Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

A PRISMA-Guided Critical Review of CNN, Transformer, and Hybrid Object Detection Architectures for Autonomous Driving

Girija G. Chiddarwar
Marthwada Mitra Mandal's College of Engineering
Madhav J. Salunkhe
Annasaheb Dange College of Engineering and Technology
Yogesh G. Joshi
Ramdeobaba University
Rahul G. Deshmukh
Dr. D. Y. Patil Institute of Technology
Swapnil Yashavant Gadgune
Padmabhooshan Vasantraodada Patil Institute of Technology
Suraj Rajaram Nalawade
Srinivas University
Arunkumar Bongale
arun.bongale@sitpune.edu.in

Symbiosis Institute of Technology, Symbiosis International (Deemed University)

Research Article

Keywords: Autonomous vehicles, Object detection, CNN–Transformer hybrid, Real-time perception, Adverse-condition robustness, Edge deployment

Posted Date: August 27th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10381781/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

A PRISMA-Guided Critical Review of CNN, Transformer, and Hybrid Object Detection Architectures for Autonomous Driving

Girija G. Chiddarwar[1], Madhav J. Salunkhe[2], Yogesh G. Joshi[3], Rahul G. Deshmukh[4*], Swapnil Yashavant Gadgune[5], Suraj Rajaram Nalawade[6], Arunkumar Bongale[7*]

[1]Department of Computer Engineering, Marthwada Mitra Mandal's College of Engineering, Pune, India

[2]Department of Computer Science Engineering, Annasaheb Dange College of Engineering and Technology, Ashta, India

[3]Department of Mechanical Engineering, School of Engineering Sciences, Ramdeobaba University, Nagpur, India

[4]Department of Mechanical Engineering, Dr. D. Y. Patil Institute of Technology, Pune, India

[5]Department of Electrical and Computer Engineering, Padmabhooshan Vasantraodada Patil Institute of Technology, Budhgaon, India

[6]Department of Artificial Intelligence & Data Science, Srinivas University, Mangalore, India

[7]Department of Robotics and Automation, Symbiosis Institute of Technology, Symbiosis International (Deemed University), Pune, India

Corresponding Author Email Address: arun.bongale@sitpune.edu.in

Abstract

Object detection is the safety-critical core of autonomous vehicle (AV) perception, yet benchmark accuracy alone provides limited evidence of real-world deployment readiness. A detector that performs well on public datasets may still fail under adverse weather, miss distant pedestrians, or exceed the latency constraints of embedded automotive hardware. Although existing reviews comprehensively describe the evolution of CNN, Transformer, and hybrid-based detectors, they do not systematically evaluate deployment evidence or account for inconsistencies in datasets, hardware platforms, and evaluation protocols. This review addresses these limitations through three original contributions. First, a unified taxonomy categorizes detector families according to the integration of local convolutional and global attention mechanisms and relates these design choices to AV-specific operational challenges. Second, an evidence-comparability protocol defines the conditions under which reported accuracy, latency, and robustness can be validly compared, exposing common sources of misleading cross-study comparisons. Third, the proposed AV Object Detection Deployment-Readiness Framework (AVOD-DRF) transforms published evidence into an eleven-dimension deployment-readiness profile, where each score is explicitly evidence-linked and missing

evidence is distinguished from poor performance. These contributions are supported by a PRISMA 2020-guided systematic review of six bibliographic databases covering 2015 to 2026, resulting in 142 eligible studies. The analysis reveals substantial reporting gaps: only 13% evaluate latency on automotive or edge hardware, 19% assess adverse weather or low-light conditions, and 15% perform cross-dataset validation. No detector family is universally superior; rather, deployment suitability depends on application constraints. The proposed framework provides a reproducible, evidence-based methodology for selecting AV object detectors based on real-world deployment readiness rather than benchmark performance alone.

Keywords Autonomous vehicles; Object detection; CNN–Transformer hybrid; Real-time perception; Adverse-condition robustness; Edge deployment

1. Introduction

Object detection is the perceptual foundation of an autonomous vehicle (AV). Every downstream module — tracking, prediction, planning, and control — acts on the objects that the detector localizes and classifies: vehicles, pedestrians, cyclists, traffic signs, traffic lights, and unexpected road hazards [1, 2]. This coupling to safety makes AV detection categorically different from general-purpose detection. A missed pedestrian is not a fractional loss of mean average precision (mAP); it is a potential collision. A deployable detector must therefore satisfy four constraints at the same time: sufficient detection accuracy, bounded end-to-end latency, robustness to environmental degradation, and conformance to the compute, memory, and power limits of embedded automotive hardware [3, 4]. Fig. 1 summarizes this problem setting and the constraints that a deployable detector must satisfy simultaneously.

These constraints pull in opposite directions, and their tension defines the central deployment conflict of the field. Detectors that lead public benchmarks are typically large and computationally demanding, exceeding what embedded automotive platforms can provide. Lightweight detectors that meet in-vehicle latency budgets tend to lose exactly the capabilities that matter most on the road: sensitivity to small and distant objects and stability under rain, fog, and low light. Published results rarely expose this conflict, because accuracy, speed, and robustness are reported under heterogeneous combinations of datasets, input resolutions, hardware platforms, numerical precisions, and evaluation thresholds. The literature

consequently contains many claims of "real-time" and "robust" detection that cannot be compared with one another and, in some cases, cannot be verified at all.

Architecturally, the field has passed through three overlapping phases within the broader rise of deep learning [5]. Convolutional neural networks (CNNs) established the modern detection pipeline, first through two-stage region-proposal designs [6, 7] and then through one-stage detectors optimized for speed [8, 9]. Transformer-based detectors subsequently reformulated detection as end-to-end set prediction, replacing the local receptive fields of convolution with global self-attention [10, 11]. Most recently, hybrid designs have sought to combine convolutional efficiency with attention-based context modelling [12, 13] and to extend detection across camera, LiDAR, and radar modalities [14, 15]. Fig. 2 gives a hierarchical view of these families and their enabling techniques, and Fig. 3 summarizes how the enabling techniques relate to deployment objectives.

Each of these phases has attracted its own reviews. Surveys of vision Transformers and their CNN–Transformer variants provide broad architectural taxonomies [16, 17], and a recent survey maps Transformer architectures across autonomous-driving tasks [18]. Reviews of vehicle detection [1] and of the YOLO lineage [19] each cover a single family. Reviews of adverse-weather perception [4] and general object detection [20, 21] each cover a single concern, while earlier surveys of deep learning for driving predate the Transformer transition [22, 23]. Broader AV reviews address research strategy and applications [24, 25] or motion prediction [26] rather than detector evidence. Table 1 compares these reviews by evidence period, method, scope, and deployment coverage.

Two limitations recur across Table 1. First, most existing reviews are narrative rather than systematic: they report no search protocol, no screening record, and no quality assessment, so their evidence base cannot be reproduced or audited. Second, and more consequentially, they compare results across studies without correcting for dataset, hardware, numerical precision, or evaluation protocol. A frame rate measured on a data-center GPU is quoted beside one measured on an embedded module; a mAP at one IoU threshold is ranked against a mAP at another. Benchmark accuracy is thereby treated as a proxy for deployment readiness, although the two are separated by precisely the evidence — latency on automotive hardware, robustness under adverse conditions, generalization across domains — that most primary studies never report. No existing review systematically links benchmark accuracy to robustness and deployment evidence under explicit comparability rules. Closing that gap is the purpose of the present review.

With the gap established, the review is organized around five research questions. RQ1: How have CNN-based and Transformer-based detection architectures evolved for autonomous-driving perception, and what are the fundamental architectural differences between them? RQ2: Which accuracy–latency–complexity trade-offs are supported by comparable evidence on shared AV datasets and deployment hardware? RQ3: How do the architecture families perform under small-object, occlusion, adverse-weather, low-light, domain-shift, and adversarial conditions? RQ4: How effectively do self-supervised learning (SSL), neural architecture search (NAS), pruning, quantization, and knowledge distillation improve the deployment readiness of AV detectors? RQ5: Which reporting, benchmarking, and validation gaps currently prevent evidence-based model selection for real-world autonomy?

The central scientific contribution of this review is the AV Object Detection Deployment-Readiness Framework (AVOD-DRF): a reusable methodology that converts the reported evidence of any detection study into an eleven-dimension readiness profile, in which every non-zero score must cite extracted evidence and missing evidence is never conflated with poor performance (Sect. 6). Three supporting contributions make the framework possible. First, a reproducible, PRISMA-guided evidence base of 142 studies searched to June 2026, with documented search strings, eligibility criteria, and exclusion reasons (Sect. 2). Second, a unified taxonomy that classifies detectors not by family labels alone but by how and where they integrate local and global representations, and that connects each design choice to AV-specific failure conditions (Sect. 3). Third, an evidence-comparability protocol that defines when published results may legitimately be compared and that the comparative synthesis of Sects. 4–5 applies throughout. An actionable research agenda, in which every priority is paired with a falsifiable validation protocol, completes the review (Sect. 7), and Sect. 8 concludes. The review deliberately offers no new detector; it offers the synthesis, the comparability rules, and the decision logic that previous reviews do not provide.

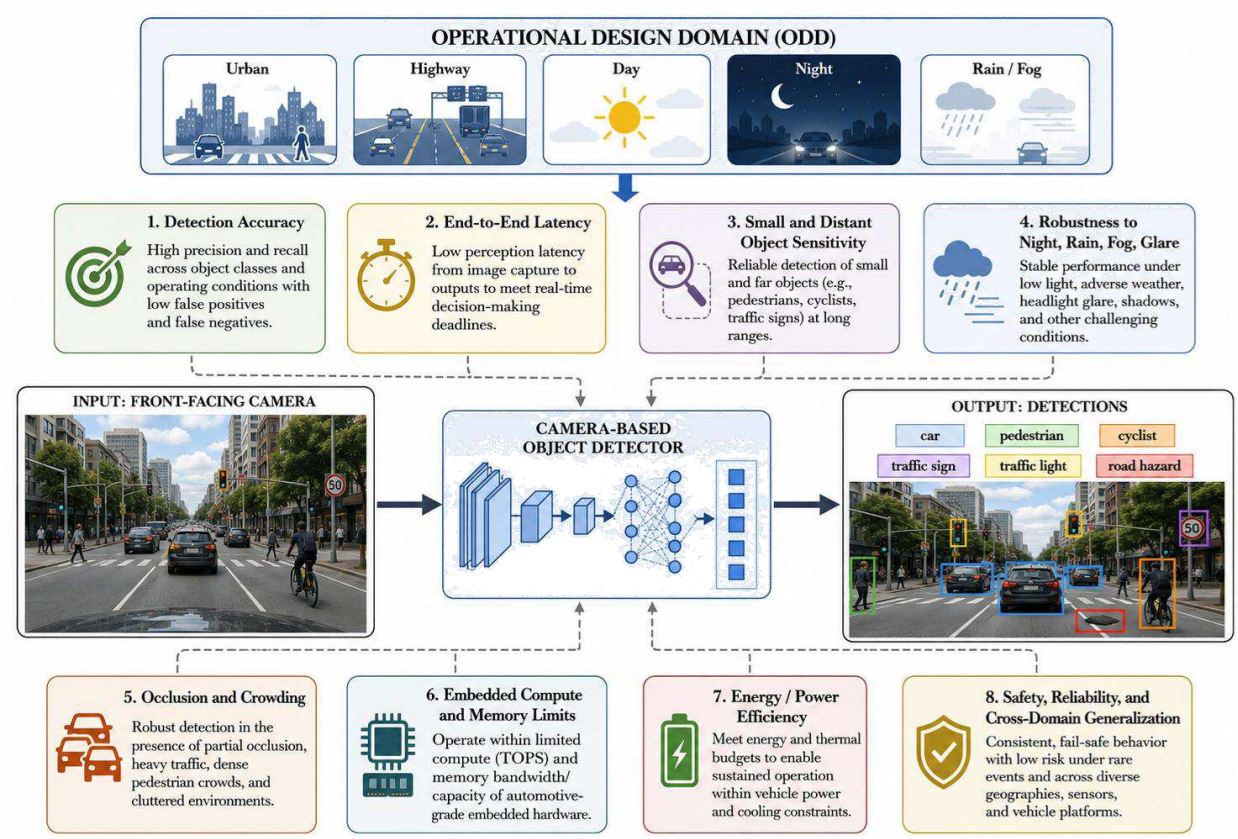


Fig. 1 Scope and simultaneous deployment constraints of camera-based object detection in autonomous vehicles.

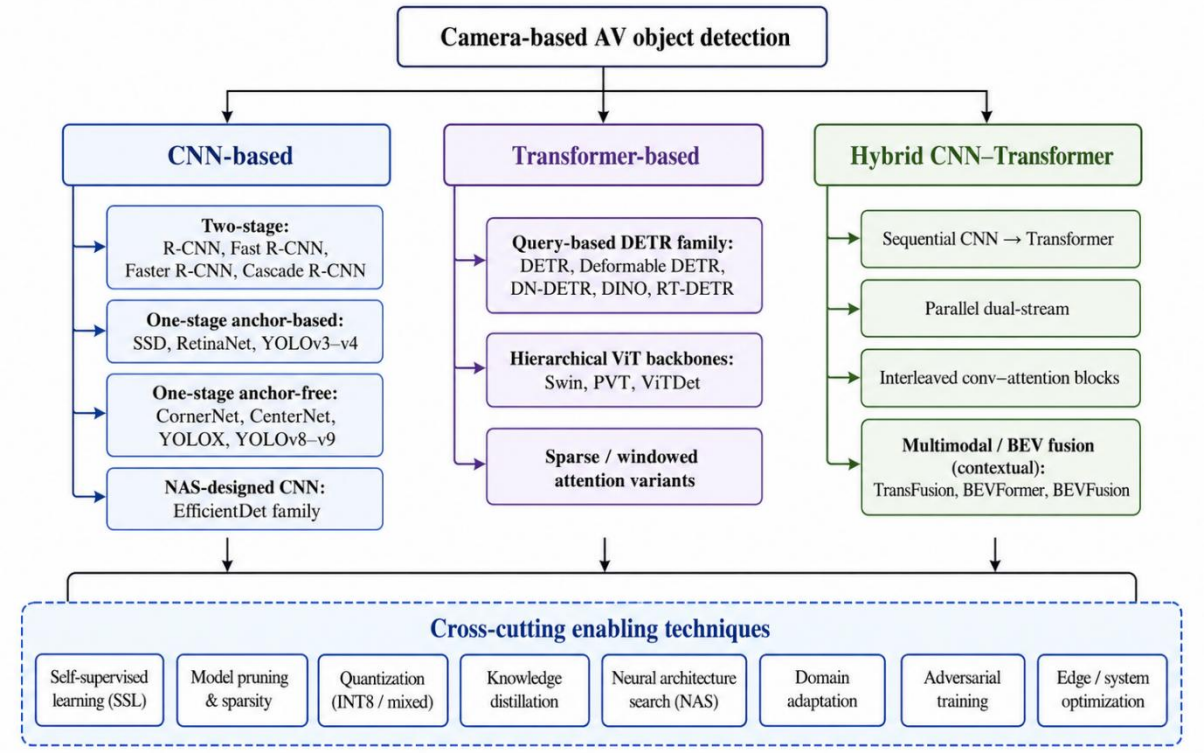


Fig. 2 Unified taxonomy of CNN-based, Transformer-based, and hybrid detector families with representative models and cross-cutting enabling techniques.

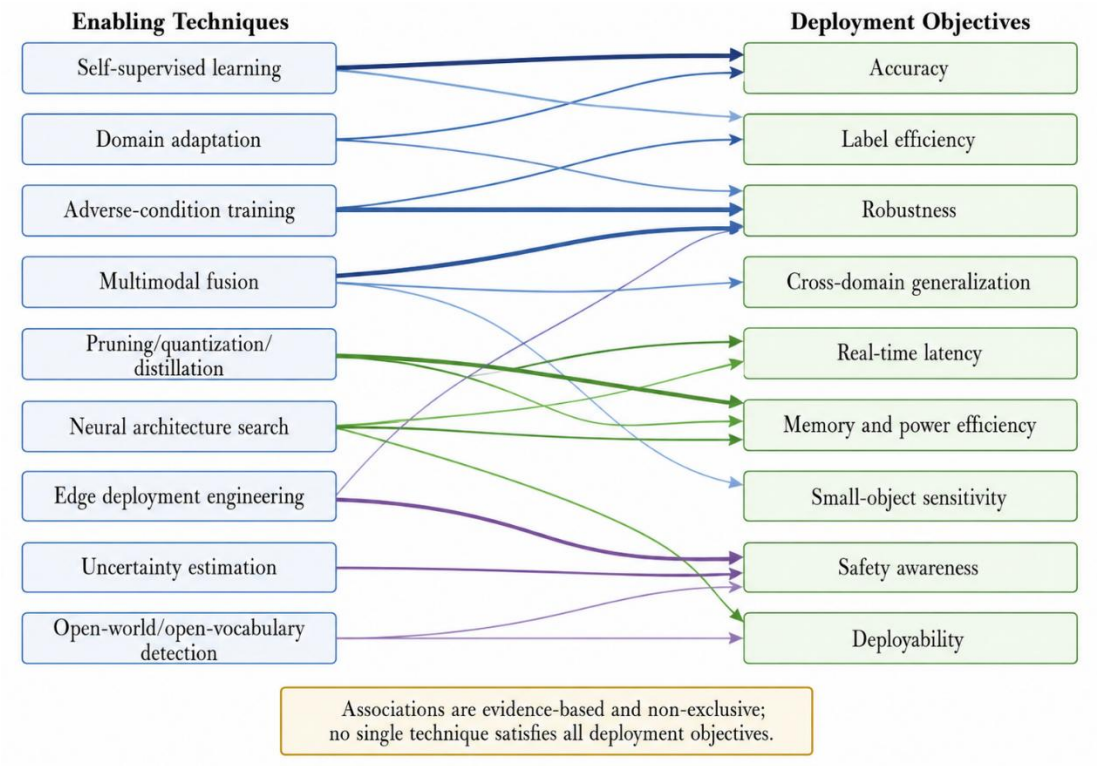


Fig. 3 Principal associations between enabling techniques and deployment objectives identified across the included studies.

Table 1. Gap analysis of recent reviews related to object detection for autonomous driving (✓ covered; P partial; — not covered)

Review	Evidence period	Systematic method	CNN	Transformer	Hybrid	Adverse conditions	Deployment / edge	Limitation addressed by the present review
Fujiyoshi et al. (2019) [23]	to 2019	—	✓	—	—	—	P	Predates the Transformer and hybrid transition
Wu et al. (2020) [22]	to 2019	—	✓	—	—	P	P	General-purpose; no AV deployment analysis
Zaidi et al. (2022) [21]	to 2021	—	✓	P	—	—	P	No AV focus; no evidence protocol
Zou et al. (2023) [20]	1998–2022	—	✓	P	—	P	P	Historical breadth; no deployment framework
Terven et al. (2023) [19]	to 2023	—	✓	—	P	—	P	Single family (YOLO)
Khan et al. (2023) [16]	to 2023	—	P	✓	✓	P	—	Not AV-specific; no comparability analysis
Huang et al. (2023) [26]	to 2023	P	P	P	—	—	—	Motion prediction, not detection

Zhang et al. (2023) [4]	to 2022	—	P	—	—	✓	P	Sensing focus; no architecture comparison
Liang et al. (2024) [1]	to 2024	—	✓	P	P	P	P	Narrative method; limited comparability analysis
Cao et al. (2025) [17]	to 2024	—	✓	✓	✓	P	—	Not AV-specific; no deployment readiness
Chu et al. (2026) [18]	to 2025	P	P	✓	P	P	P	Transformer-centric; no evidence-comparability protocol
This review	2015 – June 2026	✓ (PRISMA 2020)	✓	✓	✓	✓	✓	—

2. Review Methodology

2.1 Search Protocol and Eligibility Criteria

The review follows the PRISMA 2020 statement for identification, screening, eligibility assessment, and inclusion [27]. A systematic protocol is not a formality here. The central claim of this review is that the field's deployment conclusions outrun its evidence, and such a claim is credible only if the evidence base behind it is itself reproducible and auditable. The protocol design also follows recent methodological precedents in this journal that combine PRISMA guidance with explicit taxonomies and study-quality rubrics [28, 29]. The protocol was fixed before full-text extraction began; it was not preregistered, and this is acknowledged as a limitation in Sect. 8.

Six bibliographic databases were searched: Scopus, Web of Science Core Collection, IEEE Xplore, ACM Digital Library, ScienceDirect, and SpringerLink. This combination was chosen for the specific research field under review. IEEE Xplore and the ACM Digital Library index the engineering venues in which most detection architectures and embedded-systems evaluations appear; ScienceDirect and SpringerLink capture the applied journals that publish autonomous-driving case studies and robustness evaluations; and Scopus and Web of Science add cross-disciplinary coverage and curated citation metadata that protect the search against venue bias. Google Scholar was restricted to backward and forward citation chasing because its index is neither stable nor reproducible enough to serve as a primary source in a systematic search, although it is effective for recovering relevant work that database queries miss. The search covered 1 January 2015 to the closure date of 30 June 2026. The start year is not arbitrary: 2015 marks the consolidation of the modern deep detection pipeline around learnable

region proposals [7, 30] and unified one-stage prediction [8], so earlier studies describe a methodologically different generation of detectors. Seminal earlier works are retained for historical grounding but were not part of the systematic count.

The master search string combined three concept blocks, adapted to the syntax of each database: ("autonomous vehicle*" OR "autonomous driving" OR "self-driving" OR ADAS) AND ("object detection" OR "vehicle detection" OR "pedestrian detection" OR "traffic object detection") AND (CNN OR convolution* OR transformer* OR DETR OR YOLO OR hybrid) AND ("real-time" OR latency OR edge OR robustness OR "adverse weather" OR "domain shift" OR occlusion OR "small object"). Database-specific syntax and field restrictions are reported in Table 2, and the complete per-database strings are provided as supplementary material.

Studies were included when they (i) were peer-reviewed journal articles or full conference papers written in English; (ii) evaluated object detection in autonomous-driving or road-traffic scenes; (iii) reported a defined architecture and quantitative results with at least one recognized detection metric; (iv) identified the dataset or data source; and (v) fell within the search period. A small number of non-peer-reviewed technical reports with no formal alternative — for example, widely used detector releases available only as preprints — were retained and are labelled as preprints throughout. Studies were excluded when they were duplicates, abstract-only records, or editorials; when they were restricted to segmentation, tracking, lane detection, motion prediction, or planning without a detection contribution; when they addressed remote-sensing, industrial-inspection, face-recognition, or medical imagery without AV evaluation; when no quantitative detection results or identifiable dataset were reported; or when the full text was inaccessible or retracted. Camera–LiDAR and bird's-eye-view (BEV) 3D detection studies were retained only as a clearly separated contextual category, because the core synthesis concerns camera-based 2D detection.

2.2 Screening, Quality Assessment, and Data Extraction

Records were exported with full metadata and deduplicated by digital object identifier (DOI) and normalized title. Titles and abstracts were screened against the eligibility criteria, and the surviving full texts were assessed against the same criteria. Fig. 4 presents the PRISMA flow and the full-text exclusion reasons. Of the 1,956 database records, 683 duplicates were removed and 1,273 records were screened by title and abstract, of which 986 were excluded. Of the 287 full texts sought, 9 could not be retrieved and 278 were assessed for eligibility; 152 of these

were excluded with documented reasons, leaving 126 included database studies. Citation chasing contributed a further 24 records, of which 16 satisfied all eligibility criteria at full-text assessment, giving 142 included studies in total (126 from databases and 16 from citation chasing). The most frequent full-text exclusion reasons were an out-of-scope application domain (27%) and the absence of an object-detection evaluation (21%) (Fig. 4b). The complete list of the 142 included studies, together with the screening log, is provided as supplementary material; the reference list of this article comprises the included studies that are explicitly discussed in the text, together with the foundational and methodological works that support the analysis.

Screening was performed by a single reviewer, and this is the principal methodological limitation of the review. It was mitigated rather than ignored: a 20% random sample of excluded records was re-screened after a two-week interval, and every borderline record was re-examined against the written criteria before a final decision. Because no second independent reviewer participated, no inter-rater statistic is reported, and a residual risk of selection bias remains. Three further sources of bias shape how the synthesized evidence should be read. The English-language restriction may under-represent deployment studies from communities that publish in other languages, which plausibly narrows the geographic diversity of the driving scenes represented in the evidence base. Publication bias favours positive results, which implies that the robustness failures synthesized in Sect. 5 are more likely under-reported than over-reported; the true adverse-condition picture is unlikely to be better than the published one. Finally, the heterogeneity of benchmark protocols across studies does not merely complicate synthesis — it invalidates specific classes of cross-study comparison, which is why the comparability rules of Sect. 2.3 are integral to the methodology rather than a caveat appended to it.

Each included study was assessed with a ten-criterion quality and deployment-readiness rubric (Table 3), scored 0 (not reported), 1 (partially demonstrated), or 2 (comprehensively demonstrated). The rubric is purpose-built for the question this review asks. Generic quality checklists for machine-learning studies emphasize internal validity — datasets, baselines, ablations — but deployment-oriented claims fail for different reasons: unreported inference protocols, missing hardware specifications, absent adverse-condition evidence, and unavailable code. The ten criteria therefore cover dataset and split transparency, metric definition, hardware and inference-protocol reporting, architectural reproducibility, code or model availability, adverse-condition evaluation, ablation analysis, external or cross-dataset

validation, statistical reporting, and limitation disclosure — the union of what a reader needs to reproduce a study and what a practitioner needs to trust its deployment relevance. The scoring anchors were fixed before extraction. For example, hardware reporting receives 2 only when the device, input resolution, numerical precision, batch size, and software runtime are all stated; a desktop-GPU frame rate alone receives 1. Quality scores were used to weight the interpretation of evidence, not to exclude studies: an influential early architecture with limited reporting remains in the historical synthesis but does not support deployment conclusions.

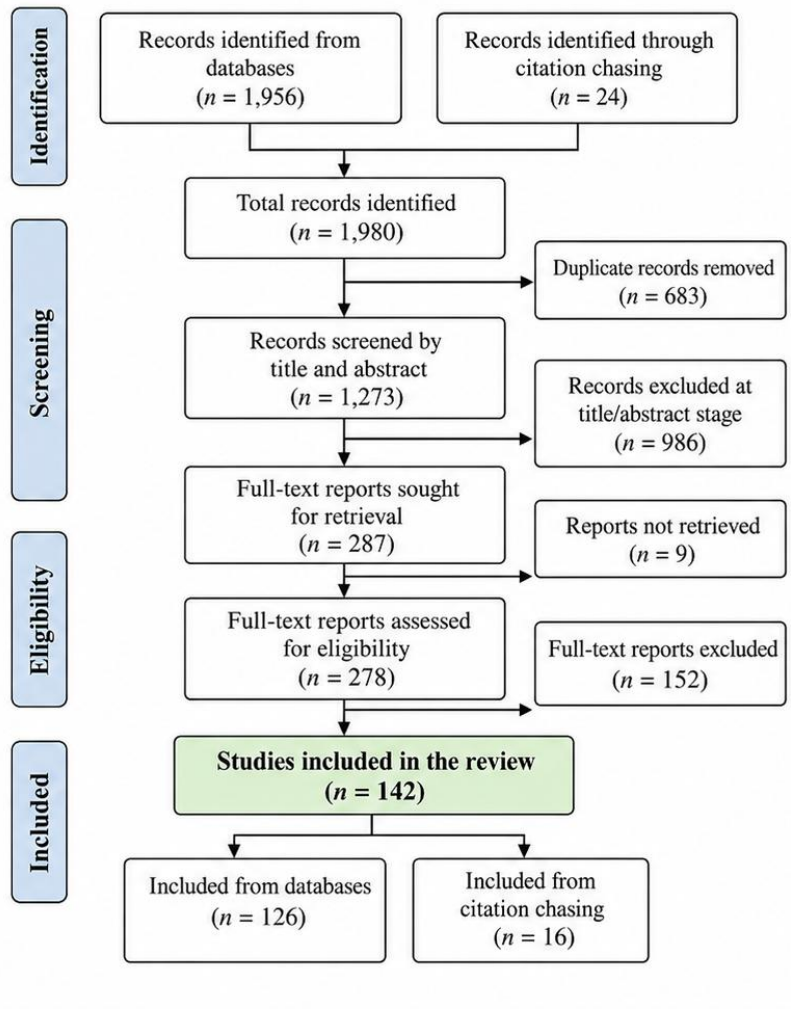
The data-extraction form captured, for every included study: authors and year; detector family and integration type; backbone, neck, and head; dataset and split; object classes; input resolution; mAP definition and intersection-over-union (IoU) threshold; precision and recall where reported; small-object results; latency or frames per second (FPS) with hardware, precision, and batch size; parameters and floating-point operations (FLOPs); memory; adverse-condition, occlusion, adversarial, and cross-dataset results; uncertainty estimation; code availability; and the main stated limitation. Fig. 5 profiles the included studies: panel (a) shows the growth of the literature to the search date, panel (b) the distribution across architecture families, and panel (c) the deployment-relevant reporting coverage that motivates much of the critique in Sects. 4–6.

2.3 Evidence Synthesis and Comparative Framework

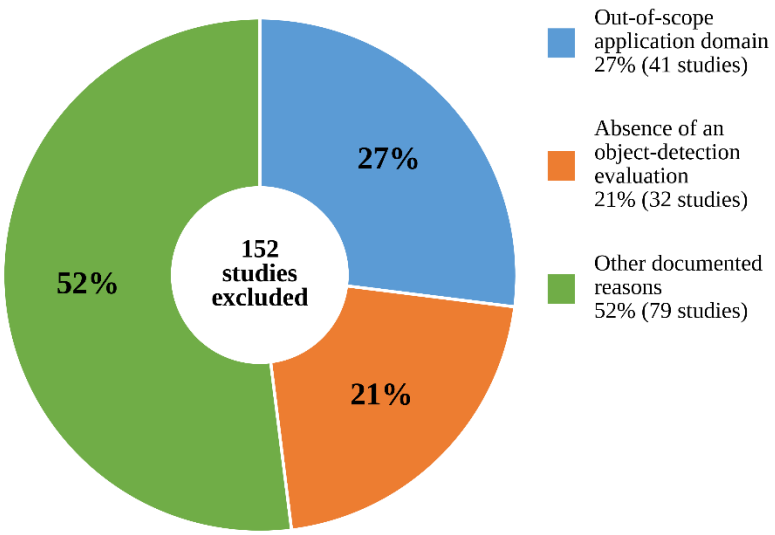
The heterogeneity of datasets, metrics, hardware, and protocols across the included studies rules out a valid meta-analysis, so the synthesis is structured and semi-quantitative. The underlying problem deserves precise statement, because it motivates the comparability protocol at the heart of this review. A reported mAP is a function not only of the detector but also of the dataset and split, the IoU threshold, the averaging convention, and the input resolution. A reported frame rate additionally depends on the hardware platform, the numerical precision, the batch size, and the inference runtime. When any of these factors differs between two studies, the difference between their reported numbers cannot be attributed to the architecture — yet cross-study tables that ignore this distinction dominate the survey literature. The protocol therefore treats two reported results as accuracy-comparable only when dataset and split, metric definition, IoU threshold, and input resolution match, and as throughput-comparable only when hardware, numerical precision, batch size, and inference runtime also match. Results that fail these checks are synthesized as evidence ranges or qualitative statements, never ranked numerically. Each quantitative claim in this review carries its

experimental context, and any value taken from a source paper is attributed to that paper's own evaluation conditions.

Applying this protocol is itself one of the review's instruments of analysis: it shows that many frequently quoted cross-model comparisons in the literature — for example, frame rates measured on desktop graphics processors quoted as evidence of embedded feasibility — are not comparable at all. Across the 142 included studies, 43% report a desktop-GPU latency, but only 13% report latency on automotive or edge hardware, and only 5% report energy or power (Fig. 5c). The deployment-readiness framework built on this evidence is presented in Sect. 6.



(a)



(b)

Fig. 4 Study selection: (a) PRISMA 2020 flow for the 1,980 records identified (1,956 database; 24 citation chasing) and 142 included studies; (b) full-text exclusion reasons.

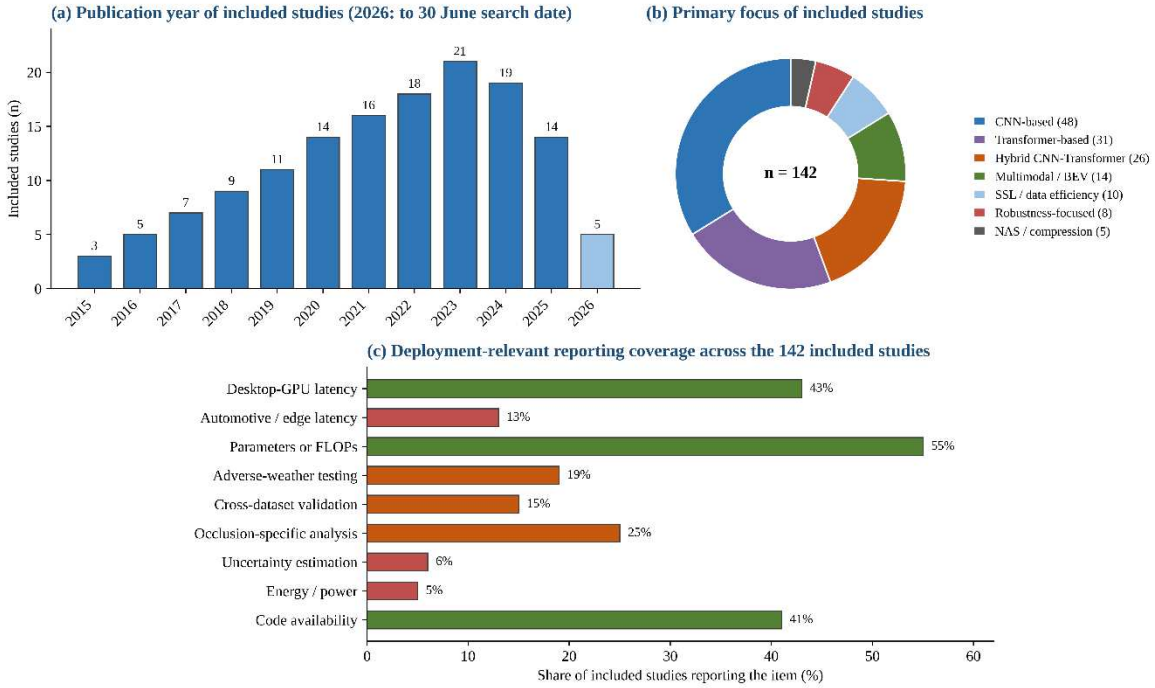


Fig. 5 Evidence profile of the 142 included studies: (a) publication year to the 30 June 2026 search date; (b) primary architecture focus; (c) deployment-relevant reporting coverage.

Table 2 Systematic review protocol summary (full database-specific search strings are provided as supplementary material)

Protocol element	Specification
Reporting guideline	PRISMA 2020 [27]

Databases	Scopus; Web of Science Core Collection; IEEE Xplore; ACM Digital Library; ScienceDirect; SpringerLink. Google Scholar used only for backward/forward citation chasing
Search period / closure	1 January 2015 to 30 June 2026 (date search was closed)
Search concepts	(AV terms) AND (object-detection terms) AND (CNN / Transformer / DETR / YOLO / hybrid) AND (real-time / latency / edge / robustness / adverse weather / domain shift / occlusion / small object)
Records identified	1,956 from databases; 24 from citation chasing
Deduplication	683 duplicates removed (DOI and normalized-title matching); 1,273 records screened
Eligibility assessment	287 full texts sought; 9 not retrieved; 278 assessed; 152 excluded with documented reasons (Fig. 4b)
Included studies	142 (126 database + 16 citation chasing)
Screening procedure	Single reviewer with structured verification: 20% random re-screen of exclusions after a two-week interval; all borderline records re-examined against written criteria
Quality assessment	Ten-criterion rubric, each scored 0–2 with pre-registered anchors (Table 3)
Data extraction	27-field form covering architecture, dataset, metrics, hardware, robustness, and reproducibility items (Sect. 2.2)
Synthesis method	Structured narrative and semi-quantitative synthesis under explicit comparability rules; no meta-analysis owing to heterogeneity of datasets, metrics, and hardware

Table 3 Study-quality and deployment-readiness rubric applied to every included study (0 = not reported; 1 = partially demonstrated; 2 = comprehensively demonstrated)

Criterion	Score 1 anchor (partial)	Score 2 anchor (comprehensive)
Dataset and split transparency	Dataset named; split unclear	Dataset, split, and any filtering fully specified
Metric definition	mAP quoted without IoU threshold or definition	Metric, IoU threshold, and averaging convention stated
Hardware and inference protocol	Device named; precision, batch size, or runtime missing	Device, input size, precision, batch size, and runtime engine all stated
Architecture reproducibility	Architecture described at block level	Full configuration reproducible from paper or repository
Code / model availability	Partial code or on-request statement	Public code and weights matching reported results
Adverse-condition evaluation	Single condition or synthetic-only test	Multiple conditions or condition-controlled (paired) evaluation
Ablation analysis	Single ablation	Systematic ablation of the claimed contributions
External / cross-dataset validation	Second split of the same dataset	Unmodified evaluation on a different dataset with degradation reported
Statistical reporting	Single-run results	Multiple seeds/runs with variance, or significance analysis
Limitation disclosure	Generic limitation statement	Specific, evidence-linked limitations discussed

3. Evolution and Unified Taxonomy of Object Detection for Autonomous Driving

This section synthesizes the architectural evidence around a single question asked of every family: where does the design place local and global reasoning, and what does that placement cost or buy under autonomous-driving conditions? The datasets on which this evidence was produced are summarized first, because they bound what the evidence can show. Fig. 6 maps the six most used driving datasets in the included studies. KITTI remains the most cited benchmark but covers a single German region in clear daytime weather [31]; Cityscapes adds urban diversity but not adverse conditions [32]; BDD100K and nuScenes add night and rain [33, 34]; Waymo Open adds scale and multiple regions [35]; and ACDC provides paired clear/adverse references designed specifically for condition-controlled evaluation [36]. Across the included studies, however, evaluation remains concentrated on the clear-weather portions of these datasets. This concentration is the first structural reason why benchmark accuracy overstates road readiness.

Fig. 7 places the architectural evolution on a single timeline, from the region-based CNNs of 2014–2015 to the real-time Transformer and open-vocabulary detectors reported up to the June 2026 search date, and Table 4 presents the unified taxonomy used throughout the remainder of the review.

Dataset (year)	Scale (camera)	Sensors	Conditions covered	2D detection labels
KITTI (2012)	15 k frames	Camera, LiDAR, GPS/IMU	Daytime, clear; one German city	Cars, pedestrians, cyclists
Cityscapes (2016)	25 k images	Stereo camera	Daytime; 50 European cities	Via instance masks (8 classes)
BDD100K (2020)	100 k images	Camera, GPS/IMU	Day/night, rain, fog, snow; US cities	10 classes incl. signs and lights
nuScenes (2020)	1.4 M images	6 cameras, LiDAR, radar, GPS/IMU	Day/night, rain; Boston, Singapore	23 classes (3D boxes; 2D derived)
Waymo Open (2020)	1 M frames	5 cameras, 5 LiDAR	Day/night, rain; multiple US regions	4 classes (2D and 3D boxes)
ACDC (2021)	4 k images	Camera	Fog, night, rain, snow (paired clear refs.)	Via semantic labels (19 classes)

Fig. 6 Public driving datasets used in the included studies, with approximate scale, sensors, condition coverage, and detection-relevant annotations.

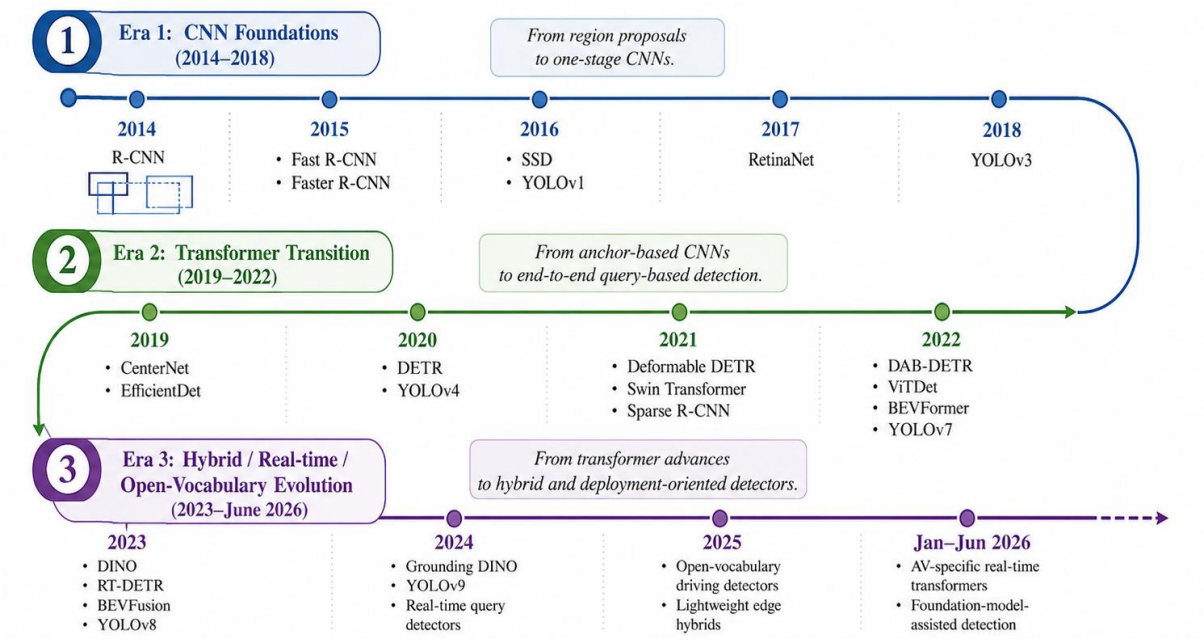


Fig. 7 Evolution of object-detection architectures relevant to autonomous driving, 2014 to the June 2026 search date.

Table 4 Unified taxonomy of detector families and integration types with autonomous-driving consequences

Architecture / integration type	Local–global representation	Fusion stage	Computational tendency	Major strength	Major limitation	AV relevance
Two-stage CNN (Faster / Cascade R-CNN)	Local; context via pooling over proposals	—	High latency; irregular compute	Localization accuracy	Latency floor from two stages	Offline analysis; research platforms
One-stage anchor-based CNN (SSD, YOLOv3–v4, RetinaNet)	Local; multi-scale via FPN	—	Low latency; regular compute	Mature embedded toolchain	Anchor priors need dataset tuning	Primary deployed family
One-stage anchor-free CNN (CornerNet, CenterNet, YOLOv8–v9)	Local; keypoint/center prediction	—	Low latency	No anchor tuning	Assignment-rule sensitivity	Embedded real-time detection
NAS-designed CNN (EfficientDet family)	Local; searched multi-scale fusion	Neck (searched)	Scalable frontier	Accuracy-compute scaling	Search cost; proxy objectives	Compute-budgeted variants
Query-based Transformer (DETR, Deformable	Global attention over CNN/ViT features	Head	High memory; improving latency	Crowded/occluded scenes; no NMS	Edge quantization evidence scarce	Urban dense traffic

DETR, DN-DETR, DINO, RT-DETR)						
Hierarchical ViT backbone (Swin, PVT, ViTDet)	Windowed / hierarchical attention	Backbone	Near-linear in tokens	Strong general features	Pre-training data demand	Backbone for either head type
Sequential hybrid (CNN → Transformer)	Local then global	Backbone→head	Moderate	Simple integration	Information bottleneck at interface	Most common hybrid pattern
Parallel hybrid (dual-stream)	Local and global side-by-side	Neck / fusion module	High (two streams)	Preserves both feature types	Cost duplication; fusion tuning	Research designs; few deployments
Interleaved hybrid (CMT-style)	Alternating conv / attention	Backbone (internal)	Mobile-scale possible	Accuracy-compute frontier	Hard to quantize; operator diversity	Candidate for embedded hybrids
Multimodal / BEV fusion (TransFusion, BEVFormer, BEVFusion)	Global cross-sensor attention	BEV space	Very high	Sensor redundancy	High-end GPU evidence only	Contextual (3D); outside core scope

3.1 CNN-Based Detectors

The two-stage lineage established detection accuracy as a learnable problem. R-CNN and Fast R-CNN demonstrated that region proposals could be classified from shared convolutional features [6, 30], and Faster R-CNN made the proposal stage itself learnable through the region proposal network, defining the template for high-accuracy detection [7]. Cascade R-CNN later showed that resampling proposals at increasing IoU thresholds improves localization quality [37]. For AV purposes, the relevant property of this lineage is not its historical accuracy leadership but its structure: two sequential stages impose a latency floor and an irregular computation pattern that embedded accelerators handle poorly. This is why the included AV studies that adopt two-stage designs almost always target offline analysis or high-compute research platforms rather than in-vehicle deployment [38].

The one-stage lineage traded proposal precision for dense direct prediction. YOLO reformulated detection as a single regression over a grid [8], SSD added multi-scale prediction from feature maps of different depths [9], and RetinaNet showed that the accuracy gap between

one-stage and two-stage detectors was largely an artifact of foreground–background imbalance, correctable by the focal loss [39]. Successive YOLO releases consolidated engineering improvements in augmentation, assignment, and architecture [40, 41]; YOLOv4 systematized these training refinements [42], YOLOv7 advanced the trainable bag-of-freebies [43], and YOLOv9 introduced programmable gradient information to reduce information loss in deep supervision [44]. The lineage is documented in detail by the YOLO survey literature [19]. Anchor-free designs removed dataset-specific anchor tuning by predicting keypoints or centers directly [45, 46].

Why do CNN detectors continue to dominate industrial autonomous-driving deployments despite the benchmark advances of Transformer-based models? The included evidence supports five mutually reinforcing reasons. First, the convolutional inductive bias — locality, translation equivariance, and weight sharing — allows CNNs to reach usable accuracy from the moderately sized, domain-specific datasets that AV teams actually possess, whereas attention-based models are more dependent on large-scale pre-training [47]. Second, a decade of investment has made convolution the most highly optimized operation in embedded inference libraries, with mature INT8 quantization paths [48] and dedicated acceleration on automotive processors [49]. Third, the dense, regular computation of one-stage CNNs is deterministic in both runtime and memory footprint, a property that simplifies the worst-case execution-time analysis required by automotive safety processes. Fourth, deployment toolchains — compilers, operator libraries, calibration tools — support the CNN operator set far more completely than attention operators, so a CNN moves from research to an electronic control unit with less engineering risk. Fifth, and empirically, one-stage CNNs dominate the included studies that report real-time operation on named embedded automotive hardware [50, 51]. These advantages are practical rather than fundamental, but in a safety-regulated industry practical advantages compound.

The same evidence, however, locates three structural weaknesses that no amount of toolchain maturity removes. First, the effective receptive field of stacked convolutions is substantially smaller than the theoretical one, limiting the contextual reasoning available for occluded or distant objects [52]. Feature pyramid networks mitigate the multi-scale problem — and remain the single most important CNN mechanism for the small, distant objects that dominate AV safety cases — but they aggregate context only locally [39, 53]. Second, CNN features degrade under distribution shift: fog, night scenes, and cross-city transfer produce substantial accuracy losses that clear-weather benchmarks never expose [54, 55]. Corruption benchmarking

confirms that this degradation grows with severity and affects every CNN family tested [56]. Third, CNNs trained on closed category sets cannot flag unknown road hazards, a limitation addressed only recently by open-world and open-vocabulary formulations [57, 58]. These weaknesses — not raw benchmark accuracy — are what motivated the Transformer and hybrid developments reviewed next, and they mark the boundary of where CNNs remain the preferred solution: inside a well-characterized operational domain with bounded compute, CNNs are the evidence-backed choice; at the domain's edges — dense occlusion, unknown objects, unfamiliar conditions — their architectural limits become fundamental rather than incremental.

Two further design dimensions of the CNN family carry direct AV consequences and are often glossed over. Anchor design is the first: anchor-based detectors encode dataset-specific priors about object sizes and aspect ratios, and the most controlled included driving study shows that anchors tuned on general benchmarks are mismatched to driving scenes, where distant vehicles and pedestrians concentrate in the smallest anchor bins; re-optimizing anchors on the target driving dataset recovers measurable accuracy at no inference cost [38]. Anchor-free designs remove this tuning burden but relocate the sensitivity into assignment and sampling rules [46]. Automated design is the second dimension: NAS-derived CNNs such as EfficientDet demonstrated that searched feature pyramids and compound scaling produce accuracy-compute frontiers that hand designs did not reach [59]. The AV reading of both dimensions is the same: CNN detector performance on driving data is substantially a function of how well its priors and scale allocation match the deployment distribution — which is exactly the information that single-benchmark evaluations do not reveal. A final, related observation concerns supervised pre-training, which the CNN family inherits almost universally: ImageNet initialization remains the default across the included CNN studies [60, 61], and transfer-learning evidence shows that early convolutional features transfer broadly while later layers specialize [62]. This dependence is benign for benchmark work but becomes a deployment consideration when the target domain — thermal imagery, unusual mounting geometries, non-Western road furniture — diverges from the pre-training distribution; the self-supervised alternatives reviewed in Sect. 5.2 are one response.

3.2 Transformer-Based Detectors

The scientific question that Transformer detection poses for autonomous driving is not which variant tops a leaderboard, but what global attention changes about the detection problem itself. DETR provides the cleanest answer because it changed two things at once: it reformulated

detection as direct set prediction — a fixed set of learned object queries matched to ground truth by bipartite matching, eliminating anchors and non-maximum suppression (NMS) — and it inserted global self-attention between feature extraction and prediction [10]. Two architectural facts about this design are frequently misstated in the survey literature and are worth fixing precisely. First, the original DETR is itself a sequential hybrid: it uses a ResNet CNN backbone rather than raw image patches; pure patch-based backbones arrived later with the vision Transformer (ViT) [47]. Second, DETR's well-known weaknesses — a roughly 500-epoch training schedule and weak small-object accuracy — stem from dense global attention over low-resolution features, not from the set-prediction formulation itself. Fig. 8 depicts the query-based pipeline.

The variant lineage addressed these weaknesses one mechanism at a time, and reading it that way explains what each variant contributes. Deformable DETR replaced dense attention with sparse, multi-scale deformable attention, cutting the training schedule by roughly an order of magnitude and improving small-object accuracy [11]. DN-DETR accelerated convergence by denoising ground-truth boxes during training [63], and DINO combined contrastive denoising with improved anchor initialization to make query-based detectors competitive at the top of the COCO leaderboard [64]. Sparse R-CNN reached a related design point from the CNN side by replacing dense proposals with a small learnable set [65]. In parallel, hierarchical backbones made attention affordable at detection resolutions: Swin Transformer restricts self-attention to shifted windows for near-linear complexity [12], and ViTDet showed that even plain ViT backbones support strong detection with minimal adaptation [66]. RT-DETR then demonstrated that a carefully designed hybrid encoder allows query-based detection to reach real-time desktop-GPU throughput, reporting 53.1% AP at 108 FPS on a T4 GPU in its own evaluation [15].

What does global attention buy for driving scenes? The mechanism is contextual reasoning (Fig. 9): a single self-attention layer can relate a partially occluded pedestrian to the vehicle occluding it, or a small distant sign to the road geometry around it, whereas convolution must construct such relations indirectly across many layers. The included studies support three cautious conclusions rather than a general superiority claim. Query-based detectors show consistent advantages in crowded and occluded scenes, where set prediction avoids the duplicate-suppression heuristics that harm overlapping objects. Their small-object performance depends on multi-scale attention design and remains sensitive to feature resolution. And their costs arrive on exactly the axes that constrain vehicles: attention

consumes memory quadratically in token count, training schedules and hyperparameter sensitivity raise development cost, and inference latency on embedded platforms is penalized by immature operator support. Data efficiency adds a further condition: plain ViT backbones require large-scale pre-training to match CNN inductive biases [47], which matters for AV teams whose labelled driving data is limited (Sect. 5.2).

Why, then, have benchmark improvements not translated into practical automotive deployment? The included evidence identifies three specific obstacles rather than a general immaturity. First, attention operators remain far less mature than convolutions on embedded inference stacks: quantization schemes, kernel fusion, and compiler support for multi-head attention lag years behind their convolutional counterparts, so a Transformer's desktop advantage shrinks or inverts on automotive hardware (Sect. 5.3). Second, the dependence on large-scale pre-training conflicts with domain-specific retraining cadences: 500-epoch schedules are incompatible with the iteration cycle of an AV development pipeline, which is why the denoising and query-initialization advances that cut schedules to tens of epochs [63, 64] are prerequisites for adoption rather than leaderboard details. Third, deployment-oriented validation is simply missing: the included studies that quantize or accelerate DETR-family models on embedded automotive hardware remain rare, so the family's readiness cannot yet be certified from published evidence. The deployed adaptations that do exist follow a consistent pattern — multi-scale or windowed attention instead of dense attention [11, 12], and hybrid encoders that reserve attention for low-resolution stages [15] — because driving scenes span object scales from full-frame trucks to ten-pixel pedestrians and input resolution is the cost driver of quadratic attention. The conclusion the evidence supports is therefore conditional: Transformer superiority is real where global context dominates and compute is available, but it remains contingent on deployment constraints rather than absolute, and the unresolved question — whether efficiency-oriented attention variants preserve the global-context advantage that motivated Transformers in the first place — is not yet directly measured by any included study.

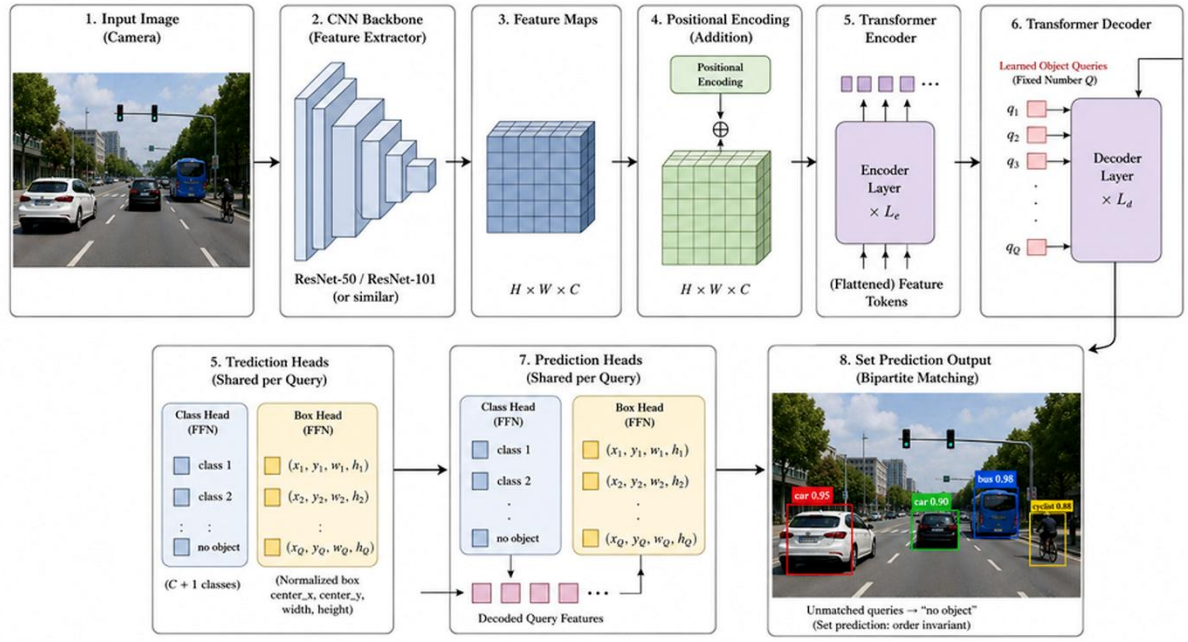


Fig. 8 Query-based end-to-end detection pipeline of the DETR family. The original DETR uses a CNN backbone; later variants replace dense attention with sparse or windowed alternatives.

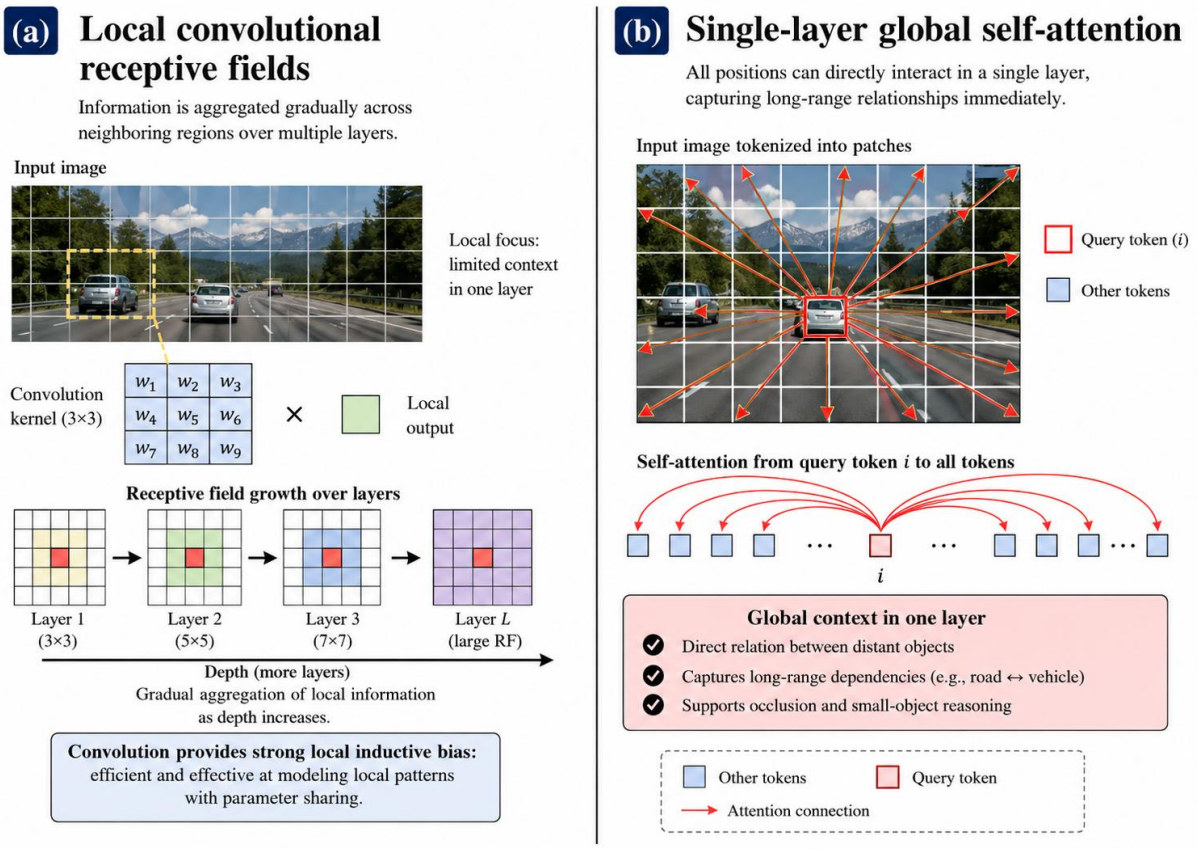


Fig. 9 Local convolutional receptive fields (a) versus single-layer global self-attention (b): the mechanism that motivates Transformer and hybrid detectors. Conceptual schematic.

3.3 Hybrid and Multimodal Detectors

Hybrid architectures are best understood not as one family but as three distinct answers to the same design question: how should local feature extraction and global contextual reasoning share a computation budget? Classifying them by where integration occurs makes the answers explicit (Fig. 10; Table 4). Sequential integration passes CNN features into Transformer reasoning — the DETR pattern, and equally the pattern of a hierarchical Transformer backbone under a CNN detection head. Its motivation is economy: convolution handles the high-resolution stages where attention is unaffordable, and attention reasons over the compact features that remain. Parallel dual-stream integration runs convolutional and attention branches side by side and fuses them by concatenation, gating, or cross-attention; its motivation is preservation, keeping both local texture and global context alive at full fidelity rather than forcing one through the other. Interleaved integration alternates convolution and attention within a single backbone, as in CMT, which reported ImageNet and COCO gains at mobile-scale compute by placing lightweight attention between convolutional stages [13]; its motivation is granularity, letting local and global reasoning exchange information at every depth. Fig. 11 places these fusion options along the detection pipeline — backbone, neck, head — with their design consequences: earlier fusion preserves more information but pays the attention cost at high resolution, while later fusion is cheaper but limits the interaction between local texture and global context.

Each strategy also carries a characteristic cost profile that benchmark tables do not show. Sequential designs are the cheapest to build and train, but they create an information bottleneck: whatever the convolutional stage discards, the attention stage cannot recover, and no included study quantifies this loss for driving-relevant small objects. Parallel designs preserve both representations but duplicate computation and introduce a fusion module whose tuning is delicate; ablations rarely isolate how much of the reported gain survives when the two streams share a backbone. Interleaved designs reach the best accuracy-compute frontiers but are the least hardware-friendly: alternating operator types defeats the operator-fusion optimizations that inference engines rely on, and the compression evidence of Sect. 5.3 indicates that these models are currently the hardest to quantize. Optimization difficulty compounds the picture, because hybrid models inherit the training sensitivities of both parents — augmentation and assignment tuning from the CNN side, schedule and initialization sensitivity from the attention side.

The benchmark evidence for hybrid advantages is positive but conditional, and it is presented here with its conditions attached. On COCO-style benchmarks, interleaved and sequential

hybrids consistently improve the accuracy-compute frontier relative to CNN-only baselines of similar size [13, 16]. On driving data, the included studies report advantages concentrated in dense urban scenes, occlusion, and small objects — the conditions where global context plausibly helps — but these studies vary widely in datasets, input sizes, and hardware, so the review treats the pattern as consistent rather than quantified [17, 18]. Multimodal fusion extends hybridization across sensors and is retained here as context rather than core synthesis: TransFusion uses cross-attention to associate camera features with LiDAR proposals robustly under degraded image quality [67]; BEVFormer builds a bird's-eye-view representation from multi-camera input via spatiotemporal attention [14]; and BEVFusion unifies camera and LiDAR features in a shared BEV space for multi-task perception [68]. For the deployment question of this review, the relevant point is that these systems inherit the computational profile of their attention mechanisms, and their reported latencies are measured almost exclusively on high-end GPUs; their embedded feasibility remains an open evidence gap.

Whether hybrids are the natural successor to both parent families — an assumption now common in the survey literature — is not yet supported by the included evidence, and this review treats it as an open question rather than a conclusion. Three observations justify the caution. First, the benchmark gains of hybrids are demonstrated predominantly on general-purpose datasets under research training budgets; the driving-specific evidence is thinner and heterogeneous. Second, the deployment costs of hybrids are systematically under-measured: models that are harder to compress, quantize, and schedule on embedded accelerators may forfeit on the vehicle the margin they earn on the benchmark, and several included studies note exactly this difficulty (Sect. 5.3). Third, no included study independently replicates a proprietary hybrid AV detector, which leaves the strongest industrial claims unverifiable. The practical conclusion is conditional: hybrids are advantageous where dense-scene accuracy is the binding constraint and the platform provides headroom for their complexity, while for tightly compute-bounded deployments the additional fusion machinery can outweigh its benefit. Stating the open questions — bottleneck quantification for sequential designs, redundancy-cost isolation for parallel designs, quantization behaviour for interleaved designs — is part of the taxonomy's purpose: they define what an evaluation must measure before "hybrid" can function as a deployment recommendation rather than a design aesthetic.

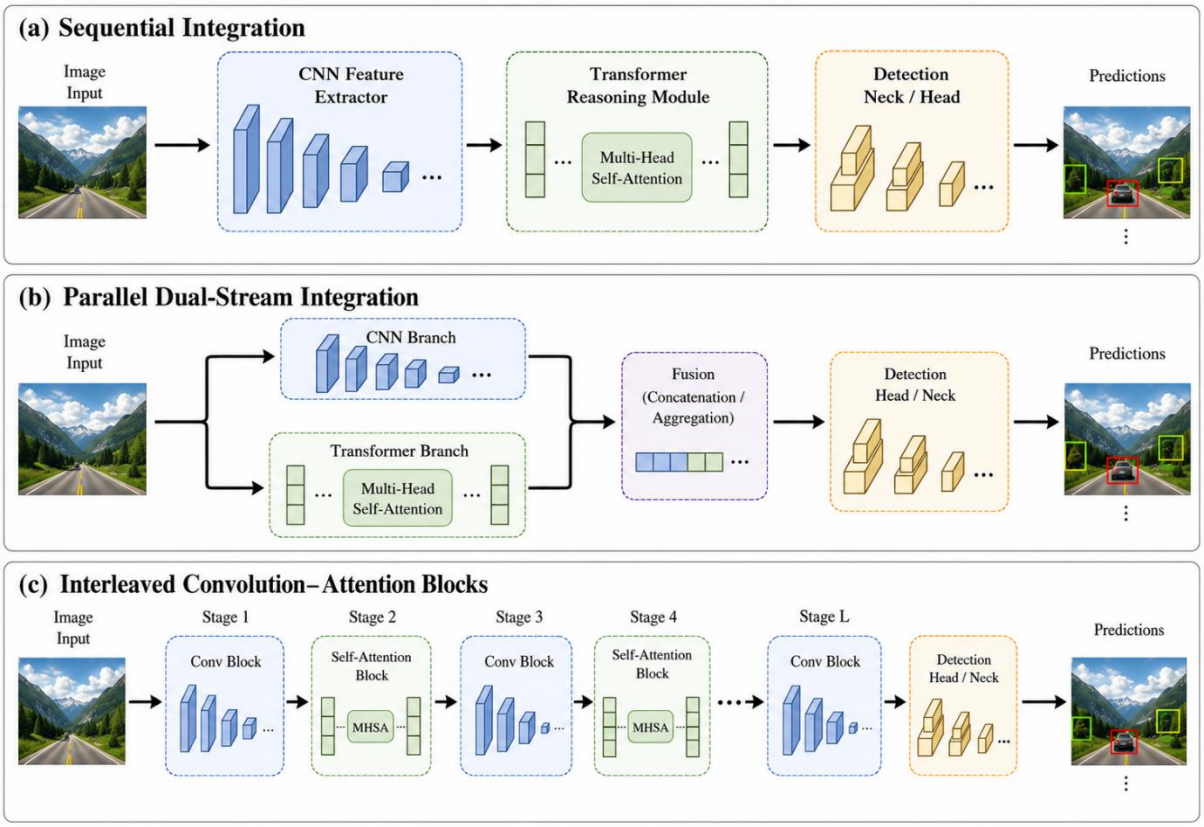


Fig. 10 Hybrid CNN-Transformer integration strategies: (a) sequential, (b) parallel dual-stream, and (c) interleaved convolution-attention blocks. Conceptual schematics.

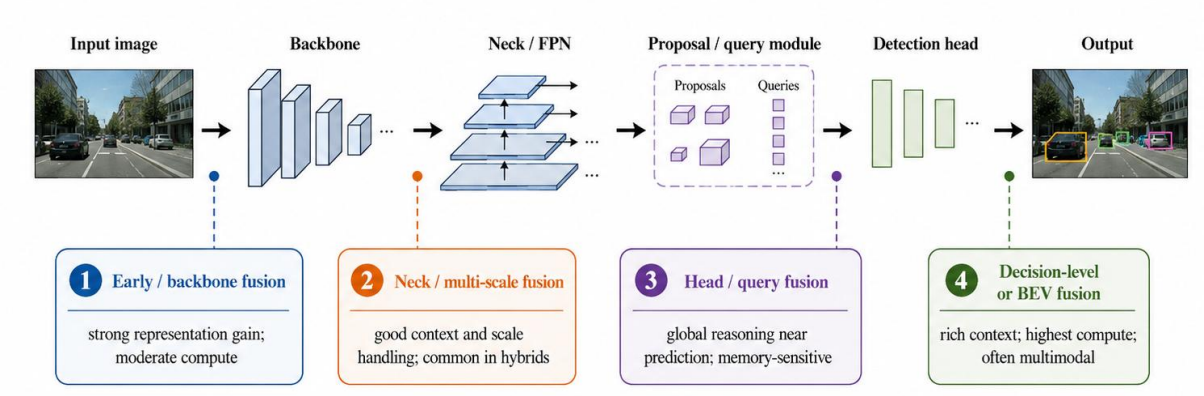


Fig. 11 Taxonomy of local-global feature-fusion locations along the detection pipeline and their design consequences.

4. Critical Comparative Synthesis of CNN, Transformer, and Hybrid Detectors

4.1 Accuracy–Efficiency Trade-offs Under Comparability Rules

Table 5 assembles the benchmark evidence that meets at least the dataset-level comparability rule of Sect. 2.3: all values are COCO results reported by the original papers, with hardware and precision stated wherever the sources state them. Even so, the table must not be read as a

ranking. Its rows span nine years of training practice, different input resolutions, different dataset splits, and different measurement hardware. The table exists to make those differences explicit — the very differences that most published comparisons leave silent.

Read under these cautions, the evidence supports three findings that are more instructive than any ordering. First, the CNN family advanced enormously between 2016 and 2024: SSD512 reported 26.8% AP in 2016 [9], whereas YOLOv7 reported 51.4% AP at 161 FPS on a V100 in its own evaluation [43] — nearly double the accuracy at high throughput within eight years. Second, the query-based family closed its efficiency gap: the original DETR matched a comparably re-trained Faster R-CNN in accuracy and V100 throughput but needed a 500-epoch schedule [10], whereas RT-DETR reports higher accuracy than contemporary YOLO variants at real-time throughput on a weaker T4 GPU [15]. Third, peak accuracy still belongs to large hybrid-backbone query detectors: DINO with a Swin-L backbone reaches 63.3% AP on COCO test-dev [64], yet no included study deploys a model of this class on automotive hardware. The correct synthesis is therefore positional, not ordinal: the families occupy different regions of the accuracy-latency frontier, and a frontier position measured on a desktop GPU does not transfer to embedded platforms, where operator support and memory bandwidth reshape the frontier itself (Sect. 5.3).

Why do these positions persist? CNNs dominate latency-sensitive deployment because everything about them — dense regular computation, mature INT8 paths, complete operator support — aligns with how embedded inference hardware is built, so their frontier position survives the transfer from desktop to vehicle. Transformers hold the advantage where scenes demand relational reasoning — crowded intersections, occlusion chains — because global attention supplies exactly the context that convolution must approximate. Hybrids sit between the two by construction, inheriting a portion of each parent's strengths and costs in proportion to where they place attention. These are architectural explanations, not benchmark artifacts, which is why they recur across otherwise incomparable studies.

The AV-dataset evidence adds a layer that COCO results cannot: the interaction between detector design and driving-data characteristics. The most controlled included comparison trained one-stage and two-stage detectors under a common protocol on Waymo camera data and found that the families trade accuracy against speed largely as their COCO reputations suggest — but that transfer-learning strategy and anchor optimization changed the ranking within families, a result invisible to leaderboard comparisons [38]. Driving-specific YOLO modifications report consistent gains over their baselines on KITTI and BDD100K under the

authors' own matched training [50, 51], and an uncertainty-aware YOLOv3 variant improved accuracy while reducing false positives at real-time rates [69]. The review deliberately reports these as within-study findings: each is internally controlled and externally incomparable, which is precisely the evidence structure the comparability protocol is designed to respect.

A methodological finding belongs alongside the numerical ones: FPS values are the least comparable numbers in the detection literature. Across the included studies, frame rates are reported variously at batch size 1 or 32; in FP32, FP16, or INT8; with or without pre- and post-processing; and on hardware ranging from embedded modules to data-center GPUs — frequently without stating which. Any review, including this one, can compare such values only within a single study's controlled evaluation. The field's habit of quoting them across studies is a primary source of unsupported deployment claims.

Table 5 Benchmark evidence as reported by the original papers on COCO. Values are NOT mutually comparable across rows: dataset split, input size, hardware, precision, batch size, and training practice differ. The table exists to make those differences explicit (Sect. 2.3)

Model (source)	Family	Split	Input	Reported AP (%)	Hardware (as reported)	Reported speed	Params	Comparability note
SSD512 [9]	One-stage CNN	test-dev	512	26.8	Titan X	— (speed reported on VOC only)	—	2016 training practice
RetinaNet R101-FPN [39]	One-stage CNN	test-dev	800	39.1	—	—	—	No speed at this setting
CenterNet DLA-34 [46]	Anchor-free CNN	val	512	37.4	Titan Xp	52 FPS	—	Preprint
EfficientDet D0–D7 [59]	NAS CNN	test-dev	512–1536	33.8–52.2	Titan V / V100	varies by scale	3.9–52 M	Family, not one model
YOLOv4 [42]	One-stage CNN	test-dev	608	43.5	V100	≈65 FPS	—	Preprint
Faster R-CNN R50-FPN, 3× (as re-trained in [10])	Two-stage CNN	val	800	40.2	V100	26 FPS	42 M	Baseline within DETR study

DETR-R50 [10]	Query-based	val	800	42.0	V100	28 FPS	41 M	500-epoch schedule
Deformable DETR [11]	Query-based (sparse)	val	800	43.8	V100	19 FPS	40 M	50-epoch schedule
YOLOv7 [43]	One-stage CNN	test-dev	640	51.4	V100, batch 1	161 FPS	36.9 M	—
YOLOv9-C [44]	One-stage CNN	val	640	53.0	—	—	25.3 M	No speed in paper table
RT-DETR-R50 [15]	Real-time query hybrid	val	640	53.1	T4, TensorRT FP16	108 FPS	42 M	Weaker GPU than V100 rows
DINO, Swin-L [64]	Query hybrid (large)	test-dev	multi-scale	63.3	—	—	—	Peak accuracy; no deployment data

4.2 Small-Object and Occlusion Performance

Small object size and long range compound the safety risk of AV detection: a pedestrian at 80 m occupies a few dozen pixels, and the difference between detection at 80 m and at 40 m is braking distance. The included evidence shows that this is where architectural differences matter most. For CNN detectors, the challenge is scale coverage, and dedicated small-object modifications — higher-resolution pyramid levels, customized assignment, slicing inference — deliver consistent improvements on driving data [50, 70]. Searched feature-pyramid topologies extend the same mechanism further [53]. The query-based family inherited a documented small-object deficit from the original DETR [10], which multi-scale deformable attention substantially remedied [11]; subsequent denoising and query-refinement variants continued the recovery [63, 64].

For occlusion and crowding, the evidence favours global-context models more decisively. Set prediction eliminates NMS, a post-processing step that systematically discards correct detections of heavily overlapping objects, while attention allows hypotheses for occluded objects to be confirmed by visible context. Dense urban scenes are accordingly where the included studies report the largest Transformer and hybrid margins [16, 18]. Two qualifications

bound this conclusion. Occlusion-specific analysis appears in only about 25% of the included studies (Fig. 5c), and standard mAP averages over occlusion levels in a way that can mask safety-relevant failures; the KITTI difficulty tiers remain the only widely used occlusion-stratified protocol [31].

4.3 Cross-Dataset Generalization and Real-Time Feasibility

Only 15% of the included studies validate on a dataset different from the training distribution, and those that do report substantial degradation: models tuned on one benchmark lose accuracy when transferred across cities, sensors, or annotation policies, with domain-adaptive training recovering only part of the loss [33, 55]. No included study demonstrates that any architecture family is systematically more robust to cross-dataset transfer at matched accuracy. Claims that Transformers generalize better rest mainly on classification-domain evidence and on adversarial-perturbation studies whose conclusions are mixed [71, 72]. The review therefore treats cross-dataset generalization as an open empirical question rather than an architectural property.

On real-time feasibility, the evidence supports a deliberately narrow conclusion. Mature one-stage CNNs are the only family for which multiple independent included studies demonstrate real-time operation on named embedded automotive hardware with stated precision [50, 51]. Query-based and hybrid detectors have demonstrated real-time desktop throughput [15] and are beginning to appear in embedded evaluations, but the embedded evidence base is currently too thin — 13% of studies overall, and disproportionately CNN-based — to certify them as deployment-ready. This is an evidence gap, not a verdict: the gap itself is the finding, and Sect. 7 converts it into a benchmarking requirement.

"Real-time" itself requires definition before it can serve as evidence. A vehicle-level latency budget covers capture, pre-processing, inference, post-processing, and hand-off to tracking. Most reported FPS values cover inference alone, and some exclude NMS — a meaningful cost for dense one-stage detectors and absent by construction in set-prediction models, which complicates even this comparison. At highway speeds, the difference between a 30 ms and a 100 ms perception cycle is measured in metres of travelled distance, so end-to-end measurement on the target platform is the only defensible basis for a real-time claim. The included studies that meet this standard are almost all CNN-based; extending the same discipline to Transformer and hybrid detectors is a precondition for the family-level comparison the field currently lacks.

5. Robustness, Data Efficiency, and Deployment Readiness

5.1 Adverse Conditions and Adversarial Robustness

The reviewed literature frequently treats "robustness" as a single property. The extracted evidence requires distinguishing four failure sources with different mechanisms, different evaluation protocols, and different defenses (Fig. 12): natural corruption, domain shift, adversarial perturbation, and open-world objects. Conflating them produces the most common category of unsupported claim found during extraction — a defense against one source cited as evidence of "robustness" in general.

Natural corruption is the best-quantified source. Corruption benchmarking shows that standard detectors lose a large fraction of their clean-data accuracy under simulated fog, snow, blur, and noise, with degradation growing monotonically with severity [56, 73]. The driving-specific evidence is consistent: synthetic fog augmentation, image restoration, and combined synthetic-plus-real training each recover only part of the fog-induced loss [54, 74], and recent weather-robustness pipelines report the same partial-recovery pattern [75]. Yet only 19% of the included studies evaluate any adverse condition, and fewer still use condition-controlled data such as ACDC's paired clear/adverse references [36]. Domain shift is mechanistically different: the input looks clean, but the distribution has moved — new cities, new cameras, new seasons — and adaptation methods again recover only part of the loss [55].

Adversarial robustness concerns deliberately crafted inputs. Digital attacks on detectors are long established [76], and physically realizable attacks — printed patches, perturbed traffic signs — have repeatedly defeated detection models in navigation contexts [77, 78]. A recent survey with standardized evaluation shows that attack success varies strongly across detector families and attack budgets, that no family is uniformly robust, and that a vision-Transformer robustness advantage reported under some attacks disappears once the attacks are adapted to the architecture [71, 72]. On the defense side, adversarial training remains the only broadly effective empirical defense, and certified defenses for detection — provable patch-attack guarantees and randomized-smoothing variants — currently cover limited threat models at a meaningful accuracy cost [79, 80]; masking-based certified defenses share the same trade-off [81]. Formal verification of full detection networks remains out of reach: existing methods handle small networks and restricted properties [82, 83]. The practical conclusion drawn across

the included safety literature is defense in depth — adversarial training where affordable, runtime monitoring of detector behaviour [84], and sensor redundancy with defined fallback behaviour within an automotive safety framework [85].

Measuring robustness requires metrics beyond clean mAP, and the included studies use them inconsistently. The most interpretable pattern reports paired values — clean mAP and mAP under the stated perturbation or condition — from which relative degradation follows directly; corruption benchmarks aggregate this over corruption types and severities [56, 73]. Adversarial evaluations additionally require the attack, its perturbation budget, and the threat model (digital or physical, white-box or transfer) to be stated, because robustness numbers are meaningless across different attack strengths [72]. The review found multiple studies quoting single robustness percentages with none of this context; such values were excluded from the synthesis, and the corresponding reporting requirement enters the agenda in Sect. 7.

Open-world failure — hazards outside the training vocabulary — is the least evaluated source in the included studies. Open-world detection formalizes the requirement to flag unknown objects and learn them incrementally [57], anomaly-detection surveys map the emerging methods [86], and open-vocabulary detectors grounded in vision-language pre-training now detect categories specified by text at real-time rates [58, 87]. The AV evidence for these methods consists mainly of benchmark transfers rather than driving-scene validations, and their reliability calibration on safety-critical unknowns is untested — an agenda item in Sect. 7.

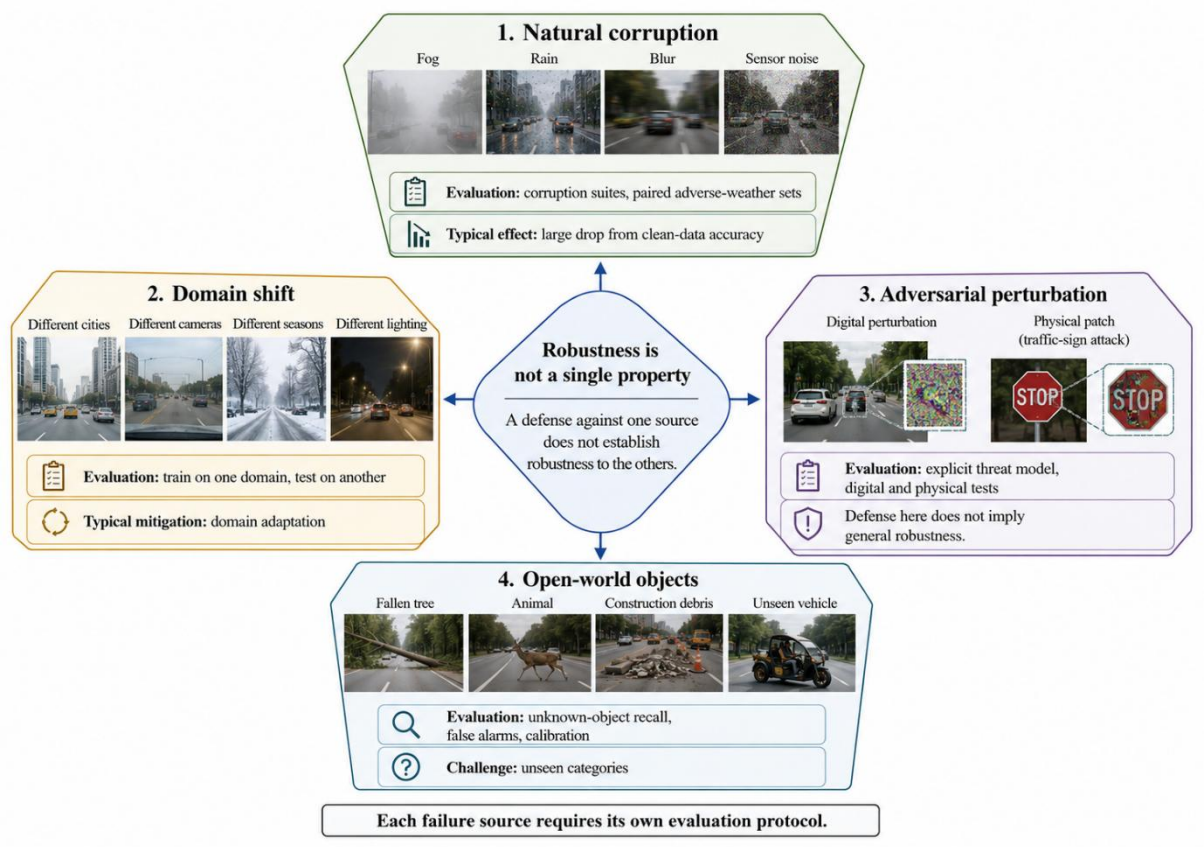


Fig. 12 Four distinct failure sources for AV object detection. Each requires its own evaluation protocol; a defense against one source does not establish robustness to the others.

5.2 Self-Supervised and Data-Efficient Learning

Annotation cost is a structural constraint for AV perception: fleets record vastly more data than can be labelled, and even the large public datasets cover a narrow slice of operating conditions [34, 35]. SSL addresses this by pre-training representations on unlabelled data before detection fine-tuning (Fig. 13). The general-vision evidence is mature. Contrastive methods [88, 89], self-distillation methods [90, 91], and masked-reconstruction methods — including their extension to temporal data [92, 93] — all produce backbones whose detection fine-tuning matches or exceeds supervised ImageNet initialization, with the largest relative gains when labelled data is scarce. Detection-specific pre-training objectives improve transfer to localization tasks over image-level objectives [94], and multimodal extensions align image and point-cloud representations [95].

The AV-specific evidence is thinner and more conditional than the enthusiasm around SSL suggests, and the review states its boundaries explicitly. Included driving studies report label-efficiency gains and improved transfer under domain shift — including a self-supervised domain-adaptation approach for thermal detection that improved performance without target-

domain labels [96] — but systematic comparisons on driving data under matched fine-tuning schedules are rare, and a recent survey of SSL for real-world detection concludes that benefits on rare classes and adverse conditions remain under-evaluated [97]. Only 10 of the 142 included studies (7%) make SSL or data efficiency their primary contribution (Fig. 5b). The reviewed evidence therefore supports SSL as a low-risk initialization improvement with the strongest returns in low-label regimes, while the stronger claims — robustness gains, rare-event coverage, continual on-vehicle adaptation — remain hypotheses awaiting AV-domain validation. Continual-learning safeguards against catastrophic forgetting are a prerequisite for any on-vehicle adaptation, and the systematic evidence for them exists in the classification domain but not yet in AV detection [98].

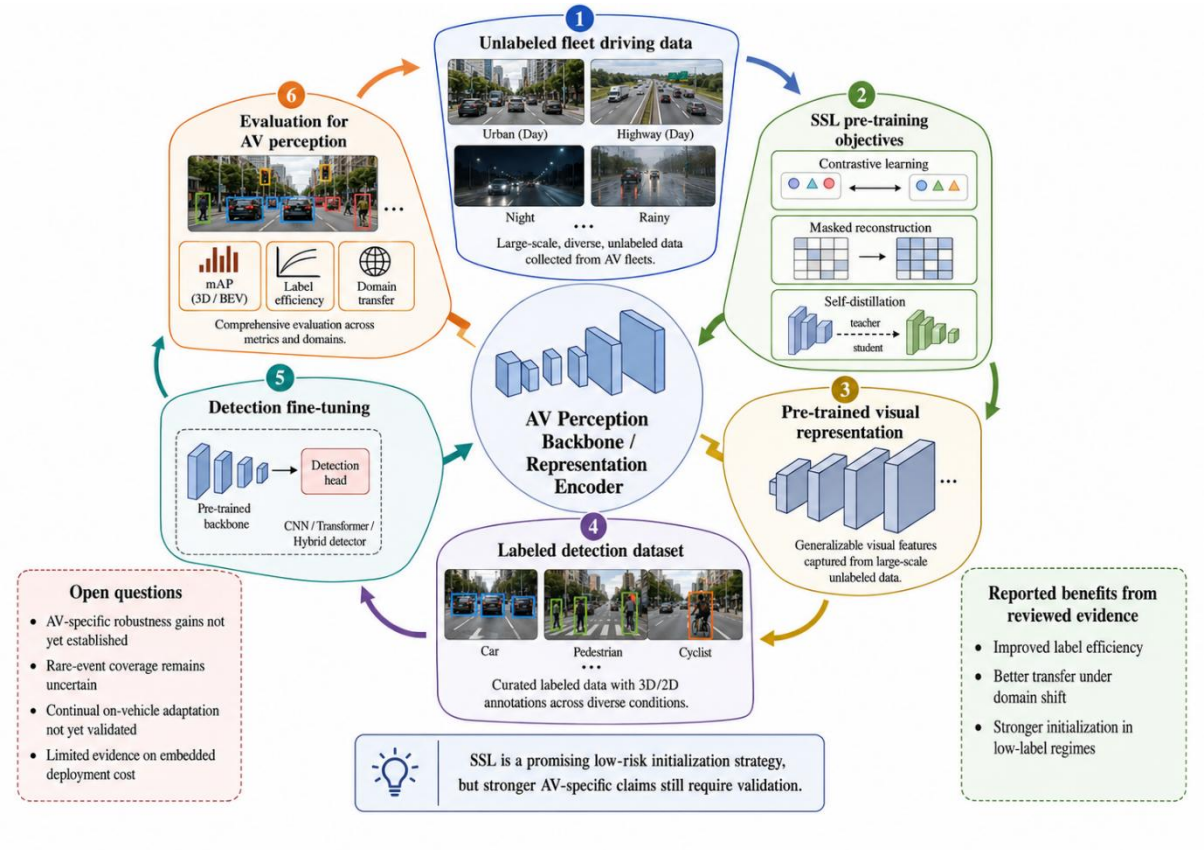


Fig. 13 Self-supervised pre-training and detection fine-tuning pipeline for AV perception, with reported benefits and open questions from the reviewed evidence.

5.3 Edge Optimization, NAS, and Hardware-Aware Design

Deployment readiness ultimately reduces to whether a detector meets its latency, memory, and power budget on the vehicle's actual hardware. The compression toolkit is well established in principle: mobile-oriented backbone design [99], magnitude pruning and weight sharing [100], integer quantization with quantization-aware training [48], knowledge distillation, and

precision-exploiting accelerators [49, 101]. Framework-level studies show that the inference stack itself changes latency by integer factors independently of the model [102]. Two evidence-based cautions apply. First, parameter and FLOP counts are unreliable proxies for latency: several included studies report models with fewer FLOPs running slower on target hardware because of memory-bound operators — attention among them — so only measured end-to-end latency on the named device supports a deployment claim. Second, quantization evidence for Transformer and hybrid detectors on automotive accelerators is scarce in the included corpus; the reviewed INT8 results concentrate on CNN detectors, leaving the deployment cost of attention under low precision an open question.

Knowledge distillation and deployment engineering complete the compression picture. Distillation transfers the behaviour of a large teacher detector to a budget-friendly student, and its effect compounds with pruning and quantization; the included AV studies use it primarily to preserve small-object accuracy that naive shrinking would destroy [51]. On the engineering side, operator fusion, kernel selection, and precision planning change measured latency substantially for identical weights [102]. The methodological parallel for research reporting is direct: a latency claim without the runtime engine and version is not a complete latency claim.

NAS automates architecture design under such constraints. For detection, NAS-FPN searched the feature-pyramid topology [53], DetNAS searched the backbone [103], SM-NAS searched the stage-wise combination of detector parts [104], EAutoDet made detector search affordable [105], and EfficientDet co-designed a scalable family covering a wide accuracy-compute range [59]. Robustness-aware variants add adversarial and out-of-distribution objectives to the search [106, 107], and the general foundations are surveyed elsewhere [108, 109]; transferable-architecture search extended the paradigm to large-scale vision [110]. The crucial point for AV deployment is that NAS adds value strictly in proportion to the fidelity of its objective. Searches driven by proxy metrics — FLOPs, parameters — inherit the proxy weaknesses noted above, whereas searches that measure latency on the target device (hardware-in-the-loop) transfer to deployment. Few included studies meet that standard, and none reports energy as a search objective, although power is a first-order constraint in production vehicles.

Uncertainty calibration belongs in this section because it is a deployment mechanism, not an accessory. Modern networks are systematically overconfident [111]; detection uncertainty can be decomposed and used at runtime [3, 112]; and the one included study that integrated localization uncertainty directly into a real-time AV detector reported accuracy gains of about three mAP points on KITTI and BDD alongside reduced false positives, all at real-time rates

[69]. That only 6% of the included studies report any uncertainty estimation (Fig. 5c) is among the clearest gaps this review identifies, because calibrated confidence is what allows a planning stack to respond safely to detector doubt [113], and interpretability methods for understanding failures presuppose it [114].

6. Evidence Gaps and the Proposed AVOD Deployment-Readiness Framework

The reporting profile of Fig. 5c defines the problem that the framework solves: most studies do not report the information a deployment decision requires, and the information they do report is scattered across incompatible protocols. The AV Object Detection Deployment-Readiness Framework (AVOD-DRF) makes that information explicit. It profiles a detector — or, more precisely, the evidence reported for a detector in a study — on eleven dimensions: accuracy on driving data, small-object sensitivity, inference latency on named hardware, hardware dependency (precision, runtime, batch size), parameter and memory load, robustness to weather and illumination, resistance to occlusion, resistance to domain shift and adversarial perturbation, cross-dataset generalization, uncertainty reporting and calibration, and reproducibility (code, weights, configuration). These dimensions are not arbitrary: each corresponds to an independent failure channel documented in the synthesis of Sects. 3–5, and together they map onto the questions an operational design domain (ODD) poses to a perception system — what must be detected (accuracy, small objects), under which conditions (weather, illumination, occlusion), across which environments (domain shift, cross-dataset), within which computational budget (latency, hardware, memory), with what assurance (uncertainty, adversarial resistance), and with what verifiability (reproducibility). Deployment readiness is conjunctive across these channels: high benchmark accuracy cannot compensate for missing latency evidence, and fast inference cannot compensate for untested robustness. This is the theoretical reason the framework scores dimensions separately and never sums them into a single number.

The scoring methodology follows the quality rubric of Sect. 2.2. Each dimension is scored 0 when the evidence is not reported, 1 when it is partially demonstrated, and 2 when it is comprehensively demonstrated against the anchors fixed before extraction (Table 3). The three-level scale is deliberate: finer scales would imply a precision that heterogeneous published evidence cannot support. Two properties distinguish AVOD-DRF from the impressionistic scoring criticized elsewhere in this review. First, "not reported" never equals

"poor": a 0 records missing evidence and asserts no failure, which protects benchmark-focused studies from being penalized for questions they never posed. Second, every non-zero score must cite the extracted evidence supporting it, which makes the profile an audit of what is known rather than an opinion about what is likely.

Score profiles acquire scientific meaning through their patterns, and three recurring patterns function as readiness levels. A benchmark-grade profile — one or two accuracy dimensions at 1 or 2 and zeros elsewhere — characterizes a study whose contribution is architectural, and it marks the distance between that contribution and any deployment claim. A development-grade profile adds partial deployment evidence: desktop latency with a stated protocol, a single adverse-condition test, or a second dataset. A deployment-grade profile requires every safety-relevant dimension to be non-zero and the constraint dimensions — latency, hardware dependency, robustness in the intended conditions — to be comprehensively demonstrated on named automotive or edge hardware. Applied at the family level, the framework summarizes the synthesis of Sects. 3–5 without overclaiming: mature one-stage CNNs score highest on latency, hardware dependency, and reproducibility, and lowest on occlusion and open-world dimensions; query-based Transformers and hybrids score highest on occlusion and peak accuracy, but their edge-latency, quantization, and energy evidence is largely missing rather than negative. No family reaches comprehensive demonstration on adverse weather, cross-dataset generalization, uncertainty, or energy — these are field-level gaps rather than model-level failures.

A worked example shows the audit in practice. Consider a study reporting a hybrid detector at 52% AP and 90 FPS on COCO. Accuracy scores 1 (benchmark evidence only, no driving data). Latency scores 1 if the 90 FPS was measured on a desktop GPU with a stated protocol, 0 if the measurement conditions are unstated, and 2 only if an automotive or edge device is named with resolution, precision, batch size, and runtime. Weather, occlusion, domain shift, and cross-dataset each score 0 unless specifically evaluated, and reproducibility follows code and configuration availability. The resulting profile — typically two or three 1s above a column of 0s — is not a criticism of the study, whose contribution may be legitimately architectural; it is a precise statement of the distance between that contribution and a deployment claim. Aggregating such profiles across the 142 included studies produces the field-level gap quantification of Fig. 5c, and keeping the profiles disaggregated prevents the framework from collapsing into the unexplained heatmap scores this review set out to replace.

AVOD-DRF differs from existing benchmark comparisons in kind, not merely in coverage. A benchmark comparison asks which model achieves the highest metric under one protocol; the framework asks whether the evidence exists to trust a model under a stated ODD, and it treats the absence of evidence as a first-class finding. It is also reusable beyond this review. Because scoring requires only the reported evidence of a study — not re-running experiments — future researchers can profile new detectors as they are published, AV teams can compare candidate detectors against the requirements of their ODD, reviewers can audit the deployment claims of submitted papers, and benchmark designers can identify which dimensions their protocols leave unmeasured. The framework is offered in this spirit: not as a scoring system built for one review, but as a standing methodology for converting heterogeneous detection literature into deployment-relevant evidence.

7. Research Agenda for Trustworthy AV Object Detection

Table 6 lists the agenda. This section explains its six priorities; each is grounded in a measured gap in the evidence base, and each is paired with a validation protocol precise enough to be falsifiable.

The first priority is standardized adverse-condition benchmarking. Weather degrades accuracy for every family, but with only 19% of studies testing any adverse condition — and few using condition-controlled data — the field cannot quantify by how much. Protocol: evaluate on matched clear/adverse pairs as provided by ACDC [36], report per-condition mAP alongside clean mAP, and include the compound conditions (night plus rain) that current suites omit [4].

The second is hardware-normalized latency and energy reporting. With 13% edge-latency and 5% energy coverage, "real-time" claims remain unverifiable. Protocol: report device, input resolution, numerical precision, batch size, runtime engine, end-to-end latency including pre- and post-processing, peak memory, and average power on at least one named automotive or edge platform [49, 102]; treat desktop-GPU FPS as a development metric only.

The third is cross-dataset and cross-geography validation. With 15% coverage, generalization claims rest on single-benchmark evidence. Protocol: train on one driving dataset, test unmodified on at least two others with documented annotation-policy mapping, and report the degradation explicitly rather than only the adapted result [33, 55].

The fourth is open-world and vision-language detection for unknown hazards. Closed-set detectors cannot flag what they were never taught [57], and open-vocabulary models have not been stress-tested on driving-specific unknowns [58]. Protocol: evaluate unknown-object recall and false-alarm rate on held-out hazard categories in driving scenes, with calibration analysis of the unknown-object scores [86, 87].

The fifth is uncertainty-aware and certifiable detection. With 6% uncertainty coverage, planners currently receive uncalibrated confidences [3]. Protocol: report calibration metrics (for example, expected calibration error) alongside mAP [69, 112]; for safety cases, state the threat model and the certified guarantee explicitly, in the style of the patch-defense literature [79].

The sixth is continual and fleet learning with forgetting safeguards. On-vehicle adaptation is repeatedly proposed and never validated in the included corpus. Protocol: sequential-domain training with held-out earlier-domain evaluation to quantify forgetting [98], privacy constraints stated, and drift monitoring defined before deployment [97].

Fig. 14 assembles these elements into the evidence-to-deployment roadmap: the ODD determines the candidate architecture family; the validation band defines the evidence that must exist before a deployment claim; and the agenda defines what the field must build so that such evidence becomes routine.

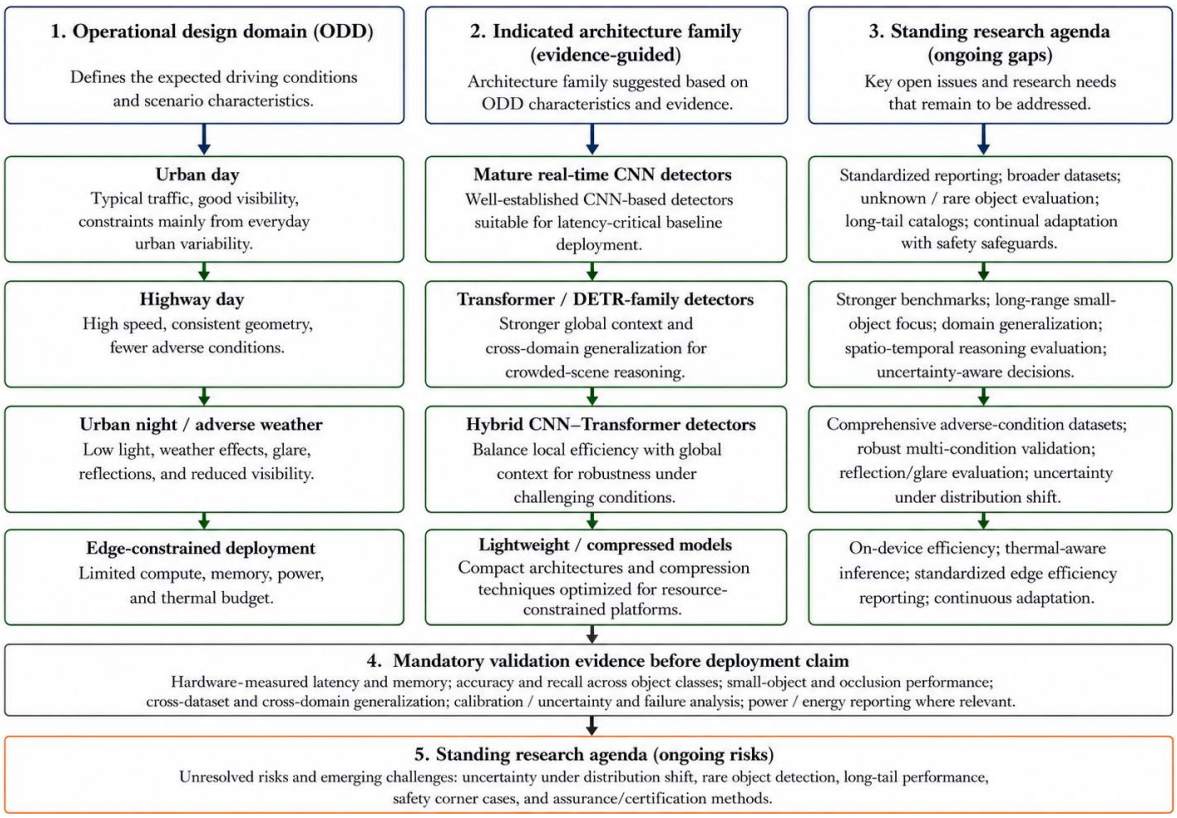


Fig. 14 Evidence-to-deployment roadmap linking the operational design domain, the indicated architecture family, mandatory validation evidence, and the standing research agenda.

Table 6 Evidence gaps and research agenda. Coverage percentages refer to the 142 included studies (Fig. 5c)

Priority	Evidence gap (coverage)	Why it matters	Required development	Validation protocol
1. Adverse-condition benchmarking	19% test any adverse condition	Weather-driven failures are silent on clear benchmarks	Matched clear/adverse test pairs; compound conditions (night + rain)	Per-condition mAP beside clean mAP on paired data (e.g., ACDC)
2. Hardware-normalized reporting	13% edge latency; 5% energy	"Real-time" claims currently unverifiable	Standard reporting: device, resolution, precision, batch, runtime, end-to-end latency, memory, power	At least one named automotive/edge platform per deployment claim
3. Cross-dataset validation	15% validate across datasets	Single-benchmark tuning inflates generalization	Train-on-one / test-on-others protocols with annotation mapping	Unmodified transfer to ≥2 driving datasets; degradation reported
4. Open-world / vision-language detection	Rarely evaluated on driving unknowns	Closed sets cannot flag unknown hazards	Driving-specific unknown-object benchmarks	Unknown-object recall and false-alarm rate with calibration analysis
5. Uncertainty and certification	6% report uncertainty	Planners need calibrated confidence; safety cases need guarantees	Calibrated detection outputs; certified defenses with stated threat models	Calibration metrics beside mAP; explicit certified-guarantee statements

6. Continual / fleet learning	Not validated in included corpus	Static models degrade as the world drifts	Forgetting-safe adaptation; privacy-preserving fleet updates	Sequential-domain training with held-out earlier-domain evaluation

8. Conclusion

This review asked how close CNN, Transformer, and hybrid object detectors are to robust, real-time, deployment-ready perception for autonomous vehicles, and answered against a PRISMA-guided evidence base of 142 studies searched to 30 June 2026. The architectural answer (RQ1) is that the families have converged structurally: modern high-performing detectors are hybrids in fact if not in name, differing in where they place attention relative to convolution rather than in whether they combine them. The trade-off answer (RQ2) is that comparable evidence supports different frontier positions rather than a ranking — one-stage CNNs hold the embedded-latency frontier, query-based and hybrid designs hold the accuracy and crowded-scene frontier, and real-time DETR variants have narrowed the gap on desktop hardware — while most cross-study comparisons in the literature fail basic comparability checks. The robustness answer (RQ3) is that all families degrade substantially under adverse conditions, domain shift, and adapted adversarial attacks; no family owns robustness, and the evidence is too sparse and heterogeneous to support architectural generalizations. The enabling-technique answer (RQ4) is that SSL, NAS, and compression each deliver documented value under specific conditions — low-label regimes, hardware-in-the-loop objectives, and CNN-family quantization respectively — and undocumented value outside them. The gap answer (RQ5) is quantified throughout: 13% edge-latency, 19% adverse-condition, 15% cross-dataset, 6% uncertainty, and 5% energy reporting across the included studies.

For practitioners, the deployment guidance is conditional by design. Restricted-compute highway operational domains are best served today by compact, quantized one-stage CNNs with condition-specific validation. Dense-urban domains with more compute justify query-based or hybrid detectors for their occlusion evidence. Safety-critical all-weather operation requires multimodal redundancy, calibrated uncertainty, runtime monitoring, and defined fallback behaviour regardless of the detector family chosen.

The review's limitations are stated plainly. Screening was performed by a single reviewer with a structured verification pass rather than dual independent screening; the protocol was not preregistered; the synthesis is qualitative where the literature's heterogeneity in hardware, mAP definitions, and datasets prevents valid quantitative pooling; publication bias favours positive results; and the field moves quickly enough that the June 2026 search date bounds the

conclusions. Within those limits, the central finding is stable: progress toward deployable AV perception now depends less on further benchmark accuracy than on standardized evidence — latency reported on named hardware, condition-controlled robustness, cross-dataset degradation, calibrated uncertainty, and reproducible artifacts. The unified taxonomy, the comparability protocol, and above all the AVOD-DRF framework contributed here are instruments for producing and auditing exactly that evidence, and they are offered as reusable methodology for the research community rather than as conclusions fixed to one review.

Funding

The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Competing interests

The authors declare that they have no competing interests.

Data availability

The authors declare that the data supporting the findings of this study is available within the paper.

References

1. Liang L, Ma H, Zhao L, Xie X, Hua C, Zhang M, Zhang Y (2024) Vehicle detection algorithms for autonomous driving: a review. Sensors 24(10):3088. https://doi.org/10.3390/s24103088
2. Hussain K, Moreira C, Pereira J, Jardim S, Jorge J (2025) A comprehensive literature review on modular approaches to autonomous driving: deep learning for road and racing scenarios. Smart Cities 8(3):79. https://doi.org/10.3390/smartcities8030079
3. Feng D, Harakeh A, Waslander SL, Dietmayer K (2022) A review and comparative study on probabilistic object detection in autonomous driving. IEEE Trans Intell Transp Syst 23(8):9961-9980. https://doi.org/10.1109/TITS.2021.3096854
4. Zhang Y, Carballo A, Yang H, Takeda K (2023) Perception and sensing for autonomous vehicles under adverse weather conditions: a survey. ISPRS J Photogramm Remote Sens 196:146-177. https://doi.org/10.1016/j.isprsjprs.2022.12.021
5. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521(7553):436-444. https://doi.org/10.1038/nature14539

6. Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), pp 580-587. https://doi.org/10.1109/CVPR.2014.81

7. Ren S, He K, Girshick R, Sun J (2017) Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans Pattern Anal Mach Intell 39(6):1137-1149. https://doi.org/10.1109/TPAMI.2016.2577031

8. Redmon J, Divvala S, Girshick R, Farhadi A (2016) You only look once: unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), pp 779-788. https://doi.org/10.1109/CVPR.2016.91

9. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, Berg AC (2016) SSD: single shot multibox detector. In: Computer vision - ECCV 2016, Lecture Notes in Computer Science, vol 9905. Springer, Cham, pp 21-37. https://doi.org/10.1007/978-3-319-46448-0_2

10. Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A, Zagoruyko S (2020) End-to-end object detection with transformers. In: Computer vision - ECCV 2020, Lecture Notes in Computer Science, vol 12346. Springer, Cham, pp 213-229. https://doi.org/10.1007/978-3-030-58452-8_13

11. Zhu X, Su W, Lu L, Li B, Wang X, Dai J (2021) Deformable DETR: deformable transformers for end-to-end object detection. In: 9th international conference on learning representations (ICLR). https://arxiv.org/abs/2010.04159

12. Liu Z, Lin Y, Cao Y, Hu H, Wei Y, Zhang Z, Lin S, Guo B (2021) Swin Transformer: hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF international conference on computer vision (ICCV), pp 9992-10002. https://doi.org/10.1109/ICCV48922.2021.00986

13. Guo J, Han K, Wu H, Tang Y, Chen X, Wang Y, Xu C (2022) CMT: convolutional neural networks meet vision transformers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 12165-12175. https://doi.org/10.1109/CVPR52688.2022.01186

14. Li Z, Wang W, Li H, Xie E, Sima C, Lu T, Qiao Y, Dai J (2022) BEVFormer: learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In: Computer vision - ECCV 2022, Lecture Notes in Computer Science, vol 13669. Springer, Cham, pp 1-18. https://doi.org/10.1007/978-3-031-20077-9_1

15. Zhao Y, Lv W, Xu S, Wei J, Wang G, Dang Q, Liu Y, Chen J (2024) DETRs beat YOLOs on real-time object detection. In: Proceedings of the IEEE/CVF conference on computer vision

and pattern recognition (CVPR), pp 16965-16974. https://doi.org/10.1109/CVPR52733.2024.01605

16. Khan A, Rauf Z, Sohail A, Khan AR, Asif H, Asif A, Farooq U (2023) A survey of the vision transformers and their CNN-transformer based variants. Artif Intell Rev 56(Suppl 3):2917-2970. https://doi.org/10.1007/s10462-023-10595-0
17. Cao J, Peng B, Gao M, Hao H, Li X, Mou H (2025) Object detection based on CNN and vision-transformer: a survey. IET Comput Vis 19(1):e70028. https://doi.org/10.1049/cvi2.70028
18. Chu F, Li H, Xie L, Zhao J (2026) A survey of transformer architectures for autonomous driving. Expert Syst Appl 299:130338. https://doi.org/10.1016/j.eswa.2025.130338
19. Terven J, Cordova-Esparza DM, Romero-Gonzalez JA (2023) A comprehensive review of YOLO architectures in computer vision: from YOLOv1 to YOLOv8 and YOLO-NAS. Mach Learn Knowl Extr 5(4):1680-1716. https://doi.org/10.3390/make5040083
20. Zou Z, Chen K, Shi Z, Guo Y, Ye J (2023) Object detection in 20 years: a survey. Proc IEEE 111(3):257-276. https://doi.org/10.1109/JPROC.2023.3238524
21. Zaidi SSA, Ansari MS, Aslam A, Kanwal N, Asghar M, Lee B (2022) A survey of modern deep learning based object detection models. Digit Signal Process 126:103514. https://doi.org/10.1016/j.dsp.2022.103514
22. Wu X, Sahoo D, Hoi SCH (2020) Recent advances in deep learning for object detection. Neurocomputing 396:39-64. https://doi.org/10.1016/j.neucom.2020.01.085
23. Fujiyoshi H, Hirakawa T, Yamashita T (2019) Deep learning-based image recognition for autonomous driving. IATSS Res 43(4):244-252. https://doi.org/10.1016/j.iatssr.2019.11.008
24. Morooka FE, Junior AM, Sigahi TFAC, Pinto JS, Rampasso IS, Anholon R (2023) Deep learning and autonomous vehicles: strategic themes, applications, and research agenda using SciMAT and content-centric analysis, a systematic review. Mach Learn Knowl Extr 5(3):763-781. https://doi.org/10.3390/make5030041
25. Aljehane NO (2024) A study to investigate the role and challenges associated with the use of deep learning in autonomous vehicles. World Electr Veh J 15(11):518. https://doi.org/10.3390/wevj15110518
26. Huang R, Zhuo G, Xiong L, Lu S, Tian W (2023) A review of deep learning-based vehicle motion prediction for autonomous driving. Sustainability 15(20):14716. https://doi.org/10.3390/su152014716
27. Page MJ, McKenzie JE, Bossuyt PM, Boutron I, Hoffmann TC, Mulrow CD, Shamseer L, Tetzlaff JM, Akl EA, Brennan SE, Chou R, Glanville J, Grimshaw JM, Hrobjartsson A, Lalu

MM, Li T, Loder EW, Mayo-Wilson E, McDonald S, McGuinness LA, Stewart LA, Thomas J, Tricco AC, Welch VA, Whiting P, Moher D (2021) The PRISMA 2020 statement: an updated guideline for reporting systematic reviews. BMJ 372:n71. https://doi.org/10.1136/bmj.n71

28. Thaw MPP, Sarwatt DS, Ning H, Ding J (2026) Mitigating cyberattacks on autonomous vehicles: a comprehensive review of generative artificial intelligence defense techniques. Artif Intell Rev. https://doi.org/10.1007/s10462-026-11636-0
29. Okafor O, Adama DA, Vinkemeier D (2026) A systematic review of multisensor methods for open-carry and concealed knife detection. Artif Intell Rev. https://doi.org/10.1007/s10462-026-11626-2
30. Girshick R (2015) Fast R-CNN. In: Proceedings of the IEEE international conference on computer vision (ICCV), pp 1440-1448. https://doi.org/10.1109/ICCV.2015.169
31. Geiger A, Lenz P, Urtasun R (2012) Are we ready for autonomous driving? The KITTI vision benchmark suite. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), pp 3354-3361. https://doi.org/10.1109/CVPR.2012.6248074
32. Cordts M, Omran M, Ramos S, Rehfeld T, Enzweiler M, Benenson R, Franke U, Roth S, Schiele B (2016) The Cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), pp 3213-3223. https://doi.org/10.1109/CVPR.2016.350
33. Yu F, Chen H, Wang X, Xian W, Chen Y, Liu F, Madhavan V, Darrell T (2020) BDD100K: a diverse driving dataset for heterogeneous multitask learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 2633-2642. https://doi.org/10.1109/CVPR42600.2020.00271
34. Caesar H, Bankiti V, Lang AH, Vora S, Liong VE, Xu Q, Krishnan A, Pan Y, Baldan G, Beijbom O (2020) nuScenes: a multimodal dataset for autonomous driving. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 11618-11628. https://doi.org/10.1109/CVPR42600.2020.01164
35. Sun P, Kretzschmar H, Dotiwalla X, Chouard A, Patnaik V, Tsui P, Guo J, Zhou Y, Chai Y, Caine B, Vasudevan V, Han W, Ngiam J, Zhao H, Timofeev A, Ettinger S, Krivokon M, Gao A, Joshi A, Zhang Y, Shlens J, Chen Z, Anguelov D (2020) Scalability in perception for autonomous driving: Waymo open dataset. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 2443-2451. https://doi.org/10.1109/CVPR42600.2020.00252
36. Sakaridis C, Dai D, Van Gool L (2021) ACDC: the adverse conditions dataset with correspondences for semantic driving scene understanding. In: Proceedings of the IEEE/CVF

international conference on computer vision (ICCV), pp 10745-10755. https://doi.org/10.1109/ICCV48922.2021.01059

37. Cai Z, Vasconcelos N (2018) Cascade R-CNN: delving into high quality object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 6154-6162. https://doi.org/10.1109/CVPR.2018.00644
38. Carranza-Garcia M, Torres-Mateo J, Lara-Benitez P, Garcia-Gutierrez J (2021) On the performance of one-stage and two-stage object detectors in autonomous vehicles using camera data. Remote Sens 13(1):89. https://doi.org/10.3390/rs13010089
39. Lin TY, Goyal P, Girshick R, He K, Dollar P (2017) Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision (ICCV), pp 2999-3007. https://doi.org/10.1109/ICCV.2017.324
40. Redmon J, Farhadi A (2017) YOLO9000: better, faster, stronger. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), pp 6517-6525. https://doi.org/10.1109/CVPR.2017.690
41. Redmon J, Farhadi A (2018) YOLOv3: an incremental improvement. arXiv preprint. https://arxiv.org/abs/1804.02767
42. Bochkovskiy A, Wang CY, Liao HYM (2020) YOLOv4: optimal speed and accuracy of object detection. arXiv preprint. https://arxiv.org/abs/2004.10934
43. Wang CY, Bochkovskiy A, Liao HYM (2023) YOLOv7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 7464-7475. https://doi.org/10.1109/CVPR52729.2023.00721
44. Wang CY, Yeh IH, Liao HYM (2024) YOLOv9: learning what you want to learn using programmable gradient information. In: Computer vision - ECCV 2024, Lecture Notes in Computer Science, vol 15089. Springer, Cham, pp 1-21. https://doi.org/10.1007/978-3-031-72751-1_1
45. Law H, Deng J (2020) CornerNet: detecting objects as paired keypoints. Int J Comput Vis 128(3):642-656. https://doi.org/10.1007/s11263-019-01204-1
46. Zhou X, Wang D, Krahenbuhl P (2019) Objects as points. arXiv preprint. https://arxiv.org/abs/1904.07850
47. Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S, Uszkoreit J, Houlsby N (2021) An image is worth 16x16

words: transformers for image recognition at scale. In: 9th international conference on learning representations (ICLR). https://arxiv.org/abs/2010.11929

48. Jacob B, Kligys S, Chen B, Zhu M, Tang M, Howard A, Adam H, Kalenichenko D (2018) Quantization and training of neural networks for efficient integer-arithmetic-only inference. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 2704-2713. https://doi.org/10.1109/CVPR.2018.00286
49. Markidis S, Der Chien SW, Laure E, Peng IB, Vetter JS (2018) NVIDIA tensor core programmability, performance and precision. In: Proceedings of the IEEE international parallel and distributed processing symposium workshops (IPDPSW), pp 522-531. https://doi.org/10.1109/IPDPSW.2018.00091
50. Jia X, Tong Y, Qiao H, Li M, Tong J, Liang B (2023) Fast and accurate object detector for autonomous driving based on improved YOLOv5. Sci Rep 13:9711. https://doi.org/10.1038/s41598-023-36868-w
51. Mahaur B, Mishra KK, Kumar A (2023) An improved lightweight small object detection framework applied to real-time autonomous driving. Expert Syst Appl 234:121036. https://doi.org/10.1016/j.eswa.2023.121036
52. Luo W, Li Y, Urtasun R, Zemel R (2016) Understanding the effective receptive field in deep convolutional neural networks. In: Advances in neural information processing systems, vol 29. Curran Associates, pp 4898-4906
53. Ghiasi G, Lin TY, Le QV (2019) NAS-FPN: learning scalable feature pyramid architecture for object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 7029-7038. https://doi.org/10.1109/CVPR.2019.00720
54. Sakaridis C, Dai D, Van Gool L (2018) Semantic foggy scene understanding with synthetic data. Int J Comput Vis 126(9):973-992. https://doi.org/10.1007/s11263-018-1072-8
55. Chen Y, Li W, Sakaridis C, Dai D, Van Gool L (2018) Domain adaptive Faster R-CNN for object detection in the wild. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 3339-3348. https://doi.org/10.1109/CVPR.2018.00352
56. Michaelis C, Mitzkus B, Geirhos R, Rusak E, Bringmann O, Ecker AS, Bethge M, Brendel W (2019) Benchmarking robustness in object detection: autonomous driving when winter is coming. arXiv preprint. https://arxiv.org/abs/1907.07484
57. Joseph KJ, Khan S, Khan FS, Balasubramanian VN (2021) Towards open world object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 5826-5836. https://doi.org/10.1109/CVPR46437.2021.00577

58. Cheng T, Song L, Ge Y, Liu W, Wang X, Shan Y (2024) YOLO-World: real-time open-vocabulary object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 16901-16911. https://doi.org/10.1109/CVPR52733.2024.01599

59. Tan M, Pang R, Le QV (2020) EfficientDet: scalable and efficient object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 10778-10787. https://doi.org/10.1109/CVPR42600.2020.01079

60. Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L (2009) ImageNet: a large-scale hierarchical image database. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), pp 248-255. https://doi.org/10.1109/CVPR.2009.5206848

61. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), pp 770-778. https://doi.org/10.1109/CVPR.2016.90

62. Yosinski J, Clune J, Bengio Y, Lipson H (2014) How transferable are features in deep neural networks? In: Advances in neural information processing systems, vol 27. Curran Associates, pp 3320-3328

63. Li F, Zhang H, Liu S, Guo J, Ni LM, Zhang L (2024) DN-DETR: accelerate DETR training by introducing query denoising. IEEE Trans Pattern Anal Mach Intell 46(4):2239-2251. https://doi.org/10.1109/TPAMI.2023.3335410

64. Zhang H, Li F, Liu S, Zhang L, Su H, Zhu J, Ni LM, Shum HY (2023) DINO: DETR with improved denoising anchor boxes for end-to-end object detection. In: 11th international conference on learning representations (ICLR). https://arxiv.org/abs/2203.03605

65. Sun P, Zhang R, Jiang Y, Kong T, Xu C, Zhan W, Tomizuka M, Li L, Yuan Z, Wang C, Luo P (2021) Sparse R-CNN: end-to-end object detection with learnable proposals. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 14449-14458. https://doi.org/10.1109/CVPR46437.2021.01422

66. Li Y, Mao H, Girshick R, He K (2022) Exploring plain vision transformer backbones for object detection. In: Computer vision - ECCV 2022, Lecture Notes in Computer Science, vol 13669. Springer, Cham, pp 280-296. https://doi.org/10.1007/978-3-031-20077-9_17

67. Bai X, Hu Z, Zhu X, Huang Q, Chen Y, Fu H, Tai CL (2022) TransFusion: robust LiDAR-camera fusion for 3D object detection with transformers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 1080-1089. https://doi.org/10.1109/CVPR52688.2022.00116

68. Liu Z, Tang H, Amini A, Yang X, Mao H, Rus DL, Han S (2023) BEVFusion: multi-task multi-sensor fusion with unified bird's-eye view representation. In: Proceedings of the IEEE international conference on robotics and automation (ICRA), pp 2774-2781. https://doi.org/10.1109/ICRA48891.2023.10160968

69. Choi J, Chun D, Kim H, Lee HJ (2019) Gaussian YOLOv3: an accurate and fast object detector using localization uncertainty for autonomous driving. In: Proceedings of the IEEE/CVF international conference on computer vision (ICCV), pp 502-511. https://doi.org/10.1109/ICCV.2019.00059

70. Mahaur B, Mishra KK (2023) Small-object detection based on YOLOv5 in autonomous driving systems. Pattern Recognit Lett 168:115-122. https://doi.org/10.1016/j.patrec.2023.03.009

71. Mahmood K, Mahmood R, van Dijk M (2021) On the robustness of vision transformers to adversarial examples. In: Proceedings of the IEEE/CVF international conference on computer vision (ICCV), pp 7818-7827. https://doi.org/10.1109/ICCV48922.2021.00774

72. Nguyen KNT, Zhang W, Lu K, Wu YH, Zheng X, Li Tan H, Zhen L (2025) A survey and evaluation of adversarial attacks in object detection. IEEE Trans Neural Netw Learn Syst 36(9):15706-15722. https://doi.org/10.1109/TNNLS.2025.3561225

73. Hendrycks D, Dietterich T (2019) Benchmarking neural network robustness to common corruptions and perturbations. In: 7th international conference on learning representations (ICLR). https://arxiv.org/abs/1903.12261

74. Hnewa M, Radha H (2021) Object detection under rainy conditions for autonomous vehicles: a review of state-of-the-art and emerging techniques. IEEE Signal Process Mag 38(1):53-67. https://doi.org/10.1109/MSP.2020.2984801

75. Yacoob S, Kumar KV, Mahi Kittur J, et al (2025) Enhancing object detection robustness in adverse weather conditions. Procedia Comput Sci 252:1014-1024. https://doi.org/10.1016/j.procs.2025.01.062

76. Xie C, Wang J, Zhang Z, Zhou Y, Xie L, Yuille A (2017) Adversarial examples for semantic segmentation and object detection. In: Proceedings of the IEEE international conference on computer vision (ICCV), pp 1378-1387. https://doi.org/10.1109/ICCV.2017.153

77. Eykholt K, Evtimov I, Fernandes E, Li B, Rahmati A, Xiao C, Prakash A, Kohno T, Song D (2018) Robust physical-world attacks on deep learning visual classification. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 1625-1634. https://doi.org/10.1109/CVPR.2018.00175

78. Gu H, Yoon HJ, Jafarnejadsani H (2025) Robust object detection under adversarial patch attacks in vision-based navigation. Automation 6(3):44. https://doi.org/10.3390/automation6030044

79. Xiang C, Mittal P (2021) DetectorGuard: provably securing object detectors against localized patch hiding attacks. In: Proceedings of the ACM SIGSAC conference on computer and communications security (CCS), pp 3177-3196. https://doi.org/10.1145/3460120.3484757

80. Levine A, Feizi S (2020) (De)Randomized smoothing for certifiable defense against patch attacks. In: Advances in neural information processing systems, vol 33. Curran Associates, pp 6465-6475

81. Xiang C, Bhagoji AN, Sehwag V, Mittal P (2021) PatchGuard: a provably robust defense against adversarial patches via small receptive fields and masking. In: Proceedings of the 30th USENIX security symposium, pp 2237-2254

82. Huang X, Kwiatkowska M, Wang S, Wu M (2017) Safety verification of deep neural networks. In: Computer aided verification, Lecture Notes in Computer Science, vol 10426. Springer, Cham, pp 3-29. https://doi.org/10.1007/978-3-319-63387-9_1

83. Ehlers R (2017) Formal verification of piece-wise linear feed-forward neural networks. In: Automated technology for verification and analysis, Lecture Notes in Computer Science, vol 10482. Springer, Cham, pp 269-286. https://doi.org/10.1007/978-3-319-68167-2_19

84. Pei K, Cao Y, Yang J, Jana S (2019) DeepXplore: automated whitebox testing of deep learning systems. Commun ACM 62(11):137-145. https://doi.org/10.1145/3361566

85. Salay R, Queiroz R, Czarnecki K (2017) An analysis of ISO 26262: using machine learning safely in automotive software. arXiv preprint. https://arxiv.org/abs/1709.02435

86. Bogdoll D, Nitsche M, Zollner JM (2022) Anomaly detection in autonomous driving: a survey. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops (CVPRW), pp 4487-4498. https://doi.org/10.1109/CVPRW56347.2022.00495

87. Liu S, Zeng Z, Ren T, Li F, Zhang H, Yang J, Jiang Q, Li C, Yang J, Su H, Zhu J, Zhang L (2024) Grounding DINO: marrying DINO with grounded pre-training for open-set object detection. In: Computer vision - ECCV 2024, Lecture Notes in Computer Science, vol 15105. Springer, Cham, pp 38-55. https://doi.org/10.1007/978-3-031-72970-6_3

88. He K, Fan H, Wu Y, Xie S, Girshick R (2020) Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 9726-9735. https://doi.org/10.1109/CVPR42600.2020.00975

89. Chen T, Kornblith S, Norouzi M, Hinton G (2020) A simple framework for contrastive learning of visual representations. In: Proceedings of the 37th international conference on machine learning (ICML), PMLR 119, pp 1597-1607

90. Grill JB, Strub F, Altche F, Tallec C, Richemond PH, Buchatskaya E, Doersch C, Pires BA, Guo ZD, Azar MG, Piot B, Kavukcuoglu K, Munos R, Valko M (2020) Bootstrap your own latent: a new approach to self-supervised learning. In: Advances in neural information processing systems, vol 33. Curran Associates, pp 21271-21284

91. Caron M, Touvron H, Misra I, Jegou H, Mairal J, Bojanowski P, Joulin A (2021) Emerging properties in self-supervised vision transformers. In: Proceedings of the IEEE/CVF international conference on computer vision (ICCV), pp 9630-9640. https://doi.org/10.1109/ICCV48922.2021.00951

92. He K, Chen X, Xie S, Li Y, Dollar P, Girshick R (2022) Masked autoencoders are scalable vision learners. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 15979-15988. https://doi.org/10.1109/CVPR52688.2022.01553

93. Tong Z, Song Y, Wang J, Wang L (2022) VideoMAE: masked autoencoders are data-efficient learners for self-supervised video pre-training. In: Advances in neural information processing systems, vol 35. Curran Associates, pp 10078-10093

94. Xie E, Ding J, Wang W, Zhan X, Xu H, Sun P, Li Z, Luo P (2021) DetCo: unsupervised contrastive learning for object detection. In: Proceedings of the IEEE/CVF international conference on computer vision (ICCV), pp 8372-8381. https://doi.org/10.1109/ICCV48922.2021.00828

95. Xie S, Gu J, Guo D, Qi CR, Guibas L, Litany O (2020) PointContrast: unsupervised pre-training for 3D point cloud understanding. In: Computer vision - ECCV 2020, Lecture Notes in Computer Science, vol 12348. Springer, Cham, pp 574-591. https://doi.org/10.1007/978-3-030-58580-8_34

96. Munir F, Azam S, Jeon M (2021) SSTN: self-supervised domain adaptation thermal object detection for autonomous driving. In: Proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS), pp 206-213. https://doi.org/10.1109/IROS51168.2021.9636353

97. Ciocarlan A, Lefebvre S, Le Hegarat-Mascle S, Woiselle A (2024) Self-supervised learning for real-world object detection: a survey. arXiv preprint. https://arxiv.org/abs/2410.07442

98. De Lange M, Aljundi R, Masana M, Parisot S, Jia X, Leonardis A, Slabaugh G, Tuytelaars T (2022) A continual learning survey: defying forgetting in classification tasks. IEEE Trans Pattern Anal Mach Intell 44(7):3366-3385. https://doi.org/10.1109/TPAMI.2021.3057446

99. Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, Andreetto M, Adam H (2017) MobileNets: efficient convolutional neural networks for mobile vision applications. arXiv preprint. https://arxiv.org/abs/1704.04861

100. Han S, Mao H, Dally WJ (2016) Deep compression: compressing deep neural networks with pruning, trained quantization and Huffman coding. In: 4th international conference on learning representations (ICLR). https://arxiv.org/abs/1510.00149

101. Suda N, Chandra V, Dasika G, Mohanty A, Ma Y, Vrudhula S, Seo JS, Cao Y (2016) Throughput-optimized OpenCL-based FPGA accelerator for large-scale convolutional neural networks. In: Proceedings of the ACM/SIGDA international symposium on field-programmable gate arrays (FPGA), pp 16-25. https://doi.org/10.1145/2847263.2847276

102. Ratul IJ, Zhou Y, Yang K (2025) Accelerating deep learning inference: a comparative analysis of modern acceleration frameworks. Electronics 14(15):2977. https://doi.org/10.3390/electronics14152977

103. Chen Y, Yang T, Zhang X, Meng G, Xiao X, Sun J (2019) DetNAS: backbone search for object detection. In: Advances in neural information processing systems, vol 32. Curran Associates, pp 6642-6652

104. Yao L, Xu H, Zhang W, Liang X, Li Z (2020) SM-NAS: structural-to-modular neural architecture search for object detection. In: Proceedings of the AAAI conference on artificial intelligence, vol 34(7), pp 12661-12668. https://doi.org/10.1609/aaai.v34i07.6958

105. Wang X, Lin J, Zhao J, Yang X, Yan J (2022) EAutoDet: efficient architecture search for object detection. In: Computer vision - ECCV 2022, Lecture Notes in Computer Science, vol 13680. Springer, Cham, pp 668-684. https://doi.org/10.1007/978-3-031-20044-1_38

106. Mok J, Na B, Choe H, Yoon S (2021) AdvRush: searching for adversarially robust neural architectures. In: Proceedings of the IEEE/CVF international conference on computer vision (ICCV), pp 12302-12312. https://doi.org/10.1109/ICCV48922.2021.01210

107. Bai H, Zhou F, Hou L, Camtepe S, Nepal S (2021) NAS-OoD: neural architecture search for out-of-distribution generalization. In: Proceedings of the IEEE/CVF international conference on computer vision (ICCV), pp 8300-8309. https://doi.org/10.1109/ICCV48922.2021.00821

108. Elsken T, Metzen JH, Hutter F (2019) Neural architecture search: a survey. J Mach Learn Res 20(55):1-21

109. Zoph B, Le QV (2017) Neural architecture search with reinforcement learning. In: 5th international conference on learning representations (ICLR). https://arxiv.org/abs/1611.01578

110. Zoph B, Vasudevan V, Shlens J, Le QV (2018) Learning transferable architectures for scalable image recognition. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 8697-8710. https://doi.org/10.1109/CVPR.2018.00907

111. Guo C, Pleiss G, Sun Y, Weinberger KQ (2017) On calibration of modern neural networks. In: Proceedings of the 34th international conference on machine learning (ICML), PMLR 70, pp 1321-1330

112. Gawlikowski J, Tassi CRN, Ali M, Lee J, Humt M, Feng J, Kruspe A, Triebel R, Jung P, Roscher R, Shahzad M, Yang W, Bamler R, Zhu XX (2023) A survey of uncertainty in deep neural networks. Artif Intell Rev 56(Suppl 1):1513-1589. https://doi.org/10.1007/s10462-023-10562-9

113. Zablocki E, Ben-Younes H, Perez P, Cord M (2022) Explainability of deep vision-based autonomous driving systems: review and challenges. Int J Comput Vis 130(10):2425-2452. https://doi.org/10.1007/s11263-022-01657-x

114. Montavon G, Samek W, Muller KR (2018) Methods for interpreting and understanding deep neural networks. Digit Signal Process 73:1-15. https://doi.org/10.1016/j.dsp.2017.10.011