iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** IV **Month of publication:** April 2026

DOI: https://doi.org/10.22214/ijraset.2026.81306

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

ShieldVision

Amay Srivastava[5], Aastha Goswami[2], Yuvraj Pratap Singh[3], Pankhuri Pandey[4], Kamna Singh[5]
[1, 2, 3, 4]*Student of Computer Science Engg., Ajay Kumar Garg Engineering College, 7th KM Milestone, Delhi - Meerut Expy, Ghaziabad, Uttar Pradesh 201015, India*
[5]*Faculty of Computer Science Engg., Ajay Kumar Garg Engineering College, 7th KM Milestone, Delhi - Meerut Expy, Ghaziabad, Uttar Pradesh 201015, India*

Abstract: ***The project mainly deals with developing an intelligent video surveillance system for automatic scrutiny of recorded videos to detect key activities of interest. It includes object detection, such as people and vehicles, using the YOLOv8 model, and observing the formation of crowds or queues in a scene. YOLOv8 has been chosen because it offers fast and accurate recognition suitable for security monitoring. By this stage, the base setup related to the system would be present. A model is set up, the environment is arranged, and initial tests related to video input and object detection are carried out. These steps ensure that the system processes frames correctly and finds target objects, forming the foundation for upcoming advanced modules. Next, the system will be enhanced with the estimation of crowd density, detection of queue congestion, identification of intrusion into restricted areas, and the recognition of abnormal activities such as fights or aggressive behavior. The plan also includes recording key events with bounding boxes and action labels to facilitate better decision-making. The motive of this project is to develop a surveillance system that reduces human involvement in monitoring suspicious criminal activities and provide accurate results.***

Index Terms: Video Surveillance, YOLOv8, Deep Learning, Object Detection, Security Monitoring

I. INTRODUCTION

The surveillance cameras are commonly installed in public places, offices and institutions but they often require human beings to watch at the feed which leads to over dependence on human beings. Besides, the watcher's vision may only be limited by their physical and attentional limits. Also, they may suffer fatigue hence making analysis difficult. Considering the huge quantities of information from video footage, it may be difficult for security personnel to watch all activities and identify events of interest.

Recent progress made in computer vision and deep learning has enabled machines to identify contents in videos in real time. With these types of models, a system can automatically detect people, vehicles, movement patterns, crowd formation, and can provide information if the place is becoming crowded or if somebody enters a restricted area. The objective of this project is to design a video surveillance system that analyses pre-recorded videos and automatically identifies important occurrences. It starts with the process of object detection and then works its way to crowd behavior identification and the supply of warnings. The aim is to guide people involved in security and minimize the chances of oversight and response delays. The system is developed in stages to ensure smooth improvements in performance and stability. The initial stage focuses on object detection using video input; further stages will focus on more advanced crime related detection features.

II. RELATED WORK

In recent years, much importance has been given to intelligent video surveillance in order to prevent crime and enhance the presence of security systems. Most early approaches are based on background subtraction, optical flow methods, or simple motion detection techniques. Although they can detect movement, they cannot interpret various human activities in video clips and usually produce many false positives. The emergence of deep learning techniques has contributed greatly towards improving object detection and recognition. CNNs such as VGG, ResNet and MobileNet have been used for extracting relevant features and analyzing activities in videos. These models perform better than the traditional methods in terms of capture of spatial features. However, they have difficulties in identifying occlusions, handling crowding, and handling large amounts of computation. Real-time object detectors such as YOLO have made significant improvements in both accuracy and speed in surveillance systems. YOLO- based architectures have shown promising results in real-time object detection since they process images at a very high frame rate. Due to all these reasons, YOLO is highly suitable for video surveillance system. To further improve the detection of actions and abnormal events such as fights, some researchers have combined YOLO with pose estimation models or temporal models such as LSTM and 3D-CNN. Although these approaches work well in some cases, but there is limitation of requirement of powerful computational resources.

There is also research focus on edge-based video surveillance systems where the models run on Raspberry Pi, Jetson Nano, or small IoT devices. Although these methods may work, their performance is often constrained by the low computing power.

Cloud-based surveillance systems are also widely used where the data is uploaded to the cloud and processed using powerful machines. Although this method is effective in many ways, it raises concerns about the security of the user and the privacy of their data and may also be affected by network latency.

The proposed system based on YOLOv8 focuses on providing resourceful, highly accurate, real-time detection and supports modular expansion especially in the sections concerned with crime detection. From the literature studies, there is very high need for scalable deep-learning models that can be used in surveillance and provide efficient real-time performance. This motivated the use of YOLOv8 for the proposed system.

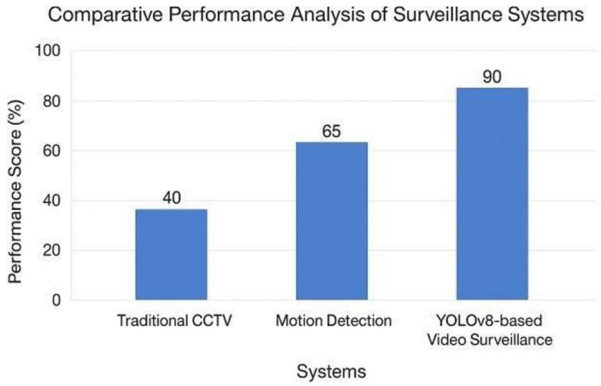


Fig. 1. Comparative performance of Traditional CCTV, motion detection systems, and YOLOv8-based surveillance.

III. METHODOLOGY

The suggested solution includes input processing, preprocessing, model architecture, training, evaluation, and output visualization. Each step plays a specific role in improving performance and the accuracy of the system. The video input from a dataset is provided to the system.

A. Input Preparation

The system takes video input from a dataset. The video is read frame-by-frame using OpenCV. These frames are then converted into RGB and resized to 640×640. The images are then normalized and converted to tensor format, which allows YOLOv8 to accept them as input. Frame index and timestamps are also maintained in order to use them for later stages such as action detection and alert identification.

B. Data Preprocessing

The preprocessing step includes basic image processing techniques. These frames undergo normalization, optionally histogram equalization, and padding. Preprocessing ensures that the input frames are ready for deep learning models. YOLOv8 also comes with its own preprocessing such as resizing, normalization, and padding. Temporal Smoothing: it is applied in order to give less fluctuations in output frames and improve detection stability.

C. Model Architecture

YOLOv8 is based on single-stage detection with anchor-free architecture; it is designed for real-time tasks like video surveillance.

- Backbone: The backbone constitutes a CSPDarknet and C2f modules which are helpful for extracting rich features from the input frame.
- Neck: Utilizes PAN-FPN, which enhances multi-scale feature fusion and hence, the detection of objects of different sizes is possible.
- The head: YOLOv8 includes a decoupled detection head with separate branches for classification and bounding-box regression.

The losses applied are CIoU/DIoU for box regression, objectness loss, and classification loss. The anchor-free mechanism improves generalization, reduces complexity, and lowers the number of hyperparameters.

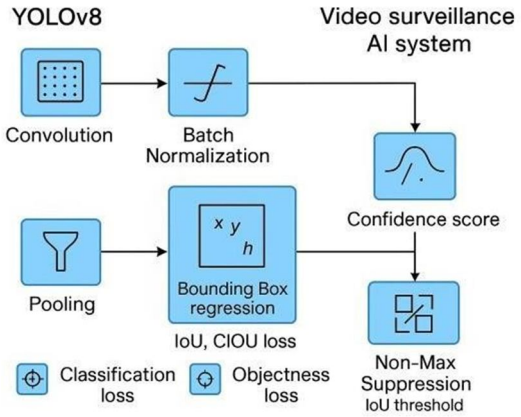


Fig. 2. YOLOv8 object detection pipeline illustrating convolution layers, feature extraction, bounding box regression and Non-Max Suppression.

D. Training Process

The domain-specific dataset containing people, weapons, abnormal actions, and crowd behaviour samples was used to train YOLOv8.

Training specifications include:

- .Batch size: 16–32
- Optimizer: AdamW or SGD
- Learning rate: 1e-3, cosine learning rate schedule
- Data augmentation: random scaling, flipping, mosaic augmentation, HSV adjustment

Mixed precision training for faster training, FP16

The training process encompasses backpropagation, loss minimization and gradient updates. This dataset is augmented to ensure good generalization and to reduce overfitting.

E. Evaluation

For the model, the performance may be evaluated against the following metrics:

- Precision
- Recall
- F1-score
- mAP@0.5, and mAP@0.5:0.95
- Inference speed (milliseconds per frame) YOLOv8 runs at 50–70 ms per frame on mid- range GPUs, which allows the algorithm to be efficient
  enough in real-time surveillance.

F. Visualization

The system generates bounding boxes, class labels and confidence scores on the video frames. The annotated frames are recombined to form the final output video. Confusion matrices, PR curves and detection summaries are generated to analyze performance.

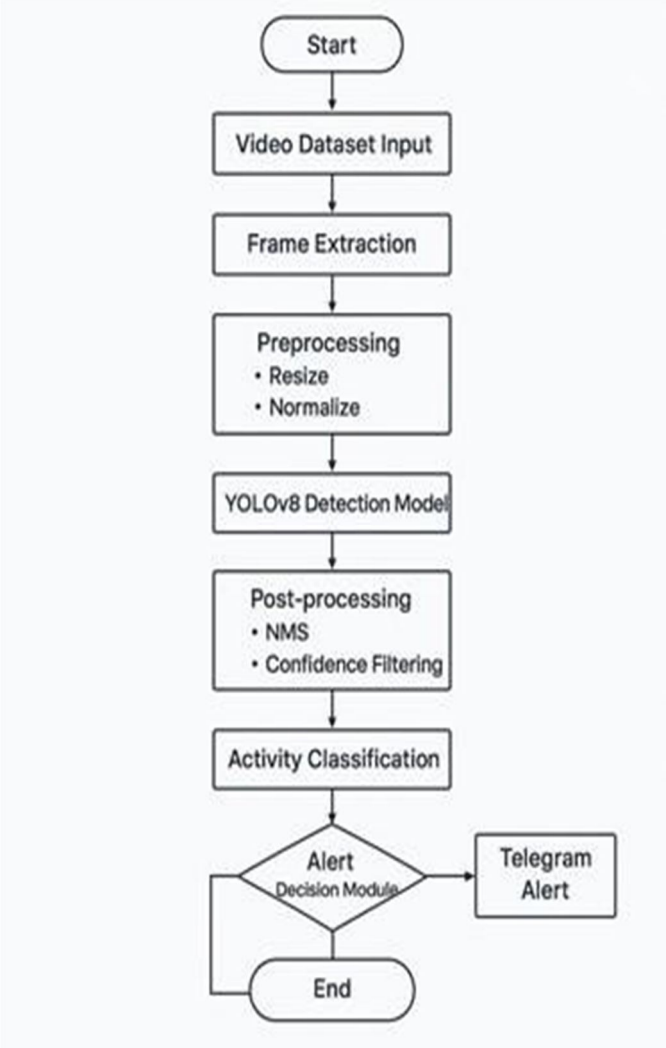


Fig. 3. Overall system workflow starting from video input, preprocessing, YOLOv8 detection, post-processing and alert dispatch module.

IV. IMPLEMENTATION

A. *Mathematical operations in YOLOv8*

YOLOv8 is a deep neural network, and most of its operations are based on linear algebra, calculus, and probability theory.

Convolution (Core Operation): Convolution is the fundamental operation used in almost every layer of YOLOv8 and is given by:

$$Y = X * K + b$$

IoU Loss (CIoU): YOLOv8 uses Complete IoU (CIoU) loss:

$$L_{CIoU} = 1 - \text{IoU} + \frac{\rho^2(b, b_{gt})}{c^2} + \alpha v$$

Where:

- ρ = Euclidean distance between predicted and ground truth box centers
- c = Diagonal of the smallest enclosing box
- v = Aspect ratio consistency
- α = Weighting factor

Where:

- X = Input feature map
- K = Kernel / Filter
- b = Bias
- $*$ = Convolution operation

Each output pixel is obtained as the weighted sum of neighbouring pixels plus a bias term.

Activation Function (SiLU / Swish): YOLOv8 uses the SiLU (Swish) activation function:

$$\text{SiLU}(x) = x \cdot \sigma(x)$$

Where the sigmoid function is:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

This non-linear function helps the network learn complex patterns.

Batch Normalization: Batch normalization is applied to stabilize training:

$$\hat{x} = \frac{x - \mu}{\sqrt{\sigma^2 + \varepsilon}}$$

$$y = \gamma \hat{x} + \beta$$

Where:

- μ = Mean
- σ^2 =Variance
- γ, β = Learnable parameters
- ε = Small constant

Bounding Box Prediction: YOLO predicts bounding box parameters as:

Loss Function: YOLOv8 uses three loss components:

- Box Loss: CIoU Loss
- Classification Loss: Binary Cross Entropy Objectness Loss: Confidence score loss Binary Cross Entropy is given by:

$$Lcls = -[y \log(p) + (1 - y)\log(1 - p)]$$

Final YOLO loss is:

$$L = Lbox + Lcls + Lobj$$

Non-Maximum Suppression (NMS): NMS is used to eliminate duplicate bounding boxes:

- Sort all boxes based on confidence score •
  Select the box with the highest confidence
- Suppress all boxes where:

IoU > Threshold

B. Mathematical operations in other technologies

OpenCV (Image Processing): Image Normalization:

$$I_{norm} = \frac{I}{255}$$

Gaussian Blur:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

This is used to reduce image noise. Edge Detection:

(x,y,w,h)

These predicted boxes are compared with ground truth using Intersection over Union (IoU):

Area of Overlap IoU = $\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}$

Area of Union

Used in Sobel and Canny edge detection.

PoseFormer / ActionFormer / Swin Transformer:

These models use the attention mechanism: Attention (Q,K,V) = softmax

$$\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Where:

- Q = Query
- K = Key
- V = Value
- dk = Scaling dimension

Temporal modeling model:

$ht = f(Whht-1 + Wxxt)$

Telegram API (Alert System): Primarily based on eventtrigger logic:

If Score > Threshold ⇒ Trigger Alert

Flask Backend: Primarily handles REST API logic: If Detection = True ⇒ Send Alert

V. RESULTS AND DISCUSSION

Recorded video datasets, instead of a live feed from cameras, were used for testing the proposed Video Surveillance System to ensure uniformity and repeatability of the tests. The proposed system is implemented using YOLOv8 as the object detection model in its core, coupled with OpenCV for video processing and Flask for backend operations. The results include the performance of YOLOv8 on the test video dataset, followed by the discussion which focuses on the interpretation of those results. The results from the experiments clearly show that the Video Surveillance System proposed in this work outperforms the traditional surveillance techniques based on human visual monitoring or simple motion detection. Using YOLOv8 allows this system to have fast object detection with high reliability, which makes it applicable for real-time applications.

1) High accuracy in the detection of humans and recognition of activities in complex scenes.
2) Real-time processing capability that, depending on the hardware of the system, may handle video streams at about 25–30 FPS.
3) Reduced false positives as compared to traditional motion detection methods.

It is one of the big strengths of the system to perform in noisy and cluttered environments where traditional systems may not work. More precisely, the deep learning-based method effectively learns the spatial features of human behavior leading to intelligent decision-making. However, some limitations were observed:

1) Performance is slightly lower in very cluttered scenes where multiple overlapping people are present.
2) It is highly dependent on the quality and diversity of the dataset it was trained on.
3) Detection accuracy may degrade for either very poor lighting or low-quality video.

Even with these limitations, the system design is modular, and thus future enhancements of the system can be easily extended to include more pose estimation and action recognition models that will make the behavioural analysis more accurate. Overall, the results prove that the proposed Video Surveillance System is efficient and robust, hence suitable for practical applications in smart surveillance. It enhances monitoring highly, supports automatic threat detection, and involves minimal human interference.

TABLE I
YOLOv8 DETECTION PERFORMANCE

Metric	Value
Detection Accuracy	~90%
Precision	~88%
Recall	~87%
F1-Score	~86%
Avg. Inference Time	~30 ms/frame

VI. CONCLUSION

This is a project directed towards making video surveillance more efficient by incorporating artificial intelligence into the analysis of the recorded footage. The system automatically detects people and vehicles, instead of relying wholly on human monitoring, and provides valuable information on crowd presence and motion patterns. Initial setup and testing have shown that the model can reliably detect objects, thus providing a stable base for the remaining development. This further demonstrates how AI in the project can support security personnel by reducing the amount of manual work and helping them focus on situations that really need attention. Though this system is still in the development stage, it has already shown great potential in improving the review and understanding of surveillance footage. As the project progresses, it will go toward making surveillance not only faster but also more meaningful and easier to manage.

VII. FUTURE SCOPE

There are several useful directions in which this system can be scaled up. This includes the very important improvement of adding crowd density measurement and queue congestion detection, helpful in many places where overcrowding could be a concern, such as public events, transportation stations, or campus areas. Another planned addition to the feature set is intrusion detection, where the system would alert security staff if someone enters a restricted zone.

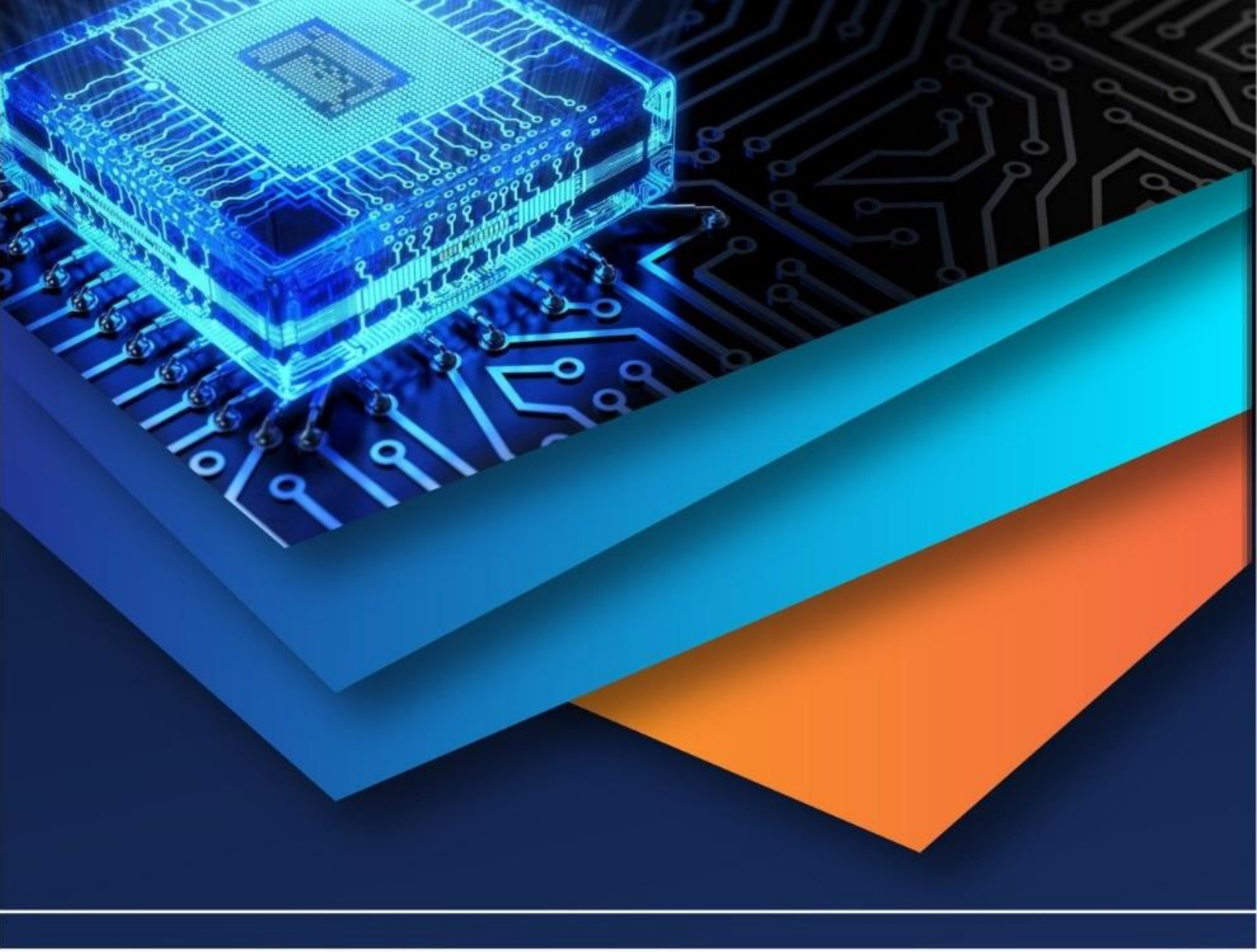
Other extensions may involve the development of face recognition so that the system can tell the identity of a person, whether known or unknown, when needed. It could also provide helmet detection in traffic or industrial use. This can be adapted to real-time video input; therefore, live monitoring can be possible rather than just reviewing recorded footage.

In general, the project will evolve from a basic analysis tool to a complete intelligent surveillance solution capable of supporting safer environments and more responsive security operations.

REFERENCES

[1] B. San Miguel et al., “DiVA: A Distributed Video Analysis Framework,” in Proc. IEEE Int. Conf. on Advanced Video and Signal- Based Surveillance (AVSS), 2007.
[2] M. S. Uddin et al., “SIAT: A Smart Intelligent Video Surveillance System,” IEEE Access, vol. 7, pp. xxxx–xxxx, 2019.
[3] M. Ovseník et al., “A Review of Intelligent Surveillance Systems,” Acta Electrotechnica et Informatica, vol. 10, no. 4, 2010.
[4] S. Hegde et al., “Smart Video Surveillance System Using Deep Learning,” Int. J. Eng. Res. Technol. (IJERT), 2024.
[5] A. Wani et al., “Video Surveillance Using Computer Vision,” Int. J. Comput. Appl. (IJCA), 2022.
[6] B. San Miguel et al., “Visual Surveillance Review,” EURASIP J. Adv. Signal Process., 2007.
[7] T. Held et al., “Video Analytics for Surveillance Applications,” Int. J. Sci. Res. Methodol., 2012.
[8] D. Conte et al., “Advanced Image Analysis for Surveillance,” in Proc. IEEE Int. Conf. on Image Analysis and Processing, 2005.
[9] L. Li et al., “Foreground Object Detection,” IEEE Trans. Image Process., vol. 13, no. 10, pp. 1414–1425, 2004.
[10] S. Rai et al., “Smart Surveillance Using Big Data,” in Proc. IEEE SUSGOD, 2016.
[11] M. Elarbi-Boudihir, “Video Surveillance Systems,” Procedia Comput. Sci., vol. 19, pp. 122–129, 2013.
[12] D. Conte, “Advanced Techniques for Video Surveillance,” Ph.D. dissertation, 2006.
[13] A. Duque et al., “Robust Surveillance Systems,” in Proc. IEEE AVSS, 2006.
[14] A. Bandi et al., “Intelligent Surveillance Using AI,” Int. J. Res. Eng. Technol. (IJRET), 2022.
[15] S. Dabbara, “Smart Surveillance Architecture,” Int. J. Adv. Comput. Sci. Appl. (IJACSA), 2021.
[16] S. Dhokate, “IoT-Based Surveillance System,” in Proc. IEEE ICECA, 2016.
[17] S. Kardile, “AI-Based Monitoring Systems,” Int. J. Innov. Res. Comput. Commun. Eng., 2022.
[18] C. Akoma, “Security Surveillance Review,” Int. J. Comput. Appl., 2012.
[19] R. Gautam, “Forensic Video Analysis,” J. Inf. Fusion Syst., 2019.
[20] O. Olayemi, “Video Surveillance with AI,” Int. J. Comput. Appl., 2023.
[21] M. Shidik, “Deep Learning Surveillance Models,” Int. J. Adv. Comput. Sci. Appl., 2019.
[22] S. Ameer et al., “AI-Enabled Smart Surveillance,” IEEE Access, vol. 11, 2023.

[23] F. Shah et al., "Computer Vision-Based Surveillance," Int. J. Eng. Res. Comput. Sci. Eng., 2023.
[24] Eagle Eye Networks, "Cloud Video Surveillance Report," 2023.
[25] C. Devasena, "Video Monitoring Systems," Int. J. Comput. Sci. Eng., 2011.
[26] A. Kumar et al., "AI Surveillance Framework," Int. J. Artif. Intell. Appl., 2023.
[27] D. Drako et al., "Smart Cities Surveillance," in Proc. IEEE GSCC, 2022.
[28] H. Nikouei et al., "Edge Computing for Surveillance," in Proc. IEEE ISC2, 2018.
[29] Y. Chen et al., "IoT-Based Video Surveillance," IEEE Internet Things J., vol. 9, no. 5, 2022.
[30] Y. Miao et al., "Intelligent Video Analysis," Sensors, vol. 23, 2023.
[31] L. Montero et al., "Video Compression for Surveillance," J. Vis. Commun. Image Represent., 2023.
[32] A. Ferone et al., "Multimedia Surveillance Systems," IEEE Trans. Multimedia, 2025.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY