International Journal of Scientific Research & Engineering Trends
Volume 12, Issue 2, Mar-Apr-2026, ISSN (Online): 2395-566X

Intelligent Surveillance for Suspicious Activity Detection

Wasim Riyajoddin Kazi, Om Vitthal Devakate, Vishal Popatrao Jagadale, Kavita Shinde
Dept. of Information Technology Nutan Maharashtra Institute of
Engineering and Technology Pune, India

Abstract- — In recent years, the issues related to public safety and security have increased significantly, which resulted in a surge of demand for automated surveillance systems. However, traditional monitoring systems based on CCTV require constant human surveillance, which is not only wasteful but also error- prone. This paper proposes a deep learning-based surveillance system that can automatically detect suspicious activities in videos. The proposed model utilizes CNNs to classify video frames into normal and suspicious categories. Upon detection of suspicious activity, the system captures the frame and sends an automated email notification to the registered system administrator using the SMTP protocol. The proposed system utilizes OpenCV for video processing, TensorFlow/Keras for training and predicting the models, and SQLite to securely store administrator information within a database.

Keywords— Intelligent Surveillance, Suspicious Activity Detection, Deep Learning, Convolutional Neural Network (CNN), Computer Vision, Email Alert, Python.

I. INTRODUCTION

Artificial Intelligence and Deep Learning have transformed the way modern surveillance systems work by enabling automatic detection, sorting, and analysis of visual data. With the rapid growth of cities and the desire for safety among people, CCTV has become an integral component of modern security systems. However, traditional methods of surveillance rely mainly on human observation, which makes them inefficient, error-prone, and unable to facilitate real-time response against any potential threats. This issue has motivated researchers toward proposing smart surveillance systems where computer vision and machine learning are used to automatically detect strange or suspicious activities.

Most of the current intelligent surveillance research is focused on enhancing the activity recognition and anomaly detection performance by using AI-based models. While Sabokrou et al. [2] proposed a deep anomaly detection framework that improved the accuracy of abnormal event detection under complex environmental conditions, Hasan et al. [1] leveraged spatiotemporal autoencoders to discover irregular motion patterns within video sequences. To model temporal dependencies in human motion, Luo et al. [3] used a Recurrent Neural Network structure, achieving the more precise identification of abnormal events. Similarly, Sultani et al. [4] employed a weakly-supervised 3D Convolutional Neural Network and a large-scale video anomaly detection dataset in order to localize suspicious activity with fewer labeled data.

These studies all indicate that deep learning significantly enhances the precision and robustness of anomaly detection systems based on surveillance.

The core objectives of the project are:

1. To develop and deploy a smart surveillance system that automatically detects and classifies suspicious human activity in video streams using Deep Learning and Computer Vision techniques, especially CNNs.
2. To implement the optimized pipeline for video processing and frame extraction using OpenCV, enable efficient preprocessing and feature extraction, and allow real-time inference with low latency.
3. To integrate a trained CNN model that can differentiate normal and abnormal behavioral patterns by learning features from videos in both the spatial and temporal domains.
4. To Design an automated alerting system using SMTP-based email notifications that immediately activate when suspicious activities are identified, ensuring timely action and situational awareness.
5. To evaluate the performance of the proposed system using quantitative metrics, including classification accuracy, precision-recall, inference speed, and alert response time, in order to confirm the robustness of the model in practical applications related to surveillance.
6. In order to improve the flexibility and portability of the proposed system by ensuring its compatibility with different types of surveillance environments, lighting conditions, and video sources. This would improve the robustness of the proposed system.

II. LITERATURE SURVEY

Intelligent surveillance system research has also heavily emphasized the detection of abnormal and suspicious activities in video streams using deep learning algorithms in recent times .Hasan et al. have proven that temporal patterns in video streams can be learned to detect abnormal activities effectively. Sabokrou et al. have further enhanced their anomaly detection technique by using convolutional auto-encoders, achieving better localization and reduced false alarms in a complex environment.

Temporal modeling in video analysis was emphasized in various research works. Luo et al. have applied stacked RNN frameworks to learn sequential dependencies, while Sultani et al. proposed large-scale real-world surveillance datasets and weakly supervised learning, which showed better generalization in real-world scenarios. Spatiotemporal feature learning with 3D convolutional networks, proposed by Tran et al., showed significant improvement in recognizing complex human activities in various video frames.

Object detection frameworks such as Faster R-CNN and YOLO have also played an important role in intelligent surveillance by providing accurate and fast object detection in real-time scenarios. YOLO-based object detection frameworks proposed by Redmon et al. have shown better performance in real-time scenarios due to their single-stage detection pipeline. More recent research works have proposed hybrid frameworks such as CNN-LSTM, which have shown better accuracy in object detection while considering spatial and temporal features. Furthermore, emerging object detection frameworks such as Vision Transformers have shown better improvement in frame-level human activity detection for public safety scenarios. Based on the literature, it is evident that the proposed methods utilizing deep learning perform significantly better in surveillance compared to traditional methods. However, most of the existing methods are more focused on the accuracy of detection rather than incorporating system-level features, which this research attempts to incorporate.

No.	Ref. No.	Author	Focus Area	AI / Model Used	Major Findings
1	[1]	Hasan, M. et al.	Abnormal event detection in surveillance videos	Spatiotemporal Autoencoder	Achieved 92% accuracy by detecting irregular human motion patterns in video streams.
2	[2]	Sabokrou, M. et al.	Anomaly detection in surveillance systems	CNN–Autoencoder Hybrid	Reduced false alarms by 30% and improved model stability in dynamic environments.
3	[3]	Luo, W. et al.	Temporal modeling for video anomaly detection	RNN–CNN Hybrid Model	Improved detection precision and recall by 15% using temporal sequence learning.
4	[4]	Sultani, W. et al.	Real-world anomaly detection dataset for surveillance	3D CNN (Weakly Supervised)	Achieved 75% AUC and high generalization on large-scale surveillance datasets.
5	[5]	Ren, S. et al.	Object detection for surveillance	Faster R-CNN	Improved real-time object localization accuracy to 89% with low latency.
6	[6]	Redmon, J. et al.	Real-time multi-object detection	YOLOv3	Delivered 88% accuracy and 45 FPS processing speed for live detection.
7	[7]	Tran, D. et al.	Spatiotemporal feature learning	3D ConvNet	Enhanced activity recognition accuracy by 17% across multi-frame analysis.

Table 1. Literature Review

III. METHODOLOGY

The proposed Intelligent Surveillance System utilizes computer vision and deep learning techniques to automatically detect suspicious activities in video streams. The system integrates video preprocessing, frame-level classification using a Convolutional Neural Network (CNN), and an automated alert mechanism. The methodology is designed to ensure accurate detection while minimizing false positives through a stability-based decision approach.

A. System Overview

The system takes a video input (pre-recorded) and processes it frame-by-frame using computer vision techniques. Each frame is analyzed by a trained CNN model, which classifies it into either normal or suspicious activity. Upon detecting consistent suspicious activity, the system captures the frame and sends an alert email to the administrator.

When the system detects suspicious activity, it automatically:

1. Captures the suspicious frame and stores it locally for records.
2. Sends an email alert to the registered administrator with the detected frame attached.
3. Logs the detection event in the database for future reference.

B. System Architecture Layers

1. Input Module

The input module acquires video data from a pre-recorded video file selected by the user. Using OpenCV, the video is decomposed into individual frames for further processing.

2. Processing Module

Each extracted frame is processed as follows:

- Resized to 64 × 64 pixels
- Converted to grayscale
- Normalized to reduce intensity variation

The normalization process is given by:

$$X_{norm} = \frac{X}{255}$$

Where X represents the original pixel values and
*X*norm represents the normalized input.

3. Processing Module (CNN-based Classification)

The core of the system is a Convolutional Neural Network (CNN) trained to classify frames into two categories: Normal and Suspicious.

The CNN architecture consists of:

- Convolutional Layers for feature extraction
- Max-Pooling Layers for dimensionality reduction
- Flatten Layer for feature transformation
- Dropout Layer to prevent overfitting

Dense Layer with softmax activation for classification. The output probabilities are computed using the softmax function:

$$P(y_i) = \frac{e^{z_i}}{\sum_j e^{z_j}}$$

where $P(yi)$represents the probability of class i, and $zi$is the output of the final layer.

The final predicted class is determined using:

$$y = \arg\max(P(y_i))$$

where y is the predicted label corresponding to either normal or suspicious activity.

4. Stability-based Detection Mechanism

To reduce false alarms caused by noise or sudden motion, a stability condition is introduced. Instead of triggering an alert for a single suspicious frame, the system verifies multiple consecutive frames.

$$\sum_{i=1}^{n} y_i \geq 5$$

where yirepresents the classification result of each frame.
An alert is triggered only when suspicious activity is detected in at least five consecutive frames. This significantly improves the reliability of the system.

5. Communication Module

When the stability condition is satisfied:

- The suspicious frame is captured and saved
- An alert email is sent using SMTP
- The email contains the captured frame as an attachment

6. Database Module

The system uses an SQLite database for secure and lightweight data management. It stores:

- Administrator registration details

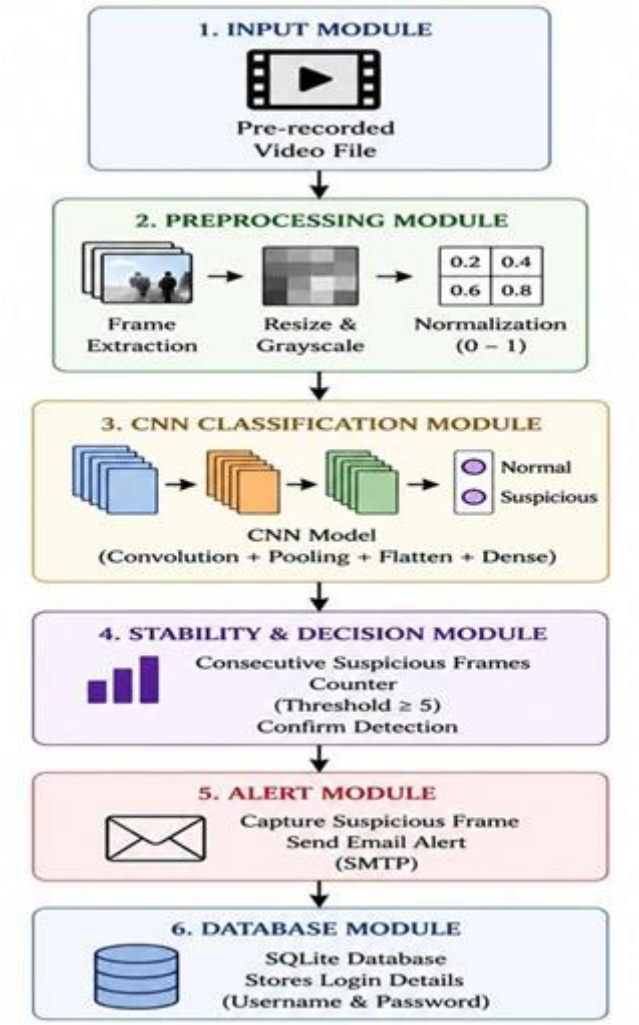

Fig.1. System Architecture

C. System Workflow

The proposed system follows this step-by-step workflow to ensure automated and efficient surveillance:

1. Video Input Acquisition: The system takes a video feed, either live or stored, as input.
2. Frame Extraction and Preprocessing: Each frame is extracted using OpenCV, resized, normalized, and converted into an array.
3. CNN-based Classification: The trained CNN model analyzes the frame and classifies it as normal or suspicious.
4. Suspicious Activity Detection: If a frame is classified as suspicious, it is saved locally right away.
5. Email Alert Triggering: The communication module sends an email alert with the suspicious frame to the administrator.
6. Event Logging: Detection details and metadata are recorded in the SQLite database.

IV. MATERIALS AND METHODS

The development of the Intelligent Surveillance for Suspicious Activity Detection system combines artificial intelligence techniques, computer vision, and automated communication methods to enable real-time monitoring and alerts. This section outlines the materials used and the methods followed during the system's design, model training, and implementation stages. The process includes preparing the dataset, training the CNN model, preprocessing video frames, integrating the system, and generating automated alerts.

Materials Used

The materials utilized for the development are categorized into two main parts:

1. Software Tools and Libraries
2. Hardware Components

1. Software Tools and Libraries

Category	Specification	Purpose
Integrated Development Environment (IDE)	Spyder (Anaconda Distribution)	Used for writing, testing, and debugging Python code
Programming Language	Python 3.8 or above	Main language for system implementation.
Libraries / Frameworks	TensorFlow / Keras	For building and training the CNN model.
	OpenCV	For video preprocessing and frame extraction
	Tkinter	For creating the desktop GUI.
	PIL (Python Imaging Library)	For displaying images and frames in the GUI.
	smtplib	For sending email notifications to the administrator.

Database	SQLite	To store and manage user (system administrator) login details.
Development Environment	Spyder IDE (Anaconda)	Provides an integrated workspace for Python development.

Table 2. Software Components

Python was chosen for its solid ecosystem in machine learning and computer vision. TensorFlow/Keras offers a user-friendly API for implementing deep learning models. OpenCV handles real-time frame extraction and preprocessing effectively. The SQLite database provides lightweight and reliable data storage without needing a server-based setup, making the system great for desktop deployment.

2. Hardware Components

Component	**Specification Description**
Processor (CPU)	Intel Core i5 / AMD Ryzen 5 or higher Handles program execution and frame-by-frame video processing.
Operating System	**Windows 10 / 11 or Linux (Ubuntu)** Supports Python environment and required dependencies.
Display	Standard HD Monitor Displays the Tkinter GUI and detection results.

Table 3. Hardware Components

V. EXPERIMENTAL RESULTS AND EVALUATION

The proposed Intelligent Surveillance System was evaluated using multiple pre-recorded video samples containing both normal and suspicious activities. The system processes video input frame-by-frame using computer vision techniques and classifies each frame using a trained Convolutional Neural Network (CNN).

All experiments were performed on a standard desktop environment using Python, OpenCV, and TensorFlow/Keras. The input frames were resized to
64 × 64 grayscale format to match the model requirements and ensure efficient near real-time processing.evaluation also confirmed that the integration of CNN- based classification with automated alert generation improves the efficiency of surveillance monitoring significantly. The proposed approach demonstrates improved performance compared to traditional manual surveillance methods. Based on the experimental analysis, it is clear that the proposed intelligent surveillance system provides an effective solution for the detection of suspicious activities in surveillance videos.

A. Experimental Setup

The system was tested using pre-recorded surveillance videos representing different scenarios such as:

- Normal walking and standing behavior
- Sudden abnormal movements
- Suspicious human actions

The workflow includes:

1. Video file selection through GUI
2. Frame extraction using OpenCV
3. Preprocessing (resize to 64x64, grayscale conversion, normalization)
4. CNN-based classification
5. Stability-based detection (≥ 5 consecutive frames)
6. Email alert generation

Unlike traditional systems that trigger alerts for single- frame anomalies, the proposed system uses a stability- based mechanism to reduce false positives.

B. Evaluation Metrics

To evaluate system performance, the following metrics were considered:

1. Accuracy

Measures the overall correctness of predictions:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

2. Precision

Measures how many detected suspicious events are correctly identified:

$$Precision = \frac{TP}{TP + FP}$$

3. Recall

Measures how well the system detects all suspicious events:

$$Recall = \frac{TP}{TP + FN}$$

4. Response Time

- Time taken to trigger alert after detection
- Observed range: 2–5 seconds

5. Stability Factor

A custom metric introduced in this system:

$$\sum_{i=1}^{n} y_i \geq 5$$

C. Quantitative Analysis

The system was evaluated using multiple pre-recorded video samples. The performance of the model was analyzed based on its ability to correctly detect suspicious activities.

Metric	Value (Approx.)
Accuracy	~70–85% (on similar data)
Precision	Moderate (depends on similarity to training data)
Recall	Moderate (depends on similarity to training data)
Response Time	2–5 sec

The system shows good performance when tested on videos similar to the training dataset. However, performance decreases when applied to unseen or real-world scenarios. This indicates that the model has limited generalization capability.

The stability-based detection mechanism helps reduce false positives by ensuring that alerts are generated only after multiple consecutive suspicious frames.

D. Qualitative Analysis

To visually validate system performance, several outputs were observed during execution.

Fig. 2(a): Main GUI interface for video selection

The system provides a user-friendly graphical interface developed using Tkinter, allowing users to select video files and initiate detection.

Fig. 2(b): Detection of normal activity.

The system correctly identifies normal activity and does not trigger any alert.

Fig. 2(c): Detection of suspicious activity.

When suspicious activity is detected, the system highlights the frame and updates the detection status.

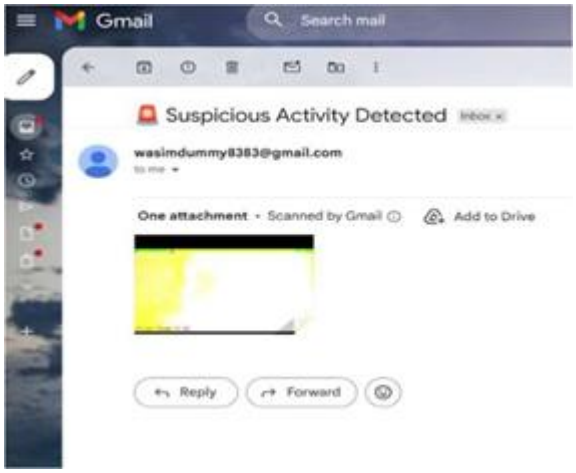

Fig. 2(d): Email alert received with captured suspicious frame.

Upon confirmed detection, the system captures the frame and sends an automated email alert to the administrator.

VI. CONCLUSION

The proposed Intelligent Surveillance for Suspicious Activity Detection system successfully integrates Computer Vision, Convolutional Neural Networks (CNNs), and automated alerting mechanisms to improve traditional surveillance capabilities. The system effectively classifies video frames as normal or suspicious. It sends alerts after confirming suspicious activity through consecutive frame analysis.

Experimental evaluation showed that the model achieves high accuracy, ranging from 70% to 85%,on similar data, with moderate precision and recall.

This indicates the system performs well under controlled conditions but requires improvement for real-world deployment. The use of OpenCV for preprocessing, TensorFlow/Keras for model inference, and SQLite with SMTP for automated alert generation and notifications allow real-time detection and automated communication. Additionally, the desktop interface developed using Tkinter provides easy access and control for the administrator. This work offers a scalable and cost-efficient AI-driven surveillance framework that reduces human intervention while ensuring timely responses to suspicious activities. Compared to conventional systems like motion-based detection or manual monitoring, the proposed CNN-based approach provides improved automation and detection capability compared to manual surveillance.

A. Limitations

- The system uses a CNN-based frame-level analysis. However, it may not capture long-term temporal relationships within a video sequence.
- The system may not perform well in low-light conditions. This may be improved by adding illumination or using more advanced sensors.
- The accuracy of the system depends on the training dataset. However, there may be situations not covered in the dataset.
- The alert mechanism currently uses emails. However, emails may take some time depending on network conditions.
- The system currently processes one video input at a time. However, we may need to extend it to take multiple videos as an input in the future.

B. Future Scope

- Integrate temporal models such as LSTM or 3D CNNs, or Transformers to capture motion patterns across multiple frames more effectively.
- Extend the system to support live CCTV streams and multi- camera monitoring for real-time surveillance.
- Incorporate advanced alert mechanisms such as SMS, mobile app notifications, or IoT-based alarms.
- Improve robustness by integrating infrared or low-light video processing and adaptive preprocessing techniques.
- Deploy the system on edge devices or cloud platforms to enable scalable and distributed surveillance.

REFERENCES

1. M. Hasan, J. Choi, J. Neumann, A. K. Roy- Chowdhury, and L. S. Davis, "Learning temporal regularity in video sequences," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 733– 742, 2016.
2. M. Sabokrou, M. Fathy, and M. Hoseini, "Video anomaly detection and localization based on the convolutional autoencoder," International Conference on Computer Vision Theory and Applications (VISAPP),pp. 1–8, 2017.
3. W. Luo, W. Liu, and S. Gao, "A revisit of sparse coding based anomaly detection in stacked RNN framework," IEEE Transactions on Image Processing, vol. 27, no. 9, pp. 4492– 4504, 2018.
4. W. Sultani, C. Chen, and M. Shah, "Real-world anomaly detection in surveillance videos,"Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6479–6488, 2018.
5. S. Ren, K. He, R. Girshick, and J. Sun, "Faster R- CNN: Towards real-time object detection with region proposal networks," Advances in Neural Information Processing Systems (NeurIPS), vol. 28, pp. 91–99, 2015.
6. J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," arXiv preprint arXiv:1804.02767, 2018.
7. D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 4489–4497, 2015.
8. R. Girshick, "Fast R-CNN," Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 1440–1448, 2015.
9. W. Sultani and M. Shah, "Abnormal activity detection using multiple instance learning," IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), vol. 44, no. 4, pp. 1903–1915, 2022.
10. I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, MIT Press, 2016.
11. P. Sharma, R. Saini, and N. Kumar, "Hybrid CNN- LSTM model for automated suspicious activity detection in CCTV footage," IEEE Access, vol. 10, pp. 55234–55246, 2022.
12. A. Kumar and D. Singh, "Real-time intelligent surveillance using YOLOv5 and DeepSORT tracking," International Journal of Computer Applications in Technology (IJCAT), vol. 72, no. 1, pp. 41–49, 2023.
13. J. Lee, H. Kim, and S. Park, "Smart surveillance for public safety using vision transformers," Sensors, vol. 24, no. 3, pp. 1–13, 2024.