Journal Publication of International Research for Engineering and Management (JOIREM)
Volume: 03 Issue: 11 | Nov-2025
ISSN (O) 3107-6696

The Military Object Detection System using YOLOv7: An Automated Deep Learning Solution for Defense and Surveillance

Srishti Gupta[1], Mr.Ritesh[2]

[1]*Scholar, B.tech (AI&DS) 4th Year Department of Artificial Intelligence and Data Science Dr. Akhilesh Das Gupta Institute of Professional Studies, New Delhi, India.*

[2]*Assistant Professor (AI&DS), Department of Artificial Intelligence and Data Science Dr. Akhilesh Das Gupta Institute of Professional Studies, New Delhi, India.*

Abstract - The Military Object Detection System represents a significant advancement in the application of artificial intelligence for defense and surveillance applications. This research presents the development and deployment of a robust, automated solution utilizing YOLOv7 (You Only Look Once, version 7) architecture for detecting military bases and related objects across diverse media formats including static images, recorded videos, and real-time webcam streams. The system is designed to operate securely in offline environments, ensuring strict data privacy and compliance with defense protocols. The project encompasses data preparation with 13 military object classes (Aircraft, Camouflage, Drone, Fire, Grenade, Hand Gun, Knife, Military- Vehicle, Missile, Pistol, Rifle, Smoke, and Soldier), comprehensive model training utilizing state-of-the-art deep learning techniques, and deployment across multiple input modalities. Performance evaluation demonstrates exceptional results with mean Average Precision (mAP) at 0.5 intersection over union reaching 86.9% and precision at 90.1%, indicating strong generalization and reliability. The system delivers real-time detection at over 30 frames per second, making it highly suitable for operational surveillance and threat assessment. This paper presents the complete methodology, architectural details, experimental results, and operational deployment strategies for a practical deep learning-based military object detection system

Keywords - YOLOv7, object detection, military applications, deep learning, real-time detection, surveillance, convolutional neural networks, automation

Introduction

The integration of artificial intelligence and machine learning into defense and surveillance systems has become increasingly critical for modern security operations. Object detection, a fundamental computer vision task, enables automated identification and classification of targets in complex environments, providing crucial support for situational awareness, tactical decision- making, and operational efficiency. Traditional surveillance methods rely heavily on human operators and manual analysis, which are inherently limited by fatigue, processing capacity, and response time. Automated detection systems address these limitations by enabling rapid, consistent, and scalable analysis of vast amounts of visual data. The military and defense sectors face unique challenges in object detection applications. These include the need to identify diverse asset types across varying environmental conditions, from high-resolution satellite imagery to low-quality video feeds from ground based cameras. Additionally, defense applications demand secure, offline operation to protect sensitive information and ensure operational continuity in environments where internet connectivity is unavailable or restricted. The ability to process multiple input formats—static images, video sequences, and real-time webcam feeds—is essential for comprehensive situational awareness across different operational scenarios. Recent advances in deep learning have revolutionized computer vision, with convolutional neural networks (CNNs) achieving unprecedented accuracy in object detection tasks. The YOLO (You Only Look Once) family of detectors represents a paradigm shift in real-time object detection, reframing the task as a single regression problem that simultaneously predicts bounding boxes and class probabilities in a single forward pass. This unified approach eliminates the need for multiple passes over an image, enabling rapid inference without sacrificing accuracy. YOLOv7, the latest iteration of this architecture, incorporates significant architectural improvements and optimization strategies that further enhance both detection speed and accuracy. The primary motivation for this project is to develop a practical, deployable system that harnesses YOLOv7's capabilities for military object detection while addressing the specific requirements of defense environments. This includes secure offline operation, support for diverse input modalities, high detection accuracy, real-time processing, and user-friendly deployment. By combining cutting-edge deep learning techniques with practical engineering solutions, this research aims to deliver a tool that enhances security operations

and supports informed decision-making in critical defense scenarios.

2. LITERATURE REVIEW

2.1 Evolution of Object Detection Algorithms

Object detection has evolved significantly over the past two decades, transitioning from handcrafted feature approaches to sophisticated deep learning-based systems. Early methods such as Haar cascades, histogram of oriented gradients (HOG), and support vector machines (SVM) achieved moderate success in controlled environments but struggled with complex backgrounds, scale variations, and occlusions. These traditional approaches required extensive manual feature engineering and domain expertise, limiting their generalizability across diverse applications. The introduction of deep learning marked a transformative shift in computer vision. Convolutional neural networks, pioneered by LeCun et al. with LeNet-5, demonstrated the potential of learned hierarchical features for image classification. The AlexNet architecture, which won the 2012 ImageNet Large Scale Visual Recognition Challenge, sparked renewed interest in deep learning and proved that CNNs could outperform handcrafted features at scale. This breakthrough led to the development of deeper and more sophisticated architectures including VGGNet, ResNet, and Inception networks, each advancing the state-of-the-art in feature extraction and classification accuracy. Region-based CNNs introduced a new paradigm for object detection by proposing candidate regions and classifying them independently. R-CNN, proposed by Girshick et al., achieved significant improvements in detection accuracy by applying CNN based feature extraction to region proposals generated by selective search. Fast R-CNN optimized this approach by proposing once and using Region of Interest (RoI) pooling to extract features more efficiently. Faster R-CNN further accelerated the process by replacing selective search with a learnable Region Proposal Network (RPN), enabling end-to-end training and faster inference.

2.2 YOLO and Deep Learning Approaches

The YOLO (You Only Look Once) family revolutionized object detection by introducing a fundamentally different approach. Rather than treating detection as a classification problem applied to region proposals, YOLO frames detection as a single regression problem, predicting bounding box coordinates and class probabilities directly from the input image. This single-pass approach enabled real-time detection on standard hardware, making object detection practical for time-sensitive applications. YOLOv1, introduced by Redmon et al. in 2015, processed the entire image in a single forward pass, dividing it into a grid and predicting bounding boxes and class probabilities for each grid cell. While this approach sacrificed some accuracy compared to region-based methods, it achieved approximately twice the frames per second compared to Fast R-CNN, making real-time detection feasible. Subsequent versions introduced grid refinements, multi-scale predictions, and architectural improvements that incrementally increased accuracy while maintaining real-time performance. YOLOv7, the latest iteration, incorporates several advanced techniques that enhance both accuracy and speed. These include Extended Efficient Layer Aggregation Networks (E-ELAN), model scaling based on concatenation-based models, convolution reparameterization, and efficient edge- guided training strategies. The architecture is structured into four main components: the input layer with Mosaic augmentation and adaptive image scaling, a backbone for hierarchical feature extraction, a neck for multi-scale feature fusion, and a head for final predictions. YOLOv7 achieves state-of-the-art performance across the speed-accuracy spectrum, delivering competitive results from 5 frames per second to 160 frames per second depending on model scale and input resolution..

2.3 Related Work in Military and Surveillance Applications

The application of deep learning to military and defense scenarios has garnered significant research attention. Studies have explored CNN-based detection for identifying military vehicles, aircraft, infrastructure, and personnel in satellite, aerial, and ground-level imagery. These applications present unique challenges including limited labeled training data, variable image quality, extreme aspect ratios, and the requirement for high-confidence predictions in operational settings. Liu et al. demonstrated that YOLOv7-based systems can achieve 75.9% mean Average Precision on traffic detection in complex scenarios, improving baseline YOLOv7 performance by 3.7%. Similar studies have integrated attention mechanisms, deformable convolutions, and lightweight modules to improve detection accuracy while maintaining real-time performance. Research on traffic sign detection using improved YOLOv7 achieved 88.7% mAP@0.5 on the TT100K dataset, outperforming baseline YOLOv7 by 5.3%, demonstrating the effectiveness of architectural modifications tailored to specific application domains. In military contexts, object detection systems have been deployed for automated surveillance and reconnaissance. Studies highlight the importance of robust training methodologies, diverse datasets representing operational scenarios, and careful validation to ensure performance in deployment. Defense applications particularly emphasize the need for confidence calibration, as false negatives can have critical consequences, while false positives may trigger unnecessary responses. The integration of multiple data modalities, sophisticated preprocessing

techniques, and domain-specific fine-tuning has proven essential for reliable military object detection systems.

2.4 Comparison with Alternative Detection Methods

Traditional handcrafted feature methods achieved moderate success but demonstrated fundamental limitations in complex, variable environments. Haar cascades and HOG descriptors required extensive tuning and struggled with scale variation and complex backgrounds. Support Vector Machines could classify regions but required manual feature extraction, making them labor-intensive and domain-specific. Region-based deep learning methods including R-CNN, Fast R-CNN, and Faster R-CNN significantly improved accuracy through end-to-end learning of hierarchical features. However, these methods' reliance on region proposals and multiple processing stages limited their speed. While Faster R-CNN with RPN achieved reasonable real- time performance, it remained slower than single-shot methods for equivalent accuracy levels. Single-shot detectors like SSD and YOLO variants addressed computational limitations by eliminating region proposal stages. SSD achieved competitive accuracy with improved speed through multi-scale feature maps. YOLO models, particularly YOLOv7, provide the most balanced solution for applications requiring both high accuracy and real-time performance, with the flexibility to scale model size to computational constraints. Compared to two-stage detectors, YOLO trades a small accuracy decrease for substantial speed improvements—critical for surveillance and military applications where processing multiple streams simultaneously is necessary.

3 OBJECTIVES AND SCOPE OF WORK

3.1 Primary Objectives

The overarching objective of this project is to develop a state-of-the-art automated object detection system specifically tailored for military and defense applications, leveraging the YOLOv7 deep learning architecture. The system is designed to provide rapid, accurate, and reliable identification of military assets across diverse input formats while maintaining strict security and data privacy requirements. Development of Multi-Class Detection Model: Construct and train a robust object detection model capable of recognizing 13 distinct military object classes: Aircraft, Camouflage, Drone, Fire, Grenade, Hand Gun, Knife, Military-Vehicle, Missile, Pistol, Rifle, Smoke, and Soldier. The model must deliver high precision and recall, minimizing false positives and false negatives in operational scenarios. Support for Diverse Input Modalities: Enable the system to process static images, recorded video files, and live webcam streams seamlessly. This multi-modal capability ensures adaptability to various operational requirements, from post-analysis of surveillance footage to real-time threat assessment and monitoring. Secure, Offline Operation: Architect the solution for complete offline functionality, eliminating dependency on internet connectivity. This requirement is critical for protecting sensitive military data and ensuring compliance with defense sector security protocols. All processing, training, and inference operations must occur locally with no external data transmission. Optimized Real-Time Performance: Achieve detection speeds exceeding 30 frames per second to enable real-time surveillance and monitoring applications. The system must maintain detection accuracy while processing video streams continuously without introducing significant latency. Comprehensive Performance Analysis: Evaluate system performance using industry-standard metrics including precision, recall, and mean Average Precision (mAP). Conduct sensitivity analysis on challenging scenarios involving occlusions, variable lighting, and cluttered backgrounds to identify performance characteristics and limitations.

3.2 Scope of Work

Dataset Curation and Preparation: Assemble a diverse, balanced dataset representing real-world military scenarios from multiple perspectives and conditions. Perform meticulous manual annotation with accurate bounding boxes and class labels for all 13 object categories. Organize data into training, validation, and test splits (typically 70%-15%-15% distribution) to facilitate rigorous model development and unbiased evaluation. Data Preprocessing and Augmentation: Standardize image and video frame resolutions to YOLOv7's input requirements (typically 640×640 pixels). Normalize pixel values to zero mean and unit variance. Implement advanced augmentation techniques including random scaling (±10%), horizontal and vertical flipping, random rotation (±15°), color jittering, and mosaic augmentation to increase dataset diversity and improve model generalization. Model Architecture Configuration: Configure YOLOv7 architecture with appropriate backbone (CSPDarknet- based), neck (Path Aggregation Network), and detection head components. Select model scale (nano, small, medium, large, or extra-large) based on computational requirements and accuracy targets. Configure anchor boxes based on dataset statistics. Hyperparameter Optimization: Systematically tune critical hyperparameters through experimentation: learning rate (typically 0.001-0.01), batch size (32-128), number of training epochs (100-300), momentum (0.9-0.99), and weight decay (0.0005). Employ adaptive learning rate schedules such as cosine annealing or step decay to optimize convergence. Model Training and Validation: Train the model using stochastic gradient descent optimization with regular validation set

evaluation. Monitor training and validation loss curves to assess convergence and detect overfitting. Calculate precision, recall, and mAP metrics at each epoch. Maintain checkpoints for model recovery and fine-tuning. Inference System Development: Develop modular inference pipelines for static images, video files, and real-time webcam feeds. Implement preprocessing for each input type, inference execution, and post-processing for bounding box non-maximum suppression. Optimize inference code for speed while maintaining detection accuracy. Performance Evaluation and Documentation: Conduct comprehensive performance analysis including quantitative metrics and visual error analysis. Create detailed technical reports documenting methodology, results, limitations, and recommendations. Prepare user guides for deployment and operational use

4. METHODOLOGY

1.1 Data Preparation and Preprocessing

The foundation of any successful deep learning system is high-quality, representative training data. This project assembled a comprehensive dataset consisting of thousands of images and video frames representing 13 military object classes. Data collection targeted diverse scenarios including different lighting conditions, viewing angles, scales, and backgrounds to ensure model robustness. Dataset Composition and Annotation: Each sample in the dataset was manually annotated using computer vision annotation tools, creating precise bounding boxes around objects of interest and assigning appropriate class labels. This meticulous annotation process ensures the model receives accurate supervision during training. The dataset was organized into training, validation, and test sets using stratified splitting to ensure each set contained representative distributions of all 13 classes. Preprocessing Operations: All images were resized to 640×640 pixels to conform to YOLOv7's input specifications. Pixel values were normalized to the range [0, 1] and further standardized using ImageNet statistics (mean subtraction and standard deviation normalization). This normalization stabilizes training by ensuring inputs to the network have consistent statistical properties, enabling faster convergence and more stable gradient flow. Data Augmentation Strategy: Advanced augmentation techniques were applied during training to increase effective dataset size and improve model robustness: Mosaic augmentation combined four training images into a single input, increasing batch diversity; random scaling varied object sizes within ±10%; horizontal and vertical flipping provided reflection invariance; random rotation (±15°) accommodated viewing angle variations; color jittering modified brightness, contrast, and saturation; Gaussian noise addition simulated sensor noise. These augmentations were applied probabilistically during training, ensuring the model encountered varied representations of the same objects

1.2 Model Architecture and Configuration

YOLOv7 Architecture Overview: YOLOv7 comprises four primary components. The input layer applies mosaic augmentation, adaptive anchor calculation, and automatic image scaling. The backbone network (CSPDarknet-based) extracts hierarchical features through convolutional operations with residual connections, progressively increasing receptive field while reducing spatial dimensions. The neck (Path Aggregation Network) fuses features across multiple scales through upsampling and concatenation, enabling the detector to handle objects of varying sizes. The detection head predicts bounding boxes, objectness scores, and class probabilities for each scale, outputting predictions in three feature map sizes (80×80, 40×40, 20×20 for 640×640 input). Anchor Box Configuration: YOLOv7 uses predefined anchor boxes to initialize predictions. Rather than using generic ImageNet derived anchors, this project computed dataset-specific anchors through k-means clustering on ground truth bounding box dimensions. This adaptation ensures anchors match the typical aspect ratios and scales present in military object images, improving initial prediction quality and convergence speed. Loss Function: The training objective combines multiple loss components. Bounding box regression loss (typically GIoU or DIoU loss) measures coordinate prediction accuracy. Objectness loss (binary cross-entropy) determines whether regions contain objects. Classification loss (cross-entropy across 13 classes) assigns class predictions. The weighted combination of these components is optimized during training: Total Loss = $\lambda\_box \times L\_box + \lambda\_obj \times L\_obj + \lambda\_cls \times L\_cls$ where λ values balance the relative importance of each component.

1.3 Training Procedure and Hyperparameter Tuning

Optimization Strategy: Model training employed Stochastic Gradient Descent with momentum 0.937 and weight decay 5e-4. A learning rate schedule was implemented, initializing at 0.01 and decaying via cosine annealing over the training period. The warmup strategy gradually increased the learning rate for the first 1,000 iterations, stabilizing training before full convergence. Batch size was set to 64, balancing computational efficiency with gradient stability. Training Protocol: Training proceeded for 300 epochs over the complete dataset. Validation set evaluation occurred every epoch, computing precision, recall, F1 score, and mAP@0.5 metrics. Early stopping was not employed, as YOLOv7 typically requires extended training for optimal convergence. The training process was performed on local hardware using GPU acceleration (NVIDIA or equivalent), with all operations conducted entirely offline to

ensure data security. Hyperparameter Justification: Learning rate 0.01 provided sufficient gradient signal for effective parameter updates without causing divergence. Momentum 0.937 stabilized gradient descent by smoothing noisy gradient estimates from mini-batches. Weight decay 5e-4 provided regularization to prevent overfitting on the relatively balanced dataset. Batch size 64 balanced memory efficiency with sufficient gradient diversity per update. Cosine annealing enabled the learning rate to gradually decay, promoting convergence to sharper minima associated with better generalization

4.4 Validation and Performance Metrics

Evaluation Metrics: Model performance was quantified using standard object detection metrics: Precision: The fraction of predicted detections that were correct (TP/(TP+FP)), measuring false positive rate Recall: The fraction of ground truth objects that were detected (TP/(TP+FN)), measuring false negative rate Average Precision (AP): The area under the precision-recall curve for a specific class, computed at IoU threshold 0.5 Mean Average Precision (mAP): The average AP across all 13 object classes, providing single-number overall performance summary Validation Procedure: Validation occurred at the end of each training epoch on a held-out validation set (15% of total data). During validation, the model operated in inference mode (disabling dropout and batch normalization updates), and predictions were made on all validation images. Non-maximum suppression with IoU threshold 0.45 eliminated redundant overlapping detections, and confidence threshold 0.5 filtered low - confidence predictions. Metrics were computed by comparing predictions to ground truth annotations. Results: The final trained model achieved exceptional performance metrics: Precision: 90.1% (indicating low false positive rate) Recall: 84.3% (indicating low false negative rate) mAP@0.5: 86.9% (indicating strong overall detection accuracy) These results demonstrate the model's ability to reliably identify military objects across diverse scenarios while maintaining low false positive and false negative rates

5. EXPERIMENTAL RESULTS AND ANALYSIS

5.1 Quantitative Performance Analysis

The trained YOLOv7 model demonstrated outstanding performance across all evaluation metrics. On the test set (15% of total data, held entirely separate from training and validation), the model achieved: Overall mAP@0.5: 86.9% - This represents the average detection accuracy across all 13 military object classes Precision: 90.1% - Indicating that 90.1% of predicted detections were true positives Recall: 84.3% - Indicating that 84.3% of all ground truth objects were successfully detected F1 Score: 0.871

- The harmonic mean of precision and recall Performance varied across object classes, with some classes achieving detection rates exceeding 92% mAP while others (typically smaller or more visually similar objects) achieved 78-82% mAP. Aircraft and Military-Vehicles, relatively distinctive large objects, achieved over 93% mAP. Smaller objects like Grenades and Knives, which share visual similarity with other objects and exhibit high scale variation, achieved 79-81% mAP. Soldiers, medium- scale highly variable objects, achieved 87% mAP.

5.2 Inference Speed and Real-Time Capability

A critical requirement for military surveillance applications is real-time processing. Testing on standard hardware (NVIDIA GeForce RTX 2060) demonstrated: Image Inference: 25-35 ms per 640×640 image, translating to 28-40 FPS for single-image processing Video Inference: 30-40 ms per frame including preprocessing, inference, and post-processing (25-33 FPS for continuous video streams) Webcam Inference: 25-35 FPS sustained over extended periods (hours of continuous operation) These frame rates comfortably exceed the 30 fps threshold typically required for surveillance and threat assessment applications, enabling real-time monitoring without excessive latency.

5.3 Sensitivity Analysis and Error Evaluation

Comprehensive analysis of detection failures revealed systematic patterns: Detection Challenges:

1. **Occlusion:** Partially obscured objects showed reduced detection rates. Objects obscured by >50% were frequently missed. However, partially visible objects (10-50% occlusion) were typically detected with minimal accuracy loss.
2. **Scale Variation**: Very small objects (64×64 pixels were detected with >90% accuracy.
3. **Lighting Conditions**: Extreme lighting (very dark images, harsh shadows) caused occasional failures. However, the data augmentation techniques effectively mitigated this issue for normal operating ranges.
4. **Complex Backgrounds**: Cluttered scenes with numerous confusing elements showed slightly reduced precision but maintained acceptable recall. The high precision (90.1%) indicates that the model avoided misclassifying background elements as military objects.

Rare Object Types: Less frequently represented classes (Grenades, Knives) showed higher false negative rates due to limited training examples. This suggests potential for improvement through expanded dataset collection for underrepresented classes.

6. CONCLUSION

The Military Object Detection System represents a practical implementation of state-of-the-art deep learning for defense and surveillance applications. By leveraging YOLOv7's capabilities and tailoring the architecture for secure, offline deployment, this project delivers a tool that effectively balances detection accuracy, processing speed, and operational security requirements. Key achievements include: Development of a multi-class detection model achieving 86.9% mAP@0.5 with 90.1% precision Real-time processing capability (>30 fps) across multiple input modalities Secure, offline operation meeting defense sector security requirements Comprehensive system implementation spanning data preparation through deployment The system successfully addresses the primary challenge of automated military object detection while maintaining the security and privacy requirements essential for defense environments. Quantitative performance metrics and operational testing demonstrate the system's reliability and effectiveness in practical scenarios. While certain limitations remain regarding detection under extreme conditions and dataset representativeness, the foundation established by this project provides a solid basis for future enhancements. By continuing to expand datasets, refine architectures, and integrate additional sensor modalities, the Military Object Detection System can be further enhanced to meet evolving defense requirements while maintaining its effectiveness in real-world operational deployment. This research contributes to the broader advancement of AI applications in defense by demonstrating how deep learning can be effectively deployed in secure, resource-constrained environments while maintaining performance standards. The work serves as both a practical tool for defense operations and a foundation for future research in automated surveillance and threat assessment systems

Abbreviations-

- AI: Artificial Intelligence
- NLP: Natural Language Processing
- STT: Speech-to-Text
- ASR: Automatic Speech Recognition
- LLM: Large Language Model
- UI: User Interface
- LMS: Learning Management System
- MB : Megabytes
- RAM : Random Access Memory

7. REFERENCES

[1] Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). YOLOv4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934.

[2] Wang, C.-Y., et al. (2022). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. arXiv preprint arXiv:2207.02696.

[3] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. Advances in Neural Information Processing Systems, 28.

[4] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 779-788).

[5] Liu, W., et al. (2016). SSD: Single shot multibox detector. European Conference on Computer Vision (pp. 21-37). Springer, Cham.

[6] Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

[7] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 770-778).

[8] Girshick, R. (2015). Fast R-CNN. Proceedings of the IEEE International Conference on Computer Vision (pp. 1440-1448).

[9] Lin, T.-Y., et al. (2014). Microsoft COCO: Common objects in context. European Conference on Computer Vision (pp. 740 755). Springer, Cham.

[10] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems, 25, 1097-1105.