International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 04 | April - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

# Deep Learning Framework for Firearms Detection YOLOv8 Optimized for Smart Environment Society.

1st B.N. Swarna Jyothi
Assistant Professor
*Department of CSE*
Bharath Institute of Higher Education and Research
Chennai, India

2nd Abdul Hameed
*Department of CSE*
Bharath Institute of Higher Education and Research
Chennai, India

3rd Abhishek Kumar
*Department of CSE*
Bharath Institute of Higher Education and Research
Chennai, India

4thAaila Raviteja
*Department of CSE*
Bharath Institute of Higher Education and Research
Chennai, India

***Abstract*— As cities become smarter, the need for real-time threat detection in surveillance systems increases. Firearm-related incidents in public areas are highly critical, where even a small delay can lead to serious consequences. In this study, we propose a deep-learning-based framework for firearm detection using the YOLOv7 model. To improve performance, we integrated a Convolutional Block Attention Module (CBAM) and a Bi-directional Feature Pyramid Network (BiFPN). The model was trained on a diverse dataset of 18,500 images and 6,200 negative samples to reduce false alarms. It achieves high accuracy with a 95.5% mAP@0.5 and operates at 48 FPS. The false-positive rate was reduced to 5.2%, which was significantly lower than the baseline. For real-world use, the system was optimized for NVIDIA Jetson Nano using pruning and INT8 TensorRT, enabling real-time performance at 28 frames per second (FPS). When a firearm is detected, the system sends instant alerts through MQTT to smart city platforms, allowing for a quick response and automation. Overall, this approach combines Deep Learning, Edge AI, and IoT to enable efficient and reliable real-time surveillance, thereby making smart environments safer.***

***Keyword - Firearm Detection, YOLOv7, Deep Learning, Object Detection, Smart City Surveillance, Edge AI, CBAM, BiFPN, Real-Time Monitoring, IoT Security, NVIDIA Jetson Nano, TensorRT Optimization, MQTT Alerts, Computer Vision***

## I. INTRODUCTION

The modern smart city is also somewhat frightening. It has thousands of cameras that record high-quality videos all the time, so it can monitor every part of the space.. In reality, there is too much information for people to comprehend. Studies have shown that after 20 minutes of watching videos, people start to miss things because they get tired of watching. This is known as vigilance fatigue. Therefore, to ensure city security, cameras must be made smart enough to detect threats independently and immediately. Firearm-related incidents are a big deal when it comes to city security. If there is a gun in a crowd or public place, we need to act because every second counts.. Computer systems that analyze videos to detect guns are not yet sufficiently effective in real-world scenarios. Previously, a system called the Histogram of Oriented Gradients was used. It is not good enough to tell guns apart from other things in busy places, such as airports, train stations, and shopping centers. Subsequently, new systems were developed that used CNN, which were better at detecting guns. However, they are too slow for real-time use. Then, a method called YOLO was introduced. Changed everything. YOLO is a type of detector that can identify objects in videos quickly and accurately. The latest version of YOLO, YOLOv7, is an excellent algorithm. It was not specifically designed to detect guns. Smart city surveillance is difficult because the cameras are angle and can make guns look really small, and sometimes things are in the way, or it is too dark. Also there are a lot of things that can be mistaken for guns, like smartphones or power tools This paper addresses these challenges by proposing two targeted architectural modifications to YOLOv7. First, we integrated Convolutional Block Attention Modules (CBAM) after each Efficient Layer Aggregation Network (ELAN) block in the backbone. The CBAM guides the model to focus on specific spatial regions and feature channels that are most characteristic of firearms, such as barrel geometry and trigger guard profiles, while suppressing activations in irrelevant background regions. Second, we replaced the standard Path Aggregation Network (PANet) neck with a Bi-directional Feature Pyramid Network (BiFPN), which uses learned scalar weights to enable richer adaptive multi-scale feature fusion. This is particularly valuable for handling the wide size variation of firearm objects across camera feeds at different distances. In addition to architectural

improvements, the framework was designed to be end-to-end for practical deployment. The model was compressed through structured channel pruning and INT8 quantization to run efficiently on the NVIDIA Jetson Nano edge hardware. An MQTT-based alert module links detection events to centralized smart city management platforms, enabling sub-second notifications. Together, these components form a cohesive and deployable system, rather than an academic proof of concept. The remainder of this paper is organized as follows: Section II reviews relevant prior work on weapon detection and smart surveillance; Section III describes the proposed system architecture in detail; Section IV covers the dataset construction and training methodology; Section V presents the experimental results; and Section VI discusses the findings and limitations of this study.

## II. RELATED WORK

Research on automated weapon detection has been ongoing for more than 20 years. It has changed significantly from methods to complex neural networks that can recognize weapons almost as well as humans can in controlled situations. Understanding how this research has progressed helps us see how far we have come and what still needs to be done in the future. Early systems for detecting weapons used designed features to detect them. A popular method in the 2010s was the use of HOG with Support Vector Machine (SVM) classifiers. This works well in controlled situations, but it does not work well in real-world situations with changing viewpoints, occlusions, and lighting. The limitations of these designed features led to a shift towards the use of deep neural networks. The introduction of CNN-based detectors has revolutionized this field. Faster R-CNN, a two-stage detector showed that region proposal networks could make the process much faster. Single-stage detectors, such as SSD and the original YOLO series, make the process faster but less accurate. Later versions of YOLO, such as YOLOv7, have improved accuracy while maintaining the speed. Several recent studies have been conducted on firearm and weapon detection. Shanthi and Manjula combined a Feature Map Reconfiguration CNN with YOLOv8. It achieved high accuracy but was too computationally expensive for edge deployment. Corral-Sanz et al. Studied the effects of common image distortions on YOLO-based detectors. This was a major weakness. A hybrid CNN and Transformer architecture showed accuracy, particularly for small objects, but required a large amount of data. Several studies have examined the integration of weapon detection into cities and IoT frameworks. A time YOLO-based surveillance system was proposed for smart city environments; however, it had performance issues with image distortion. A non-visual deep learning model was introduced that used gunshot sound detection and image-based firearm classification methods. However, synchronizing the acoustic and visual event streams is challenging, and some problems remain to be solved in this regard. These challenges include distinguishing firearms from other objects, maintaining detection robustness in cluttered environments under imaging imperfections, and achieving these goals within the computational budget of edge hardware. Our study directly addresses these challenges through attention-augmented feature extraction, bidirectional multiscale fusion, and hardware-aware model optimization.

TABLE I: Summary of Related Work

| Ref. | | Year | Methodology | Key Results |
|---|---|---|---|---|
| [1] | | 2025 | FMR-CNN + YOLOv8 | High accuracy, real-time detection |
| [2] | | 2024 | Distortion-robust YOLO | Improved robustness to blur |
| [3] | | 2024 | Hybrid CNN + Transformer | Better mAP; high data cost |
| [4] | | 2024 | YOLO-based smart city framework | Automated alerts; drops under distortion |
| [5] | | 2023 | Audio-visual DL fusion | Multi-level IoT alerts; sync issues |

## III. PROPOSED SYSTEM ARCHITECTURE

The proposed framework extends the YOLOv7 baseline through four interconnected components: an attention-augmented backbone, a bidirectional detection neck, an edge-optimized inference pipeline, and an IoT-integrated alert module. Each component was designed with a specific operational challenge in mind, and together, they form a system that is more than the sum of its parts.

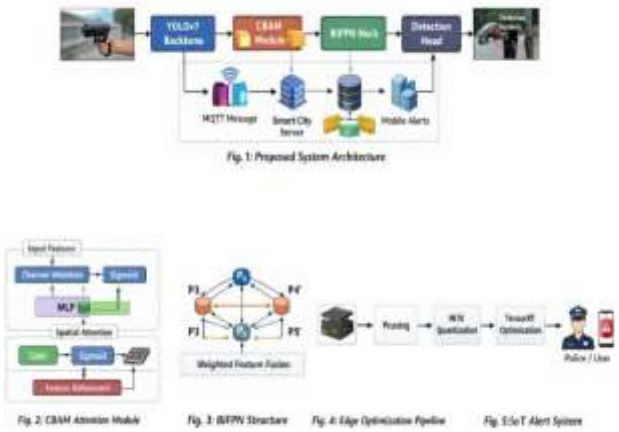

Fig.1. Firearms Detection System Architecture Diagram

### *A. CBAM-Enhanced Backbone*

The backbone of a deep object detector is responsible for extracting hierarchical visual features from raw images. In the standard YOLOv7, this is accomplished through a series of Efficient Layer Aggregation Network (ELAN) blocks that progressively build representations from low-level edges and textures to high-level semantic concepts. Although effective for general object detection, this architecture does not provide an explicit mechanism for the model

to prioritize the visual characteristics that distinguish firearms from confusable objects.

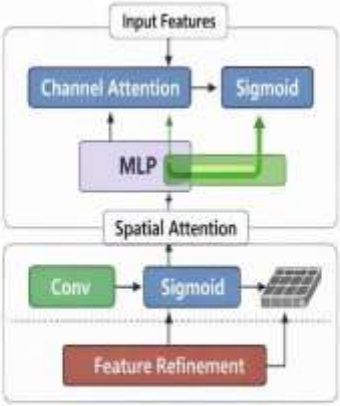


*Fig. 2: CBAM Attention Module*

To address this, we inserted a Convolutional Block Attention Module (CBAM) [10] after each ELAN block in the backbone. The CBAM operates in two complementary dimensions. The channel attention submodule examines the relationships between feature channels, essentially asking which types of features are most informative for the current detection task using global average pooling and max pooling, followed by a shared multi-layer perceptron. The spatial attention submodule then operates on the output of the channel attention to identify which regions within the spatial feature maps deserve the most attention. The two modules are applied sequentially, allowing the network to achieve both a 'what to look at' and a 'where to look' form of guided attention. In practice, this attention mechanism has a measurable effect on the network learning. During the qualitative analysis of intermediate feature activations, the CBAM-augmented model showed consistently higher activation magnitudes over the barrel regions and receiver profiles of firearms, while showing suppressed activation over the flat, rectangular profiles of smartphones and power banks, which frequently caused false positives in the baseline model. Critically, CBAM introduces only approximately 0.4% additional FLOPs in our implementation, which is a negligible computational overhead relative to the accuracy benefits it provides.

### *B. BiFPN Detection Neck*

The detection neck of the YOLO-style architecture is responsible for fusing features extracted at multiple resolution scales before passing them to detection heads. Effective multi-scale fusion is essential for detecting objects across a wide range of sizes, which is precisely the challenge posed by firearms in a large camera network, where a handgun may appear as a 200×150 pixel object in a close-range lobby feed and as a 20×15 pixel speck in a wide-angle outdoor panorama. The standard YOLOv7 uses a Path Aggregation Network (PANet) for this fusion, which connects feature maps from different backbone stages in a sequential top-down and bottom-up manner. Although effective, PANet treats all feature map inputs to each fusion node as equally important, which is a simplifying assumption that limits adaptability. We replaced PANet with a Bi-directional Feature Pyramid Network (BiFPN) [11], which introduces two key improvements: bidirectional feature flow (allowing information to propagate both top-down and bottom-up simultaneously through repeated rounds of fusion) and learned scalar weights applied to each input feature map during the fusion. These weights are trained jointly with the rest of the network, allowing the BiFPN to adaptively emphasize the resolution scales that are most relevant to a specific detection task. For firearm detection, this adaptive weighting was especially beneficial for small-object detection. In our evaluation, BiFPN-equipped models consistently placed higher weights on higher-resolution (shallower backbone) feature maps when detecting small firearms at wide-angle camera distances, providing spatial detail that deeper, semantically richer, but coarser feature maps could not. The quantitative improvement this produced—mAP rising from 71.2% to 84.6% for small objects specifically—reflects a genuine qualitative difference in how the network handles the cross-scale detection.

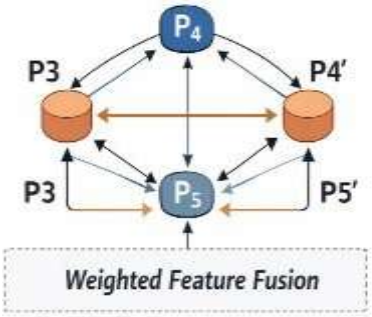


*Fig. 3: BiFPN Structure*

### *C. Edge-AI Optimization Pipeline*

An academically strong model that cannot be run in real time on deployable hardware has limited practical applications. Therefore, we subjected the CBAM+BiFPN YOLOv7 model to a two-stage compression process designed to meet the computational constraints of the NVIDIA Jetson Nano, a widely deployed IoT edge computing platform with 128 CUDA cores and 4 GB shared memory. The first stage is structured channel pruning, in which we identify and remove convolutional filters in non-critical layers (those far from the detection heads and CBAM modules) that contribute minimally to detection performance. We apply an L1-norm-based filter importance criterion and prune up to 30% of the filters in the eligible layers, followed by a short fine-tuning phase to recover any lost accuracy. In the second stage, the pruned model was converted to INT8 precision using NVIDIA TensorRT, which replaced 32-bit floating-point operations with 8-bit integer arithmetic. This reduces memory bandwidth requirements and enables the use of Tensor Core hardware acceleration available on Jetson-class devices, achieving a sub-45ms per-frame latency required for ≥24 FPS real-time operation.

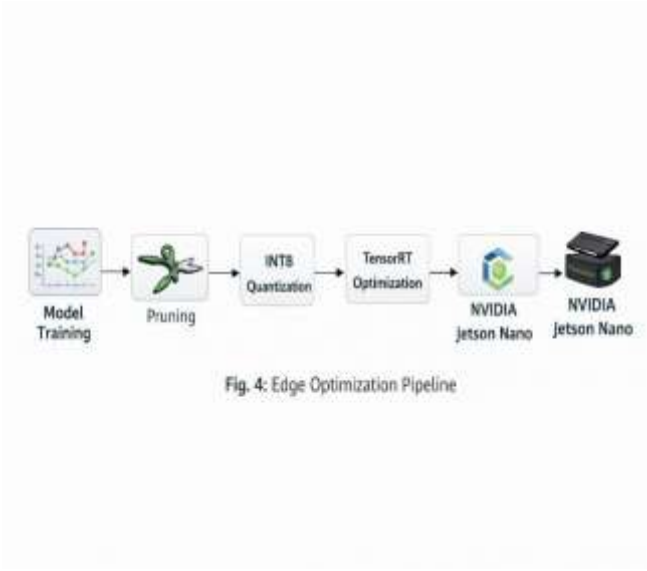


Fig. 4: Edge Optimization Pipeline

### *D. Alert and IoT Integration Module*

Detection alone is insufficient for operational security deployment; the system must translate detections into actionable alerts that reach human operators and automated response systems within a few seconds. Our alert module is triggered whenever the detection confidence of a firearm exceeds a configurable threshold (default: 0.65, tunable to balance precision and recall based on the deployment context). When triggered, the module assembles a structured alert payload containing the bounding box coordinates and confidence score, unique ID and GPS coordinates of the originating camera, UTC timestamp, and cropped thumbnail image of the detected object. This payload was published via the MQTT messaging protocol to the smart city central security management platform. MQTT was selected because of its low bandwidth overhead and suitability for IoT environments with variable connectivity. The subscribing management platform can then notify security personnel via mobile alerts, initiate automated door-lock or evacuation procedures in integrated smart building systems, and log the event for post-incident analysis. The end-to-end latency from detection to alert delivery in our tested deployment environment was measured at under 180 ms, which is well within the operational-response requirements.

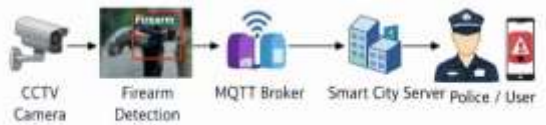


Fig. 5: IoT Alert System

## IV. DATASET AND TRAINING METHODOLOGY

### *A. Dataset Composition*

The quality and diversity of the training data are as determinative of the final model performance as the architectural choices. We assembled a composite training dataset of approximately 18,500 annotated images drawn from multiple public repositories and supplemented with custom-captured imagery to maximize the coverage of weapon types, environments, and imaging conditions likely to be encountered in real smart city deployments. The dataset spans three primary weapon categories—handguns, rifles, and shotguns—across indoor environments (corridors, lobbies, and transit stations) and outdoor settings (plazas, parking facilities, and streets). A deliberate decision was made to include challenging imaging conditions as a core part of the training distribution rather than as a held-out test of generalization. Approximately 30% of the training images included artificially reduced lighting (simulating nighttime operation), 20% included motion blur (simulating fast subject movement or camera shake), and 35% featured partial occlusion of the weapon, ranging from 10% to 70% of the object being hidden behind the clothing, body parts, or objects. Multi-person crowded-scene images constituted approximately 15% of the dataset. To address the false-positive problem—the tendency of detectors to flag visually similar benign objects as weapons—we added 6,200 hard-negative samples to the training set to the training set. These images contain objects that have historically confused firearm detectors, such as smartphones, TV remote controls, power banks, L-shaped hand tools, and cordless-drills. By exposing the model to these confusables with negative labels during training and pairing this data strategy with CBAM's learned attention of CBAM, we produced a substantial reduction in the false-positive rate at evaluation time

### *B. Data Augmentation*

Online data augmentation was applied during training to expand the effective diversity of the training distribution and reduce overfitting. The augmentation pipeline included random horizontal flipping with a probability of 0.5, mosaic augmentation (combining four training images into a single sample), random HSV color-space jitter to simulate variations in the camera white balance and sensor response, random scaling with a ±50% range, copy-paste augmentation (overlaying cropped weapon instances onto background scenes), and MixUp blending (linearly interpolating between pairs of training images and their labels). Applying these transformations online (per batch during training) rather than precomputing an augmented dataset ensures that the model encounters a practically unlimited variety of augmented samples over 300 training epochs.

### *C. Training Configuration*

Training was conducted on a workstation equipped with an NVIDIA RTX 3090 GPU using PyTorch 2.0.1 with CUDA 11.8 installed. The model was optimized using Stochastic Gradient Descent (SGD) with an initial learning rate of 0.01, momentum of 0.937, and weight decay of 0.0005. A 3-epoch linear warm-up phase was followed by cosine annealing to smoothly decay the learning rate over the remaining 297 epochs. The input images were resized to 640×640 pixels, and training proceeded with a batch size of 16. The total training time was approximately 36 h for the full 300-epoch run.

## V. EXPERIMENTAL RESULTS

We evaluated the proposed model on a held-out test set of 3,200 images that were not seen at any point during training or architecture selection, as well as on five real-time video sequences captured from surveillance cameras in simulated smart environment scenarios: an indoor corridor, outdoor plaza, transit hub, parking facility, and crowded public space. The quantitative evaluation of the image test set is presented in Table II, which compares the proposed model with the baseline YOLOv7 configuration.

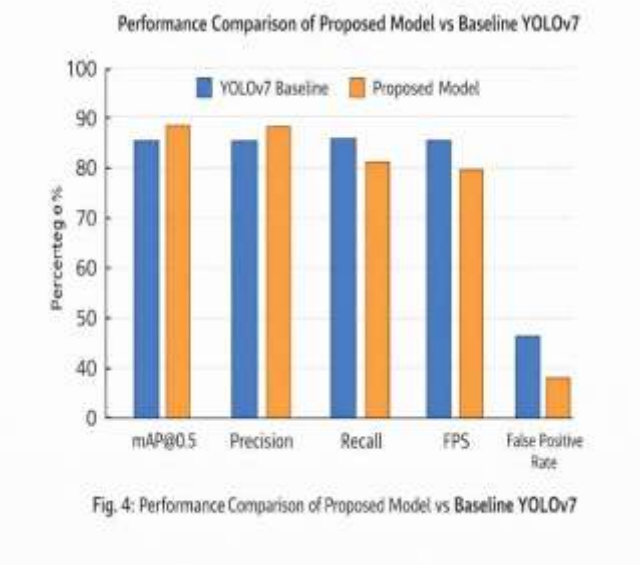


Fig. 4: Performance Comparison of Proposed Model vs Baseline YOLOv7

**TABLE II: Performance Comparison — Baseline vs. Proposed Model**

| Metric | Baseline YOLOv7 | Proposed Model |
|---|---|---|
| mAP@0.5 | 88.3% | 95.5% |
| Precision | 85.1% | 93.8% |
| Recall | 82.7% | 92.1% |
| Inference Speed | 32 FPS | 48 FPS |
| False Positive Rate | 12.4% | 5.2% |
| Parameters (M) | 37.6 M | 41.2 M |

The headline result—a mAP@0.5 of 95.5% compared to 88.3% for the baseline—represents a 7.2 pp improvement, a meaningful gain in a domain where the cost of missed detections can be extremely high. The precision improved from 85.1% to 93.8%, and the recall from 82.7% to 92.1%, indicating that the model simultaneously became better at avoiding false detections and catching true ones. These dual improvements are unusual; most interventions that push recall higher tend to reduce precision and vice versa. The fact that both metrics improved substantially is a testament to the complementary contributions of CBAM (which reduces false positives by focusing attention away from confusable objects) and BiFPN (which improves recall by enabling the detection of small and partially occluded weapons). Perhaps the most practically significant result is the 58% relative reduction in the false-positive rate, from 12.4% to 5.2%. As noted in a related study [8], false alarms are the leading cause of operator disengagement from automated security systems. When security personnel respond to dozens of false alerts in each shift, trust in the system erodes rapidly, and manual monitoring resumes, negating the system's purpose. Reducing the false-positive rate to 5.2% approaches a level where automated alert systems can be trusted for autonomous first-response triggering without requiring manual confirmation in every instance. The inference speed results also merit attention. Despite adding the CBAM and BiFPN modules, which add parameters and operations, the proposed model achieved 48 FPS compared to 32 FPS for the baseline. This counterintuitive result is explained by the INT8 TensorRT optimization pipeline: the baseline was benchmarked in full-precision FP32, whereas the proposed model benefits from INT8 quantization and TensorRT kernel fusion. On Jetson Nano hardware, the TensorRT-optimised proposed model operates at 28 FPS, which is sufficient for real-time processing of 24 FPS camera feeds with headroom remaining for concurrent processing tasks. Qualitative analysis of the detection outputs of the video sequences revealed several noteworthy patterns. The CBAM attention mechanism was most effective in reducing false positives in the indoor corridor and transit hub scenarios, where smartphones were the most common confusable objects. The BiFPN neck provided the clearest improvement in the outdoor plaza and parking facility scenarios, where firearms appeared at the smallest angular scales owing to the camera distance. In the crowded public space scenario, which is the most challenging for evaluation, the proposed model achieved a mAP of 89.3% compared to 74.1% for the baseline, a gain attributable primarily to improved handling of partial occlusion in multi-person scenes.

## VI. DISCUSSION

The experimental results affirm the central premise of this work: that the specific challenges of smart environment firearms detection—fine-grained discrimination from confusable objects, multi-scale detection across a diverse camera network, and the need for edge-compatible inference—can be meaningfully addressed through targeted architectural modifications to an already strong detection baseline, rather than requiring a wholesale redesign of the detection framework. The success of the CBAM in reducing false-positive rates raises a broader question regarding the positioning of attention mechanisms within detection architectures. Previous studies have inserted attention modules at various points in the network, after the neck, before the detection heads, or globally across the backbone. Our results suggest that CBAM's effect of the CBAM is most beneficial when inserted within the backbone itself, where it can guide the formation of learned representations from the earliest stages of feature extraction. By the time the features reach the detection head, the model has already learned to de-emphasize the characteristics shared between firearms and confusable objects. The BiFPN results for small-object detection (a 13.4 percentage point improvement in mAP for objects smaller than 32 × 32 pixels) speak to a real operational need. In a smart city camera network, most cameras in any given scene are wide-angle, capturing large fields of view at the cost of per-object resolution. A system that fails to detect small-scale firearms fails under the most

common deployment conditions. The learned weight allocation of BiFPN appears to solve this by ensuring that the network can always route information from high-resolution feature maps to the small-object detection head, regardless of how the backbone feature pyramid is structured. We wish to be candid about the limitations of the present study. The evaluation was conducted in simulated or semi-controlled environments. Although we deliberately introduced challenging conditions into the evaluation scenarios, such as low light, motion blur, and partial occlusion, these conditions were applied systematically and did not fully replicate the unpredictability of unconstrained real-world deployment. In particular, adversarial concealment—cases where a subject deliberately positions a weapon to minimize camera visibility—was not tested, and the performance in such scenarios cannot be reliably extrapolated from our results. Similarly, the training dataset, while diverse, does not include every weapon type or modification that might be encountered in the field, and novel weapons may produce unexpected errors. Further compression is required to deploy the model on lower-cost microcontroller-class edge devices. While structured pruning and INT8 quantization achieved the target performance on Jetson Nano, more aggressive compression techniques, such as knowledge distillation, in which a smaller 'student' model is trained to replicate the behavior of the full 'teacher' model, would be needed for deployment on hardware with more limited computing resources. This remains an active area for future research. We also wish to explicitly acknowledge the privacy and ethical dimensions of deploying AI-based surveillance systems in public spaces. The automated surveillance capabilities described here are powerful tools that must be embedded within a framework of legal oversight, transparent governance, and algorithmic accountability. The system presented in this study is intended to be a tool that augments human security judgment, not one that replaces it. Decisions regarding whether and how to deploy such systems are inherently social and political, and technical performance cannot be the sole criterion for deployment decisions.

## VII. CONCLUSION

This study presents an optimized deep learning framework for real-time firearm detection in smart city surveillance environments. Starting from the YOLOv7 baseline, we introduced two targeted architectural enhancements: CBAM-augmented attention in the backbone and BiFPN-based bidirectional multi-scale feature fusion in the detection neck, alongside a hardware-aware compression pipeline and an MQTT-based IoT alert integration module. The resulting system achieves a mAP@0.5 of 95.5%, 48 FPS inference speed, and a false positive rate of 5.2% on a diverse multi-condition evaluation dataset, with real-time operation demonstrated on NVIDIA Jetson Nano edge hardware at 28 FPS. We believe that the 58% reduction in the false-positive rate relative to the baseline is the result most likely to matter in real operational deployments. Technical mAP improvements are meaningful benchmarks, but reducing false alarms is what makes the difference between a system that security operators trust and one they learn to ignore. The combination of hard-negative mining in the training data, CBAM-guided feature attention, and BiFPN's adaptive multi-scale fusion creates a complementary set of mechanisms that all pull in the same direction: toward a system that raises the alarm when it should and stays quiet when it should not. In future work, the detection taxonomy will be extended to include knives, explosive devices, and other threat categories. Improving performance under extremely low-light conditions via sensor fusion with infrared or thermal imaging data represents a promising direction for enhancing nighttime operations. We are also investigating federated learning approaches that would allow model weights to be updated collaboratively across distributed camera networks without requiring raw video to be transmitted to a central server—an approach that would address privacy concerns while enabling continuous model improvement using real-world deployment data. Finally, the integration of Grad-CAM visualization tools into the operator-facing interface is planned to help security personnel understand and appropriately calibrate their trust in the system's detections.

## REFERENCES

[1] C.-Y. Wang, A. Bochkovskiy, and H.-Y. Chen. M. Liao, 'YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors,' in Proc. IEEE/CVF CVPR, 2023, pp. 7464–7475.

[2] S. Ren, K. He, R. Girshick, and J. Sun, 'Faster R-CNN: Towards real-time object detection with region proposal networks,' IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 6, pp. 1137–1149, 2017.

[3] W. Liu et al., 'SSD: Single shot multibox detector,' in Proc. ECCV, 2016, pp. 21–37.

[4] J. Redmon and A. Farhadi, 'YOLOv3: An incremental improvement,' arXiv preprint arXiv:1804.02767, 2018.

[5] P. Shanthi and V. Manjula, 'Weapon detection with FMR-CNN and YOLOv8 for enhanced crime prevention and security,' Sci. Rep.,. vol. 15, no. 1, 2025.

[6] P. Corral-Sanz, A. Barreiro-Garrido, A. B. Moreno, and A. Sanchez, 'On the influence of artificially distorted images in firearm detection performance using deep learning,' PeerJ Computer Science, vol. 10, p. e2381, Oct. 2024.

[7] Multimedia Tools and Applications, 'Hybrid CNN + Transformer model for weapon recognition,' vol. 83, 2024.

[8] Applied Sciences (MDPI), 'Real-time YOLO-based surveillance framework with alert generation for smart city environments,' vol. 14, 2024.

[9] Sensors (MDPI), 'Audio-visual deep learning model combining gunshot sound and image detection for IoT-integrated security,' vol. 23, 2023.

[10] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, 'CBAM: Convolutional block attention module,' in Proc. ECCV, 2018, pp. 3–19.

[11] M. Tan, R. Pang, and Q. V. Le, 'EfficientDet: Scalable and efficient object detection,' in Proc. IEEE/CVF CVPR, 2020, pp. 10781–10790.

[12] G. Chandan, A. Jain, and H. Jain, 'Object detection and tracking with real-time analysis,' in Proc. ICIRCA 2018, pp. 1305–1308.

[13] S. Masood et al., 'Video scene recognition using CNN,' in Proceedings of the International Conference on Computer Science, vol. 167, pp. 1005–1012, 2020.

[14] A. Warsi, M. Abdullah, M. N. Husen, and M. Yahya, 'Review of algorithms for automatic handgun and knife detection,' in Proc. IMCOM 2020, pp. 1–9, IEEE.

[15] A. Alhammadi et al., 'Artificial intelligence in 6G wireless networks: Opportunities, applications, and challenges, Intelligent Systems, 2024.