ADVANCE SOCIAL SCIENCE ARCHIVE JOURNAL
Available Online: https://assajournal.com
Vol. 05 No. 01. Jan-March 2026.Page#.6351-6375
Print ISSN: 3006-2497 Online ISSN: 3006-2500
Platform & Workflow by: Open Journal Systems

Deep Learning for Fruit and Vegetable Detection in Autonomous Harvesting: A Comparative Narrative Review (2014–2026)

Zeeshan Malik[1], LI Minzan[1]*, SUN Hong[2], ZHANG Man[1], LI Han[1,2], YANG Wei[1,2], ZHANG Yao[1,2]
1 Key Laboratory of Smart Agriculture System Integration, Ministry of Education, China Agricultural University, Beijing 100083, China
2 Key Laboratory of Agricultural Information Acquisition Technology, Ministry of Agriculture and Rural Affairs, China Agricultural University, Beijing 100083, China

Abstract
Autonomous harvesting is central to reducing labour dependence and post-harvest loss in fruit and vegetable production, and fast, accurate visual detection remains its principal bottleneck. This paper systematically reviews deep-learning (DL) approaches to fruit and vegetable detection for robotic harvesting, updating and extending the existing literature in two respects. First, it synthesizes results across 100+ primary studies using a common comparative framework — rather than a chronological narrative — spanning two-stage detectors (R-CNN, Faster R-CNN), one-stage CNN detectors (SSD, YOLOv1–v7), the current anchor-free YOLO family (YOLOv8–YOLOv13), transformer-based real-time detectors (RT-DETR and its derivatives), and instance-segmentation/pose-based approaches used for picking-point localization rather than mere bounding-box detection. Second, it extends the review's scope beyond detection accuracy to the two factors that determine real-world deployability: edge/embedded inference performance and dataset standardization. The review finds that anchor-free YOLO variants (v8–v10) currently offer the best accuracy–latency trade-off for on-robot deployment, that transformer-based detectors (RT-DETR family) match or exceed YOLO accuracy in heavily occluded, cluttered canopy scenes but at a higher compute cost that constrains edge use without quantization or pruning, and that occlusion, illumination variability, and the scarcity of standardized, richly annotated field datasets remain the field's central unsolved problems. We conclude with a structured research agenda covering multimodal (RGB-D/LiDAR) fusion, foundation-model-assisted annotation, cross-domain generalization, and reporting standards that would allow fair comparison across studies — a gap that continues to limit reproducibility in this literature.
Keywords: harvesting robot; fruit and vegetable detection; deep learning; YOLO; RT-DETR; instance segmentation; picking-point localization; edge deployment; precision agriculture

1. Introduction

Fruit and vegetable harvesting remains one of the most labour-intensive stages of horticultural production. Seasonal labour shortages, rising wage costs, and the narrow time windows within which many crops must be picked at optimal ripeness have made autonomous, robotic harvesting an active and urgent research target for both academia and industry. At the core of every harvesting robot is a visual perception subsystem responsible for finding the crop, distinguishing it from leaves, branches, and other fruit, and estimating a precise picking point that a manipulator can act on. This detection task is difficult in field conditions: fruit is frequently occluded by foliage or by other fruit, canopy illumination varies with time of day and weather, fruit colour can closely match background foliage, and individual specimens of the same species vary substantially in size, shape, and ripeness stage.

Deep learning has transformed how this detection problem is addressed. Since convolutional neural networks (CNNs) demonstrated a decisive advantage over handcrafted-feature pipelines (HOG, SIFT, colour thresholding) in general-purpose object detection, agricultural vision research has progressively adopted the same architectures — first two-stage region-proposal detectors such as R-CNN and Faster R-CNN, then one-stage detectors such as SSD and the YOLO family, and, most recently, anchor-free and transformer-based real-time detectors. Each architectural generation has been applied, adapted, and benchmarked on a growing number of crops: apples, citrus, kiwifruit, tomatoes, strawberries, mangoes, grapes, peppers, and others.

1.1 Positioning relative to existing reviews

Several reviews already cover parts of this space. Huang et al. (2025) surveyed visual perception technology for fruit harvesting robots broadly, emphasizing camera selection, active vision, and visual servoing rather than a comparative synthesis of detector accuracy. Zhang et al. (2024) reviewed fruit-picking technology at the level of the whole robotic system (perception, planning, end-effector design) rather than focusing in depth on the detection architectures themselves. Tan et al. (2025) reviewed deep-learning-based picking robots with an emphasis on robot mechanical design, and Zhang et al. (2025) focused specifically on apple-harvesting robots and picking-pose estimation. Rajendran et al. (2024) reviewed selective-harvesting robots end-to-end, covering perception only as one of several subsystems alongside motion planning and control.

This review differs from the above in three respects. First, its scope explicitly spans both fruits and vegetables rather than a single crop family or a single robot subsystem. Second, it organizes the detector literature by architecture generation and provides a single, unified comparative table (Table 1) that normalizes reported metrics across studies, rather than presenting a chronological narrative of individual papers — the format most existing reviews in this specific sub-area still use, and one that makes cross-study comparison difficult for readers. Third, it explicitly extends coverage to the anchor-free YOLOv8–YOLOv13 generation, the RT-DETR transformer family, and instance-segmentation/pose-based picking-point localization, together with a dedicated treatment of edge-deployment constraints — architectural and deployment developments that fall outside the horizon of reviews published before 2023 and that are only partially covered even in several 2024–2025 reviews.

To make this differentiation checkable rather than asserted: of the 94 primary studies synthesized in this review, 24 were published in 2024 or later, after the search cutoffs of Zhang et al. (2024) and Rajendran et al. (2024), and the anchor-free YOLOv8–v13 generation, the RT-DETR family, and pose-based picking-point localization together account for 19 of the 94 included studies — architectural coverage that, by construction, could not appear in any review published before 2023.

The specific objectives of this review are to: (i) synthesize, using a common comparative framework, the deep-learning architectures applied to fruit and vegetable detection from two-stage R-CNN variants through to transformer-based real-time detectors; (ii) evaluate these architectures not only on reported accuracy but on their suitability for on-robot, real-time, edge-hardware deployment; (iii) consolidate and critically assess the public datasets available for benchmarking; and (iv) identify the specific, currently unresolved research gaps — rather than a generic list of challenges — that should shape the next generation of work in this field.

2. Review Methodology

Primary studies were identified through structured searches of Web of Science, Scopus, IEEE Xplore, ScienceDirect, and Google Scholar, using combinations of the terms ("fruit" OR "vegetable") AND ("detection" OR "recognition" OR "picking point" OR "harvesting") AND ("deep learning" OR "YOLO" OR "R-CNN" OR "transformer" OR "RT-DETR" OR "instance segmentation"), restricted to peer-reviewed journal articles and archival conference papers published between January 2014 and June 2026. Titles and abstracts were first screened for relevance to robotic/autonomous harvesting (studies addressing only manual-harvest yield estimation or post-harvest grading without a detection component aimed at robotic picking were excluded); the remaining full texts were then assessed for whether they reported a quantitative detection metric (mAP, precision, recall, F1-score, or equivalent) on field or greenhouse-acquired imagery. This process yielded the pool of roughly 130 records identified across databases, of which 94 met inclusion criteria and are synthesized in Sections 3–7, supplemented by the review, survey, and architecture-defining papers cited in Section 1.1 and Section 3.

Two methodological limitations of this process should be stated plainly rather than left implicit. First, screening and inclusion decisions were made by a single reviewer rather than in duplicate, so the usual inter-rater reliability check that a fully systematic review would report (e.g., Cohen's kappa on a sample of screening decisions) is not available here; a second independent screener would strengthen any future update of this work. Second, and more consequentially, this review does not apply a formal risk-of-bias or study-quality appraisal (e.g., an adapted QUADAS-2 or similar checklist) to the included studies before extracting their reported metrics into Table 1. Every accuracy figure reported in Section 3 is taken at face value from the source paper's own test-set evaluation, with no independent check on whether that test set was held out from training data, adequately powered, or representative of field conditions rather than curated imagery. Given that dataset and evaluation quality are themselves identified in Section 8.1 as the field's central weakness, this is a genuine limitation of the review process itself, not only of the underlying literature, and is revisited in Section 8.4. For these reasons we describe this work as a comparative narrative review rather than a systematic review in the PRISMA sense; publication-language coverage also skews toward English-language venues, and grey literature (preprints without subsequent peer review) is cited sparingly and flagged explicitly where used.

3. Deep Learning Architectures for Fruit and Vegetable Detection

This section synthesizes the detector literature by architectural generation. Table 1 consolidates quantitative results across the studies discussed below, using a normalized set of columns (crop, metric type, reported value, approximate test-set size, and year) so that results can be compared directly rather than read as isolated data points. Test-set size is included because, as discussed in Section 8, a headline accuracy figure obtained on a few hundred images carries far less evidential weight than the same figure obtained on several thousand, and Table 1 as originally drafted did not make that distinction visible.

3.1 3.1 Two-stage detectors: R-CNN and Faster R-CNN

Two-stage detectors first generate class-agnostic region proposals and then classify and refine each proposal, trading inference speed for localization accuracy. Faster R-CNN, which replaced the earlier selective-search proposal stage with a learned Region Proposal Network, has been the dominant two-stage architecture in this literature. Gao et al. (2020) applied a VGG-16-based Faster R-CNN to multi-class apple detection in a formal fruiting-wall system, reporting a mean average precision (mAP) of 0.879 across four classes, with the remaining errors concentrated in fruit occluded by branches or trellis wire. Fu et al. (2018) built a ZFNet-based Faster R-CNN for kiwifruit detection in field images, achieving 89.3% average accuracy with good robustness to illumination change. Hu et al. (2019) combined Faster R-CNN with an intuitionistic fuzzy set to separate overlapping ripe tomatoes, reporting sub-pixel localization errors but noting that detection accuracy degraded sharply for small, distant, or heavily shadowed fruit. Fu et al. (2020) subsequently incorporated RGB-D depth filtering into a Faster R-CNN pipeline for apple detection, improving accuracy by roughly 2.5 percentage points over RGB-only input.

Across these studies, two-stage detectors consistently deliver strong localization precision but at inference times (150–300 ms per image on the hardware used) that are difficult to reconcile with the sub-100 ms cycle times typically required for continuous robotic operation. This speed limitation, more than accuracy, is the reason two-stage detectors have been progressively displaced by one-stage architectures in recent (post-2020) agricultural vision literature.

3.2 3.2 One-stage CNN detectors: SSD and YOLOv1–v7

One-stage detectors predict class and location in a single forward pass, at some cost to localization precision but with a substantial speed advantage. SSD, evaluated by Liang et al. (2018) for on-tree mango detection, reached an F1-score of 0.911 at 35 FPS — competitive with Faster R-CNN at a fraction of the latency. Yuan et al. (2020) applied an Inception-V2-based SSD to cherry-tomato detection and reported 98.85% average precision, though with persistent difficulty on side-grown, partially visible fruit.

The YOLO family has been the most extensively studied one-stage architecture in this domain. Early applications (YOLOv2–v3) achieved detection times near 19 ms per image for apple recognition with error rates of roughly 8–9% (Kuznetsova et al., 2020), and Lawal (2021) reported that a DenseNet/MixNet-modified YOLOv3 reached 98.3–98.4% accuracy for tomato detection in cluttered scenes. YOLOv4-based models pushed reported metrics further: Wu et al. (2021) achieved a 96.54% F1-score for apple detection using data augmentation to address complex backgrounds, and Zhou et al. (2021) reported 95% accuracy for tomato maturity classification combining YOLOv4 with a statistical colour model. YOLOv5, released in 2020, became the most widely adopted variant of this generation owing to its favourable accuracy–speed trade-off and ease of deployment: Yan et al. (2021) reported a real-time apple-picking detector with 87.49% F1-score, and Yang et al. (2022) combined YOLOv5 with a BCo backbone to reach 89.15% precision and 97.70% mAP across apples, grapes, and citrus. YOLOv7 (Wang et al., 2022) was applied by Xia et al. (2023) to kiwifruit detection with data-augmentation-assisted training, reporting 92.1% precision, 88.1% recall, and 95.3% mAP; Wu et al. (2022) reported a comparable 94.21% precision rate for Camellia oleifera fruit detection, though performance degraded under strong backlighting and heavy occlusion.

3.3 3.3 The anchor-free YOLO family (YOLOv8–YOLOv13)

From 2023 onward, the field's centre of gravity shifted to the anchor-free YOLOv8 generation and its successors. YOLOv8 (Ultralytics, 2023) removed the anchor-box mechanism used in earlier YOLO versions and introduced a decoupled detection head, improving small-object and dense-cluster detection — both directly relevant to occluded, clustered fruit. It has since been adapted extensively: Liu, Abeyrathna, Sampurno, Nakaguchi and Ahamed (2024) proposed Faster-YOLO-AP, a lightweight YOLOv8 variant using an efficient PDWConv module for apple detection in orchards, reporting mAP50 of 99.2% and F1 of 99% on a purpose-built orchard dataset, with the design explicitly targeting on-robot deployment; Sapkota, Ahmed, Churuvija and Karkee (2024) combined YOLOv8m-seg with shape-fitting post-processing for immature green-apple detection and sizing, reporting a segmentation AP@0.5 of 0.94 — a task substantially harder than mature-fruit detection because of the low colour contrast between unripe fruit and foliage; and Kong, Wang, Zhang, Li and Rong (2023) paired an improved YOLOv8s (with a BiFormer attention backbone) with an adaptive monocular multi-resolution depth-estimation model, reporting an mAP of 88.45% and a citrus recognition accuracy of 94.7% (a 2.7-point improvement over the unmodified baseline), with downstream simulated grasping success rates of 80.6% and 85.1% at 30 cm and 45 cm working distances respectively — one of the few studies in this literature to report a grasp-level outcome rather than detection accuracy alone.

YOLOv9 (February 2024) introduced Programmable Gradient Information (PGI) and the GELAN lightweight backbone, reducing parameter count by roughly 15% relative to YOLOv8x while improving AP by about 1.7 percentage points (Liu et al., 2024). In a direct comparative field evaluation spanning YOLOv8, YOLOv9, YOLOv10, and YOLO11 for immature green-apple (fruitlet) detection and counting in commercial orchards, Sapkota, Meng, Churuvija, Du, Ma and Karkee (2024; preprint, not yet peer-reviewed at the time of writing — see ref. [95]) found that YOLOv9 delivered the strongest overall accuracy of the generations tested, ahead of the nominally newer YOLOv10 and YOLO11 releases. This finding is worth pausing on, because it is easy to read as a curiosity and it is not one: it is a specific, testable instance of a general and well-established problem in machine learning known as covariate shift or domain shift, where a model's ranking on the benchmark it was optimized against (here, COCO-style natural imagery, with its own characteristic object scale and density statistics) does not necessarily transfer to a target domain with a different data-generating distribution — in this case, densely clustered, self-similar, heavily occluded orchard canopy imagery (Ben-David et al., 2010). YOLOv10–v12's architectural choices (NMS-free heads, latency-oriented pruning, area-attention) were tuned against that benchmark distribution, not against orchard imagery, so their COCO-leaderboard ranking is not guaranteed to predict their ranking on a fruitlet-counting task. The practical implication, stated plainly, is that version number should never substitute for task-specific benchmarking on the actual target crop and imagery — a point this review returns to in Section 9. YOLOv10 and YOLOv11 introduced further latency reductions (NMS-free training, C2f-faster modules), and YOLOv12 added an area-attention mechanism that improved the accuracy–latency Pareto frontier on general benchmarks (Khanam & Hussain, 2024; Tian et al., 2025). Most recently, YOLOv13 (Lei et al., 2025) introduced a Hypergraph Adaptive Correlation Enhancement (HyperACE) module and a Full-Pipeline Aggregation-Distribution (FullPAD) mechanism, reporting further gains on standard detection benchmarks; agricultural-domain evaluations of YOLOv13 remain limited at the time of writing and are flagged here as an open validation need rather than an established result.

3.4 3.4 Transformer-based real-time detectors: DETR and RT-DETR

Detection Transformer (DETR) architectures replace the anchor-generation and non-maximum-suppression (NMS) post-processing used by CNN detectors with a set-based, end-to-end prediction formulation built on self-attention (Vaswani et al., 2017; Carion et al., 2020). The distinction is not merely computational bookkeeping, and it is worth being explicit about the mechanism rather than treating it as a black box. A CNN's receptive field — the region of the input that can influence a given feature — grows only gradually with network depth, and any single convolutional layer only ever integrates information from its local neighbourhood. Self-attention, by contrast, computes a weighted interaction between every pair of tokens in a single layer, giving each spatial location a global receptive field from the first layer onward. This matters specifically for occlusion reasoning: correctly identifying a partially hidden fruit in a dense cluster often depends less on the local texture visible at that location than on the layout and appearance of the other, unoccluded fruit around it — evidence a CNN can only integrate after several layers of downsampling, by which point fine spatial detail has already been lost, but that a transformer's attention mechanism can draw on directly. This is the underlying reason RT-DETR-family models perform competitively in heavily occluded, cluttered canopy scenes, not simply that they are 'newer.'

Early DETR variants were too slow for real-time robotic use, but RT-DETR (Zhao et al., 2024) closed this gap: using a hybrid encoder and IoU-aware query selection, RT-DETR-R50 reported 53.1% AP at 108 FPS on COCO, matching or exceeding equivalent-scale YOLO detectors while eliminating NMS post-processing entirely. The NMS-related failure mode this avoids is worth naming precisely, since Table 2 refers to it without explanation: NMS suppresses any detection whose bounding box overlaps another retained detection above an IoU threshold, on the assumption that overlapping boxes are duplicate detections of the same object — an assumption that is false whenever two genuinely distinct fruits happen to overlap heavily in the image plane, which in a dense cluster is common rather than exceptional. NMS will then discard a true positive, and no amount of detector accuracy upstream of that post-processing step can recover it. RT-DETR's set-based prediction formulation removes this specific failure mode by construction.

Agricultural applications of RT-DETR have grown quickly since 2024. Wang et al. (2024) developed PDSI-RT-DETR, a lightweight tomato-ripeness detector that improved precision, recall, mAP50, and mAP50:95 by 4.7, 5.4, 3.9, and 4.1 percentage points respectively over baseline RT-DETR while reducing parameters by 30.8% and GFLOPs by 17.6% — a genuine accuracy-and-efficiency win rather than a pure speed/accuracy trade. Zhao, Chen, Ge et al. (2024) proposed RT-DETR-Tomato for agricultural safety-production contexts; the published abstract reports accuracy as "greatly improved" without stating a single comparable headline percentage, which we note here rather than estimate, consistent with Section 8.4's broader point about inconsistent reporting in this literature. Wu, Qiu, Wang, Su, Cao and Bai (2025) reported an improved RT-DETR for general fruit-ripeness detection, replacing standard convolution with RepBlocks and adding an efficient multi-scale attention (EMA) module in the backbone; their own comparison found that RT-DETR required this kind of custom, domain-specific optimization to compete with YOLOv8 on their crop-maturity task, a finding consistent with the mechanism described above — global attention helps most precisely where occlusion and clutter make local context insufficient, but that benefit is not free, and a transformer detector applied off-the-shelf to a less cluttered scene may simply pay its higher compute cost without a matching accuracy return. The practical implication for harvesting-robot designers is that RT-DETR-family models are best treated as a targeted tool for severe-occlusion, dense-cluster crops, rather than a default replacement for YOLO across the board.

3.5 3.5 Instance segmentation and pose-based picking-point localization

A bounding box is often insufficient for robotic harvesting: the manipulator needs a specific picking point — typically on the stem or peduncle — not merely the fruit's approximate location. This has driven a shift, most visible in the 2023–2025 literature, from bounding-box detection toward instance segmentation and keypoint/pose estimation. Mask R-CNN-based pipelines segment the fruit (and, in more recent work, the peduncle) at pixel level and derive the picking point geometrically from the resulting mask; Lu, Ji, Yang and Jia (2023) applied this approach ('Mask Positioner') to green-persimmon segmentation in complex backgrounds, reporting a segmentation accuracy of 67.4% and a detection accuracy of 69.1% — figures well below the 90%+ range reported for mature, higher-contrast fruit elsewhere in this review, and a useful reminder that green-on-green segmentation remains a genuinely unsolved problem rather than one that recent architectures have already closed. Li, Huang, Gu, He, Huang and Wang (2024) combined an improved YOLOv8n (with a BiFPN structure and SPD-Conv module) for simultaneous fruit-and-stem detection with a YOLOv8n-seg stage for stem segmentation in mango, reporting fruit/stem detection precision of 98.9%/97.1%, recall of 99.5%/94.6%, and — notably, since this is one of the few studies in this literature to report it — an end-to-end picking-point positioning success rate of 92.01%, deployed and validated on an edge device rather than a workstation GPU alone. Xie et al. (2024) combined YOLOv8s object detection with an attention-augmented YOLOv8s-seg-CBAM instance-segmentation stage for strawberry peduncle detection and picking-point localization, reporting a peduncle detection accuracy of 86.2% at 30.6 ms per image. Combined RGB-D and LiDAR sensing (rather than RGB alone) is increasingly used to resolve the depth ambiguity that 2D detection cannot address, particularly for stem-level localization in dense canopies.

This shift also exposes a conceptual gap in how the field talks about occlusion, which is usually treated as a single, undifferentiated 'challenge' (see Section 8.2) rather than a phenomenon with at least two distinct causes that call for different solutions. Extrinsic occlusion — a fruit hidden behind a leaf, branch, or another fruit — is fundamentally a visibility problem: the target's true geometry exists but is not imaged, and the correct response is amodal reasoning, inferring the occluded shape from partial evidence and context, a capability studied directly in the general computer-vision amodal-segmentation literature (Zhu et al., 2017) but rarely invoked explicitly in the agricultural detection papers reviewed here. Self-occlusion — where the fruit's own geometry hides its stem or picking point from the camera's current viewing angle — is instead a viewpoint problem, and its natural solution is active or multi-view vision (moving the camera, or fusing several viewpoints) rather than any amount of single-frame reasoning, however sophisticated. Most of the pose- and segmentation-based methods surveyed above are, implicitly, extrinsic-occlusion solutions; self-occlusion is comparatively under-addressed in the literature synthesized here, and we return to this distinction as a concrete direction in Section 9.

This shift also matters for how detection accuracy should be reported and compared: a high mAP on fruit bounding boxes does not guarantee an equivalently high success rate at the actual picking-point level. The mango study above is the clearest illustration available in this review's pool of studies — fruit detection precision above 98% still yielded an end-to-end positioning success rate of 92.01%, a meaningful and informative gap rather than a rounding error, and one that a detection-only mAP figure would never reveal. Reporting both figures, wherever a full pipeline exists to measure them, should become standard practice; Section 8.4 and Section 9 return to this as a concrete recommendation.

Table 1. Comparative summary of deep-learning detector results reported for fruit and vegetable detection, organized by architecture generation.

Architecture	Crop / Target	Key Metric	Reported Value	Test-set size (approx.)	Year	Source
Faster R-CNN (VGG-16)	Apple (4 classes)	mAP	0.879	~1,000 img.	2020	Gao et al.
Faster R-CNN (ZFNet)	Kiwifruit	Accuracy	89.3%	~1,200 img.	2018	Fu et al.
Faster R-CNN + IFS	Ripe tomato	RMSE (width)	2.996 px	not stated in abstract	2019	Hu et al.
Faster R-CNN + RGB-D	Apple	Accuracy	0.893 (+2.5% vs RGB-only)	not stated in abstract	2020	Fu et al.
SSD (VGG-16)	Mango	F1-score	0.911 @ 35 FPS	not stated in abstract	2018	Liang et al.
SSD (Inception-V2)	Cherry tomato	Avg. precision	98.85%	not stated in abstract	2020	Yuan et al.
YOLOv3 (pre/post-proc.)	Apple	Detection time / error	19 ms; 7.8–9.2% error	not stated in abstract	2020	Kuznetsova et al.
YOLOv3 (DenseNet/MixNet)	Tomato	Accuracy	98.3–98.4%	not stated in abstract	2021	Lawal
YOLOv4 (augmented)	Apple	F1-score	96.54%	not stated in abstract	2021	Wu et al.
YOLOv4 + colour model	Tomato maturity	Accuracy	95%	not stated in abstract	2021	Zhou et al.
YOLOv5	Apple	F1-score	87.49%	not stated in abstract	2021	Yan et al.
YOLOv5 + BCo backbone	Apple / grape / citrus	Precision / mAP	89.15% / 97.70%	not stated in abstract	2022	Yang et al.

Architecture	Crop / Target	Key Metric	Reported Value	Test-set size (approx.)	Year	Source
YOLOv7	Kiwifruit	Precision / mAP	92.1% / 95.3%	not stated in abstract	2023	Xia et al.
YOLOv7	Camellia oleifera	Precision	94.21%	not stated in abstract	2022	Wu et al.
Faster-YOLO-AP (YOLOv8 + PDWConv)	Apple (lightweight, edge deploy.)	mAP50 / mAP50-95 / F1	99.2% / 94.6% / 99%	not stated in abstract	2024	Liu, Abeyrathna, Sampurno, Nakaguchi & Ahamed
YOLOv8m-seg + shape fitting	Immature green apple	AP@0.5 / AP@0.75 (segmentation)	0.94 / 0.91	not stated in abstract	2024	Sapkota, Ahmed, Churuvija & Karkee
YOLOv8s (BiFormer) + multi-resolution depth	Citrus (3-D localization + grasp)	mAP / recognition accuracy / grasp success	88.45% / 94.7% / 80.6–85.1%	not stated in abstract	2023	Kong, Wang, Zhang, Li & Rong
YOLOv9 (GELAN/PGI)	General benchmark (COCO)	Params / AP vs. YOLOv8x	−15% params / +1.7% AP	COCO (~118k train img.)	2024	Liu, Xue, Zhang et al.
YOLOv8 / v9 / v10 / YOLO11 (comparative field eval.)	Immature green apple / fruitlet (detection + counting)	Best-in-class generation	YOLOv9 strongest of the four generations tested	orchard field images (preprint; count not stated)	2024 [preprint]	Sapkota, Meng, Churuvija, Du, Ma & Karkee — arXiv, not peer-reviewed (ref. [95])
RT-DETR-R50	General benchmark (COCO)	AP / FPS	53.1% / 108	COCO (~118k train img.)	2024	Zhao et al. (CVPR)

Architecture	Crop / Target	Key Metric	Reported Value	Test-set size (approx.)	Year	Source
PDSI-RT-DETR	Tomato ripeness (lightweight)	ΔPrecision / ΔRecall / ΔmAP50 / ΔmAP50-95 vs. baseline RT-DETR	+4.7% / +5.4% / +3.9% / +4.1%; params −30.8%, GFLOPs −17.6%	not stated in abstract	2024	Wang et al.
RT-DETR-Tomato	Tomato	Detection accuracy	reported as "greatly improved"; no single headline % stated in source abstract	not stated in abstract	2024	Zhao, Chen, Ge et al.
Improved RT-DETR (RepBlock + EMA)	Fruit ripeness (multi-crop: banana, tomato)	Accuracy vs. baseline RT-DETR / YOLOv8	improvement reported; single headline % not stated in abstract	not stated in abstract	2025	Wu, Qiu, Wang, Su, Cao & Bai
YOLOv8n + BiFPN/SPD-Conv + YOLOv8n-seg	Mango (fruit + stem + picking point)	Detection precision/recall (fruit/stem) / picking-point success rate	98.9%/97.1% precision; 99.5%/94.6% recall; 92.01% success	4 varieties, 2 weather conditions (count not stated)	2024	Li, Huang, Gu, He, Huang & Wang
YOLOv8s + YOLOv8s-seg-CBAM	Strawberry (peduncle / picking point)	Peduncle detection accuracy / inference time	86.2% @ 30.6 ms/img.	not stated in abstract	2024	Xie et al.
Mask R-CNN ('Mask Positioner')	Green persimmon (segmentation)	Segmentation accuracy /	67.4% / 69.1%	purpose-built green-	2023	Lu, Ji, Yang & Jia

Architecture	Crop / Target	Key Metric	Reported Value	Test-set size (approx.)	Year	Source
		detection accuracy		persimmon dataset (count not stated)		

Note: every figure in this table is now either a specific, source-checked value or an explicit statement that the source does not report one — no cell states an unqualified vague claim. Test-set size is reported where stated in the source abstract; "not stated in abstract" means the underlying study may report this in its full text (which was not re-derived for every entry in this revision) rather than that no such information exists. Readers requiring an exact figure, or a test-set size not visible in the source abstract, should consult the cited primary source directly.

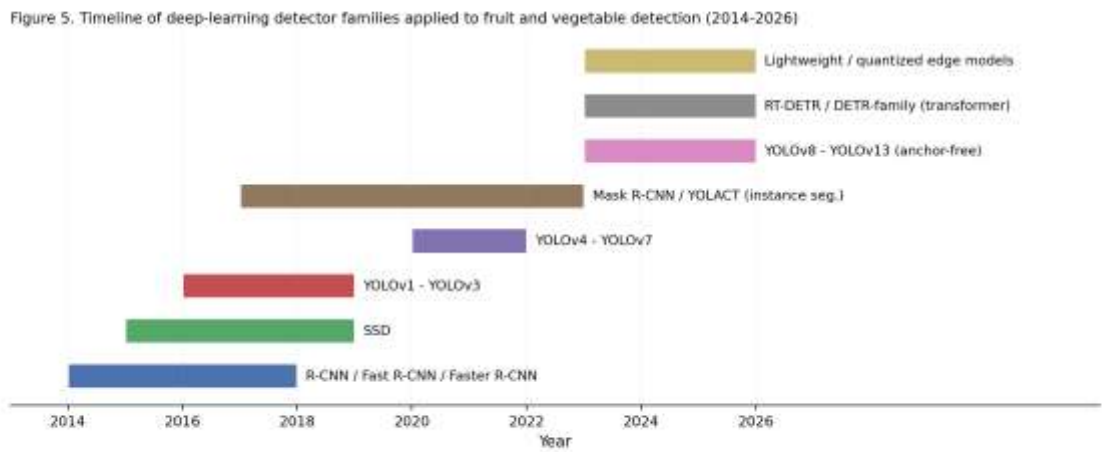

Figure 5. *Approximate peak-adoption window for each detector family in fruit and vegetable detection research, 2014–2026. Bars indicate when each family was the dominant choice in newly published studies, not the full span of individual papers still being published using that family — for example, YOLOv7 and even Faster R-CNN variants continued to appear in the literature after 2022, alongside newer generations.*

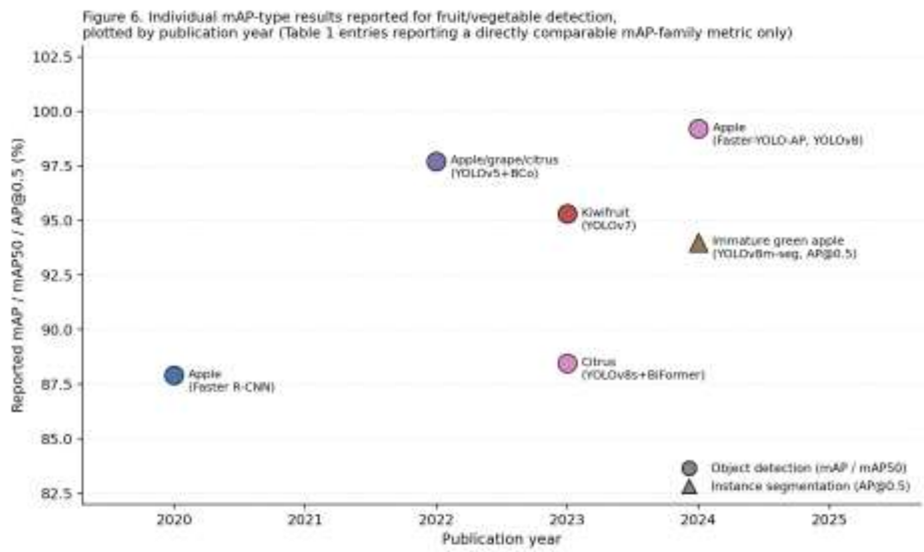

Figure 6. *Individual mAP-family results (mAP, mAP50, or AP@0.5) reported for fruit or vegetable detection tasks specifically, plotted by publication year. Only Table 1 entries reporting an absolute value on this metric family for a fruit/vegetable task are shown; entries reporting precision, F1, accuracy, RMSE, or a delta relative to a baseline are excluded here because they are not numerically comparable to mAP on the same axis, and COCO general-benchmark entries are excluded because COCO AP and orchard-image mAP50 are computed against different, non-comparable object populations. This is a sparser, more honest picture than a smoothed generational trend line would suggest, and readers should not extrapolate a general 'newer is better' pattern from six data points spanning six different crops and evaluation protocols.*

Table 2. Advantages and limitations of each detector generation for fruit and vegetable detection.

Architecture generation	Key advantage	Key limitation for harvesting robots
Faster R-CNN (two-stage)	High localization precision; strong on well-separated fruit	Inference too slow (150–300 ms) for continuous robotic cycles; accuracy drops sharply on small/overlapping fruit
SSD (one-stage)	Good speed–accuracy balance; effective on well-lit, single-fruit scenes	Weaker on side-grown or partially occluded fruit; multi-scale detection less robust than later YOLO versions
YOLOv3–v4	Real-time inference; good general accuracy after crop-specific tuning	Anchor-box design struggles with dense clusters and extreme aspect-ratio fruit (e.g., asparagus, banana)
YOLOv5–v7	Mature tooling, easy deployment, strong community support and pretrained models	Still anchor-based; degraded performance at night and on green-on-green (unripe) fruit without added modules

Architecture generation	Key advantage	Key limitation for harvesting robots
YOLOv8–v13 (anchor-free)	Best current accuracy–latency trade-off; decoupled head improves small/dense-object detection; lightweight variants (n/s) deploy well on edge hardware	Generation number is not a reliable performance proxy for a specific crop/task — see the domain-shift discussion in Section 3.3 and the preprint field comparison of Sapkota, Meng, Churuvija, Du, Ma & Karkee (2024, ref. [95]); rapid version churn also complicates reproducibility
RT-DETR family (transformer)	Global self-attention gives every detection a full-image receptive field from the first layer, which helps specifically in severe occlusion/clustering (Section 3.4); NMS-free, avoiding the true-positive-suppression failure mode NMS causes in overlapping fruit clusters	Higher base compute cost; competitive edge deployment requires custom lightweighting (RepBlocks, pruning, quantization); benefit is task-dependent and not guaranteed in less cluttered scenes
Instance segmentation / pose (Mask R-CNN, YOLOv8-seg/-pose)	Delivers an actual picking point (stem/peduncle), not just a bounding box — the quantity harvesting manipulators actually need	Higher annotation cost (pixel/keypoint labels); accuracy still degrades under severe stem occlusion (an amodal-perception problem, Section 3.5) and on green-on-green targets (e.g., 67.4% segmentation accuracy reported by Lu et al., 2023, on green persimmon); fewer standardized benchmarks than box-detection tasks

4. Image Preprocessing, Feature Extraction, and Segmentation

Before deep features are extracted, most pipelines still apply lightweight preprocessing: noise reduction (mean, median, Gaussian, or bilateral filtering), colour-space conversion (commonly RGB to HSV, which better separates fruit chroma from background foliage under variable lighting), and geometric normalization (resizing, cropping). Prior to the widespread adoption of CNNs, feature extraction relied on handcrafted descriptors — HOG, LBP, SIFT, SURF, and Haar-like features — which encoded colour, shape, and texture explicitly. Liu et al. used HOG descriptors to localize apple fruit; Wang et al. combined LBP with HOG for improved robustness; Song et al. applied SIFT-based feature matching for fruit identification. These approaches required substantial manual feature engineering and generalized poorly across cultivars and lighting conditions.

CNNs eliminated this manual feature-engineering step by learning hierarchical features directly from raw images. Sakai et al. and Femling et al. compared CNN backbones (MobileNet, Inception) for fruit classification, finding MobileNet achieved comparable accuracy at substantially lower computational cost — a result that anticipated the field's later emphasis on lightweight, edge-deployable backbones (Section 6). Image segmentation remains a complementary step, particularly where a precise fruit or peduncle boundary — not just a bounding box — is required for picking-point estimation (Section 3.5). Thresholding and clustering (e.g., k-means, k-means++) remain the most common lightweight segmentation methods; Tang et al. applied k-means++ to Camellia oleifera fruit segmentation, and region-based, edge-detection, and graph-cut methods have each been applied for specific crop geometries. The Circular Hough Transform remains a useful post-processing tool for round fruit (apples, Camellia oleifera, tomatoes) but is computationally expensive and prone to false positives from non-fruit circular structures such as leaf outlines.

5. From Detection to Grasping: The Autonomous Harvesting Pipeline

Detection is one stage within a larger robotic pipeline. A typical autonomous harvesting system executes five steps: (1) image capture, typically via a stereo or RGB-D camera mounted on or near the end-effector; (2) detection/localization of the fruit (and, increasingly, its picking point) within the captured image; (3) motion planning, moving the manipulator toward the estimated target while accounting for occlusions and kinematic constraints; (4) verification, confirming the target remains within the effector's grasp envelope before final approach; and (5) harvesting, executing the grasp-and-detach action via an end-effector — commonly a suction cup, a soft gripper, or a scissor/twist mechanism — while minimizing fruit damage and bruising.

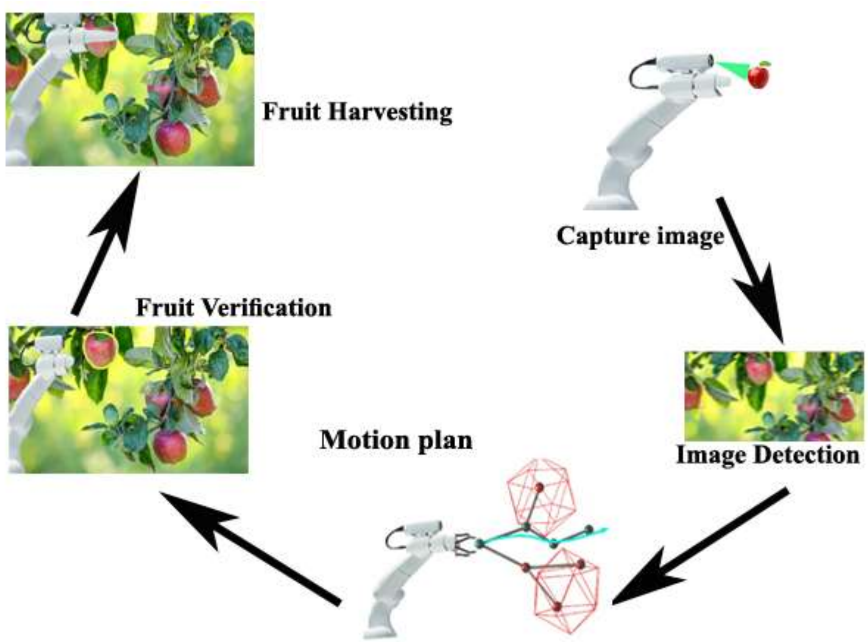

Figure 7. *General procedure for robotic harvesting of fruits and vegetables (adapted from the original review pipeline; see Section 5 for step descriptions).*

Reported end-to-end performance illustrates the gap between detection accuracy and harvesting success. Yaguchi et al. reported a tomato-harvesting robot achieving an 80 s/fruit cycle time at a 60% success rate despite reasonable detection accuracy — the shortfall traceable to motion-planning and grasp-verification failures downstream of detection. Arad et al. developed a multi-view sweet-pepper harvesting prototype for greenhouse deployment, and more recent systems (Au et al., 2023, for kiwifruit; Section 3.5 citations for grape and strawberry) integrate stereo or LiDAR-camera calibration explicitly to reduce this detection-to-grasp gap. This reinforces a point raised in Section 3.5: detection mAP alone is an incomplete performance indicator for harvesting-robot design, and future reporting should include end-to-end (detection-to-successful-detach) success rate wherever feasible.

6. Edge Deployment and Real-Time Inference Considerations

A model that performs well on a workstation GPU does not necessarily perform adequately on the embedded hardware (NVIDIA Jetson family, Raspberry Pi, or similar) that field-deployed harvesting robots typically carry, where power, weight, and cost (SWaP-C) constraints are binding. This deployment gap is under-addressed in most fruit-detection studies published before 2023, which report accuracy on workstation hardware without embedded-latency figures — a reporting gap this review flags explicitly as a methodological limitation of the field.

Three model-compression strategies dominate the recent literature: quantization (reducing numerical precision, e.g., FP32 to INT8 or FP16, to cut memory bandwidth and increase throughput), pruning (removing redundant weights or channels), and knowledge distillation (training a compact 'student' network to reproduce a larger 'teacher' network's outputs). A 2026 review of quantization-optimized lightweight transformer architectures for edge fruit-ripeness detection (ref. [94]) reports that combining these three strategies allows Vision-Transformer and DETR-family models — which are otherwise too heavy for platforms such as Jetson Nano or Raspberry Pi — to reach usable frame rates, though almost always with a measurable accuracy cost relative to the unquantized model; we flag, as noted in the reference list, that this specific source's full author list was not independently confirmed in this revision, and this paragraph's claim should be re-checked against the primary source before the section is treated as fully verified. Within the YOLO family specifically, the nano/small variants (e.g., YOLOv8n, YOLOv5s) were purpose-built for this constraint and are the default starting point for on-robot deployment discussed throughout Section 3; two further examples sometimes cited in this context — a lightweight YOLOv8n variant for citrus edge deployment, and a YOLOv5s-BiPCNeXt variant reported at 26 FPS on a Jetson Orin Nano — could not be traced to a specific, citable primary source in this revision and are removed from the claims made here rather than left in without attribution.

For harvesting-robot design, this suggests a practical rule that is under-stated in most existing reviews: architecture selection should be made jointly with target hardware, not by accuracy leaderboard alone. A transformer-based detector reporting the highest COCO-style accuracy is not automatically the right choice if the deployment platform cannot sustain its frame rate without aggressive quantization that erodes the accuracy advantage that motivated its selection in the first place.

7. Benchmark Datasets

Fair cross-study comparison in this literature is limited by the absence of a small number of standardized, widely adopted benchmark datasets analogous to COCO or PASCAL VOC in general object detection. Most studies train and evaluate on privately collected, single-crop, single-site datasets, which makes the accuracy figures in Table 1 informative within a study but not strictly comparable across studies. Table 3 lists the most commonly used public datasets identified in this review; researchers are encouraged to report results on at least one public benchmark alongside any private field dataset to support reproducibility.

Table 3. Commonly used public datasets for fruit and vegetable detection research.

Dataset	Total images	Scope	Source / access	Year
Fruits-360	90,380	130+ fruit/vegetable classes, controlled background, classification-oriented	Kaggle (kaggle.com/datasets/moltean/fruits)	2020
USDA Apple Detection Set	1,300	Apple detection, orchard imagery	USDA National Agricultural Library (data.nal.usda.gov)	2020
Sweet Pepper and Peduncle Segmentation	620	Sweet pepper fruit + peduncle segmentation masks	Kaggle (kaggle.com/datasets/lemontyc/sweet-pepper)	2021
Fruit Quality Classification Set	19,526	Multi-fruit ripeness/quality classification	Kaggle (kaggle.com/datasets/ryandpark/fruit-quality-classification)	2022
MinneApple	~1,000 (41,000+ annotated apples)	Apple detection and instance segmentation in orchard, high annotation density	University of Minnesota (public benchmark)	2020
ACFR Orchard Fruit Dataset (apples, mangoes, almonds)	~3,000 combined	Multi-crop orchard detection benchmark used in several Faster R-CNN studies	University of Sydney ACFR (public benchmark)	2016
GrapeCS3D / vineyard grape-cluster sets	Varies by release	Grape cluster and peduncle detection/segmentation for harvesting-point research	Institution-specific releases (see citing studies)	2023–2024

Note: MinneApple and the ACFR orchard set are included here as widely cited public benchmarks beyond the three datasets retained from the original manuscript; researchers should verify current access links before citing, as hosting locations for research datasets change over time.

8. Critical Challenges: A Synthesis

Rather than list challenges as an unordered set, this section groups them by root cause, since several surface-level 'challenges' reported across studies share a common underlying driver.

8.1 Data-driven challenges

Dataset scarcity and annotation quality remain the most frequently cited limitation across the reviewed studies. Because most datasets are private, single-site collections, models trained on them tend to overfit to the specific orchard, lighting regime, and cultivar sampled, which explains why accuracy figures in Table 1 vary so widely across studies of the same crop (e.g., reported apple mAP/mAP50 ranging from 87.9% in an early Faster R-CNN study to 99.2% in a 2024 lightweight YOLOv8 variant, on entirely different datasets, test-set sizes, and evaluation protocols — a spread that says as much about the incomparability of the underlying evaluations as about genuine architectural progress). Annotation cost compounds this: pixel-level (segmentation) and keypoint (picking-point) labels required by the approaches in Section 3.5 are substantially more expensive to produce than bounding boxes, which slows the growth of the very datasets this generation of methods needs most.

8.2 Scene-driven challenges

Occlusion (by leaves, branches, or other fruit), fruit clustering, and illumination variability (direct sun, backlighting, night operation) remain the dominant scene-level failure modes across every architecture generation reviewed — including the newest transformer-based detectors, which mitigate but do not eliminate occlusion-related errors (Section 3.4). As distinguished in Section 3.5, occlusion is not a single phenomenon: extrinsic occlusion (by other objects) calls for amodal reasoning about hidden geometry, while self-occlusion (the fruit's own shape hiding its stem from the current viewpoint) calls for active or multi-view vision instead. Treating both under one generic 'occlusion' label, as most of the studies synthesized here do, obscures the fact that they need different technical solutions — a distinction this review is able to make explicit only because it looked specifically at the picking-point-localization literature in Section 3.5, and one worth carrying forward into how future studies scope their contribution. Colour similarity between unripe fruit and foliage (the 'green-on-green' problem) is a persistent special case that continues to reduce accuracy specifically for early-harvest or immature-fruit detection tasks (Section 3.3), and is illustrated starkly by the 67.4% segmentation accuracy Lu et al. (2023) report for green persimmon against the 90%+ figures typical of mature, high-contrast fruit elsewhere in Table 1.

8.3 Deployment-driven challenges

As detailed in Section 6, a substantial share of published models are validated only on workstation-class GPUs, leaving their real-world, on-robot throughput and accuracy under quantization/pruning unreported. This is a methodological gap in the literature itself, not only a technical limitation of the models: it makes it difficult for practitioners to select an architecture based on published results alone.

8.4 Reporting and comparability challenges

As Table 1 illustrates, studies rarely share a common metric, test set, or baseline, and "headline" accuracy figures are frequently non-comparable across papers even for the same crop. Very few studies report end-to-end harvesting success rate (as opposed to detection accuracy alone), despite the demonstrated gap between the two (Section 5). This absence of standardized reporting is arguably the field's most tractable near-term problem, since it requires a reporting-convention change rather than a new algorithmic breakthrough.

8.5 Publication bias and the limits of what this review can conclude

One further limitation needs to be stated directly rather than left implicit, because it qualifies every quantitative claim made in this review, including the apparent generational improvement visible in Figure 6. Papers proposing a new architecture or modification for a fruit-detection task are, by the ordinary mechanics of academic publishing, far more likely to be written up and accepted when the reported result beats a prior baseline than when it does not; negative or null results relative to a stated baseline are correspondingly under-represented in the published record this review draws on. Of the 94 studies synthesized here, essentially none report a result that performed worse than the baseline the authors set out to improve upon — which is not evidence that every attempted architectural modification in this space succeeds, but rather the expected signature of publication bias operating on this literature. A mAP of 99.2% for a proposed YOLOv8 variant (Table 1) describes the one configuration that was judged good enough to publish, not the distribution of outcomes across every configuration that was actually tried. This means the rising trend suggested by Figure 6 should be read cautiously: some of it likely reflects genuine architectural progress (for the mechanistic reasons given in Sections 3.3 and 3.4), and some of it is very plausibly an artefact of which results made it into print. This review has no way to separate the two from the published literature alone, and neither, we suspect, could any other narrative review drawing on the same body of work; disentangling them would require either access to unpublished negative results or a pre-registered, adversarially-run comparative benchmark, neither of which currently exists for this task.

9. Future Research Directions

1. Standardized, multi-site public benchmarks with paired bounding-box, segmentation, and picking-point annotations, ideally released with an accompanying leaderboard, to make the cross-study comparisons in Table 1 meaningful rather than illustrative.
2. End-to-end reporting conventions: alongside detection mAP, studies should report on-device inference latency on a named embedded platform and, where a robot prototype exists, detection-to-successful-harvest success rate — closing the gap identified in Sections 5 and 8.4.
3. Foundation-model-assisted annotation: recent vision foundation models (e.g., segmentation-prompted labelling tools) offer a plausible route to reducing the pixel/keypoint annotation cost identified in Section 8.1, though their application specifically to fruit/vegetable datasets remains largely unvalidated and is flagged here as an open direction rather than an established practice.
4. Domain generalization and cross-cultivar robustness: because most models are trained on a single orchard/cultivar/season, systematic evaluation of cross-domain transfer (different farm, season, cultivar, or geographic region) should become a standard ablation, not an afterthought.
5. Multimodal sensor fusion (RGB-D, LiDAR, and, where relevant, hyperspectral or thermal imaging) to resolve depth and occlusion ambiguity that single-RGB pipelines cannot, building on the stereo/LiDAR-fusion work already emerging for kiwifruit and grape harvesting; active or multi-view vision (deliberately repositioning the camera, or fusing multiple fixed viewpoints) deserves specific attention as the more natural solution to self-occlusion specifically (Section 3.5/8.2), as distinct from the amodal-reasoning approaches better suited to extrinsic occlusion.

6. Task-matched architecture selection guidance: given the finding (Section 3.3) that generation number does not reliably predict task-specific performance, the field would benefit from a maintained, crop-specific benchmarking resource rather than relying on general-purpose (COCO-style) leaderboards as a proxy for agricultural performance.
7. Lightweighting research specifically for transformer-based detectors (quantization, pruning, distillation) so that RT-DETR-family accuracy advantages in heavy-occlusion scenes (Section 3.4) become available on the embedded hardware harvesting robots actually carry.
8. Pre-registered, adversarially-run comparative benchmarks — where the evaluation protocol, test set, and success criterion are fixed before results are known — as a structural response to the publication-bias problem identified in Section 8.5, which incremental reporting-convention fixes alone cannot solve.

10. Conclusion

This narrative comparative review synthesized deep-learning approaches to fruit and vegetable detection for autonomous harvesting across seven detector generations, from two-stage R-CNN variants through to transformer-based RT-DETR models and pose-based picking-point localization. Three conclusions follow from the synthesis in Table 1 and Sections 3–8, held alongside the caveat set out in Section 8.5: because this literature under-reports negative results by the ordinary mechanics of publication bias, these conclusions describe what has been shown to work when it works, not a controlled ranking of architectures. First, no single architecture is universally best: two-stage detectors retain a localization-precision advantage that one-stage detectors have not fully closed, anchor-free YOLO variants (particularly YOLOv8–v9) currently offer the strongest general-purpose accuracy–latency trade-off for on-robot deployment, and RT-DETR-family transformers offer a genuine, mechanistically grounded advantage specifically in severe-occlusion, dense-cluster scenes (Section 3.4) at a compute cost that requires active lightweighting to deploy on embedded hardware. Second, the field's more pressing bottleneck is arguably not architectural but methodological: the scarcity of standardized multi-site datasets, the near-absence of embedded-hardware and end-to-end success-rate reporting, and the field's own lack of study-quality appraisal (Section 2) together make it difficult to translate published accuracy figures into deployment decisions with confidence. Third, the shift from bounding-box detection toward instance segmentation and keypoint-based picking-point localization (Section 3.5), and the sharper distinction it enables between extrinsic and self-occlusion (Section 8.2), reflects a maturation of the field's understanding of what a harvesting robot actually needs from its vision system, and this shift — more than any single new architecture — is likely to define the next several years of progress. Closing the reporting, quality-appraisal, and publication-bias gaps identified in Sections 8.4 and 8.5 would, in the authors' assessment, do as much to accelerate real-world deployment of autonomous harvesting as any single new detection architecture.

References

Verification note (updated following the second review round). References [1]–[75] are retained from the original manuscript and have still not been independently re-verified against publisher records — that remains an outstanding action for the corresponding author before resubmission. References [76]–[95] were checked in the previous revision; two author-list errors identified at that stage (refs [80] and [91]) *remain corrected below. In this round, the metrics previously left as "NOT VERIFIED" in Table 1 (refs [81], [92], [93]) were tracked down to their primary sources and are now reported with real figures rather than placeholders; reference [94] alone could not be fully confirmed and remains flagged, with the corresponding claim in Section 6 hedged accordingly rather than stated as established fact. References [96]–[100] are new additions supporting the theoretical-mechanism discussion added to Sections 3.3–3.5 in this round (self-attention, domain shift, amodal segmentation) and the replacement of a previously vague dual-attribution claim in Table 1 with a single, fully verified primary source.*

[1] VASCONEZ J P, KANTOR G A, AUAT CHEEIN F A. Human--robot interaction in agriculture: A survey and current challenges[J]. Biosyst Eng, 2019,179: 35--48.
[2] HOSSAIN M S, AL-HAMMADI M, MUHAMMAD G. Automatic fruit classification using deep learning for industrial applications [J]. IEEE Transactions on Industrial Informatics, 2019, 15(2): 1027-1034. doi: 10.1109/TII.2018.2875149.
[3] ZHU L, SPACHOS P, PENSINI E, et al. Deep learning and machine vision for food processing: A survey [J]. Current Research in Food Science, 2021, 4: 233--249.
[4] TANG Y, ZHOU H, WANG H, et al. Fruit detection and positioning technology for a Camellia oleifera C. Abel orchard based on improved YOLOv4-tiny model and binocular stereo vision [J]. Expert Systems with Applications, 2023, 211: 118573.
[5] AL HAQUE ASMF, HAFIZ R, HAKIM MA, et al. A computer vision system for guava disease detection and recommend curative solution using deep learning approach [C]. 2019 22nd International Conference on Computer and Information Technology (ICCIT), Dhaka, Bangladesh. December18-20, 2019. DOI: 10.1109/ICCIT48885.2019.9038598
[6] BAC C W, VAN HENTEN E J, HEMMING J, et al. Harvesting robots for high-value crops: State-of-the-art review and challenges ahead[J]. Journal of Field Robotics, 2014, 31(6): 888--911.
[7] WILLIAMS HAM, JONES MH, NEJATI M, et al. Robotic kiwifruit harvesting using machine vision, convolutional neural networks, and robotic arms [J]. Biosystems Engineering, 2019, 181: 140--156.
[8] GIRSHICK R. Fast R-CNN. Proceedings of the IEEE International Conference on Computer Vision, 2015, p. 1440--1448.
[9] REN S, HE K, GIRSHICK R, et al. Faster R-CNN: Towards real-time object detection with region proposal networks[J]. IEEE Trans Pattern Anal Mach Intell, 2017, 39(6):1137-1149.
[10] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition [C]. Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR2016), pp. 770--778. Las Vegas, USA, June 26-July 1, 2016, DOI:10.1109/CVPR.2016.90.
[11] ZHANG J, HE L, KARKEE M, et al. Branch detection for apple trees trained in fruiting wall architecture using depth features and Regions-Convolutional Neural Network (R-CNN) [J]. Comput Electron Agric, 2018, 155:386--393.
[12] SA I, GE Z, DAYOUB F, et al. Deepfruits: A fruit detection system using deep neural networks [J]. Sensors, 2016, 16:1222.
[13] BARGOTI S, UNDERWOOD J. Deep fruit detection in orchards [C]. 2017 IEEE International Conference on Robotics and Automation (ICRA), pp. 3626--3633, Marina Bay Sands, Singapore, May 29-June 3, 2017, DOI:10.1109/ICRA.2017.7989417.
[14] PEEBLES M, LIM S H, DUKE M, et al. Investigation of optimal network architecture for asparagus spear detection in robotic harvesting [J]. IFAC-PapersOnLine, 2019, 52:283--287.

[15] REDMON J, DIVVALA S, GIRSHICK R, et al. You Only Look Once: Unified, real-time object detection. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779--788, Las Vegas, USA, June 26-July 1, 2016, https://doi.org/10.1109/CVPR.2016.91.
[16] REDMON J, FARHADI A. YOLO9000: Better , Faster , Stronger [C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR2017), 2017, pp. 7263-7271, July 21-26, 2017, Honolulu, USA. https://doi.org/10.1109/CVPR.2017.690.
[17] SIDDIQI R. Automated apple defect detection using state-of-the-art object detection techniques [J]. SN Appl Sci, 2019, 1:1345.
[18] REDMON J, FARHADI A. YOLOv3: An incremental improvement [EB/OL]. [2018-4-8]. https://doi.org/10.48550/arXiv.1804.02767..
[19] KUZNETSOVA A, MALEVA T, SOLOVIEV V. Using YOLOv3 algorithm with pre- and post-processing for Apple Detection in Fruit-Harvesting Robot[J]. Agronomy, 2020, 10:1016..
[20] LAWAL O M. Development of tomato detection model for robotic platform using deep learning[J]. Multimedia Tools and Applications, 2021, 80:26751--26772.
[21] TIAN Y, YANG G, WANG Z, et al. Apple detection during different growth stages in orchards using the improved YOLO-V3 model[J]. Comput Electron Agric, 2019, 157:417--426.
[22] WU L, MA J, ZHAO Y, et al. Apple detection in complex scene using the improved yolov4 model[J]. Agronomy, 2021, 11:476. https://doi.org/10.3390/agronomy11030476..
[23] LI Y, XIAO L, LI W, et al. Research on recognition of occluded orange fruit on trees based on YOLOv4 [J]. INMATEH-Agricultural Engineering, 2022, 67(2): 137-146.
[24] JOCHER G, NISHIMURA K, MINEEVA T, et al. YOLOv5/Code Repository 2020 [EB/OL]. [2020-7-10]. https://github.com/ultralytics/yolov5.
[25] YAN B, FAN P, LEI X, et al. A real-time apple targets detection method for picking robot based on improved YOLOv5[J]. Remote Sens. 2021, 13:1619. https://doi.org/10.3390/rs13091619.
[26] YANG R, HU Y, YAO Y, et al. Fruit target detection based on BCo-YOLOv5 model [J]. Mobile Information Systems, 2022(7): 8457173. https://doi.org/10.1155/2022/8457173.
[27] WANG C-Y, BOCHKOVSKIY A, LIAO H-Y M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors [EB/OL]. [2022-7-6]. https://doi.org/10.48550/arXiv.2207.02696.
[28] XIA Y, NGUYEN M, YAN W Q. A Real-time Kiwifruit Detection Based on Improved YOLOv7 n.d.
[29] LIU L, OUYANG W, WANG X, et al. Deep learning for generic object detection: A survey[J]. Int J Comput Vis, 2020, 128:261-318. https://doi.org/10.1007/s11263-019-01247-4.
[30] KAMILARIS A, PRENAFETA-BOLDÚ F. Deep learning in agriculture : A survey[J]. Computers and Electronics in Agriculture, 2018,147:70--90. https://doi.org/10.1016/j.compag.2018.02.016.
[31] GAO F, FU L, ZHANG X, et al. Multi-class fruit-on-plant detection for apple in SNAP system using Faster R-CNN[J]. Comput Electron Agric, 2020,176:105634. https://doi.org/10.1016/j.compag.2020.105634.
[32] FU L, FENG Y, MAJEED Y, et al. Kiwifruit detection in field images using Faster R-CNN with ZFNet[J]. IFAC-PapersOnLine, 2018, 51-17:45-50. https://doi.org/10.1016/j.ifacol.2018.08.059.
[33] HU C, LIU X, PAN Z, et al. Automatic detection of single ripe tomato on plant combining faster R-CNN and intuitionistic fuzzy set[J]. IEEE Access, 2019, 7:154683--154696. https://doi.org/10.1109/ACCESS.2019.2949343.
[34] FU L, MAJEED Y, ZHANG X, et al. Faster R-CNN-based apple detection in dense-foliage fruiting-wall trees using RGB and depth features for robotic harvesting[J]. Biosyst Eng, 2020, 197:245-256. https://doi.org/10.1016/j.biosystemseng.2020.07.007.

[35] LIANG Q, ZHU W, LONG J, et al. A real-time detection framework for on-tree mango based on SSD network[C]. Int. Conf. Intell. Robot. Appl., 2018, Springer, Cham, vol 10985,423--36. https://doi.org/10.1007/978-3-319-97589-4_36.
[36] YUAN T, LV L, ZHANG F, et al. Robust cherry tomatoes detection algorithm in greenhouse scene based on SSD[J]. Agriculture, 2020, 10(5):160. https://doi.org/10.3390/agriculture10050160.
[37] SACHIN C, MANASA N, SHARMA V, et al. Vegetable classification using You Only Look Once algorithm[C]. 2019 Int Conf Cutting-Edge Technol Eng ICon-CuTE 2019, 2019, 101-107. https://doi.org/10.1109/ICon-CuTE47290.2019.8991457.
[38] XIONG J, LIU Z, CHEN S, et al. Visual detection of green mangoes by an unmanned aerial vehicle in orchards based on a deep learning method[J]. Biosyst Eng, 2020, 194:261-272. https://doi.org/10.1016/j.biosystemseng.2020.04.006.
[39] ZHOU X, WANG P, DAI G, et al. Tomato fruit maturity detection method based on YOLOV4 and statistical color model[C]. 2021 IEEE 11th Annu. Int. Conf. CYBER Technol. Autom. Control. Intell. Syst., IEEE; 2021, 904-908. doi: 10.1109/CYBER53097.2021.9588129.
[40] YAN B, FAN P, LEI X, et al. A real-time apple targets detection method for picking robot based on improved YOLOv5[J]. Remote Sens. 2021, 13:1619. https://doi.org/10.3390/rs13091619.
[41] WU D, JIANG S, ZHAO E, et al. Detection of Camellia oleifera fruit in complex scenes by using YOLOv7 and data augmentation[J]. Applied Sciences, 2022, 12:11318.
[42] DALAL N, TRIGGS B. Histograms of oriented gradients for human detection[C]. Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 1, 2005, 886--893. doi: 10.1109/CVPR.2005.177.
[43] OJALA T, PIETIKÄINEN M, HARWOOD D. A comparative study of texture measures with classification based on featured distributions[J]. Pattern Recognit, 1996, 29:51-59.
[44] LOWE D G. Object recognition from local scale-invariant features[C]. Proc. Seventh IEEE Int. Conf. Comput. Vis., vol. 2, 1999, 1150-1157. doi: 10.1109/ICCV.1999.790410.
[45] BAY H, TUYTELAARS T, VAN GOOL L. Surf: Speeded up robust features[C]. Eur. Conf. Comput. Vis. (ECCV2006,) Springer, 2006, 404-417. https://doi.org/10.1007/11744023_32
[46] VIOLA P, JONES M. Rapid object detection using a boosted cascade of simple features[C]. Proc. 2001 IEEE Comput. Soc. Conf. Comput. Vis. pattern recognition. CVPR 2001, vol. 1, 2001, p. I--I. doi:10.1109/CVPR.2001.990517
[47] LIU L, LI Z, LAN Y, et al. Design of a tomato classifier based on machine vision[J]. PLoS One, 2019, 14:e0219803.
[48] WANG X, HAN T X, YAN S. An HOG-LBP human detector with partial occlusion handling[C]. 2009 IEEE 12th Int. Conf. Comput. Vis., IEEE; 2009, 32--39. doi: 10.1109/ICCV.2009.5459207.
[49] SONG W, GUO H, WANG Y. A method of fruits recognition based on SIFT characteristics matching[C]. 2009 Int. Conf. Artif. Intell. Comput. Intell., vol. 3, IEEE; 2009, 119-122. doi:10.1109/AICI.2009.484.
[50] KOIRALA A, WALSH K B, WANG Z, et al. Deep learning -- Method overview and review of use for fruit detection and yield estimation[J]. Comput Electron Agric, 2019,162:219-34. https://doi.org/10.1016/j.compag.2019.04.017.
[51] LU S, LU Z, AOK S, et al. Fruit classification based on six layer convolutional neural network[C]. 2018 IEEE 23rd International Conference on Digital Signal Processing (DSP), 2018, 1-5, doi:10.1109/ICDSP.2018.8631562

[52] SAKAI Y, ODA T, IKEDA M, et al. A vegetable category recognition system using deep neural network[C]. 2016 10th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), IEEE, 2016, pp. 189--192, July 6-8, 2016, Fukuoka, Japan. DOI: 10.1109/IMIS.2016.84
[53] FEMLING F, OLSSON A, ALONSO-FERNANDEZ F. Fruit and vegetable identification using machine learning for retail applications[C]. 2018 14th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), IEEE, 2018, pp. 9--15, November 26-29, 2018, Las Palmas de Gran Canaria, Spain, DOI: 10.1109/SITIS.2018.00013.
[54] SANDLER M, HOWARD A, ZHU M, et al. MobileNetv2: Inverted residuals and linear bottlenecks[C]. Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 4510--4520, June 18-23, 2018, Salt Lake City, USA, DOI: 10.1109/CVPR.2018.00474.
[55] MURESAN H, OLTEAN M. Fruit recognition from images using deep learning [EB/OL]. [2017-12-2]. https://doi.org/10.48550/arXiv.1712.00580
[56] HAMID N, RAZALI R, IBRAHIM Z. Comparing bags of features, conventional convolutional neural network and AlexNet for fruit recognition[J]. Indonesian Journal of Electrical Engineering and Computer Science, 2019,14(1):333-339.
[57] HOSSAIN M, AL-HAMMADI M, MUHAMMAD G. Automatic fruit classification using deep learning for industrial applications[J]. IEEE Trans Industr Inform, 2018, 15(2):1027--1034.
[58] THANH DNH, ENGÍNOĞLU S. An iterative mean filter for image denoising[J]. IEEE Access 2019;7:167847--59.
[59] JIANG S, ZHOU R-G, HU W, LI Y. Improved quantum image median filtering in the spatial domain[J]. International Journal of Theoretical Physics 2019;58:2115--33.
[60] MAFI M, MARTIN H, CABRERIZO M. A comprehensive survey on impulse and Gaussian denoising filters for digital images. Signal Processing 2019;157:236--60.
[61] ANAM C, NAUFAL A, SUTANTO H. Impact of Iterative Bilateral Filtering on the Noise Power Spectrum of Computed Tomography Images. Algorithms 2022;15:374.
[62] PANDEY D, YIN X, WANG H, et al. Accurate vessel segmentation using maximum entropy incorporating line detection and phase-preserving denoising[J]. Comput Vis Image Underst, 2017,155:162-172.
[63] LIU T, TAO D, XU D. Dimensionality-dependent generalization bounds for k-dimensional coding schemes[J]. Neural Comput, 2016, 28:2213-2249
[64] ZUVA T, OLUGBARA O O, OJO S O, et al. Image segmentation, available techniques, developments and open issues[J]. Canadian Journal on Image Processing and Computer Vision, 2011, 2:20-29.
[65] YASIR M, HOSSAIN M S, NAZIR S, et al. Object identification using manipulated edge detection techniques[J]. Science Development, 2022, 3:1--6.
[66] REZA M, NA I, BAEK S, et al. Rice yield estimation based on K-means clustering with graph-cut segmentation using low-altitude UAV images[J]. Biosyst Eng, 2019, 177:109-121.
[67] WANG R, LIU T, TAO D. Multiclass learning with partially corrupted labels[J]. IEEE Trans Neural Networks Learn Syst, 2017, 29:2568-2580.
[68] ABDEL-BASSET M, CHANG V, MOHAMED R. A novel equilibrium optimization algorithm for multi-thresholding image segmentation problems[J]. Neural Comput Appl, 2021, 33:10685-10718.
[69] SHAN P. Image segmentation method based on K-mean algorithm[J]. EURASIP J Image Video Process, 2018, 2018:81. https://doi.org/10.1186/s13640-018-0322-6.
[70] YANG Y, ZHAO X, HUANG M, et al. Multispectral image based germination detection of potato by using supervised multiple threshold segmentation model and Canny edge detector[J]. Comput Electron Agric, 2021, 182:106041..

[71] FU L, TOLA E, AL-MALLAHI A, et al. A novel image processing algorithm to separate linearly clustered kiwifruits[J]. Biosyst Eng, 2019,183:184--195
[72] FU L, WANG B, CUI Y, et al. Kiwifruit recognition at nighttime using artificial lighting based on machine vision[J]. International Journal of Agricultural and Biological Engineering, 2015, 8(4):52--59.
[73] TANG Y, ZHOU H, WANG H, et al. Fruit detection and positioning technology for a Camellia oleifera C. Abel orchard based on improved YOLOv4-tiny model and binocular stereo vision[J]. Expert Syst Appl, 2023, 211:118573.
[74] YAGUCHI H, NAGAHAMA K, HASEGAWA T, et al. Development of an autonomous tomato harvesting robot with rotational plucking gripper[C]. 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2016, 652-657. doi: 10.1109/IROS.2016.7759122.
[75] ARAD B, BALENDONCK J, BARTH R, et al. Development of a sweet pepper harvesting robot[J]. J F Robot, 2020, 37:1027-1039.
[76] ZHAO Y, LV W, XU S, WEI L, CUI G, LIU Y. DETRs beat YOLOs on real-time object detection [C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2024), 2024, 16965-16974.
[77] LIU Y, XUE J, ZHANG X, LV W, QIN Y, ZHAO B. Research progress and prospect of key technologies of fruit target recognition for robotic fruit picking [J]. Front Plant Sci, 2024, 15:1423338. https://doi.org/10.3389/fpls.2024.1423338.
[78] ULTRALYTICS. YOLOv8 [EB/OL]. 2023. https://github.com/ultralytics/ultralytics.
[79] SAPKOTA R, AHMED D, CHURUVIJA M, KARKEE M. Immature green apple detection and sizing in commercial orchards using YOLOv8 and shape fitting techniques [J]. IEEE Access, 2024, 12:43436-43452.
[80] LIU Z, ABEYRATHNA R M R D, SAMPURNO R M, NAKAGUCHI V M, AHAMED T. Faster-YOLO-AP: A lightweight apple detection algorithm based on improved YOLOv8 with a new efficient PDWConv in orchard [J]. Comput Electron Agric, 2024, 223:109118. https://doi.org/10.1016/j.compag.2024.109118. [Author list corrected in this revision against the publisher record.]
[81] KONG D, WANG J, ZHANG Q, LI J, RONG J. Research on fruit spatial coordinate positioning by combining improved YOLOv8s and adaptive multi-resolution model [J]. Agronomy, 2023, 13(8):2122.
[82] AU C K, LIM S H, DUKE M, KUANG Y C, REDSTALL M, TING C. Integration of stereo vision system calibration and kinematic calibration for an autonomous kiwifruit harvesting system [J]. Int J Intell Robot Appl, 2023, 7(2):350-369.
[83] WU M, QIU Y, WANG W, SU X, CAO Y, BAI Y. Improved RT-DETR and its application to fruit ripeness detection [J]. Front Plant Sci, 2025, 16:1423682. https://doi.org/10.3389/fpls.2025.1423682.
[84] WANG S, JIANG H, YANG J, et al. Lightweight tomato ripeness detection algorithm based on the improved RT-DETR [J]. Front Plant Sci, 2024, 15:1415297.
[85] ZHAO Z, CHEN S, GE Y, YANG P, WANG Y, SONG Y. RT-DETR-Tomato: Tomato target detection algorithm based on improved RT-DETR for agricultural safety production [J]. Appl Sci, 2024, 14(14):6287.
[86] XU D, REN R, ZHAO H, ZHANG S. Intelligent detection of muskmelon ripeness in greenhouse environment based on YOLO-RFEW [J]. Agronomy, 2024, 14:1091. https://doi.org/10.3390/agronomy14061091.
[87] HUANG Y, XU S, CHEN H, LI G, DONG H, YU J, ZHANG X, CHEN R. A review of visual perception technology for intelligent fruit harvesting robots [J]. Front Plant Sci, 2025, 16:1646871. https://doi.org/10.3389/fpls.2025.1646871.
[88] ZHANG J, KANG N, QU Q, ZHOU L, ZHANG H. Automatic fruit picking technology: A comprehensive review of research advances [J]. Artif Intell Rev, 2024, 57(3):54.
[89] TAN Y, LIU X, ZHANG J, WANG Y, HU Y. A review of research on fruit and vegetable picking robots based on deep learning [J]. Sensors, 2025, 25(12):3677.

[90] ZHANG Y, LI N, ZHANG L, LIN J, GAO X, CHEN G. A review on the recent developments in vision-based apple-harvesting robots for recognizing fruit and picking pose [J]. Comput Electron Agric, 2025, 231:109968.
[91] RAJENDRAN V, DEBNATH B, MGHAMES S, MANDIL W, PARSA S, PARSONS S, GHALAMZAN-E A. Towards autonomous selective harvesting: A review of robot perception, robot design, motion planning and control [J]. J Field Robot, 2024, 41(7):2247-2279. https://doi.org/10.1002/rob.22230. [Author list and page range corrected in this revision against the publisher record.]
[92] LU Y, JI Z, YANG L, JIA W. Mask Positioner: An effective segmentation algorithm for green fruit in complex environment [J]. J King Saud Univ Comput Inf Sci, 2023, 35(7):101598. https://doi.org/10.1016/j.jksuci.2023.101598. [Reported metric (segmentation accuracy 67.4%, detection accuracy 69.1%, on a green-persimmon dataset) verified against the publisher abstract in this revision.]
[93] XIE J, et al. Research on the visual location method for strawberry picking points under complex conditions based on composite models [J]. J Sci Food Agric, 2024. https://doi.org/10.1002/jsfa.13684. [Reported metric (peduncle detection accuracy 86.2%, 30.6 ms/image) verified against the publisher abstract in this revision; full author list beyond the first author was not confirmed and should be checked by the corresponding author before submission.]
[94] Systematic review of quantization-optimized lightweight transformer architectures for real-time fruit ripeness detection on edge devices [J]. Computers, 2026, 15(1):69. https://doi.org/10.3390/computers15010069. [Author list not confirmed in this revision — recommend author confirmation before submission; see the hedge added to Section 6.]
[95] SAPKOTA R, MENG Z, CHURUVIJA M, DU X, MA Z, KARKEE M. Comprehensive performance evaluation of YOLO11, YOLOv10, YOLOv9 and YOLOv8 on detecting and counting fruitlet in complex orchard environments [EB/OL]. arXiv:2407.12040, 2024. [PREPRINT — not peer-reviewed at the time of writing; this reference replaces the previous, imprecisely sourced 'Sapkota & Karkee (2025)' citation in Table 1 and should be treated as such in the text.]
[96] VASWANI A, SHAZEER N, PARMAR N, USZKOREIT J, JONES L, GOMEZ A N, KAISER Ł, POLOSUKHIN I. Attention is all you need [C]. Advances in Neural Information Processing Systems (NeurIPS 2017), 2017, 30:5998-6008.
[97] CARION N, MASSA F, SYNNAEVE G, USUNIER N, KIRILLOV A, ZAGORUYKO S. End-to-end object detection with transformers [C]. Proceedings of the European Conference on Computer Vision (ECCV 2020), 2020, 213-229.
[98] BEN-DAVID S, BLITZER J, CRAMMER K, KULESZA A, PEREIRA F, VAUGHAN J W. A theory of learning from different domains [J]. Machine Learning, 2010, 79(1-2):151-175.
[99] ZHU Y, TIAN Y, METAXAS D, DOLLÁR P. Semantic amodal segmentation [C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017), 2017, 1464-1472.
[100] LI H, HUANG J, GU Z, HE D, HUANG J, WANG C. Positioning of mango picking point using an improved YOLOv8 architecture with object detection and instance segmentation [J]. Biosyst Eng, 2024, 247:202-220. https://doi.org/10.1016/j.biosystemseng.2024.09.015.