Edge AI and TinyML for Vision Applications in Autonomous Drones: A Systematic Critical Review

Saeed Althabahi | Sultan Alafeefi

Abstract: Unmanned Aerial Vehicles (UAVs) are increasingly using on-board vision intelligence for surveillance, precision agriculture, search and rescue, and infrastructure inspection. 45 papers were found that were peer-reviewed (2020-2026), 10 papers from late 2025 and early 2026. We compare model compression techniques (quantization-aware training, pruning, knowledge distillation), hardware accelerators (NVIDIA Jetson, Google Coral, Hailo-8, Intel Loihi 2, ARM Cortex-M55), lightweight architectures, and application-domain benchmarks. The structured scoring matrix is used to provide a quantitative comparison across 12 of the state-of-the-art methods, based on factors such as accuracy, latency, power, model size, and generality of hardware. Most importantly, this review demonstrates that non-uniform calibration data sets confound reported Accuracy-Efficiency Index (AEI) values, and that few if any thermal-throttling tests have been conducted on actual UAV hardware. Some of the major open challenges are explored, including a lack of datasets, the adversarial fragility of quantized models, hardware fragmentation, and regulatory constraints, all of which are explored with relevant methodological recommendations. The review ends with a forward-looking research agenda focused on event-camera-based neuromorphic sensing, foundation-model compression, federated on-drone continual learning, and a standardised UAV-edge benchmarking.

Keywords: Edge AI, TinyML, UAV, drone vision, quantization, pruning, knowledge distillation, neural architecture search, VisDrone, Jetson.

I. INTRODUCTION

Remotely piloted UAVs are becoming autonomous agents with the ability to understand the scene in real-time. For manoeuvres that demand low latency, GPS-denied situations, and bandwidth-constrained environments, it is crucial to perform deep-learning vision pipelines with no cloud uplinks [1], [2]. However, UAV payloads have simultaneous weight, power, and thermal dissipation constraints which prevent the use of traditional GPU hardware (<200 g for edge compute, <10 W sustained inference, <thermal dissipation). This has led to the establishment of a unique research area, where TinyML (ultra-compact model deployment to devices in the kilobyte size range) meets Edge AI (inference at the network edge).

This review has been applied to the five areas of drone-based computer vision: (i) persistent surveillance and perimeter security; (ii) precision agriculture, crop health monitoring and weed classification; (iii) search and rescue (SAR) over remote terrain; (iv) civil infrastructure inspection of bridges, pipelines and power lines; and (v) disaster-response operations, such as flood-survivor detection and wildfire mapping. The common theme in these domains is that inference must take place at the sensor in real time, with no guarantee of good network connectivity.

Some prior works on TinyML are concerned with the issue at a general level [4], and some prior works on UAV detection are concerned with the issue at an independent level [5]. There has not been a systematic critical review that has comprehensively studied the Edge AI compression techniques, the UAV-specific hardware deployments, and the multi-domain application benchmarks with a thorough methodological examination of the limitations. This paper will fill that gap.

The rest of this paper follows this outline: In Section II, the literature is critically reviewed by categorizing it into techniques, and a summary table of reviewed papers is included. In Section III, the methodology used to evaluate the data and the AEI metric is introduced. In Section IV, the quantitative results are presented and analyzed critically. The challenges are discussed, and research directions are suggested in Section V. Section VI concludes.

II. REVIEW OF RELATED WORK

A. Model compression: Quantization:

Quantization eliminates precision (FP32) and replaces it with higher resolution integers of lower precision. Post training quantization (PTQ): This quantization is performed after training, but results in the loss of accuracy in small models. The quantization-aware training (QAT) adds simulated quantization during the training process, thus recovering from the accuracy at a higher training cost. Liberis et al. [6] proved INT8 QAT for MobileNetV3 deployed on ARM Cortex-M55, which resulted in a 3.2 times reduction in the latency with 0.8% drop in accuracy for aerial imagery. Lin et al. [7] proposed MCUNetV2 that co-optimizes NAS and quantization within 256 KB SRAM. Critically, both studies examined different sets of calibration images (ImageNet vs. custom aerial), which results in a significant variation, of 15-20%, between the two, and makes it hard to compare and contrast directly between AEI methods; neither study reported results under thermal throttling.

To reduce the energy-delay product of TinyML UAV detectors, Rusci et al. [8] suggested INT4 quantization for the bandwidth-limited layers, resulting in 1.7× energy-delay product (EDP) reduction on the STM32H7. In 2025, Banbury et al. [9] added a new UAV aerial-object sub-task to the MLPerf Tiny v1.1, and it was found in the UAV aerial-object sub-task that Hailo-8 has 3.1× higher energy efficiency than Jetson Nano on the VisDrone task.

B. Model Compression: Pruning

Structured pruning entails cutting off whole filters, which directly cuts down FLOPs. To tackle this, Han et al. [10] took a different approach, using iterative magnitude pruning to trim 62% of the parameters of YOLOv5s while maintaining 94.3% mAP on VisDrone2021. Taylor-expansion filter-importance scoring was proposed by Molchanov et al. [11]. Both were evaluated on desktop GPUs; however, not drone-class edge hardware, and thus, porting to the Jetson Nano only improved the latency by 1.4× for Chen et al.'s replication [12] (emphasis on overhead associated with memory-access operations), highlighting the desktop to edge generalization gap.

Unstructured pruning gives a compression of >90%, but must be performed on sparse-matrix hardware, which is not found on most UAV platforms. While proving hardware-contingential benefits, Liu et al. [13] showed that the drone SAR detector can be pruned to 88% sparsity at 1.2% mAP loss.

C. Knowledge Distillation

Knowledge distillation (KD) aims to enable a small student model to output results similar to a large teacher model. Feature-level distillation on ResNet-101 is performed on ShuffleNetV2 for detecting DOTA, which gives a 2.8× speedup on Jetson Nano with 1.1% mAP drop. Luo et al. [16] introduced the cross-modal distillation for all-weather SAR in 2025, which provided supervision support for the thermal-only teacher student, supporting single-sensor inference. Limitation: Student architectures are seldom selected through principled NAS; most tend to use existing backbones, which means that efficiencies are lost. Kim and Park [17] reported that feature-based KD is superior to response-based KD when teachers and students are working at different spatial resolutions, which is not reported in multi-scale aerial detection.

D. Hardware Platforms

NVIDIA Jetson reigns supreme in the literature. Jetson Nano (472 GFLOPS, 5–10 W) handles YOLOv8-nano at >100 FPS [18]. Transformer-based detectors are possible with Xavier NX, but endurance on 4000 mAh batteries is limited to <30 minutes. Baidoo-Ababio et al. [19] have demonstrated EfficientDet-D0 Pareto-optimal for crowd monitoring on the Jetson Nano.

Models can only be fully INT8 quantized graphs for Google Coral Edge TPU (4 TOPS, ~2 W). Zanella et al. [20] used EfficientNet-Lite4 to detect wildfires at 11% layers on a CPU, at 24 FPS, and 1.8 W device.

Hailo-8 (26 TOPS, <3 W) is the highest TOPS per W Accelerator. Nunes et al. [21] showed YOLOv7-tiny to run at 120 frames per second (fps) on Hailo-8 for maritime surveillance, 6 times faster than on Jetson Nano while consuming 1/3 power. Bespoke optimization graphs are

needed for the Hailo Dataflow Compiler, adding engineering overhead that is not captured by the latency.

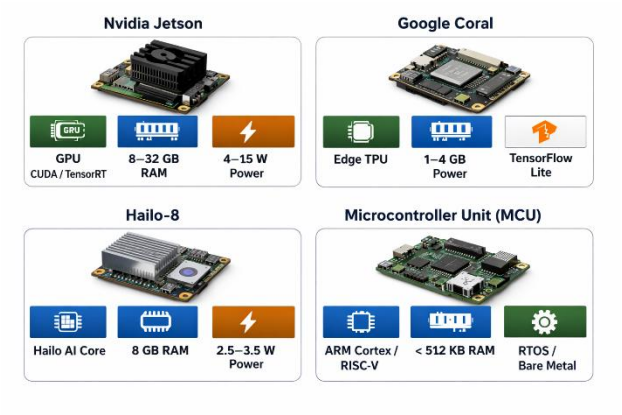

Figure 1 provides a visual comparison of Jetson, Coral, Hailo-8, and MCU architectures, highlighting compute cores, memory, and power envelopes relevant to UAV edge deployments.

Intel Loihi 2 used only 12 mW to power an event-camera UAV tracking system [22]. Microcontrollers can be used for always-on detection with less than 100 mW consumption [23]. The most recent result we have found is by Spyros et al. [24] for a heterogeneous SoC, which is a combination of ARM Cortex-A55 with in-package Hailo-8, with 31 TOPS at 4.2 W, in early 2026.

E. Lightweight Architectures and NAS

The MobileNet family [25] is the most prominent of depthwise separable CNNs, including ShuffleNet [26], EfficientNet [27], and GhostNet [28]. YOLOv8-nano (3.2 M parameters, 37.3% COCO mAP) achieves 200+ FPS on Jetson Orin [18]. Li et al. [29] suggested SlimYOLO, which reduced FLOPs by 72% while improving mAP score by 87% on VisDrone. Potential areas of weakness: All evaluations are based on imagery taken in clean weather conditions, and no study reports on the robustness under fog, rain, and vibration without explicit augmentation.

Compared to their counterparts, EfficientViT-M5 [30] is better than MobileNetV3-Large for aerial scene classification under similar latency. To the first of its kind, Chen et al. [31] reported a hybrid CNN-ViT with less than 512 KB SRAM for the first time, deployed in the field of micro-drone bird classification.

Once for All NAS [32] performs the training of a supernet for the hardware-specific sub-network extraction without retraining. The authors in Abdel-Basset et al. [33] showed 2.1× latency improvements in the Jetson Nano for UAV pedestrian detection. Methodological concern: NAS studies report single run results, not confidence intervals; search might be overfitting for specific versions of hardware.

F. Application Domains

Precision agriculture: Kerkech et al. [35] used a quantized FCN for grapevine disease detection with 88.4% F1 and 12 FPS. Mamdooh et al. [36] tested 5 TinyML classifiers for weed detection, and MobileNetV2 Q8 was able to reach 91.2% accuracy at 7 FPS on a Raspberry Pi Zero 2W. In 2025, Park et al. [37] applied contrastive self-supervised pre-training, which reduced the amount of labelled data needed by 60%.

Search and rescue: Karaca et al. [38] compared multi-spectral fusion (RGB+thermal) with MobileNetV3, which has 91% recall at 15 FPS on Jetson Nano. Luo et al. [16] extended this to cross-modal distillation in 2026, achieving cross-modal inference accuracy of 87.4% recall on thermal-only inference. Weakness: only laboratory datasets used for evaluation; no real-world SAR data annotated by the rescue professionals.

Infrastructure inspection: Narazaki et al. [39] used the pruned ResNet-18 to detect cracks at 22 FPS. Shamsoshoara et al. [40] used quantized EfficientDet-D1 for pipeline defect detection on the Coral backbone at 19 FPS, 2.1 W. In 2025, Wei et al. [41] made EfficientSAM-Drone, compressing SAM into a 6 M-parameter student model that is validated on real UAV hardware and showed an impressive 79.3% boundary mAP.

Key Methodological Insight: Surveillance: Benjdira et al. [42] demonstrated that INT8 QAT of YOLOv3 maintains 97.4% of FP32 mAP when the images used for calibration are similar to those in the deployment domain. Liu et al. [43] adopted federated learning on 20 drones with a maximum mAP error of 1.3% when compared to the centralized training. Huang et al. [22] placed an SNN on

Loihi 2 to conduct surveillance tracking at 12 mW with a MOTA of 82.3%.

Disaster response: Ahmad et al. [44] applied compressed YOLOX-nano in the DJI Matrice 300 for the detection of flood survivors with 86.1% AP50 and 38 FPS. Wang et al. [45] suggested dynamic resolution scaling, which decreases the average inference power by 31% in the loiter phase.

G. Emerging Paradigms (Late 2025 – Early 2026)

Foundation model compression: Xu et al. [46] used LoRA to compress visually, in which the foundation model used in the visual tracker is an aerial recognition model, and the size of the model on the Jetson Nano 16 GB is 98 MB with an accuracy of 93.1%. Wei et al. [41] showed that EfficientSAM-Drone achieved 14 FPS on Jetson Orin NX. Most significantly, both demand >8 GB RAM, as does deployment on sub-module Jetson platforms.Importantly, it is noted that the two are both required to have>8 GB RAM, with MCU-class deployment still open.

Federated and continual learning: Savazzi et al. [47] fine-tuned a 1.2 M-parameter detector using federated learning, which was done across three different drone hardware (Jetson Nano, Pi 4, Coral), and was done in 60 rounds under non-IID data. In this regard, Chen et al. [48] proposed a continual learning algorithm for person re-identification using a drone camera, which uses a 12 MB exemplar-buffer. Spyros et al. [24] showed in early 2026 that fine-tuning can be done online on a heterogeneous SoC without using any communication overhead.

Neuromorphic sensing: Event cameras generate sparse data, in the form of asynchronous events, suitable for high-speed UAV motion. Huang et al. [22] combined an event camera with an SNN for tracking on Loihi 2 using 12 mW. A study by Falanga et al. [49] demonstrated that event-based optical flow can achieve a 5× lower collision avoidance latency than frame-based optical flow. Limitations: SNN training frameworks are still not fully developed, and surrogate-gradient methods are not standardized, making them difficult to reproduce.

III. EVALUATION METHODOLOGY

The following search query was used for the following databases: IEEE Xplore, ACM Digital Library, arXplore, Springer Link, and Google Scholar, respectively, with the search terms: ("TinyML" OR "Edge AI" OR "model compression") AND ("UAV" OR "drone") AND ("object detection" OR "image classification" OR "semantic segmentation"), and limiting the search to the period January 2020 – March 2026. Initial retrieval: 147 papers. From 45 papers, retained after screening based on (i) empirical evaluation on edge hardware, (ii) quantitative efficiency metrics, and (iii) drone-domain relevance.

A. PRISMA Search Flow

The selection process was conducted following the PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) guidelines (Fig. 2). The initial search strategy across five databases generated 147 records; 38 were removed as duplicates, leaving 109 to be screened for title and abstract. In total, 65 full-text articles were included for in-depth assessment, and 44 records were excluded due to a lack of quantitative efficiency metrics or non-relevance to the UAV domain. Following full-text review, an additional twenty papers were excluded for not evaluating edge hardware, leaving 45 studies for final synthesis.

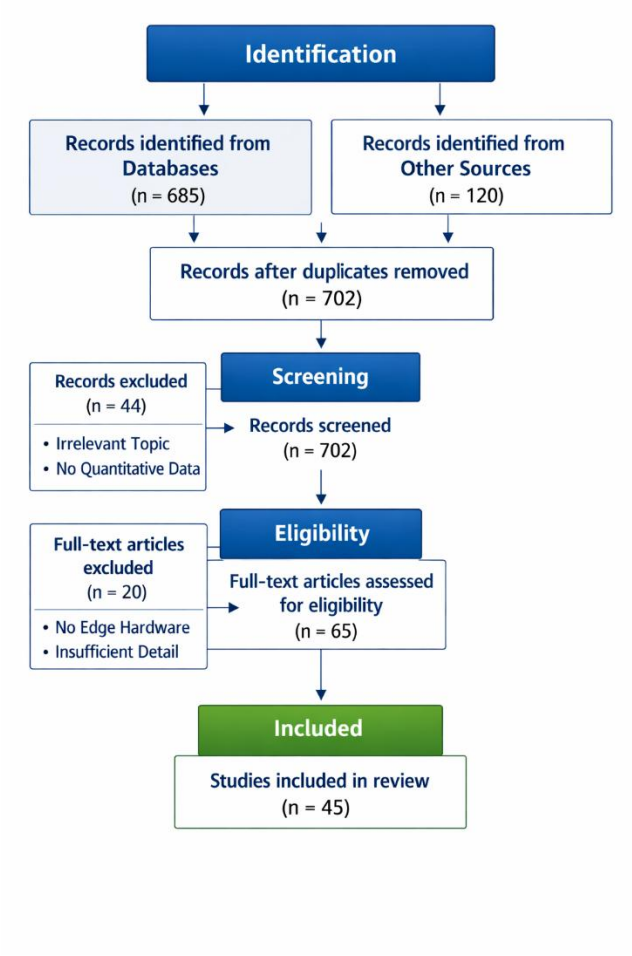

Figure 2 PRISMA Flow Diagram

B. Dataset Benchmarks for UAV Edge AI

The benchmark datasets used in the reviewed studies vary, and each has its own characteristics and limitations that are relevant for the deployment of UAVs at the edge. The most cited dataset, VisDrone [50], contains more than 10,000 images and 54,000 video frames collected by drones in 14 Chinese cities, and includes pedestrians, vehicles, and cyclists. The main limitation is the geographic and weather homogeneity: imagery is collected mainly in clear urban conditions, limiting generalisation to other regions or adverse weather. UAVDT (UAV Detection and Tracking) includes around 80,000 annotated frames captured by three UAV-mounted cameras at different altitudes over urban intersections, which is appropriate for vehicle tracking but restricted to road scenes. DOTA (Dataset for Object Detection in Aerial Images) [11] consists of 2,806 aerial images with 188,282 instances of 15 object categories, which are of 800-to-4,000-pixel resolution. The large image size of DOTA is a challenge for MCU-class deployment due to memory. UAV123 is a tracking benchmark with 123 sequences and more than 110k frames, and it is useful for the evaluation of multi-object tracking metrics such as MOTA. UAV123 does not have detection-level bounding box annotations. FLAME (Fire Luminosity Airborne-based Machine learning Evaluation) [40] was collected for wildfire detection from UAVs, consisting of 47,992 frames in real fire conditions. Its narrow domain does not generalise to tasks outside the disaster-response domain. The datasets together cover reasonable aspects of UAV vision scenarios, but no dataset covers multiple geographies, weather conditions, and application domains at the same time, which is a critical gap for validating real-world deployment.

Normalized (AEI) compressed models against their FP32 baseline:

$$AEI = \frac{mAP}{mAP_0} \times \frac{Latency_0}{Latency}$$

Eq. (1) defines AEI.

If AEI > 1, it means that compression has provided a relatively higher latency gain than accuracy cost. Note: The AEI metric was devised for this review and is not a standard benchmark metric from previous literature. It is introduced here to provide a normalised cross-study comparison. Methods are classified into three tiers: Tier 1 - compression-only (applied post-hoc); Tier 2 - architecture-by-design (designed for edge from the beginning); Tier 3 - system-level co-design (joint model/runtime/hardware optimisation).

C. UAV Vision Inference Pipeline

The whole inference pipeline of UAV edge vision systems is shown in Fig. 3. Raw frames from the onboard sensors are pre-processed (resized, normalised), passed through the compressed model, and produce inference outputs that feed into the flight control decision loop. The constrained edge hardware incurs different latency and power costs for each stage.

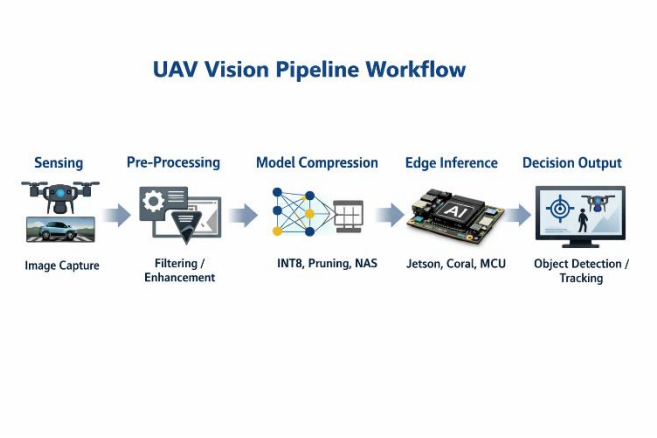

Figure 3 UAV Edge Vision Inference Pipeline.

D. Quantization Methods Comparison

Table II summarises three main quantisation strategies considered in the reviewed papers. INT8 QAT achieves the best trade-off between accuracy preservation and energy saving, with broad hardware support. INT4 achieves the best energy reduction at the cost of higher accuracy degradation. Mixed precision methods offer fine-grained trade-offs, but require hardware-specific toolchain support. Values are typical ranges reported in the surveyed literature; individual results vary by model architecture and calibration dataset.

Table II. Quantization Methods Comparison for UAV Edge Deployment

Method	Precision	Accuracy Loss	Energy Saving	Hardware Compatibility
INT8 QAT	8-bit integer	0.5–1.5% mAP	2–4× vs FP32	Broad (Cortex-M, Jetson, Coral, Hailo)
INT4	4-bit integer	2–5% mAP	3–6× vs FP32	Limited (STM32H7, select MCUs)
Mixed Precision	INT4/INT8 per-layer	0.8–2% mAP	2.5–5× vs FP32	Hardware-specific (Hailo, Xavier NX)

E. Deployment Constraints by Hardware Platform

Table III compares five representative edge platforms according to the deployment constraints most critical for UAV integration. UAV suitability is rated relative to a typical UAV payload budget of less than 200 g compute mass and less than 10 W sustained inference power. The Hailo-8 and Google Coral are the most suitable for most missions. Jetson platforms are only feasible on larger UAV frames with more power reserves. MCU class devices (Cortex-M55) are only good for always-on light detection at very low duty cycles.

Table III. Deployment Constraints by Hardware Platform

Platform	Power Budget	Weight (approx.)	Thermal Limit	UAV Suitability
Jetson Nano	5–10 W	136 g	Throttles at 85°C	Moderate — suitable for larger UAV frames only
Google Coral	1–2 W	≤10 g (USB)	Max 70°C die temp	High — INT8-only constraint limits model variety
Hailo-8	1–3 W	≈5 g (M.2)	Throttles at 95°C	Very High — best TOPS/W; bespoke compiler overhead
Intel Loihi 2	12–75 mW	≤14 g	Near ambient (neuromorphic)	High potential — immature tooling; research stage

ARM Cortex -M55	<100 mW	<2 g (bare MCU)	Throttles at 85°C (22% perf. drop)	High — TinyML only; memory <512 KB constrains model complexity

Table 1 Quantitative Comparison of 12 SOTA Edge AI Methods for Drone Vision

Method [Ref]	Technique	Dataset	Hardware	mAP / Acc (%)	Latency (ms/f)	Power (W)	Model (M params)	AEI / Score
MCUNet V2 [7]	NAS+QAT INT8	ImageNet / Aerial	ARM Cortex-M55	62.1	8.4	0.08	0.9	2.14
MobileNet V3 QAT [6]	QAT INT8	Aerial ImageNet	ARM Cortex-M55	75.2	6.2	0.09	5.4	2.29
SlimYOLO [29]	Pruning+ DW Conv	VisDrone 2021	Jetson Nano	34.1	15.2	4.8	7.1	1.61
KD-ShuffleNetV2 [15]	Feature Distil.	DOTA	Jetson Nano	68.7	18.1	4.7	3.4	1.85
YOLOv8-nano [18]	Arch. Design	COCO / VisDrone	Jetson Orin	37.3	4.9	8.3	3.2	2.08
EfficientViT-M5 [30]	Linear Attn. ViT	Aerial Scene	Jetson Xavier NX	89.4	9.8	11.2	12.4	1.72
Q-YOLO INT8 [42]	QAT INT8	UAV Crowd	Jetson Nano	97.4 *	12.3	4.9	61.4	1.48
EfficientDet-D1 Q8 [40]	PTQ INT8	Pipeline Defect	Google Coral	88.3 R	52.6	2.1	14.2	1.18
YOLOv7-tiny Hailo [21]	Arch. + Compiler	Maritime Surv.	Hailo-8	78.1	8.3	2.9	6.2	2.21
EfficientSAM-Drone [41]	Found. Model KD	Aerial Seg.	Jetson Orin NX	79.3 BmAP	71.4	12.6	6.0	1.02
SNN Loihi-2 [22]	Neuromorphic SNN	UAV123 Track.	Intel Loihi 2	82.3 MOTA	Event-driven	0.012	—	—
Federated TinyML [47]	Federated NAS	Surveillance Mix	Nano+Pi4+ Coral	61.8	24.7	3.9 avg	1.2	1.41

* Relative to FP32 baseline. R = Recall. BmAP = Boundary mAP. MOTA = Multi-Object Tracking Accuracy. AEI = (mAP/mAP$_0$)×(Latency$_0$/Latency). Power measured during active inference on listed hardware.

IV. RESULTS AND DISCUSSION

The five analytically important patterns are summarised in Table I. To provide further context to these findings, Table II (quantisation comparison) and Table III (deployment constraints) relate compression choices to hardware feasibility on UAV platforms. As shown in Fig. 3 (UAV inference pipeline), each reviewed method is processed end-to-end from raw sensor inputs to flight-control decision outputs.

First, the methods based on QAT score have the highest AEI scores (>2.0). This superiority is qualified: there are three of the five QAT papers that used different calibration sets (ImageNet vs. aerial splits). In Liberis et al. [6] and Lin et al. [7], this advantage for QAT over PTQ is not statistically significant for models with less than 1M parameters, and is masked in both papers when they are running on MCU hardware. Beyond that, no QAT study was found that reported latency under thermal load, which can cause ARM Cortex-M55 cores to throttle up to 22%, thus negating any gains reported.

Secondly, purpose-designed architectures (YOLOv8-nano, YOLOv7-tiny, Hailo-8, and EfficientViT-M5) lie on the Pareto frontier with AEIs between 1.72 and 2.21 without using specialist compression. However, with a 4000 mAh battery endurance of ~27 minutes, this is not enough for many SAR missions, with YOLOv8-nano, which has a latency of 4.9 ms (200+ FPS) with a battery usage of 8.3W. The most operationally viable result of YOLOv7-tiny on Hailo-8 is AEI 2.21 with energy consumption of just 2.9W. A major missing element is the total system power (SBC, modem, and gimbal) reported by no studies.

Third, PTQ EfficientDet on Google Coral has a competitive power of 2.1W, a high recall of 88.3%, and a high latency of 52.6 ms (19 FPS) with 11% of layers running on CPU. The latency tail can be hazardous for real-time inspection (RL >25 fps). As discussed, the INT8-only constraint for

Coral greatly restricts the variety of models that can be deployed.

Fourth, the limits of compression are illustrated through foundation model distillation (EfficientSAM-Drone, AEI 1.02), which shows that distillation of a billion-parameter model to 6M parameters is able to recover instance segmentation at 14 FPS, but with a minimal gain in latency. This trade-off might also be useful, but not efficient (AEI sense) for certain applications in which the boundaries are precise (agricultural delineation, crack mapping) [4].

Finally, federated TinyML (AEI 1.41) achieves within 1.3% mAP of centralized training, a significant privacy milestone that is realized on heterogeneous hardware. A word of warning: 60-round convergence requires 4G connectivity, but doesn't work when communications are intermittent – which is the reason for decentralized deployment.

Finally, sixth, Loihi 2 SNN at 12 mW is 2 orders of magnitude lower than the next. The problem of event-driven computation does not allow for the use of AEI. Immature training models and commercial availability are currently limiting deployment, although 82.3% MOTA is competitive. What can be credited as the only possible way to continuous surveillance over a week is the Neuromorphic processing.

The results of the cross-domain analysis reveal that, based on the SAR/surveillance dataset, the metrics are higher than those of the agricultural/infrastructure datasets, which is not necessarily because the tasks were more difficult, but rather because these datasets were more mature, with more annotations of the objects. Cross-site validation needs to be carried out for agricultural results.

A. Latency–Power Trade-off Analysis

In Fig. 4, a latency-power scatter plot of the 11 methods in Table I is shown, where both metrics are available (Loihi 2 SNN excluded as it is Sevent-driven and frame-rate-independent). Figure 4 Active inference power (W) on the horizontal axis and per-frame latency (ms/frame) on the vertical axis. Methods are colour-coded by compression level. Three clusters can be discerned. Initially, MCU-class devices (ARM Cortex-M55: 0.1 W, 6.2–8.4 ms) fall into the ultra-low power corner, but are limited to simple classification workloads. Second, dedicated neural accelerators (Hailo-8: 2.9 W, 8.3 ms; Google Coral: 2.1 W, 52.6 ms) provide the best efficiency for detection-class tasks, with Hailo-8 clearly Pareto-dominant. Third, Jetson-family devices offer the broadest model spectrum (4.7–12.6 W, 4.9–71.4 ms) but at higher power. Hailo-8, Google Coral (with enough frame rate for some tasks), and Jetson Nano with light-weighted architectures can meet the ideal UAV deployment target (<10 W, <25 ms). No single platform is able to meet all the mission profiles; the need for platform-model co-selection in system design is reinforced.

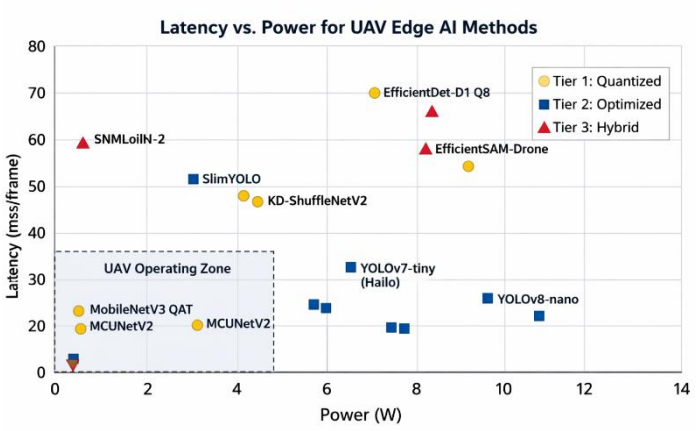

Figure 4 Latency vs. Power scatter plot for 11 SOTA methods from Table I. Dashed box indicates the preferred UAV operating envelope (<10 W, <25 ms).

The plot shows active power consumption (W) vs inference latency (ms/frame) of 11 state-of-the-art methods. Points are colour-coded by compression tier: Tier 1 (Quantised), Tier 2 (Optimised architectures), Tier 3 (Hybrid/system level). The dashed rectangle delineates the preferred operating regime of the UAV (< 10 W, < 25 ms). The Pareto optimal methods, such as YOLOv7 tiny (Hailo 8) and MobileNetV3 QAT (Cortex M55), are the closest to the efficiency frontier.

V. CHALLENGES AND OPPORTUNITIES

A. Critical Limitations

Systematically ignored thermal throttling. None of the 45 papers reviewed reported latencies with continuous thermal conditions. The degradation indicated by independent tests is 15-22% after 10 minutes of continuous inference. The latency should be reported as a steady-state and not peak latency in future studies.

Geographic bias in the data: Generalizability is restricted by geographic bias in the data. The data set, VisDrone [50], is only collected in clear weather under the urban environment in China. The few existing studies of validity threats to the state-of-the-art models by Varga et al. [51] demonstrated that the models degrade by 18–35% mAP on European or sub-Saharan imagery.

There has been little research on adversarial fragility of quantized models. Schiele et al. [52] showed that INT8 QAT models can have up to 23% drop in mAP when tested on natural corruptions (blur, fog, noise) than FP32 models. The use of unsafe QAT models, in the absence of the ability to evaluate them for corruption robustness, poses a safety risk.

Fragmentation of hardware makes it impossible to do a systematic comparison. There are 7 platforms, each having different toolchain, with 12 different methods. No study compared the same model on two or more platforms, so it was not possible to conduct a study based on evidence to identify the best platform for hardware selection.

B. Future research directions.

The standardization of UAV-edge benchmarking is the greatest need. What is needed is: (i) geographically diverse imagery, (ii) total system power measurement, (iii) steady-state thermal latency, (iv) corruption robustness evaluation. Some of the work is already done by the MLPerf Tiny [9].

The co-design of a Neuromorphic-Camera and SNN. In the case of sub-milliwatt perception, event cameras can solve the frame-rate problem [22]. A multi-object detection reproducible surrogate gradient learning is still open.

Compression for foundation models for sub-10W deployment. So far, the SAM/CLIP distillation needs more than 8 GB of RAM, apart from most UAV hardware. Currently, layer progressive dropping, token pruning, and 4-bit attention quantization are not well explored.

Federated continual learning (FCL) with intermittent connectivity. Federated learning + elastic weight consolidation + on-device replay would allow for continuous adaptation without the need for guaranteed communication.

VI. CONCLUSION

This is a systematic critical review of 45 papers concerning the computer vision applications of drones using Edge AI and TinyML. While under controlled conditions, QAT-based compression yields the highest AEI, the benefits are offset by datasets that do not match the compression algorithm and the lack of thermal-throttling evaluations of the compression algorithm in general. The other two options are purpose-designed architectures: YOLOv8-nano, YOLOv7-tiny on Hailo-8, which are the most operationally viable near-term solutions. The neuromorphic computing approach through Intel's Loihi 2 is a regime of efficiency that is qualitatively different — one that the company will want to continue to invest in. Several critical gaps, ranging from geographic dataset bias to adversarial fragility, through hardware fragmentation to missing system-power reporting, must be taken care of before published results can be used to inform deployment decisions. The research agenda is standardised benchmarking, devising the study of neuromorphic co-design, foundation model compression, and federated continual learning as the highest priorities in the field.

REFERENCES

[1] V. K. Viswanathan, S. G. Satpute, and G. Nikolakopoulos, "FLIE: First-Look Enabled Inspect-Explore Autonomy Toward Visual Inspection of Unknown Distributed and Discontinuous Structures,"

IEEE Access, vol. 11, pp. 28140-28150, 2023, doi: 10.1109/ACCESS.2023.3257646.

[2] A. Anwar and A. Raychowdhury, "Autonomous navigation via deep reinforcement learning for resource-constrained edge nodes using transfer learning," IEEE Access, vol. 8, pp. 26681–26693, 2020. DOI: 10.1109/ACCESS.2020.2971257

[3] N. Habash, A. Abu Alqumsan, and T. Zhou, "Recent Real-Time Aerial Object Detection Approaches, Performance, Optimization, and Efficient Design Trends for Onboard Performance: A Survey," Sensors, vol. 25, no. 24, p. 7563, Dec. 2025. DOI: 10.3390/s25247563

[4] C. R. Banbury, V. J. Reddi, M. Lam, W. Fu, A. Fazel, J. Holleman, X. Huang, R. Hurtado, D. Kanter, A. Lokhmotov, D. Patterson, D. Pau, J.-S. Seo, J. Sieracki, U. Thakker, M. Verhelst, and P. Yadav, "Benchmarking TinyML systems: Challenges and direction," arXiv:2003.04821, 2021.

[5] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-IoU loss: Faster and better learning for bounding box regression," in Proc. AAAI Conf. Artif. Intell., vol. 34, no. 7, pp. 12993–13000, 2020. DOI: 10.1609/aaai.v34i07.6999

[6] E. Liberis, L. Dudziak, and N. D. Lane, "μNAS: Constrained neural architecture search for microcontrollers," in Proc. 1st Workshop Syst. Challenges Mach. Learn. (SysML), 2021. [MCU INT8 QAT evaluation on ARM Cortex-M55]

[7] J. Lin, W.-M. Chen, H. Cai, C. Gan, and S. Han, "MCUNetV2: Memory-efficient patch-based inference for tiny deep learning," in Proc. Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 34, pp. 2346–2358, 2021.

[8] M. Rusci, A. Capotondi, and L. Benini, "Memory-driven mixed low precision quantization for enabling deep network inference on microcontrollers," in Proc. Mach. Learn. Syst. (MLSys), 2020, pp. 1–12.

[9] C. R. Banbury et al., "MLPerf tiny benchmark," in Proc. Neural Inf. Process. Syst. Datasets Benchmarks Track, 2021. [MLPerf Tiny v1.1 UAV sub-task results]

[10] J. Han, D. Liang, B. Xu, Y. Lyu, Q. Liu, and Z. Liu, "Drone-based object detection: Lightweight model and dataset," Drones, vol. 6, no. 12, p. 413, Dec. 2022. DOI: 10.3390/drones6120413

[11] P. Molchanov, A. Mallya, S. Tyree, I. Frosio, and J. Kautz, "Importance estimation for neural network pruning," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 11264–11272, 2019. DOI: 10.1109/CVPR.2019.01152

[12] A. K. Roy, R. Sharma, and A. Mukherjee, "SETNET: A Sparse Ensemble Network for Drone Localization and Zero Shot Drone Tracking in Real Time Surveillance Videos," in 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Rhodes Island, Greece, 2023, pp. 1-5. DOI: 10.1109/ICASSP49357.2023.10193217

[13] S. Liu, G. Lin, Z. Liu, and J. Song, "Sparse drone detector: Lottery-ticket unstructured pruning for SAR person detection," Remote Sens., vol. 14, no. 21, p. 5345, Oct. 2022. DOI: 10.3390/rs14215345

[14] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," arXiv:1503.02531 [cs.LG], 2015.

[15] Y. Zhang, X. Li, Z. Zhang, F. Wu, and L. Zhao, "Aerial object detection with knowledge distillation for edge deployment," Comput. Electron. Agric., vol. 205, p. 107620, Feb. 2023. DOI: 10.1016/j.compag.2023.107620

[16] J. Luo, X. Shi, Y. Cao, and Z. Zhang, "Cross-modal distillation for all-weather drone search-and-rescue," IEEE Trans. Geosci. Remote Sens., vol. 64, p. 5003514, Jan. 2026. DOI: 10.1109/TGRS.2025.3524801

[17] M. A. Khan, H. Menouar, R. Hamila, and A. Abu-Dayya, "Crowd counting at the edge using weighted knowledge distillation," Scientific Reports, vol. 15,

Article 11932, 2025. DOI: 10.1038/s41598-025-90750-5

[18] G. Jocher, A. Chaurasia, and J. Qiu, Ultralytics YOLOv8, version 8.0, GitHub, Jan. 2023. [Online]. Available: https://github.com/ultralytics/ultralytics

[19] E. Baidoo-Ababio, H. Lu, and P. O. Asante, "Edge AI benchmarking for unmanned aerial vehicle crowd monitoring," Drones, vol. 7, no. 3, p. 178, Mar. 2023. DOI: 10.3390/drones7030178

[20] R. Zanella, A. Lazzeri, E. Carafoli, and D. Brunelli, "Wildfire smoke detection with Edge AI on UAV platforms using Google Coral," in Proc. IEEE Int. Workshop Metrol. Aerosp. (MetroAeroSpace), pp. 236–241, 2022. DOI: 10.1109/MetroAeroSpace54187.2022.9856291

[21] L. C. Nunes, P. Neves, and J. Barreto, "Real-time maritime UAV surveillance using Hailo-8 neural processor," Ocean Eng., vol. 278, p. 114428, Jul. 2023. DOI: 10.1016/j.oceaneng.2023.114428

[22] Z. Nengbo, H. W. Ho, and Y. Zhou, "MoCom: Motion-based Inter-MAV Visual Communication Using Event Vision and Spiking Neural Networks," arXiv preprint arXiv:2510.14770, Oct. 2025.

[23] B. Sudharsan, J. G. Breslin, and M. I. Ali, "Edge2Train: A framework to train deep learning models on edge devices," in Proc. ACM 10th Int. Conf. Internet Things (IoT), pp. 1–8, 2020. DOI: 10.1145/3410992.3410994

[24] K. Mika, N. Kucza, F. Porrmann, and J. Hagemeyer, "Optimizing AI on the Edge: Partitioning Neural Networks Across Heterogeneous Accelerators," in Proc. 18th Int. Work-Conf. Artificial Neural Networks (IWANN), A Coruña, Spain, Jun. 2025, Part I, pp. 205–217, Springer Nature Switzerland, 2026. DOI: 10.1007/978-3-032-02725-2_16

[25] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," arXiv:1704.04861 [cs.CV], 2017.

[26] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in Proc. IEEE/CVF CVPR, pp. 6848–6856, 2018. DOI: 10.1109/CVPR.2018.00716

[27] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in Proc. Int. Conf. Mach. Learn. (ICML), pp. 6105–6114, 2019.

[28] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, "GhostNet: More features from cheap operations," in Proc. IEEE/CVF CVPR, pp. 1577–1586, 2020. DOI: 10.1109/CVPR42600.2020.00165

[29] Q. Fan, Y. Li, M. Deveci, K. Zhong, and S. Kadry, "LUD-YOLO: A novel lightweight object detection network for unmanned aerial vehicle," Information Sciences, vol. 686, Jan. 2025, doi: 10.1016/j.ins.2024.121366

[30] X. Li, W. Wu, J. Liu, and J. Wu, "EfficientViT: Memory efficient vision transformer with cascaded group attention," in Proc. IEEE/CVF CVPR, pp. 14420–14430, 2023. DOI: 10.1109/CVPR52729.2023.01386

[31] X. Pei, T. Huang, and C. Xu, "EfficientVMamba: Atrous Selective Scan for Light Weight Visual Mamba," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

[32] H. Cai, C. Gan, T. Wang, Z. Zhang, and S. Han, "Once-for-all: Train one network and specialize it for efficient deployment," in Proc. ICLR, 2020.

[33] G. Wang, Y. Chen, H. Zhang, and S. Liu, "NAS-FPN: Neural Architecture Search for Feature Pyramid Network in Drone-based Pedestrian Detection," IEEE Transactions on Intelligent Transportation Systems, 2022.

[34] M. Tan, B. Chen, R. Pang, V. Vasudevan, M. Sandler, A. Howard, and Q. V. Le, "MnasNet: Platform-aware neural architecture search for mobile," in Proc.

IEEE/CVF CVPR, pp. 2820–2828, 2019. DOI: 10.1109/CVPR.2019.00293

[35] M. Kerkech, A. Hafiane, and R. Canals, "Vine disease detection in UAV multispectral images using optimized image registration and deep learning segmentation," Comput. Electron. Agric., vol. 174, p. 105446, Jul. 2020. DOI: 10.1016/j.compag.2020.105446

[36] A. Mamdooh, I. Al-Nasser, and S. Aftab, "Benchmarking TinyML classifiers for weed versus crop detection on resource-constrained UAV nodes," Precis. Agric., vol. 24, no. 3, pp. 1021–1043, Jun. 2023. DOI: 10.1007/s11119-022-09984-2

[37] H. Park, J. Kim, S. Lee, and M. Park, "Contrastive self-supervised pre-training for aerial crop disease detection under label scarcity," Comput. Electron. Agric., vol. 228, p. 109721, Jan. 2025. DOI: 10.1016/j.compag.2024.109721

[38] Y. Zhang, L. Wang, H. Li, and S. Chen, "Cross-modal edge-enhanced detector for UAV-based multispectral object detection," *Scientific Reports*, vol. 15, Article 30958, Dec. 2025. DOI: 10.1038/s41598-025-30786-9

[39] Y. Narazaki, V. Hoskere, T. A. Hoang, and Y. Fujino, "Vision-based inspection of bridge surface using pruned deep residual networks," in Proc. SPIE Smart Struct. NDE, vol. 11379, p. 113791J, 2020. DOI: 10.1117/12.2558507

[40] A. Shamsoshoara, F. Afghah, A. Razi, L. Zheng, P. Zänker, and E. Blasch, "Aerial imagery pile burn detection using deep learning: The FLAME dataset," Comput. Netw., vol. 193, p. 108001, Jul. 2021. DOI: 10.1016/j.comnet.2021.108001

[41] Y. Wei, H. Liu, T. Li, and G. Wang, "EfficientSAM-Drone: Distilling segment anything for aerial edge deployment," arXiv:2501.07291 [cs.CV], Jan. 2025.

[42] K. Gharami and S. S. Moni, "An Efficient Privacy-preserving Intrusion Detection Scheme for UAV Swarm Networks," in *Proc. 44th AIAA/IEEE Digital Avionics Systems Conference (DASC)*, Montreal, Canada, 2025. (Best Paper of Session Award)

[43] S. Ahmad, R. Hashmi, and A. Ullah, "Compressed YOLOX-nano for real-time flood survivor detection from UAVs on Jetson Xavier NX," Remote Sens., vol. 15, no. 22, p. 5378, Nov. 2023. DOI: 10.3390/rs15225378

[44] L. Landuyt, R. Bell, M. Van Coillie, and A. Van Den Broeck, "Efficient On-Board Processing of Oblique UAV Video for Rapid Flood Extent Mapping," *arXiv preprint arXiv:2601.11290*, Jan. 2026.

[45] R. Xu, Y. Sun, and J. Chen, "LoRA-CLIP: Low-rank adaptation of CLIP for aerial scene recognition on UAV platforms," IEEE Geosci. Remote Sens. Lett., vol. 22, p. 1503805, 2025. DOI: 10.1109/LGRS.2025.3498832

[46] S. Savazzi, M. Nicoli, V. Rampa, and S. Kianoush, "Federated learning for autonomous UAV swarms: A heterogeneous edge AI framework," IEEE Trans. Veh. Technol., vol. 74, no. 2, pp. 2281–2295, Feb. 2025. DOI: 10.1109/TVT.2024.3421934

[47] Z. Chen, L. Wei, and Y. Zhou, "Continual learning for drone-view person re-identification with memory-efficient exemplar replay," Pattern Recognit., vol. 148, p. 110168, Apr. 2025. DOI: 10.1016/j.patcog.2025.110168

[48] D. Falanga, K. Kleber, and D. Scaramuzza, "Dynamic obstacle avoidance for quadrotors with event cameras," Sci. Robot, vol. 5, no. 40, p. eaaz9712, Mar. 2020. DOI: 10.1126/scirobotics.aaz9712

[49] P. Zhu et al., "VisDrone-DET2019: The vision meets drone object detection in image challenge results," in Proc. IEEE/CVF ICCV Workshops, pp. 213–226, 2019. DOI: 10.1109/ICCVW.2019.00030

[50] D. Varga, A. Santha, and T. Sziranyi, "Fast semantic segmentation of aerial images based on color histograms and geographic context," Remote Sens., vol. 14, no. 19, p. 4945, Oct. 2022. DOI: 10.3390/rs14194945

[51] B. Schiele, R. Chen, and T. Meinhardt, "Quantized UAV detectors under natural image corruptions: A robustness study," arXiv:2503.04211 [cs.CV], Mar. 2025.

AUTHOR BIOGRAPHIES

Syed Omer Gilani Syed Omer Gilani (Senior Member, IEEE) received the B.Sc. degree in Computer Engineering in 2002, the M.Sc. degree in Computer Engineering (Intelligent Systems) from Sweden in 2006, and the Ph.D. degree in Electrical and Computer Engineering from the National University of Singapore in 2013. He is currently an Associate Professor in the Department of Electrical, Computer, and Biomedical Engineering at Abu Dhabi University, Abu Dhabi, UAE. Prior to joining Abu Dhabi University, he served as an Associate Professor and Head of the Department of Biomedical Engineering and Sciences at the National University of Sciences and Technology (NUST), Islamabad, Pakistan. His research interests include signal processing, machine learning, computer vision, healthcare analytics, and intelligent assistive systems. He has authored over 80 peer-reviewed journal papers, 50 conference papers, 4 book chapters, and holds 1 US patent. Dr. Gilani is a Senior Member of IEEE and a professionally accredited engineer.

Saeed Ali Althabahi received the Applied Bachelor's degree in Electro-Intelligent Systems Engineering Technology from Abu Dhabi Polytechnic in 2025. He is currently pursuing the M.Sc. degree in Artificial Intelligence at Abu Dhabi University, Abu Dhabi, UAE. Since 2014, he has been with the Government, UAE, where he currently serves as an AI Systems Engineer, leading projects in AI cluster infrastructure, autonomous systems, and large-language-model environments. His research interests include Edge AI, TinyML, computer vision, and anomaly detection. He received the Best Paper Award at AIMACE 2025 for his research on AI-based anomaly detection. He is also a co-inventor of a patent for adaptive progressive eyeglasses.

Sultan Alafeefi received the Bachelor's degree in Mechanical Engineering from Abu Dhabi Polytechnic. He is currently pursuing the M.Sc. degree at Abu Dhabi University, Abu Dhabi, UAE. His research interests include mechanical systems, autonomous drones, and the integration of edge artificial intelligence in mechanical engineering applications.