IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, November 2026
Impact Factor: 8.2

Edge Guard AI: An Intelligent Real-Time CCTV Surveillance and Smart Security Analytics Framework Using Local Edge Processing

Lakshmimaaya A[1], Nihas R S[2], Shijin Shaji[3], Sreehari P S[4], Vinitha A V[5], Heera B M[6]
Student, Department of Computer Science and Engineering[1,2,3,4]
Associate Professor and HOD, Department of Computer Science and Engineering[5]
Assistant Professor, Department of Computer Science and Engineering[6]
Thejus Engineering College, Thrissur, Kerala, India

Abstract: *Conventional CCTV systems record continuous video, but identifying important incidents often depends on a person watching feeds or reviewing recordings after an event. In this work, Edge Guard AI, a proposed intelligent CCTV analytics framework, is presented for adding automated, locally executed video analysis to existing camera infrastructure. The system receives live video from CCTV cameras, IP cameras, webcams, or DVR/NVR streams and processes frames on a desktop or laptop computer, reducing the need to send complete video streams to cloud servers and addressing bandwidth, latency, cost, and privacy concerns. YOLO detects persons, weapons, and other predefined security-related objects; MTCNN detects and extracts faces; and FaceNet generates facial embeddings for comparison with registered records. Detected persons and events are classified as familiar, unknown, missing-person record matches, or law-enforcement watchlist record matches, with additional trained models supporting masked-face and suspicious-activity detection. Events are assigned Normal, Warning, or Danger alert levels, and significant events trigger image capture, short event-specific video recording, timestamping, and notification through a connected application, with a searchable event history. Role-based Admin, User, and Police modules manage records, alerts, and evidence. The framework is at the proposal and design stage; the capabilities described are objectives to be implemented and evaluated, not completed test results.*

Keywords: Edge AI, CCTV Surveillance, YOLO, MTCNN, FaceNet, Face Recognition, Event Detection, Real-Time Alerts.

I. INTRODUCTION

Closed-Circuit Television (CCTV) is widely used to observe and record activity in homes, buildings, public spaces, and other monitored locations. A conventional installation connects cameras to a Digital Video Recorder (DVR), Network Video Recorder (NVR), or other local storage, providing footage that can be reviewed when required. However, continuous recording does not by itself explain what is happening in a scene or indicate which events need immediate attention.

Manual monitoring is a major limitation of traditional CCTV. An operator may need to watch several feeds at once or search long recordings after an incident, which is time-consuming and may allow important events to go unnoticed. Continuous recording also consumes substantial storage because routine activity and significant events are stored together. Some systems use cloud-based analytics, sending video to remote servers; bandwidth usage, latency, operating cost, internet dependency, and privacy are concerns associated with this approach.

Convolutional neural networks and YOLO-based detectors have shown effective performance in tasks such as fall detection [1], [2], human activity detection [3], and face detection [4]. However, many existing solutions address a

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, November 2026 Impact Factor: 8.2

single task, such as falls, crowds, violence, or weapons, and may depend on cloud processing. To overcome these limitations, this work proposes Edge Guard AI, which places the main AI inference on a local desktop or laptop computer. Existing cameras supply the video input, while the computer analyzes frames, categorizes persons and security events, captures evidence, and manages event information.

The system combines several related but distinct tasks. Person detection identifies a person-shaped object in a frame, face detection locates a face, face recognition compares a face representation with registered records, and object detection identifies predefined objects such as weapons when the model has been trained for that class. Suspicious-activity detection is a separate video-analysis task that may require an action-recognition or other trained model. The application is organized around Admin, User, Police, Visitor Detection, and Camera/Edge AI Engine functions.

The main contributions of this work are summarized as follows:

- A locally executed surveillance framework that adds AI analytics to existing CCTV, IP camera, webcam, or DVR/NVR infrastructure.
- A modular AI pipeline using YOLO for person and object detection, MTCNN for face detection, and FaceNet for face comparison, with additional trained models for masked-face and suspicious-activity detection.
- A three-level (Normal, Warning, Danger) alert scheme linked to familiar, unknown, missing-person, watchlist, weapon, masked-face, and suspicious-activity events.
- An event-specific evidence workflow that captures images and short video clips with date and time and stores them in a searchable event history.
- A role-based application design for administrators, users, and authorized police personnel, delivering notifications through a connected application.

The remainder of this paper is organized as follows. Section II reviews the relevant literature, while Section III details the proposed system design. Section IV summarizes the work and highlights directions for future research. As the project is at Phase 1, all statements describe proposed functionality and not a completed or validated implementation.

II. LITERATURE SURVEY

Automated video analysis for surveillance has been explored across activity and fall detection, face detection, anomaly and crowd analysis, weapon detection, crime scene analysis, tracking, and edge-based processing. The surveyed works are discussed below in the order of the project literature review.

A real-time fall detection framework combines YOLOv11-Pose keypoint extraction with a compact Transformer network that identifies falls from body movements over time. It reports high detection accuracy and suitability for real-time monitoring, but its performance may be affected by heavy occlusion, poor visibility, and unreliable pose information [1]. HFD-YOLO modifies YOLOv8n with a Sequential Efficient Attention Module and Residual Bottleneck Self-Attention for single-stage fall detection on devices with limited resources, though it focuses mainly on falls and needs further evaluation across environments [2]. LightSTATE performs frame extraction, motion analysis, and filtering at the edge and sends selected frames to a cloud-hosted Vision Language Model, which reduces unnecessary processing but depends on predefined activity states and cloud-based interpretation [3].

A YOLOv9-based approach detects and counts human faces among animal faces and other distracting objects under varied image conditions, but false detections may occur with artificial faces or similar objects, and noise or blur reduce accuracy [4]. A lightweight framework for anomaly detection and prediction combines a convolutional autoencoder with a Dynamic Graph CNN-LSTM and self-supervised learning; it is evaluated mainly on the UCF-Crime dataset, so reliability on unseen anomalies needs further testing [5]. Another framework combines crowd density estimation, using a Self-Modulating CNN, with anomaly localization using a Pixel-Associated Autoencoder, but generalization to unfamiliar crowd patterns remains limited [6].

For weapons and crime scenes, an enhanced YOLO12 model with a custom dataset of knives, pistols, and long guns detects small, partially hidden, and low-contrast weapons, although performance may vary for weapon types not represented in training data [7]. An integrated model for crime scene analysis uses multimodal fusion of RGB frames,

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, November 2026 Impact Factor: 8.2

motion data, and optical flow for person identification across cameras, YOLO-based weapon detection, and spatiotemporal analysis to summarize long recordings, at the cost of higher computational complexity and coordination requirements [8]. A tracking framework combines GPS, indoor positioning, and RFID with YOLOv8 person detection, FaceNet identity recognition, and a CNN-LSTM activity model, where sensor inaccuracies, occlusion, and communication issues may limit performance [9].

A hybrid edge-cloud smart door system processes video locally using YOLOv8, facial recognition, and behavior analysis with a Hidden Markov Model, supports local face recognition for offline operation, and uses cloud services for additional risk assessment, but may suffer false alarms and network delays [10]. An edge-assisted smart city framework combines Edge AI, federated learning, and cross-domain event correlation to reduce communication latency and bandwidth while keeping sensitive video local, though coordinating multiple edge devices adds complexity [11]. A human-exclusive alert approach combines YOLOv8 with OpenCV image processing to distinguish people from animals and moving objects and thereby reduce false CCTV alerts, but requires validation under poor lighting, occlusion, and crowded scenes [12].

An explainable crowd density framework uses bilateral filtering, a Squeeze-and-Excitation DenseNet, the Osprey Optimization Algorithm, and an Attention Bidirectional GRU to classify crowd density, with performance affected by severe occlusion [13]. Violence detection has been evaluated using VGG16, MobileNetV2, YOLOv6, an ensemble, and a Transformer model, where the Transformer performed strongly on the evaluated datasets but complex actions may still be misclassified [14]. A hybrid model, ResViViT50, combines a reduced ResNet-50 with a Video Vision Transformer and motion-aware token selection to recognize suspicious group activities, but is affected by occlusion, lighting, and camera angle [15].

Three observations from the surveyed work shape the proposed design. First, YOLO-family detectors recur across fall, face, weapon, tracking, access-control, and alert-filtering tasks [2], [4], [7], [8], [9], [10], [12], which supports their use for real-time detection of persons and predefined objects. Second, edge-based processing reduces transmitted video, latency, and bandwidth [3], [11], while hybrid edge-cloud designs can introduce network delays [10]. Third, almost every surveyed system reports reduced reliability under occlusion, poor lighting, blur, or unfamiliar conditions, which means that face matches and detections should be treated as predictions requiring verification rather than as proof.

The survey indicates that existing work is often specialized to one task, may depend on cloud processing, and requires further testing in varied real-world conditions. Despite these advances, an integrated, locally executed system combining person and object detection, face comparison, severity-based alerting, and evidence recording for existing CCTV remains to be realized. Motivated by these limitations, this work proposes Edge Guard AI.

III. SYSTEM DESIGN

A. Existing System and Problem Definition

In a conventional CCTV installation, cameras continuously capture video that is stored on a DVR, NVR, or other local storage device. This provides recorded evidence but does not necessarily classify events automatically: a person entering the scene, a weapon appearing, a masked face, or a suspicious activity may be present in the video without being identified as a separate event. If no one is watching the relevant feed at the right time, the incident may be recognized only after the recording is reviewed. The limitations of such manual-monitoring setups and some cloud-dependent approaches are:

- Continuous recording consumes large storage because routine footage is stored together with important incidents.
- Operators may miss events when monitoring multiple feeds, and searching recordings after an incident delays identification.
- Familiar and unfamiliar visitors are not automatically categorized, and missing-person or law-enforcement watchlist matching is not integrated into the recording workflow.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, November 2026
Impact Factor: 8.2

- Weapon and masked-face detection may be unavailable, and notifications may not distinguish normal activity, warning conditions, and high-risk events.
- Cloud-based analysis can require substantial bandwidth, continuous internet connectivity, and the transmission of surveillance footage to remote servers, raising privacy concerns.

The problem addressed is therefore how to transform an existing CCTV/DVR/NVR installation into a locally processed, AI-assisted surveillance system that detects relevant persons and security events, classifies them, captures evidence, and provides alerts according to severity. The system does not assume that face recognition alone can prove identity or criminal status; matches are comparisons with registered records and can be uncertain when faces are small, blurred, occluded, or poorly illuminated, so high-risk matches should be reviewed by authorized users.

B. Objectives

The planned scope of Edge Guard AI is to: (i) use existing CCTV, IP cameras, webcams, or DVR/NVR streams as the video source; (ii) process live frames locally using computer-vision and deep-learning components; (iii) detect persons and predefined objects, including weapons, with YOLO; (iv) detect and extract faces with MTCNN and compare embeddings with FaceNet; (v) classify persons as familiar, unknown, missing-person matches, or watchlist matches; (vi) support masked-face and predefined suspicious-activity detection through trained models; (vii) classify events into Normal, Warning, and Danger levels; (viii) capture image evidence and short event clips; and (ix) store searchable event records and provide notifications through the connected application, including relevant alerts for authorized law-enforcement personnel where enabled. These objectives do not imply that every model has been trained or that notifications and police integrations have been tested.

C. System Architecture

The architecture is organized around a local host computer connected to existing CCTV infrastructure and consists of five layers, as shown in Fig. 1: the input layer (CCTV, IP cameras, webcams, DVR/NVR), the video-processing layer (frame extraction), the AI layer (YOLO, MTCNN, FaceNet, and other trained models), the event-management layer (event categories, alert levels, evidence records), and the application layer (web and mobile interfaces with role-based functions) supported by a MySQL database.

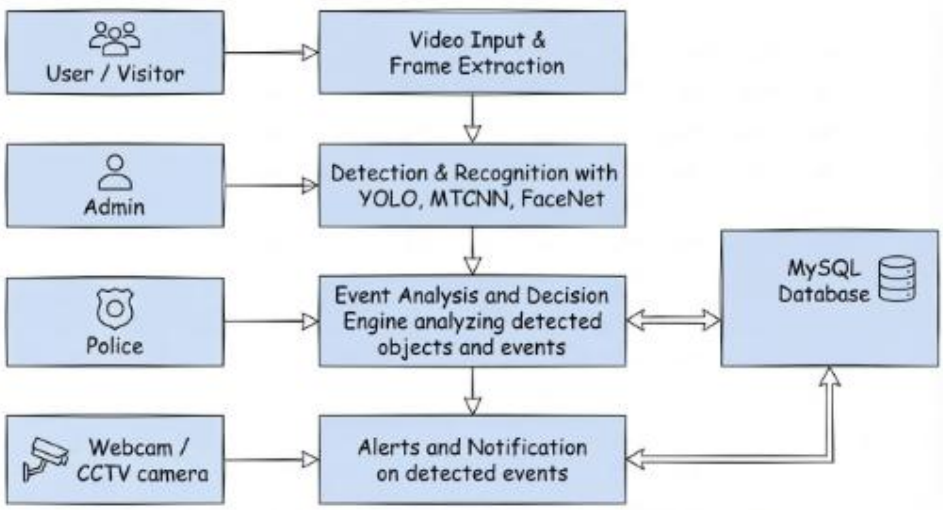


Fig. 1. The architecture of the proposed Edge Guard AI framework.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, November 2026
Impact Factor: 8.2

The main AI inference stays on the desktop or laptop, reducing continuous cloud dependence for core analysis. This does not mean every function works without a network: camera access may use LAN or Wi-Fi, and mobile notifications may require connectivity. Direct integration with police emergency dispatch is not claimed.

D. AI Components

YOLO detects persons and predefined security-related objects, including weapons. **MTCNN** detects and extracts faces from regions where a person is detected. **FaceNet** converts each face crop into an embedding that is compared with registered embeddings of familiar persons, missing-person records, and watchlist records. Masked-face detection, suspicious-activity detection (using video-based deep learning or pose/action recognition), and gender classification of unfamiliar visitors (using a CNN-based model) require separate trained models and data. A person detector alone cannot determine whether a person is fighting or carrying a weapon, so each task is a separate component whose outputs are combined by the event-management stage.

E. Processing Workflow

The processing workflow is shown in Fig. 2. A camera or DVR/NVR stream is connected to the local computer and frames are extracted. YOLO detects persons and objects; if a person is detected, MTCNN locates the face, and FaceNet compares its embedding with registered records. Object outputs are checked for predefined objects such as weapons. The event-management stage associates outputs with user, familiar-person, missing-person, and watchlist records, assigns an event category and alert level, captures evidence, stores the event, and sends a notification.

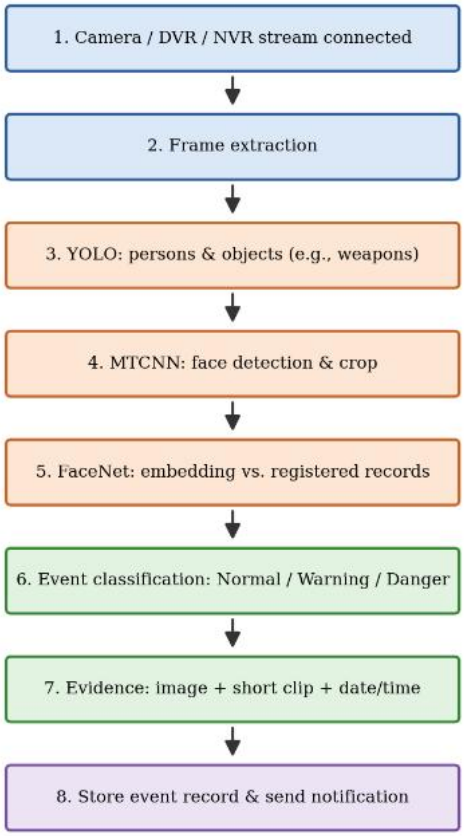


Fig. 2. Processing workflow of the proposed Edge Guard AI framework.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, November 2026
Impact Factor: 8.2

F. Alert Classification

The event-management stage assigns one of three alert levels using planned rules that may be refined during implementation based on evaluation. A **Normal** alert is generated for a registered familiar or authorized person. A **Warning** alert is generated for an unknown or unfamiliar person, or for an unregistered person remaining in a monitored area. A **Danger** alert is generated for a law-enforcement watchlist match, a missing-person match, a weapon, a masked face, or a high-risk suspicious activity. The camera source only supplies video and does not decide the alert level.

G. Evidence Recording and Event History

For significant events, the system captures a still image and records a short event-specific video clip instead of relying on continuous footage review. Each event record stores event type, alert level, date, time, camera location, description, evidence image, video clip, and status, allowing authorized users to review previous detections without manually searching all recordings and potentially reducing storage of routine footage.

H. Application Modules and Data Store

The Admin module manages users, police-station information, criminal and missing-person records, alerts, and complaints. The User module registers familiar persons, shows detected persons and events, provides evidence review and notifications, and supports complaints. The Police module manages criminal and missing-person records and shows related alerts and evidence to authorized personnel. The database retains user profiles, familiar-person records, camera details, visitor logs, surveillance events, criminal and missing-person records, alerts, notifications, complaints, and police-station details.

The Camera/Edge AI Engine is responsible for receiving video streams, extracting frames, running the relevant AI models, classifying events, recording evidence, storing timestamps, creating event logs, and initiating notifications. The Visitor Detection function categorizes detected persons using the registered familiar-person records and visitor logs.

I. Data Flow Diagrams

The Level 0 data flow diagram (Fig. 3) shows the main process, the Intelligent Edge AI Surveillance System, exchanging data with four external entities: User, Admin, Police personnel, and the CCTV/IP camera or DVR/NVR source. The camera supplies the live stream and does not decide the alert level. The system returns event category, alert level, date and time, camera location, evidence, notifications, and searchable history to authorized interfaces.

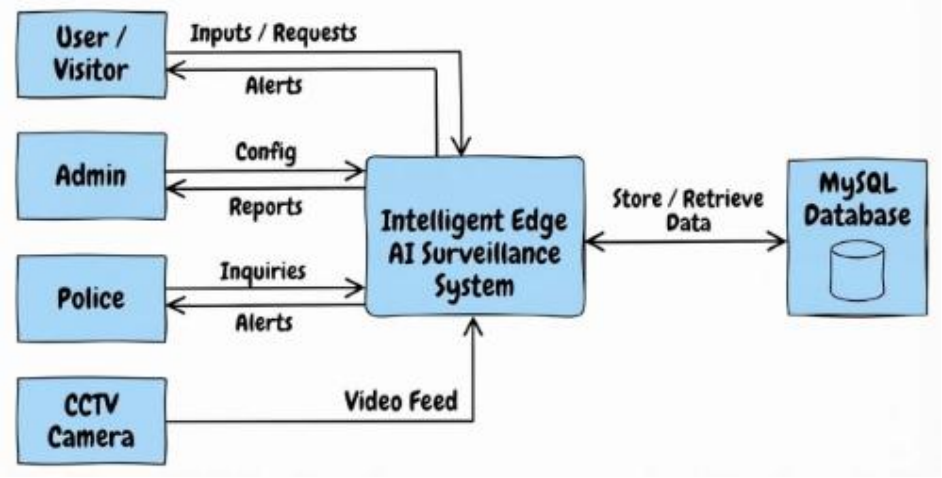


Fig. 3. Level 0 data flow diagram.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, November 2026
Impact Factor: 8.2

The overall data flow diagram (Fig. 4) shows the User, Admin, Police, and CCTV camera supplying management and configuration data and video input to the detection and recognition stage (YOLO, MTCNN, FaceNet). Detection results pass to the event analysis engine, which covers anomaly detection, threat assessment, activity recognition, and object classification, and exchanges data with the MySQL database. The alerts and notification stage then delivers information to the User (mobile app), Admin (web dashboard), and Police (control center), using historical data from the database.

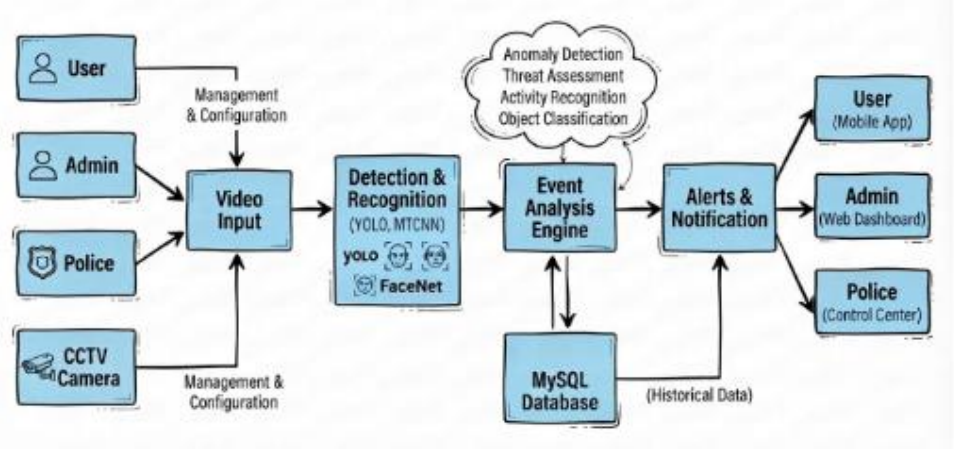


Fig. 4. Overall data flow diagram of the proposed system.

J. System Requirements

The functional requirements are that the system shall: receive video from supported CCTV/IP cameras, DVR/NVR systems, or webcams; process video locally and detect persons and security-related objects; compare detected faces with registered records; support detection of weapons, masked faces, unauthorized persons, and suspicious activities; classify events into Normal, Warning, and Danger levels; capture image evidence and event clips with date and time; generate severity-based notifications; support the Admin, User, and Police functions; and maintain searchable event and notification records.

The non-functional requirements are real-time operation for important events, privacy through local Edge AI processing, resource use suited to a desktop or laptop, compatibility with supported cameras and the Windows environment, usability through web and mobile interfaces, and storage efficiency through event-specific recording.

The specified hardware is an Intel Core i5 (or equivalent AMD) processor, a minimum of 8 GB RAM (16 GB recommended), a minimum 256 GB SSD, and LAN or Wi-Fi connectivity, with a dedicated GPU optional depending on the selected models. The software environment is Windows 10 or 11; HTML, CSS, JavaScript, and Bootstrap for the web interface; Python with Django for the back end; Flutter and Dart for the mobile application; MySQL for the database; and Python, OpenCV, YOLO, MTCNN, FaceNet, TensorFlow or PyTorch, NumPy, Pandas, and Scikit-learn for AI and video processing.

K. Expected Advantages

The expected advantages of the proposed design are real-time monitoring through local analysis; automatic detection of persons and selected security objects; comparison of faces with registered records; Normal, Warning, and Danger alert classification; automatic image and short-clip evidence with date and time; a searchable surveillance history; reduced need for continuous manual monitoring; potential reduction in unnecessary footage storage; lower video-transmission requirements than continuous cloud upload; improved privacy control by keeping major inference on the local computer; potentially lower latency; and integration with existing CCTV infrastructure. These are expected benefits; actual results depend on the selected models, camera quality, lighting, hardware, and network configuration, and must be established through testing of detection accuracy and false alerts.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, November 2026
Impact Factor: 8.2

L. Planned Evaluation

Because the project is at the design stage, no accuracy figures are reported. The planned evaluation will test the system under different lighting conditions, camera angles, image quality, occlusion, and network conditions. It will measure detection accuracy separately for each supported class, namely person detection, face detection, face recognition, weapon detection, and suspicious-activity detection, rather than combining them into a single accuracy value. False-positive and false-negative rates, event-to-notification delay, local processing speed, and the amount of evidence stored compared with continuous recording will also be measured.

M. Constraints and Ethical Considerations

Processing speed depends on hardware, selected models, video resolution, and the number of camera streams, and may vary when models run on a CPU without a GPU. Face recognition depends on image quality and suitable registered records, so every match is treated as a possible identification requiring verification. Access to criminal and missing-person records is restricted to authorized personnel. Notifications and remote services depend on the configured communication method and network availability, and mobile notifications may need network connectivity even though core analysis is local. Detection accuracy, alert delay, storage savings, and reliability remain to be measured after implementation.

IV. CONCLUSION AND FUTURE WORK

This work presented Edge Guard AI, an AI-assisted extension to conventional CCTV monitoring that uses existing camera infrastructure and a local desktop or laptop computer to analyze video, identify selected persons and objects, classify events, capture evidence, and provide severity-based alerts. YOLO, MTCNN, and FaceNet have distinct roles in object detection, face detection, and facial-embedding comparison, while additional trained models may support masked-face, gender, and suspicious-activity detection. The event-management process assigns Normal, Warning, or Danger alerts and links event details with images, video clips, timestamps, and a searchable history. The project remains at the Phase 1 design stage, so implementation, integration, and performance testing are necessary before the expected benefits can be confirmed.

Future work will implement and test the core pipeline of camera input, local processing, record comparison, event classification, evidence storage, and notification. Evaluation will consider lighting, camera angles, image quality, occlusion, and network conditions, and will separately measure person, face, weapon, and suspicious-activity detection along with false-positive and false-negative rates, event-to-notification delay, processing speed, and storage used. Further extensions include vehicle detection, which would widen the range of monitored objects; crowd monitoring, which could analyze crowd density in relevant camera scenes; anomaly detection, which could help identify unusual video patterns; and advanced behavior analysis, which could extend the range of activities handled by the workflow. These are future possibilities and are not claimed as completed modules.

Further development may also refine the database and application interfaces, improve search and review of event evidence, test different local hardware configurations, and evaluate additional alert rules that reduce unnecessary notifications without delaying important warnings. Access to criminal and missing-person records should be limited to authorized personnel, and possible face matches should be treated as predictions requiring appropriate verification. Extensions will be considered only after the core workflow has been implemented and tested, so that the project remains achievable and aligned with its original objectives.

REFERENCES

[1] A. BenAbdennour et al., “Real-Time Human Fall Detection From Video Using YOLOv11 With Pose Estimation: A Paradigm Shift Toward Efficient Transformer-Based Architectures,” *IEEE Access*, vol. 14, pp. 45658–45675, 2026, doi: 10.1109/ACCESS.2026.3674843.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, November 2026 Impact Factor: 8.2

[2] A. Priadana, D.-L. Nguyen, X.-T. Vo, J. Choi, R. Ashraf, and K. Jo, "HFD-YOLO: Improved YOLO Network Using Efficient Attention Modules for Real-Time One-Stage Human Fall Detection," *IEEE Access*, vol. 13, pp. 41248–41258, 2025, doi: 10.1109/ACCESS.2025.3547360.

[3] A. Debnath, Y.-W. Kim, and Y.-C. Byun, "LightSTATE: A Generalized Framework for Real-Time Human Activity Detection Using Edge-Based Video Processing and Vision Language Models," *IEEE Access*, vol. 13, pp. 97609–97627, 2025, doi: 10.1109/ACCESS.2025.3574659.

[4] S. Perikamana Narayanan, M. Sabarimalai Manikandan, and L. R. Cenkeramaddi, "YOLOv9-Based Human Face Detection and Counting Under Human-Animal Faces, Complex Imaging Environments, and Image Qualities," *IEEE Access*, vol. 13, pp. 129600–129637, 2025, doi: 10.1109/ACCESS.2025.3591247.

[5] S. AlShamsi and N. AbuAli, "Lightweight Real-Time Anomaly Detection and Prediction for Surveillance Videos Using a Self-Supervised Hybrid Autoencoder and Dynamic Graph CNN-LSTM Framework," *IEEE Access*, vol. 13, pp. 195423–195435, 2025, doi: 10.1109/ACCESS.2025.3631395.

[6] Chaudhary, S. Kumar, and V. S. Dhaka, "Intelligent Anomaly Detection and Localization With Crowd Density Estimation for Improved Public Safety," *IEEE Access*, vol. 13, pp. 169912–169928, 2025, doi: 10.1109/ACCESS.2025.3614146.

[7] Catargiu and I. B. Ciocoiu, "Real-Time Weapon Detection Using Enhanced YOLO12 Models and a Custom Dataset," *IEEE Access*, vol. 14, pp. 120965–120983, 2026, doi: 10.1109/ACCESS.2026.3721157.

[8] S. B. Veesam and A. R. Satish, "Design of an Integrated Model for Video Summarization Using Multimodal Fusion and YOLO for Crime Scene Analysis," *IEEE Access*, vol. 13, pp. 25008–25025, 2025, doi: 10.1109/ACCESS.2025.3538282.

[9] Mummaneni, T. S. Sappa, and V. G. D. Katakam, "Real-Time Human Tracking and Activity Recognition using GPS, IPS, RFID, and Computer Vision," in *2025 6th Int. Conf. Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, 2025, pp. 1422–1427, doi: 10.1109/ICICV64824.2025.11085683.

[10] "A Hybrid Edge-Cloud Smart Door Surveillance System with Real-Time Risk Assessment and Secure Access Control," 2025. [Authors, venue, DOI to be added.]

[11] S. Kannan et al., "Real-Time Video Surveillance Analysis at the Edge in Smart Cities," in *2025 Int. Conf. Innovations and Emerging Technologies in AI & Communication Systems (IETACS)*, 2025, pp. 1509–1514, doi: 10.1109/IETACS68750.2025.11385417.

[12] "Human Exclusive Alerts in CCTV Surveillance Using Computer Vision," 2025. [Authors, venue, DOI to be added.]

[13] "Integrating Explainable Artificial Intelligence With Advanced Deep Learning Model for Crowd Density Estimation in Real-World Surveillance Systems," 2025. [Authors, venue, DOI to be added.]

[14] "Advanced Detection of Violence From Video: Performance Evaluation of Transformer and State of the Art of Convolution of Neural Network Transformer," 2025. [Authors, venue, DOI to be added.]

[15] S. Waghchaware and R. Joshi, "Hybrid CNN-Transformer Model for Enhanced Real-Time Surveillance of Suspicious Group Activities," *IEEE Access*, vol. 14, pp. 26202–26222, 2026, doi: 10.1109/ACCESS.2026.3663444.