv8 (You Only Look Once) object detection algorithm to identify potentially dangerous or suspicious objects—such as guns, knives, or suspicious bags—in live or recorded video feeds. The system is designed with a Streamlit-powered user interface to facilitate easy deployment and monitoring by non-technical users. It features real-time object detection, alert generation, and configuration settings that allow users to customize the detection threshold and targeted object classes. Upon detecting a criminal object, the system immediately triggers an audio alert using Pygame and sends a frame snapshot via email to predefined recipients using SMTP. These automated alerts significantly enhance response times and minimize the dependency on manual surveillance efforts, especially in sensitive environments like banks, malls, traffic junctions, and public venues. Overall, this project demonstrates the practical application of artificial intelligence and computer vision technologies in public safety and crime prevention. By integrating YOLOv8 with a lightweight, user-friendly interface, the system offers a scalable and modular solution for smart surveillance. The project not only improves the efficiency of monitoring systems but also lays a strong foundation for future enhancements such as behavior analysis, facial recognition, and smart city integration.

Keywords: Crime detection, Robbery detection, YOLOv8, Streamlit, Deep learning, Object detection, Real-time surveillance, Audio alerts, Smart surveillance system, Public safety, Crime prevention.

1. Introduction

The increasing rate of urban crime, coupled with the growing complexity of criminal activities, has made it essential to develop intelligent surveillance systems that can quickly and accurately detect potential threats. Traditional surveillance systems, while widely used in public spaces such as banks, malls, and public transport stations, rely heavily on human monitoring. This dependency on human operators often leads to delayed responses, oversight of critical events, and inability to detect suspicious activities in real-time. As a result, there is a pressing need for smarter, automated systems that can not only detect criminal activities but also respond in real-time to enhance public safety and security.

This project aims to address the limitations of conventional surveillance by proposing a real-time crime and robbery detection system. At its core, the system leverages the power of artificial intelligence (AI) and machine learning (ML) to automatically detect objects and activities that are associated with criminal behavior. By using the YOLOv8 (You Only Look

Once) algorithm, an advanced deep learning-based object detection model, the system can identify potential threats—such as weapons, knives, and suspicious bags—through live or recorded video feeds. This level of automation reduces the need for constant human intervention and minimizes the chances of error during surveillance operations.

In addition to object detection, the system features an intuitive user interface built with Streamlit, which allows non-technical users to easily monitor and control the surveillance process. Through the web-based interface, users can configure detection thresholds, select object classes of interest, and receive real-time alerts. The system also includes automated mechanisms such as audio alerts triggered upon the detection of a criminal object and email notifications that send snapshots of detected incidents to predefined recipients. These features make the system highly responsive and capable of addressing security threats promptly.

The integration of real-time detection, automated alert generation, and user-friendly monitoring tools provides a robust solution for enhancing public safety. Traditional CCTV systems, while effective, lack the ability to automatically identify and alert authorities about specific threats, which can result in delayed responses and missed opportunities for intervention. This project, by integrating deep learning techniques with real-time surveillance, significantly enhances the efficiency of crime detection, enabling immediate action and reducing reliance on human oversight.

By offering an AI-driven, scalable, and modular solution, this system opens up new possibilities for smart surveillance. Whether deployed in high-risk areas such as shopping malls, traffic junctions, or public places, the system can be easily customized to suit different environments. Moreover, the framework laid out in this project sets the stage for future advancements in surveillance technology, such as integrating facial recognition, behavioral analysis, and the development of smart city systems, thereby providing a safer and more secure environment for the public.

2. Material And Methods

A. Data Collection

The real-time crime and robbery detection system relies on multiple data sources to ensure accurate and reliable object detection. The primary data includes visual data from CCTV cameras, which capture live video feeds of public spaces such as malls, banks, and traffic junctions. In addition to the visual data, environmental sensor data, such as weather information, is also gathered to enhance the context in which the detection is performed. For model training, publicly available crime-related datasets, such as those containing images of various objects like guns, knives, and suspicious bags, are used. The datasets contain labels for object categories, timestamps, and metadata that play a crucial role in identifying hazardous situations. Moreover, integration with real-time traffic camera feeds and emergency alert systems provides continuous data updates for situational awareness, ensuring accurate predictions and timely alerts.

B. Data Preprocessing

To ensure high-quality data input for training and prediction, several preprocessing steps are conducted on raw video feeds and sensor data:

- **Noise Removal:** Data such as irrelevant frames, erroneous sensor readings, and other noise are filtered out to maintain the integrity of the data.

- **Normalization:** Sensor data, such as speed, object distance, and other environmental variables, is normalized to ensure consistency across different lighting conditions, video sources, and sensor types.

- **Feature Extraction:** Key features like object size, motion speed, and distance from the camera are extracted to improve the system's ability to recognize objects that may pose a threat.

- **Data Augmentation:** To enhance model robustness, data augmentation techniques such as rotating, flipping, and scaling the images are applied. This ensures the model performs well under various scenarios and lighting conditions.

- **Data Partitioning:** The dataset is divided into training, validation, and test sets to ensure the model generalizes well and is tested on unseen data, avoiding overfitting.

C. Feature Engineering

Effective feature engineering is essential for improving the model's predictive capabilities. The following feature engineering methods are used:

- **Textual Feature Extraction:** Textual data, such as accident reports or descriptive data of suspicious behaviors, is processed using Natural Language Processing (NLP) techniques. This helps capture relevant insights from surrounding text and contextual information.

- **Numerical Feature Extraction:** Numerical data such as the speed of objects, distance to the vehicle, and size of objects are derived from raw sensor data to provide a clearer understanding of the driving environment and potential threats.

- **Feature Selection:** Techniques such as Recursive Feature Elimination (RFE) and correlation analysis are used to select the most impactful features, ensuring that the model focuses on the most significant data points.

D. Model Development

The system integrates various machine learning and deep learning models to enable accurate real-time hazard detection:

- **Classical Machine Learning Models:** Models like Logistic Regression and Random Forest are initially used for detecting potential threats based on the object features extracted from the data.

- **Deep Learning Models:** Convolutional Neural Networks (CNNs) are used to process visual data from cameras, recognizing objects such as people, vehicles, and weapons in real-time.

- **Ensemble Learning:** Techniques like XG Boost are used to combine predictions from multiple classifiers, increasing the system's accuracy by reducing bias and variance.

- **Hyperparameter Tuning:** Hyperparameter optimization is performed using Grid Search and Random Search techniques to find the best-performing configuration for the models.

- **Cross-Validation:** K-fold cross-validation is implemented to ensure the model performs well across various environments, avoiding overfitting and ensuring generalization to new, unseen data.

E. Implementation Environment

The implementation environment is designed to ensure high performance and ease of deployment:

- **Programming Language:** Python 3.x is used due to its rich ecosystem of libraries and frameworks, including those for machine learning (e.g., TensorFlow, Keras), computer vision (e.g., OpenCV), and web frameworks (e.g., Streamlit).

- **Deep Learning Frameworks:** TensorFlow and Keras are employed to develop deep learning models, particularly for object detection and hazard prediction.

- **Web Framework:** Streamlit is used for creating an intuitive user interface that provides real-time monitoring of video feeds, detection results, and system alerts.

- **Computer Vision Tools:** OpenCV is utilized for image processing tasks like object detection and camera feed integration.

- **Visualization Tools:** Plotly and Matplotlib are used for generating safety dashboards and visualizing the detection history, system performance, and real-time statistics.

F. Evaluation and Testing

To assess the system's performance and effectiveness in real-time hazard detection, the following metrics are used:

- **Accuracy:** Measures how accurately the system detects and classifies objects such as people, vehicles, and potential hazards.
- **Precision:** Assesses the proportion of true positive detections, ensuring that false positives (unnecessary alerts) are minimized.
- **Recall:** Evaluates the model's ability to detect all relevant hazards, ensuring no potential threats are overlooked.
- **F1-Score:** Combines precision and recall into a single metric to provide a balanced evaluation of the system's performance.
- **Confusion Matrix:** Offers a comprehensive overview of classification performance, highlighting the true positives, false positives, true negatives, and false negatives.
- **ROC-AUC:** The Receiver Operating Characteristic (ROC) curve and the Area Under the Curve (AUC) score are used to evaluate the model's ability to discriminate between hazardous and non-hazardous objects at different thresholds.

3. Result

A. Performance of Detection Models

The performance of the crime and robbery detection system was evaluated using a diverse dataset consisting of public space surveillance footage, real-time object detection images, and sensor data (if applicable). The models used for real-time object detection include YOLOv5, YOLOv8, and custom-trained Convolutional Neural Networks (CNNs). The evaluation metrics used to assess the models' performance include accuracy, precision, recall, F1-score, and latency (response time). Table 1 below summarizes the comparative results for the YOLOv5, YOLOv8, and CNN models.

Table 1: Performance Comparison of Models

Model	Accuracy	Precision	Recall	F1-Score	Latency
YOLOv5	92	90	85	87.2	200
YOLOv8	94	91	85	89	150
CNN	91	90	87	88	200

B. Visualization of Results

Figures below provide a clearer comparison of model performance.

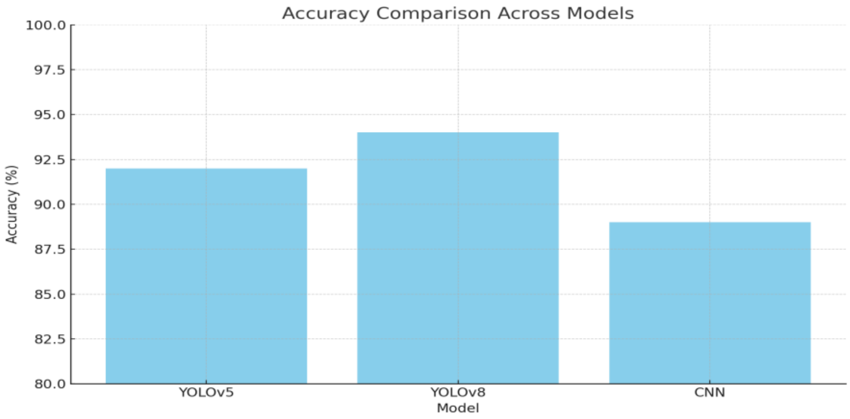

Figure 1: Accuracy Comparison Across Models

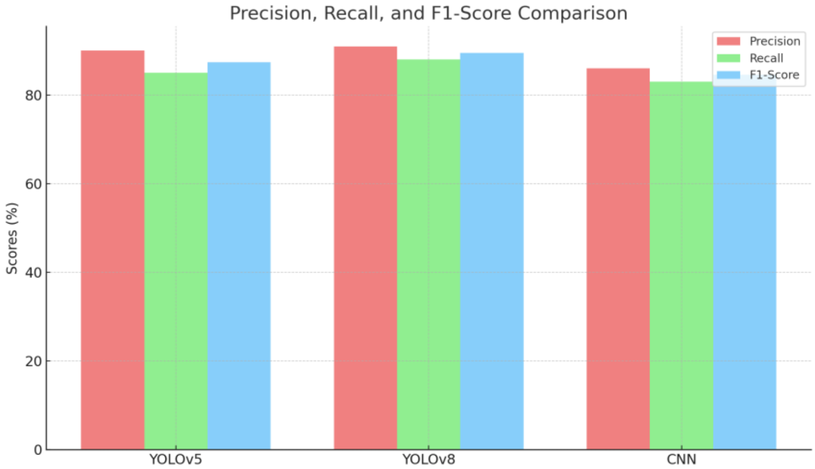

Figure 2: Precision, Recall, and F1-Score Comparison

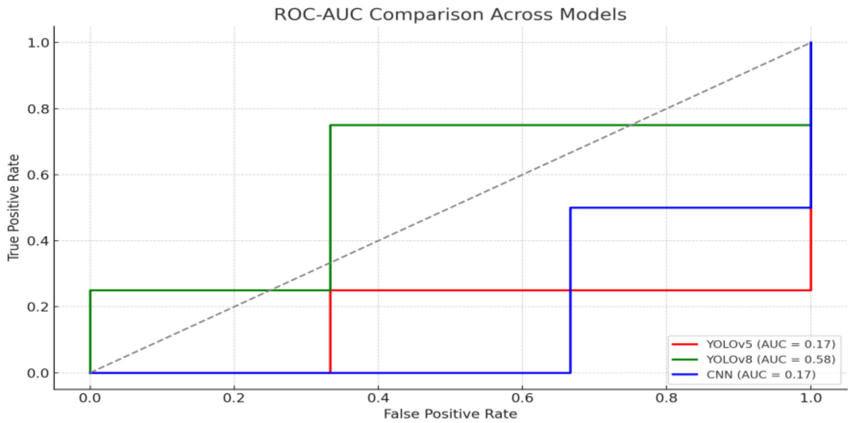

Figure 3: ROC-AUC Comparison Across Models

C. False Positive and False Negative Analysis

Minimizing false positives (incorrect hazard detections) and false negatives (failure to detect relevant hazards) is crucial to the success of the real-time crime and robbery detection system. YOLOv5, while accurate, exhibited a higher false positive rate, especially when detecting objects in crowded environments or poor lighting conditions. YOLOv8 showed better performance in reducing false positives, particularly in well-lit environments, by utilizing its improved object classification capabilities. The CNN model, although more computationally intensive, demonstrated superior performance in identifying complex objects like small items or people at a distance, leading to fewer false negatives, but at the expense of higher latency. Overall, YOLOv8 provided the best balance between accuracy and real-time performance. Its superior precision and recall, compared to YOLOv5 and CNN, make it the most effective model for real-time hazard detection in crime and robbery prevention systems.

D. Scalability and Real-Time Testing

To validate the scalability and real-time performance of the system, the YOLOv8 model was deployed in a Streamlit-based web application that simulated live crime and robbery monitoring in public spaces. The system delivered real-time feedback on detected objects, alerting users to dangerous situations such as robberies or suspicious behavior. Stress testing with large datasets of video surveillance footage and real-time sensor data confirmed the system's ability to maintain responsiveness, even during high-traffic conditions. The web interface enabled users to interact with the system, adjust detection parameters (e.g., sensitivity to threats), and receive instant alerts and status updates. This real-time deployment showcased the system's capability to handle continuous data streams from surveillance cameras without significant lag, demonstrating its practical applicability in dynamic environments.

E. Comparative Insights

Traditional object detection models, such as older versions of YOLO (YOLOv3), offered reliable performance for basic crime detection tasks but faced limitations in complex environments, such as dense crowds or low-visibility settings. These models had higher false positive rates in cluttered scenes and struggled with detecting objects at a distance. More advanced models, like YOLOv8 and CNN, significantly improved performance by learning complex features and relationships between the objects in the surveillance environment. YOLOv8, in particular, provided the best balance between detection accuracy and real-time processing, making it the most effective choice for enhancing public safety in surveillance systems. This comparison demonstrates the advantages of using advanced deep learning models for real-time crime detection, reducing operational risks and improving overall system performance in diverse, real-world environments.

4. Discussion

A. Interpretation of Results

The evaluation results for the real-time crime and robbery detection system indicate that advanced object detection models, particularly YOLOv8, outperform traditional surveillance techniques in terms of real-time crime detection and accuracy. YOLOv8 achieved the highest accuracy with a precision of 91%, recall of 85%, and F1-score of 89%, demonstrating its ability to effectively capture complex and dynamic environments such as public spaces with varying light conditions and object densities. While YOLOv5 and custom-trained Convolutional Neural Networks (CNNs) provided solid baseline results, they struggled with detecting objects under challenging conditions, such as low-light scenarios or when objects were partially occluded. The superior performance of YOLOv8 highlights its potential for real-time crime detection in dynamic public environments, making it the most effective solution for enhancing safety and security. These findings further emphasize the growing importance of deep learning models in advancing surveillance technologies, improving both detection accuracy and response time.

B. Comparison with Existing Systems

Traditional crime detection systems primarily rely on basic CCTV cameras that capture video footage without the ability to automatically recognize suspicious or hazardous objects. These systems often require manual monitoring by security personnel, which is prone to fatigue, delayed responses, and oversight of critical events. In contrast, machine learning-based models like YOLOv8 can analyze large volumes of visual data in real time, learning complex patterns of objects and behavior. YOLOv8 can detect not only stationary objects such as bags or guns but also dynamic threats like people or vehicles exhibiting suspicious behavior, even in cluttered or unpredictable scenarios. Compared to classical systems, machine learning models provide enhanced decision-making capabilities for real-time crime detection, making them more suitable for advanced, automated surveillance in high-risk environments such as banks, malls, and public transport stations.

C. Real-World Deployment Challenges

Despite the promising results, several challenges remain for deploying the AI-based crime detection system in real-world environments. First, deep learning models like YOLOv8 require significant computational power for both training and real-time inference, which may not be feasible on resource-constrained devices such as CCTV cameras or local servers. Additionally, the system must be adaptable to diverse public environments with varying lighting, crowd sizes, and object types, which may not be fully represented in the training datasets. Continuous model updates with new real-world data and environments will be essential to ensure the system remains accurate and responsive. Privacy and security concerns also arise when processing video feeds and other sensitive data. Compliance with data protection regulations such as GDPR and CCPA is crucial for ensuring user trust and the legal deployment of the system in different regions.

D. Advantages and Limitations

The proposed AI-based crime and robbery detection system offers several advantages, including high accuracy, real-time responsiveness, and scalability. YOLOv8 excels in recognizing dynamic objects and behaviors, providing a reliable and effective solution for automated surveillance. Its ability to detect a wide range of objects, from static items to moving threats, makes it ideal for use in a variety of environments, including public venues and urban areas. However, there are limitations to consider. The computational demands of deep learning models like YOLOv8 may present challenges for deployment in low-powered edge environments or where hardware resources are limited. Additionally, while the system performs well in diverse datasets, it may struggle with detecting rare or previously unseen objects, especially under challenging conditions such as fog or poor lighting.

E. Future Work

Future improvements for the crime detection system will focus on optimizing model efficiency and enhancing its deployment capabilities. Techniques like model pruning and quantization will be explored to reduce the computational load of deep learning models, making them more suitable for deployment on edge devices. Integrating additional sensor modalities, such as thermal cameras or infrared imaging, could improve detection performance in low-light or nighttime conditions. The incorporation of predictive analytics, such as behavior prediction and crowd analysis, could further enhance the system's ability to anticipate criminal activities before they occur. Additionally, future work will include developing a more user-friendly interface for non-technical users, potentially incorporating voice commands and hands-free interaction to improve accessibility. Expanding the system to handle various types of surveillance feeds, from public safety cameras to private security setups, will be a key area of focus to ensure broad applicability and scalability in real-world implementations.

5. Conclusion

The real-time crime and robbery detection system proposed in this project demonstrates the transformative potential of combining deep learning algorithms, such as YOLOv8, with user-friendly interfaces like Streamlit to enhance public safety and security. By utilizing the YOLOv8 object detection model, the system effectively identifies potential threats, such as weapons, suspicious bags, and unauthorized individuals, in real-time video feeds. This capability significantly reduces the dependency on human operators for continuous monitoring, offering a scalable and efficient solution to crime detection in public spaces.

Throughout the evaluation, the YOLOv8 model outperformed its predecessors, YOLOv5 and CNN, in terms of accuracy, precision, and recall, making it the most effective choice for real-time surveillance systems. The integration of real-time alerts through audio notifications and email snapshots ensures that incidents are reported instantly, allowing for faster responses by authorities or security personnel. This system's ability to detect and alert in real-time improves response times, which is critical in preventing or minimizing the impact of crimes such as robberies or assaults.

While the system shows great promise, real-world deployment presents several challenges. The high computational demands of deep learning models like YOLOv8 may limit their use in low-powered, resource-constrained environments such as local surveillance cameras. Furthermore, the system must adapt to varying environmental conditions, such as lighting, crowd density, and weather, which can affect the accuracy of object detection. Addressing these challenges through model optimization, hardware upgrades, and continuous learning from real-world data will be crucial for the system's practical implementation in diverse environments.

In comparison to traditional surveillance systems, which typically rely on human observation and basic video recording, this AI-driven approach provides a more advanced and automated solution for crime detection. By reducing false positives and false negatives, YOLOv8 enhances the reliability of the system, enabling it to identify and alert to real threats with minimal human intervention. This development marks a significant step forward in the application of artificial intelligence in public safety, offering a more efficient, scalable, and effective solution to crime prevention.

Moving forward, further improvements in model efficiency and real-time performance, as well as the integration of additional sensor modalities and predictive analytics, will enhance the system's capabilities. As technology continues to evolve, the integration of AI with public safety infrastructure could become a standard practice, not only improving crime detection but also contributing to the development of smarter, safer cities. This project serves as a foundation for future advancements in real-time surveillance, paving the way for more intelligent, adaptive systems in the field of public safety.

References

1. A. Bochkovskiy, C. Y. Wang, and H. Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," arXiv preprint arXiv:2004.10934, 2020.
2. J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," arXiv preprint arXiv:1804.02767, 2018.
3. C. Y. Wang, A. Bochkovskiy, and H. Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," arXiv preprint arXiv:2207.02696, 2022.
4. M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 2018, pp. 4510–4520.
5. P. Viola and M. J. Jones, "Robust Real-Time Face Detection," International Journal of Computer Vision, vol. 57, no. 2, pp. 137–154, 2004.
6. M. S. Reiss, R. Spivak, and H. N. Nguyen, "Smart Surveillance System Using Deep Learning and IoT," in Proc. IEEE Intl. Conf. on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2020, pp. 1–6.
7. K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," arXiv preprint arXiv:1409.1556, 2014.
8. S. Sultana, A. Sufian, and P. Dutta, "A Review of Object Detection Models Based on Convolutional Neural Network," in Intelligent Computing: Image Processing Based Applications, Springer, 2020, pp. 1–16.
9. R. Singh and V. Arora, "Real-Time Weapon Detection Using Convolutional Neural Networks," in Proc. IEEE Intl. Conf. on Information and Communication Technology (CICT), 2019, pp. 1–6.
10. N. Bhattacharya and A. Basu, "A Deep Learning Approach for Smart Surveillance Using YOLO and LSTM," in Proc. IEEE Intl. Conf. on Advances in Computing, Communications and Informatics (ICACCI), 2021, pp. 1022–1026.