

# AI-Based CCTV Anomaly and Fire Detection System

S.SomeshwaranAP/CSE,P.Divyarasu,K.Balaji,A.Dharanivendhan,R.Ajith,G.Harikrishnan

*S.SomeshwaranAP/CSE, (GUIDE), AVSEngineeringCollege, Ammapet, Salem,*

*P.DivyarasuDept.ofComputerScienceandEngineering, AVSEngineeringCollege, Ammapet, Salem,*

*K.BalajiDept.ofComputerScienceandEngineering, AVSEngineeringCollege, Ammapet, Salem,*

*A.DharanivendhanDept.ofComputerScienceandEngineering, AVSEngineeringCollege, Ammapet, Salem,*

*R.Ajith.ofComputerScienceandEngineering, AVSEngineeringCollege, Ammapet, Salem,*

*G.HarikrishnanDept.ofComputerScienceandEngineering, AVSEngineeringCollege, Ammapet, Salem.*

\*\*\*

**Abstract** - This paper presents the design and implementation of an intelligent AI-driven CCTV surveillance system capable of real-time behavioral anomaly detection and autonomous emergency fire response. Traditional passive security camera installations offer limited capability for proactive threat detection, requiring continuous human monitoring to identify incidents. Our proposed system transforms this paradigm by integrating computer vision, deep learning, and Internet of Things (IoT) technologies into a unified surveillance platform. The system employs a convolutional neural network (CNN) pipeline built on TensorFlow/Keras and OpenCV to continuously analyze live video feeds, detecting physical altercations, falls, panic-induced running, unauthorized intrusions, abnormal crowd gatherings, and suspicious object presence. A dedicated fire and smoke detection module combines color-space thresholding with a trained deep learning classifier. Upon fire detection, the system autonomously triggers ESP32/Arduino-controlled relay circuits that activate water pumps or sprinkler systems. Events are classified into four severity tiers and stored with metadata in a MySQL database. A JavaScript-based web dashboard provides operators with real-time visualizations and historical analytics. Experimental evaluation demonstrates 91.3% average anomaly detection accuracy and 94.7% fire detection precision, with an alert latency under 1.2 seconds on standard hardware.

**Key Words:** Computer Vision, Deep Learning, Anomaly Detection, Fire Detection, Real-time Video Processing, IoT Emergency Response, Convolutional Neural Networks, Smart Surveillance, OpenCV, TensorFlow, ESP32.

## 1. INTRODUCTION

The rapid proliferation of closed-circuit television (CCTV) cameras in urban infrastructure, commercial establishments, transportation hubs, and residential complexes has created an unprecedented volume of visual data that far exceeds the capacity of human operators to monitor effectively [1]. A typical urban surveillance deployment may encompass hundreds of cameras generating continuous 24-hour video streams, making real-time anomaly identification through manual oversight both operationally impractical and economically unsustainable. Studies indicate that human attention begins to degrade significantly after approximately 20 minutes of passive video monitoring, leading to missed incidents and delayed emergency responses [2].

Artificial intelligence and computer vision technologies have matured to a point where automated video analytics can supplement or replace human observers in many monitoring scenarios. Recent advances in deep learning, particularly convolutional neural networks (CNNs) and spatial-temporal models, have demonstrated remarkable capabilities in understanding complex visual scenes, recognizing human activities, and detecting environmental hazards such as fire and

smoke [3][4]. These developments open the possibility of transforming passive CCTV infrastructure into intelligent, proactive safety systems capable of independent decision-making and emergency response.

This paper presents the architecture, implementation, and experimental validation of an integrated AI-based CCTV system designed to address two critical safety challenges: (1) behavioral anomaly detection encompassing a range of threatening or unusual human activities, and (2) automated fire detection coupled with autonomous physical suppression response via IoT-controlled hardware. The system is distinguished by its end-to-end integration of video analytics, risk classification, database logging, web-based monitoring, and direct hardware actuation within a unified, deployable platform.

## 2. RELATED WORK

Video-based anomaly detection has been an active research domain for over two decades. Early approaches relied on handcrafted features such as optical flow, histogram of oriented gradients (HOG), and background subtraction models to identify motion patterns inconsistent with normal activity [5]. While computationally efficient, these methods suffered from high false-positive rates under varying illumination and complex scene backgrounds.

The advent of deep learning significantly advanced the state of the art. Hasan et al. [6] proposed a convolutional autoencoder framework for unsupervised anomaly detection, learning compact representations of normal video frames and flagging high reconstruction error as anomalous. Subsequent work by Luo et al. [7] introduced spatiotemporal autoencoders that capture both appearance and motion in a unified latent space. More recently, attention-based transformer architectures have achieved state-of-the-art results on benchmarks including UCF-Crime [8].

Fire and smoke detection using computer vision has evolved from color-space heuristics to deep learning classifiers. Chen et al. [9] exploited chrominance properties of flames in YCbCr and HSV color spaces. Khan et al. [10] demonstrated that fine-tuned CNNs substantially outperform handcrafted methods across varied environmental conditions. IoT integration for automated emergency response remains underexplored; existing commercial systems typically limit automated actions to alarm activation without direct physical intervention [11]. Our work bridges this gap by coupling AI-driven detection to IoT hardware capable of activating suppression equipment.

## 3. SYSTEM ARCHITECTURE

### Overview

The proposed system adopts a layered, modular architecture comprising five functional tiers: (1) video

ingestion and preprocessing, (2) AI-driven analytics, (3) event management and risk classification, (4) data persistence and web monitoring, and (5) IoT emergency actuation. Incoming RTSP video streams from IP cameras feed into an OpenCV-based capture pipeline. Preprocessed frame sequences are forwarded to the deep learning inference engine, which runs specialized detection modules for behavioral anomalies and fire events. Detected events are scored, classified by severity, and persisted in a MySQL database. Simultaneously, the Spring Boot backend serves real-time event data to the web dashboard via WebSocket. Critical-severity fire events trigger ESP32 microcontrollers via HTTP commands to actuate relay-controlled suppression hardware.

## Technology Stack

The AI analytics engine is implemented in Python 3.10 with OpenCV 4.8 for video capture and TensorFlow 2.12/Keras for deep learning inference. The backend is built on Java Spring Boot 3.1, providing RESTful API endpoints for system interoperability. Storage is managed by MySQL 8.0. The monitoring dashboard is a single-page web application using HTML5, CSS3, and JavaScript with WebSocket support. IoT actuation uses ESP32 development boards running C++ firmware via the Arduino framework, interfacing with 10A relay modules connected to water pump circuits.

## 4. ANOMALY DETECTION MODULE

### Video Preprocessing Pipeline

Incoming video streams are decoded at 15 frames per second (fps). Each frame undergoes spatial downscaling to 224x224 pixels, RGB conversion, normalization to [0, 1], and temporal stacking into clips of 16 consecutive frames to capture motion context. A Gaussian Mixture Model (GMM)-based background subtraction algorithm [12] generates foreground segmentation masks, isolating regions of motion activity. These masks serve as spatial attention priors directing the classifier toward active scene regions, reducing computational load and false positive rates.

### Deep Learning Classifier

The primary anomaly classifier employs a two-branch architecture. The spatial branch uses a modified MobileNetV2 backbone pre-trained on ImageNet and fine-tuned on UCF-Crime and ShanghaiTech Campus datasets. The temporal branch processes dense optical flow fields computed using the Farneback algorithm, encoding motion as two-channel maps fed to a lightweight ResNet-18 variant. Concatenated feature maps are passed through a bidirectional LSTM with 256 hidden units to model temporal dependencies across the 16-frame clip window. The classification head produces independent probabilities for seven classes: fighting, falling, panic running, unauthorized intrusion, crowd gathering, suspicious object, and normal.

### Risk Scoring and Severity Classification

Each detected event is assigned a composite risk score  $R$  computed as a weighted combination of the model's classification confidence, a normalized duration factor, and a camera zone sensitivity weight. Scores map to four tiers: Low ( $R < 25$ ), Medium ( $25 \leq R < 50$ ), High ( $50 \leq R < 75$ ), and Critical ( $R \geq 75$ ). Response actions escalate from silent logging through operator notification, audio alarm, and IoT hardware triggering at the Critical tier, as summarized in Table 1.

**Table 1:** Severity Classification and Response Actions

| Severity | RiskScore | ResponseAction                  |
|----------|-----------|---------------------------------|
| Low      | 0–24      | Log event; no alert             |
| Medium   | 25–49     | Log + operator notification     |
| High     | 50–74     | Alert + audio alarm             |
| Critical | 75–100    | Alert + alarm + IoT suppression |

## 5. FIRE DETECTION AND IOT RESPONSE

### Dual-Stage Fire Detection

Fire detection employs a two-stage pipeline. In the first stage, each frame is converted to HSV and YCbCr color spaces. A pixel is labeled a fire candidate if it satisfies chrominance criteria: hue  $0^\circ$ – $35^\circ$ , high saturation ( $S > 0.4$ ) and value ( $V > 0.6$ ) in HSV, plus YCbCr conditions  $C_b < 120$  and  $C_r > 150$ . Candidate regions exceeding 200 pixels are passed to a fine-tuned InceptionV3 classifier trained on 14,000 fire and non-fire patches from the Kaggle Fire Dataset and VisiFire benchmark. This achieves 94.7% precision and 92.1% recall on the test set. The dual-stage design reserves CNN inference for frames containing plausible fire candidates, enabling real-time operation on CPU hardware.

### Smoke Detection

Smoke detection complements flame identification by analyzing frame regions for low saturation, high brightness variance, and diffuse texture. A classifier using Local Binary Patterns (LBP) histograms distinguishes smoke from benign haze, fog, or steam with 88.4% accuracy on the AIDER dataset [13]. Smoke detection provides early warning before visible flames appear, averaging 34 seconds lead time in smoldering fire scenarios, enabling earlier protective intervention.

### Autonomous IoT Emergency Response

Upon confirmation of fire (InceptionV3 confidence  $> 0.85$  or sustained smoke detection exceeding 5 seconds), the system transmits an HTTP POST request to the ESP32 microcontroller's onboard web server. The ESP32 firmware drives a 5V digital signal to a 10A 250V AC relay module, closing the circuit to a water pump or solenoid-operated sprinkler valve. A hardware watchdog timer deactivates the relay after 120 seconds if no command is received, preventing flooding from network disruption. IoT communication latency from detection to relay actuation measured  $187\text{ms} \pm 23\text{ms}$  in controlled testing.

## 6. DATA MANAGEMENT AND MONITORING INTERFACE

### Database Schema

All events are persisted in MySQL 8.0. The core events table stores: event\_id (primary key), camera\_id, event\_type, severity\_level, risk\_score, timestamp, frame\_image\_path, bounding\_box\_coordinates, and model\_confidence. A cameras table maintains metadata including location descriptions and zone sensitivity weights. Indexing on timestamp and camera\_id fields enables query response times under 50ms for datasets of up to 10 million events.

## WebMonitoringDashboard

The real-time monitoring interface is a single-page web application with a persistent WebSocket connection for low-latency push notification. The dashboard displays a live multi-camera grid with severity-color-coded overlay annotations. A sidebar presents a chronological event feed with thumbnail snapshots, event type, severity, and timestamp. An analytics section provides historical visualizations: event frequency heat maps, severity distribution charts, per-camera incident counts, and CSV-exportable reports for audit and compliance purposes. Operators can acknowledge alerts, add notes, and escalate events via integrated email and SMS notification hooks.

## 7. EXPERIMENTAL EVALUATION

### Experimental Setup

The system was deployed in a controlled laboratory environment with four CCTV cameras simulating a multi-room facility. Hardware consisted of an Intel Core i7-12700 workstation with 32 GB RAM running Ubuntu 22.04 LTS. Model training used an NVIDIA GeForce RTX 3060 GPU; inference was evaluated on CPU only to represent resource-constrained deployments. Two ESP32 boards connected to relay modules and water pump units formed the IoT subsystem. Evaluation spanned 72 hours of continuous operation across three phases: anomaly detection performance, fire detection performance, and end-to-end latency.

### Anomaly Detection Results

Detection performance was evaluated on 1,200 manually labeled test clips spanning all seven classes. The system achieves 91.3% overall accuracy and macro-averaged F1 of 0.904. Performance is strongest for fighting (F1 = 0.943) and falling (F1 = 0.931). Crowd gathering achieves the lowest F1 (0.871) due to variability in group size definitions in ground truth labeling. Per-class results are summarized in Table 2.

**Table 2:** Per-Class Anomaly Detection Performance

| Event Class       | Prec.        | Recall       | F1           |
|-------------------|--------------|--------------|--------------|
| Fighting          | 0.951        | 0.935        | 0.943        |
| Falling           | 0.924        | 0.938        | 0.931        |
| Panic Running     | 0.912        | 0.903        | 0.907        |
| Intrusion         | 0.918        | 0.896        | 0.907        |
| Suspicious Object | 0.903        | 0.889        | 0.896        |
| Crowd Gathering   | 0.864        | 0.878        | 0.871        |
| <b>Overall</b>    | <b>0.920</b> | <b>0.907</b> | <b>0.904</b> |

### Fire Detection and Latency Results

Fire detection was evaluated on 800 positive and 600 negative video clips from the VisiFire dataset and local footage. The dual-stage pipeline achieved precision of 94.7%, recall of 92.1%, and F1 of 0.934. False positives (5.3%) occurred mainly in scenes with bright orange illumination. Smoke-only events were detected with 88.4% accuracy, with an average 34-second lead time before visible flames appeared in smoldering scenarios. End-to-end alert latency averaged

$1.18 \pm 0.21$  s; total physical fire suppression response latency including IoT actuation was 1.37 s, well below typical human operator response times of 8–12 s [2].

### Comparison with Baseline Methods

Table 3 compares the proposed system against three baselines: GMM-based background subtraction, a single-branch CNN without temporal modeling, and a commercial off-the-shelf video analytics SDK. The proposed dual-branch CNN+LSTM architecture outperforms all baselines in accuracy and F1 while maintaining 18.3 fps on CPU.

**Table 3:** Comparison with Baseline Methods

| Method                 | Acc.         | F1           | FPS         |
|------------------------|--------------|--------------|-------------|
| GMM Background Sub.    | 72.4%        | 0.691        | 28.1        |
| Single CNN (no LSTM)   | 84.7%        | 0.831        | 21.5        |
| Commercial SDK         | 87.2%        | 0.856        | 15.2        |
| <b>Proposed System</b> | <b>91.3%</b> | <b>0.904</b> | <b>18.3</b> |

## 8. CONCLUSION

This paper has presented a comprehensive AI-based CCTV surveillance system integrating real-time behavioral anomaly detection, fire and smoke recognition, IoT-driven emergency response, and web-based monitoring into a unified, deployable platform. The system achieves 91.3% overall anomaly detection accuracy, 94.7% fire detection precision, and end-to-end alert latency under 1.2 seconds—a substantial improvement over both traditional computer vision baselines and commercial alternatives.

The autonomous IoT fire suppression capability, with total actuation latency of approximately 1.37 seconds, represents a meaningful advance beyond passive alerting systems, offering potential to significantly reduce property damage and casualties in fire emergencies. The multi-tiered risk scoring framework provides operators with actionable, graduated alerts rather than binary alarms, reducing alert fatigue and supporting more nuanced incident management. Future work will expand the anomaly taxonomy to include weapon detection, deploy federated learning across distributed camera networks, and investigate edge inference on dedicated hardware accelerators for improved fault tolerance.

## ACKNOWLEDGMENT

The authors thank the Department of Computer Science and Engineering at AVS Engineering College for providing laboratory facilities and computational resources used in this research. We gratefully acknowledge the open-source communities behind TensorFlow, OpenCV, and the Spring Boot framework.

## REFERENCES

- [1] A. Brodsky, "The economics of CCTV monitoring: scale, staffing, and technology," *Secur. J.*, vol. 34, no. 2, pp. 112–128, 2021.
- [2] M. Tickner and J. Bhatt, "Limits of sustained attention in security monitoring tasks," *Hum. Factors*, vol. 63, no. 4, pp. 601–613, 2021.
- [3] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [4] A. Karpathy et al., "Large-scale video classification with convolutional neural networks," in *Proc. IEEE CVPR*, Columbus, OH, 2014, pp. 1725–1732.

- [5] R.Cucchiara, C. Grana, M. Piccardi, and A. Prati, "Detecting moving objects, ghosts, and shadows in video streams," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.25, no. 10, pp.1337–1342, 2003.
- [6] M.Hasan, J. Choi, J. Neumann, A. K. Roy-Chowdhury, and L.S. Davis, "Learning temporal regularity in video sequences," in *Proc. IEEE CVPR*, Las Vegas, NV, 2016, pp. 733–742.
- [7] W.Luo, W. Liu, and S. Gao, "A revisit of sparse coding based anomaly detection in stacked RNN framework," in *Proc. IEEE ICCV*, Venice, Italy, 2017, pp. 341–349.
- [8] W. Sultani, C. Chen, and M. Shah, "Real-world anomaly detection in surveillance videos," in *Proc. IEEE CVPR*, Salt Lake City, UT, 2018, pp. 6479–6488.
- [9] T.-H.Chen, P.-H. Wu, and Y.-C. Chiou, "An early fire-detection method based on image processing," in *Proc. IEEE ICIP*, Singapore, 2004, pp. 1707–1710.
- [10] Z. A. Khan, W. Jawad, and A. Usman, "Fire detection in still images using convolutional neural networks," in *Proc. IEEE SMC*, Bari, Italy, 2019, pp. 1018–1023.
- [11] D. Wagner and P. Nguyen, "Automated emergency response systems: a survey of commercial IoT solutions," *IEEE IoT J.*, vol. 9, no. 7, pp. 5201–5215, 2022.
- [12] Z. Zivkovic, "Improved adaptive Gaussian mixture model for background subtraction," in *Proc. IEEE ICPR*, Cambridge, UK, 2004, pp. 28–31.
- [13] E. Ince, M. Ozalp, and I. Basyigit, "AIDER: Aerial image dataset for emergency response," *IEEE Access*, vol.9, pp.72422–72435, 2021.