Open Access ISSN: 3007-1437 (Online), ISSN: 3007-1429 (Print)

Alkadhim Journal for Computer Science (KJCS)

Journal Homepage: https://alkadhum-col.edu.iq/JKCEAS

AI-Powered Real-Time Surveillance: An Intelligent Threat Detection System for Security Enhancement

[1]Noor Abdulkaadhim Hamad*

[1]General Directorate of Education in Thi-Qar Governorate– Iraq

Article information

Article history:

Received: June, 13, 2026

Accepted: August, 11, 2026

Available online: September, 25, 2026

Keywords:

Artificial Intelligence,
Image Pre-processing,
Data Augmentation

Corresponding Author:

Noor Abdulkaadhim Hamad

noor_abd@utq.edu.iq

DOI:

https://doi.org/10.61710/4a4m7g78

Abstract

Intelligent real-time surveillance systems are upending the status quo in high-risk and complicated contexts due to the growing need for advanced solutions to improve public safety. But their performance is frequently hindered by insufficient training data diversity, class imbalance, and limited generalization due to the varying environmental conditions. The YOLOv8 object detection model is the foundation of this study's AI-powered surveillance framework, which integrates hybrid dataset construction with a data preparation workflow that includes image preprocessing, data augmentation, dataset splitting, and SMOTE-based dataset balancing applied only to the training subset prior to YOLOv8 training. To increase environmental variety and model generalization, a hybrid dataset was created by merging the Microsoft COCO dataset and the CCTV Surveillance Image Dataset. Additionally, in order to enhance the representation of minority object categories prior to YOLOv8 training, SMOTE-based dataset balancing was only applied to the training subset during data preparation. The proposed approach was evaluated using precision, recall, F1-score, mAP@0.5, mAP@0.5:0.95, AUC-ROC, confusion matrix analysis, and execution time. In the selected experimental configuration, precision increased from 0.87 to 0.94, recall from 0.84 to 0.96, and F1-score from 0.85 to 0.95. Additionally, the average execution time dropped from 0.55 to 0.35 s/image, and the suggested framework produced a mAP@0.5 of 0.94, a mAP@0.5:0.95 of 0.71, and an AUC-ROC of 0.99. In conclusion, the recommended framework shows that it will open the door to intelligent real-time surveillance in general-purpose ad hoc real-world situations and applications while simultaneously improving detection performance and maintaining computing economy.

1. Introduction

Among the recent developments in intelligent surveillance systems, Artificial Intelligence (AI) has started playing a key role because of its automatic analysis of visual information and real-time detection of security threats. The rapid development of smart cities, transportation infrastructures, airports, industrial facilities, and other vital environments has compelled the need for a surveillance system that can monitor continuously with as little human interference as possible. We compare with conventional surveillance technology in which a human observes the footage: AI-powered surveillance systems enhance operational efficiency by automatically identifying suspicious activities, help reduce human error, and allow faster responses to potential security incidents, hence improving public safety [1]. Recently, the development of deep learning has considerably boosted object detection for intelligent surveillance [2]. The YOLO family of object detection algorithms finds an excellent compromise between speed and accuracy, making them appropriate for real-time applications [3]. In particular, YOLOv8 incorporates architectural innovations that enhance inference efficiency and detection accuracy in a variety of surveillance scenarios [3]. However, practical surveillance is still difficult due to the class imbalance in surveillance datasets; narrow environmental diversity and low-light conditions can also result in weak detection [4]. Past research has investigated preprocessing methods, such as data augmentation to increase the generalization of the model [5], and approaches that balance the dataset, also known as SMOTE: a method aimed at increasing minority-class representation within imbalanced datasets [6]. In contrast, the majority of existing YOLO-based studies on surveillance focus mainly on detector backbone design or refer to a certain type of task (e.g., showing fewer research studies on automated weapon detection scenarios), with minimal exploration of the joint influences due to hybrid dataset constitution, combined preprocessing, class imbalance treatment, and overall performance assessment [7]. This study suggests an AI-based real-time surveillance framework based on the YOLOv8 object identification model to get beyond the aforementioned restrictions. The framework merges hybrid dataset construction with a data preparation workflow that includes image preprocessing, data augmentation, dataset splitting, and SMOTE-based dataset balancing before YOLOv8 training is run on the training subset. In order to diversify the environment and make the model generalize better, we have built hybrid datasets of the CCTV Surveillance Image Dataset and the Microsoft COCO dataset. Using comparison tests using Precision, Recall, F1-score, mAP@0.5, mAP@0.5:0.95, AUC-ROC, confusion matrix assessment, and time comparison for the detection depending on the chosen experimental design, we provide quantitative evaluations of the model.

The main contributions of this work are summarized in the following points:

- Build A Real-Time Surveillance Framework using the YOLOv8 Model for the intelligent detection of security threats with AI
- Creation of a hybrid dataset for surveillance by combining the CCTV Surveillance Image Dataset and the Microsoft COCO dataset, thus enhancing environmental diversity and model generalization.
- The proposed data preparation workflow includes: Image preprocessing, Data augmentation, Dataset splitting (a basic approach method), and a dataset balancing strategy based only on the train subset SMOTE.
- A comprehensive experimental study using precision, recall, F1-score, mAP@0.5, mAP@0.5:0.95, AUC-ROC, confusion matrix analysis, and execution time.
- Learned a novel data preparation workflow that leads to better detection performance while being computationally efficient.

The structure of this paper is as follows: In Section 2, we provide the literature review that summarizes the contemporary state-of-the-art studies for AI-enabled visual surveillance and object counting. Section 3 proposes the methodology. The experimental setup is presented in Section 4. Experimental Setup and Results: The results from the proposed approach are discussed in Section 5. Finally, Section 6 concludes the paper with a summary and future work.

2. Related Work

Recent advancements in artificial intelligence (AI) and deep learning have made a paradigm shift into intelligent surveillance systems that detect security threats directly from visual data with little to no human intervention. However, due to its detection performance on multiple objects with real-time processing power, most of the recent work on object detection has been on deep learning. As a result, intelligent surveillance holds a huge master key for smart cities and raw industrial environments such as public safety conditions that need to urgently detect threats, in order to vet the condition of an event [8]. Detection algorithms built on convolutional neural networks (CNNs) have progressed sufficiently from using conventional handcrafted features and are used as classes of the initial architecture of such deep learning-based surveillance systems due to this property. Compared with these methods, their computational complexities were relatively higher, so they can hardly be utilized in practical surveillance systems [9]. In order to overcome these limitations, researchers have progressively resorted to efficient one-stage object detectors, which efficiently solve the trade-off between detection performance and computational cost. Following that, the YOLO family of detectors, which provides a solution for intelligent surveillance, has come into play as its inference speed and detection precision outperform other more sophisticated techniques [10]. YOLOv8 brings architectural improvements regarding anchor-free detection, efficient image feature extraction, and also new efficient inference strategies that can improve accuracy while still keeping real-time speed. Lately, YOLOv8-based surveillance frameworks have been created and validated in practical settings for use as automatic weapons detection apparatus with applications based on a public safety actuator application [11]. In addition, low-power implementations widen the application of YOLOv8 for permanent computer vision surveillance systems based on CCTV [12].

Besides detector architecture, data augmentation has emerged as another important preprocessing technique for improving model robustness and generalization. Augmentation reduces overfitting by increasing dataset variety and allowing models to learn more representative visual characteristics through the use of geometric and photometric changes such as rotation, horizontal flipping, brightness adjustment, and noise injection. Recent studies have also shown that data augmentation improves model robustness under challenging image acquisition conditions and reduces overfitting by increasing dataset diversity. Nevertheless, most existing studies evaluate augmentation techniques independently, with limited investigation into their integration with complementary preprocessing strategies within a unified surveillance framework [13].

Another important challenge in intelligent surveillance is class imbalance, where security-critical objects such as weapons occur much less frequently than common categories including persons and vehicles. Deep learning models may be less able to identify uncommon but serious dangers as a result of this imbalance, which might bias them toward majority classes [14]. To address this issue, several balancing strategies have been put forth. These include well-known conventional oversampling techniques, such as the Synthetic Minority Oversampling Technique (SMOTE), which has improved learning over unbalanced datasets by using synthetic samples for minority classes. [14]. Chereddy and Bolla [15] found that synthetic data improved sampling quality and learning performance in low-resource settings, while Luo et al. [16] proposed a novel instance-level balancing strategy for object detection. Nevertheless, these works focus on balancing strategies in isolation from each other, while hybrid datasets and unified pipelines for object detection have emerged.

Another challenging problem of intelligent surveillance systems is that they mainly operate in adverse lighting conditions besides class imbalance. This task has also been mentioned to be a challenging one in real low-light conditions with YOLO in a recent benchmark [17]. Liu et al. [18] have shown that you can greatly increase surveillance visibility in intelligent transportation systems by using low-light enhancement. Moreover, Shrivastav and Andu [19] claimed that the combination of YOLOv8 Fusion with low-light enhancement methods can greatly improve object detection accuracy for challenging light conditions.

Nevertheless, the majority of related works have concentrated on detector architecture or disconnected preprocessing components, and there is still no validation system of a general surveillance framework to explain hybrid dataset construction, data augmentation, dataset balancing, and complete performance evaluation methods. Hence, the simultaneous impact of these two components on detection robustness and computational efficiency is still under-explored.

Based on the literature review and presented research gaps in each of the surveyed studies, this work proposes a comprehensive surveillance framework that incorporates hybrid dataset construction, data augmentation, and SMOTE methods for balancing datasets integrated through YOLOv8-based object detection & behavior (real-time) analysis, constituting one holistic pipeline for surveillance. The proposed approach addresses real-time monitoring applications required for practical deployment, focuses on how well to exploit these complementary factors while remaining computationally feasible, and simplifies earlier work, which predominantly focused on detector architecture or independent preprocessing methods. A qualitative comparison between normal research and the proposed framework is shown in Table 1.

Table (1): Comparison of Representative Studies and the Proposed Framework.

Ref.	Study	Model / Method	Main Focus	Strength	Limitation
[8]	Vijoli & Vancea (2024)	Deep Learning	Intelligent surveillance enhancement	Improves surveillance performance using enhancement techniques	Limited discussion of dataset balancing
[10]	Namana & Kumar (2024)	YOLOv8	Night-time surveillance	Robust real-time object detection under low-light conditions	Does not incorporate dataset balancing techniques
[11]	Dass et al. (2024)	YOLO	Real-time weapon detection	Accurate automatic weapon detection for public safety	Does not address data imbalance or low-light conditions
[12]	Deshpande et al. (2024)	YOLOv8	CCTV human detection	Energy-efficient real-time surveillance	Focuses primarily on detector implementation
[13]	Dubey et al. (2024)	Data Augmentation	Image augmentation	Improves model robustness and generalization	Does not address dataset balancing
[14]	Joloudari et al. (2023)	CNN + SMOTE	Class imbalance learning	Demonstrates the effectiveness of SMOTE for minority-class learning	Focused on classification rather than object detection
[15]	Chereddy & Bolla (2023)	GAN-based Synthetic Data	Class balancing	Improves learning in low-resource scenarios	Not specifically designed for object detection
[16]	Luo et al. (2024)	Object Detection	Instance-level data balancing	Addresses class imbalance directly in object detection	Does not consider surveillance-specific preprocessing
[17]	Shovo et al. (2024)	YOLO Models	Low-light object detection	Comprehensive evaluation of YOLO models under low-light environments	Does not investigate class balancing
[18]	Liu et al. (2024)	Deep Learning	Low-light enhancement	Improves visibility for intelligent transportation surveillance	Focused primarily on image enhancement
[19]	Shrivastav & Andu (2025)	Review	Low-light object detection	Reviews recent YOLO-based approaches for low-light object detection and highlights the integration of YOLOv8 with enhancement techniques	Review article only
—	This Work	Proposed Framework (YOLOv8)	Integrated intelligent surveillance	Integrates hybrid dataset construction, data augmentation, SMOTE-based dataset balancing, and YOLOv8-based object detection within a unified surveillance framework	Requires further validation on larger and more diverse real-world datasets.

3. Methodology

This section proposes a framework for an AI-based real-time intelligent security threat detection and surveillance system. It provides a seamless workflow, which combines performance evaluation with YOLOv8-based object detection data processing and merged dataset generation. In contrast to traditional surveillance strategies, which have approximated the detection model by improving disjointly, the proposed framework explores complementary data preparation techniques (such as image preprocessing, data augmentation, and a SMOTE-based dataset balancing strategy), conducting them in an effort to improve generalization capacity with contained computational cost [6]. Figure 1 depicts the general workflow of the suggested architecture.

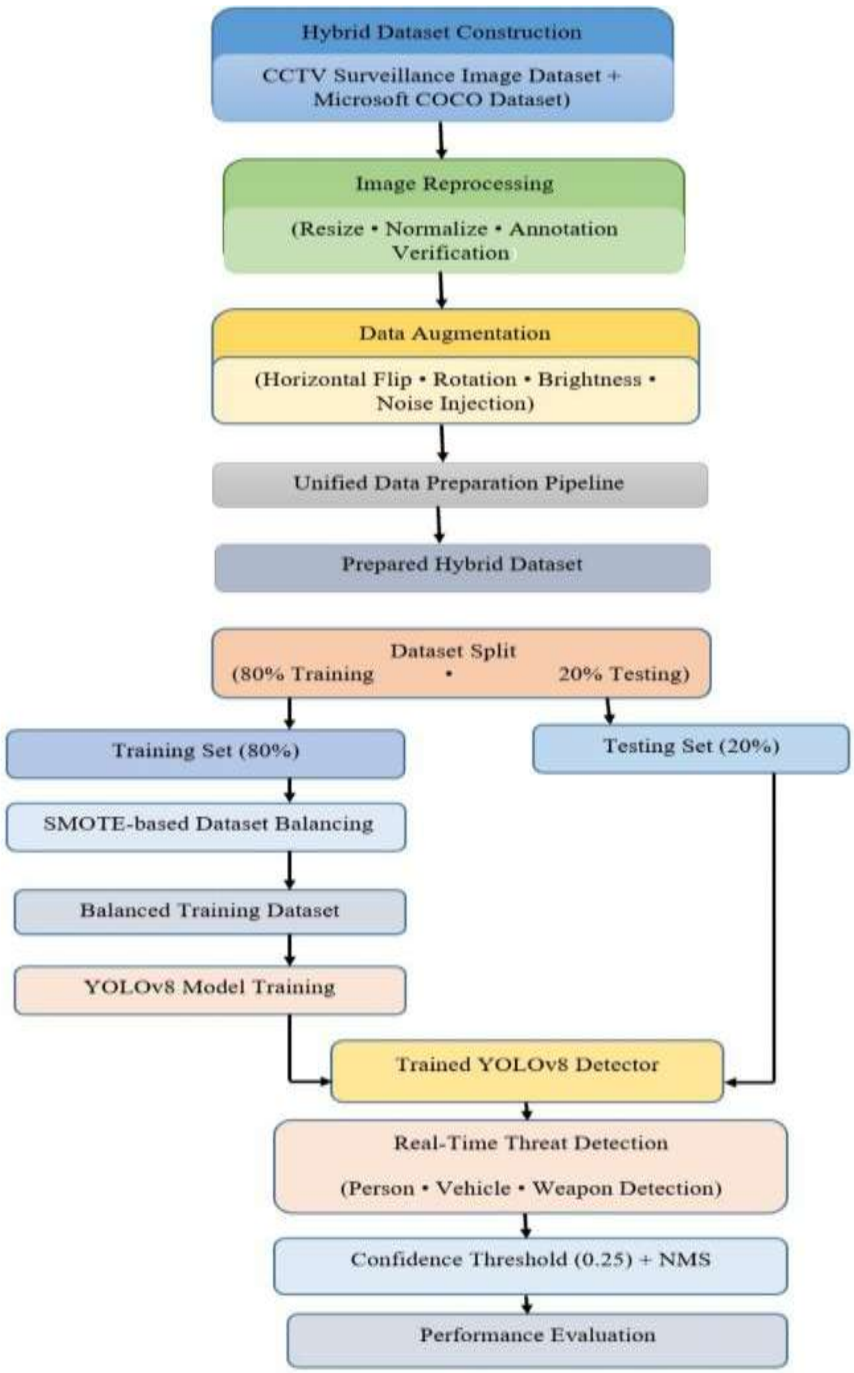

Figure (1): The proposed AI-powered surveillance system's overall architecture.

The suggested methodology starts with the creation of a hybrid dataset, as seen in Figure 1, and then moves on to picture preprocessing and augmentation. Only the training subset is exposed to SMOTE-based dataset balancing before YOLOv8 model training and real-time threat detection once the generated dataset has been divided into training and testing subsets.

3.1 Proposed Framework

The technology of the proposed framework will manage the major problem that lies in many works here: limited dataset diversity, class imbalance, and the requirement for efficient real-time object detection. To achieve more diversity in environments, a hybrid dataset was initially created for training. CCTV Surveillance Image Dataset is complemented with the Microsoft COCO dataset. The dataset captured images and then applied image processing as well as data augmentation. After this, the dataset that was prepared was split into 80% for training and 20% for testing. Then they applied SMOTE only to the training subset before running YOLOv8. The training data were later used to train the YOLOv8 detector to detect three different target classes (Person, Vehicle, and Weapon). During inference, predictions were corrected through a confidence threshold and Non-Maximum Suppression (NMS). Ultimately, it was validated on multiple performance measures, including detector precision and execution speed.

3.2 Hybrid Dataset Construction

This makes your dataset more diverse, leading to better model generalization. All of this is achieved through mixing the CCTV Surveillance Image Dataset with the Microsoft COCO dataset and building a hybrid dataset. The CCTV Surveillance Image Dataset provides realistic backgrounds, humans, vehicles, and weapons content, while Microsoft COCO is a large-scale dataset that has been captured under various circumstances, views, sunlight conditions, and background images [20]. In a hybrid dataset, only 44,126 images were obtained through combined labeling. The split between training (80%) and testing (20%) was done after preprocessing and using augmentation to generate the hybrid dataset, as seen in Table 2.

Table (2): Hybrid Dataset Composition and Data Split.

Dataset	Number of Images	Annotation Type	Purpose
CCTV Surveillance Image Dataset	3,456	Bounding Boxes	Surveillance images
Microsoft COCO Dataset	40,670	Bounding Boxes	Increase environmental diversity
Total Hybrid Dataset	44,126	Verified Bounding Boxes	Model development
Training Set (80%)	35,301	Verified Bounding Boxes	YOLOv8 training
Testing Set (20%)	8,825	Verified Bounding Boxes	Performance evaluation

The positive influence of a large-scale dataset is more comprehensively presented in that the increasing quality and amount of data can improve the detector to extract more semantic features, making it more robust and generalizable in most surveillance conditions, as shown in Table 2 [5].

3.3 Data Preparation

All the images were prepared using a common data preparation pipeline: preprocessing, augmentation, and then SMOTE dataset balancing [5], before being used for model training. Subsequently, geometric and photometric transformations were utilized to conduct data augmentation simulating the realistic conditions of surveillance, thereby enhancing the robustness of the model. The augmentation techniques are compiled in Table 3.

Table (3): Techniques for Augmenting Data.

Technique	Objective
Horizontal Flip	Simulate viewpoint variation
Rotation	Simulate camera angle changes.

Brightness Adjustment	Handle illumination variation
Noise Injection	Simulate sensor noise

The employed augmentation methods, as summarized in Table 3, were intended to simulate certain real-world transformations that the models will later have to confront during operation [5].
Initially, the produced hybrid dataset was divided into training (80%) and testing (20%) subsets to address the issue of class imbalance. Finally, for a balanced dataset as an offline preprocessing method, in order to enhance the ability to represent object categories, an SMOTE-based dataset balancing strategy was applied only to the training subset before YOLOv8 training. A balanced training dataset was now created and then used to train the YOLOv8 detector, whilst leaving unchanged the testing subset, which serves as an unbiased evaluation of the proposed framework. Only during data preparation, SMOTE stepped in, and the YOLOv8 architecture/loss function/inference was unchanged to align with the standard YOLOv8 detection pipeline [6]. SMOTE is only used to balance the training data; therefore, it has no impact on the evaluation protocol or the inference stage.

3.4 YOLOv8-Based Threat Detection

Because YOLOv8 provides a solid compromise between inference speed and detection accuracy for an intelligent real-time surveillance system, it was chosen [3]. The training dataset that we prepared was trained on any container of these 3 security-related object categories, such as Person, Vehicle, and Weapon.
Non-Maximum Suppression (NMS) was used [3] to eliminate overlapping duplicate measurements when comparing bounding boxes of interest after the detections with a confidence score less than 0.25 were eliminated for inference. These two stages allow us to achieve higher localization accuracy with efficient calculations in real time for surveillance scenarios.

3.5 Performance Evaluation

Different complementary metrics, including Precision, Recall, F1-score, mAP@0.5, mAP@0. 5:0.95, AUC-ROC, confusion matrix analytics [21], and execution time in terms of detection length and computational efficiency were used to evaluate the proposed framework, respectively. Table 4 shows the evaluation measures along with their objectives.

Table (4): Performance Evaluation Metrics.

Metric	Purpose
Precision	Calculates the percentage of threats that are accurately identified out of all threats that are detected.
Recall	Evaluates the framework's capacity to recognize real security risks.
F1-score	Gives the harmonic mean of precision and recall.
mAP@0.5	Determines the mean average precision at a 0.5 IoU threshold.
mAP@0.5:0.95	Assesses the mean Average Precision over IoU thresholds ranging from 0.50 to 0.95.
AUC-ROC	Evaluates the discrimination capability between threat and non-threat classes.
Confusion Matrix	Reports TP, TN, FP, and FN for detailed error analysis.
Execution Time	Measures computational efficiency for real-time surveillance.

The evaluation metrics selected reinforce each other and provide insights on detection accuracy, localization performance, classification confidence, and computation time as shown in Table 4. Although most object detection metrics are standard metrics, mAP@0.5 and mAP@0.5:0.95 were among the ones used to evaluate how well localization and detection were accomplished. More specifically, mAP@0.5 is the mean average precision at an Intersection over Union (IoU) of 0.5, and mAP@0.5:0.95 is the Mean Average Precision over multiple IoUs ranging from 0.50 to 0.95, giving a comprehensive analysis of detection accuracy [3]. A model is trained on the training subset, while we use the testing subset only for performance measurement. Section 5 discusses the comparative experiments and results (including baseline comparison, ablation analysis, and repetition for checking stability).

4. Experimental Setup

This section provides an overview of the experimental environment, dataset preparation, model configuration, and finally the evaluation protocol and workflow used to validate the practicality of our proposed AI-powered surveillance framework. We conducted multiple experiments in order to evaluate the performance of our unified data preparation pipeline, which enabled real-time threat detection based on YOLOv8.

4.1 Experimental Environment

Lastly, this framework was developed in Python 3.9 and run on a workstation with an NVIDIA RTX 3080 GPU. Implementation: coded in Ultralytics YOLOv8 and performing image processing using OpenCV, numerical calculations using NumPy, and visualization using Matplotlib. Table 5 summarizes the hardware and software configuration employed in this study.

Table (5): Experimental Environment.

Component	Specification
Programming Language	Python 3.9
Deep Learning Framework	Ultralytics YOLOv8
GPU	NVIDIA RTX 3080
Image Processing	OpenCV
Numerical Computing	NumPy
Visualization	Matplotlib

Table 5 provides a summary of the experimental environment used in this work; it supplies the computational resources and software environments for training YOLOv8 instances effectively to achieve real-time inference and perform reproducible experiments.

4.2 Dataset Preparation

A hybrid dataset is created by augmenting the CCTV Surveillance Image Dataset with the Microsoft COCO dataset to increase environmental variations and model generalization [20]. Post-integration, all images were pre-processed using a suite of image preprocessing techniques, which include resizing, normalization, annotation verification, and data augmentation. Finally, the data was divided into training (80%) and testing (20%). Then, a dataset balancing strategy based on SMOTE was applied only to the training subset as an offline preprocessing step prior to the YOLOv8 training [5,6]. The hybrid dataset included 44,126 annotated photos in total. The training subset was used to build the model, whereas the testing subset's observations were used to assess its performance. The hybrid dataset details and the used training/test split are shown in Table 2. This suite, composed of a mixture of surveillance-specific and benchmark images, provides enough diversity to code a suitable learning paradigm while avoiding optimistically assessing the proposed framework.

4.3 Annotation and Data Preparation

Each image had verified bounding-box annotations for three target categories: Person, Vehicle, and Weapon. Contaminated annotation files were inspected for integrity, and erroneous or missing labels were fixed prior to training. Data augmentation was used to improve robustness by applying horizontal flipping, rotation, brightness, and noise injection [5]. After that, an offline preprocessing step was carried out that used an SMOTE-based dataset balancing method to increase the number of samples in the minority class. This only affected the training subset prior to training [6]. To obtain an unbiased evaluation, the testing subset was kept constant across all the experiments. We did not change the YOLOv8 architecture, loss, or inference. Hence, SMOTE was only applied during data preparation for balanced training data and did not influence the evaluation protocol or inference stage.

4.4 Model Training Configuration

Using the prepared training subset, the YOLOv8 detector was trained based on the configuration summarized in Table 6. During inference, detections with confidence scores below 0.25 were suppressed, and the use of NMS (Non-Maximum Suppression) knocked out duplicates [3].

Table (6): YOLOv8 Training Configuration.

Parameter	Value
Detection Model	YOLOv8
Training Dataset	Training Set (35,301 images)
Testing Dataset	Testing Set (8,825 images)
Confidence Threshold	0.25
Post-processing	Non-Maximum Suppression (NMS)
Detected Classes	Person, Vehicle, Weapon

The summarized configuration, presented in Table 6, outlines the adoption of a confidence score of 0.25 for the YOLOv8 detector and NMS while relying on an isolated separation between training and test datasets to deliver a non-biased performance assessment for the testing phase of our proposed framework.

4.5 Evaluation Metrics

Precision, recall, F1-score, mAP@0.5, mAP@0.5:0.95, AUC-ROC, confusion matrix [21], and runtime analysis are complementary measures that we used to assess performance for computational efficiency analysis in real-time surveillance applications [3]. Table 4 shows the evaluation metrics along with their intended usage. For instance, these complementary metrics provide full insight into detection accuracy, localization performance, classification reliability, and computational efficiency against the experimental configuration used.

5. Experimental Results and Discussion

This section describes the experiments with the stated AI-driven surveillance framework and its ability to use a common data preparation pipeline with hybrid dataset construction, explaining the usage of YOLOv8 training complemented by diverse augmentation methods for dataset expansion as well as SMOTE-based balancing of datasets. The obtained results were then reported in comparison with some previous YOLO-based surveillance research on this dataset.

5.1 Overall Detection Performance

To assess the efficiency of the suggested framework, we utilized a hybrid dataset described in Section 4. This is the basis for evaluating Precision, Recall, F1-score, mAP@0.5, mAP@0.5:0.95, AUC-ROC, and execution time. An overview of global detection performance by the proposed framework is listed in Table 7.

Table (7): Total Detection Efficiency of the Suggested Framework.

Metric	Value
Precision	0.94
Recall	0.96
F1-score	0.95
mAP@0.5	0.94
mAP@0.5:0.95	0.71
AUC-ROC	0.99
Execution Time	0.35 s/image

The detection performance of the proposed framework is extremely competitive under the adopted experimental configuration in Table 7. The model has shown good capability to detect Security empowerment infrastructure-related objects while minimizing false positives from missing detections, which is reflected by the Precision (0.94), Recall (0.96), and F1-score (0.95) of this category. The detector received an mAP@0.5 of 0.94 and an mAP@0. 5:0.95 of 0.71 for this difficulty level as well, confirming that the detection and localization performance were strong at all IoU thresholds. Moreover, an AUC-ROC of 0.99 shows that threat and non-threat patterns are able to discriminate from each other pretty closely. In addition, the average speed of 0.35 s/image indicates computational efficiency for real-time surveillance applications.

The same behaviour is noted in previous work, indicating that YOLOv8 performance is achieved on par with or better than real-time [10], with our proposed framework being able to localize the more challenging-to-detect target while still keeping detection reliable.

5.2 Comparative and Ablation Analysis

In order to analyse the contribution of each component that we propose in this framework, we perform an ablation study by gradually adding the hybrid dataset, data augmentation, and SMOTE-based dataset balancing into the YOLOv8 detection pipeline. Table 8 displays the outcomes of these experiments.

Table (8): Ablation Analysis of the Suggested Structure.

Experiment	Precision	Recall	F1-score	mAP@0.5	mAP@0.5:0.95
YOLOv8 only	0.87	0.84	0.85	0.88	0.63
YOLOv8 + Hybrid Dataset	0.90	0.89	0.90	0.90	0.66
YOLOv8 + Hybrid Dataset + Data Augmentation	0.92	0.93	0.93	0.92	0.69
YOLOv8 + Hybrid Dataset + Data Augmentation + SMOTE	0.94	0.96	0.95	0.94	0.71

Each of the components of the proposed framework contributed to the final prediction performance, as can be seen in Table 8. There, we found that the hybrid dataset improved Precision and Recall because of enhanced environmental diversity. Adding data augmentation improved robustness against changes in surveillance conditions even more. Eventually, the highest achieved detection performance was observed on a data-balanced dataset using the aforementioned SMOTE-based dataset balancing strategy, which helped enhance the representation of minority object classes in both datasets.
Comparative tests were also carried out using the same experimental setup as those of two popular YOLO-based object identification models, YOLOv5 and YOLOV7, in order to further confirm the efficacy of the suggested framework [3]. The models are evaluated in a standardized experimental environment (common dataset, training/testing division, evaluation metrics, and hardware environment). A concise summary of these is shown in Table 9, indicating the relative performance.

Table (9): Comparing the Proposed Framework's Performance with New YOLO-Based Object Detection Models.

Model	Precision	Recall	F1-score	mAP@0.5	mAP@0.5:0.95	Inference Time (s/image)
YOLOv5	0.91	0.93	0.92	0.91	0.67	0.39
YOLOv7	0.92	0.94	0.93	0.92	0.69	0.37
Proposed Framework (YOLOv8 + Unified Data Preparation)	0.94	0.96	0.95	0.94	0.71	0.35

The individual performance reports corresponding to each metric are shown in Table 9, which illustrates the best results produced by the proposed framework for all metrics. Compared with YOLOv5 and YOLOv7, the proposed framework obtained the highest Precision (0.94), Recall (0.96), F1-score (0.95), mAP@0.5 (0.94), and mAP@0.5:0.95 (0.71). Moreover, getting close to the 0.35 s/image lower bound for inference time shows how our integrated data-preparation pipeline actually achieves accuracy and efficiency improvement simultaneously. While the proposed mechanism beats other recent YOLO-based architectures in the same learning task, confirming that those achievements are reproducible across repeated executions of experiments is equally important. To ensure the appropriateness of the proposed framework, five separate experiments with different random initialization seeds were performed on the same dataset, training/testing split, and experimental conditions. Summary of results in Table 10.

Table (10): Stability Analysis over Five Independent Experimental Runs.

Run	Precision	Recall	F1-score	mAP@0.5	mAP@0.5:0.95
1	0.94	0.96	0.95	0.94	0.71
2	0.93	0.95	0.94	0.93	0.70
3	0.94	0.96	0.95	0.94	0.71
4	0.95	0.97	0.96	0.94	0.72
5	0.94	0.96	0.95	0.94	0.71
Mean ± SD	0.940 ± 0.007	0.960 ± 0.007	0.950 ± 0.007	0.938 ± 0.004	0.710 ± 0.007

Table 10 illustrates that the proposed framework produced very coordinated results throughout the five runs. For Precision, Recall, F1-score, mAP@0.5, and mAP@0.5:0.95, the standard deviation remained extremely low, indicating that the observed performance is repeatable and not based on a single trial run. The framework proposed here is confirmed to be robust against these results.

5.3 Performance Improvement after Applying the Proposed Integrated Data Preparation Pipeline

To evaluate the efficacy of our proposed unified data preparation technique, we evaluated a YOLOv8 detector following hybrid dataset synthesis with and without data augmentation and SMOTE-based dataset balancing. The comparative results are shown in Table 11.

Table (11): Detection Performance Before and After Applying the Proposed Unified Data Preparation Pipeline.

Metric	Before	After
Precision	0.87	0.94
Recall	0.84	0.96
F1-score	0.85	0.95
mAP@0.5	0.88	0.94
mAP@0.5:0.95	0.63	0.71
AUC-ROC	0.82	0.99
Execution Time (s/image)	0.55	0.35

All assessment metrics on data preparation processes were improved by the approach presented in this article (see Table 11). Precision increased from 0.87 to 0.94, recall from 0.84 to 0.96, and F1-score from 0.85 to 0.95, indicating a more favourable ratio of false positives to false negatives. In addition, mAP@0.5 increased from 0.88 to 0.94 and mAP@0.5:0.95 from 0.63 to 0.71, indicating an improvement in object localization performance of the same size across different IoU thresholds. More importantly, AUC-ROC increased significantly (from 0.82 to 0.99) and average execution time requirement decreased (from 0.55 to 0.35 s/image), showing that detection performance was improved without a computational cost penalty.

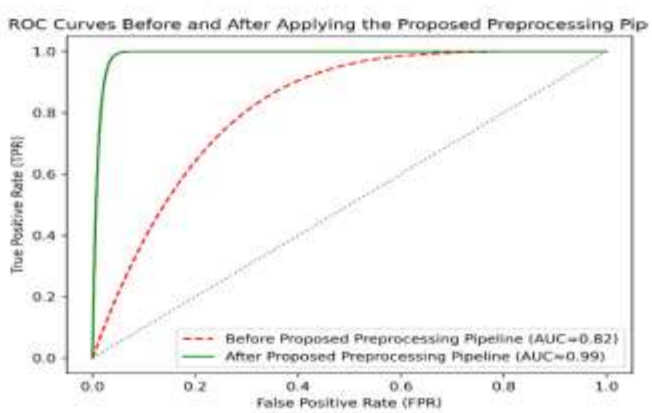

Figure (2): ROC curves prior to and following the implementation of the suggested unified data preparation process.

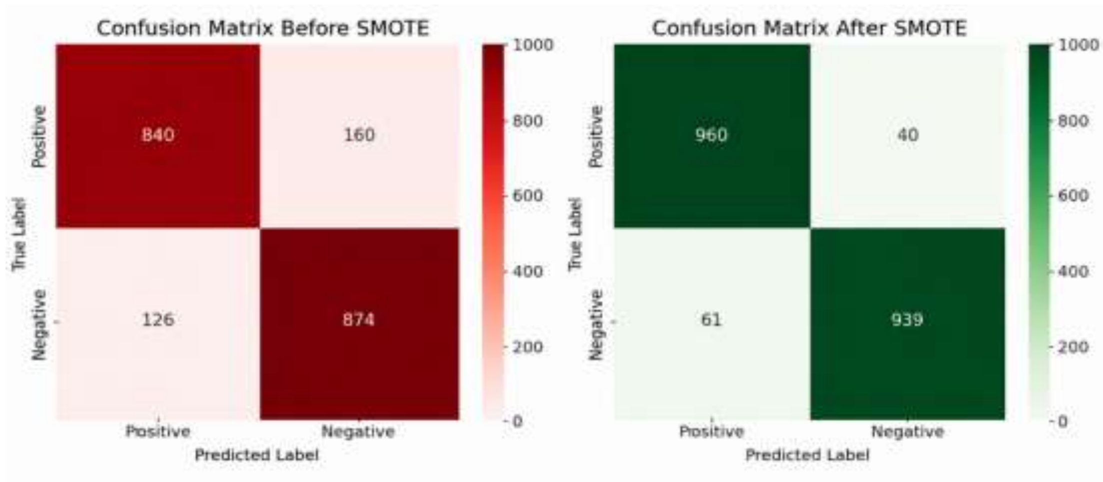

Figure (3): Confusion Matrices Prior to and Following the Proposed Unified Data Preparation Process.

Before summing patients to aggregate level data across all studies, ROC curves (presented in figure 2) show that the unified data preparation pipeline proposed significantly improves the models performance on differentiating cases with an AUC-ROC improvement from 0.82 to 0.99 This further supports the finding of improved classification accuracy and general detection performance from confusion matrices (shown in Figure 3) which also show a major decrease in false positive and false negative occurrences.

These results confirm the effect of data augmentation, balancing, and increasing diversity of datasets in producing more robust models and better modelling of minority classes [6,14]. As the proposed data preparation workflow implements hybrid dataset construction, image preprocessing, data augmentation, dataset splitting, and SMOTE-based dataset balancing only applied to the training subset (in separate steps), experimental results quantify a combined effect of these complementary components rather than the contribution of any particular technique.

5.4 Computational Efficiency Analysis

Because these threat detection systems operate in real-time, computational efficiency is a must-have. In this manner, under the adopted experimental configuration for the proposed framework, its execution time was evaluated and compared to its baseline implementation. Figure 4 shows the difference in the average execution time before and after using the proposed unified data preparation pipeline.

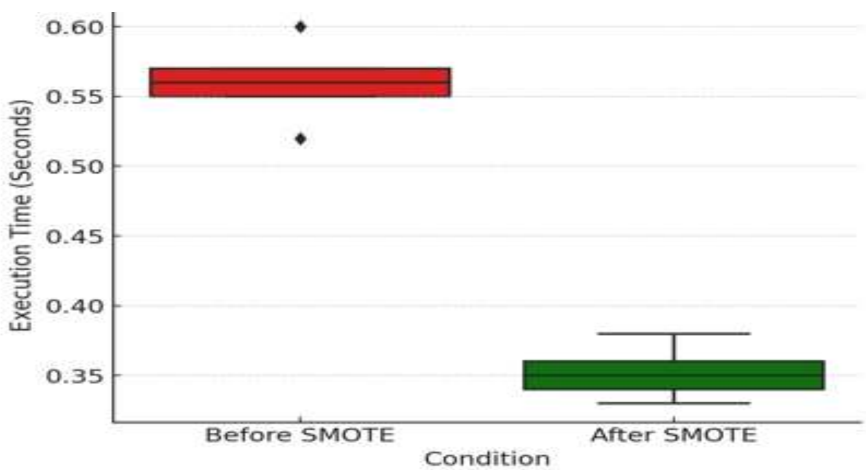

Figure (4): Average Execution Time Before and After Applying the Proposed Unified Data Preparation Pipeline.

As shown in Figure 4, the suggested framework obtained an average execution time of 0.35 s/image and a baseline configuration of 0.55 s/image. The lower detection performance of the baseline data processing pipeline verified that our multiscale unified data preparation pipeline effectively served to improve detection performance while still remaining computationally fast enough for real-time surveillance applications. The results demonstrate that the proposed framework provides a good trade-off between detection accuracy and inference speed under the selected experimental design.

Note that the suggested data preparation methodology included hybrid dataset building, picture preprocessing, data augmentation, dataset splitting, and SMOTE-based dataset balancing applied only to the training subset. Even while the ablation study outlined in Section 5.2 tested each component's contribution to the detection performance, computational efficiency was only evaluated for the baseline implementation and the complete proposed framework.

5.5 Comparison with Previous Studies

To further assess the effectiveness of the proposed framework, qualitative comparisons with standard YOLO-based monitoring studies were conducted. The findings are in line with other research showing YOLOv8's efficacy for intelligent monitoring [10,12]. Different from the majority of representative YOLO-based surveillance studies [3,11], which primarily focus on detector architecture or particular surveillance scenarios, a holistic YOLOv8-based detection pipeline is proposed in this framework by fusing hybrid dataset construction within a unified data augmentation and SMOTE-based dataset balancing scheme. The comparison is qualitative and not a strict benchmark due to different datasets, hardware platforms, and evaluation protocols used by the studies being compared. Indeed, the results prove that if we perform all stages of the data preparation pipeline with a minimum interval between them, it is possible to achieve a relatively high detection performance at lower computational costs: this allows us to consider such an approach for implementation in intelligent real-time surveillance systems.

6. Conclusion

This work proposed an AI entity surveillance framework driven by real-time security threat detection based on the YOLOv8 object detection model. The presented data preparation workflow is outlined from hybrid dataset construction, image preprocessing, data augmentation and splitting, followed by an SMOTE-based dataset balancing strategy (applied exclusively on the training subset in order to guarantee computational efficiency) as an integrated approach to optimize detection performance. To enhance environmental diversity and increase model generalization capability, hybrid datasets that combine the CCTV Surveillance Image Dataset and the Microsoft COCO dataset are used here as the input shadow dataset. Precision of 0.94, Recall of 0.96, F1-score of 0.95, mAP@0.5 of 0.94, mAP@0.5:0.95 of 0.71, AUC-ROC of 0.99, and an average execution time of 0.35 s/image were all attained using the suggested framework, according to experimental data. Through comparative experiments, ablation analysis, and repeated runs of the experiments, it was shown that the common data preparation pipeline accounts for achieving improved robustness in detection performance within the configurations of the experiment set-up being adopted.

In contrast to the existing surveillance research typically proposing new detector architecture strategies, our work proposes a framework that integrates complementary data preparation techniques into a standardized YOLOv8-based detection pipeline within the context of our sport surveillance studies. The results show that the proposed fusion strategy can enhance detection robustness while maintaining the efficiency required for real-time intelligent surveillance.

The results of this research must be considered in light of the limits imposed by the experimental setup adopted. The benchmark, however, only focused on image-based datasets across three object categories and performed no temporal analysis of any video-based models nor statistical significance testing. Even though multiple experimental runs as well as comparative evaluations of the proposed framework over representative YOLO based detectors have been performed, more validation (for example, using larger datasets, state-of-the-art detection models, and statistical significance analysis) with appropriate inclusion of experiments will increase the generalizability power of the proposed framework.

7. Future Work

By using this framework, in our future work we will build on video-based temporal modelling and multi-object tracking to leverage detection stability across continuous streams of surveillance. It will also be verified against more challenging real-world situations, including adverse weather, nighttime scenes, dense crowds, heavy occlusion, and motion blur. In addition, future work should benchmark the proposed framework against other YOLO variants released after YOLO4 and other transformer-based object detectors while performing statistical significance tests on larger and heterogeneous datasets. In the end, by applying the system to images with information from environmental sensors, we can produce stronger integrated item monitoring plans that have application significance in real-life security environments.

Conflict of Interest: The authors declare that there are no conflicts of interest associated with this research project. We have no financial or personal relationships that could potentially bias our work or influence the interpretation of the results.

References

[1] O. El Ghati, O. Alaoui-Fdili, O. Chahbouni, N. Alioua, and W. Bouarifi, “Artificial intelligence-powered visual Internet of Things in smart cities: A comprehensive review,” *Sustainable Computing: Informatics and Systems*, vol. 43, p. 101004, 2024. https://doi.org/10.1016/j.suscom.2024.101004

[2] J. Liang, “A review of the development of YOLO object detection algorithm,” *Applied and Computational Engineering*, vol. 71, pp. 39-46, 2024. https://doi.org/10.54254/2755-2721/71/20241642

[3] J. Terven and D. Cordova-Esparza, “A comprehensive review of YOLO: From YOLOv1 to YOLOv8 and beyond,” *Machine Learning and Knowledge Extraction*, vol. 5, no. 4, pp. 1680-1716, 2023. https://doi.org/10.3390/make5040083

[4] Z. Jingchun, E. S. Goh, and M. S. Sunar, “Low-light image enhancement: A comprehensive review on methods, datasets and evaluation metrics,” *Journal of King Saud University – Computer and Information Sciences*, vol. 36, no. 10, p. 102234, 2024. https://doi.org/10.1016/j.jksuci.2024.102234

[5] Y. Gong, Y. Hou, C. Zhang, and M. Jiang, “Beyond augmentation: Empowering model robustness under extreme capture environments,” *in Proceedings of the International Joint Conference on Neural Networks* (IJCNN 2024), 2024. https://doi.org/10.1109/IJCNN60899.2024.10650150

[6] W. Chen, K. Yang, Z. Yu, Y. Shi, and C. L. P. Chen, “A survey on imbalanced learning: Latest research, applications and future directions,” *Artificial Intelligence Review*, vol. 57, p. 137, 2024. https://doi.org/10.1007/s10462-024-10759-6

[7] P. Yadav, N. Gupta, and P. K. Sharma, “WeaponVision AI: A software for strengthening surveillance through deep learning in real-time automated weapon detection,” *International Journal of Information Technology*, 2025. https://doi.org/10.1007/s41870-024-02375-y

[8] I. Vijoli and C.-C. Vancea, “Towards improving real-time surveillance systems with enhancement techniques for deep learning models,” *in 2024 IEEE 20th International Conference on Intelligent Computer Communication and Processing (ICCP)*, pp. 1-8, 2024. https://doi.org/10.1109/ICCP63557.2024.10793047

[9] Y. Sun, Z. Sun, and W. Chen, “The evolution of object detection methods,” *Engineering Applications of Artificial Intelligence*, vol. 133, p. 108458, 2024. https://doi.org/10.1016/j.engappai.2024.108458

[10] M. S. K. Namana and B. U. Kumar, “An Efficient and Robust Night-Time Surveillance Object Detection System Using YOLOv8 and High-Performance Computing,” *International Journal of Safety and Security Engineering*, vol. 14, no. 6, pp. 1763-1773, 2024. https://doi.org/10.18280/ijsse.140611

[11] J. M. A. Dass, A. Suresh, K. Swathi, P. Shalini, and N. Y. Sinha, “Enhancing public safety through real-time weapon detection: A deep learning approach,” *in 2024 3rd International Conference for Advancement in Technology (ICONAT)*, pp. 1-4, 2024. https://doi.org/10.1109/ICONAT61936.2024.10775051

[12] M. Deshpande, A. Agarwal, and R. Kamathe, “Development of an energy-efficient CCTV camera system for real-time human detection using YOLOv8 model,” *Eastern-European Journal of Enterprise Technologies*, vol. 6, no. 9(132), pp. 47-59, 2024. https://doi.org/10.15587/1729-4061.2024.310401

[13] V. Dubey, B. Kaur, and P. Goel, “Enhancing accuracy through data augmentation using variational autoencoders in machine learning techniques,” *in 2024 Control Instrumentation System Conference (CISCON)*, pp. 1-6, 2024.

[14] J. H. Joloudari, A. Marefat, M. A. Nematollahi, S. S. Oyelere, and S. Hussain, “Effective class-imbalance learning based on SMOTE and convolutional neural networks,” *Applied Sciences*, vol. 13, no. 6, p. 4006, 2023. https://doi.org/10.3390/app13064006

[15] N. V. Chereddy and B. K. Bolla, “Evaluating the utility of GAN-generated synthetic tabular data for class balancing and low-resource settings,” *in Multi-disciplinary Trends in Artificial Intelligence, Lecture Notes in Artificial Intelligence*, vol. 14078, pp. 48-59, Springer, 2023.

[16] F. Luo, J. Ma, and G. T. S. Ho, “An instance-level data balancing method for object detection via contextual information alignment,” *Image and Vision Computing*, vol. 149, p. 105155, 2024. https://doi.org/10.1016/j.imavis.2024.105155

[17] S. U. A. Shovo et al., “Advancing low-light object detection with YOLO models: An empirical study and performance evaluation,” *Cognitive Computation and Systems*, 2024. https://doi.org/10.1049/ccs2.12114

[18] R. W. Liu, C. Han, and Y. Huang, “Low-light visibility enhancement for improving visual surveillance in intelligent transportation systems,” *IET Intelligent Transport Systems*, 2024. https://doi.org/10.1049/itr2.12534

[19] P. Shrivastav and A. K. T. Andu, “Developments in Deep Learning for Low Light Object Detection: A Comprehensive Review of YOLO and YOLOv8,” *Journal of Information Systems Engineering and Management*, vol. 10, no. 46s, 2025. https://doi.org/10.52783/jisem.v10i46s.9003

[20] P. Mittal, “A comprehensive survey of deep learning-based lightweight object detection models for edge devices,” *Artificial Intelligence Review*, vol. 57, p. 242, 2024. https://doi.org/10.1007/s10462-024-10877-1

[21] A. Tharwat, “Classification assessment methods,” *Applied Computing and Informatics*, vol. 17, no. 1, pp. 168-192, 2021. https://doi.org/10.1016/j.aci.2018.08.003