

Research Article /Review Article:

Comparative Evaluation of Corruption Robustness and Confidence Calibration in Real-Time Object Detectors for UAV Imagery

Abdüsselam Kesikoğlu ^{*1}, Ahmet Emin Karkınlı ²

¹ Tomarza MA Vocational School, Kayseri University, Kayseri 38280, Türkiye, e-mail: kesikoglu@kayseri.edu.tr

² Department of Geomatic Engineering, Niğde Ömer Halisdemir University, Niğde, Türkiye

* Correspondence: kesikoglu@kayseri.edu.tr

Article History:

Received: 09.08.2025

Revised 1: 07.10.2025

Revised 2: 14.10.2025

Published: 23.10.2025

Keywords:

Unmanned Aerial

Vehicle;

Object Detection;

Corruption Robustness;

Confidence Calibration;

Small Object Detection.



This article is an open-access article distributed under the terms and conditions of the Creative Commons Attribution 4.0 International License ([CC BY](https://creativecommons.org/licenses/by/4.0/)).

Publisher's Note:

This Journal remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Abstract

Detectors for unmanned aerial vehicles are usually selected on clean-image average precision, yet an airborne camera delivers blurred, noisy, hazy and compressed frames. Four detectors, three lightweight convolutional designs and one real-time transformer, were trained under a matched data exposure and epoch budget on the UAVDT vehicle benchmark and evaluated on temporally subsampled frames of the official test sequences under a COCO-style protocol: clean, six controlled corruptions at five severities, and across the weather, altitude and camera-view attributes of the benchmark, with all partitions formed from whole video sequences and all uncertainty from a sequence-level cluster bootstrap. The transformer reference reached 16.8 per cent average precision against 11.1 to 12.7 per cent for the lightweight detectors, among which no significant difference was detected. The detectors retained 65.3 to 71.4 per cent of their clean accuracy across the corruption benchmark, so relative robustness separated them far less than absolute accuracy did, while small objects lost about 40.6 per cent against about 16.7 per cent for large objects. Contrary to the familiar overconfidence of image classifiers, the three lightweight detectors were underconfident and the gap widened with severity, while the transformer reference was close to calibrated on clean images. A calibrator fitted only on clean held-out sequences reduced the detection expected calibration error by 67 per cent on clean images and by 42 per cent across the corruption benchmark, without changing average precision. All four detectors are non-dominated once accuracy, robustness, calibration, latency and size are considered jointly.

Copyright © 2025 by the author(s).

Contribution to the field statement:

This study evaluates four real-time object detectors for UAV imagery by jointly examining detection accuracy, image degradation, confidence reliability, and computational cost. It shows that small objects are more vulnerable to corruption and that calibration fitted on clean images can reduce confidence error under degraded conditions without changing detection accuracy.

Highlights:

- Four detectors were tested under six image corruptions at five severity levels.
- Small objects lost about 41% of accuracy, compared with 17% for large objects.
- Gaussian noise and motion blur caused the greatest accuracy losses.
- Clean-image calibration reduced confidence error under corruption by 42%.

Cite this article?

Kesikoğlu, A., & Karkınli, A. E. (2025). Comparative evaluation of corruption robustness and confidence calibration in real-time object detectors for UAV imagery. *Advanced UAV*, 5(1), 1–18.

1. Introduction

Unmanned aerial vehicles (UAVs) have become a routine sensing platform for traffic monitoring, infrastructure inspection, search and rescue and precision agriculture, and deep learning is now the dominant way of turning the imagery they collect into decisions [1]. In most of these applications the perception task is object detection: locating and classifying every object of interest in each frame, on board and in real time. An object detector, in the sense used throughout this paper, is a model that returns for each frame a set of bounding boxes,



a category label for each box, and a confidence score that is intended to express how likely that prediction is to be correct.

Detection from a low-altitude airborne camera is markedly harder than detection from a ground-level camera [2]. Objects are small: in the benchmark used here 61.8 per cent of the annotated vehicles occupy fewer than 32×32 pixels, and the median object covers about 798 px in area. Scenes are dense and the object distribution is severely imbalanced, with 94.6 per cent of the annotations belonging to a single category. The camera moves continuously, the viewing geometry changes within a flight, and objects are frequently truncated or occluded. Purpose-built architectures for drone-captured scenes [3] and dedicated benchmarks [4], [5] exist precisely because general-purpose detectors underperform in this regime, and small object detection remains an open problem in its own right [6], [7].

The airborne setting also constrains computation. On-board processors are limited in power, memory and thermal budget, so the detectors that are practical are those in the lightweight, real-time family: single-stage designs that predict boxes and categories in one pass [8], [9], anchor-free dense prediction that removes anchor hyper-parameters [10], focal weighting to handle the foreground-background imbalance [11] and multi-scale feature pyramids for objects of very different size [12]. The lineage has been refined repeatedly [13], [14], most recently by removing non-maximum suppression from the inference path altogether [15] and by revising the aggregation and head design [16]. In parallel, set-prediction transformers [17], [18] have been made fast enough for real-time use [19], which raises the question of whether the additional capacity pays for itself on an airframe.

Accuracy on clean images, however, is not the quantity that determines whether an airborne detector is fit for its mission. Frames captured in flight are blurred by airframe vibration and forward motion, defocused as altitude changes, veiled by haze, noisy at low illumination and degraded by lossy on-board or downlink compression. Benchmarking networks against graded common corruptions has become the standard way of measuring this exposure [20], and object detectors have repeatedly been shown to lose a large fraction of their accuracy under them [21], [22], with the loss concentrated in the degradations that destroy high-frequency texture [23]–[25]. Corruption benchmarks now exist for aerial imagery [26], for air-to-air detection [27] and for UAV tracking [28], and the evidence is that robustness measured on synthetic corruptions is informative but not a perfect substitute for real degradation [29].

A second quantity matters just as much and is measured far less often. A detection score is only useful downstream if it behaves like a probability: a system that fuses detections over time, hands control to an operator when evidence is weak, or thresholds on risk needs the confidence to mean what it says [30]. Modern neural networks are known to be poorly calibrated [31], and calibration for detectors is harder than for classifiers because correctness depends on localisation as well as on the label. Detection-specific calibration measures have therefore been proposed [32], building on binned estimators of calibration error developed for classifiers [33], extended to account for localisation quality [34] and to joint classification-localisation calibration [35], and given density-based definitions [36]; at the same time the pitfalls of these measures have been documented in detail [37], [38]. Simple post-hoc maps remain the cheapest remedy [39], [40].

These three literatures rarely meet. Robustness studies report accuracy under corruption but seldom ask whether the surviving detections are trustworthy; calibration studies work almost exclusively on clean benchmarks; and UAV detection studies optimise clean accuracy under a computational budget. As a result there is little evidence on what happens to a lightweight airborne detector when degradation, object scale, confidence reliability and computational cost are examined together on the same data, with the natural acquisition conditions of the flight held separate from the degradations that are injected under control, and with uncertainty estimated in a way that respects the fact that video frames are not independent samples.

This paper addresses that gap. Its contributions are: (i) a reproducible evaluation protocol for UAV detection that splits at the level of whole video sequences, so that no near-duplicate frame is shared between training, calibration and test, and that keeps the observational condition attributes of the benchmark strictly separate from six physically motivated corruptions applied at five severities; (ii) a joint analysis of accuracy, corruption robustness, object-scale sensitivity and natural flight conditions for four detectors trained under a matched data exposure and epoch budget, with all uncertainty from a sequence-level cluster bootstrap and all comparisons corrected for multiplicity; (iii) a confidence-calibration study across the whole corruption grid in which the calibrator is selected out-of-fold on held-out sequences and never on test data, reporting two calibration measures that disagree; and (iv) a multi-objective selection analysis showing that the detector with the highest clean accuracy is not dominant once robustness, calibration, latency and model size are considered.

2. Method

Figure 1 summarises the study. Data are split at the level of whole video sequences; four detectors are trained with the same training data, input resolution and epoch count; each trained detector is evaluated on clean test frames, on the same frames after controlled corruption, and on the natural condition subsets of the benchmark; confidence calibrators are fitted on a held-out partition and applied unchanged to every test condition; and all uncertainty and detector comparisons are computed at the sequence level. A global random seed controls the sequence-level partition, model initialisation and bootstrap resampling. Every corrupted frame is a deterministic



function of the source frame, corruption, severity and frame index, and the same corruption realisation is used for all detectors. All detectors use one pinned release of the same open-source framework. Tables and figures are generated programmatically from common result files.

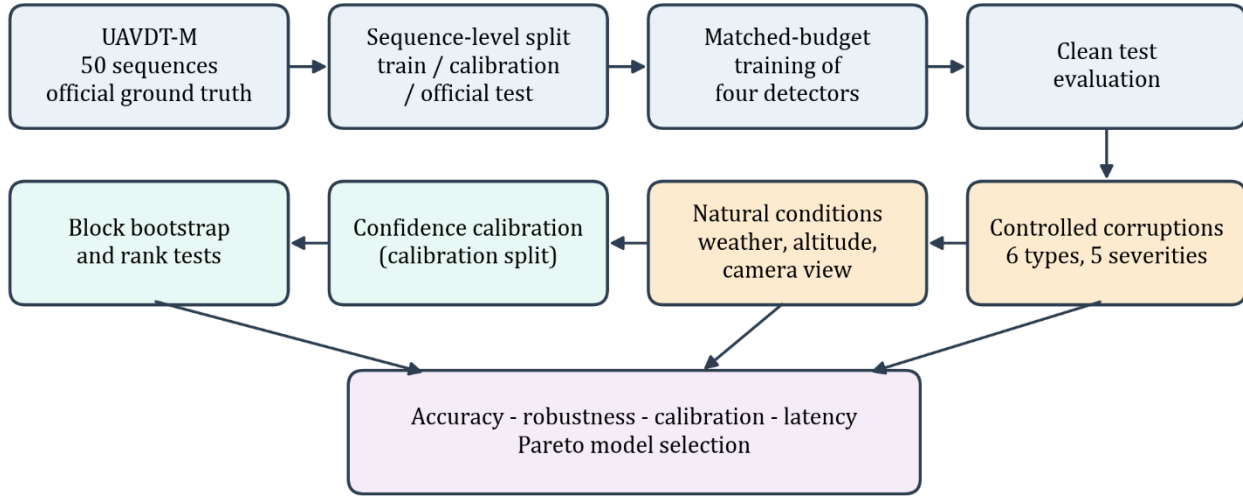


Figure 1. Study workflow. The data are split at the level of whole video sequences, the four detectors share the same training data, input resolution and epoch count, and every detector is evaluated on the identical clean and corrupted frame set before the calibration and statistical analyses.

2.1. Dataset and experimental design

All experiments use the detection subset of the UAVDT benchmark [4], which contains 50 annotated video sequences recorded from a low-altitude UAV over urban roads, together with 40,735 frames and 798,795 vehicle annotations in three categories: car (755,688 instances), truck (25,086) and bus (18,021). The official detection ground truth, the official ignore-region annotations and the official sequence-level attribute files are used without modification. Object density averages 19.6 annotations per frame over the whole benchmark, with a maximum of 107. Following the COCO convention [41], objects smaller than 32×32 pixels are small and objects larger than 96×96 pixels are large; on this basis 61.8 per cent of the annotations are small, 36.4 per cent medium and 1.7 per cent large. Figure 2 shows the category, size, partition and condition composition of the benchmark. No image is resized before annotation analysis, so the scale statistics refer to native pixels.

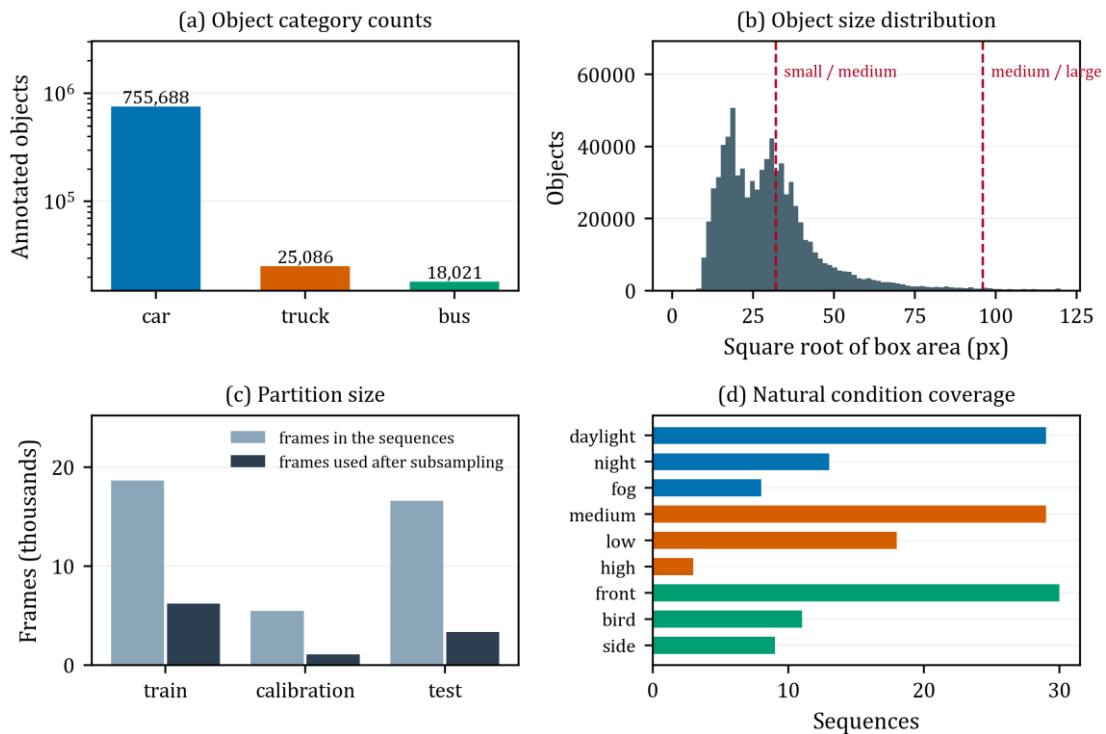


Figure 2. Composition of the UAVDT detection benchmark: (a) object category counts on a logarithmic scale, (b) object size distribution with the small, medium and large boundaries, (c) partition sizes before and after temporal subsampling, and (d) the number of sequences carrying each natural condition label.



The benchmark's own partition of 30 training and 20 test sequences is preserved. Because evaluation uses temporally subsampled frames rather than every frame, the values reported here are not official benchmark scores for UAVDT: they are obtained on subsampled frames of the official test sequences under a COCO-style protocol. From the training sequences, 6 whole sequences are held out as a joint validation and calibration partition; the selection is stratified by the weather attribute and fixed by the global random seed. Sequence-level splitting prevents near-duplicate frames from appearing across training, calibration and test partitions because consecutive frames from the same flight are highly similar. The official test sequences are used only for the final evaluation and never for training, model selection, threshold setting or calibrator fitting. To keep a benchmark of 4 detectors and 31 test conditions tractable, every 3rd frame of each training sequence and every 5th frame of each evaluation sequence is used, which yields the partition sizes in Table 1. The identical frame set is used for every detector and every condition, so subsampling cannot favour any comparison.

Table 1. Composition of the three partitions. Frames used are the temporally subsampled frames on which training and evaluation are actually performed.

Partition	Sequences	Frames in the sequences	Frames used	Objects in the sequences	Objects per frame in the sequences	Small objects (%)
Training	24	18,665	6,230	342,663	18.4	52.2
Validation and calibration	6	5,478	1,098	80,248	14.7	41.6
Test sequences (official split)	20	16,592	3,325	375,884	22.7	74.9

Each sequence carries three groups of condition attributes. Weather (daylight, night, fog) and flight altitude (low, medium, high) are exclusive: every sequence carries exactly one label from each group. Camera view (front, side, bird) is not exclusive; 14 of the 50 sequences carry two view labels, so the view subsets overlap and are reported as overlapping subsets rather than as a partition. Ignore regions are exported as crowd annotations in the evaluation format, so a detection inside such a region is counted neither as a true nor as a false positive. This is an approximation of the original ignore-region rule implemented inside the standard evaluation code rather than a reimplement of the original evaluator, and the two were not compared directly. The data are released by their authors for research use; the imagery contains public road scenes and no personally identifying information was collected by the present authors.

2.2. Detector training and corruption protocol

Four detectors are compared (Table 2). Three are contemporary lightweight CNN-based detectors from the broader YOLO family: an anchor-free one-stage detector released by Ultralytics, an independently developed end-to-end detector that removes non-maximum suppression from inference [15], and an Ultralytics detector with a revised aggregation and head design. No formal architecture paper accompanies the latter release, so it is identified by the framework version below and described through a secondary overview [16]. The fourth is a real-time detection transformer [19] that is an order of magnitude larger and is included as a higher-capacity reference rather than a like-for-like competitor. All four models and their pretrained general-purpose detection weights were publicly available when the experiments were conducted. They were trained within one pinned framework release, sharing the dataset interface and evaluation pipeline while retaining architecture-specific losses and framework-recommended optimisation and augmentation settings.

Table 2. The four detectors compared in this study.

Detector	Architecture family	First public release	Parameters (M)	GFLOPs at 640 px (2 FLOPs per MAC)	Weight file (MB)
YOLOv8-N	Anchor-free one-stage convolutional detector	2023	3.01	8.2	6.2
YOLOv10-N	NMS-free end-to-end convolutional detector	2024	2.71	8.4	5.7
YOLO11-N	One-stage convolutional detector, revised aggregation and head	2024	2.59	6.4	5.5
RT-DETR-L	Real-time detection transformer	2023	32.81	109.9	66.2

Data exposure and epoch count are matched; computational budget is not. The detectors share the same training sequences, the same held-out sequences, an input image resolution of 640 px, 20 epochs, mixed-precision optimisation, no early stopping, and the same model-selection criterion, namely the best fitness on the held-out sequences. The per-step batch size is 16 for the lightweight detectors and 4 for the transformer reference because

of device memory, but gradient accumulation targets the same nominal batch of 64 for every detector. The epoch budget was fixed by a one-epoch pilot on the cheapest and the most expensive detector before any full run started. Each architecture retains the optimisation and augmentation settings recommended by the framework; the comparison therefore concerns the resulting detector configurations rather than an isolated architectural ablation. All four detectors come from Ultralytics 8.3.161 and are initialised from the COCO checkpoints of that project's v8.3.0 weight release; the parameter and computation counts in Table 2 were measured on the trained three-category checkpoints rather than taken from the published model cards. Training, inference and timing all used one NVIDIA GeForce RTX 4070 Laptop GPU with 8 GiB of memory, under Windows 10 with Python 3.14.5 and PyTorch 2.12 built against CUDA 12.6.

Robustness is measured by training only on clean data and testing on clean and progressively corrupted frames, so what is measured is behaviour under distribution shift rather than adaptation to it. Six corruptions are applied at five increasing severities (Table 3, Figure 3). They are operationally motivated proxies rather than physical sensor or atmospheric models, each standing for one failure mode: motion blur for airframe vibration and along-track motion during exposure, defocus blur for focus loss at changing altitude, fog for atmospheric haze, Gaussian noise for sensor read-out noise at low illumination, JPEG compression for lossy on-board or downlink coding, and resolution loss for a bandwidth-limited or altitude-limited effective ground sampling distance. Signal-domain corruptions reuse the severity constants of the established common-corruption benchmark [20], so the parameter values match that literature, although the perceptual and task-level effect of a given value depends on image resolution and object size and is therefore not identical. The spatially defined corruptions are specified in absolute pixels, because the physical magnitude of camera motion or defocus does not scale with image width; transferring radii designed for small classification crops to full-resolution aerial frames would erase every small vehicle at the mildest severity. Their ranges were fixed by a visual inspection carried out before any detector was evaluated. No corruption alters object geometry, so the ground-truth boxes remain valid unchanged, and each corrupted frame is a deterministic function of the frame, the corruption, the severity and the frame index, so every detector sees bit-identical inputs.



Figure 3. The six corruptions at all five severities on one test frame, with the clean crop for reference. Rows, from top to bottom: motion blur, defocus blur, fog, Gaussian noise, JPEG compression and resolution loss. All detectors are evaluated on bit-identical corrupted frames.



Table 3. Corruption types, their operational counterpart, the operation applied and the parameter value at severity 1 to 5.

Corruption	Operational counterpart	Operation	1	2	3	4	5
Motion blur	Airframe vibration and along-track motion during exposure	Convolution with a linear kernel of length L px at a fixed per-frame direction	5	9	13	17	21
Defocus blur	Loss of focus at changing flight altitude	Convolution with a disc point-spread function of radius r px	1	2	3	4	5
Fog	Haze and reduced atmospheric visibility	Plasma field with intensity C and decay d ; contrast renormalisation	1.5/2.0	2.0/2.5	2.5/3.0	3.0/3.5	3.5/4.0
Gaussian noise	Sensor and read-out noise at low illumination	Additive zero-mean Gaussian noise of standard deviation σ on $[0,1]$ intensities	0.08	0.12	0.18	0.26	0.38
JPEG compression	Lossy on-board or downlink image compression	JPEG encode-decode cycle at quality factor q	25	18	15	10	7
Resolution loss	Reduced ground sampling distance, bandwidth-limited downlink	Area downsampling by factor S , then bilinear upsampling to the original size	0.75	0.6	0.5	0.4	0.3

Post-hoc confidence calibration is fitted on the held-out calibration sequences under clean conditions only and applied unchanged to every test condition, which is the operationally realistic case because a deployed detector cannot know in advance which degradation it will meet. Three candidate families are considered: a logistic map of the score logit [39], temperature scaling of the same logit [31], and isotonic regression [40]. Calibrators are fitted per category when a category has at least 50 calibration detections and on the pooled detections otherwise. A detection is labelled correct by a greedy match: within an image, detections are taken in decreasing confidence order and each is matched to the still unmatched ground-truth box of the same category with the highest overlap, and is counted correct when that overlap reaches 0.50; a detection whose area lies more than half inside an ignore region is discarded before matching. Matching is applied after non-maximum suppression and after the per-image detection cap. The candidate with the lowest out-of-fold detection expected calibration error is selected by three-fold cross-validation over whole calibration sequences, with ties broken in favour of the simpler family: scoring candidates on the data they were fitted to would select isotonic regression unconditionally, because it drives its own binned calibration error to zero by construction. The slope of the logistic map is parameterised as an exponential so that the fitted map is strictly increasing; a calibrator that inverted the score order would change the detection ranking, and an unconstrained fit produced exactly that on one sparse category. The selected candidate is refitted on the whole calibration partition. No test detection enters the fitting or the selection.

2.3. Evaluation metrics and statistical analysis

A prediction is matched to a ground-truth box of the same category when their intersection-over-union reaches a threshold, Equation (1), where B_p and B_g are the predicted and ground-truth regions and $\mu(\cdot)$ denotes region area in pixels.

$$\text{IoU}(B_p, B_g) = \frac{\mu(B_p \cap B_g)}{\mu(B_p \cup B_g)} \quad (1)$$

Precision and recall follow from the counts of true positives, false positives and false negatives, Equation (2).

$$\text{Precision} = \frac{TP}{TP + FP}, \quad \text{Recall} = \frac{TP}{TP + FN} \quad (2)$$

Average precision at a fixed overlap threshold is the mean of the interpolated precision over the 101-point recall grid $R = \{0, 0.01, \dots, 1\}$, Equation (3), where $N_R = 101$ is the number of recall levels, p_τ is the interpolated precision at overlap threshold tau, π_τ is the corresponding raw precision, and the interpolation takes the largest precision attained at any recall of at least r . The primary accuracy measure is its mean over the ten overlap thresholds from 0.50 to 0.95, Equation (4) [41], [42]. Average precision is also reported separately for small, medium and large objects, for each category, and at the single thresholds 0.50 and 0.75. At most 100 detections per frame are retained, the usual setting for this metric, which covers all but a handful of frames; the densest frame of the benchmark holds 107 objects, so recall is capped in those rare frames.

$$AP_\tau = \frac{1}{N_R} \sum_{r \in R} p_\tau(r), \quad p_\tau(r) = \max_{u \geq r} \pi_\tau(u) \quad (3)$$

$$AP = \frac{1}{10} \sum_{k=0}^9 AP_{0.50+0.05k} \quad (4)$$

Degradation is summarised in two ways. The relative AP loss of a detector under a corruption at a severity, Equation (5), compares accuracy to that detector's own clean accuracy, so it is a within-model quantity and is always reported alongside the raw values; it is signed, and the few mild conditions that did not reduce accuracy appear as small negative losses. In Equations (5) and (6), A denotes average precision, m indexes the detector, c



the corruption, and s the severity. $A_{m,0}$ is the clean average precision of detector m , while C and S are the sets of corruptions and severities; N_C and N_S are their respective numbers of elements. The normalised robustness ratio, Equation (6), averages the retained fraction over all corruptions and severities. It is a summary measure defined for this study and is not an established named metric; because it is normalised by each detector's own clean accuracy, a detector with a low clean score can obtain a high ratio, so it is always read together with the absolute corrupted accuracy.

$$L_{m,c,s} = 100 \times \frac{A_{m,0} - A_{m,c,s}}{A_{m,0}} \quad (5)$$

$$R_m = \frac{1}{N_C N_S} \sum_{c \in C} \sum_{s \in S} \frac{A_{m,c,s}}{A_{m,0}} \quad (6)$$

Calibration is assessed on the detections whose confidence reaches a fixed floor of 0.05, binned into 15 equal-width confidence intervals. The same floor is used for every detector rather than a model-specific operating point, which is a deliberate simplification and is revisited among the limitations. The detection expected calibration error, Equation (7), is the population-weighted discrepancy between the mean confidence and the precision of each bin, in which K is the number of bins, n_k the number of detections in bin k , N their total, C_k the mean confidence of bin k , and P_k its precision [32]. The function abs in Equations (7) and (8) denotes the absolute value. The measure conditions on confidence alone, and richer formulations that also condition on box position and size exist. Because a detection is correct only if it is also well localised, a second, overlap-weighted measure is reported in which the target of a bin is its precision discounted by the mean overlap achieved by its true positives, U_k , in Equation (8). This follows the idea of localisation-aware calibration [34] but pools all categories into one binning instead of averaging category-wise errors, so it is not identical to the published measure of that name; it is called the overlap-weighted calibration error, abbreviated OWCE, throughout this paper. The Brier score of the matched binary decisions, Equation (9) [43], is also reported together with reliability diagrams. In Equation (9), s_i denotes the confidence of detection i , and y_i in $\{0,1\}$ is its matched correctness label. For a bin containing no true positives, U_k is defined as zero. Reporting more than one calibration measure is deliberate: single calibration numbers for detectors are known to be fragile [37]. Because the confidence floor is fixed rather than a model-specific operating point, the calibration results of this study are a fixed-threshold descriptive analysis rather than an operating-point evaluation.

$$\text{D-ECE} = \sum_{k=1}^K \frac{n_k}{N} \text{abs}(P_k - C_k) \quad (7)$$

$$\text{OWCE} = \sum_{k=1}^K \frac{n_k}{N} \text{abs}(C_k - P_k U_k) \quad (8)$$

$$\text{BS} = \frac{1}{N} \sum_{i=1}^N (s_i - y_i)^2 \quad (9)$$

Computational cost is measured at batch size one, the relevant setting for an on-board processor, on the same device, at the same input resolution and in the same numerical precision for every detector. The first 50 inferences are discarded as warm-up and the following 300 are timed. The reported latency is end to end, including resizing, device transfer and the postprocessing that turns raw outputs into final boxes, because that is what a deployed system experiences and because it uses an identical code path for all four detectors. Mean, 95th percentile, frame rate and peak device memory are reported together with parameter count, floating-point operation count (counting two operations per multiply-accumulate, as reported by the profiler of the training framework) and weight file size.

The unit of statistical analysis is the video sequence, not the frame. Confidence intervals come from a cluster bootstrap that resamples whole test sequences with replacement, 2,000 times [44], [45], reported as 95 per cent percentile intervals. The same resampling scheme is used for every detector and every condition, so differences between detectors are formed replicate by replicate and are properly paired. Detectors are compared with a Friedman test over sequences [46] with the coefficient of concordance as effect size, followed, when the omnibus test rejects, by Wilcoxon signed-rank post-hoc tests [47] with Holm correction [48] and rank-biserial effect sizes, which is the standard non-parametric protocol for comparing several methods over many units [49]. The monotonicity of the severity response is assessed by the Kendall rank correlation between severity and accuracy. Corruption-severity cells are never treated as independent observations. For the corrupted detector comparison, each sequence contributes its mean AP over the 30 corruption-severity conditions. The primary metric, the exclusion rules and the statistical tests were specified before the final comparisons were made; the per-corruption rank-inversion analysis was added afterwards and is reported as such.



3. Results

3.1. Clean-image accuracy and computational efficiency

Table 4 reports accuracy on 3,325 temporally subsampled frames from the 20 official test sequences under clean conditions, with 95 per cent sequence-level cluster bootstrap intervals. The transformer reference reaches the highest average precision, 16.8 per cent [13.1, 25.3], followed by YOLO11-N (12.7 per cent), YOLOv8-N (12.3 per cent) and YOLOv10-N (11.1 per cent). At the looser overlap threshold of 0.50 the same ordering holds, from 28.9 per cent down to 19.7 per cent.

Table 4. Clean-image accuracy on temporally subsampled frames from the official test sequences, in per cent, with 95 per cent sequence-level cluster bootstrap intervals for the three headline measures.

Detector	AP	AP50	AP75	AP small	AP medium	AP large	car	truck	bus
YOLOv8-N	12.3 [9.3, 19.2]	22.7 [18.8, 31.0]	11.6	8.1 [5.9, 10.7]	19.7	32.7	31.4	0.5	5.0
YOLOv10-N	11.1 [7.3, 18.8]	19.7 [13.7, 29.9]	11.1	6.7 [4.1, 10.6]	18.1	33.3	25.5	1.3	6.5
YOLO11-N	12.7 [9.3, 19.8]	23.4 [18.8, 31.8]	12.3	7.4 [5.4, 10.1]	22.3	27.3	29.6	5.4	3.2
RT-DETR-L	16.8 [13.1, 25.3]	28.9 [25.7, 38.1]	17.2	13.6 [9.6, 21.0]	23.3	37.8	40.9	0.9	8.7

The relative advantage of the transformer reference over the three lightweight detectors is largest for small objects. Average precision restricted to objects below 32×32 pixels is 13.6 per cent for the transformer reference against 8.1 to 7.4 per cent for the lightweight detectors, a relative advantage of 67.6 per cent over the best lightweight detector, whereas for large objects the advantage falls to 13.5 per cent. All four detectors are far weaker on the two rare categories than on cars: average precision on cars ranges from 25.5 to 40.9 per cent, while truck and bus average precision stays below 8.7 per cent for every detector. Because average precision is averaged with equal weight over the three categories, these two weak categories carry the same weight in the overall value as cars do, so the overall figure is held down by them; the 94.6 per cent share of cars in the annotation set is a plausible reason for the weakness of the rare categories but was not tested directly here.

Table 5 reports the measured cost of one forward inference at batch size one. The three lightweight detectors run between 11.2 and 14.3 ms per frame (90 to 70 frames per second) and occupy at most 91 MiB of device memory, while the transformer reference needs 40.0 ms (25 frames per second) and 524 MiB. Its parameter count is 12.7 times and its floating-point operation count 17.1 times that of the smallest lightweight detector. YOLOv8-N is the fastest detector and YOLO11-N the smallest; the two are not the same model.

Table 5. Measured inference cost at batch size one on the same device, at 640 px input resolution and in single precision.

Detector	Latency, mean (ms)	Latency, 95th percentile (ms)	Frames per second	Peak GPU memory (MiB)
YOLOv8-N	11.16	11.52	89.6	60
YOLOv10-N	12.36	13.01	80.9	82
YOLO11-N	14.33	18.10	69.8	91
RT-DETR-L	39.98	41.55	25.0	524

A Friedman test over the 20 test sequences rejects the hypothesis that the four detectors are interchangeable under clean conditions (chi-square = 21.48, $p < 0.001$, Kendall coefficient of concordance = 0.36). In the Wilcoxon signed-rank post-hoc tests with Holm correction reported in Table 6, the transformer reference is ahead of every lightweight detector (adjusted $p = 0.011$ in the least favourable comparison, rank-biserial correlation between 0.73 and 0.94), while no significant difference between any pair of lightweight detectors was detected under this protocol (adjusted p at least 0.089). The paired sequence bootstrap agrees: the difference in pooled average precision between RT-DETR-L and the best lightweight detector is 4.10 percentage points with an interval that excludes zero, whereas every interval between two lightweight detectors contains zero.

**Table 6.** Pairwise detector comparisons over the 20 test sequences: Wilcoxon signed-rank tests with Holm correction and rank-biserial effect sizes

Comparison	Setting	Median paired difference in AP (%)	Wilcoxon p	Holm-adjusted p	Rank-biserial effect size
YOLOv8-N vs YOLOv10-N	Clean	+1.80	0.1893	0.3787	+0.343
YOLOv8-N vs YOLO11-N	Clean	-0.17	0.2024	0.3787	-0.333
YOLOv8-N vs RT-DETR-L	Clean	-7.74	0.0002	0.0008	-0.876
YOLOv10-N vs YOLO11-N	Clean	-2.34	0.0296	0.0887	-0.552
YOLOv10-N vs RT-DETR-L	Clean	-8.32	< 0.0001	0.0002	-0.943
YOLO11-N vs RT-DETR-L	Clean	-5.97	0.0027	0.0108	-0.733
YOLOv8-N vs YOLOv10-N	Corrupted	+0.96	0.1140	0.1392	+0.410
YOLOv8-N vs YOLO11-N	Corrupted	-0.72	0.0696	0.1392	-0.467
YOLOv8-N vs RT-DETR-L	Corrupted	-5.41	0.0003	0.0016	-0.848
YOLOv10-N vs YOLO11-N	Corrupted	-2.27	0.0240	0.0719	-0.571
YOLOv10-N vs RT-DETR-L	Corrupted	-6.19	< 0.0001	< 0.0001	-0.990
YOLO11-N vs RT-DETR-L	Corrupted	-4.24	0.0094	0.0377	-0.648

3.2. Robustness to natural conditions and progressive corruptions

Figure 4 shows the relative AP loss for every detector, corruption and severity, and Figure 5 shows the same data as severity-response curves with bootstrap bands. Accuracy falls monotonically with severity in almost every case: the Kendall rank correlation between severity and average precision is -0.80 or lower in all 24 detector-corruption combinations, and equals the minimum value of -1.00 in 21 of them.

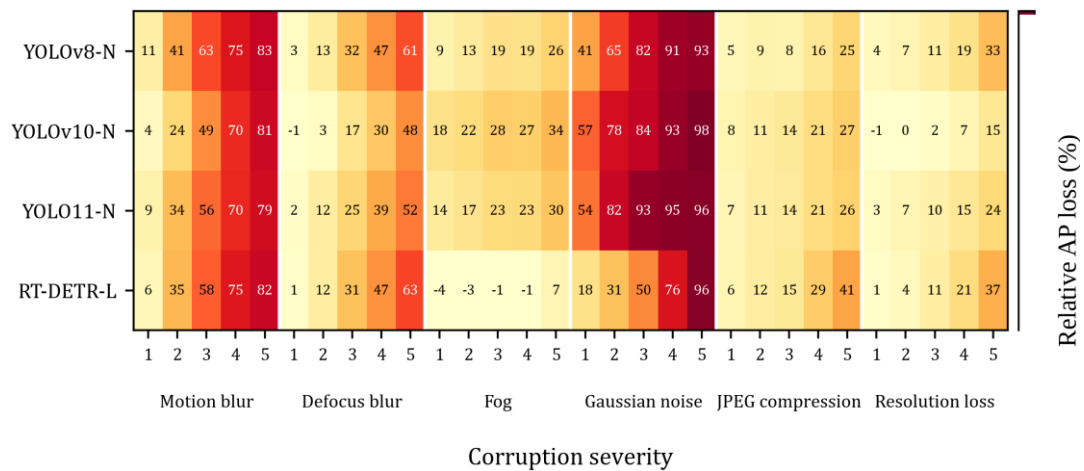


Figure 4. Relative AP loss, in per cent of each detector's own clean accuracy, for every corruption and severity. Negative values indicate the few cells in which a mild corruption did not reduce accuracy.

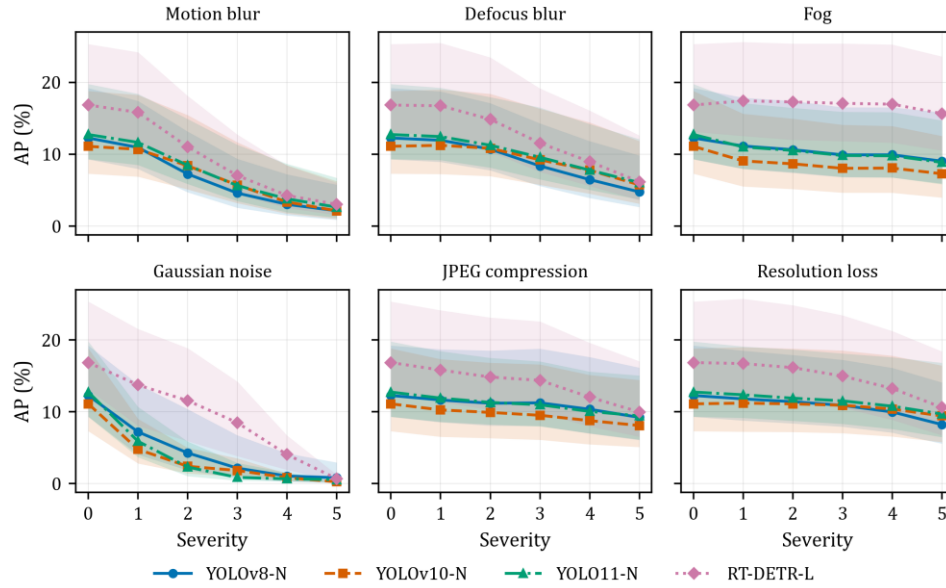


Figure 5. Severity response of the four detectors for each corruption. Severity zero is the clean condition. Bands are 95 per cent sequence-level cluster bootstrap intervals.

The corruptions are not equally damaging. Averaged over detectors and severities, Gaussian noise costs 73.7 per cent of clean average precision and motion blur costs 50.1 per cent, while resolution loss costs only 11.5 per cent (Table 7). At the highest severity the two sensor-domain and motion-domain corruptions remove most of the usable signal: Gaussian noise at severity five costs between 93.4 and 97.8 per cent of clean average precision.

Table 7. Mean relative AP loss per corruption, in per cent, and the study-defined normalised robustness ratio with its 95 per cent interval.

Detector	Motion blur	Defocus blur	Fog	Gaussian noise	JPEG compression	Resolution loss	Mean relative AP loss	Robustness ratio
YOLOv8-N	54.4	31.0	17.4	74.7	12.5	14.7	34.1	65.9 [58.4, 74.1]
YOLOv10-N	45.5	19.3	25.9	81.9	16.2	4.5	32.2	67.8 [62.8, 73.9]
YOLO11-N	49.5	26.0	21.5	83.9	15.8	11.7	34.7	65.3 [58.9, 73.4]
RT-DETR-L	51.1	30.8	-0.2	54.3	20.5	14.9	28.6	71.4 [59.2, 80.6]

Averaged over the whole benchmark the detectors retain between 65.3 and 71.4 per cent of their clean average precision, and the bootstrap intervals of the four normalised robustness ratios overlap substantially (Table 7). RT-DETR-L has the highest ratio, 71.4 per cent [59.2, 80.6]. Relative robustness therefore separates the detectors far less than absolute accuracy does. The Friedman test over sequences again rejects interchangeability under corruption (chi-square = 27.42, $p < 0.001$, coefficient of concordance = 0.46), and the post-hoc pattern is the same as on clean data: the transformer reference is ahead of each lightweight detector and the lightweight detectors do not separate from one another.

The ordering of the detectors is stable at the top and unstable below it. Ranking the four detectors by average precision averaged over all corruptions reproduces the clean ranking exactly (Kendall tau = 1.00). Within individual corruptions the agreement is weaker: the rank correlation with the clean ordering is 0.67 for 5 of the six corruptions, and across the 30 corruption-severity cells 15.0 per cent of all pairwise orderings differ from the clean ordering, while the detector with the highest clean average precision is displaced in only 1 of 30 cells. The most visible single reversal is YOLOv10-N, which has the lowest clean average precision of the four detectors but the smallest loss under resolution loss (4.5 per cent against 11.7 to 14.9 per cent for the others) and the largest loss under Gaussian noise among the lightweight detectors.

Figure 6 separates the loss by object scale. Small objects lose 38.0 to 43.0 per cent of their clean average precision under corruption, medium objects 13.2 to 27.1 per cent and large objects 10.1 to 20.3 per cent. The ordering small > medium > large holds for every detector, so the objects that dominate this dataset are also the objects that degradation removes first.

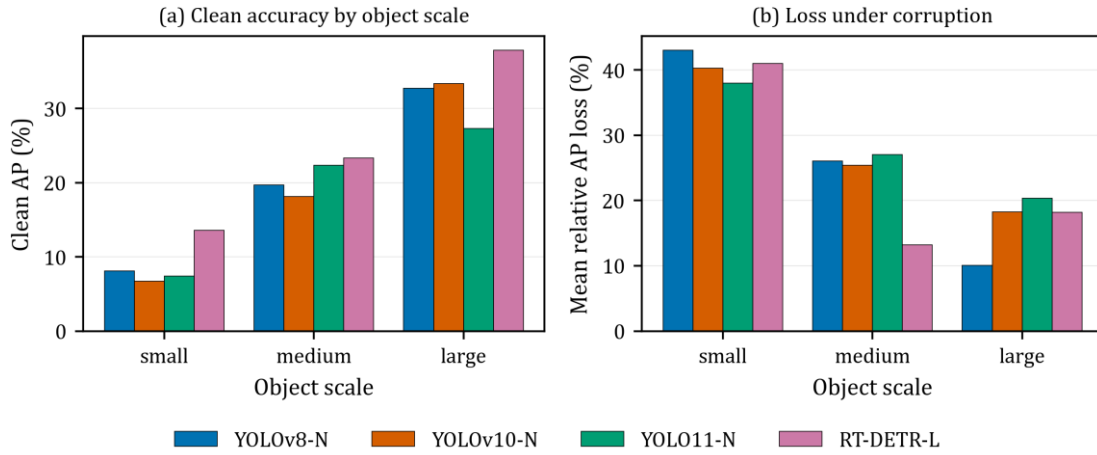


Figure 6. Accuracy by object scale: (a) clean average precision for small, medium and large objects, and (b) the mean relative loss of each scale over the whole corruption benchmark.

Figure 7 reports accuracy under the natural conditions annotated in the benchmark itself. These are observational subsets of unequal and partly very small size, from 2 sequences for high altitude to 12 for daylight, and they are confounded with object scale, density and acquisition site, so they are read descriptively rather than as controlled contrasts. Flight altitude separates performance more strongly than any synthetic corruption at low severity: on clean images the detectors reach 32.0 to 39.9 per cent average precision on low-altitude sequences but only 3.8 to 10.3 per cent on high-altitude sequences, and the relative loss under corruption is larger at high altitude (39.5 to 59.8 per cent) than at low altitude (18.4 to 22.7 per cent). Naturally foggy sequences are the weakest weather condition on clean images (4.6 to 11.9 per cent) and also the condition that loses most under additional corruption. Night sequences are not the hardest weather condition in this benchmark: every detector scores higher at night than in daylight, by 3.9 to 7.3 percentage points.

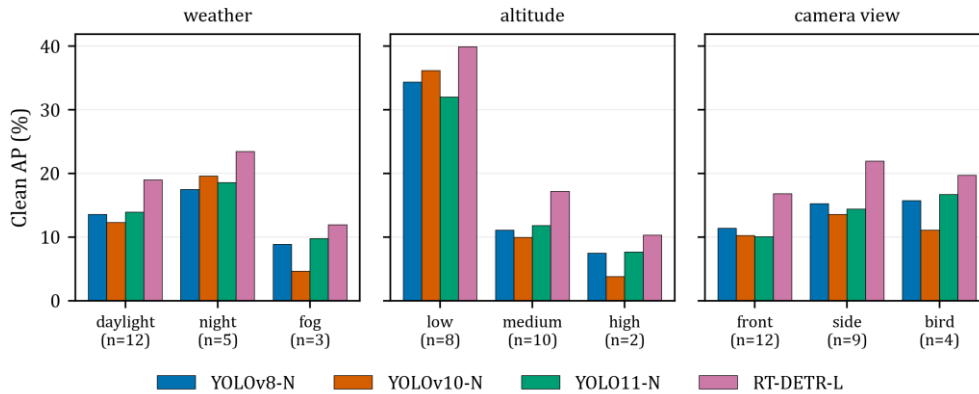


Figure 7. Clean average precision on the natural condition subsets recorded in the benchmark. Weather and altitude are partitions of the test sequences; the camera-view subsets overlap, because a third of the sequences carry two view labels. The number of test sequences behind each bar is given under its label; subsets of two or three sequences are descriptive only.

Figure 8 shows one test frame under clean, moderate and severe conditions with the detections of RT-DETR-L. Under motion blur most vehicles are still found but the boxes loosen; under severe fog the detector both misses vehicles in the haze and produces additional false positives on background structure, which is the behaviour the aggregate numbers describe.

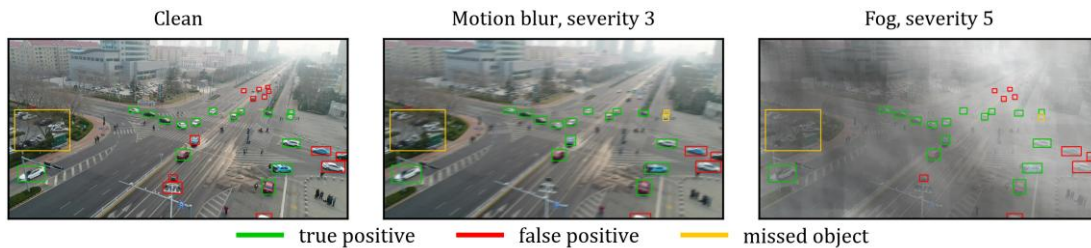


Figure 8. Detections of the transformer reference on one low-altitude daylight test frame under clean, moderately blurred and heavily fogged conditions, at a confidence threshold of 0.25.



3.3. Confidence calibration and multi-objective model selection

Table 8 and Figure 9 report calibration on the test partition for the raw confidence and for the calibrator selected on the calibration sequences alone. On clean test images the detection expected calibration error of the raw confidence ranges from 4.12 per cent for RT-DETR-L to 14.09 per cent for YOLOv8-N. For the three lightweight detectors the direction of the miscalibration is the opposite of the classical result for image classifiers: mean confidence lies below the precision of the retained detections, so they are underconfident rather than overconfident, by 6.2 to 14.1 percentage points on clean images. The transformer reference is the exception: its mean confidence sits 0.8 percentage points above its precision on clean images, which is close agreement in either direction.

Table 8. Calibration on the test partition for the raw confidence and for the selected calibrator, in per cent except for the Brier score.

Detector	Confidence	Condition	D-ECE	Overlap-weighted error	Brier score	Mean confidence	Precision
YOLOv8-N	Raw	Clean	14.09	3.65	0.1847	33.4	47.5
		Mean over corruptions	16.28	7.16	0.1974	29.9	45.0
	Platt calibrated	Clean	1.42	13.16	0.1500	48.9	47.5
		Mean over corruptions	5.44	12.03	0.1626	45.3	45.0
YOLOv10-N	Raw	Clean	9.34	11.67	0.1845	57.1	66.2
		Mean over corruptions	18.52	15.71	0.2349	50.8	69.2
	Platt calibrated	Clean	4.50	17.53	0.1687	68.5	66.2
		Mean over corruptions	9.41	16.83	0.1918	63.1	69.2
YOLO11-N	Raw	Clean	6.35	5.75	0.1690	40.4	46.6
		Mean over corruptions	17.66	15.12	0.2513	36.3	52.9
	Platt calibrated	Clean	2.55	13.81	0.1638	49.2	46.6
		Mean over corruptions	13.56	18.77	0.2333	44.5	52.9
RT-DETR-L	Raw	Clean	4.12	8.17	0.0944	32.1	31.3
		Mean over corruptions	6.37	6.14	0.0949	23.5	24.5
	Temperature calibrated	Clean	2.76	6.87	0.0943	30.6	31.3
		Mean over corruptions	5.91	5.25	0.0956	20.9	24.5

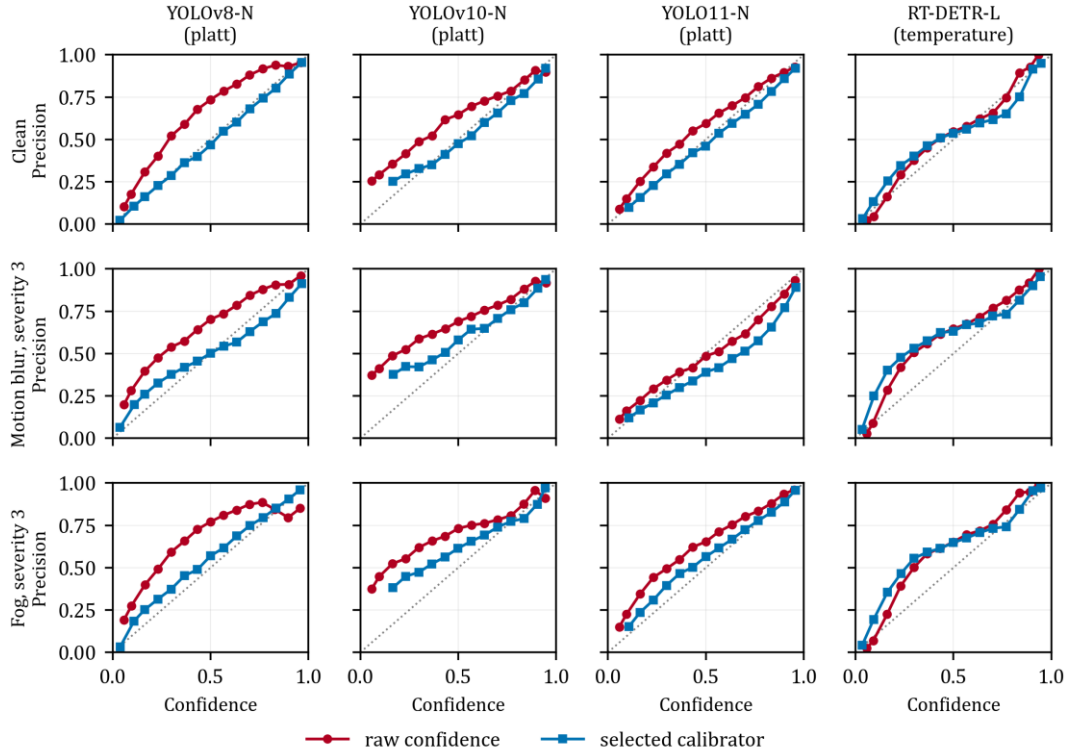


Figure 9. Reliability diagrams for the raw confidence and for the calibrator selected on the calibration sequences, under clean conditions and under two corruptions at severity three. The calibrator selected for each detector is given under its name. Points below the diagonal indicate overconfidence and points above it underconfidence. Bins are of equal width and only bins containing detections are drawn.

Corruption generally increases this miscalibration. Averaged over the whole corruption benchmark the raw detection expected calibration error rises to 6.37 to 18.52 per cent, and for the three lightweight detectors it grows monotonically with severity, from 14.9 per cent at severity one to 20.7 per cent at severity five. The mechanism is visible in the underlying quantities: as severity increases the mean confidence of the retained detections falls while their precision does not fall as fast, so the underconfidence gap widens. The transformer reference behaves differently: its global mean confidence and global precision differ by about 1.0 percentage point in both settings, which does not imply bin-wise calibration, since its detection expected calibration error over the corruption benchmark is still 6.37 per cent.

Post-hoc calibration fitted only on clean calibration sequences transfers to the corrupted test conditions. The selected calibrator reduces the detection expected calibration error on clean test images from 8.48 to 2.81 per cent on average, a reduction of 67 per cent, and over the corruption benchmark from 14.71 to 8.58 per cent, a smaller reduction of 42 per cent. The Brier score of the matched detection decisions improves for the three lightweight detectors, from 0.2279 to 0.1959 over the corruption benchmark, but not for the transformer reference, whose corrupted Brier score changes from 0.0949 to 0.0956, a marginal deterioration. Because both selected calibrator families are strictly increasing and are applied within each category, they cannot change the ordering of detections inside a category, so average precision is mathematically unchanged; the reduction in D-ECE is obtained without changing AP and at negligible computational cost.

The overlap-weighted calibration error does not agree with this conclusion. For the three lightweight detectors it *increases* after calibration, on clean images from 7.03 to 14.83 per cent, while it decreases for the transformer reference, from 8.17 to 6.87 per cent. The two measures target different quantities: the detection expected calibration error asks the confidence to match the precision, whereas the overlap-weighted error asks it to match the precision discounted by the mean overlap achieved by the true positives. Raising an underconfident score towards the precision therefore moves it away from the lower, overlap-discounted target. The direction of the calibration effect in this study is measure-dependent, and no single calibration number should be read as the reliability of a detector.

Figure 10 places the detectors in the accuracy-latency and robustness-calibration planes. Considering jointly, without weighting, clean average precision, the normalised robustness ratio, the uncalibrated detection expected calibration error, latency and parameter count together, all four detectors are non-dominated: RT-DETR-L leads on accuracy, robustness ratio and calibration, YOLOv8-N on latency and YOLO11-N on model size, so no detector is better than another on every criterion. Selecting on clean average precision alone would choose a detector that is 3.6 times slower and 12.7 times larger than the lightest alternative.

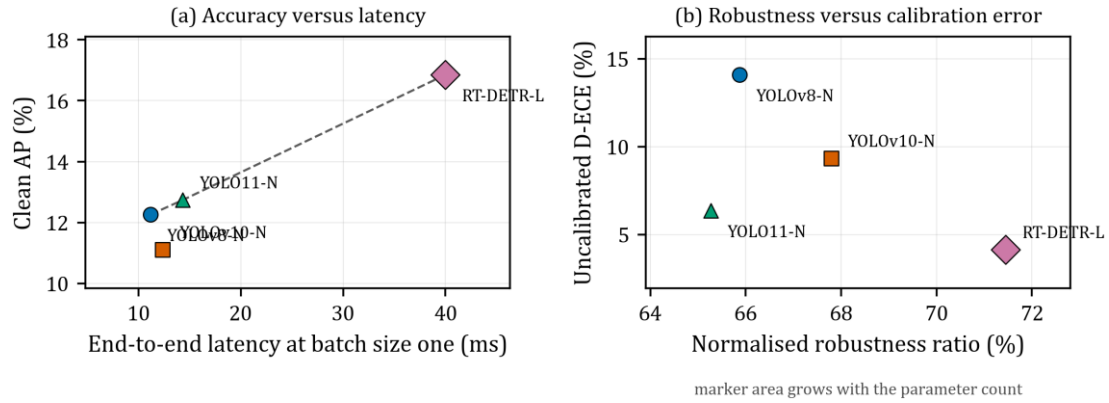


Figure 10. Multi-objective view of the four detectors: (a) clean accuracy against measured latency and (b) the normalised robustness ratio against the uncalibrated detection expected calibration error. The dashed line joins the detectors that are non-dominated with respect to the two measures shown in that panel only; the statement in the text that all four detectors are non-dominated refers to the five criteria considered jointly.

4. Discussion

Four questions motivated this study: how lightweight detectors compare on clean UAV imagery once computation is accounted for, how quickly they lose accuracy as degradation intensifies, whether the natural acquisition conditions recorded in the benchmark shift that behaviour, and whether the confidence these detectors emit can be trusted. The answers are not symmetric. Absolute accuracy separates the detectors clearly and the separation survives corruption, but the *relative* rate at which they lose accuracy separates them very little. Reliability, in contrast, separates them clearly, and in a direction that the classification literature would not predict.

The transformer reference is ahead of every lightweight detector on clean images and in the corruption-averaged comparisons, with large effect sizes and adjusted significance in all six comparisons. That result is expected from its capacity: it carries 12.7 times the parameters and 17.1 times the floating-point operation count of the smallest lightweight detector, and it is included here as a higher-capacity reference rather than as a like-for-like competitor [19]. The three lightweight detectors, contemporary CNN-based members of the broader YOLO family released by Ultralytics or developed independently [15], [16], showed no statistically significant difference from one another on this dataset, either before or after corruption. A failure to reject is not evidence of equivalence, and with 20 test sequences and one training run per architecture the power of these tests is limited; what the data support is that architectural progress measured on general-purpose benchmarks did not produce a detectable advantage under this training budget, framework version and evaluation protocol.

The corruption ranking is consistent with what is known about the failure modes of convolutional detectors. Additive sensor noise and along-track motion blur are by far the most damaging conditions, while lossy compression and reduced effective resolution cost comparatively little. Degradations that destroy high-frequency texture hurt disproportionately, which matches both the classification evidence that networks lean on texture cues [24] and the general finding that blur and noise degrade deep networks far more than they degrade human observers [23]. The magnitudes seen here are larger than those reported on ground-level benchmarks [20], [21], which is what a dataset dominated by objects of a few dozen pixels should produce: a corruption that removes a fixed spatial frequency band removes a larger fraction of the evidence available about a small object than about a large one.

That intuition is confirmed directly by the scale analysis. Under the same corruptions, small objects lose about 40.6 per cent of their clean average precision while large objects lose about 16.7 per cent, and the ordering holds for every detector. The natural-condition analysis points the same way from an observational direction: flight altitude, which was associated with object scale in this dataset, coincides with a three- to five-fold difference in clean accuracy and with how much is lost under corruption, although altitude is also confounded with camera geometry, scene layout and object density. Reports that small object detection remains the hard core of aerial perception [2], [6] are therefore not only about clean accuracy: scale and robustness are closely linked in this dataset.

Two observational results deserve care because they are confounded rather than controlled. Night sequences score higher than daylight sequences for all four detectors. This may reflect the particular content, object-scale distribution and object density of the night sequences of this benchmark rather than any advantage of low illumination; brightness, contrast and object separation were not measured here. The camera-view attribute is also not a partition in the official annotation: a third of the sequences carry two view labels, so the view subsets overlap and cannot be read as a controlled contrast. These are exactly the reasons the natural attributes and the injected corruptions are reported separately throughout this paper and are never pooled into one condition axis.

The calibration findings are a notable part of the study. Modern classifiers are often reported to be overconfident, and a common remedy is to soften their scores [31]. The detectors studied here behave differently,



with one exception: on clean test images, the mean confidence of the three lightweight detectors is 9.8 percentage points below the precision of the retained detections, and corruption widens rather than narrows that gap. The transformer reference shows a global mean confidence-precision difference of about one percentage point, although this does not imply bin-wise calibration. As the image degrades, fewer detections remain above the fixed confidence floor, and the precision of those retained detections falls more slowly than their confidence. A downstream system that treats these scores as probabilities could therefore discard usable detections under degraded conditions. This illustrates the importance of reliable uncertainty estimates for downstream use [30], [34].

Post-hoc calibration provides a low-cost adjustment. A calibrator with one or two parameters per category, fitted on clean held-out sequences and applied without refitting, reduced the detection expected calibration error by 67 per cent on clean data and by 42 per cent across the whole corruption benchmark. It improved the Brier score of the three lightweight detectors, while the corrupted Brier score of the transformer reference changed marginally from 0.0949 to 0.0956. Because the selected families are strictly increasing, average precision is unchanged. Out-of-fold calibrator selection avoids optimistic in-sample estimates: scoring candidates on the data used for fitting selects isotonic regression because it can drive its own binned calibration error to zero, whereas sequence-level out-of-fold scoring showed that the simpler logistic map generalised at least as well [39], [40]. The overlap-weighted error moves in the opposite direction to the classification-style error for three of the four detectors. A confidence that matches precision cannot simultaneously match precision discounted by overlap quality, so the two measures reward different calibrators. These results indicate that calibration improved confidence as a probability of correctness but did not improve its relationship with localisation quality. Post-hoc calibration should therefore be validated against the measure that matches the intended downstream use rather than applied as a default.

For deployment, no single detector dominates all criteria. All four lie on the non-dominated front when accuracy, robustness, calibration, latency and size are considered together. On the evaluated hardware, RT-DETR-L provides the highest accuracy and the highest estimated robustness ratio, provided that its 40 ms latency and 524 MiB peak GPU memory at batch size one are acceptable; the confidence intervals of the robustness ratios overlap. Among the lightweight detectors, release recency does not correspond to consistent superiority under this protocol. The choice can therefore be based on the mission constraint, such as latency, model size or stability under resolution loss. These trade-offs are not captured by clean-image accuracy alone.

Several limitations bound these conclusions. The study uses one benchmark, recorded by one group with a limited set of platforms and scenes, so the numbers characterise UAVDT vehicle detection and not UAV perception in general; no external dataset was used, so cross-dataset generalisation is untested. The corruptions are controlled and physically motivated but synthetic, and a growing body of evidence shows that synthetic corruption robustness does not transfer perfectly to real degraded imagery [26], [29]. The severity parameters of the spatial corruptions are expressed in absolute pixels and were fixed by visual inspection, so they are internally consistent but not identical to any published severity scale. Every detector was trained for the same 20 epochs, chosen from a pilot measurement; the study was not designed as a convergence analysis, and one detector was still improving at the end of its budget. Latency was measured on a single laptop-class device and is a proxy for, not a measurement of, the energy cost on an embedded airframe. Evaluation used every fifth frame of each test sequence, so absolute values are not comparable with papers that evaluate all frames, although all comparisons here use the identical frame set. Each detector was trained once, from a single random seed, so the reported uncertainty covers sampling over test sequences but not training variability; conclusions therefore apply to these trained models rather than to the architectures in general. The six sequences held out for validation were also used to fit the calibrators, so the calibration partition is not independent of model selection even though it is independent of the test data. The ignore regions were mapped to crowd annotations inside the standard evaluation code, which is an approximation of the original protocol rather than a reimplement of it. At most 100 detections per frame were kept, which is the common evaluation setting but is below the 107 objects of the densest frame, so recall is capped in the very densest scenes. Finally, the calibration analysis conditions on detections above a fixed confidence floor that is the same for every detector rather than a model-specific operating point, and both that floor and the definition of a correct detection influence every calibration number reported; the calibration results are therefore a descriptive sensitivity analysis at a fixed threshold, not an operating-point analysis.

5. Conclusion

This study asked whether clean-image accuracy is an adequate basis for choosing an object detector for UAV imagery. Four detectors spanning three lightweight convolutional designs and one real-time transformer were trained with the same training sequences, input resolution and epoch count, each keeping its own recommended training configuration on the official UAVDT training sequences and evaluated on temporally subsampled frames of the official test sequences under a COCO-style protocol: clean, six controlled corruptions at five increasing severities, and the natural weather, altitude and camera-view attributes recorded in the benchmark, with all uncertainty estimated by a sequence-level cluster bootstrap.



The transformer reference achieved 16.8 per cent average precision against 11.1 to 12.7 per cent for the lightweight detectors, a difference that is significant after correction in every comparison, while no significant difference among the three lightweight detectors was detected. Averaged over the corruption benchmark the detectors retained 65.3 to 71.4 per cent of their clean accuracy, with overlapping intervals, so relative robustness distinguished them far less than absolute accuracy did. Additive sensor noise and motion blur removed most of the usable signal at high severity, whereas compression and resolution loss cost comparatively little. Small objects lost about 40.6 per cent of their accuracy against about 16.7 per cent for large objects, and the same pattern appeared observationally as a three- to five-fold accuracy gap between low-altitude and high-altitude sequences.

Confidence proved to be one of the clearest axes of difference. The three lightweight detectors were underconfident rather than overconfident and their gap between confidence and precision widened monotonically with corruption severity, and their global confidence-precision gap widened with corruption severity, while the transformer reference showed a smaller global gap, although its bin-wise D-ECE remained non-zero. A one- or two-parameter calibrator fitted only on clean held-out sequences and applied unchanged to every corrupted condition reduced the detection expected calibration error by 67 per cent on clean images and by 42 per cent across the corruption benchmark, at no cost in average precision, because the selected maps are strictly increasing. The Brier score improved for the three lightweight detectors but not for the transformer reference, and the overlap-weighted calibration error moved the other way for three of the four detectors, which together show that a single calibration number cannot certify a detector.

For practice the recommendation is to select on the axis the mission constrains rather than on a leaderboard. All four detectors are non-dominated once accuracy, robustness, calibration, latency and size are considered jointly. On the laptop-class GPU used here and among the criteria considered, the transformer reference gives the highest accuracy, but its 40 ms per frame and 524 MiB peak GPU memory at batch size one may put it out of reach of a small airframe; power draw, thermal behaviour and embedded hardware were not measured. Among the lightweight detectors the most recently released model did not outperform the others consistently under this protocol, so the decision should rest on latency, memory or the specific degradation the platform expects. Whichever detector is chosen, post-hoc calibration is computationally inexpensive, but it should be applied only after validation against a calibration criterion that matches the intended downstream use.

The main limitations are a single benchmark, synthetic rather than recorded degradations, a modest common epoch count, and latency measured on one laptop-class device. The natural next steps are to repeat the protocol on genuinely degraded aerial imagery, to test whether corruption-aware calibration fitted on a small amount of degraded data closes the residual error that clean-fitted calibration leaves behind, and to measure energy on the airframe rather than inferring cost from latency.

Acknowledgements

The authors thank the creators of the UAVDT benchmark for releasing the imagery, the detection ground truth, the ignore-region annotations and the sequence-level condition attributes for public research use, and the maintainers of the open-source detection framework used to train and evaluate the detectors.

Funding

This research received no specific grant from any funding agency in the public, commercial or not-for-profit sectors.

Conflicts of Interest

The authors report no conflicts of interest.

Data Availability Statement

This study uses the publicly available UAVDT benchmark, which is distributed by its authors for research purposes as an open dataset; the imagery, the detection ground truth, the ignore regions and the sequence-level attribute files were obtained from that public release and were not modified, so all input data are openly accessible. The derived materials and analysis code are available from the corresponding author upon reasonable request.

Institutional Review Board Statement

Not applicable. This study analyses an existing, publicly released aerial imagery benchmark of road vehicles and involves no human participants, no animals and no personally identifying data collected by the authors.

CRedit Author Statement

Both authors contributed to conceptualisation, methodology, investigation, formal analysis, validation, writing of the original draft, and writing through review and editing. Software, data curation and visualisation were carried out jointly. Both authors have read and approved the submitted version of the manuscript.



References

- [1] L. P. Osco et al., "A review on deep learning in UAV remote sensing," *Int. J. Appl. Earth Obs. Geoinf.*, vol. 102, Art. no. 102456, Oct. 2021, doi: 10.1016/j.jag.2021.102456.
- [2] P. Mittal, R. Singh and A. Sharma, "Deep learning-based object detection in low-altitude UAV datasets: A survey," *Image Vis. Comput.*, vol. 104, Art. no. 104046, Dec. 2020, doi: 10.1016/j.imavis.2020.104046.
- [3] X. Zhu, S. Lyu, X. Wang and Q. Zhao, "TPH-YOLOv5: Improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW)*, Montreal, QC, Canada, 2021, pp. 2778–2788, doi: 10.1109/ICCVW54120.2021.00312.
- [4] D. Du et al., "The unmanned aerial vehicle benchmark: Object detection and tracking," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Munich, Germany, 2018, pp. 375–391, doi: 10.1007/978-3-030-01249-6_23.
- [5] P. Zhu et al., "Detection and tracking meet drones challenge," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 11, pp. 7380–7399, Nov. 2022, doi: 10.1109/TPAMI.2021.3119563.
- [6] G. Cheng et al., "Towards large-scale small object detection: Survey and benchmarks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 11, pp. 13467–13488, 2023, doi: 10.1109/TPAMI.2023.3290594.
- [7] M. Kisantal, Z. Wojna, J. Murawski, J. Naruniec and K. Cho, "Augmentation for small object detection," in *Proc. 9th Int. Conf. Adv. Comput. Inf. Technol. (ACITY)*, Sydney, Australia, 2019, pp. 119–133, doi: 10.5121/csit.2019.91713.
- [8] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, 2016, pp. 779–788, doi: 10.1109/CVPR.2016.91.
- [9] W. Liu et al., "SSD: Single shot MultiBox detector," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Amsterdam, The Netherlands, 2016, pp. 21–37, doi: 10.1007/978-3-319-46448-0_2.
- [10] Z. Tian, C. Shen, H. Chen and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Seoul, South Korea, 2019, pp. 9626–9635, doi: 10.1109/ICCV.2019.00972.
- [11] T.-Y. Lin, P. Goyal, R. Girshick, K. He and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, 2017, pp. 2999–3007, doi: 10.1109/ICCV.2017.324.
- [12] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, 2017, pp. 936–944, doi: 10.1109/CVPR.2017.106.
- [13] C.-Y. Wang, A. Bochkovskiy and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Vancouver, BC, Canada, 2023, pp. 7464–7475, doi: 10.1109/CVPR52729.2023.00721.
- [14] Z. Ge, S. Liu, F. Wang, Z. Li and J. Sun, "YOLOX: Exceeding YOLO series in 2021," *arXiv:2107.08430*, 2021, doi: 10.48550/arXiv.2107.08430.
- [15] A. Wang et al., "YOLOv10: Real-time end-to-end object detection," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, Vancouver, BC, Canada, 2024, pp. 107984–108011, doi: 10.52202/079017-3429.
- [16] R. Khanam and M. Hussain, "YOLOv11: An overview of the key architectural enhancements," *arXiv:2410.17725*, 2024, doi: 10.48550/arXiv.2410.17725.
- [17] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Glasgow, U.K., 2020, pp. 213–229, doi: 10.1007/978-3-030-58452-8_13.
- [18] X. Zhu, W. Su, L. Lu, B. Li, X. Wang and J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2021.
- [19] Y. Zhao et al., "DETRs beat YOLOs on real-time object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, 2024, pp. 16965–16974, doi: 10.1109/CVPR52733.2024.01605.
- [20] D. Hendrycks and T. Dietterich, "Benchmarking neural network robustness to common corruptions and perturbations," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, New Orleans, LA, USA, 2019.
- [21] C. Michaelis et al., "Benchmarking robustness in object detection: Autonomous driving when winter is coming," *arXiv:1907.07484*, 2019, doi: 10.48550/arXiv.1907.07484.
- [22] Y. Dong et al., "Benchmarking robustness of 3D object detection to common corruptions in autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Vancouver, BC, Canada, 2023, pp. 1022–1032, doi: 10.1109/CVPR52729.2023.00105.
- [23] S. Dodge and L. Karam, "Understanding how image quality affects deep neural networks," in *Proc. 8th Int. Conf. Qual. Multimedia Exper. (QoMEX)*, Lisbon, Portugal, 2016, pp. 1–6, doi: 10.1109/QoMEX.2016.7498955.
- [24] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann and W. Brendel, "ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, New Orleans, LA, USA, 2019.
- [25] C. Kamann and C. Rother, "Benchmarking the robustness of semantic segmentation models with respect to common corruptions," *Int. J. Comput. Vis.*, vol. 129, no. 2, pp. 462–483, Feb. 2021, doi: 10.1007/s11263-020-01383-2.



- [26] H. He, J. Ding, B. Xu and G.-S. Xia, "On the robustness of object detection models on aerial images," *IEEE Trans. Geosci. Remote Sens.*, vol. 63, Art. no. 5600512, pp. 1–12, 2025, doi: 10.1109/TGRS.2024.3514741.
- [27] A. Arsenos et al., "Common corruptions for evaluating and enhancing robustness in air-to-air visual object detection," *IEEE Robot. Autom. Lett.*, vol. 9, no. 7, pp. 6688–6695, Jul. 2024, doi: 10.1109/LRA.2024.3408485.
- [28] X. Liu, Y. Feng, S. Hu, X. Yuan and H. Fan, "Benchmarking the robustness of UAV tracking against common corruptions," in *Proc. IEEE 7th Int. Conf. Multimedia Inf. Process. Retrieval (MIPR)*, San Jose, CA, USA, 2024, pp. 465–470, doi: 10.1109/MIPR62202.2024.00079.
- [29] J. Liu et al., "Benchmarking object detection robustness against real-world corruptions," *Int. J. Comput. Vis.*, vol. 132, no. 10, pp. 4398–4416, Oct. 2024, doi: 10.1007/s11263-024-02096-6.
- [30] D. Feng, A. Harakeh, S. L. Waslander and K. Dietmayer, "A review and comparative study on probabilistic object detection in autonomous driving," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 9961–9980, Aug. 2022, doi: 10.1109/TITS.2021.3096854.
- [31] C. Guo, G. Pleiss, Y. Sun and K. Q. Weinberger, "On calibration of modern neural networks," in *Proc. 34th Int. Conf. Mach. Learn. (ICML)*, Sydney, Australia, 2017, pp. 1321–1330.
- [32] F. Kupperts, J. Kronenberger, A. Shantia and A. Haselhoff, "Multivariate confidence calibration for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Seattle, WA, USA, 2020, pp. 1322–1330, doi: 10.1109/CVPRW50498.2020.00171.
- [33] M. P. Naeni, G. Cooper and M. Hauskrecht, "Obtaining well calibrated probabilities using Bayesian binning," in *Proc. AAAI Conf. Artif. Intell.*, Austin, TX, USA, 2015, pp. 2901–2907, doi: 10.1609/aaai.v29i1.9602.
- [34] K. Oksuz, T. Joy and P. K. Dokania, "Towards building self-aware object detectors via reliable uncertainty quantification and calibration," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Vancouver, BC, Canada, 2023, pp. 9263–9274, doi: 10.1109/CVPR52729.2023.00894.
- [35] B. Pathiraja, M. Gunawardhana and M. H. Khan, "Multiclass confidence and localization calibration for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Vancouver, BC, Canada, 2023, pp. 19734–19743, doi: 10.1109/CVPR52729.2023.01890.
- [36] T. Popordanoska, A. Tiulpin and M. B. Blaschko, "Beyond classification: Definition and density-based estimation of calibration in object detection," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Waikoloa, HI, USA, 2024, pp. 574–583, doi: 10.1109/WACV57701.2024.00064.
- [37] S. Kuzucu, K. Oksuz, J. Sadeghi and P. K. Dokania, "On calibration of object detectors: Pitfalls, evaluation and baselines," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Milan, Italy, 2024, pp. 185–204, doi: 10.1007/978-3-031-72664-4_11.
- [38] D. Huseljic, M. Herde, P. Hahn, M. Mjude and B. Sick, "Systematic evaluation of uncertainty calibration in pretrained object detectors," *Int. J. Comput. Vis.*, vol. 133, no. 3, pp. 1033–1047, Mar. 2025, doi: 10.1007/s11263-024-02219-z.
- [39] B. Zadrozny and C. Elkan, "Transforming classifier scores into accurate multiclass probability estimates," in *Proc. 8th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Edmonton, AB, Canada, 2002, pp. 694–699, doi: 10.1145/775047.775151.
- [40] A. Niculescu-Mizil and R. Caruana, "Predicting good probabilities with supervised learning," in *Proc. 22nd Int. Conf. Mach. Learn. (ICML)*, Bonn, Germany, 2005, pp. 625–632, doi: 10.1145/1102351.1102430.
- [41] T.-Y. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Zurich, Switzerland, 2014, pp. 740–755, doi: 10.1007/978-3-319-10602-1_48.
- [42] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn and A. Zisserman, "The Pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010, doi: 10.1007/s11263-009-0275-4.
- [43] G. W. Brier, "Verification of forecasts expressed in terms of probability," *Mon. Weather Rev.*, vol. 78, no. 1, pp. 1–3, Jan. 1950, doi: 10.1175/1520-0493(1950)078<0001:VOFEIT>2.0.CO;2.
- [44] H. R. Künsch, "The jackknife and the bootstrap for general stationary observations," *Ann. Statist.*, vol. 17, no. 3, pp. 1217–1241, Sep. 1989, doi: 10.1214/aos/1176347265.
- [45] B. Efron, "Bootstrap methods: Another look at the jackknife," *Ann. Statist.*, vol. 7, no. 1, pp. 1–26, Jan. 1979, doi: 10.1214/aos/1176344552.
- [46] M. Friedman, "The use of ranks to avoid the assumption of normality implicit in the analysis of variance," *J. Amer. Statist. Assoc.*, vol. 32, no. 200, pp. 675–701, Dec. 1937, doi: 10.1080/01621459.1937.10503522.
- [47] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics Bull.*, vol. 1, no. 6, pp. 80–83, Dec. 1945, doi: 10.2307/3001968.
- [48] S. Holm, "A simple sequentially rejective multiple test procedure," *Scand. J. Statist.*, vol. 6, no. 2, pp. 65–70, 1979.
- [49] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, 2006.