

# Fusion-Based small UAV Detection and Tracking: A Technical Report for IEEE VIP Cup 2025

Tamzeed Mahfuz, Ali Asif Khan, Nafis Nahian

Mustafa Muhaimin, Arnob Biswas, Zaki Rehnoom Unmona, Aritra Debnath

Asif Azad, A.K.M. Ashikur Rahman

Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology  
Dhaka, Bangladesh

**Abstract**—The rapid rise of drones in civilian and security settings demands intelligent, real-time solutions that can not only detect but also track aerial objects under the toughest environmental and sensor conditions. Addressing this urgent need, the IEEE VIP Cup 2025 challenge calls for robust, real-time systems capable of reliable drone detection and tracking in complex scenarios. In this work, we deliver a dynamic, end-to-end solution for the competition, harnessing the power of multi-modal sensor fusion with RGB and infrared (IR) imagery. Our approach combines state-of-the-art deep learning models and innovative fusion strategies—such as channel replacement and 4-channel concatenation—to boost detection accuracy and resilience to visual distortions. Beyond detection, we implement robust tracking using advanced algorithms and introduce motion filtering techniques that effectively suppress spurious detections, ensuring reliable object trajectories even in noisy or cluttered scenes. The system further tackles small object detection, class imbalance, and visual noise through targeted data augmentation, architectural enhancements, and post-processing. Achieving strong results in both detection and payload classification, our real-time pipeline demonstrates the transformative impact of modality fusion, tracking, and motion-aware filtering for next-generation drone surveillance in complex, real-world environments.

**Index Terms**—UAV detection, sensor fusion, RGB-IR, drone vs bird, distortion handling, real-time tracking.

## I. INTRODUCTION

Unmanned Aerial Vehicles (UAVs), commonly known as drones, have seen rapid adoption in both civilian and military domains, leading to increased concerns regarding airspace security, privacy, and public safety. The IEEE VIP Cup 2025 challenge addresses the urgent need for robust, real-time UAV detection and tracking systems capable of distinguishing drones from birds and other aerial objects, even under adverse environmental conditions and sensor distortions.

This paper presents a comprehensive technical report on our solution for the IEEE VIP Cup 2025, which leverages multi-modal sensor fusion using both RGB and infrared (IR) imagery. Our approach integrates state-of-the-art deep learning models, advanced data augmentation, and fusion strategies to achieve high-accuracy detection and tracking of UAVs. We systematically evaluate single-modality and fusion-based models, address the challenges of small object detection, and implement robust distortion handling techniques. Furthermore, we demonstrate real-time tracking and payload classification

capabilities, providing a complete pipeline suitable for practical deployment. Our results highlight the effectiveness of RGB-IR fusion and targeted optimizations in advancing the state of UAV surveillance.

## II. PROBLEM DESCRIPTION

**RGB and IR-based Object Detection:** Deep learning models such as YOLO, Faster R-CNN, and DETR have achieved state-of-the-art results in object detection using visible spectrum (RGB) imagery. However, IR imaging provides complementary information, especially in low-light or camouflaged scenarios. Recent works have explored using IR or multi-spectral data to improve detection robustness in adverse conditions.

**Sensor Fusion Strategies:** Sensor fusion combines information from multiple modalities to enhance detection accuracy and robustness. Early, mid, and late fusion approaches have been proposed, including channel concatenation, feature-level fusion, and decision-level fusion. Channel replacement (e.g., substituting the blue channel with IR) and adversarial fusion (e.g., TarDAL) have shown promise in recent literature.

**Tracking Methods for Aerial Objects:** Multi-object tracking algorithms such as SORT, DeepSORT, and ByteTrack are widely used for real-time tracking in video streams. These methods rely on bounding box association, motion prediction, and appearance features to maintain object identities across frames. Tracking small, fast-moving aerial objects remains a challenging problem, especially with limited annotated data.

**Vision under Distortions:** Robustness to noise, blur, and environmental distortions is critical for real-world deployment. Techniques such as data augmentation, denoising filters (median, bilateral, Restormer), and temporal filtering have been explored to mitigate the impact of distortions. Fusion of RGB and IR data further enhances resilience to challenging conditions. The IEEE VIP Cup 2025 competition challenges participants to develop a robust, real-time system for detecting and tracking UAVs (drones) using both RGB and infrared (IR) video streams. The primary objectives are:

- **Drone vs Bird Classification:** Accurately distinguish UAVs from birds and other aerial objects, minimizing false positives and negatives.

- **Distortion-Robust Detection:** Maintain high detection accuracy under various visual distortions, such as noise, blur, and adverse weather or lighting conditions.
- **Direction-Aware Tracking:** Track detected UAVs across frames, estimate their trajectories, and maintain consistent object identities in real time.
- **Payload Detection:** Identify and classify payloads attached to UAVs as harmful or normal, supporting safety-critical applications.

The competition emphasizes real-time performance, robustness to environmental challenges, and the ability to generalize across different sensor modalities and scenarios.

#### A. Competition Tasks

The competition requires participants to develop solutions for real-time drone and payload detection and tracking using RGB and IR datasets. The main tasks are:

- **Drone Detection:** Detect and classify drones versus other aerial entities (e.g., birds) in real-time under various environmental and distortion scenarios.
- **Drone Tracking:** Track the trajectory of drones across video frames, determine their motion relative to the camera (approaching or receding), and maintain tracking continuity under challenging conditions.
- **Payload Identification:** Detect and classify payloads carried by drones, distinguishing between harmful and normal payloads using fused RGB and IR data.

#### B. Dataset Overview

The provided datasets include both RGB and IR images and videos, annotated for drone and payload detection and tracking. Key characteristics include:

- **Drone Detection and Tracking:** 45,000 IR-RGB image pairs for training, 6,500 images and 40 video sequences for validation, and 13,000 images and 25 video sequences for testing. Videos are 10 seconds long at 30 fps, with a resolution of 320x256 pixels. Scenarios include single/multiple classes per frame, objects far from the FoV, swarms, and extreme environmental conditions (e.g., hilly regions, fog, thick forest cover).
- **Payload Identification:** 25,000 IR-RGB image pairs for training, 4,000 images for validation, and 8,000 images for testing. The dataset includes similar environmental and distortion scenarios as the detection dataset.
- **Distortions:** Speckle noise, salt and pepper noise, Gaussian blur, uneven illumination, motion blur, camera instability, and AWGN, with varying intensities.

#### C. Evaluation Protocols

**Drone Detection:** Evaluated on accuracy, F1 score, precision, recall, and robustness under adverse conditions (e.g., low light, fog, distortions). Real-time performance is assessed by displaying detection confidence on test videos and measuring inference speed (fps) and latency. Results from fusion models are compared to single-modality models.

**Drone Tracking:** Assessed on tracking consistency (missed frames < 15), spatial accuracy (IoU between predicted and ground truth bounding boxes), and ability to determine drone motion direction. Robustness is tested under environmental and distortion scenarios, with real-time operational metrics reported.

**Payload Detection:** Evaluated on accuracy, F1 score, precision, recall, and mean average precision (mAP). The system is tested on a dedicated, annotated dataset under challenging conditions. The benefits of RGB-IR fusion are highlighted, and real-time reporting of detection confidence is required.

Participants must submit readable, well-documented code (preferably in Python), a demo for test video inference, and report system specifications and inference times. The solution must use the provided RGB and IR datasets to train three models: one on RGB, one on IR, and one on fused data.

### III. BACKGROUND

Unmanned Aerial Vehicles (UAVs), commonly referred to as drones, have become increasingly prevalent due to their wide-ranging applications in surveillance, delivery, agriculture, logistics, disaster response, and military operations. However, this proliferation has introduced significant challenges, including unauthorized aerial activity, potential threats from payload delivery systems, and heightened security and privacy concerns. These issues necessitate robust detection and tracking of drones, as well as the identification of their payloads, to ensure safety and security, particularly in sensitive or restricted areas.

Traditional vision-based approaches for drone detection primarily rely on RGB images, which are often limited by environmental factors such as low light, fog, or glare. Infrared (IR) imaging, on the other hand, captures thermal signatures and enables robust detection in challenging conditions, such as nighttime or occlusion, even when drones are far from the field of view (FoV) of surveillance cameras. While IR imaging can outperform RGB in certain scenarios, it lacks the spatial and textural richness provided by RGB images. Therefore, fusing RGB and IR modalities is essential to leverage their complementary strengths for more reliable detection.

In addition to drone detection, identifying payloads carried by drones is critical due to the risks associated with unauthorized or malicious payload delivery. Payloads may include hazardous materials, surveillance equipment, or contraband, posing threats to public safety and privacy. Real-time identification of payloads enables proactive risk mitigation, making it a vital aspect of drone monitoring systems. However, payloads vary in size, shape, and thermal properties, making single-modality detection unreliable. RGB imaging provides visual cues for recognizing payload shapes and textures, while IR imaging highlights heat signatures, especially for heat-generating payloads. Thus, a fusion-based approach is crucial for accurate and reliable payload detection.

## IV. METHODOLOGY

### A. System Architecture Overview

Our solution uses YOLOv5, a deep object detection model trained on the official dataset. YOLOv5 follows a modular architecture comprising three main components: Backbone, Neck, and Head. The Backbone uses CSPDarknet to extract rich spatial features from input images efficiently. The Neck employs a combination of Feature Pyramid Network (FPN) and Path Aggregation Network (PANet) structures to enhance multi-scale feature fusion, crucial for detecting objects of varying sizes. Finally, the Head predicts bounding boxes, objectness scores, and class probabilities using anchor-based detection at multiple scales. This streamlined, end-to-end design enables YOLOv5 to achieve real-time performance with competitive accuracy across various detection tasks.

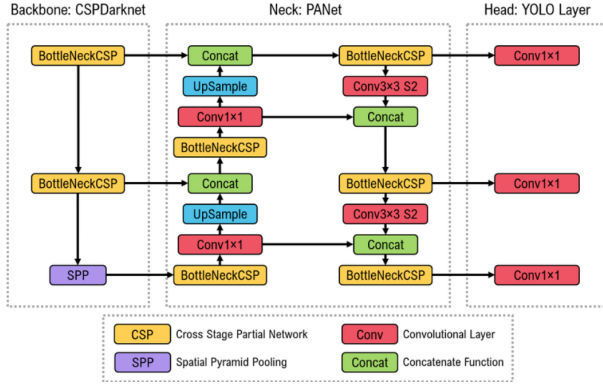


Fig. 1: YOLOv5 architecture

### B. Dataset Preparation and Splitting Strategy

The official IEEE VIP Cup 2025 dataset comprises image frames extracted from aerial surveillance videos, captured in both RGB and IR formats. To facilitate robust training and ensure generalization, we split the dataset into training, validation, and test sets with a ratio of 80:10:10. Specifically, the training set contained 45,700 frames, the validation set had 5,318 frames, and the test set included 6,400 frames. Importantly, to prevent data leakage, we performed a video-wise stratified split—ensuring that frames from the same video sequence remained within a single split. This process was conducted separately for RGB and IR datasets to maintain modality independence during the initial phases of experimentation.

TABLE II: Bird and Drone Instances Count in Different Dataset Splits

Dataset Type	Train		Validation		Test	
	Bird	Drone	Bird	Drone	Bird	Drone
RGB	17068	28794	1896	3422	2548	4454
IR	17068	28794	1896	3422	2548	4454
Fusion (RGB+IR, 4-chan)	34136	57588	3792	6844	5096	8908
RG-IR (Blue → IR)	17068	28794	1896	3422	2548	4454
Bird Augmented (Train only)	33360	28794	—	—	—	—
Both Augmented (Train only)	33402	33402	—	—	—	—
Shuffled Fusion (RGB ∪ IR)	34136	57588	3792	6844	5096	8908

### C. Model Selection and Single-Modality Training

We began by benchmarking a variety of object detection architectures on both RGB and IR datasets independently. Among the models evaluated were several from the YOLO family (YOLOv5, YOLOv8, YOLOv11, and the most recent YOLOv12), as well as Faster R-CNN and detection transformer-based architectures including RF-DETR and RT-DETR.

Faster R-CNN demonstrated reasonable detection accuracy, but its high computational cost made it unsuitable for real-time applications. Detection transformer models (e.g., RF-DETR, RT-DETRv2 [1]) provided a favorable trade-off between performance and speed, offering decent results with moderate inference time. However, YOLO-based models consistently outperformed the others in terms of both detection accuracy and inference speed. As a result, we narrowed our focus to the YOLO family, and in particular, YOLOv12, which initially delivered the best performance.

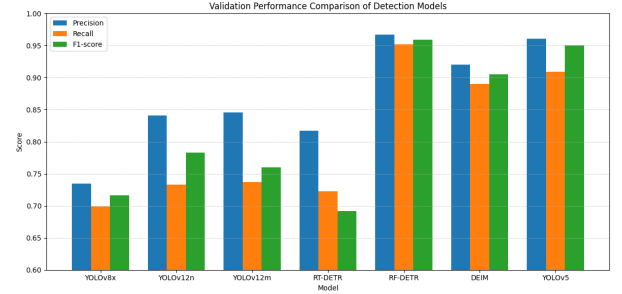


Fig. 2: Comparison of different experimental models

These early experiments were conducted separately on the RGB and IR modalities, allowing us to evaluate the capacity of each model to learn discriminative features from a single sensor type. While YOLOv12 showed strong results, we recognized the potential benefits of leveraging both RGB and IR information through fusion.

### D. Revisiting Anchor-Based Models for Small Object Detection

After extensive experimentation with fusion models, we revisited the problem from a single-modality perspective with focus on improving detection accuracy for small aerial targets. We observed that more recent YOLO versions, including YOLOv12 and YOLOv8, emphasize generalization and objectness-based detection without relying on anchor boxes. While effective in many general-purpose scenarios, these anchor-free methods underperformed on our UAV dataset, which predominantly features small drones that require high localization precision.

Motivated by this insight, we explored the older YOLOv5 architecture, which utilizes anchor boxes tailored to different object scales. To our surprise, YOLOv5 significantly outperformed all other models, including more modern YOLO variants, particularly in detecting small, fast-moving drones. Table IX displays the results. This suggests that anchor-based

TABLE I: YOLO Dataset Splits Across Different Modalities and Strategies

Dataset Type	Train	Test	Validation	Description
RGB	45,700	6,400	5,318	3-channel visible light images.
IR	45,700	6,400	5,318	1-channel infrared images.
Fusion (RGB+IR, 4-chan)	45,700	6,400	5,318	4-channel input formed by stacking RGB and IR.
RG-IR (Blue $\rightarrow$ IR)	45,700	6,400	5,318	Blue channel of RGB replaced with IR for 3-channel compatibility.
Bird Augmented (Train only)	62,154	6,400	5,318	Train set augmented with flips, blurs, and distortions in both RGB and IR.
Bird and Drone Both Augmented (Train only)	66,804	6,400	5,318	Train set augmented with flips, blurs, and distortions in both RGB and IR.
Shuffled Fusion (RGB $\cup$ IR)	91,400	12,800	10,636	RGB and IR samples combined and shuffled within each split.

models retain an advantage in specialized tasks involving small object detection, making YOLOv5 the most suitable backbone for our final pipeline.

#### E. Exploration of Small Object Detection Enhancements

Given the challenge of accurately detecting small objects such as drones, we explored enhancements inspired by recent advances in small object detection research. One such method involved incorporating attention mechanisms into our backbone model. Specifically, we implemented the Channel Block Attention Module (CBAM) [2], which has shown promising results in other works involving small and occluded object detection.

CBAM integrates two sequential attention operations: channel attention and spatial attention. The channel attention mechanism emphasizes "what" is important by weighing feature maps based on their channel-wise significance. This is followed by spatial attention, which highlights "where" important features are located by learning a spatial mask. The original CBAM implementation had been used successfully on YOLOv8 to improve object detection under dense and cluttered environments. We adopted a similar strategy by integrating CBAM into the YOLOv8 architecture and retraining on our UAV dataset.

Despite its theoretical appeal, this modification did not lead to performance improvements on our task, as shown in Table XII. We hypothesize that YOLOv8's original architecture is already optimized for general-purpose object detection, and the additional attention mechanisms may have introduced unnecessary complexity without effectively capturing the very small-scale features present in our aerial data.

#### F. Addressing Class Imbalance through Data Augmentation

Upon reviewing the dataset distribution, we observed a noticeable class imbalance, with the ratio of drone to bird instances being approximately 2:1. This skew potentially biased the model during training, causing poorer performance on the underrepresented bird class—often leading to missed detections or misclassifications.

To mitigate this, we applied targeted data augmentation techniques to rebalance the dataset, particularly increasing diversity in bird-class samples while also enriching the overall training data. The following augmentation strategies were employed:

- **Random Crop:** Cropping images at random locations to force the model to learn contextual cues from partial views and improve robustness to occlusion.
- **Horizontal Flip:** Reflecting images horizontally with a 50% probability, useful in aerial scenarios where object orientation is variable.
- **Random Brightness and Contrast Adjustment:** Simulating lighting variations by randomly modifying brightness and contrast, thereby increasing resilience to illumination changes.
- **CLAHE (Contrast Limited Adaptive Histogram Equalization):** An adaptive enhancement technique that improves local contrast in low-visibility conditions by limiting the amplification of noise. CLAHE was particularly helpful in emphasizing faint object boundaries in both RGB and IR images.

By augmenting the dataset with a more balanced distribution of drone and bird instances, and applying the above transformations, we were able to train a more equitable detection model, the results displayed in Table X. The improvements were particularly evident in the bird-class detection precision, which had previously lagged behind due to the limited and under-diverse training samples.

#### G. Fusion Techniques for Dual-Modality Learning

To explore the potential of RGB-IR fusion, we first attempted a sequential fine-tuning approach—taking a YOLOv12 model pretrained on the RGB dataset and continuing its training on the IR dataset. Contrary to our expectations, this led to a sharp decline in the model's performance on RGB data, suggesting catastrophic forgetting. The model essentially adapted to the IR modality at the expense of previously learned RGB representations, indicating that sequential fine-tuning across modalities was ineffective for our use case.

We then turned our attention to leveraging the inherent pairing in the dataset, where each RGB image has a corresponding IR counterpart. Our initial fusion strategy involved concatenating the 1-channel IR image with the 3-channel RGB image to form a 4-channel composite image. The YOLOv12 model was modified accordingly to accept this 4-channel input. This approach resulted in decent performance (Table VIII), demonstrating the viability of joint modality learning.

Building on this, we experimented with a channel-replacement strategy in which we replaced the blue channel of the RGB image with the IR channel, thereby maintaining a 3-



Fig. 3: Image after replacing the Blue channel with IR channel

channel input structure compatible with the original YOLOv12 architecture. The reasoning behind selecting this particular channel was that, most of the small UAV detection scenarios involved the blue sky background as a large portion, not containing much information. Thus, we replaced this hollow channel with the information-rich IR channel. This simple yet effective method yielded improved performance over the 4-channel approach, suggesting that implicit fusion through channel substitution allowed the network to better integrate modality-specific cues. Results are displayed in Table VII.

#### H. Target-Aware Adversarial Fusion: TarDAL

We further explored an advanced fusion strategy based on the TarDAL (Target-aware Dual Adversarial Learning) [3] method, which employs a generator-discriminator framework to learn joint RGB-IR representations. In this setup, a generator network synthesizes fused representations from RGB and IR inputs, while two discriminators evaluate their modality-specific realism and semantic alignment. This method produced the most accurate detection results, significantly outperforming earlier fusion strategies, as shown in Table XI. However, the computational cost introduced by TarDAL’s generator-discriminator loop was prohibitively high for real-time deployment. Given that the VIP Cup mandates real-time operation, we opted not to integrate this method into our final system.

#### I. Cross-Modality Generalization through Mixed Training

A critical challenge in multimodal systems is ensuring generalization across different sensor types—especially in scenarios where only one modality may be available at inference time. Initially, our fusion experiments relied on paired RGB-IR images for training, but we observed that the resulting models failed to generalize when tested on single-modality inputs (e.g., only RGB or only IR).

To address this, we adopted a modality-agnostic training strategy. We combined the RGB and IR image datasets into a single unified pool and shuffled them uniformly. The resulting dataset contained both RGB and IR samples in equal proportion. We then trained a YOLOv5 model on this mixed-modality dataset.

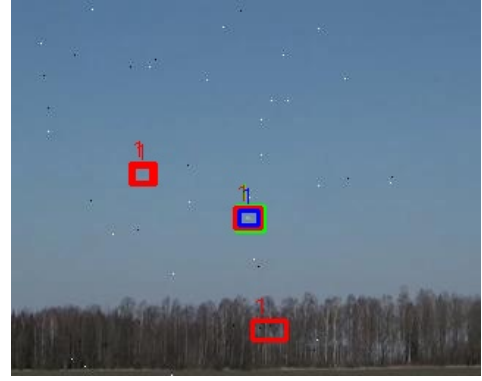


Fig. 4: Result after filtering, Green box for ground truth, Red boxes for noisy predictions, Blue box for final filtered prediction

This simple modification encouraged the model to learn representations that were effective across both RGB and IR inputs (Table XVI). The trained model generalized well to either modality during inference, although it did not achieve the same level of accuracy as some of our paired-fusion models.

#### J. Denoising and Noise-Robust Detection

As the competition’s problem statement emphasized the presence of various visual distortions and adverse environmental conditions, we investigated the effect of image noise on detection accuracy. The dataset exhibited multiple noise types including speckle noise, salt-and-pepper noise, Gaussian blur, motion blur, and uneven illumination.

To address this, we experimented with a range of image filtering techniques including:

- **Lee Filtering:** A noise-reducing filter particularly effective for speckle noise in SAR and IR imagery, which preserves edges by adapting the filter strength to local statistics.
- **Bilateral Filtering:** A non-linear technique that smooths images while preserving edges, combining spatial proximity and pixel intensity similarity in its weighting.
- **Median Filtering:** Especially effective against salt-and-pepper noise, this filter replaces each pixel with the median of its neighborhood, preserving edges while removing outliers.
- **Canny Edge Detection:** Used primarily to enhance object contours and suppress noise, providing strong gradient-based features to the detection model.

Despite the theoretical advantages of these preprocessing filters, none of them led to meaningful improvements in detection performance when evaluated on our validation set. In several cases, the filters even caused slight performance degradation, likely due to the loss of fine texture information critical for small object detection.

We also explored a deep learning-based denoising approach using **Restormer**—a state-of-the-art transformer architecture for image restoration tasks [4]. Restormer uses multi-Dconv

head transposed attention and gated feedforward layers to capture both local and long-range dependencies. It has shown strong performance in denoising, deraining, and motion deblurring benchmarks. However, even with Restormer’s advanced capabilities, the improvement in our UAV detection task was marginal at best (Table XIII) and came at the cost of significantly increased inference time, which again conflicted with our real-time constraints.

Upon closer inspection, we found that salt-and-pepper noise was particularly problematic—often manifesting as small, bright or dark dots that the detector would mistakenly classify as drones or birds. To address this without compromising the model pipeline, we developed a simple but effective post-processing technique. For every detected instance in a frame, we checked its temporal persistence across a window of surrounding frames (past and future). If a detection appeared only in a single frame and had no spatial or temporal continuity, it was likely a false positive due to noise and was subsequently filtered out. This temporal filtering mechanism significantly reduced spurious predictions and improved the system’s robustness to noisy frames, as displayed in Table XIV.

#### K. Payload Detection and Classification

The competition also involved a secondary task of detecting and classifying payloads as harmful or normal, attached to UAVs. In contrast to the main detection task, this dataset was comparatively simpler, with clearer object boundaries and larger object sizes. We trained a YOLOv12 model specifically on this payload dataset. Given the larger size of the payloads and minimal noise, the model achieved high detection accuracy without requiring architectural changes or preprocessing enhancements. Results are displayed in Table XV. Therefore, no additional optimization was pursued for this sub-task.

#### L. Object tracking with ByteTrack

We extended our detection pipeline with object tracking using ByteTrack [5], a lightweight and high-performance multi-object tracking algorithm. ByteTrack improves tracking robustness by associating both high-confidence and low-confidence detections across frames using a two-stage IoU-based matching strategy.

We integrated ByteTrack with the outputs of YOLOv5 to maintain consistent object identities over time. The tracking results were qualitatively reasonable, showing smooth ID assignments across frames. However, due to the lack of a provided tracking-labeled dataset from the organizers, we could not conduct quantitative evaluations. This component remains a demonstration of potential applicability for future work in motion-aware systems.

### V. EXPERIMENTAL SETUP

All experiments in this study were conducted on the Kaggle cloud platform. The hardware configuration provided by Kaggle includes an Intel Xeon CPU operating at 2.20 GHz with 4 virtual cores, 32 GB of system memory (RAM), and



Fig. 5: Drone tracking with ByteTrack

two NVIDIA T4 GPUs. Each T4 GPU is equipped with 2560 CUDA cores and 16 GB of video memory. The CUDA version available in the environment was 11.8.

We implemented and trained our models using the PyTorch deep learning framework. The training was conducted using a batch size of 32 for 25 epochs. We employed Stochastic Gradient Descent (SGD) as the optimizer with an initial learning rate of 0.01.

For model evaluation, we utilized several key metrics to assess the performance of our detection pipeline comprehensively. These metrics include:

- **Precision:** Measures the proportion of true positive drone detections among all positive detections.
- **Recall:** Measures the proportion of correctly detected drones out of all actual drone instances.
- **F1 Score:** Harmonic mean of precision and recall, providing a balance between the two.
- **mAP@0.5 (mean Average Precision at IoU threshold 0.5):** A widely used metric in object detection that summarizes the precision-recall curve.
- **Inference Time (ms):** Average time taken to process a single frame during inference.
- **Frames Per Second (FPS):** Indicates the real-time capability of the model, inversely related to inference time.

These metrics collectively capture both the accuracy and efficiency of our system, which are crucial for real-time UAV surveillance applications.

### VI. RESULTS AND DISCUSSION

In this section, we present a comparative evaluation of various object detection models across different input modalities (RGB, IR, and fused), fusion techniques, and dataset splits (train, validation, test). We also analyze the training progression and loss convergence trends.

TABLE III: Performance Comparison of Various Detection Models on Drone Dataset

Model	Split / Settings	Precision	Recall	F1-score
YOLOv8x	pre-trained	0.7347	0.6990	0.7164
YOLOv12n [6]	Test set	0.8990	0.6230	0.7360
	Validation set	0.8410	0.7330	0.7830
YOLOv12m [6]	Test set	0.9030	0.6560	0.7360
	Validation set	0.8460	0.7370	0.7600
RF-DETR [7]	Validation set	0.9670	0.9520	0.9590
	Test set	0.8050	0.7370	0.7700
DEIM [8]	Validation set	0.9200	0.8900	0.9050
RT-DETR [9]	Test set	0.8167	0.7230	0.6926

#### A. DETR Model Performance

Table IV shows that RT-DETRv2 [1] achieves solid performance on the test set, particularly in detecting birds with an F1 score of 0.83 and a respectable AP50 of 0.853. However, the AP75 and AP50–95 metrics indicate challenges in tighter localization and generalized detection. As shown in Table IV, RT-DETRv2 performs moderately well on the validation set. While the precision remains above 0.8, the recall drops slightly, particularly for the bird class. Additionally, due to the heavy architecture and high inference time, we opted out of using this model for further experimentation.

TABLE IV: RT-DETRv2 Performance on Validation vs. Test Sets

Split	Class	Precision	Recall	F1	AP50
Validation	Bird	0.831	0.635	0.720	0.745
	Drone	0.802	0.785	0.793	0.737
	Overall	0.816	0.710	0.759	0.741
Test	Bird	0.912	0.762	0.830	0.853
	Drone	0.873	0.840	0.856	0.805
	Overall	0.892	0.801	0.843	0.829

#### B. YOLOv12n: RGB and IR Results

1) *RGB Modality*: Tables V illustrate that YOLOv12n struggled on the RGB test set, particularly with drone recall. However, performance on the validation set was significantly better, indicating possible overfitting or variance in the test data.

TABLE V: RGB Test and Validation Metrics (YOLOv12n)

Split	Class	Precision	Recall	F1
Test	Bird	0.4968	0.2115	0.2967
	Drone	0.6125	0.0062	0.0123
	Overall	0.5547	0.1089	0.1820
Validation	Bird	0.8906	0.7365	0.8063
	Drone	0.8668	0.8660	0.8664
	Overall	0.8787	0.8013	0.8382

2) *IR Modality*: The IR results (Tables VI follow a similar trend to the RGB data, with test recall particularly low. Validation metrics again suggest reasonable performance.

TABLE VI: IR Test and Validation Metrics (YOLOv12n)

Split	Class	Precision	Recall	F1
Test	Bird	0.5761	0.1985	0.2952
	Drone	0.9355	0.0037	0.0074
	Overall	0.7558	0.1011	0.1783
Validation	Bird	0.8947	0.7966	0.8428
	Drone	0.8635	0.8650	0.8643
	Overall	0.8791	0.8308	0.8543

#### C. RGB Blue Channel Replaced by IR

Table VII shows the metrics for RGB with IR-in-Blue replacement across both validation and test sets.

TABLE VII: RGB with IR-in-Blue Replacement Test and Validation Set Metrics

Split	Class	Precision	Recall	F1	AP50
Validation	Bird	0.9984	0.9768	0.9875	0.9878
	Drone	0.8732	0.7729	0.8200	0.7808
	Overall	0.9358	0.8749	0.9024	0.8843
Test	Bird	0.9572	0.6236	0.7552	0.7881
	Drone	0.9239	0.8772	0.8999	0.8694
	Overall	0.9405	0.7504	0.8376	0.8287

#### D. 4-Channel Fusion Performance

Fusing RGB and IR via 4-channel concatenation produced consistently strong results, especially on the test set, as shown in Table VIII

TABLE VIII: 4-Channel Dataset – Test and Validation Metrics

Split	Class	Precision	Recall	F1	AP50
Validation	Bird	0.8935	0.5874	0.7088	0.7446
	Drone	0.8096	0.8027	0.8061	0.7358
	Overall	0.8515	0.6951	0.7507	0.7402
Test	Bird	0.9595	0.7779	0.8592	0.8676
	Drone	0.9170	0.9020	0.9094	0.8752
	Overall	0.9382	0.8399	0.8884	0.8714

#### E. YOLOv5m Baseline Results

YOLOv5m, despite its older architecture, achieved excellent detection results, particularly for drone instances, confirming its advantage in small object detection.

TABLE IX: YOLOv5m RGB – Test and Validation Set Metrics

Split	Class	Images	Instances	P	R	mAP50
Validation	All	6400	7002	0.921	0.732	0.803
	Bird	6400	2548	0.949	0.588	0.763
	Drone	6400	4454	0.894	0.876	0.842
Test	All	6400	7002	0.900	0.786	0.819
	Bird	6400	2548	0.916	0.682	0.797
	Drone	6400	4454	0.883	0.890	0.841

#### F. RGIR Dataset with Augmentation Results

We applied augmentation techniques to the RGIR dataset and observed notable improvements in validation metrics, particularly for the bird class, as shown in Table X. Precision

and recall remain high across both classes, and the mAP50–95 value improved compared to the non-augmented dataset.

TABLE X: RGIR Dataset with Augmentation  
Test and Validation Set Metrics

Split	Class	Images	Instances	P	R	mAP50
Validation	All	5318	5318	0.944	0.935	0.924
	Bird	5318	1896	0.979	0.963	0.976
	Drone	5318	3422	0.909	0.907	0.872
Test	All	6400	7002	0.835	0.727	0.742
	Bird	6400	2548	0.832	0.606	0.697
	Drone	6400	4454	0.838	0.848	0.788

n, test). We also analyze the training progression and loss convergence trends.

#### G. Tardal Fusion Performance at 0.3 Confidence Threshold

Table XI presents the performance of our Tardal fusion technique using a confidence threshold of 0.3. On the validation set, bird detection achieved an exceptionally high F1-score (0.9808), while drone detection remained competitive. However, test set results indicate a notable drop in bird recall (from 0.9816 to 0.6354), affecting overall F1. This drop suggests sensitivity to distribution shift or environmental variation between validation and test sets.

TABLE XI: Tardal Fusion Metrics at 0.3 Confidence (Validation vs. Test)

Split	Class	Precision	Recall	F1	AP50	AP75
Validation	Bird	0.9800	0.9816	0.9808	0.9889	0.4544
	Drone	0.8737	0.8863	0.8800	0.8485	0.1423
	Overall	0.9269	0.9339	—	0.9187	0.2983
Test	Bird	0.9131	0.6354	0.7494	0.7795	0.1658
	Drone	0.8506	0.8691	0.8597	0.8333	0.2048
	Overall	0.8819	0.7523	—	0.8064	0.1853

#### H. Small object detection enhancements

Table XII reports detection results for the CBAM-enhanced model, which integrates attention mechanisms into the convolutional backbone. This model demonstrates strong overall performance, achieving an F1-score of 0.9180 for birds and 0.8888 for drones. Notably, AP50 for both classes remains competitive, although tighter localization (as reflected by AP75 and AP50–95) still presents room for improvement.

TABLE XII: Detection Metrics for CBAM-Enhanced Model

Class	Precision	Recall	F1	AP50	AP75	AP50–95
Bird	0.9855	0.8592	0.9180	0.9214	0.3632	0.4566
Drone	0.8934	0.8843	0.8888	0.8640	0.1519	0.3109
Overall	0.9394	0.8717	—	0.8927	0.2575	0.3838

#### I. Impact of Denoising Methods: Restomer vs. Median Filtering

To assess the effect of image denoising on detection performance, we evaluated two methods—Restomer (a transformer-based restoration network) and traditional Median Filtering.

Table XIII summarizes the performance comparison on the RGB dataset.

Restomer yielded slightly better recall, especially for bird detection, while Median Filtering showed improved precision and higher AP50–95 for bird class, possibly due to its smoothing effect on sensor noise. Overall, both methods helped maintain high detection performance, with Restomer slightly outperforming in mAP50.

TABLE XIII: Denoising Methods RGB: Restomer and Median Filtering

Split	Class	P	R	mAP50	mAP50–95
Restomer	All	0.799	0.810	0.809	0.280
	Bird	0.726	0.794	0.799	0.315
	Drone	0.872	0.825	0.818	0.245
Median Filtering	All	0.816	0.736	0.794	0.312
	Bird	0.738	0.727	0.777	0.388
	Drone	0.893	0.744	0.811	0.235

#### J. Effect of Motion Filtering on Detection Performance

Motion filtering is applied to suppress spurious static detections and emphasize temporally consistent object movement. Table XIV compares detection metrics with and without motion filtering at a low confidence threshold of 0.01.

The results indicate that motion filtering slightly improves overall recall and mAP50 without sacrificing precision. Notably, bird recall improves from 0.951 to 0.975, leading to a small but meaningful gain in mAP50 from 0.985 to 0.989. These findings suggest that motion filtering is effective in refining detection outputs, particularly under low confidence settings.

TABLE XIV: Detection Metrics With and Without Motion Filtering  
(confidence threshold = 0.01)

Class	Images	No Filtering			With Filtering		
		P	R	mAP50	P	R	mAP50
All	5318	0.961	0.909	0.950	0.960	0.932	0.954
Bird	5318	0.989	0.951	0.985	0.989	0.975	0.989
Drone	5318	0.934	0.868	0.915	0.932	0.889	0.919

#### K. Payload Classification on YOLOv12n

To evaluate the payload classification capability of YOLOv12n, we trained the model on a custom-labeled dataset distinguishing between harmful and normal payloads. Table XV reports the validation set performance.

The results indicate that YOLOv12n is highly effective in classifying both categories with F1-scores above 0.92 and exceptionally high AP50 and AP75 scores. The harmful payload class, in particular, achieves perfect precision and recall, suggesting the model is well-calibrated and capable of handling safety-critical distinctions in payload types.

TABLE XV: YOLOv12n Payload Dataset – Validation Set Metrics

Class	Precision	Recall	F1	AP50	AP75	AP50-95
Harmful	0.9740	0.9740	0.9740	0.9863	0.9863	0.9764
Normal	0.9231	0.9231	0.9231	0.9463	0.9463	0.9278
Overall	0.9486	0.9486	—	0.9663	0.9663	0.9521

#### L. Combined RGB-IR Dataset Performance

To evaluate the effectiveness of training with a combined RGB and IR dataset (shuffled fusion of both modalities), we computed performance metrics on the mixed-modality validation set. As shown in Table XVI, the fused dataset led to a balanced detection performance for both bird and drone classes, demonstrating strong generalization across modalities.

TABLE XVI: Combined RGB-IR Validation Set Metrics

Class	Images	Instances	P	R	mAP50
All	10 636	10 636	0.908	0.864	0.865
Bird	10 636	3 792	0.916	0.853	0.868
Drone	10 636	6 844	0.899	0.874	0.863

#### M. Training and Validation Curves

For each modality/model (IR, RGB, RGIR, 4-channel, and payload), we plot the training and validation loss curves individually. These curves illustrate the convergence behavior and help identify overfitting or underfitting trends. Notably, the 4-channel and RGIR models demonstrate faster convergence and lower final loss, indicating more effective learning from fused modalities.

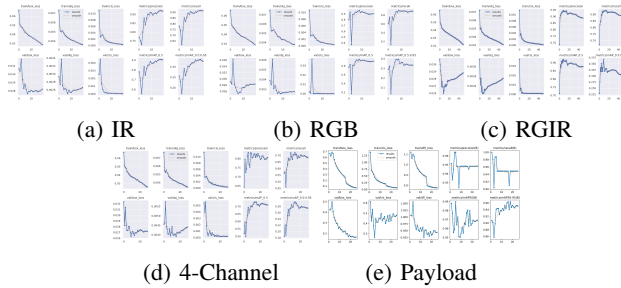


Fig. 6: Training and validation loss curves for all tasks.

#### N. F1 and Precision-Recall Curves

To further assess model performance, we plot the F1-score and precision-recall (PR) curves over epochs for each modality/model individually. These curves highlight the evolution of detection quality and the trade-off between precision and recall. The RGIR and 4-channel models consistently achieve higher F1 and PR values, confirming the benefit of modality fusion.

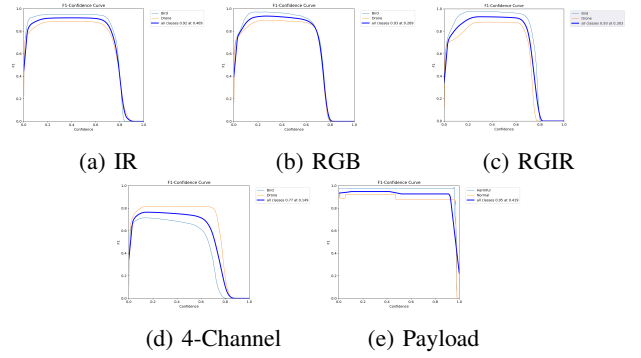


Fig. 7: F1-score curves for all tasks.

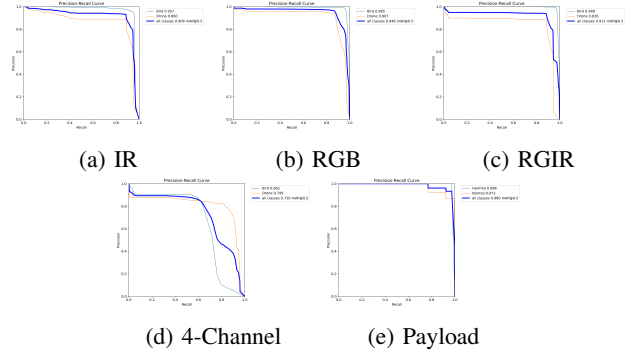


Fig. 8: Precision-Recall curves for all tasks.

#### O. Confusion Matrices

We present confusion matrices for each model and modality individually. These matrices provide insight into class-wise performance, particularly for bird vs. drone and for payload classification. The 4-channel and RGIR models show reduced misclassification rates, especially for the underrepresented bird class. For the payload task, the confusion matrix indicates near-perfect separation between harmful and normal payloads.

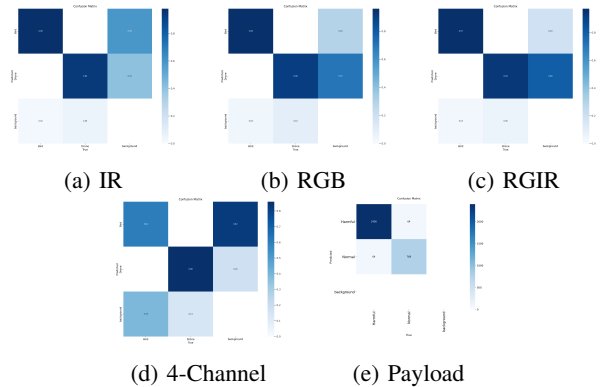


Fig. 9: Confusion matrices for all tasks.

## VII. CONCLUSION AND FUTURE WORK

In this work, we developed and evaluated a robust UAV detection and tracking system based on RGB-IR sensor fusion,

tailored for the IEEE VIP Cup 2025 challenge. Our experiments demonstrate that fusion-based models, particularly those leveraging both RGB and IR modalities, consistently outperform single-modality baselines in terms of detection accuracy, robustness to distortions, and generalization across diverse conditions. We addressed key challenges such as small object detection, class imbalance, and environmental noise through targeted data augmentation, architectural enhancements, and post-processing techniques.

Despite these advances, several limitations remain. Real-time deployment on edge devices is constrained by computational requirements, and performance may degrade under extreme weather or lighting conditions not represented in the training data. Additionally, the lack of large-scale annotated tracking datasets limited quantitative evaluation of tracking performance.

Future work will focus on expanding the system to incorporate additional sensor modalities (e.g., radar, acoustic), optimizing models for edge deployment, and developing fine-grained payload detection and classification. We also plan to explore self-supervised and domain adaptation techniques to further improve generalization in unseen environments. Our findings provide a strong foundation for advancing UAV surveillance systems in both research and real-world applications.

## REFERENCES

- [1] W. Lv, Y. Zhao, Q. Chang, K. Huang, G. Wang, and Y. Liu, "Rt-detr v2: Improved baseline with bag-of-freebies for real-time detection transformer," 2024. [Online]. Available: <https://arxiv.org/abs/2407.17140>
- [2] D. I. Krasnov, S. N. Yarishev, V. A. Ryzhova, and T. S. Djamiykov, "Improved yolov8 network for small objects detection," in *2024 XXXIII International Scientific Conference Electronics (ET)*, 2024, pp. 1–4.
- [3] J. Liu, X. Fan, Z. Huang, G. Wu, R. Liu, W. Zhong, and Z. Luo, "Target-aware dual adversarial learning and a multi-scenario multi-modality benchmark to fuse infrared and visible for object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5802–5811.
- [4] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, and M.-H. Yang, "Restormer: Efficient transformer for high-resolution image restoration," in *CVPR*, 2022.
- [5] Y. Zhang, P. Sun, Y. Jiang, D. Yu, F. Weng, Z. Yuan, P. Luo, W. Liu, and X. Wang, "Bytetrack: Multi-object tracking by associating every detection box," 2022.
- [6] Y. Tian, Q. Ye, and D. Doermann, "Yolov12: Attention-centric real-time object detectors," *arXiv preprint arXiv:2502.12524*, 2025.
- [7] I. Robinson, P. Robicheaux, and M. Popov, "Rf-detr," <https://github.com/roboflow/rf-detr>, 2025, sOTA Real-Time Object Detection Model.
- [8] S. Huang, Z. Lu, X. Cun, Y. Yu, X. Zhou, and X. Shen, "Deim: Detr with improved matching for fast convergence," 2025. [Online]. Available: <https://arxiv.org/abs/2412.04234>
- [9] Y. Zhao, W. Lv, S. Xu, J. Wei, G. Wang, Q. Dang, Y. Liu, and J. Chen, "Detrs beat yolos on real-time object detection," 2023.