AN AI-POWERED SURVEILLANCE SYSTEM FOR WEAPON AND VIOLENCE DETECTION IN REAL-TIME CCTV STREAMS

SRIDEVI P, SHAFA D, SAFA SULAIHA A, CIME MONIKA

Department Of Computer Science and Engineering, Arunachala College of Engineering for Women.

ABSTRACT

Public safety in crowded environments such as transportation hubs, public gatherings, and critical infrastructures is a significant concern. Traditional surveillance systems rely heavily on manual monitoring, which is prone to human fatigue, delayed response, and missed incidents. This paper proposes an AI-powered surveillance framework that utilizes deep learning techniques for real-time detection of weapons and violent activities in CCTV video streams. The system integrates YOLO (You Only Look Once) for object detection and CNN-LSTM models for action recognition. The proposed framework enables automated monitoring, instant threat detection, and real-time alert generation. Experimental results demonstrate improved efficiency, faster response time, and enhanced security compared to conventional surveillance systems.

Keywords: Artificial Intelligence, Surveillance System, YOLO, CNN-LSTM, Weapon Detection, Violence Detection, Computer Vision

1. Introduction

Surveillance systems have become an integral part of modern security infrastructure, widely deployed in public spaces, airports, railway stations, and commercial establishments. Despite their widespread use, traditional CCTV systems rely on continuous human monitoring, which introduces several limitations such as fatigue, reduced attention span, and delayed threat identification.

With advancements in artificial intelligence and computer vision, it is now possible to automate surveillance processes. AI-based systems can analyse video streams in real time, detect suspicious activities, and generate alerts without human intervention. This paper presents an intelligent surveillance system that combines object detection and action recognition techniques to identify weapons and violent behaviour efficiently.

2. Related Work

Existing surveillance systems primarily depend on manual monitoring and basic motion detection techniques. Object detection models like YOLO have been widely used for identifying objects in images and videos due to their speed and accuracy. Similarly, deep learning architectures such as Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks have shown promising results in action recognition tasks.

However, limited work has been done in integrating both object detection and temporal action recognition for real-time threat detection in surveillance systems. The proposed system addresses this gap by combining YOLO and CNN-LSTM models for enhanced detection capabilities.

3. Proposed AI-Powered Surveillance Framework

3.1 System Architecture

The proposed framework consists of the following core components:

1. **Video Input Module** – Captures live CCTV video streams.
2. **Frame Preprocessing Unit** – Normalizes and prepares frames for analysis.
3. **YOLO-Based Weapon Detection Module** – Detects weapons such as guns and knives.
4. **CNN-LSTM Violence Detection Module** – Identifies violent activities using sequential frame analysis.
5. **Alert and Response System** – Generates real-time alerts and notifications.
6. **Data Storage Module** – Stores detected events, screenshots, and video clips.

3.2 System Workflow

The workflow of the system is as follows:

1. Live video is captured from CCTV cameras.
2. Video frames are pre-processed for analysis.
3. YOLO detects weapons in each frame.
4. CNN-LSTM analyses frame sequences for violent actions.
5. If a threat is detected, alerts are triggered immediately.
6. Evidence such as screenshots and video clips is stored securely.

4. Software Requirements Specification

4.1 Overview

The system is implemented as a web-based application that processes real-time video streams using deep learning models. It provides automated threat detection and monitoring through an interactive interface.

4.2 Functional Requirements

4.2.1 Video Processing

- Live video streams are captured continuously from CCTV cameras.
- Input frames are extracted and prepared for further analysis.
- Preprocessing techniques such as resizing and normalization are applied to improve model performance.

4.2.2 Weapon Detection

- Weapons such as guns and knives are identified using the YOLO model.
- Each frame is analysed to locate and classify potential threats.
- Detection results include bounding boxes and confidence scores for identified objects.

4.2.3 Violence Detection

- Sequential video frames are analysed using the CNN-LSTM model.
- Patterns of movement are examined to recognize violent activities.
- Temporal information is utilized to improve accuracy in action recognition.

4.2.4 Alert System

- Alerts are generated immediately when suspicious activity is detected.
- Notifications are delivered to security personnel in real time.
- Alert messages include relevant details such as time, type of threat, and location.

4.2.5 Evidence Storage

- Screenshots are captured automatically during detected events.
- Video clips corresponding to incidents are recorded and saved.
- Event data is stored securely in a database for future reference.

4.2.6 Web Dashboard

- A web-based interface provides access to live video feeds.
- Users can review previously detected events through the dashboard.
- Monitoring and system control functionalities are integrated into a single platform.

5. Non-Functional Requirements

5.1 Security Requirements

- Confidentiality of surveillance data is maintained through secure storage mechanisms.
- Access to the system is restricted using proper authentication and authorization controls.
- Sensitive information such as video feeds and alerts is protected against unauthorized access.
- Secure communication channels are used for transmitting alerts and stored evidence.

5.2 Performance Requirements

- Real-time detection is achieved with minimal processing delay.
- Continuous video streams are handled efficiently without noticeable lag.
- Detection models are optimized to balance both speed and accuracy.
- System resources are utilized effectively to ensure smooth performance on standard hardware.

5.3 Usability Requirements

- A user-friendly interface is provided for easy monitoring and control.
- Alerts are displayed in a clear and understandable manner.
- Navigation between live feeds and stored event history remains simple and intuitive.
- Minimal training is sufficient for operators to use the system effectively.

5.4 Scalability Requirements

- Multiple CCTV camera inputs can be supported without performance degradation.
- The architecture allows expansion to large-scale surveillance environments.
- New features and AI models can be integrated with minimal changes.
- Deployment flexibility is ensured across cloud and distributed systems.

5.5 Reliability Requirements

- Continuous operation is maintained without unexpected interruptions.
- Errors and failures are handled gracefully to prevent system crashes.
- Consistent detection performance is ensured across repeated executions.
- Backup and recovery mechanisms safeguard stored data.

6. Software Requirements

- Programming Language: Python
- Web Framework: Flask / Django
- Deep Learning Models: YOLO, CNN-LSTM
- Libraries: OpenCV, TensorFlow/Keras, NumPy
- Frontend Technologies: HTML, CSS, JavaScript
- Database: MySQL / MongoDB

7. Hardware Requirements

- Processor: Intel i5 or higher
- RAM: Minimum 8 GB
- Storage: 256 GB or more
- GPU: Optional (recommended for faster deep learning processing)
- Network: Stable internet connection

8. Constraints

- Requires high computational resources for real-time processing

- Performance depends on video quality
- Accuracy may vary based on training data

9. Assumptions

- Cameras provide continuous video feed
- Models are pre-trained and optimized
- Network connectivity is stable

10.Implementation Details

The system is implemented using deep learning frameworks for real-time video analysis. YOLO is used for fast object detection, while CNN-LSTM processes temporal information for action recognition. The backend handles video processing and alert generation, while the frontend provides monitoring and control functionalities.

11.Experimental Evaluation

11.1 Experimental Setup

The system was evaluated using real-time CCTV video streams and sample datasets containing both normal and suspicious activities. YOLO was used for weapon detection, while a CNN-LSTM model was applied for violence recognition. Video frames were processed using OpenCV, and the models were executed in a Python-based environment. Testing was conducted under different conditions such as varying lighting, crowd density, and camera angles to assess system performance.

11.2 Results and Discussion

The system successfully detected weapons and violent activities in real time with minimal delay. YOLO demonstrated fast and accurate object detection, while CNN-LSTM effectively captured temporal patterns for violence recognition. Automatic alerts and evidence capture improved response time and monitoring efficiency. Compared to traditional CCTV systems, the proposed approach provided faster detection and reduced reliance on manual observation.

12. Security Analysis

The proposed system enhances surveillance security by integrating real-time threat detection with deep learning techniques. Confidentiality of data is maintained through secure storage of video streams and event records. Integrity is ensured by preventing unauthorized modification of stored evidence such as screenshots and video clips. Availability is supported through continuous system monitoring and real-time alert generation without significant delays.

The use of YOLO and CNN-LSTM models enables proactive identification of threats, reducing the risk of delayed response. Additionally, access control mechanisms restrict system usage to authorized personnel only. Overall, the system provides a reliable and efficient approach to improving security in surveillance environments.

13. Conclusion and Future Work

The proposed AI-powered surveillance system effectively detects weapons and violent activities in real-time CCTV streams using YOLO and CNN-LSTM models. By automating threat detection, the system reduces dependence on manual monitoring and significantly improves response time and overall security efficiency. The integration of real-time alerts, evidence capture, and data storage ensures a reliable and proactive surveillance solution.

Future enhancements can focus on integrating facial recognition for identity tracking, supporting multi-camera environments for wider coverage, and deploying the system on edge devices for faster processing. Additionally, incorporating advanced deep learning models can further improve detection accuracy and system performance in complex real-world scenarios.

14.References

1. J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified Real-Time Object Detection,” in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, pp. 779–788, doi: 10.1109/CVPR.2016.91.

2. J. Redmon and A. Farhadi, “YOLO9000: Better, Faster, Stronger,” in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2017, pp. 7263–7271, doi: 10.1109/CVPR.2017.690.

3. A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “YOLOv4: Optimal Speed and Accuracy of Object Detection,” *arXiv preprint arXiv:2004.10934*, 2020.

4. 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,” in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, 2023.

5. G. Jocher *et al.*, “Ultralytics YOLOv8: State-of-the-Art Real-Time Object Detection,” GitHub repository, 2023. [Online]. Available: https://github.com/ultralytics/ultralytics

6. M. Sultani, C. Chen, and M. Shah, “Real-World Anomaly Detection in Surveillance Videos,” in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 6479–6488, doi: 10.1109/CVPR.2018.00678.

7. B. Hassner, Y. Itcher, and O. Kliper-Gross, "Violent Flows: Real-Time Detection of Violent Crowd Behavior," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshops*, 2012, pp. 1–6, doi: 10.1109/CVPRW.2012.6239348.

8. A. Sajjad, A. Khan, K. Muhammad, S. Rho, and S. W. Baik, "Multi-Grade Deep Learning for Violence Detection in Smart Surveillance Systems," *IEEE Access*, vol. 7, pp. 124–138, 2019, doi: 10.1109/ACCESS.2018.2885039.

9. R. Verma, A. Singh, and N. Sharma, "Real-Time Weapon Detection Using Deep Learning for Surveillance Applications," in *Proc. Int. Conf. Computing and Communication Systems (ICCCS)*, 2022, pp. 1–6, doi: 10.1109/ICCCS55188.2022.10079776.

10. S. K. Yadav, A. Sharma, and P. Gupta, "YOLOv8-Based Real-Time Weapon Detection and Classification System," in *Proc. Int. Conf. Smart Computing and Artificial Intelligence (ICSCAI)*, 2024, pp. 1–6.