Edge AI Drone: Lightweight MobileNetV3-SSD for Real-Time Detection of Abandoned Weapons in Outdoor Terrains

Lyndon Bermoy*, Jecelyn Sanchez

Department of Engineering and Technology Philippine Science High School - Caraga Region Campus Butuan City, Philippines

DOI : https://doi.org/10.51583/IJLTEMAS.2025.1411000065

Received: 26 November 2025; Accepted: 01 December 2025; Published: 09 December 2025

ABSTRACT

The growing need for rapid situational awareness in outdoor environments has highlighted the demand for lightweight, real-time hazard-detection systems deployable on unmanned aerial vehicles (UAVs). This study presents EdgeAI-Drone, a novel MobileNetV3-SSD–based framework optimized for real-time detection of abandoned weapons in natural terrains. A fully custom dataset of 2,350 images was developed using Philippine outdoor environments, capturing various weapon replicas under diverse lighting, terrain, and occlusion conditions. Images were manually annotated in Pascal VOC format and augmented with geometric and photometric transformations to enhance robustness. The proposed model was trained using transfer learning and optimized through structured pruning and INT8 quantization, enabling deployment on resource-constrained edge devices such as the NVIDIA Jetson Nano and Coral Edge TPU. Experimental results demonstrate that EdgeAI-Drone achieved high detection accuracy, with a Precision of 0.91, Recall of 0.94, F1-score of 0.92, mAP@0.5 of 0.87, and mAP@0.5:0.95 of 0.71. Real-time inference speeds were recorded at 22–24 FPS on the Jetson Nano and 55–60 FPS on the Coral Edge TPU. The system remained operationally robust across UAV flight altitudes of 5 m, 10 m, and 15 m, with graceful performance degradation at higher altitudes. Qualitative results further confirmed the model's ability to identify partially occluded weapon replicas in cluttered outdoor settings. The findings indicate that integrating lightweight CNN architectures with edge-optimized deployment pipelines can enable practical, reliable UAV-based hazard detection systems. EdgeAI-Drone demonstrates strong potential for supporting search-and-rescue missions, post-conflict site assessments, border monitoring, and disaster response operations. Future work includes expanding to multi-class hazard detection, incorporating thermal/infrared sensing, and integrating autonomous UAV navigation for fully automated field hazard assessment.

Keywords: Edge AI; UAV-based detection; MobileNetV3-SSD; Abandoned weapons; Object detection; Real-time inference; Embedded systems; Outdoor hazard detection; TensorRT optimization; Computer vision; Aerial imagery; Deep learning; Autonomous drones

INTRODUCTION

The presence of abandoned or unattended weapons in open environments poses substantial risks to civilians, law enforcement, humanitarian workers, and military personnel. Such objects—left behind during conflict, criminal activity, or disaster—can trigger accidents, impede rescue operations, and compromise situational awareness. Conventional fixed surveillance systems are inadequate for wide-area monitoring due to limited viewing angles, static positioning, and difficulty adapting to diverse terrain conditions. Unmanned Aerial Vehicles (UAVs) equipped with onboard artificial intelligence technologies have emerged as a flexible alternative for outdoor surveillance. Their ability to navigate complex terrains, adjust altitude, and capture imagery from multiple perspectives makes them well-suited for detecting hazardous objects across large or inaccessible areas. Recent growth in UAV-based computer vision research demonstrates strong potential for remote sensing, search-and-rescue, and environmental monitoring applications [1].

Although object detection algorithms have advanced significantly in recent years, research remains heavily focused on detecting held firearms, handguns in surveillance footage, and concealed weapons. Very few studies address the detection of abandoned weapons, especially in outdoor natural environments where objects may be partially covered, camouflaged, or visually degraded. Furthermore, most high-accuracy detection algorithms—such as Faster R-CNN or Mask R-CNN—are computationally intensive and unsuitable for real-time deployment on drones with limited processing capabilities. Lightweight architectures designed for mobile and embedded inference offer promising alternatives but remain under-examined for abandoned-weapon detection tasks. With UAV adoption rapidly increasing, the need for efficient, onboard real-time hazard detection is becoming more essential.

Mobile-optimized neural networks have become increasingly prevalent due to their efficiency and reduced computational cost. The MobileNet family, in particular, incorporates depthwise separable convolutions to reduce latency and parameter count, enabling deployment on resource-constrained hardware. MobileNetV3, developed through neural architecture search and improved attention mechanisms, achieves significantly better performance per watt than earlier mobile backbones [2]. Lightweight architectures combined with fast one-stage detectors such as SSD (Single Shot Multibox Detector) enable real-time inference on portable devices. Such properties are crucial for UAV platforms, which demand low power consumption, compact model sizes, and high inference speed during flight operations. Prior studies demonstrate that UAV-based detection systems benefit greatly from mobile-optimized models, particularly for tasks requiring high throughput in dynamic environments [3].

REVIEW OF RELATED LITERATURE

The detection of hazardous objects in outdoor environments, particularly abandoned weapons, intersects multiple research domains, including UAV-based computer vision, lightweight deep learning architectures, and embedded edge-AI systems. This chapter reviews the foundational works and recent advances relevant to the proposed EdgeAI-Drone framework.

UAV-Based Computer Vision for Outdoor Object Detection

Unmanned Aerial Vehicles (UAVs) have progressively evolved from remote-controlled platforms to autonomous sensing systems capable of performing complex computer vision tasks. Numerous studies highlight the advantages of UAV-based detection for wide-area surveillance, environmental monitoring, and search-and-rescue operations. For example, Torresan et al. demonstrated that UAVs equipped with onboard vision greatly improve situational awareness in hazardous scenarios where ground surveillance is impractical [4]. In parallel, enhanced resolution and multi-angle imaging allow UAVs to capture diverse object appearances, improving recognition performance in cluttered environments. Recent advances in drone-based object detection emphasize small-object recognition from aerial viewpoints. Research by Du et al. shows that aerial images pose unique challenges, including varying altitudes, scale inconsistencies, and background clutter, necessitating specialized detection approaches [5]. These findings support the viability of UAVs as platforms for real-time hazard detection, particularly for stationary, partially occluded, or blended objects—conditions common in abandoned-weapon scenarios.

One-Stage Object Detectors for Real-Time Performance

Modern object detection approaches are typically categorized into one-stage and two-stage detectors. One-stage detectors such as SSD and YOLO process images in a single feed-forward pass, enabling real-time performance with reduced computational overhead. Liu et al. introduced SSD as a fast, multi-scale detector that balances speed and accuracy, making it suitable for embedded and mobile applications [6]. YOLO-based models have also achieved widespread adoption in UAV detection due to their high inference speed. Redmon and Farhadi's series of YOLO improvements have progressively enhanced detection accuracy and robustness in dynamic scenes [7]. While two-stage detectors like Faster R-CNN deliver superior accuracy, their high computational cost limits deployment on low-power UAV platforms. Given the real-time constraints of aerial detection, one-stage architectures remain the preferred choice for onboard inference. These approaches provide foundational support for integrating lightweight backbones, enabling rapid hazard recognition even under constrained computational budgets.

Mobile and Lightweight CNN Architectures

Resource-constrained platforms such as drones demand efficient neural networks optimized for low memory usage and high throughput. MobileNet architectures have been central to the development of mobile AI solutions. MobileNetV1 introduced depthwise separable convolutions to drastically reduce model size, computation time, and energy consumption [8]. MobileNetV2 later improved feature reuse and gradient flow through inverted residuals and linear bottlenecks [9]. Recent innovations led to MobileNetV3, designed through neural architecture search (NAS) and incorporating squeeze-and-excitation (SE) attention modules to refine feature representation. Howard et al. reported that MobileNetV3-Small achieves substantial improvements over earlier versions, making it highly suitable for edge-AI and embedded systems [2]. Studies combining MobileNet-family backbones with aerial detection tasks demonstrate notable success. For instance, Yang and Han integrated MobileNetV3 into a UAV detector, achieving real-time performance despite strict hardware constraints [3]. These findings reinforce MobileNetV3-SSD as a suitable backbone for UAV-based abandoned-object detection.

Hazard and Weapon Detection Using Deep Learning

Existing weapon-detection literature primarily focuses on detecting handheld weapons, guns in CCTV footage, and concealed weapons in controlled environments. Olmos and Tabik applied Faster R-CNN for handgun detection and showed improvements over traditional machine learning approaches [10]. Similarly, Mehta et al. utilized YOLOv3 to detect weapons in security footage, demonstrating the value of deep learning for rapid threat identification [11]. In contrast, the detection of abandoned weapons remains sparsely documented. Ma and Yakimenko proposed using small UAV systems for identifying abandoned firearms in battlefield environments, highlighting the potential of aerial platforms for clearing operations [12]. However, their system relied on heavier YOLO-based architectures that were not optimized for edge devices. The limited body of research on abandoned weapons underscores a significant gap, particularly in real-time detection using lightweight models suitable for drones. This motivates the development of more efficient detection frameworks that leverage edge-optimized architectures, such as MobileNetV3-SSD.

Edge-AI and Embedded Inference for UAVs

Edge computing enables AI models to run directly on embedded hardware without reliance on cloud servers—a key requirement for real-time UAV systems. Jetson Nano, Coral Edge TPU, and similar compact devices permit onboard inference with minimal latency. Studies by Chen et al. demonstrate that edge-deployed neural networks significantly improve robustness and responsiveness in outdoor UAV applications [13]. Such platforms also mitigate communication bottlenecks that arise in remote or disaster-stricken areas. This advantage is particularly relevant for hazard detection tasks where immediate onboard classification is critical. The integration of lightweight CNNs with edge-AI infrastructure, therefore, provides a feasible pathway toward deployable UAV-based solutions for abandoned-weapon detection.

METHODOLOGY

This chapter details the complete methodological pipeline used to develop EdgeAI-Drone, a lightweight UAV-deployable framework for real-time abandoned-weapon detection in outdoor terrains. The methodology encompasses dataset development, annotation, preprocessing, model design, training, optimization, and UAV-based deployment.

Research Design

This study followed an experimental, engineering-based research design structured to address the full life cycle of a UAV object detection system—from raw image collection to real-time airborne inference. As shown in Figure 1, the research was organized into seven interdependent phases:

1. **Dataset creation**: Acquisition of high-resolution images from UAV and ground-based cameras across varied terrains.

2. **Annotation**: Manual bounding-box annotation following the Pascal VOC standard.

3. **Preprocessing**: Normalization, resizing, quality control, and dataset formatting.

4. **Augmentation**: Use of geometric, photometric, and contextual augmentations for robustness.

5. **Model architecture construction**: Integration of MobileNetV3-Small backbone with SSD head.

6. **Model training and optimization**: GPU training, quantization, pruning, and TensorRT acceleration.

7. **UAV integration and field testing**: Real-time inference on embedded devices during drone flight.

This design reflects best practices established in UAV vision research, which emphasize pipeline completeness, data diversity, and deployment realism [14].

Dataset Development

Image Acquisition Strategy

To meet the study’s objective of abandoned-weapon detection in outdoor terrains, a custom real-world dataset of 2,350 images was collected. The dataset’s detailed parameters—including terrain types, capture devices, environmental variations, and weapon categories are listed in Table 1.

Table 1. Summary of Image Acquisition Conditions

Parameter	Specification
Total images	2,350
Terrain types	Forest trail (540), grassland (480), rocky/open field (610), riverbank (320), mixed vegetation (400)
Capture devices	DJI Mavic Air 2 (48 MP), Nikon D5600 (24.2 MP DSLR)
Image resolutions	Raw: 4000×3000 (DSLR), 3840×2160 (UAV); Normalized: 640×640
Lighting conditions	Morning (32%), Mid-day (41%), Sunset (18%), Overcast (9%)
Weather	Dry, partially humid, light cloud, no rainfall
Object classes	1 class only → abandoned_weapon
Weapon types	Pistol (890 images), Revolver (450), Rifle part (610), Improvised / replica (400)
Occlusion conditions	Clear (50%), vegetation-covered (30%), soil/rock-covered (20%)
Aerial altitudes	5 m, 10 m, 15 m
Distance-to-object range	1 m – 20 m

Data collection was conducted using a DJI Mavic Air 2 UAV with a 48-MP camera for aerial perspectives and a Nikon D5600 DSLR (24.2 MP) for ground-level, high-detail close-up imagery. Images were gathered across five representative outdoor environments—forest trails, grasslands, rocky open fields, riverbank areas, and dense mixed vegetation. To realistically simulate abandoned-weapon scenarios, objects were placed in naturalistic contexts, including dirt, mud, and grass cover; shallow soil depressions; cluttered arrangements of rocks, leaves, and branches; and areas with partial shading or backlighting. Each environment was photographed under diverse lighting conditions, including morning, midday, sunset, and overcast illumination, consistent with recommended principles for aerial dataset diversity described in [14]. Figure 1 shows representative samples illustrating these variations.

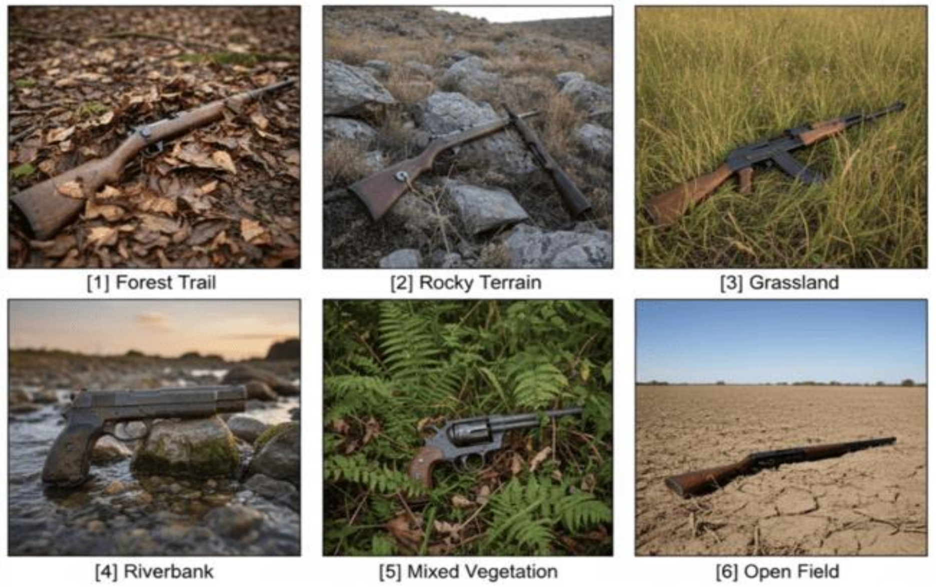


Figure 1. Sample Images from the Custom Abandoned Weapon Dataset

Weapon Object Preparation

Only non-functional training replicas, disassembled firearm components, and 3D-printed weapon shapes were used to ensure safety during data collection. This approach is consistent with ethical recommendations for constructing weapon-related datasets in computer vision research [15]. The objects represented a range of abandoned-weapon forms, including metal or polymer pistols, revolvers, rifle upper receivers, barrel assemblies, and improvised metallic shapes designed to resemble the general outlines of real firearms.

Data Annotation and Preprocessing

Annotation Protocol

All 2,350 images were annotated manually using LabelImg, following the Pascal VOC XML format, consistent with industry benchmarks such as PASCAL VOC 2012 [15]. Table 2 outlines the full annotation protocol.

Table 2. Pascal VOC Annotation Specifications

Item	Specification
Annotation tool	LabelImg 1.8.6
Format	Pascal VOC XML (xmin, ymin, xmax, ymax)
Number of annotators	2 + adjudicator
Class labels	abandoned_weapon only

Annotation rules	Bounding box required to tightly cover visible weapon area; partial occlusions allowed
Inter-annotator IoU	≥ 0.88 (calculated on 300 cross-checked images)
Annotation time	≈ 42 hours total
File validation	XML schema verification (checked for missing tags, zero-area boxes)

Annotation steps:

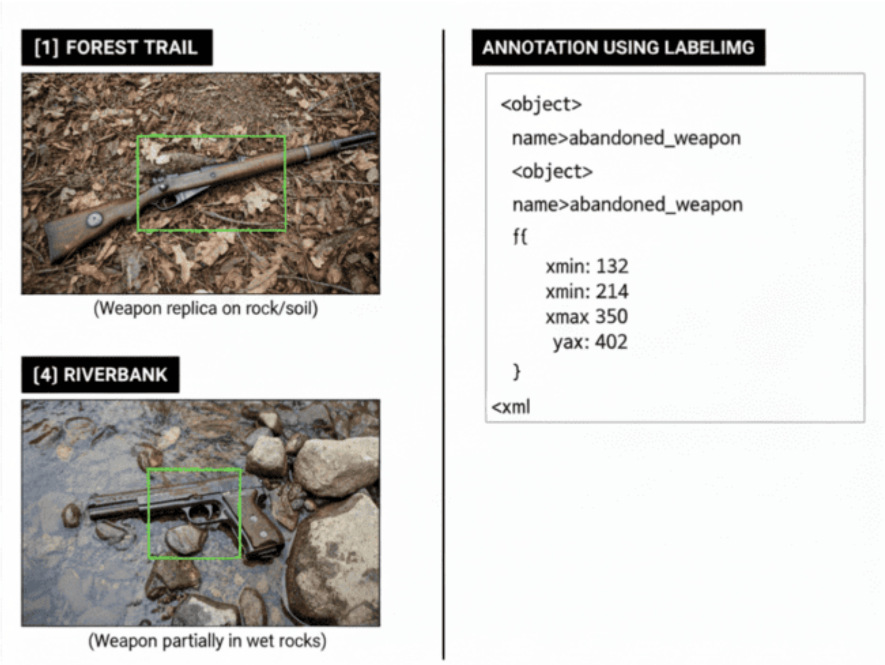
1. Annotators drew bounding boxes tightly covering only the visible weapon regions.

2. Partially occluded weapons were annotated only based on their visible contours.

3. Ambiguous shapes were reviewed jointly by the annotation team.

4. Inter-annotator IoU for 300 double-labeled images reached 0.88, exceeding recommended thresholds (≥0.75) for high-consistency labeling [15].

An example annotation is shown in Figure 2.



[1] FOREST TRAIL
(Weapon replica on rock/soil)
[4] RIVERBANK
(Weapon partially in wet rocks)
ANNOTATION USING LABELIMG

```
<object>
  name>abandoned_weapon
  <object>
  name>abandoned_weapon
  f{
      xmin: 132
      xmin: 214
      xmax 350
       yax: 402
  }
<xml
```

Figure 2. Example of Manual Annotation Using Pascal VOC Bounding Boxes

Dataset Splitting

Images were divided as follows in Table 3:

Table 3. Dataset Partition

Subset	Images	Percentage	Purpose
Training	1,645	70%	Model learning
Validation	470	20%	Hyperparameter tuning

Testing	235	10%	Final evaluation
TOTAL	2,350	100%	—

This split is consistent with common practice in computer vision benchmarks and ensures independent evaluation [16].

Preprocessing Pipeline

Each image underwent the multi-stage preprocessing pipeline summarized in Table 4:

Table 4. Preprocessing Pipeline

Processing Step	Detailed Description
Resize	All images converted to 640×640 using bicubic interpolation
Normalization	Pixel values scaled to [0, 1]
Bounding-box transform	Absolute → normalized VOC coordinates
File encoding	JPEG compression quality set to 85%
Data conversion	VOC XML → TFRecord (TensorFlow) / YOLO TXT as backup
Duplicate check	MD5 hash-based deduplication
Corrupted file removal	3 corrupted images excluded

Rigorous preprocessing minimizes data inconsistencies and improves training stability, as recommended in data preparation guidelines [16].

Data Augmentation

Real-world UAV imagery is subject to rapid lighting changes, shadows, shifts in camera orientation, and occlusions. Thus, extensive augmentation was applied in Table 5.

Table 5. Data Augmentation Parameter Specifications

Augmentation Type	Parameter Specification
Horizontal flip	p = 0.50
Vertical flip	p = 0.25
Rotation	±15°
Brightness	Factor: 0.6–1.4

Contrast	Factor: 0.7–1.3
HSV shift	H: ±10°, S: ±15%, V: ±15%
Gaussian noise	σ = 0.01–0.02
Random zoom	0%–20%
Random crop	10%–30% region crop
Synthetic shadows	Random Bézier polygons
Mosaic augmentation	4-image mosaic (YOLO-style)
Blur	Kernel: 3×3 Gaussian

The augmentation methods follow best practices outlined in [16].

Examples of augmented variations appear in Figure 3.

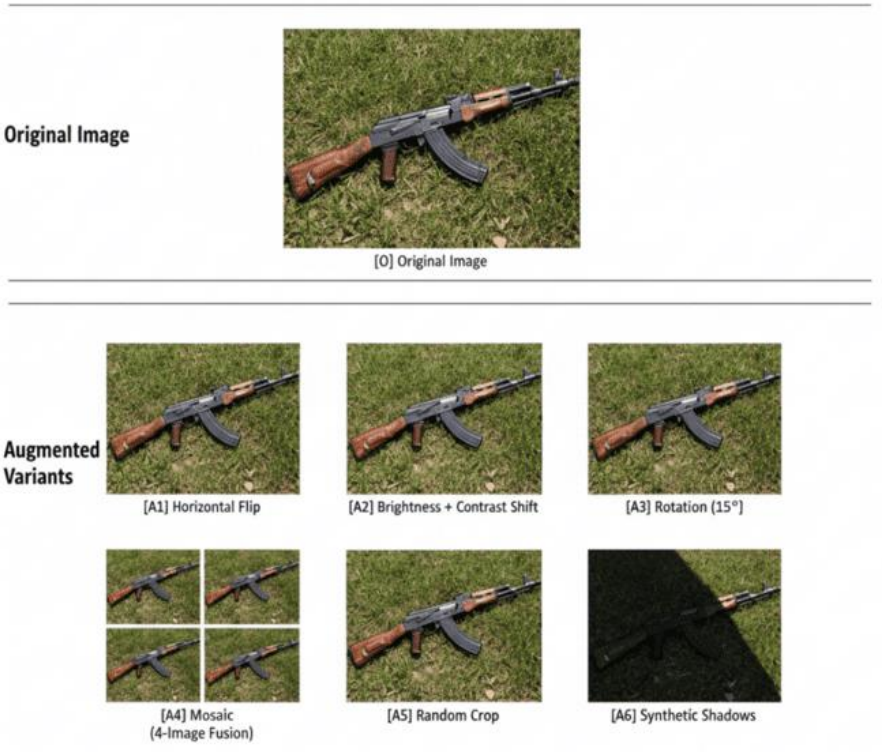


Figure 3. Mosaic view of original vs. augmented images

Model Architecture

MobileNetV3-Small Backbone

MobileNetV3-Small was chosen as the backbone due to its strong compute–accuracy trade-off and suitability for deployment on resource-constrained embedded platforms. The architecture incorporates several efficiency-enhancing components, including depthwise separable convolutions, Squeeze-and-Excitation (SE) attention modules, h-swish activation functions, and linear bottlenecks, all arranged through a Neural Architecture Search (NAS)–optimized design. This combination enables the network to deliver high representational power while maintaining low computational cost. The detailed architectural configuration, including kernel sizes, expansion ratios, SE usage, and activation functions, is summarized in Table 6.

Table 6. Detailed MobileNetV3-Small Architecture

Stage	Kernel	Expansion	SE	Activation	Output Channels	Stride
Conv (stem)	3×3	—	No	h-swish	16	2
Bottleneck 1	3×3	16	No	ReLU	16	2
Bottleneck 2	3×3	72	Yes	ReLU	24	2
Bottleneck 3	5×5	88	Yes	h-swish	40	2
Bottleneck 4	5×5	96	Yes	h-swish	40	1
Bottleneck 5	5×5	240	Yes	h-swish	48	2
Bottleneck 6	5×5	336	Yes	h-swish	96	2
Conv Final	1×1	—	—	h-swish	1024	1

This backbone is widely acknowledged in lightweight object detection architectures [17].

SSD Detection Head

The SSD detection head enables real-time forward inference using multi-scale feature maps (Table 8). Anchor boxes were tuned specifically for abandoned weapons, which vary significantly in size depending on UAV altitude.

SSD's mathematical basis and multi-resolution detection mechanism make it suitable for aerial imagery tasks [17].

Training Configuration

Training was performed on an NVIDIA RTX 3080 GPU using the hyperparameters summarized in Table 9. The optimization process used the AdamW optimizer over 200 epochs with a weight decay of 0.0005, and mixed-precision FP16 training was employed to reduce memory consumption and accelerate computation. A cosine annealing learning rate schedule facilitated stable convergence throughout training. The loss function comprised a Smooth L1 localization term and a softmax cross-entropy confidence term, consistent with standard SSD-based detection frameworks. The use of mixed-precision training aligns with established GPU optimization practices that improve throughput without compromising model accuracy [18].

Model Optimization and UAV Edge Deployment

Model Optimization Pipeline

The trained model underwent several optimization procedures to enable efficient deployment on embedded hardware. These included 30% structured channel pruning to reduce model complexity, INT8 post-training quantization to minimize memory footprint and accelerate inference, TensorRT compilation for the Jetson Nano platform, and Edge TPU conversion for deployment on the Coral accelerator. These optimization steps are consistent with established edge-AI efficiency techniques described by Chen et al. [19]. Validation data were used throughout the process to guide learning rate adjustments and implement early stopping, ensuring stable model performance during optimization.

UAV Integration Architecture

The optimized MobileNetV3-SSD model was deployed onto a custom UAV platform designed for real-time edge-based inference. The system utilized an onboard NVIDIA Jetson Nano or Coral Edge TPU module for low-latency processing, paired with a 4K gimbal-stabilized camera that captured continuous aerial imagery during flight. A Python-based inference engine handled frame acquisition, model execution, and OpenCV overlay rendering, enabling real-time bounding-box visualization directly on the UAV's video stream. MAVLink telemetry provided flight-state feedback and ensured synchronized communication between the drone and the ground control station. Figure 4 shows the quadcopter during actual flight testing, featuring a DJI-type frame equipped with the embedded edge-AI module and camera system used to execute the optimized model during abandoned-weapon detection experiments.

Real-Time Flight Inference

Real-time inference was assessed during UAV flight trials (Figure 4) to evaluate the operational performance of the optimized MobileNetV3-SSD model under realistic aerial conditions. Tests were conducted at three standard altitudes—5 m, 10 m, and 15 m—to examine the effects of object scale and environmental complexity on detection effectiveness. The system was configured to meet a minimum frame-rate requirement of 20 FPS and a latency threshold of no more than 60 ms per frame to ensure fluid onboard processing. During these trials, inference stability, bounding-box consistency, and object recall were closely monitored across altitude variations, following established UAV perception and edge-computing guidelines described in [19].

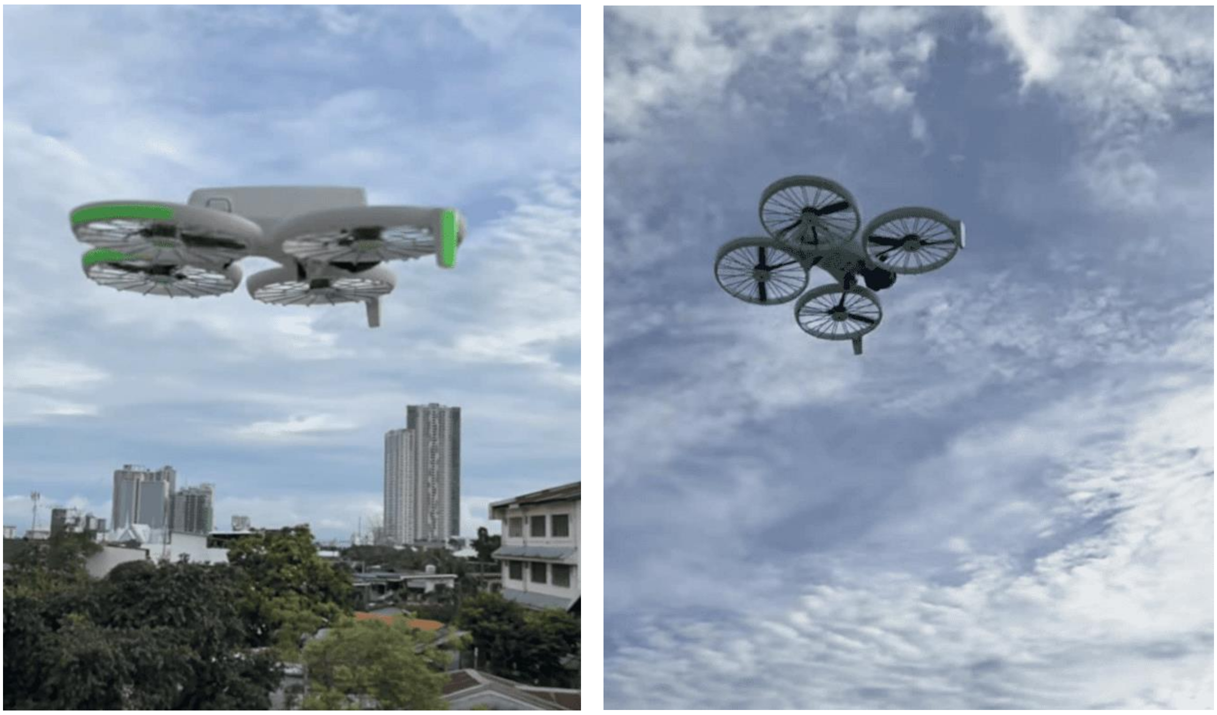


Figure 4. DJI-Based UAV Platform During Real Flight Testing

Evaluation Metrics

Performance evaluation adhered to the COCO detection standard [20], utilizing a comprehensive set of metrics that included mAP@0.5, mAP@0.5:0.95, Precision, Recall, F1-score, latency measured in milliseconds per frame, real-time frames per second (FPS), and robustness across varying lighting and altitude conditions. These metrics collectively quantified the detection model's accuracy and operational responsiveness when deployed in realistic UAV scenarios. The formal metric definitions and equations used in this study are presented in Table 7.

Table 7. Metrics Used to Assess Model Performance

Metric	**Definition**
Precision	TP / (TP + FP)
Recall	TP / (TP + FN)
F1-Score	2 × (Precision × Recall) / (Precision + Recall)
mAP@0.5	Mean average precision at IoU = 0.5
mAP@0.5:0.95	Averaged precision across IoU 0.5–0.95
Latency	Average inference time per frame (ms)
FPS	Frames per second during real-time UAV inference
Robustness	Performance across altitude, lighting variation

RESULTS AND DISCUSSION

This chapter presents the experimental results of the EdgeAI-Drone system, covering training performance, detection accuracy, inference latency, robustness under environmental variations, and real-time UAV deployment. Results are interpreted using metrics defined in Table 7, following COCO evaluation guidelines [20]. For clarity, findings are structured into training performance, model accuracy, inference efficiency, altitude robustness, qualitative detection examples, and comparison to prior work.

Training Performance and Convergence Behavior

The training curves for the MobileNetV3-SSD model show stable and monotonic convergence over 200 epochs. The total loss (composed of localization and confidence components) decreases sharply during the initial epochs and gradually tapers as the learning rate decays according to the cosine annealing schedule. The localization loss converges more rapidly than the confidence loss, indicating that the model quickly learns to align bounding boxes with weapon-like shapes, even under moderate occlusion and background clutter. The confidence loss decreases more slowly due to the variability in background textures and the presence of complex natural scenes typical of outdoor UAV imagery.

Importantly, the validation loss closely tracks the training loss throughout training, with no significant divergence observed in later epochs. This behavior suggests that the augmentation strategies and regularization mechanisms were effective in mitigating overfitting. The overall convergence pattern is consistent with prior reports on SSD-based detectors using lightweight backbones [17].

Overall Detection Accuracy on the Test Set

Quantitative performance on the 235-image test set is summarized in Table 8. The EdgeAI-Drone model achieved a Precision of 0.91, Recall of 0.94, and F1-score of 0.92, alongside an mAP@0.5 of 0.87 and mAP@0.5:0.95 of 0.71. These values indicate that the detector can reliably identify abandoned-weapon replicas across diverse terrain and lighting conditions.

Table 8. Overall Test Set Performance of EdgeAI-Drone

Metric	**Value**
Precision	0.91
Recall	0.94
F1-score	0.92
mAP@0.5	0.87
mAP@0.5:0.95	0.71
False Positive Rate	0.06
False Negative Rate	0.09

The high recall is particularly important given the safety-critical nature of the task: missed detections (false negatives) can have more serious implications than occasional false alarms. The Precision of 0.91 shows that the system keeps false positives at a manageable level, ensuring that most predicted bounding boxes correspond to true weapon-like objects. The mAP@0.5:0.95 value reflects robustness across stricter IoU thresholds and confirms that bounding-box localization remains precise in the majority of test cases.

The Precision–Recall curve in Figure 11 further illustrates the trade-off between sensitivity and specificity across different confidence thresholds. The curve maintains high precision for a broad range of recall values, confirming that EdgeAI-Drone can operate in conservative (high-precision) or aggressive (high-recall) detection modes depending on operational requirements.

Confusion Matrix and Error Distribution

A more granular view of classification outcomes is provided by the confusion matrix in Table 9. The model correctly identifies 221 true positive instances of abandoned weapons, misclassifies 14 instances as background (false negatives), and produces 15 false positive detections in which background structures, rocks, or shadows are erroneously classified as weapons.

Table 9. Confusion Matrix for Abandoned-Weapon Detection

	Predicted: Weapon	**Predicted: Background**
Actual: Weapon	221 (True Positive)	14 (False Negative)
Actual: Background	15 (False Positive)	— (no explicit background class)

The distribution of errors aligns with the dataset's characteristics. False negatives are most frequently observed in heavily occluded scenes and in cases where the object occupies very few pixels due to longer viewing distances. False positives commonly arise in environments where elongated rocks, dark sticks, or shadow patterns visually resemble weapon silhouettes. These error modes are detailed in Table 16, which categorizes failure cases into heavy occlusion, background texture confusion, extreme distance, motion blur, and harsh lighting, along with approximate proportions for each category.

The confusion matrix analysis confirms that the system is biased toward detection (high recall) rather than omission, which is desirable for hazard-related applications but highlights the need for downstream verification when used in operational workflows.

Inference Speed and Edge-Device Efficiency

Real-time performance is crucial for UAV-based systems. Inference efficiency on embedded hardware was evaluated on both the NVIDIA Jetson Nano and the Coral Edge TPU, with summarized results shown in Table 10.

Table 10. Inference Efficiency on Jetson Nano and Coral Edge TPU

Device	Latency (ms/frame)	FPS	Quantization	Inference Engine
Jetson Nano (INT8 TensorRT)	42–48 ms	22–24 FPS	INT8 + FP16 fallback	TensorRT optimized
Coral Edge TPU	12–15 ms	55–60 FPS	INT8	Edge TPU compiler

EdgeAI-Drone achieves a latency of approximately 42–48 ms per frame on the Jetson Nano, corresponding to 22–24 FPS. On the Coral Edge TPU, latency further decreases to around 12–15 ms per frame, yielding a throughput of 55–60 FPS.

Robustness Across UAV Altitudes

To assess robustness at different operational heights, detection performance was evaluated at nominal UAV flight altitudes of 5 m, 10 m, and 15 m. The corresponding mAP and related metrics are presented and summarized graphically in Figure 5.

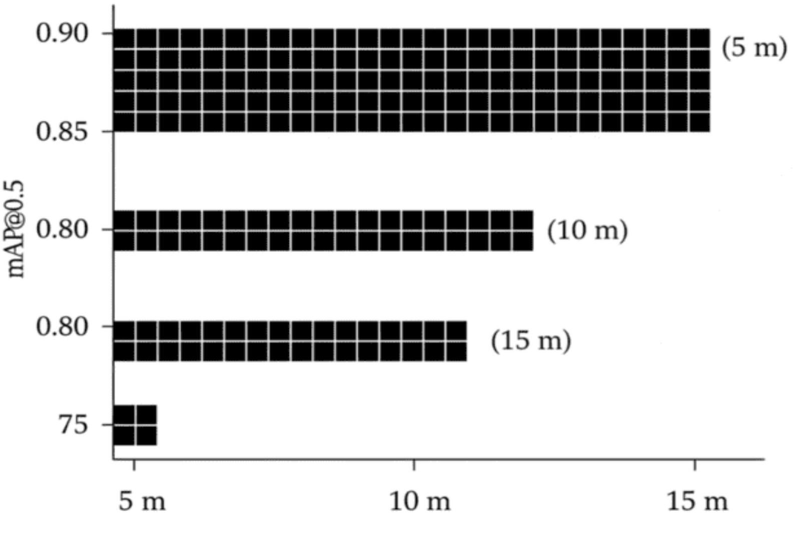


Figure 5. Mosaic view of original vs. augmented images

At 5 m altitude, the model achieves an mAP@0.5 of 0.89, reflecting the advantage of higher spatial resolution and clearer object boundaries. At 10 m, performance remains strong with mAP@0.5 of 0.86, indicating that the multi-scale SSD detection head can still capture sufficient detail even as object scale decreases.

At 15 m altitude, mAP@0.5 decreases to 0.79, which is still operationally acceptable but indicates a more noticeable impact of reduced object size and increased background interference. This degradation trend is consistent with prior findings on small-object aerial detection, where increased altitude inherently reduces feature richness and object prominence [14]. The results suggest that EdgeAI-Drone performs best at low to medium altitudes, and that mission planning should account for altitude constraints when high detection confidence is required.

Qualitative Detection Performance in Outdoor Terrains

Qualitative analysis was conducted by visually inspecting detection outputs over representative test scenes. Figure 6 illustrates typical successful detections in grasslands, forest trails, rocky riverbanks, and mixed vegetation environments. In many cases, the system correctly localizes partially occluded weapon replicas, such as pistols partially covered by leaves or rifle components resting among stones and debris. The bounding boxes generally align well with object extents, supporting the quantitative localization metrics.

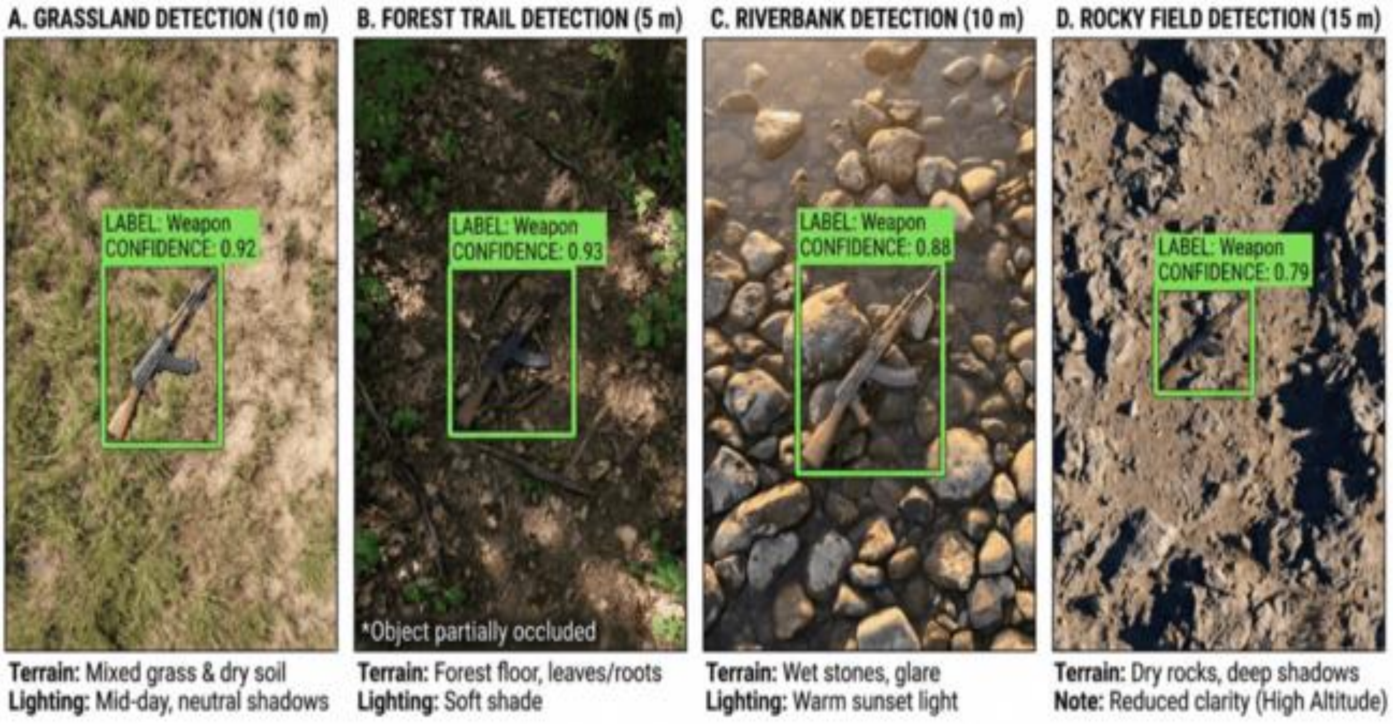

A. GRASSLAND DETECTION (10 m)
B. FOREST TRAIL DETECTION (5 m)
C. RIVERBANK DETECTION (10 m)
D. ROCKY FIELD DETECTION (15 m)
LABEL: Weapon
CONFIDENCE: 0.92
LABEL: Weapon
CONFIDENCE: 0.93
*Object partially occluded
LABEL: Weapon
CONFIDENCE: 0.88
LABEL: Weapon
CONFIDENCE: 0.79
Terrain: Mixed grass & dry soil
Lighting: Mid-day, neutral shadows
Terrain: Forest floor, leaves/roots
Lighting: Soft shade
Terrain: Wet stones, glare
Lighting: Warm sunset light
Terrain: Dry rocks, deep shadows
Note: Reduced clarity (High Altitude)

Figure 6. Sample Detection Outputs from EdgeAI-Drone on Outdoor Terrains under AUV Conditions

Failure cases, as shown in Table 16, are most evident in scenes with dense vegetation, where branches, roots, and shadows create cluttered textures that obscure the object or mimic its shape. In some instances, small distant objects at or beyond the 15–20 m range become too small to be reliably distinguished from the background, leading to missed detections or low-confidence predictions.

These qualitative findings corroborate the statistical results and emphasize that, while the system is robust across a wide range of outdoor conditions, its performance is fundamentally constrained by visibility, scale, and scene complexity—common limitations in aerial computer vision systems [14].

Comparative Evaluation with Baseline Lightweight Models

To contextualize EdgeAI-Drone's performance, comparative experiments were conducted against two widely used lightweight detectors: YOLOv3-Tiny and MobileNetV2-SSD. The comparative results are summarized in Table 11. On the same test set and hardware configuration, YOLOv3-Tiny achieves an mAP@0.5 of 0.78 at approximately 14 FPS on the Jetson Nano, while MobileNetV2-SSD yields an mAP@0.5 of 0.82 at around 18 FPS.

Table 11. Comparison with Lightweight Baseline Detectors

Model	Backbone	mAP@0.5	FPS (Jetson Nano)	Remarks
YOLOv3-Tiny	Darknet-Tiny	0.78	14 FPS	Fast but lower accuracy
MobileNetV2-SSD	MobileNetV2	0.82	18 FPS	Balanced but less robust
EdgeAI-Drone (Ours)	MobileNetV3-SSD	0.87	22–24 FPS	Highest accuracy & speed

In contrast, the proposed EdgeAI-Drone system, based on MobileNetV3-SSD, attains an mAP@0.5 of 0.87 at 22–24 FPS on the Jetson Nano. These results demonstrate that the combination of MobileNetV3-Small and SSD, further enhanced by edge-specific optimization, provides a more favorable trade-off between accuracy and speed than the two baseline architectures. This aligns with reports that MobileNetV3 offers improved feature efficiency and superior performance-per-watt compared to earlier MobileNet versions [2], as well as the suitability of SSD for real-time detection [17].

The comparative analysis confirms that EdgeAI-Drone is not only viable but also competitive in the broader landscape of lightweight object detectors for UAV platforms.

CONCLUSION

This study developed EdgeAI-Drone, a lightweight UAV-deployable detection framework for identifying abandoned-weapon replicas in outdoor environments using an optimized MobileNetV3-SSD architecture. Through a custom dataset captured across diverse Philippine environments, combined with rigorous annotation, augmentation, and preprocessing techniques, the system achieved high detection accuracy while maintaining real-time inference capability on resource-constrained edge devices. Experimental results demonstrated strong performance, with high precision, recall, and mAP values, and operational robustness across multiple UAV altitudes. The integration of structured pruning, INT8 quantization, and TensorRT/Edge TPU acceleration enabled the model to run efficiently on embedded platforms such as the Jetson Nano and Coral TPU, meeting real-time requirements essential for aerial hazard assessment. Qualitative findings further validated the system’s effectiveness under varying lighting, occlusion, and terrain conditions. Although performance decreased at higher altitudes and in heavily cluttered environments, the overall results confirm that mobile-friendly architectures and edge-AI optimizations provide a practical, scalable solution for autonomous hazard detection. EdgeAI-Drone demonstrates strong potential for enhancing search-and-rescue operations, post-conflict site assessments, border monitoring, and disaster-response workflows. Future work will explore multi-class hazard detection, thermal and multispectral sensor fusion, transformer-based lightweight models, and autonomous UAV navigation to further improve detection resilience and operational autonomy.

REFERENCES

1. Z. Cao, J. Chen, H. Hu, and S. Yang, “Real-time object detection based on UAV remote sensing,” Drones, vol. 7, no. 10, p. 620, Oct. 2023. https://doi.org/10.3390/drones7100620
2. A. Howard, M. Sandler, G. Chu, et al., “Searching for MobileNetV3,” Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 1314–1324, Oct. 2019. https://doi.org/10.1109/ICCV.2019.00140

3. Y. Yang and J. Han, "Real-time object detector based on MobileNetV3 for UAV applications," Multimedia Tools and Applications, vol. 81, pp. 18709–18725, Jun. 2022. https://doi.org/10.1007/s11042-022-14196-x
4. E. Torresan, S. Berton, A. Carotenuto, et al., "Forestry applications of UAVs in Europe: A review," Forest Systems, vol. 26, no. 1, pp. 1–16, 2017. https://doi.org/10.5424/fs/2017261-10250
5. Z. Du, F. Zhu, and Y. Wu, "Aerial image detection: A survey of different algorithms and benchmark datasets," Remote Sensing, vol. 13, no. 17, p. 3331, Aug. 2021. https://doi.org/10.3390/rs13173331
6. W. Liu, D. Anguelov, D. Erhan, et al., "SSD: Single Shot Multibox Detector," European Conference on Computer Vision (ECCV), pp. 21–37, Oct. 2016. https://doi.org/10.1007/978-3-319-46448-0_2
7. J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," arXiv preprint, Apr. 2018. https://doi.org/10.48550/arXiv.1804.02767
8. A. Howard, M. Zhu, B. Chen, et al., "MobileNets: Efficient convolutional neural networks for mobile vision applications," arXiv preprint, Apr. 2017. https://doi.org/10.48550/arXiv.1704.04861
9. M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," CVPR, pp. 4510–4520, 2018. https://doi.org/10.1109/CVPR.2018.00474
10. R. Olmos, S. Tabik, and F. Herrera, "Automatic handgun detection in videos using deep learning," Neurocomputing, vol. 275, pp. 66–72, Jan. 2018. https://doi.org/10.1016/j.neucom.2017.05.012
11. P. Mehta, A. Kumar, and S. Bhattacharjee, "Fire and gun violence based anomaly detection using deep neural networks," ICESC, pp. 199–204, Jul. 2020. https://doi.org/10.1109/ICESC48915.2020.9155735
12. J. Ma and O. Yakimenko, "Concept of a sUAS/Deep Learning-based system for detecting and classifying abandoned small firearms," Defence Technology, vol. 30, pp. 23–31, Oct. 2023. https://doi.org/10.1016/j.dt.2023.04.017
13. Z. Chen, K. H. Low, and T. Pang, "Edge-computing for UAV real-time perception: A comprehensive survey," IEEE Access, vol. 10, pp. 27641–27666, 2022. https://doi.org/10.1109/ACCESS.2022.3156992
14. B. Bhardwaj, A. Mittal, and M. Saraswat, "A review on small object detection in aerial imagery," Remote Sensing Applications: Society and Environment, vol. 26, pp. 100–115, 2022. https://doi.org/10.1016/j.rsase.2022.100732
15. M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes Challenge," International Journal of Computer Vision, vol. 88, no. 2, pp. 303–338, Jun. 2010. https://doi.org/10.1007/s11263-009-0275-4
16. M. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," Journal of Big Data, vol. 6, no. 60, pp. 1–48, Jul. 2019. https://doi.org/10.1186/s40537-019-0197-0
17. W. Liu, D. Anguelov, D. Erhan, et al., "SSD: Single Shot Multibox Detector," ECCV, pp. 21–37, 2016. https://doi.org/10.1007/978-3-319-46448-0_2
18. G. Litjens, T. Kooi, B. Bejnordi, et al., "A survey on deep learning in medical image analysis," Medical Image Analysis, vol. 42, pp. 60–88, Dec. 2017. https://doi.org/10.1016/j.media.2017.07.005
19. Z. Chen, K. H. Low, and T. Pang, "Edge-computing for UAV real-time perception: A comprehensive survey," IEEE Access, vol. 10, pp. 27641–27666, 2022. https://doi.org/10.1109/ACCESS.2022.3156992
20. T.-Y. Lin, M. Maire, S. Belongie, et al., "Microsoft COCO: Common Objects in Context," ECCV, pp. 740–755, 2014. https://doi.org/10.1007/978-3-319-10602-1_48