International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 13 Issue: 07 | Jul 2026 www.irjet.net p-ISSN: 2395-0072

A YOLOv26 Based Early Warning System for Weapon Detection in Public-Safety Surveillance

Harsh Chourasia[1], Dr. Devendra Singh Rathore[2], Dr. Vivek Richhariya[3]

[1]*Research Scholar, Department of CSE, Lakshmi Narain College of Technology, Bhopal, India*
[2]*Associate Professor, Department of CSE, Lakshmi Narain College of Technology, Bhopal, India*
[3]*Professor, Department of CSE, Lakshmi Narain College of Technology, Bhopal, India*

---***---

Abstract - *Real-time weapon detection in surveillance video is essential for proactive public-safety monitoring but still it is challenging due to small object size, occlusions, illumination variations, and strict latency constraints in operational CCTV deployments. This paper proposes an end-to-end weapon detection framework based on YOLOv26 that can effectively work on low-power devices. The pipeline integrates (i) surveillance-aware preprocessing and augmentation to improve robustness under domain shift, and (ii) a YOLOv26 detector consisting of a backbone feature extractor, multi-scale neck fusion, and an end-to-end detection head. Training is stabilized using Small-Target-Aware Label Assignment (STAL) and ProgLoss. This improves the learning consistency for small and partially occluded weapons and usage of NMS-free end-to-end inference reduces post-processing overhead and supports real-time operation. Experimental evaluation demonstrates that YOLOv26 achieves the overall trade-off with mAP@0.5 of 0.7424. These results indicate that the proposed YOLOv26-based framework improves detection reliability and runtime efficiency for practical weapon detection in surveillance streams.*

Key Words: Weapon Detection, YOLO, Smart Surveillance, Deep Learning, Object Detection, Real-Time Systems, Anchor-Free Detection, Computer Vision.

1. INTRODUCTION

Around the world, armed violence and crime involving weapons continue to pose a serious threat to critical infrastructure, public safety, and economic stability. According to global crime statistics, a significant amount of homicides and other violent crimes that occur each year involve firearms [1]. In order to automatically identify threats, surveillance cameras are now widely used in public areas such as schools, transit stations, and cities. The majority of surveillance systems remain dependent on continuous video feed for monitoring, but this strategy has certain drawbacks. Operators grow exhausted, become distracted, and react more slowly over time. Additionally, prior studies have shown that extended manual monitoring sessions can significantly impair anomaly detection [2]. As a result, computer vision-powered smarter surveillance systems are becoming essential for early threat detection. Specifically, deep learning has changed the way that objects are detected. Despite their high accuracy, two-stage, region-based techniques such as R-CNN and Faster R-CNN can be slow because they generate region proposals before making predictions [3]. Single-stage detectors, like the You Only Look Once (YOLO), on the other hand, approach detection as a direct prediction problem, which allows them to operate much more quickly while maintaining competitive accuracy [4]. By adding multi-scale detection, the Darknet-53 backbone, and residual connections, YOLOv3 enhanced performance [5]. On top of that, YOLOv4 and YOLOv5 improved results even more by using more efficient feature aggregation techniques and better training methods [6]. Despite all of these improvements, anchor-based detection still has some real-world limitations. It is frequently less dependable when detecting small objects, necessitates manual anchor-box tuning, increases computational expense, and can be sensitive to changes in the dataset. YOLOv26 switches to an anchor-free detection strategy in order to solve these problems. Additionally, it presents enhanced IoU-based loss optimization, C2f modules, and a decoupled head architecture. When combined, these modifications improve small-object detection, streamline the architecture overall, and make gradients flow more naturally during training. Although earlier research has already investigated weapon detection using models such as YOLOv3 and Faster R-CNN [7][8], the performance of more recent YOLO versions for real-time weapon detection in the real-world surveillance domain has received comparatively little attention. In order to fill this research gap, this paper suggests an intelligent surveillance framework for firearm detection based on YOLOv26. The primary contributions are:

- Development of a YOLOv26-based anchor-free weapon detection architecture.
- Mathematical modelling of the detection and optimization process.
- Design of a scalable smart surveillance pipeline.
- Formulation of detection loss optimization for enhanced precision.

2. LITERATURE REVIEW

The shift to Convolutional Neural Networks (CNNs) was a turning point because CNNs can learn useful features automatically. Models based on AlexNet and VGG were later used for firearm classification tasks with improved performance [10]. For detection and localization, Faster R-CNN introduced region proposal networks to generate bounding boxes more accurately [3]. However, because it is a two-stage approach, it typically needs more computation. To overcome these speed limitations, Redmon et al. [4] introduced YOLO, which reframed detection as a single unified process and enabled real-time object detection. Later, YOLOv3 improved this further with multi-scale prediction using the Darknet-53 backbone [5]. Since then, many researchers have explored YOLOv3 for weapon detection. For example, Grega et al. [7] proposed a CCTV-based firearm detection system using CNNs. Warsi et al. [8] compared YOLOv3 with Faster R-CNN for handgun detection and found that YOLOv3 offered much better speed. Pang et al. [11] applied YOLOv3 for concealed weapon detection using millimeter-wave images. Anchor-free detectors eliminate predefined anchor boxes, simplifying training and improving generalization [12]. Sun and Songsuwankit [21] propose a two-tier AI surveillance framework in which YOLO-based detection runs at the edge to perform fast weapon pre-screening, while more computationally intensive analysis and storage are offloaded to the cloud. Their results show that such an edge–cloud collaboration reduces bandwidth usage and improves end-to-end latency without sacrificing detection performance. Román et al. [22] review artificial intelligence in security operations and describe IoT-based weapon detection systems in which YOLO serves as the primary object detection engine deployed on embedded devices and smart cameras. Researchers have also started building hybrid systems that combine YOLO with models designed to capture time-based patterns and detect unusual activity. For instance, Beca et al. [23] propose an anomaly-detection framework that uses 3D CNNs to learn spatio-temporal cues while using a YOLO-based detector to identify objects in surveillance footage. In their setup, cues like weapon presence and abnormal crowd behavior contribute to an overall anomaly score, helping flag high-risk situations earlier. This reflects a broader shift: weapon detection is increasingly treated not as a standalone task, but as one important signal inside larger situation-awareness and anomaly-detection pipelines. More recently, YOLOv8 and other modern YOLO variants have been explored specifically for weapon detection. Govil et al. [24] present a YOLOv8-based method called Detection Dynamics for real-time weapon identification in surveillance environments. Their approach pairs wavelet-based preprocessing with YOLOv8 to strengthen edge details and improve firearm localization in low-contrast, noisy footage. Their results suggest that YOLOv8 performs better than earlier YOLO versions in both mAP and recall for small weapons, while still running in real time. Gomulka [25] looks at efficient weapon detection by comparing convolutional and transformer-based models against YOLO-style baselines using CCTV-like data. While transformer models can improve robustness in some settings, the study notes that YOLO-style one-stage detectors often remain the better choice when real-time speed and limited compute are non-negotiable. Alongside YOLO-based approaches, other work explores different backbones and classification strategies for weapon recognition. For example, Elgaier and Abushaala [26] propose using ResNet-50 feature extraction followed by traditional machine-learning classifiers, and they compare their method with YOLO and SSD detectors. Although their approach performs well on controlled datasets, YOLO-based pipelines generally continue to offer a stronger speed–accuracy trade-off for live, streaming surveillance use cases.

3. METHODOLOGY USED

This section explains the proposed YOLO26-based approach for real-time weapon detection in smart surveillance. The objective is to reliably detect, locate, and classify weapons in continuous CCTV or IP-camera video streams, while meeting practical requirements such as low latency, high recall, and easy deployment on edge devices. Building on the one-stage detection idea introduced by YOLO [4], [5], the proposed method moves beyond earlier YOLOv8-based modeling and adopts YOLO26 instead. YOLO26 introduces key architectural upgrades and performance benchmarking aimed at improving real-time object detection across multiple domains [27]. Overall, the system is designed as a complete end-to-end pipeline, as illustrated in Fig. 1.

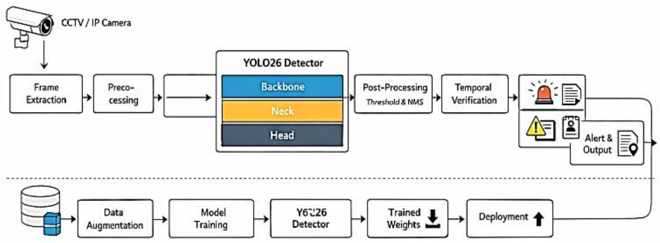

Fig. 1. Proposed Model

3.1 System Overview and Problem Formulation

Let the surveillance input be a video stream represented by a sequence of frames:

$$v = \{F_t\}_{t=1}^{T}, \quad F_t \epsilon R^{H \times W \times 3} \tag{1}$$

For each frame F_t, the detector outputs a set of predictions:

$$D_\theta(F_t,) = \{b_i, c_i, s_i\}_{i=1}^{N}, \tag{2}$$

where $b_i = (x_i, y_i, w_i, h_i)$ denotes the predicted bounding box (center and size), c_i is the predicted class label (weapon type), and $s_i \in [0,1]$ is the confidence score. The learnable network parameters are θ. The design objective is to achieve high detection performance under real-time constraints:

$$FPS \geq \tau_{FPS}, T_{inf} \leq \tau_{latency} \quad (3)$$

where T_{inf} is inference time per frame. Weapon detection applications typically prioritize high recall, while keeping false alarms within an operationally acceptable rate.

3.2 Dataset Construction and Annotation

Weapon detection is highly sensitive to dataset diversity, particularly due to small object size in surveillance footage, occlusions, and background clutter. The dataset is constructed to include multiple weapon categories as well as negative samples. Each image is annotated with bounding boxes in pixel coordinates.

3.3 Preprocessing and Data Augmentation

Surveillance environments exhibit strong domain shift: varying illumination, camera compression, motion blur, and scale differences. Therefore, preprocessing and augmentation are applied to increase robustness. Resizing and normalization: Each input frame is resized to the YOLO26 operating resolution (640×640) and normalized as F_t.

Augmentation: Let A(·) denote a stochastic augmentation function. The augmented image is:

$$\widehat{F_t} = A(F_t) \quad (4)$$

where A, includes photometric (brightness/contrast/gamma), geometric (flip/scale/translation), and contextual augmentations (random crop; mosaic-style composition where supported). Bounding boxes are transformed accordingly. Augmentations are tuned to preserve realism; excessive geometric distortion may harm generalization.

3.4 YOLO26 Model Adaptation and Initialization

The proposed detector uses YOLO26 as the backbone detection framework. YOLO26 is presented as a real-time object detection architecture with key architectural enhancements and benchmarking emphasis, supporting deployment in time-sensitive applications such as surveillance [27]. In this work, YOLO26 is adapted for weapon detection via transfer learning and domain-specific fine-tuning. Let the YOLO26 architecture be represented as three components (Fig 2):

- Feature extractor (Backbone)
- Multi-scale feature fusion module (Neck)
- Detection head for class and box prediction

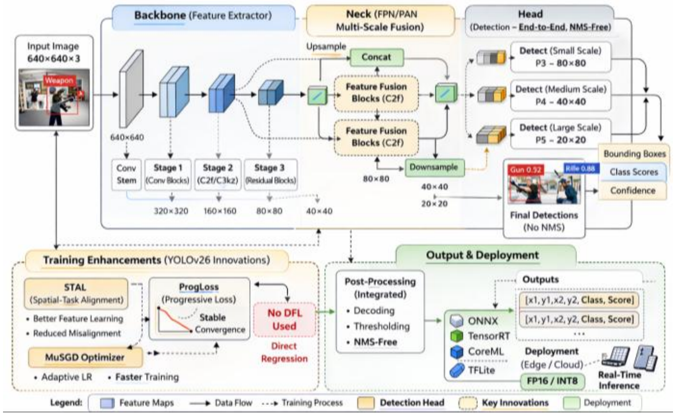

Fig. 2. Yolo26 Network Architecture

The feature extractor (backbone) is the first stage of the detector. Its job is to turn each input frame into a set of meaningful visual features. After a surveillance image is resized and normalized, the backbone applies a sequence of convolutional layers that gradually reduce the image resolution while increasing the amount of semantic information captured. The early layers keep fine details like edges and textures important for spotting small objects such as handguns while deeper layers learn higher-level patterns such as weapon shape, grip–barrel structure, and surrounding context. In the end, the backbone outputs multiple feature maps at different depths, each representing the scene at a different scale.
Next, the multi-scale feature fusion module (neck) combines these feature maps so the detector can recognize weapons at different distances and sizes. In surveillance footage, a weapon might be only a few pixels wide when someone is far from the camera, but it may fill a large part of the frame when close. If detection relied on just one feature-map resolution, small or distant weapons would be easy to miss. The neck addresses this by blending high-resolution maps (rich in spatial detail) with lower-resolution maps (rich in semantic meaning), often using both top-down and bottom-up fusion paths. This helps preserve fine weapon cues, strengthens small-object detection, and improves reliability in cluttered or crowded scenes.

Finally, the detection head generates the actual outputs: weapon class labels and bounding-box coordinates for likely weapon locations. Using the fused features from the neck, the head predicts (1) class probabilities (e.g., handgun vs. rifle vs. background) and (2) bounding box parameters (center location and size), along with a confidence score indicating how likely it is that a weapon is present. During inference, the model typically produces multiple overlapping candidate boxes, which are then refined using a confidence threshold and Non-Maximum Suppression (NMS) to remove duplicates and keep only the most reliable detections for alerting and logging. To make training practical, the model parameters θ are initialized with pretrained weights learned from large, general-purpose datasets. This speeds up convergence and reduces the need for extremely large weapon-specific datasets. Fine-tuning then adapts the model to the surveillance domain often by training the neck and head first (sometimes freezing parts of the backbone for the first few epochs), and then gradually unfreezing deeper backbone layers so the features can better capture weapon-specific patterns.

3.5 Loss Function

Training minimizes a composite loss that balances localization and classification. A general form is:

$$L_{total} = \lambda_{box} L_{box} + \lambda_{cls} L_{cls} + \lambda_{obj} L_{obj} \tag{5}$$

Where, L_{box} is the bounding box regression loss which is IoU-based losses are used to directly optimize overlap and localization quality. L_{cls} is classification loss for C weapon classes. Here cross-entropy is used. L_{obj} is the objectness/confidence loss that reduces background false positives.

3.6 Optimization strategy

The model is trained using SGD or AdamW with learning-rate scheduling. Regularization reduces overfitting. Mixed-precision training can be used to accelerate training and inference on modern GPUs. During training, hyperparameters $\lambda_{box}, \lambda_{cls}, \lambda_{obj}$ are tuned to favor recall in safety-critical settings.

At deployment time, YOLO26 processes each frame and outputs candidate detections. First, low-confidence detections are removed using a threshold $s_i > \tau_s$. Next, overlapping predictions are suppressed using Non-Maximum Suppression (NMS). This stabilizes outputs by removing duplicates. Therefore, this proposed model will help in surveillance by removing the transient false positives may occur due to motion blur and compression artifacts. Therefore, a lightweight temporal verification can be applied. This improves reliability without adding significant computational overhead.

4. RESULTS AND DISCUSSION

In this section, descriptions of the tools used are presented. For experimental implementation, this study has adopted python. The Entire model is trained for 25 epochs on GPU. The dataset used in this study was constructed by integrating two publicly available Roboflow Universe datasets, namely FIX_VW_Dataset [28] and FWD AI [33]. The original datasets contained multiple weapon categories with different annotation schemes and class names. To ensure consistency with the firearm categorization adopted in previous work, the annotations were harmonized into four firearm classes: “Pistol, Subfusil, Shotgun, and Fusil”. The merged dataset contains surveillance-style images captured under diverse environmental conditions, including variations in illumination, viewing angle, background clutter, object scale, and partial occlusion, thereby improving the diversity and robustness of the training data. The resulting dataset was divided into training, validation, and testing subsets to ensure an unbiased evaluation of the proposed weapon detection framework. The work presented the result using following parameters:

Mean average Precision (mAP): To evaluate mAP, first precision need to be evaluated as:

$$Precision = \frac{(TP)}{(TP + FP)} \tag{6}$$

Then, mAP is mathematically represented as:

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{7}$$

$$"Precision = \frac{(True\_Positive)}{(True\_Positive + False\_Positive)}" \tag{8}$$

$$"Recall = \frac{(True\_Positive)}{(True\_Positive + False\_Negative)}" \tag{9}$$

Fig. 3 shows the class-wise precision–recall (PR) curves of the proposed YOLOv26 model at IoU = 0.5. The overall curve yields mAP@0.5 = 0.827. Importantly, PR curves reflect performance across all confidence thresholds, so a class can achieve high AP even if its recall at a single operating threshold is moderate. Fig. 4 shows the confusion matrix. Fig 5 presents the visual representation of testing images.

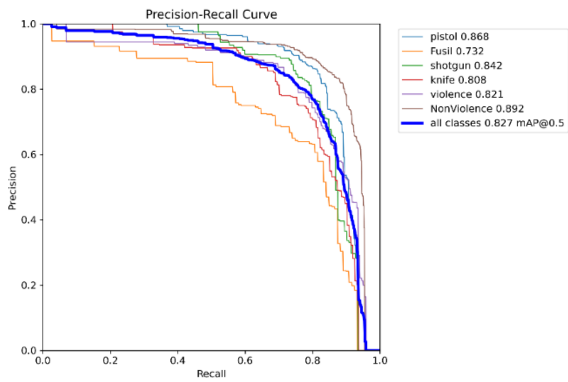

Fig. 3. Precision-Recall Analysis for YOLOv26

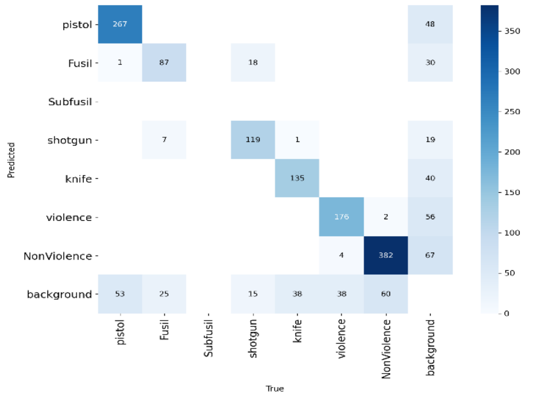

Fig. 4. Confusion Matrix for YOLOv26

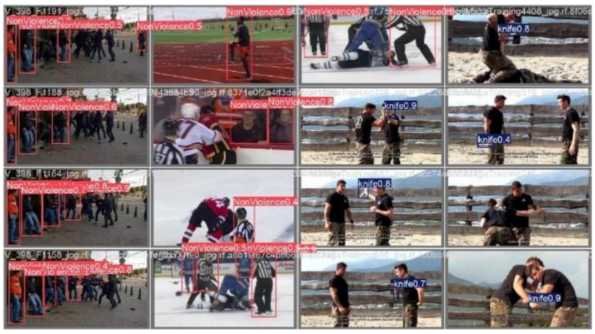

Fig. 5. Visual Representation of Prediction

Table 1 presents a feature-wise comparison of the proposed YOLOv26 framework against recent YOLO-based weapon detection approaches [29]–[32] across key architectural and deployment characteristics. While prior works progressively adopt anchor-free detection and data augmentation, all of them remain dependent on NMS post-processing, use no specialized loss formulation, and are limited to three or four weapon classes. In contrast, the proposed YOLOv26 is the only method that is simultaneously anchor-free and NMS-free (end-to-end), incorporates a custom loss with small-target-aware label assignment, and supports a broader set of six weapon classes while remaining suitable for low-power deployment. Table 2 shows proposed model have achieved the best overall performance with an average F1-score of 74.90%, recall of 70.80%, and precision of 79.50%, outperforming EfficientNet0 and YOLO v3 in most overall metrics. It also showed strong class-wise results, especially for Pistol with 72.4% F1-score and Shotgun with 92.5% precision. Although YOLO v3 performed slightly better for Shotgun F1-score, the proposed model achieved comparable or better accuracy while requiring only 1.32 hours of training time, much lower than EfficientNet0 and YOLO v3. Therefore, the proposed approach offers the best balance between detection accuracy and computational efficiency.

Table 1. Comparative State-of-Art

Model / Backbone	Anchor-Free	NMS-Free	Weapon Classes	Data Augmentation	Custom Loss	Low-Power
YOLOv3 [29]	✗	✗	3	✗	✗	✓
YOLOv3 + EfficientDet-D0 [30]	✗	✗	4	✓	✗	△
YOLOv3 + YOLOv8 [31]	✓	✗	3	✓	✗	✗
YOLOv8–v12 [32]	✓	✗	3	✓	✗	△
YOLOv26 (Proposed)	**✓**	**✓**	**6**	**✓**	**✓**	**✓**

Table 2. Comparative Performance Analysis

Models	Sets	Avg F1	Avg Recall	Avg Precision	Metric	Pistol	Shotgun	Fusil	Training time (hours)
Efficient 0	Set 1	36%	25%	74%	F1	18%	43%	37%	4.57
					REC	10%	30%	26%	
					PRE	73%	76%	63%	
Yolo v3		63%	61%	61%	F1	53%	76%	55%	4.99
					REC	47%	84%	47%	
					PRE	55%	69%	65%	
Proposed		74.90 %	70.80%	79.50%	F1	72.4%	55.9%	71.1 %	1.32
					REC	66.5%	61.5%	57.8 %	
					PRE	79.5%	51.3%	92.5	

								%	
Efficient 0	Set2	53%	47%	69%	F1	67%	27%	32%	2.54
					REC	58%	18%	23%	
					PRE	80%	59%	53%	
Yolo v3		77%	74%	80%	F1	84%	67%	64%	3.60
					REC	81%	62%	57%	
					PRE	88%	73%	74%	
Proposed		77%	72%	82%	F1	77%	73%	52%	1.03
					REC	67%	61%	62%	
					PRE	91%	92%	45%	

5. CONCLUSION

This paper presented a YOLOv26-based real-time weapon detection framework for smart surveillance systems, targeting practical deployment challenges in CCTV environments, including small object appearance, partial occlusion, background clutter, and strict low-latency requirements. The proposed model integrates an anchor-free and NMS-free end-to-end detection architecture with Small-Target-Aware Label Assignment (STAL) and ProgLoss to enhance localization accuracy, classification confidence, and computational efficiency. Experimental results demonstrate that YOLOv26 achieves strong detection performance, with a precision of 0.8506 and mAP@0.5 of 0.7424, while maintaining suitability for real-time surveillance applications. Although the recall of 0.6453 indicates that some difficult or partially occluded weapon instances remain challenging, the model's higher precision reflects its ability to reduce false alarms, which is critical in security-sensitive monitoring systems. Overall, the proposed framework offers an effective balance between accuracy and efficiency, making it a promising solution for automated weapon detection in smart surveillance scenarios. Future work will focus on improving recall for small and occluded weapons, enhancing robustness under low-light and cross-camera domain shifts, integrating real-time alert and notification mechanisms, and extending the system toward multi-threat detection with unified risk scoring across distributed camera networks.

REFERENCES

1) United Nations Office on Drugs and Crime (UNODC), Global Study on Homicide 2019. Vienna, Austria: United Nations Publications, 2019. [Online]. Available: https://www.unodc.org

2) S. A. Velastin, B. A. Boghossian, and M. A. Vicencio-Silva, "A motion-based image processing system for detecting potentially dangerous situations in underground railway stations," Transportation Research Part C: Emerging Technologies, vol. 14, no. 2, pp. 96–113, 2006.

3) S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137–1149, 2017.

4) J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779–788.

5) J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," arXiv preprint arXiv:1804.02767, 2018.

6) Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," arXiv preprint arXiv:2004.10934, 2020.

7) M. Grega, A. Matiolanski, P. Guzik, and M. Leszczuk, "Automated detection of firearms and knives in a CCTV image," Sensors, vol. 16, no. 1, p. 47, 2016.

8) Warsi, M. Abdullah, M. N. Husen, M. Yahya, S. Khan, and N. Jawaid, "Gun detection system using YOLOv3," in Proc. IEEE Int. Conf. Smart Instrumentation, Measurement and Application (ICSIMA), Kuala Lumpur, Malaysia, 2019, pp. 1–4.

9) S. B. Kibria and M. S. Hasan, "An analysis of feature extraction and classification algorithms for dangerous object detection," in Proc. 2nd Int. Conf. Electrical & Electronic Engineering (ICEEE), 2017, pp. 1–4.

10) Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in Advances in Neural Information Processing Systems (NeurIPS), vol. 25, 2012.

11) L. Pang, H. Liu, Y. Chen, and J. Miao, "Real-time concealed object detection from passive millimeter wave images based on the YOLOv3 algorithm," Sensors, vol. 20, no. 6, p. 1678, 2020.

12) Z. Tian, C. Shen, H. Chen, and T. He, “FCOS: Fully convolutional one-stage object detection,” in Proc. IEEE Int. Conf. Computer Vision (ICCV), 2019, pp. 9627–9636.

13) G. Jocher et al., “Ultralytics YOLOv8,” Ultralytics, 2023. [Online]. Available: https://github.com/ultralytics/ultralytics

14) K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.

15) T.-Y. Lin et al., “Feature pyramid networks for object detection,” in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2117–2125.

16) W. Liu et al., “SSD: Single shot multibox detector,” in Proc. European Conf. Computer Vision (ECCV), 2016, pp. 21–37.

17) K. Bochkovskiy and G. Jocher, “Ultralytics YOLO architecture documentation,” Ultralytics, 2023.

18) H. Law and J. Deng, “CornerNet: Detecting objects as paired keypoints,” in Proc. European Conf. Computer Vision (ECCV), 2018, pp. 734–750.

19) R. Girshick, “Fast R-CNN,” in Proc. IEEE Int. Conf. Computer Vision (ICCV), 2015, pp. 1440–1448.

20) K. He et al., “Mask R-CNN,” in Proc. IEEE Int. Conf. Computer Vision (ICCV), 2017, pp. 2961–2969.

21) S. Sun and K. Songsuwankit, “Two-Tier AI Surveillance: Enhancing Weapon Detection Through Edge and Cloud Collaboration,” ICT for Intelligent Systems: Proceedings of ICTIS 2025, Volume 10: 439.

22) Román, Jorge J., et al. "Artificial Intelligence in Security Operations: A Brief Review of Current Applications." Intelligent Systems Conference. Cham: Springer Nature Switzerland, 2025.

23) V. Beca, B. A. Zamora, C. M. Paredes, and S. Dinas, “Anomaly Detection System Based on 3D Convolutional Neural Networks and YOLO on Surveillance Videos,” in Proc. Workshop on, Springer, 2025, ch. 27. doi: 10.1007/978-3-032-08203-9_27. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-032-08203-9_27

24) Govil, Abhi, et al. "Detection Dynamics: A YOLOv8 Approach to Real-Time Weapon Identification." International Conference on Communication and Computational Technologies. Singapore: Springer Nature Singapore, 2025.

25) K. Gomulka, “Efficient weapon detection using convolutional and transformer-based Deep Learning Models,” International Journal of Electronics and Telecommunications, 2026. doi: 10.24425-ijet.2026.157893. [Online]. Available: https://ijet.pl/index.php/ijet/article/view/10.24425-ijet.2026.157893

26) Elgaier, Areej Mokhtar, and Ahmed Mohamed Abushaala. "Weapons Detection and Classification by Using ResNet50 with Machine Learning Classifiers." International Conference on AI: Current Research, Industry Trends, and Innovations. Cham: Springer Nature Switzerland, 2025.

27) R. Sapkota, R. H. Cheppally, A. Sharda, et al., “YOLO26: Key Architectural Enhancements and Performance Benchmarking for Real-Time Object Detection,” arXiv preprint, 2025. [Online]. Available: https://arxiv.org/abs/2509.25164.

28) Train1-DLPBB, "FIX_VW_Dataset," Roboflow Universe, Version 1, 2024. [Online]. Available: https://universe.roboflow.com/train1-dlpbb/fix_vw_dataset-mevob/dataset/1. [Accessed: Jul. 2026].

29) Narejo, Sanam, et al. "Weapon detection using YOLO V3 for smart surveillance system." Mathematical Problems in Engineering 2021.1 (2021): 9975700.

30) Ramon, Anthony Ortiz, and Luis Barba Guaman. "Detection of weapons using Efficient Net and Yolo v3." 2021 IEEE Latin American Conference on Computational Intelligence (LA-CCI). IEEE, 2021.

31) Arivalagan, Divya, et al. "Weapon detection system for security and surveillance using yolov8." 2024 2nd International Conference on Advancement in Computation & Computer Technologies (InCACCT). IEEE, 2024.

32) Srinivasan, R., and Sharmila Devi Chandariah. "Advanced Weapon Detection System using Multi-Version YOLO Models for Surveillance." 2025 International Conference on Sustainable Communication Networks and Application (ICSCN). IEEE, 2025.

33) Blyke, "FWD AI," Roboflow Universe, Version 4, 2024. [Online]. Available: https://universe.roboflow.com/blyke/fwd-ai-hhwkh/dataset/4. [Accessed: Jul. 2026].