WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal Homepage: www.ijrpr.com ISSN 2582-7421

AI-Based Multimodal Surveillance System for Threat Detection and Decision Support

Sanjana S Marigoudar, Sneha B S, Soujanya T, Soumya L K

Department of Computer Science and Engineering, GM institute of Technology,Davangere, Karnataka, India
Email: sanjanasmarigoudar@gmail.com

ABSTRACT

Video surveillance systems are widely used to monitor environments and identify objects of interest. However, object detection alone is often insufficient for security-related decision support because the presence of an object does not necessarily indicate a meaningful threat. This paper presents an AI-based surveillance prototype that integrates object detection, lightweight object tracking, zone-based monitoring, temporal event analysis, threat scoring, and explainable decision support. The system uses a YOLOv8 Nano detector trained and evaluated on the KIIT Military Target Archive (KIIT-MiTA), an aerial imagery dataset containing seven military-related object categories. A hash-based dataset audit identified exact duplicate images across data partitions, and a cleaned training split was created by removing identified duplicate images. Under the reported validation protocol, the baseline detector achieved a precision of 0.6461, recall of 0.6502, mAP@0.5 of 0.6726, and mAP@0.5:0.95 of 0.4535. The cleaned-training variant achieved a precision of 0.6207, recall of 0.6928, mAP@0.5 of 0.6786, and mAP@0.5:0.95 of 0.4391. The results show a modest improvement in recall and mAP@0.5, accompanied by reductions in precision and mAP@0.5:0.95. The proposed decision-support pipeline uses a rule-based temporal state machine and a bounded threat score to distinguish initial object presence from verified crossing events. Explanations are generated through deterministic rules that identify the evidence contributing to a score. The system is implemented as a Streamlit application with SQLite-based storage. The results demonstrate the feasibility of integrating visual detection with interpretable event logic, while also highlighting limitations related to dataset generalization, simulated contextual inputs, and the scope of the evaluation.

Keywords: Object Detection, Surveillance, YOLOv8, Threat Detection, Temporal Analysis, Explainable AI, Military Imagery, Decision Support

1. INTRODUCTION

The increasing availability of cameras, unmanned aerial vehicles, and computer vision technologies has created opportunities for automated surveillance and security monitoring. Conventional surveillance systems often require human operators to continuously observe video feeds and identify events of interest. This process can become difficult when multiple cameras, large areas, or continuous video streams are involved.

Deep learning-based object detection has improved the ability of computer vision systems to identify and localize objects in images and video frames. However, object detection provides only frame-level information. It identifies what objects are visible, but it does not necessarily determine whether their presence or movement represents a meaningful event.

For example, detecting a vehicle within a monitored region does not automatically indicate a security threat. The significance of an object may depend on its location, movement, access conditions, and temporal behavior. Therefore, surveillance systems intended for decision support should consider more than object classification and localization.

This paper presents an AI-Based Multimodal Surveillance System for Threat Detection and Decision Support. The system combines a YOLOv8 Nano object detector with lightweight tracking, zone-based monitoring, temporal event logic, threat scoring, and rule-based explanations. The objective is to provide structured information about detected events and make the reasoning behind the assigned score understandable to a human operator. [18]

The prototype uses the KIIT Military Target Archive (KIIT-MiTA), which contains aerial images annotated with military-related object categories. A dataset audit was conducted to identify exact duplicate images across the training, validation, and test partitions. The detector was evaluated before and after cleaning the training split.

The term multimodal is used in the project title and system design. In the current implementation, the primary visual evidence comes from video-derived detections, while additional contextual information is represented through configured or simulated inputs. The prototype does not establish validated fusion of independent physical sensors.

1.1 Objectives

The major objectives of this work are:

To detect military-related objects in surveillance imagery using a YOLOv8 Nano model.

To associate detected objects across consecutive frames using lightweight tracking.

To monitor objects with respect to configured zones and spatial reference points.

To distinguish initial object presence from verified crossing events using temporal logic.

To calculate a bounded threat score using explicitly defined evidence components.

To provide deterministic explanations for the assigned threat score.

To evaluate the detector using standard object detection metrics and report the limitations of the prototype.

1.2 Contributions

The main contributions of this work are:

An integrated surveillance decision-support prototype combining detection, tracking, zone evaluation, temporal event logic, scoring, and explanations.

A hash-based dataset audit and cleaned training split intended to reduce exact-image overlap between data partitions.

A baseline-versus-cleaned detector comparison using precision, recall, mAP@0.5, and mAP@0.5:0.95.

A rule-based temporal mechanism that distinguishes object presence from verified crossing.

A transparent threat scoring mechanism whose components can be inspected and interpreted.

2. RELATED WORK

2.1 Object Detection

Object detection is a fundamental computer vision task involving the identification and localization of objects within an image. Traditional detection approaches relied on manually designed features, whereas deep learning-based approaches learn feature representations directly from data.

The You Only Look Once (YOLO) framework introduced a unified object detection approach that predicts object locations and classes in a single network evaluation [1]. Subsequent developments, including YOLO9000 and YOLOv3, improved detection performance and expanded the capabilities of the YOLO family [2], [3]. YOLOv4 further explored architectural and training improvements to balance speed and accuracy [4]. More recent YOLO-family implementations continue to provide practical object detection models for real-time applications.

In this project, YOLOv8 Nano is used as the detector because it provides a compact model suitable for an integrated surveillance prototype. The work focuses on evaluating the project-specific detector and its integration with event logic rather than proposing a new object detection architecture.

2.2 Multi-Object Tracking

Object tracking aims to maintain the identity of detected objects across consecutive video frames. Tracking can support the analysis of object movement, direction, and behavior over time.

Simple Online and Realtime Tracking (SORT) combines motion estimation with data association to track objects across frames [5]. Deep SORT extends this approach by incorporating appearance information to improve association under challenging conditions [6]. ByteTrack uses detection boxes at different confidence levels to improve object association [7].

The tracker used in this project is a lightweight greedy Intersection-over-Union (IoU) tracker. It associates detections between adjacent frames based on bounding-box overlap. It does not use learned appearance embeddings and is not an implementation of SORT, Deep SORT, or ByteTrack.

2.3 Explainable Artificial Intelligence

Explainable Artificial Intelligence (XAI) aims to make the behavior and outputs of AI systems more understandable to users. Methods such as LIME and SHAP provide approaches for interpreting model predictions [8], [9]. Explainability is particularly important in decision-support applications, where users should be able to understand the evidence behind system outputs.

The proposed prototype uses deterministic explanations rather than a separate learned explanation model. The explanation layer identifies the rule conditions and score components that contributed to an event. These explanations describe the system's configured decision logic; they do not explain the internal neural-network representations of the detector.

2.4 Contextual and Multisensor Decision Support

Sensor fusion combines information from multiple sources to produce a more complete representation of an environment. The effectiveness of fusion depends on the quality, reliability, and compatibility of the available sources [10].

The current prototype incorporates visual detections and configured contextual rules. Some contextual inputs, such as access-related information, are simulated. Therefore, the implementation is best understood as a visual surveillance system with contextual rule-based decision support rather than a fully validated physical multisensor fusion platform.

3. SYSTEM OVERVIEW

The proposed system is organized as a sequence of processing stages. Video input is sampled into frames, and the object detector predicts object classes, confidence values, and bounding boxes. A lightweight tracker associates detections between frames. Spatial reference points are then evaluated against configured zones.

The temporal module determines whether an object is merely present or has completed a verified crossing event. The threat scoring module combines the relevant evidence components and assigns a severity category. Finally, a rule-based explanation is generated and presented through the application interface.

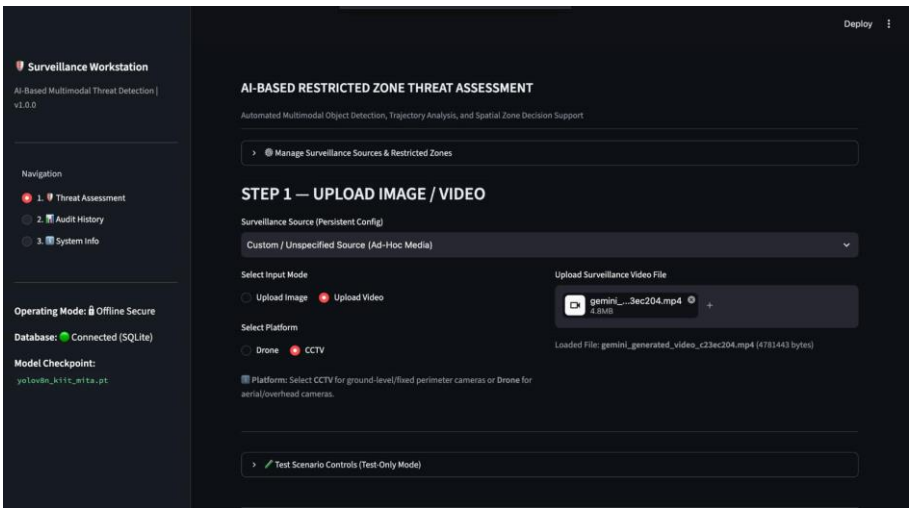


Figure 1. Surveillance workstation interface for media upload and source selection.

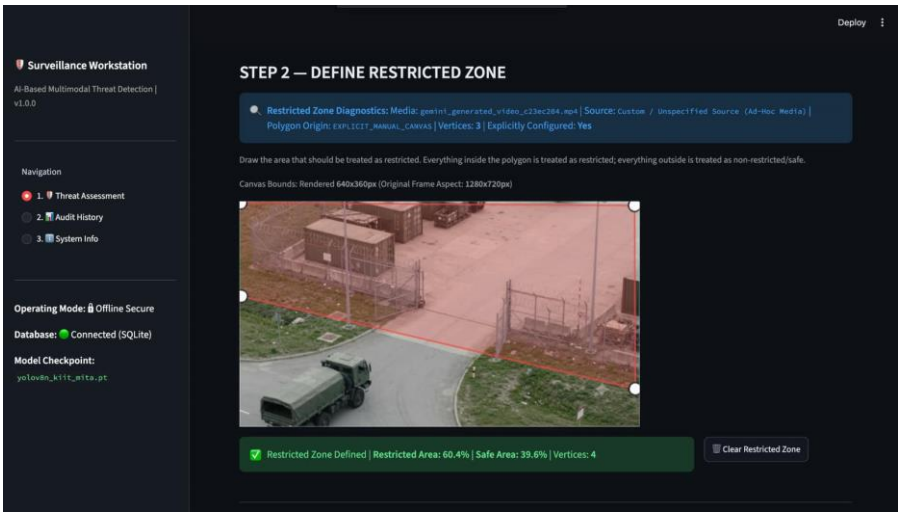


Figure 2. Restricted-zone configuration interface with a manually defined polygon.

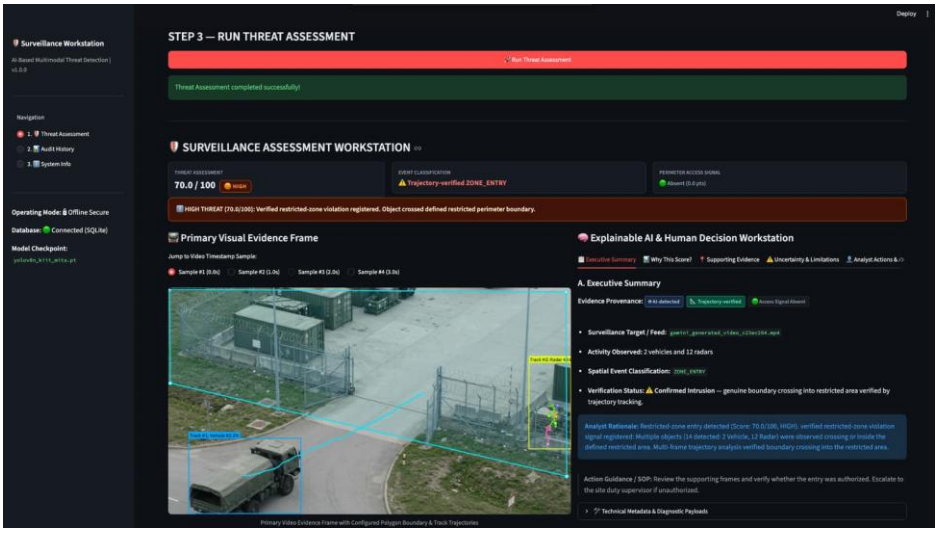


Figure 4. Example high-threat assessment with trajectory-verified zone-entry evidence.

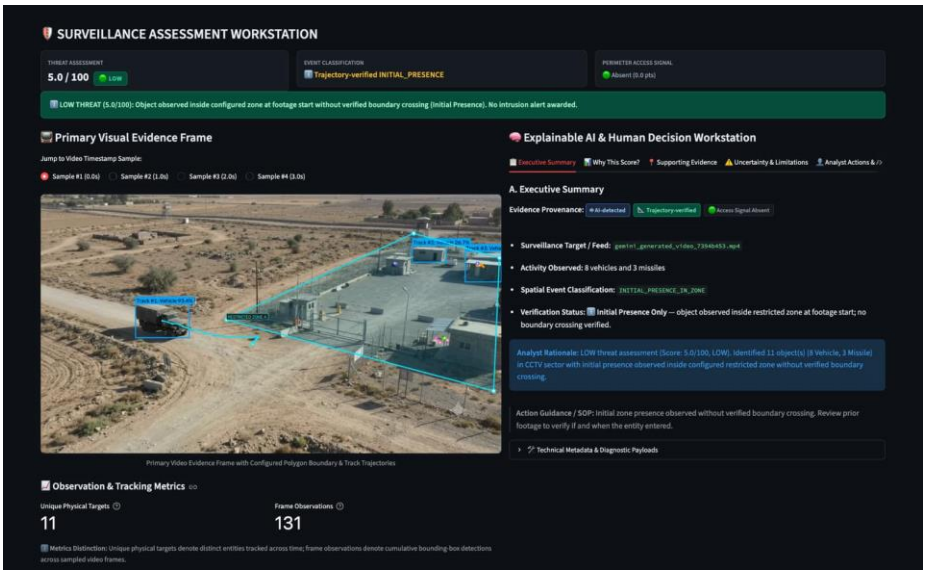


Figure 5. Example low-threat assessment for initial zone presence without verified boundary crossing.

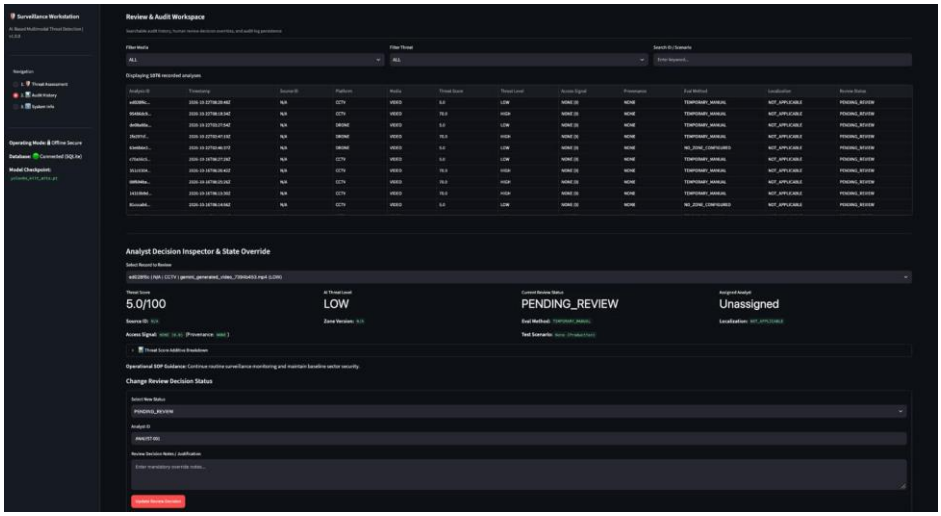


Figure 6. Review and audit workspace for recorded analyses and analyst decision review.

The application is implemented using Streamlit, with SQLite used for application data storage.

3.1 System Architecture

The system consists of the following modules:

1. Video Input Module: Accepts video input for processing and samples frames at the configured sampling rate.

2. Object Detection Module: Uses the YOLOv8 Nano model to identify objects and estimate their bounding boxes and class labels.

3. Object Tracking Module: Associates detections across consecutive frames using greedy IoU matching.

4. Zone-Based Monitoring Module: Evaluates object reference points against configured monitoring zones.

5. Behavior Analysis Module: Uses temporal states and crossing logic to distinguish initial presence from a verified event.

6. Threat Scoring Module: Combines event, zone, and access-related evidence into a bounded score.

7. Explainable AI Module: Generates deterministic textual explanations based on activated rules and score components.

8. Application and Storage Module: Provides the user interface through Streamlit and stores application records using SQLite.

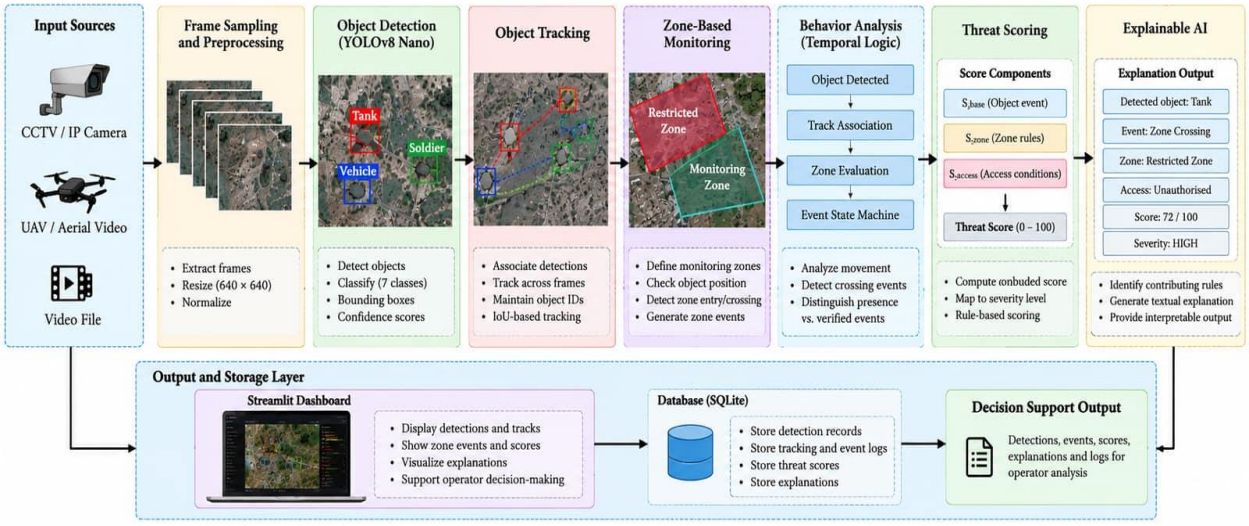


Figure 1. System architecture of the proposed surveillance and threat assessment system.

4. DATASET AND DATA PREPARATION

4.1 Dataset Description

The system uses the KIIT Military Target Archive (KIIT-MiTA), a dataset containing UAV or aerial images annotated for military-related targets. The project audit records a total of 1,700 images distributed across training, validation, and test partitions. [17]

The dataset contains seven object categories:

Artillery

Missile

Radar

M. Rocket Launcher

Soldier

Tank

Vehicle

The initial dataset split consists of 1,360 training images, 170 validation images, and 170 test images.

Table 1. Dataset summary

Dataset characteristic	**Reported value**
Total images	1,700
Training images	1,360
Validation images	170
Test images	170
Number of classes	7
Raw training bounding boxes	3,711

4.2 Dataset Duplicate Audit

Duplicate images across dataset partitions can affect the reliability of evaluation results. If an identical image appears in both the training and test partitions, the test result may not represent generalization to unseen imagery.

A hash-based audit was conducted to identify exact duplicate images. The audit identified four exact duplicates between the training and test partitions, four between the training and validation partitions, and eleven additional internal training duplicates.

A total of 19 duplicate images were identified, corresponding to 95 annotated bounding boxes. These identified duplicate images were removed from the training split to create a cleaned training dataset.

Table 2. Dataset duplicate audit

Audit item	**Reported value**
Exact train–test duplicate images	4
Exact train–validation duplicate images	4
Additional internal training duplicates	11
Total duplicate images identified	19
Bounding boxes removed	95
Cleaned training images	1,341
Bounding boxes in cleaned training split	3,616
Exact-hash residual duplicates after cleaning	0

The four exact train–test duplicates represent 2.35% of the 170 test images.

The audit uses exact image hashes. Therefore, it detects byte-identical images but does not establish the absence of near-duplicates, related video frames, or source-level leakage. The cleaned dataset should be interpreted as a training split with identified exact duplicates removed.

5. METHODOLOGY

5.1 Object Detection Using YOLOv8 Nano

The object detection module uses YOLOv8 Nano to detect and classify military-related objects in the input frames. The model predicts bounding boxes, object class labels, and confidence scores. [18]

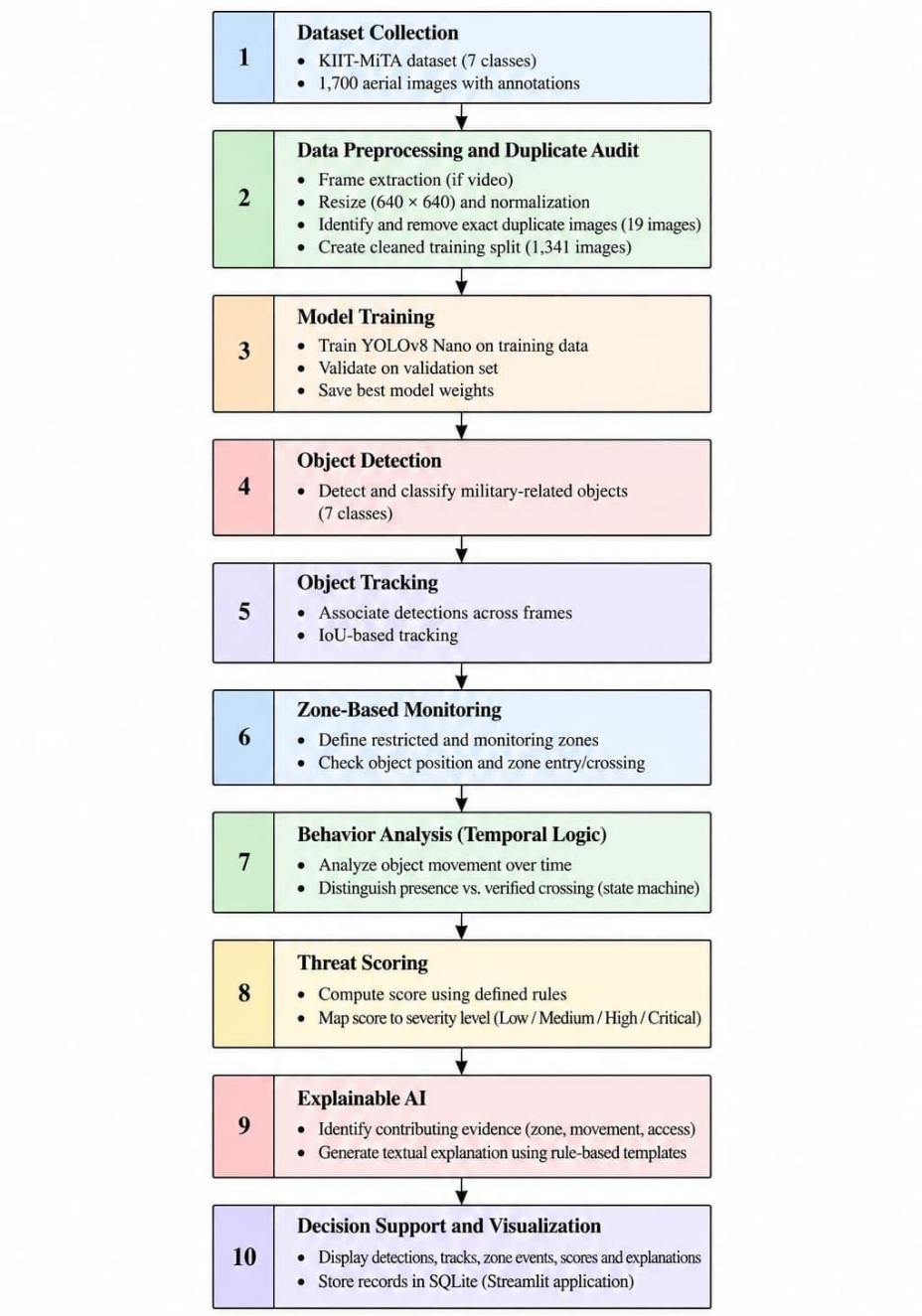


Fig. 2. Methodology flowchart of the proposed AI-based surveillance system for threat detection and decision support.

Figure 7. Methodology flowchart of the proposed surveillance and threat assessment system.

The project audit reports the following model characteristics:

Table 3. YOLOv8 Nano model characteristics

Characteristic	**Value**
Model	YOLOv8 Nano
Number of layers	225
Number of parameters	3,012,233
Computational complexity	8.2 GFLOPs
Evaluation image size	640

The detector is evaluated using precision, recall, mean Average Precision at IoU 0.5 (mAP@0.5), and mean Average Precision over IoU thresholds from 0.5 to 0.95 (mAP@0.5:0.95).

Precision measures the proportion of predicted detections that are correct, while recall measures the proportion of ground-truth objects that are detected. Average Precision summarizes the precision–recall relationship for a class, and mean Average Precision averages the corresponding values across classes.

The macro precision–recall curve reported in the project audit is calculated as the unweighted arithmetic mean across the seven classes.

5.2 Object Tracking

Object tracking is used to associate detected objects across consecutive frames. The implementation uses greedy IoU matching.

The Intersection-over-Union metric measures the overlap between two bounding boxes:

$$IoU(B_1, B_2) = \frac{|B_1 \cap B_2|}{|B_1 \cup B_2|}$$

where \(B_1\) and \(B_2\) represent two bounding boxes.

A pair of detections is considered eligible for association when its IoU meets the configured threshold. The audited tracker uses an IoU threshold of 0.30 and allows a track to remain unmatched for a maximum of two frames.

The tracker does not use appearance embeddings or a learned re-identification model. As a result, it is lightweight and deterministic, but may be affected by occlusion, crowded scenes, or significant object movement between frames.

5.3 Zone-Based Monitoring

The zone-based monitoring module evaluates the spatial relationship between detected objects and configured regions.

The reference point used for zone evaluation depends on the viewing context:

For CCTV-like views, the bottom-center point of the bounding box is used.

For aerial imagery, the centroid of the bounding box is used.

These reference points are used to evaluate whether an object is inside a monitored region or has crossed a configured boundary.

The zone-based module provides contextual information to the threat scoring and temporal analysis modules. It does not independently determine whether an object represents a threat.

5.4 Temporal Behavior Analysis

Object detection produces frame-level predictions. To analyze events over time, the system uses a temporal state machine containing seven states.

The temporal logic distinguishes initial object presence from a verified crossing event. Initial presence is not treated as equivalent to a confirmed boundary crossing. The audited temporal contribution for initial presence is zero, while a verified crossing contributes 65 points.

The state machine is designed to reduce false alarms that may occur when a detected object is simply present within a monitored region.

A small functional comparison was performed between the proposed temporal logic and a naive alternative. The proposed logic passed four out of four tested cases, while the naive alternative passed three out of four cases and produced a false alarm on initial presence.

This result represents a limited functional check of the implemented logic. It should not be interpreted as a broad reliability evaluation.

5.5 Threat Scoring

The threat scoring module combines the relevant evidence components into a bounded score.

The score is calculated as:

$$S = clamp(0,100, S_{base} + S_{zone} + S_{access})$$

where:

S_{base} represents the base evidence contribution.

S_{zone} represents zone-related evidence.

S_{access} represents access-related evidence.

The final score is restricted to the interval from 0 to 100.

The audited scoring components include a contribution of +5 for observation or presence, +65 for verified crossing, and +20 for access-related evidence.

Table 4. Threat scoring components

Evidence component	**Score contribution**
Observation or presence	+5
Verified crossing	+65
Access-related evidence	+20
Final score	Clamped to [0, 100]

The implementation distinguishes the temporal contribution for initial presence from the overall scoring component associated with observation. These are separate logic contexts and should not be treated as identical.

5.6 Threat Severity Classification

The final threat score is mapped to a severity category.

Table 5. Threat severity categories

Severity	**Score interval**
LOW	[0, 25)
MEDIUM	[25, 50)
HIGH	[50, 75)
CRITICAL	[75, 100]

The severity category provides a compact representation of the score for the application interface. It does not represent a validated prediction of hostile intent or operational threat.

5.7 Explainable AI Module

The explainable AI module generates textual explanations using deterministic templates and rules.

The explanation identifies the evidence components that contributed to the score, such as observation, verified crossing, zone conditions, and access-related evidence. The explanation is based on the system's configured decision logic.

The module does not use a large language model to generate explanations. Its purpose is to make the score composition inspectable and to support human understanding of the system's rule-based decisions.

6. EXPERIMENTAL SETUP

The detector evaluation follows the validation protocol documented in the project audit. The evaluation uses an image size of 640, a confidence threshold of 0.001, an IoU threshold of 0.7, and CPU execution.

The baseline detector is compared with a variant trained using the cleaned training split. The comparison uses precision, recall, mAP@0.5, and mAP@0.5:0.95.

A separate operational confidence setting of 0.25 produces an mAP@0.5 value of approximately 0.6153–0.6154. This value is affected by confidence-threshold truncation and should not be directly compared with the low-confidence precision–recall evaluation as though both were obtained under the same evaluation setting.

6.1 Runtime Evaluation

The runtime evaluation uses the video real_checkpoint_movement.mp4.

Table 6. Runtime evaluation configuration

Parameter	**Reported value**
Video duration	15.02 seconds
Resolution	640 × 360
Sampled frames	75
Sampling rate	5 frames per second
Internal pipeline timer mean	2,953.82 ms
Outer wall-clock mean	2,936.15 ms
Outer wall-clock standard deviation	59.35 ms
Approximate wall-clock time per sampled frame	39.15 ms/frame

The per-frame estimate is calculated from the reported outer wall-clock measurement and the 75 sampled frames. These measurements describe the stated execution context and should not be generalized to other hardware, resolutions, models, or video inputs.

7. RESULTS

7.1 Object Detection Performance

The baseline and cleaned-training variants were evaluated using the same reported validation protocol.

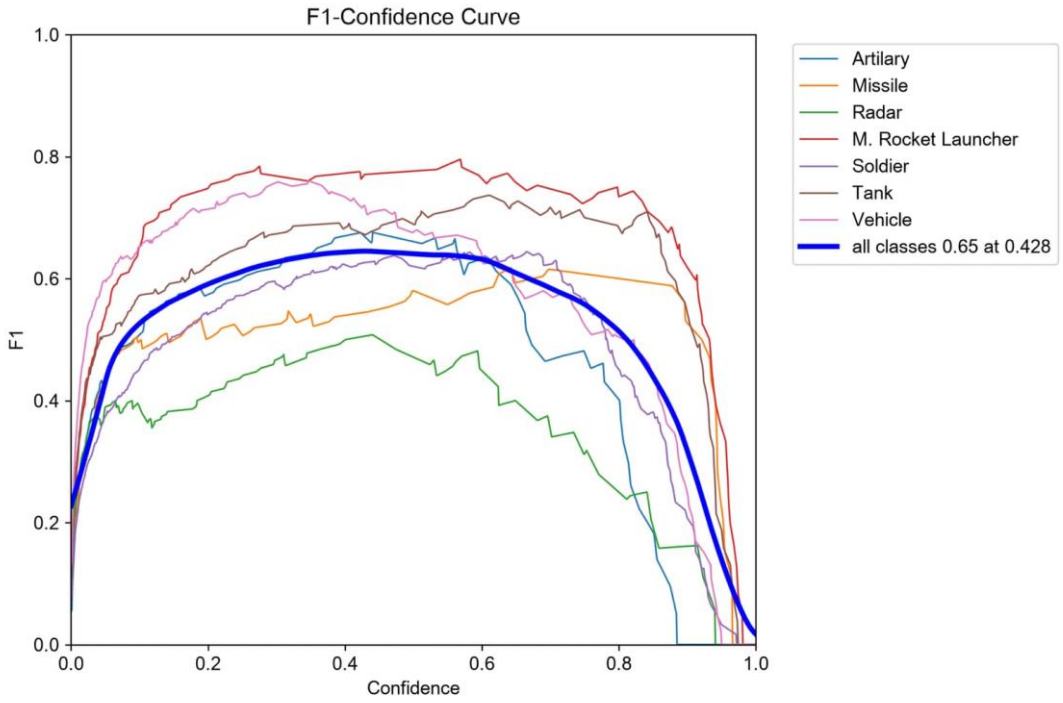


Figure 8. F1–confidence curves by target class (plot identity as a specific model variant is not labeled in the supplied figure).

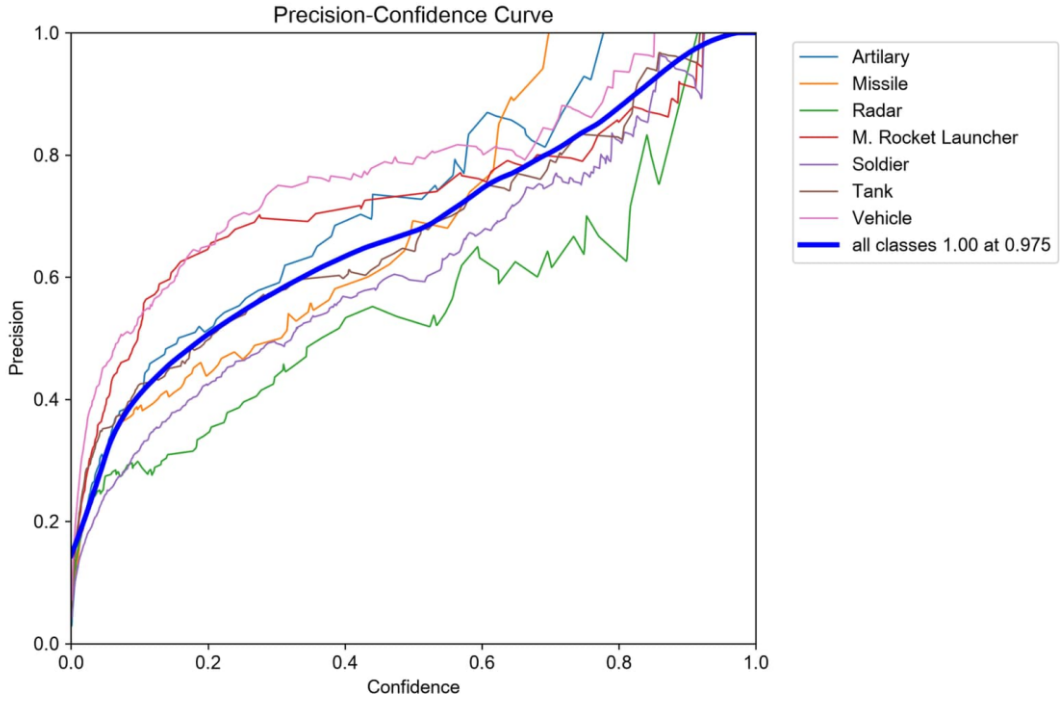


Figure 9. Precision–confidence curves by target class (plot identity as a specific model variant is not labeled in the supplied figure).

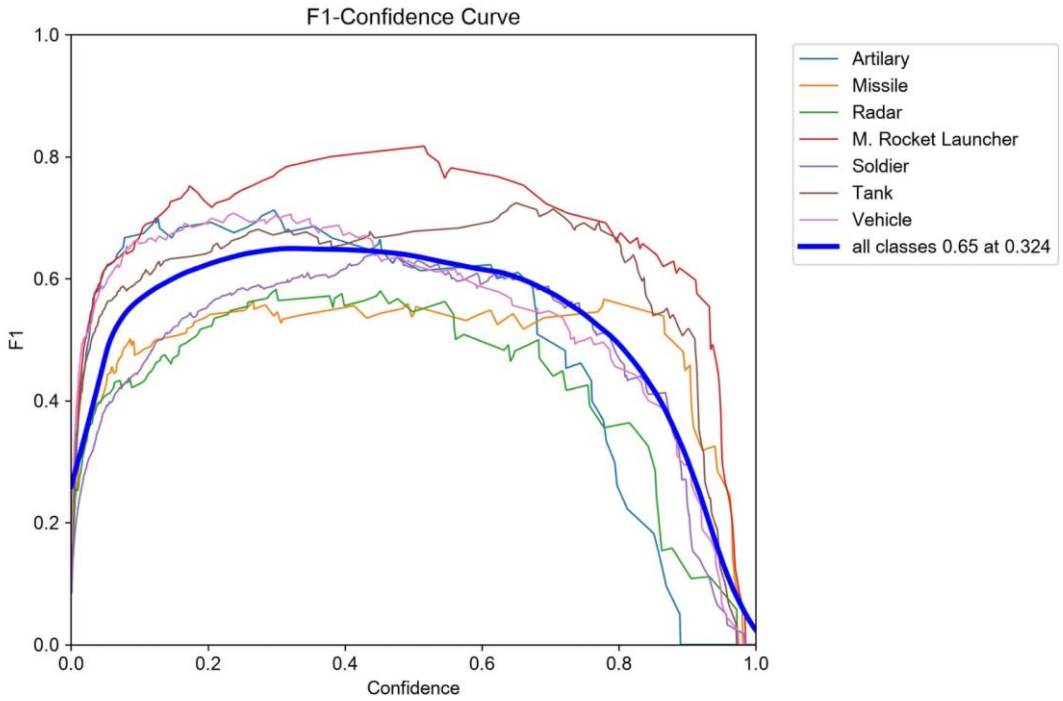


Figure 10. F1–confidence curves by target class (second supplied plot; model variant is not labeled).

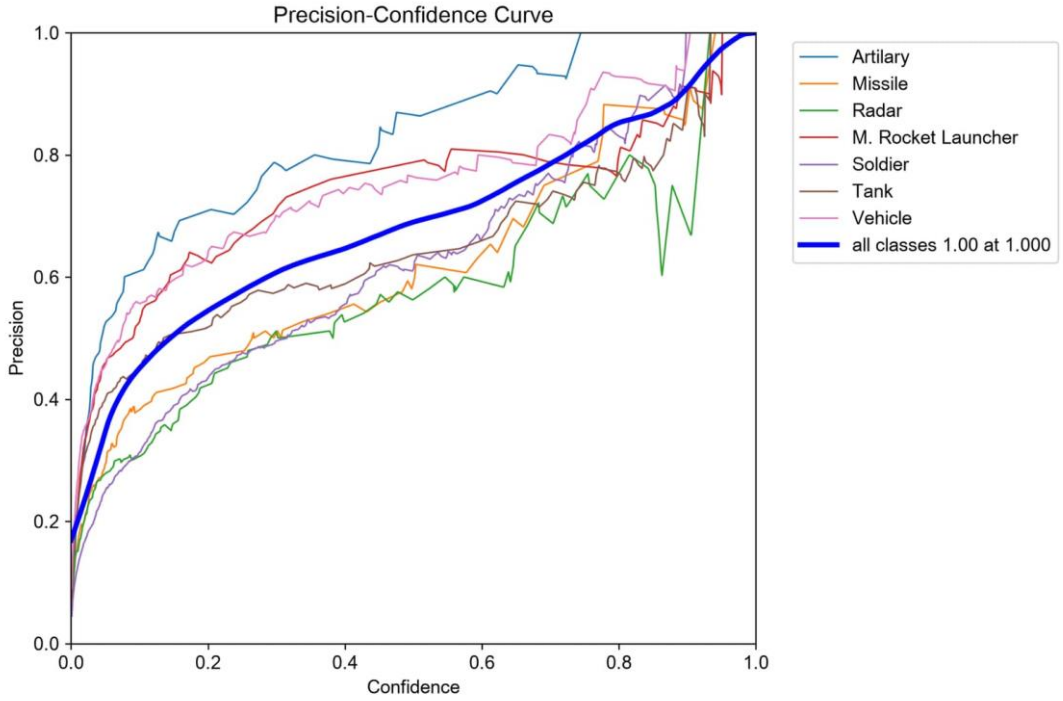


Figure 11. Precision–confidence curves by target class (second supplied plot; model variant is not labeled).

Table 7. Baseline and cleaned-training detector comparison

Metric	**Baseline**	**Cleaned-training variant**	**Difference**
Precision	0.6461	0.6207	−0.0254
Recall	0.6502	0.6928	+0.0426

Metric	Baseline	Cleaned-training variant	Difference
mAP@0.5	0.6726	0.6786	+0.0060
mAP@0.5:0.95	0.4535	0.4391	−0.0144

The cleaned-training variant achieved higher recall and a small increase in mAP@0.5. However, its precision and mAP@0.5:0.95 were lower than those of the baseline.

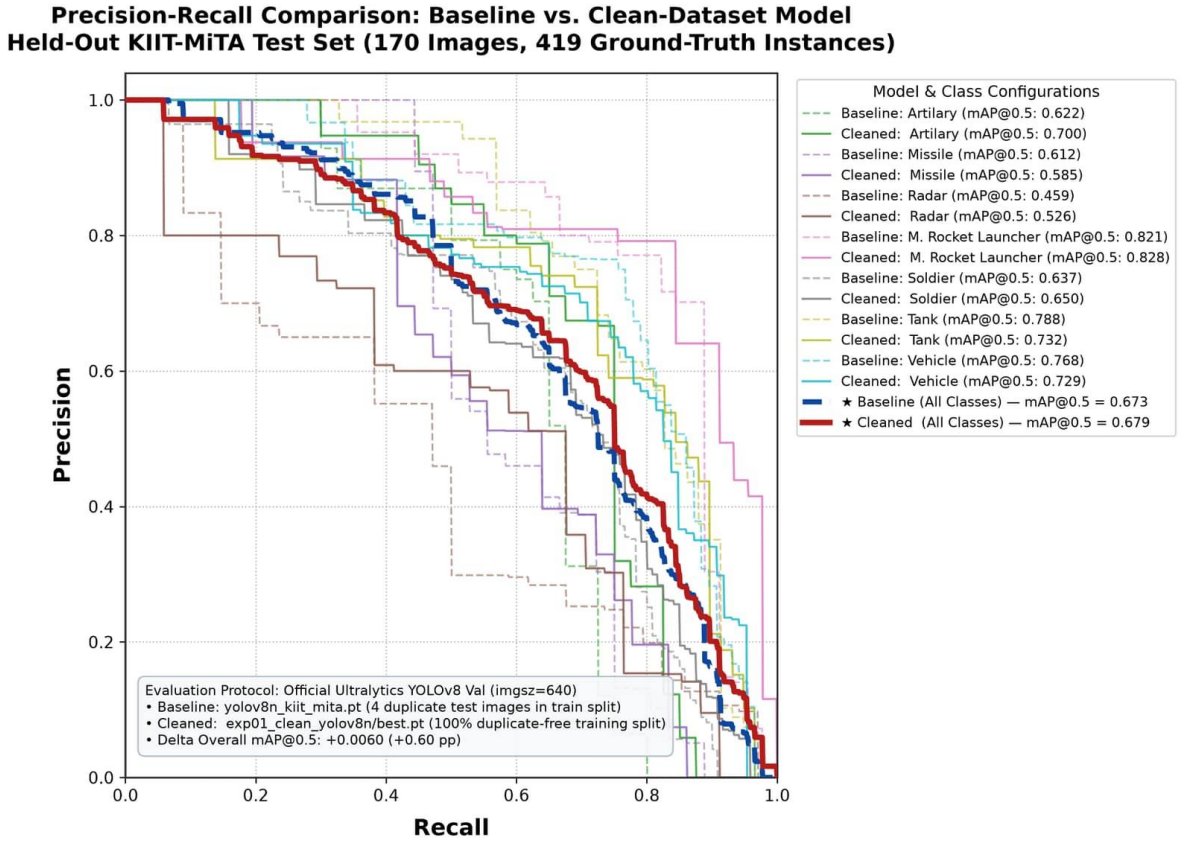


Figure 12. Precision–recall curve comparison between the baseline and cleaned-training-data model variants.

The mAP@0.5 difference of 0.0060 corresponds to an increase of 0.60 percentage points when expressed as a percentage-point change.

These results indicate a trade-off between the evaluated metrics. The cleaned-training variant should not be described as universally superior to the baseline. Instead, the comparison suggests that removing identified exact duplicates from the training split was associated with a change in the detector's precision–recall balance.

7.2 Temporal Logic Functional Check

The temporal behavior module was tested using four functional cases.

Table 8. Temporal logic functional check

Method	Cases passed	Observed behavior
Proposed temporal logic	4/4	Correctly distinguished the tested presence and crossing cases
Naive alternative	3/4	Produced a false alarm on initial presence

The result demonstrates that the proposed state logic handled the four tested cases as intended. However, the number of cases is small. The test does not establish the reliability of the temporal module across diverse real-world scenes.

7.3 Synthetic Scenario Benchmark

A separate six-scenario synthetic benchmark recorded the following outcomes.

Table 9. Synthetic scenario benchmark

Metric	**Count**
True positives	2
True negatives	4
False positives	0
False negatives	0
Total scenarios	6

The benchmark indicates that the implemented rules behaved as expected within the tested synthetic scenarios. Since the scenarios are limited and synthetic, these counts should be interpreted as a functional demonstration rather than evidence of field-level threat detection accuracy.

7.4 Runtime Results

The runtime evaluation was conducted using a 15.02-second video at 640 × 360 resolution, with 75 sampled frames processed at 5 frames per second.

The reported outer wall-clock mean was 2,936.15 ms, with a standard deviation of 59.35 ms. The approximate wall-clock time per sampled frame was 39.15 ms in the stated 75-frame execution context.

The internal pipeline timer mean was 2,953.82 ms. The internal and outer measurements are retained separately because they represent different timing contexts.

The results provide an observation of the prototype's runtime behavior under the specified configuration. They do not establish real-time performance for other video resolutions, hardware platforms, or deployment conditions.

8. DISCUSSION

The results demonstrate the feasibility of integrating object detection with tracking, zone-based monitoring, temporal event analysis, threat scoring, and deterministic explanations in a single surveillance prototype.

The detector comparison highlights the importance of evaluating multiple metrics rather than relying on a single performance measure. The cleaned-training variant achieved higher recall and a modest increase in mAP@0.5, while precision and mAP@0.5:0.95 decreased. This indicates a trade-off in the model's detection behavior.

The observed increase in recall may be useful in settings where missed detections are costly. However, lower precision may also increase the number of incorrect detections. The appropriate balance depends on the application, the consequences of false alarms, and the reliability requirements of the deployment environment.

The temporal state machine provides additional context beyond frame-level object detection. By distinguishing initial presence from verified crossing, it reduces the risk of treating every object observed in a monitored area as an event. The functional tests support the intended behavior for the tested cases, but broader evaluation is necessary.

The threat scoring module provides a structured representation of configured evidence. Its deterministic explanations allow the score components to be inspected. This improves transparency of the rule-based decision process, but does not mean the system understands the intent of detected objects.

The current implementation is a research prototype. Its results should be interpreted in the context of the dataset, validation protocol, runtime configuration, and limited functional tests.

9. LIMITATIONS AND ETHICAL CONSIDERATIONS

The current prototype has several limitations.

9.1 Dataset Generalization

The KIIT-MiTA dataset consists of aerial imagery. The trained detector may not generalize to ground-level CCTV footage, different camera viewpoints, lighting conditions, weather, or sensor characteristics.

9.2 Dataset Leakage Audit

The duplicate audit uses exact image hashes. It does not establish the absence of near-duplicate images, related video frames, or source-level leakage. Additional data auditing would be required to evaluate these possibilities.

9.3 Tracking Limitations

The tracking module uses greedy IoU matching. It does not use appearance features or learned re-identification. Consequently, tracking may be affected by occlusion, crowded scenes, rapid movement, or changes in camera viewpoint.

9.4 Simulated Contextual Inputs

Access-related and some contextual inputs are simulated or configured within the prototype. They are not connected to verified physical access-control systems or independent sensor sources. Therefore, the system does not demonstrate validated physical multimodal sensor fusion.

9.5 Threat Interpretation

The threat score is generated from explicit rules and evidence components. It does not predict hostile intent, establish the legality of an action, or provide a validated operational defense recommendation. A high score represents the configured rule output, not a confirmed real-world threat.

9.6 Limited Evaluation

The temporal logic evaluation consists of four functional cases, and the separate synthetic benchmark contains six scenarios. These tests are useful for checking the behavior of implemented rules but are insufficient for broad reliability or safety claims.

9.7 Privacy and Human Oversight

Surveillance technologies can affect privacy, civil liberties, and individual safety. Any real-world deployment should include human oversight, data governance, access controls, security safeguards, impact assessment, and compliance with applicable laws and institutional policies.

The system should be treated as a decision-support tool rather than an autonomous authority for high-impact decisions.

10. CONCLUSION

This paper presented an AI-based surveillance prototype for threat detection and decision support. The system integrates YOLOv8 Nano object detection, lightweight IoU-based tracking, zone-based monitoring, temporal behavior analysis, bounded threat scoring, and deterministic explanations.

The detector was evaluated using the KIIT-MiTA dataset. A hash-based audit identified exact duplicate images across dataset partitions, and a cleaned training split was created by removing the identified duplicate images. The baseline detector achieved a precision of 0.6461, recall of 0.6502, mAP@0.5 of 0.6726, and mAP@0.5:0.95 of 0.4535. The cleaned-training variant achieved a precision of 0.6207, recall of 0.6928, mAP@0.5 of 0.6786, and mAP@0.5:0.95 of 0.4391.

The results show a modest improvement in recall and mAP@0.5, accompanied by decreases in precision and mAP@0.5:0.95. Therefore, the cleaned-training variant represents a metric trade-off rather than an across-the-board improvement.

The temporal state machine and rule-based scoring module provide event-level context and explainable score contributions. The functional checks demonstrate the intended behavior for the tested cases, but do not establish operational reliability.

Future work should include evaluation on independent and diverse video sources, additional auditing for near-duplicate and source-level leakage, per-class and event-level analysis, comparison with stronger tracking methods, validation of contextual inputs, and human-centered safety and privacy assessments. These steps are necessary before considering deployment in operational surveillance environments.

REFERENCES

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779–788.

[2] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 7263–7271.

[3] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," arXiv preprint arXiv:1804.02767, 2018.

[4] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," arXiv preprint arXiv:2004.10934, 2020.

[5] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple Online and Realtime Tracking," in Proceedings of the IEEE International Conference on Image Processing (ICIP), 2016, pp. 3464–3468.

[6] N. Wojke, A. Bewley, and D. Paulus, "Simple Online and Realtime Tracking with a Deep Association Metric," in Proceedings of the IEEE International Conference on Image Processing (ICIP), 2017, pp. 3645–3649.

[7] Y. Zhang et al., "ByteTrack: Multi-Object Tracking by Associating Every Detection Box," in Proceedings of the European Conference on Computer Vision (ECCV), 2022.

[8] M. T. Ribeiro, S. Singh, and C. Guestrin, "'Why Should I Trust You?' Explaining the Predictions of Any Classifier," in Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 1135–1144.

[9] S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," in Advances in Neural Information Processing Systems (NeurIPS), 2017.

[10] R. C. Hall and J. Llinas, "An Introduction to Multisensor Data Fusion," Proceedings of the IEEE, vol. 85, no. 1, pp. 6–23, 1997.

[11] T.-Y. Lin et al., "Microsoft COCO: Common Objects in Context," in Proceedings of the European Conference on Computer Vision (ECCV), 2014, pp. 740–755.

[12] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," International Journal of Computer Vision, vol. 88, pp. 303–338, 2010.

[13] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, "Selective Search for Object Recognition," International Journal of Computer Vision, vol. 104, pp. 154–171, 2013.

[14] A. Padilla, C. F. Costa Filho, and D. R. R. Gomes, "A Comparative Analysis of Object Detection Metrics with a Companion Open-Source Toolkit," Electronics, vol. 10, no. 3, Art. no. 279, 2021.

[15] D. Gunning and D. Aha, "DARPA's Explainable Artificial Intelligence (XAI) Program," AI Magazine, vol. 40, no. 2, pp. 44–58, 2019.

[16] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2023.

[17] S. Chakrabarty, R. Chatterjee, S. Chakraborty, S. Roy Shuvo, and R. Chowdhury, "Drones in Defense: Real-Time Vision-Based Military Target Surveillance and Tracking," in Proc. 2025 3rd Int. Conf. Intelligent Systems, Advanced Computing and Communication (ISACC), Silchar, India, 2025, pp. 508–513, doi: 10.1109/ISACC65211.2025.10969335. KIIT-MiTA dataset record: https://github.com/Sudip-329/KIIT-MiTA.

[18] Ultralytics, "Explore Ultralytics YOLOv8," Ultralytics Documentation. [Online]. Available: https://docs.ultralytics.com/models/yolov8. [Accessed: Oct. 24, 2026].