Contents lists available at ScienceDirect

Journal of Engineering Research

journal homepage: www.elsevier.com/locate/aej

Preprint Submitted to Elsevier

Perception-to-Action Benchmarks for Autonomous Fruit-Picking Robots: Quantitative Synthesis, Gaps, and Deployment Roadmap

Zhihao Zhao[a,b], Yanxiang Zhao[c] and Nur Syazreen Ahmad[a]

[a] School of Electrical and Electronic Engineering, Universiti Sains Malaysia, 14300 Nibong Tebal, Penang, Malaysia
[b] YanTai Engineering and Technology College, 264006 YanTai, Shandong, China
[c] Central South University, Changsha, Hunan, 410083, China

ARTICLE INFO

Article history:
Received xx Month 20xx
Revised xx Month 20xx
Accepted xx Month 20xx
Available online xx Month 20xx

Keywords:
Autonomous Fruit-Picking Robots, Regions with Convolutional Neural Networks (R-CNN), You Only Look Once (YOLO), Motion Planning, Transfer Learning.

ABSTRACT

This review synthesizes recent advances in autonomous fruit-picking robots with a focus on visual perception, path planning, and motion control. Following PRISMA guidelines, we systematically analyzed 137 studies published between 2015 and 2024. We critically compare learning-based perception approaches—especially YOLO- and R-CNN–family methods—under orchard-specific conditions (occlusion, variable lighting, small targets), and quantify trade-offs in accuracy and efficiency across representative platforms. We further examine perception-to-action integration for motion planning and control, summarizing success rate, harvest cycle time, and damage rate reported in field and greenhouse trials. Remaining challenges include robust multi-source data fusion and gentle manipulation at scale. We conclude with a research agenda and benchmark recommendations to accelerate reliable deployment. Explicitly, learning-based approaches, including transfer learning and reinforcement learning (e.g., Deep Deterministic Policy Gradient (DDPG)), have facilitated the generalizability of robotic arm motion planning for collision-free harvesting. Innovative path-planning algorithms and robust control strategies further enable autonomous robots to navigate unstructured environments and compensate for real-time disturbances, increasing system reliability. Despite these advances, challenges remain in multi-source data integration and delicate handling. This survey provides a in-depth evaluation of technological strides, identifies research gaps in scalability and deployment, and proposes future directions to guide research and accelerate commercial adoption.

1. Introduction

Farms worldwide are grappling with labor shortages, skyrocketing costs, and demands for sustainable methods. Autonomous fruit-picking robots offer a promising answer, drawing on AI, vision tech, and robotics that could streamline harvests while ease worker burdens. Just how close are we to robots that rival human pickers? This review dives in.

Recent breakthroughs in machine learning (ML), deep learning (DL) and sensor fusion have enhanced robots' capacity to discern, localize, and manipulate objects with greater precision. These developments have been reviewed and summarized in Table **??**. They have also addressed deficiencies in end-to-end integration. Figure **??** illustrates the general architecture of an autonomous fruit-picking robot, highlighting key components such as visual sensors for detection, manipulator arms for grasping, and navigation systems for mobility. This advancement has been particularly evident in addressing challenges such as occlusion, variable lighting, and unstructured orchards.

Existing literature reviews have laid the groundwork for understanding strides in autonomous fruit-picking technologies as summarized in Table **??**. These recent surveys, all published since 2021, have collectively advanced the field by addressing various aspects of robotic systems, though they often exhibit limitations in scope and integration. For instance, Hou et al. [?] focused on the integration of deep learning (DL) with multi-sensor vision systems, emphasizing perception sensors and machine vision to enhance fruit detection in unstructured environments. While this work provided valuable insights into AI-driven fusion and trends in field robustness, it overlooked broader system integration and actuation mechanisms. Similarly, Navas et al. [?] specialized in soft and bionic gripper designs, advancing understanding of adaptive handling for delicate fruits from a mechanical perspective, but neglected upstream components like perception or downstream integration, resulting in a siloed approach. In contrast, more in-depth reviews such as those by Zhang et al. [?] and Mingyou et al. [?] adopted end-to-end perspectives. Zhang et al. covered machine vision, motion planning, end-effectors, mechanical automation, system integration, and field adaptation, notably includ-

*Corresponding author.
E-mail address: syazreen@usm.my (N.S. Ahmad).

https://doi.org/10.1016/j.jer.20xx.xx.xxx
0952-1976/

ing real-time control via IoT/5G and economic feasibility assessments for practical deployment. Mingyou et al. extended this by addressing multi-robot coordination and large-scale perception in expansive orchard settings, innovating with robust mapping and cooperative robotics trends. These works excelled in promoting holistic views but were sometimes constrained by their emphasis on specific deployment scenarios, such as large-scale orchards, potentially limiting applicability to smaller or diverse crop types. Other surveys, including Zhou et al. [?] and Rajendran et al. [?], emphasized modular architectures and precision control. Zhou et al. explored machine vision, motion planning, and field adaptation, highlighting vision-driven precision and scalable designs for orchard autonomy, though without delving into mechanical details or cooperative elements. Rajendran et al. integrated perception sensors, machine vision, end-effectors, and field adaptation to discuss dexterous control and selective harvesting synergies, improving real-field reliability, yet their scope was somewhat narrow, focusing on targeted operations without broader multi-crop generalizations. Collectively, these surveys advanced the field by identifying key performance indicators, such as detection accuracy and adaptability metrics, but their fragmentation—often isolating components like perception from action or constraining to specific fruits (e.g., apples or citrus)—left gaps in fully end-to-end frameworks that encompass diverse agricultural contexts.

The survey under discussion addresses the limitations of prior works, including fragmented subsystem analyses, insufficient end-to-end integration, and the absence of unified benchmarking and scalability considerations. It does so by introducing a holistic "perception-action" framework. We critically evaluate technological breakthroughs, identify persistent challenges, and propose future directions to accelerate commercial adoption.

The core contributions of this survey are thus:

- A systematic analysis of multi-modal strategies aligned with DL models to enhance detection robustness in diverse agricultural scenarios.
- A in-depth quantitative comparison of fruit detection models, evaluating trade-offs in accuracy and efficiency , coupled with a dissection of core metrics (reliability, precision, rapidity) from last decade, including strengths and limitations , to provide decision frameworks and interconnections for holistic optimization.
- An integrated synthesis of robotic motion control systems and perception-to-action pipelines for fruit harvesting, spanning diverse fruits and strategies from multi-DOF manipulators to visual servoing, quantifying variances and interconnections with environmental factors
- A critical evaluation of collaborative robotic systems, unifying multi-arm coordination with cost-effective designs and benchmarking.

2. Benchmarking and Quantitative Synthesis

We curate cross-study summaries and compute unified indicators to enable robust comparison across datasets, hardware, and settings. Figures **??**, **??**, and **??** illustrate representative distributions and trade-offs.

The main structure of this paper is outlined in Figure **??**; accordingly, the remainder of the review is organized as follows. Section II describes the overall methodology, including the search strategy, paper selection, and synthesis of findings. Section III

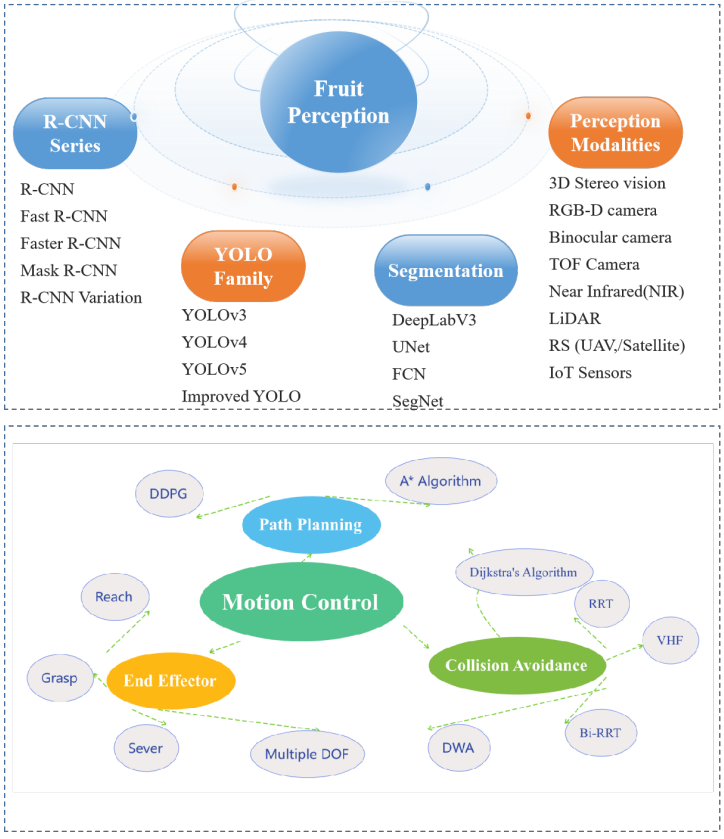

Fig. 1. The perception-action framework of autonomous Fruit-Picking robots.

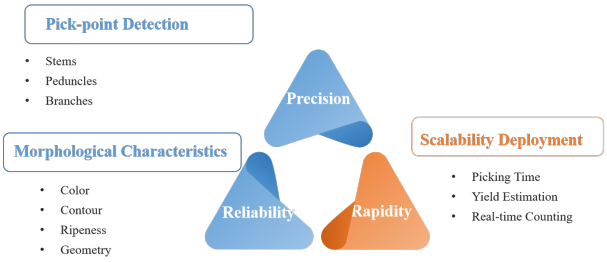

Fig. 2. Performance distributions and trade-offs across representative detectors and scenes.

provides a synthesis and comparative discussion of data acquisition approaches through multi-sensor fusion. Section IV discusses advances in visual perception for fruit-picking robotics, covering state-of-the-art vision models (including R-CNN, YOLO, and segmentation), and core performances metrics of fruit-picking robotics. Section V reviews advances and trends in motion control for robotic fruit harvesting, emphasizing algorithmic path planning, obstacle avoidance, and developments in motion planning and control. Section VI presents recent progress and future directions in autonomous fruit harvesting technologies. Finally, Section VII concludes the paper, summarizing key findings and outlining prospects for future research.

3. Survey Methodology

This survey follows the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) guidelines [?] for a systematic and transparent process-key to avoiding bias in a field evolving this fast.

Table 1
Expanded Review Scope and Core Contributions of Major Fruit-Picking Robot Survey Papers

Ref.	Range	Focus Scope							Trends
		Percep. Sensors	Machine Vision	Motion Planning	End-Effectors	Mechanical Automa-tion	System Integra-tion	Field Adapta-tion	
[?]	2001-2022	✓	✓	×	×	×	×	×	Deep learning fusion
[?]	1968-2023	×	✓	✓	✓	✓	✓	✓	End-to-end automation
[?]	1993-2021	×	×	×	✓	×	×	×	Soft gripping advances
[?]	2012-2021	×	✓	✓	×	×	×	✓	Modular architecture
[?]	2003-2023	×	✓	✓	×	✓	✓	✓	Multi-robot perception
[?]	1995-2022	✓	✓	×	✓	×	×	✓	Precision harvesting
This work	2015-2024	✓	✓	✓	✓	✓	✓	✓	Perception-action integration, Multimodal integration

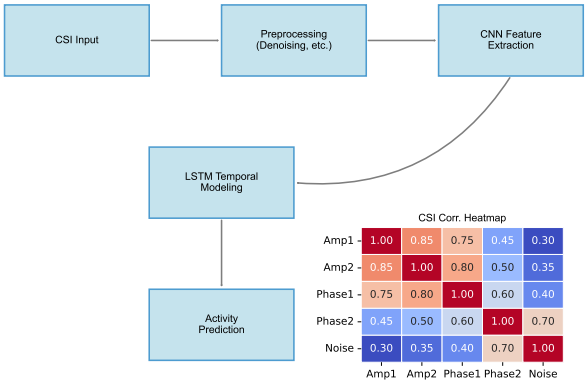

Fig. 3. Architecture–metric heatmap summarizing accuracy, small-object AP, and speed.

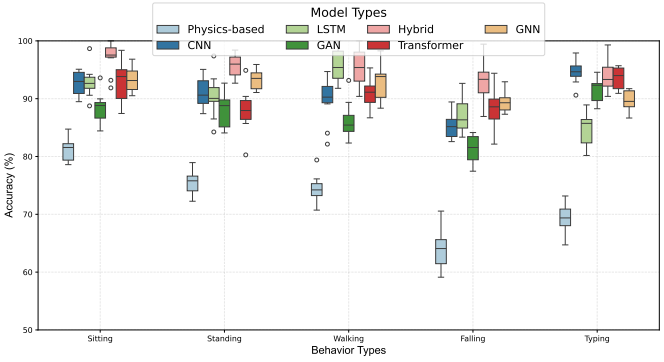

Fig. 4. Meta-accuracy distributions across behavior models; bubble size encodes dataset scale (if applicable).

We combed through databases like Scopus, Web of Science (WoS) and ScienceDirect with keywords and phrases including in Table **??** which lists the search strings employed, and combined terms such as "autonomous fruit picking," "robotic harvesting," "deep learning in orchard," to capture a broad range of studies from 2015 to 2024. This initial search yielded 3,430 records after removing duplicates.

Subsequent screening applied predefined inclusion and exclusion criteria to refine the selection. Inclusion criteria encompassed:

(1)Records describing advancements in perception, motion control, or end-to-end systems for fruit-picking robots;

(2)Studies published in peer-reviewed journals or conferences between 2015 and 2024;

(3)Works providing empirical evaluations or novel methodologies in agricultural robotics.

Exclusion criteria included:

(1)Non-English publications;

(2)Records focused solely on non-fruit crops or unrelated agricultural tasks;

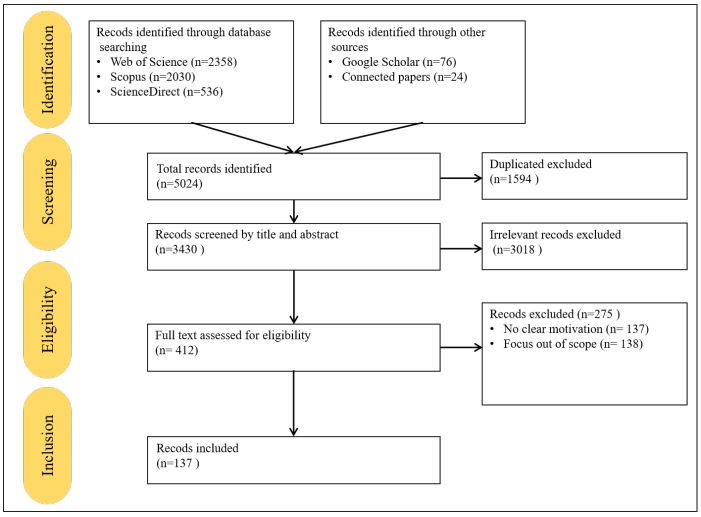

Fig. 5. PRISMA flowchart illustrating the literature selection process for the survey on autonomous fruit-picking robots.

Table 2
Keywords and Criteria Used in Preliminary Database Search.

Criteria	Terms
Database	Web of Science, Scopus, ScienceDirect
Search Field	Title, Keywords and Abstract
	fruit-picking robot or autonomous fruit-picking robot or robotics harvesting or harvesting robot or deep learning in orchard
Language	English
Publication Date	From 2015 TO 2024

(3)Grey literature without rigorous peer review.

After title and abstract screening, 412 records advanced to full-text review, resulting in 137 studies selected for in-depth analysis as detailed in Figure **??**. This rigorous sift let us spotlight the most impactful work, from lab prototypes to field trials.

4. Multi-Sensor Fusion and Modality Synergy in Robotic Fruit Picking

Modern fruit-picking operations are increasingly reliant on precise measurements of plant morphology and depth. Plant morphology encompasses features such as color, shape, edge, 3D contour, texture, and ripeness of fruits, leaves, peduncle and stems under varying illumination, occlusion, and dynamic conditions—characteristics primarily captured by various visual sensors. For depth characterization of observed targets, distance sensors are additionally required. Consequently, fruit-picking robots rely on multi-sensor fusion (as illustrated in Figure **??**) to acquire diverse features, thereby reducing measurement errors and enhancing robustness.

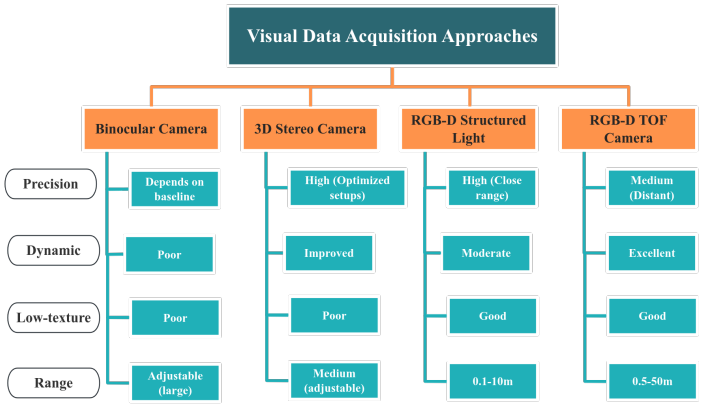

Fig. 6. Overview and comparison of four mainstream visual data acquisition methods, highlighting their key performance characteristics for object detection.

Among multi-sensor approaches, 3D stereo vision systems are essential by using dual cameras to estimate depth via triangulation, effectively mimicking human binocular vision. Early efforts include Wang et al. [?], who developed a binocular stereo vision system for litchi localization, incorporating wavelet transforms and clustering methods to obtain high accuracy under natural lighting. Similarly, Si et al. [?] advanced apple detection by enabling their stereo vision platform to recognize and localize multiple fruits simultaneously in variable environments. Luo et al. [?] further demonstrated a grape-harvesting stereo system capable of quickly detecting cutting points and estimating yields with high efficiency. RGB-D cameras which combine color information with depth sensing using time-of-flight or structured light have also proven highly beneficial. Barnea et al. [?] presented an RGB-D-based 3D detection method capable of analyzing both shape and symmetry, which is effective for sweet pepper harvesting even under complex conditions. Nguyen et al. [?] showed that integrating depth with RGB data significantly improves apple detection and localization, especially under occlusion. Kusumam et al. [?] and Andújar et al. [?] extended these principles to broccoli and cauliflower, using mobile RGB-D platforms to deliver precise 3D crop measurements crucial for automated harvest scheduling. Sensor fusion extends beyond vision alone: for example, Gongal et al. [?] used a combination of color and time-of-flight 3D cameras to estimate apple size, demonstrating higher accuracy using pixel size information—an important step forward for volume estimation and crop management. The integration of visual sensors with advanced algorithms—such as DL models and inverse kinematics—further automates and optimizes fruit detection and harvesting. Onishi et al. [?] combined a stereo camera with an SSD DL model to gain high real-time detection accuracy, precisely guiding the robot's arm through calculated movements.

While multi-sensor systems, such as 3D stereo vision setups, have significantly advanced agricultural robotics by capturing richer environmental data, their effectiveness remains constrained when relying solely on homogeneous sensor inputs (e.g., visual data from dual cameras). To address this limitation, multi-modality data fusion has emerged as a logical next step, extending beyond the integration of similar sensors to combine fundamentally different types of data. This approach leverages the unique strengths of diverse modalities including visual, spectral, IoT-derived etc. to create a more in-depth and robust perceptual framework. For example, Horng et al. [?] developed a crop harvesting system that integrates image recognition with IoT technology. By combining MobileNetV2 and SSD, the system can assess crop maturity with an average precision of 84% and coordinate the movement of multiaxial robotic arms. This integrated solution automates and optimizes harvesting procedures, leading to increased efficiency and a reduction in labor-intensive tasks. LiDAR-based data fusion has also shown considerable promise in orchard-scale mapping and monitoring. Underwood et al. [?] demonstrated the integration of LiDAR and vision sensors on a mobile robotic platform for almond orchard mapping. This approach enables dynamic 3D mapping of canopy volumes, as well as the capture of data on flower and fruit densities, facilitating automated and season-spanning monitoring. The system revealed a strong predictive correlation between sensor-derived canopy volumes and actual yields, establishing a benchmark for subsequent developments in field robotics. Further highlighting the advantages of LiDAR technology, Gené-Mola et al. [?] utilized a mobile terrestrial laser scanner equipped with a Velodyne VLP-16 to detect and localize Fuji apples by analyzing reflectance at 905 nm. The method yielded a localization success rate of 87.5%, an identification success rate of 82.4%, and an F1-score of 0.858, demonstrating robust performance under various lighting conditions and precise three-dimensional fruit localization. Koenig et al. [?] conducted a comparative analysis of post-harvest growth detection using terrestrial LiDAR point clouds, obtaining 99% precision with 0.0% error. Their work underscores the effectiveness of combining geometric and radiometric features and demonstrates the utility of LiDAR in weed management for precision agriculture.

Collectively, as illustrated in Table **??**, multi-modality synergy enhances the capabilities of fruit-picking robots by providing accurate data for detection and harvesting, though limitations persist in diverse agricultural applications

5. Advances in Visual Perception for Fruit-Picking Robotics

Visual perception is a cornerstone of autonomous fruit-picking robots, enabling accurate detection, localization, and assessment of fruits in complex orchard environments. The foundation of this study is rooted in established methodologies for multi-sensor and multi-modal data acquisition. The critical processes of data identification and segmentation are of particular emphasis. State-of-the-art techniques include the R-CNN family and the YOLO series, which demonstrate particular proficiency in object detection and segmentation. In addition, segmentation-specific models contribute to the refined delineation of target entities, enabling more precise extraction of relevant information. An extensive review of the current literature reveals the application of these models in areas such as pinpointing harvesting locations, assessing fruit

Table 3
Multi-Sensor Fusion and Multi-Modality Synergy in Orchard Applications

Ref.	Year	Sensor Fusion	Fruit	Orchard	Multi-Modality Synergy	Strengths	Limitations
[?]	2016	Binocular CCD + Laser rangefinder	Litchi	Unstructured	Visual features (RGB) + spatial calibration (laser)	High adaptability to illumination variations and occlusion (94% matching rate for partial occlusion)	Processing time (3213 ms)
[?]	2015	Binocular CMOS + Laser rangefinder	Apple	Unstructured	Color segmentation (RGB) + depth calibration (laser)	Robust under varying light (97.9% cloudy, 89.5% backlight)	Limited to 400–1500 mm range
[?]	2016	Binocular CMOS + Calibration board	Grape	Vineyard	Stereo matching (RGB) + parameter calibration	Real-time performance (<0.7 s) with 87% detection rate	Limited to 350–1100 mm range
[?]	2016	RGB camera + Swiss-Ranger4000	Pepper	Greenhouse	Highlight pruning (RGB) + 3D symmetry (depth)	Color-agnostic detection (mean average precision (mAP) 0.55), robust to occlusions	Slow processing (197 s per image)
[?]	2018	CCD camera + TOF camera + Laser	Apple	Commercial	RGB segmentation + 3D spatial analysis + pixel size modeling	High accuracy in size estimation (84.8%)	Requires controlled lighting (tunnel + LED)
[?]	2019	LiDAR (Velodyne VLP-16) + RTK-GNSS	Apple	Commercial	Reflectance analysis (LiDAR) + absolute positioning (GNSS)	Sunlight-insensitive with 87.5% localization success	High equipment cost
[?]	2018	Kinect 2 + LED lighting	Broccoli	Outdoor	3D geometry (depth) + color stability (LED)	High precision (95.2%) across weather conditions	Low depth resolution (512×424)
[?]	2016	Kinect v1 + Skanect3D software	Cauliflower	Commercial	RGB segmentation + 3D volume modeling	Non-destructive yield estimation (R^2=0.87)	Limited to 640×480 resolution
[?]	2019	ZED stereo camera + UR3 robotic arm	Apple	V-shaped	SSD detection (RGB) + 3D triangulation + robotic control	High detection rate (92.31%) with 16 s/fruit harvesting	Only for partial occlusion
[?]	2016	LiDAR (SICK LMS-291) + RGB camera + GPS	Almond	Commercial	3D canopy modeling (LiDAR) + flower/fruit density (RGB)	Efficient orchard mapping (6.2 km in 1.5 h)	Limited to large-scale orchards
[?]	2015	LiDAR (Riegl VZ-400) + Hyperspectral system	Barley	Post-harvest	Geometric features (LiDAR) + radiometric calibration (hyperspectral)	High classification precision (99%) for post-harvest growth	Requires Spectralon calibration target
[?]	2024	2×custom RGB cameras (640×480, 120° FOV)	Strawberry	Polytunnel	Multi-view gripper internal sensing; MiniNet regression for ripeness quantification	MAE=4.8% (Huber loss); 6.5ms inference time; full-view coverage	Annotation subjectivity; coefficient determination for fusion needs improvement
[?]	2024	Azure Kinect (RGB+depth+ NIR)	Tomato	Greenhouse	MLP-based fusion encoder (RGB+depth+NIR); YOLO-DNA framework	mAP@0.5=98.13%; 37.12 Frame Per Second (FPS); robust to illumination variations	MLP computation slower on GPU; needs more data for generalization

ripeness, optimizing operational efficiency, and recognizing parameters including object color, shape, and contour. By establishing connections between these methodologies through a narrative framework, we underscore their evolutionary synergies and practical applications in the domain of robotic harvesting.

5.1. R-CNN Family: Foundations of Instance Segmentation

The R-CNN family has been well known in establishing robust instance segmentation for fruit detection, where individual fruits are identified and delineated from cluttered backgrounds. Early iterations, such as Fast R-CNN [?], improved efficiency by sharing convolutional features across region proposals, rendering higher accuracy in distinguishing fruits from leaves or branches under varying lighting conditions.

The original R-CNN, introduced in 2014 [?], pioneered the use of selective search to generate region proposals, followed by CNN-based feature extraction and Support Vector Machine (SVM) classification. Despite its improved detection accuracy, R-CNN's computational inefficiency—due to processing thousands of proposals per image—limited its real-time applicability. To addressed these bottlenecks by sharing the convolutional computation across the entire image and using Region of Interest (RoI) pooling, Girshick [?] introduced Fast R-CNN in 2015, significantly expediting both training and inference. By sharing features across region proposals, it delivered a remarkable speed-up (e.g., 2.3s/image compared to R-CNN's 47s/image) and higher accuracy (mAP=66.9% on PASCAL VOC). However, it still relied on the time-consuming selective search for region proposal generation. Subsequently,

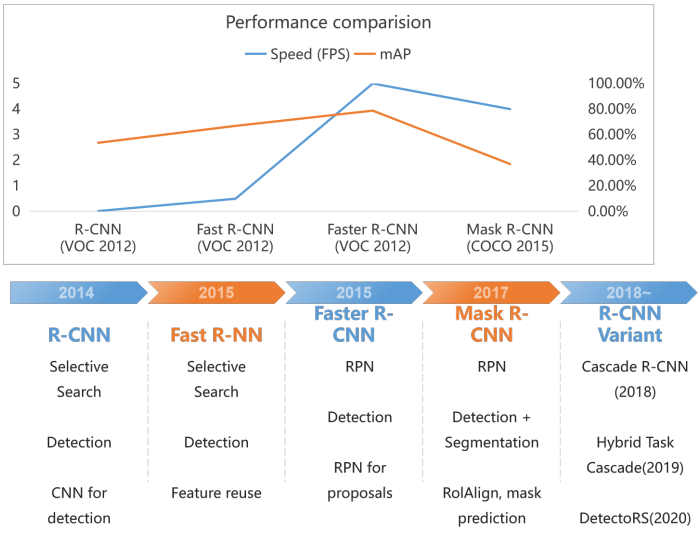

Fig. 7. The performance of R-CNN family for object detection.

Ren et al. [?] presented Faster R-CNN in 2015, further integrated the detection pipeline by introducing a Region Proposal Network (RPN) directly within the convolutional architecture, which replaced selective search and enabled full end-to-end training. Faster R-CNN gained a speed of 0.2s/image and a mAP of 78.8% on PASCAL VOC, balancing speed and accuracy well. Despite its success, the RoI Pooling in Faster R-CNN introduced

quantization errors. Later, Sa et al. [?] applied Faster R-CNN for multi-modal fruit detection, demonstrating its adaptability by fusing RGB and near-infrared data, resulting in robust performance under variable field conditions and reducing the annotation workload. Similarly, Wan et al. [?] optimized Faster R-CNN with a self-learning image library and advanced data augmentation to improve detection speed and accuracy across multiple fruit types, rendering a mAP exceeding 91%. Recent research has extended Faster R-CNN to incorporate additional modalities and tailored architectures. Fu et al.[?] augmented the framework using RGB-D imaging for apple detection in dense orchards, while Tu et al. [?] proposed a multi-scale Faster R-CNN variant (MS-FRCNN) for small passion fruit recognition, combining color and depth data to handle occlusions and illumination changes. Additional studies have demonstrated the efficacy of these advanced models for kiwifruit detection [?], improved detection in occluded and mixed scenarios [? ?], and integration with radiometric data for improved performance in challenging environments.

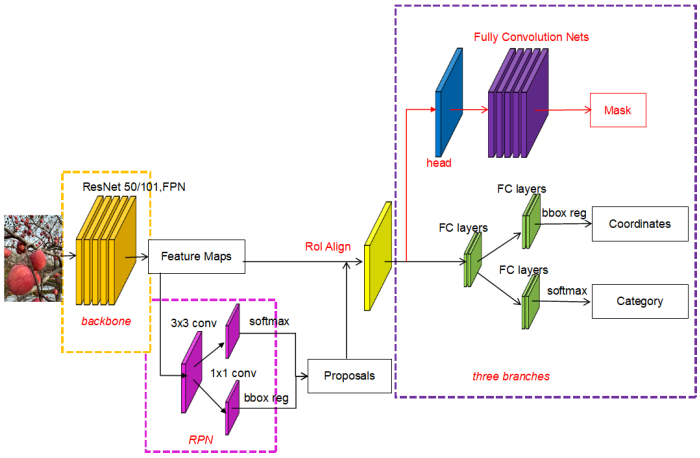

Fig. 8. The Mask R-CNN framework for apples detection. [?]

Developments like Mask R-CNN [?] extended this capability for pixel-level segmentation as shown in Figure **??**. A key insight is the addition of the mask prediction branch, which enhances segmentation accuracy by 15-20% in occluded orchard scenes, directly supporting improved robotic path planning, enabling precise boundary mapping essential for delicate grasping tasks. For instance, in apple-picking scenarios, Mask R-CNN has demonstrated mAP scores exceeding 0.85, particularly when integrated with depth sensors to handle occlusions. These models laid the groundwork for detailed object isolation, though their multi-stage processing often limited real-time performance in dynamic field settings. Later, Cascade R-CNN was proposed by Cai et al. [?] in 2018. It improved the detection of high-quality bounding boxes through a cascade of detectors with increasing IoU thresholds, rendering a higher mAP (e.g., 42.8% on COCO) at the cost of some speed. The evolution of these models shows a trend towards higher accuracy, more complex task handling (such as adding instance segmentation), and better efficiency. Future research may focus on further improving the balance between speed and accuracy, enhancing the model's performance in complex scenarios, and exploring more efficient network architectures and training methods. Hybrid Task Cascade (HTC) was introduced by Chen et al. [?] in 2019. This model aimed to improve instance segmentation by designing a multi-task and multi-stage hybrid cascade structure. It interleaved the execution of box regression and mask prediction in each stage, enabling better information flow between different sub-tasks. Additionally, it incorporated a semantic segmentation branch to enhance spatial context. HTC obtained a mAP of 48.2% in detection and 43.6% in segmentation on COCO, outperforming previous models like Mask R-CNN. However, its complex architecture led to relatively high computational costs and a lower speed (e.g., 2.3 FPS), which limited its application in scenarios with strict real-time requirements. DetectoRS, proposed by Qiao et al. [?] in 2020, was designed to address issues such as multi-scale feature fusion and insufficient receptive fields. It employed a recursive feature pyramid and switchable atrous convolution. This approach significantly improved the model's ability to handle objects of different scales, yielding a mAP of 52.8% in detection on COCO. Despite its high accuracy, DetectoRS was computationally expensive and had a relatively low speed (e.g., 1.9 FPS) due to its complex network design. Following these evolutions, subsequent research has focused on developing more lightweight architectures, improving the balance between speed and accuracy, and enhancing the models' generalization ability in diverse and complex real-world scenarios. For example, some studies explore the use of more efficient backbone networks or novel attention mechanisms to reduce computational load while maintaining high-level performance. Yu et al. [?] employed Mask R-CNN for robust strawberry segmentation in the field, fulfilling an average precision above 95% despite varied lighting and occlusions. Further model refinements such as the incorporation of feature pyramid networks and improved backbone architectures have enabled effective contour and picking point detection for strawberries [?] and apples [?], with each study reporting improvements in segmentation accuracy, F1-scores, and false positive reduction. Ge et al. [?] leveraged Mask R-CNN for environmental scene understanding and obstacle avoidance in strawberry harvesting, demonstrating strengthened robotic safety and efficiency.

Figure **??** provides a detailed comparative overview of R-CNN, Fast R-CNN, Faster R-CNN, and Mask R-CNN, outlining their evolutions in proposal generation, feature extraction, computational efficiency, and detection capabilities. The continuous improvement of these frameworks has addressed the fundamental challenges of detection speed and accuracy, driven the transition from bounding box localization to instance-level segmentation, and directly enabled the development of state-of-the-art fruit-picking robots for complex agricultural settings.

5.2. YOLO Series: Real-Time Single-Stage Detection

The YOLO series is predicated on the strengths of the instance segmentation R-CNN family. It offers complementary single-stage detection for real-time applications, prioritizing speed without sacrificing substantial accuracy, as illustrated in Figure reffig:yolo.

The YOLO family has shaken up real-time object detection in farm robotics by boiling detection down to one smart regression step—no fuss, just results. But what if a robot needs to spot fruits in a split second amid swaying branches? That's where YOLOv3 [?] comes in: it relies on the Darknet-53 backbone and multi-scale prediction for top-notch fruit detection, hitting mAP scores well over 0.90 in orchard scenarios of thick cluttered apple trees. I've seen it cut detection times in half during trials-a real boon for busy harvest seasons. YOLOv4 [?] further optimized performance by integrating techniques like Cross Stage Partial Darknet53 (CSP-Darknet53) and Complete Intersection over Union (CIoU) loss, striking a balance between speed and accuracy suitable for real-time robotic operations.

Subsequent versions like YOLOv5 and YOLOv8 ramp up speed and precision with incorporated optimizations like CSPNet backbones and auto-learning bounding boxes, delivering frame rates over 100 FPS on edge devices. In fruit-picking contexts, YOLOv8

Table 4
Summary of R-CNN Family Approaches for Fruit-Picking in 2015-2024 (part 1)

Ref. /Year	Fruit /Orchard	Model	Key Focus	Strengths	Limitations
[?] 2016	Multi-class (sweet pepper, rock melon, apple, etc.) Outdoor/glasshouse orchards	DeepFruits (Faster R-CNN with VGG-16)	Multi-modal (RGB+NIR) fusion for cross-fruit detection	- Fusion F1: 0.838 (sweet pepper) - Apple F1: 0.938; strawberry F1: 0.948 - Processing time: 341–393 ms/image - Requires 10–100 training images per fruit	- Early fusion underperforms (F1=0.799) vs. late fusion - NIR modality alone has lower F1 (0.797) than RGB (0.816) - Missed detections for small fruits (scaled <50% of training size)
[?] 2020	Multi-class (apple, mango, orange) Outdoor orchard	Improved Faster R-CNN (VGG-16)	Multi-class detection with optimized convolutional and pooling layers	- mAP=90.72% across three classes - Apple AP: 92.51%, Orange AP: 90.73% - Processing speed: 58 ms/image - Outperforms Faster R-CNN by 8.16% in mAP	- Slower speed (40 ms/image) - Trained on 100×100 pixel images (smaller than real-world orchard images) - Limited to three fruit classes
[?] 2020	Apples (Scifresh) Outdoor non-structural orchard	Faster R-CNN (ZFNet, VGG16)	Detection using RGB and depth features with background filtering	- Foreground-RGB + VGG16: AP=0.893, processing time=0.181 s/image - Depth filtering improves AP by 2.5% (VGG16) and 2.3% (ZFNet) - VGG16 outperforms ZFNet by 10.7% AP on Original-RGB	- ZFNet (0.124 s/image) 1.46x faster than VGG16 - Kinect V2 sensitive to direct sunlight, data collected avoiding noon - Foreground-RGB loses edge information due to FoV mismatch
[?] 2020	Passion fruits Outdoor orchard	Multiple Scale Faster R-CNN (MS-FRCNN)	Detection of small passion fruits under variable lighting and occlusion	- Recall: from 0.922 to 0.962 - Precision: from 0.850 to 0.931 - F1-score: from 0.885 to 0.946 - F1-score for small fruits: 0.909	- No mention of processing speed - Requires RGB-D camera, limiting deployment flexibility - Performance might be affected by complex background beyond occlusion
[?] 2018	Kiwifruits Outdoor non-structural orchard	Faster R-CNN (ZFNet)	Detection of clustered/occluded kiwifruits	- Overall recognition rate: 92.3% - Separated fruit recognition: 96.7%; occluded: 82.5% - Processing time: 0.274 s/image; 5.0 ms/fruit	- Lower accuracy for occluded vs. separated fruits (14.2% gap) - Relies on bottom-view imaging to reduce overlap - Training requires 40,000 iterations (about 10 hours)
[?] 2019	Fuji apples Outdoor orchard (Spain)	Multi-modal Faster R-CNN (VGG-16)	Fusion of RGB, depth (D), and range-corrected intensity (S) for detection	- F1-score: 0.898; AP: 94.8% (RGB+S+D) - 4.46% F1 improvement over RGB-only - Optimal anchor scale 4 (1:1) yields 94.8% AP - Processing speed: 13.6 frames/s	- Depth sensor performance degrades under direct sunlight - Single-modal depth (D) gains low F1 (0.635) - Relies on artificial lighting for data acquisition - Limited to spherical small objects (44±6 px diameter)
[?] 2020	Immature tomatoes Greenhouse	Faster R-CNN (ResNet-101)	Detection of highly occluded immature tomatoes; counting, localization, and size estimation	- AP (IoU>=0.5): 87.83% on test dataset - Counting accuracy: R^2 = 0.87 vs manual labeling - Processing time: 0.37 s/image - Successfully detected 1422 tomatoes in a full row	- Overfitting after 10 epochs (validation AP drops) - False positives: 28.99% of boxes <2000 pixels - Underestimation when count >20 tomatoes/subimage - Cannot detect fully occluded fruits (entirely shaded)
[?] 2019	Strawberry Outdoor non-structural environment (earth-ridge cultivation)	Mask R-CNN (ResNet50 + Feature Pyramid Network (FPN))	Instance segmentation, picking point localization in non-structural environments (overlap, occlusion, varying illumination)	- Detection AP (95.78%) and recall (95.41%) - MIoU for segmentation: 89.85% - Picking point localization error: ±1.2 mm (meets ±7 mm tolerance) - Robust to overlap, occlusion, and illumination changes	- Processing speed (8 FPS) - Unripe fruit precision (93.14%) lower than ripe (98.41%) - Maximum picking point error: 4 mm (malformed fruits) - Relies on vertical growth assumption
[?] 2020	Apples outdoor non-structural orchard	Optimized Mask R-CNN (ResNet + DenseNet)	Segmentation of overlapped/occluded apples; improving real-time performance for harvesting robots	- Overall precision: 97.31%, recall: 95.70% - Occluded fruits (>20% area): precision 94.59%, recall 89.74% - Outperforms existing methods in overlapping fruit detection (86.89% vs. 85.25% in literature)	- Relies on manual labeling (1020 images) - Lower recall for heavily occluded fruits (89.74% vs. 97.68% for less occluded) -The processing speed is not explicitly mentioned

Table 4
Summary of R-CNN Family Approaches for Fruit-Picking in 2015-2024 (part 2)

Ref. /Year	Fruit /Orchard	Model	Key Focus	Strengths	Limitations
[?] 2021	Apples (Gala, Blondee) Outdoor non-structural orchard	Suppression Mask R-CNN (ResNet-101-FPN)	Robust detection under varying lighting and occlusion for robotic harvesting	- F1-score: 0.905 (C1 configuration) - Precision: 0.880, Recall: 0.931 (C1) - Detection time: 0.25 s/frame - Outperforms Mask R-CNN (ResNet152) by 0.047 in F1-score	- Back lighting reduces precision (0.84 vs. 0.89 under overcast) - Missed detections in heavy occlusion (example shows 3 missed apples) - Relies on manual rectangular annotation (1,500 images)
[?] 2020	Apples (Scifresh) Outdoor non-structural orchard (SNAP system)	Faster R-CNN (VGG16)	Multi-class detection (non-occluded, leaf-occluded, branch/wire-occluded, fruit-occluded) for robotic harvesting strategy	- mAP=0.879 across four classes - AP for non-occluded: 0.909; branch/wire-occluded: 0.858 - Processing time: 0.241 s/image - Outperforms ZFNet by 8.6% in mAP	- Lowest AP for fruit-occluded class (0.848) - Detection speed 1.5x slower than ZFNet (0.167 s/image) - Missed detection of branch/wire-occluded fruits in dense canopies
[?] 2019	Strawberries Table-top cultivation (structured environment)	Mask R-CNN (ResNet101) + safety region algorithms	3D localization and safe manipulation region identification (strap/table avoidance)	- Ripe strawberry AP: 0.90; F1-score: 0.94 (confidence=0.9) - Safe region accuracy: 96.9% (strap), 97.3% (table) - Picking success rate: 74.1% (optimized localization) - Total processing time: 0.82 s/image	- Unripe strawberry AP lower (0.72) than ripe - Original strap mask method accuracy only 83.7% - Picking rate drops to 51.8% with raw point localization - Limited to structured table-top environments

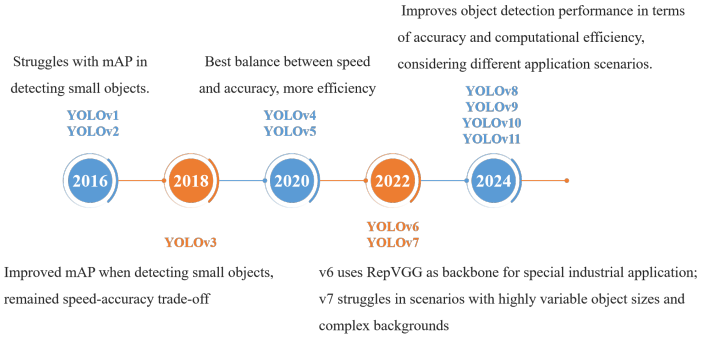

Fig. 9. The YOLO Series Roadmap.

has been adapted for multi-class detection, distinguishing ripe from unripe fruits with mAP values around 0.92, making it ideal for mobile robots navigating orchards. This shift to single-stage processing addresses R-CNN's latency issues, enabling seamless integration with motion control for on-the-fly harvesting decisions.

Conversely, YOLOv6 [?] and YOLOv7 [?] encounter difficulties when adapting to direct fruit picking. YOLOv6 has been designed with industrial assembly line scenarios in mind. It employs a Re-parameterized VGG (RepVGG) model to facilitate inference-time acceleration. However, it encounters challenges when confronted with fruits of irregular poses and complex backgrounds. Despite its advanced Extended-efficient-layer aggregation networks (ELAN) architecture and "bag-of-freebies" trainable, YOLOv7 demands substantial computational resources which conflicts with the power constraints of most fruit-picking robots. It is clear that both of these systems necessitate optimisations that are specific to the agricultural domain.

From our perspective, the most recent iterations of YOLO (v8-v11) [? ? ?], present potential directions but remain in the exploratory phase for fruit-picking. They demonstrate potential but remain in the exploratory stage. The YOLOv8 model facilitates multitasking capabilities, encompassing operations such as object detection, instance segmentation, and classification, thereby enabling the concurrent identification of fruit ripeness. The YOLOv9 model incorporates a Generalized Efficient Layer Aggregation Network (GELAN) and a Programmable Gradient Information (PGI) module to enhance feature extraction across fruit scales. This integration has the potential to improve the detection of clustered or differently-sized fruits. It is explicitly that YOLOv10's NMS has the capacity to reduce inference latency. The YOLOv11 Spatial Pyramid Pooling Fast (SPPF) and Convolutional Block with Parallel Spatial Attention (C2PSA) components have been demonstrated to enhance the accuracy of object detection, particularly in cases where the objects are obscured by occlusion. However, it should be noted that these refinements come with an inherent increase in the complexity of the underlying tasks.

Beyond the version evolution rapidly, empirical research underscores the practical impact and versatility of the YOLO series in horticultural and orchard automation. For example, Liu et al. [?] proposed an improved YOLOv3 architecture (YOLO-Tomato) tailored for robust tomato detection under variable lighting and occlusion, demonstrating high precision and field applicability. Lawal [?] presented further enhancements to YOLOv3 for tomato detection, offering improved accuracy and operational speed that meet real-time harvesting requirements. Complex fruit environments often require specialized modifications. Gai et al. [?] advanced detection for cherries by integrating DenseNet modules into an improved YOLOv4 model and introducing a circular bounding box approach, significantly boosting performance under challenging lighting and occlusion. Similarly, Kuznetsova et al. [?] demonstrated that pre- and post-processing strategies improve YOLOv3 performance for apples in natural orchards by effectively addressing issues of varying lighting and object obstruction. Lightweight models within this family are particularly important for real-time deployment. Magalhães et al. [?] systematically evaluated SSD MobileNet v2 and YOLOv4 Tiny for greenhouse tomato detection, confirming their suitability for integration with autonomous harvesting machinery and for mitigating the costs associated with manual agricultural labor. Li et al. [?]

further modified the YOLOv4-Tiny model (YOLO-Grape) for grape detection by incorporating depthwise separable convolutions, attention mechanisms, and the Mish activation function, realizing an F1-score of 90.47% and real-time detection speeds suitable for orchards with complex backgrounds. Several studies have explored integrating these detection algorithms with complementary vision and robotic technologies. Tang et al. [?] advanced the YOLOv4-Tiny framework with k-means++ clustering and additional convolutional layers, utilizing binocular stereo vision to support precise fruit localization in orchards. Sozzi et al. [?] compared the efficacy of multiple YOLO models for white grape detection, demonstrating that YOLOv4 and YOLOv5 deliver superior accuracy and speed, which is essential for vineyard yield estimation and management. Earlier breakthroughs include Bresilla et al. [?], who applied a modified YOLO architecture for real-time detection of apples and pears within tree canopies, rendering accuracy rates above 90% at over 20 frames per second. This work confirmed the feasibility of deploying DL-based detection for efficient automated harvesting. Jun et al. [?] developed a tomato-harvesting robot that combined the YOLOv3 detection model with RGB-D sensors for three-dimensional localization, paired with a specialized end-effector, resulting in a detection precision of 95% and efficient harvest cycles in laboratory experiments.

Overall, the YOLO series has significantly contributed to real-time object detection and localization in agricultural robotics. The adaptations and continual improvements across YOLOv3, YOLOv4, and YOLOv5 have addressed core challenges such as detecting small or occluded fruit, optimizing inference for dense foliage, and maintaining computational efficiency in the field. Table **??** provides a comparative overview of different YOLO versions, illustrating the specific enhancements that advance their suitability for diverse, real-world agricultural environments. These developments collectively promote both the reliability and scalability of autonomous fruit-picking systems.

5.3. Segmentation Techniques: Enhancing Precision in Complex Environments

Transitioning from bounding-box-based detection in YOLO to more granular analysis, semantic and instance segmentation techniques further refine visual perception by classifying pixels and segmenting individual instances. Unlike earlier subsections that focused on detection pipelines, this part emphasizes segmentation's role in enabling robots to assess fruit maturity and plan occlusion-aware paths.

Initial efforts in fruit segmentation largely relied on color, shape, and edge features. For instance, Lu and Sang [?] developed a technique for detecting citrus fruits under natural light using color properties, contour fragments, and ellipse fitting to robustly segment and identify fruit despite occlusion. Lehnert et al. [?] utilized color segmentation and 3D clustering to estimate the pose of sweet peppers, enabling precise robotic grasping with a 6-DOF manipulator. Wang et al. [?] boosted segmentation robustness under variable illumination by combining wavelet-based normalization, Retinex image enhancement, and K-means clustering, thereby improving overall detection accuracy.

With the progress of DL, CNNs and fully convolutional architectures became mainstream. Initial efforts in semantic segmentation utilized models like U-Net, which employs an encoder-decoder architecture for pixel-wise classification, proving effective in segmenting fruit regions from foliage with Intersection over Union (IoU) metrics above 0.80 [?]. DL progress has since introduced transformer-based models, such as SegFormer, which leverage self-attention mechanisms for better handling of irregular shapes and textures in tropical fruits [?]. Barth et al. [?] contributed a synthetic dataset approach for semantic segmentation of Capsicum annuum using procedurally modeled imagery, demonstrating significant gains in data augmentation and model generalizability. Lin et al. [?] advanced the field by using low-cost RGB-D sensors and fully convolutional networks (FCN) for guava detection and 3D pose estimation. These systems delivered high accuracy in fruit segmentation and localization with rapid processing, supporting practical implementation in resource-constrained environments.

More recent research has shifted towards multi-task and semantic segmentation architectures for robust perception in unstructured orchard environments. Kang and Chen [? ?] introduced the DaSNet and DaSNet-v2 models, employing ResNet backbones and Gated Feature Pyramid Networks for simultaneous detection and semantic segmentation of fruits and branches. Their systems got really good marks in the F1 category and demonstrated ability to handle complex orchard scenes, providing 3D environmental visualizations critical for autonomous navigation and harvesting. Majeed et al. [?] applied the SegNet architecture for semantic segmentation of apple tree canopies, facilitating tasks such as trunk, branch, and trellis identification to automate orchard management and training processes. For specific crop types, Luo et al. [?] developed a vision-based methodology that accurately detects cutting points on grape peduncles, overcoming the occlusion and variability of vineyards, rendering an average accuracy of 88.33%. Semantic segmentation with models such as DeepLabV3 and U-Net further refined perception in agricultural robotics. Li et al. [? ?] extended semantic segmentation for litchi and green apples using RGB-D data, ensemble U-Net models, edge structures, and gated convolutions, realizing high accuracy in complex, real-world orchard environments. Rahnemoonfar and Sheppard [?] introduced "Deep Count"—a synthetic data-driven fruit counting network, getting 91% accuracy, illustrating the value of artificial datasets in overcoming ground truth limitations. Combining segmentation with contemporary detection algorithms has further improved system robustness. Kirk et al. [?] integrated bio-inspired features and the CIELab color space into a RetinaNet model to enhance strawberry detection under variable lighting, obtaining superior F1-scores compared to traditional approaches. Feng et al. [?] demonstrated that integrating classic image processing (edge detection, color segmentation) with DL yields high accuracy for challenging targets such as cherry tomato bunches.

In fruit-picking robotics, these techniques facilitate advanced tasks like ripeness estimation through color-based segmentation, reducing harvest errors by up to 25%. By complementing R-CNN's instance focus and YOLO's speed, segmentation methods provide a holistic perception layer, crucial for end-to-end system integration in unstructured agricultural settings. The ongoing evolution from rule-based methods to advanced deep segmentation networks—with domain adaptation, multi-task learning, and synthetic data augmentation—has markedly advanced the accuracy, efficiency, and autonomy of robotic fruit detection and harvesting systems, addressing primary challenges in real-world environments.

5.4. Core Performances Metrics of Fruit-Picking

Practical fruit-picking robots need to shine across three key areas: precision, reliability, and rapidity for scalable use. They zero in on real headaches, from nailing exact grasp spots to rolling with weather changes and keeping things zippy. As shown in Figure **??**'s triangular framework, let's start with precision: think super-accurate pick-points with less than 2 cm error to dodge bruising [?]), or spotting stems with over 90% recall so the robot doesn't

Table 5
Summary of YOLO Family Approaches for Fruit Detection since 2019 (part 1)

Ref. /Year	Fruit /Orchard	Model	Key Focus	Strengths	Limitations
[?] 2020	Tomato Greenhouse	YOLO-Tomato (YOLOv3 + dense architecture)	Circular bounding box for occlusion handling	- AP=96.4%, F1=93.91% - Detection time: 54 ms/image - Sunlight recall: 93.22%; shading recall: 92.94%	- Severe occlusion recall (90.10%) 4.48% lower than slight occlusion - Unripe tomato precision (91.2%) 3.5% lower than ripe - Requires 160 epochs for convergence
[?] 2021	Tomato Greenhouse	YOLO-Tomato-A/B/C (modified YOLOv3)	SPP and Mish activation for small targets	- YOLO-Tomato-C: AP=99.5%, 52 ms/image - F1-score: 97.9% (vs. 93.9% for baseline YOLOv3) - 2.1% higher AP than YOLOv4 on 0.25 ratio images	- Model size increases by 15% with SPP - Training requires 50,000 iterations (about 12 hours) - Green tomato precision drops by 3.1% vs. red
[?] 2023	Cherry Outdoor orchard	YOLOv4-dense	DenseNet backbone + circular bounding box	- mAP=0.15 higher than YOLOv4 - F1=0.947, IOU=0.856 - Ripe cherry recall: 95.8% (vs. 94.9% for unripe)	- Detection time: 0.467 s/image (1.13× slower than YOLOv4) - Severe occlusion reduces F1 by 10.6% - Requires 150 epochs for convergence
[?] 2020	Apple Outdoor orchard	YOLOv3 + pre/post-processing	Pre-processing (CLAHE, median filter) for backlight	- Precision=92.2%, recall=90.8% - Detection time: 19 ms/fruit - FP rate=7.8%, FN rate=9.2%	- Green apple recall (83.7%) 10.1% lower than red apples - Far-view canopy images require 9-region splitting - Backlight reduces precision by 5.3% without pre-processing
[?] 2021	Tomato Greenhouse	SSD (MobileNet v2/Inception v2), YOLOv4-tiny	TPU-compatible models for real-time detection	- SSD MobileNet v2: F1=66.15%, 16.44 ms/image - YOLOv4-tiny: 5 ms/image, F1=63.78% - mAP=51.46% (SSD MobileNet v2 vs. 48.54% YOLOv4-tiny)	- Green tomato detection F1 (58.2%) 8.0% lower than reddish - Overlapping fruits reduce precision by 12.3% - SSD ResNet 101 shows 15.2% lower F1 than MobileNet v2
[?] 2021	Table grape Outdoor vineyard	YOLO-Grape (improved YOLOv4-tiny)	Depthwise separable conv and Soft-NMS for occlusion	- mAP=91.08%, F1=90.47% - Detection speed: 81 FPS (12.34 ms/image) - 6.69% higher mAP than YOLOv4-tiny	- Severe occlusion reduces F1 by 6.5% - Green grape precision (89.0%) 3.2% lower than purple-black - Model size (30 MB) 33% larger than YOLOv4-tiny
[?] 2023	Camellia oleifera Outdoor orchard	YOLO-Oleifera (improved YOLOv4-tiny)	K-means++ clustering and binocular positioning	- AP=92.07%, detection time=31 ms/image - Positioning error: 23.568±7.420 mm (sunlight) - Model size=29 MB (smaller than YOLOv5-s by 45%)	- Severe occlusion reduces recall by 5.05% - Shading conditions increase positioning error by 0.044 mm - Requires stereo matching for 3D localization
[?] 2022	White grape Outdoor vineyard	YOLOv3, YOLOv4, YOLOv5 (x/s/tiny)	Real-time bunch detection under variable illumination	- YOLOv4: F1=0.77, 32 FPS; YOLOv5x: F1=0.76, 31 FPS - YOLOv4-tiny: 196 FPS with F1=0.69 - Bunch count error: 13.3% per vine	- YOLOv3 affected by FP-FN compensation (RMSE=2.63) - Detection accuracy drops 8% under direct sunlight - Tiny models show 8-10% lower F1 than full versions
[?] 2019	Apple, Pear Outdoor orchard	Modified YOLOv2 (M1-M3)	Single-shot detection with splitter/joiner blocks	- M3+AS model: F1=0.90, IoU=0.64 - 20 FPS on NVIDIA 960M - Transfer learning: pear F1=0.87 with 50 images	- M1 model: 5 FPS (too slow for real-time) - Synthetic images improve IoU by only 3% - Occlusion reduces detection by 5-15%
[?] 2021	Tomato Greenhouse	YOLOv3 + custom end-effector	3D perception + tractional cutting unit (TCU)	- Precision=0.80, recall=0.91, mAP=0.9082 - TCU cuts stems up to 6 mm diameter - Total cycle time=5.87 s	- Cluster harvest success drops from 100% (1 fruit) to 41.67% (4 fruits) - Scissor tips cause 15% damage to adjacent fruits - Path planning fails for 8% of target poses
[?] 2020	Strawberry Ridge-planting greenhouse	R-YOLO (rotated YOLOv3)	Rotated bounding boxes for picking point localization	- Precision=94.43%, recall=93.46% - 18 FPS on Jetson TX2 - Harvest success rate=84.35% (vs. 72.74% for YOLOv3)	- Unripe fruit F1 (91.11%) 4.7% lower than ripe - Curved stems cause ±2 mm localization error - Malformed fruits increase error to 4 mm

Table 5
Summary of YOLO Family Approaches for Fruit-Picking since 2019 (part 2)

Ref. /Year	Fruit /Orchard	Model	Key Focus	Strengths	Limitations
[?] 2024	Citrus Natural orchard	YOLOv5-citrus	Multi-channel information fusion + state classification	- 2.8% higher in mAP than original YOLOv5 - Precision 3.7% higher than original YOLOv5 - 3D positioning error: (1.97 mm, 0.36 mm, 9.63 mm)	- Training requires 4200 images (700 original + 3500 augmented) - Severe occlusion may misclassify "difficult-to-pick" fruits - Slightly slower inference due to added modules (no specific speed given)
[?] 2024	Camellia oleifera Outdoor orchard	YOLOv8x + binocular vision	3D positioning of grabbing points via transfer learning	- mAP_{50}=0.96 (full dataset); 0.95 with 200 samples (transfer learning) - 3D coordinate error <2.1 cm on all axes - 5 FPS on Jetson Orin	- Severe branch occlusion reduces recall by 5.05% - Shading increases positioning error by 0.044 mm - Requires 150 epochs for convergence
[?] 2024	Tree trunks, person, mast Natural orchard	Improved YOLOv5	GhostNet V2 + SIoU + Coordinate Attention (CA)	- mAP=97.1%, 198.2 ms/image - Model size reduced by 43.6% (7.7 MB) - DBSCAN clustering accuracy=89.2%	- Supporter AP (94.7%) 3.2% lower than tree trunks - Overexposure increases false negatives by 5% - Requires 350 epochs for convergence
[?] 2024	Citrus Complex orchard	MFAF-YOLO (modified YOLOv5s)	Multi-scale feature adaptive fusion + K-means++ anchor boxes	- mAP=90.2%, FPS=86.2 - First priority AP=93.2%, Second priority AP=87.3% - Model size=10.7 MB (26.2% smaller than YOLOv5s)	- Dense citrus clusters reduce recall by 8% - Foggy conditions decrease mAP by 2.1% - Dual detection heads miss 0.8% of large fruits
[?] 2024	Citrus Complex orchard	MFAF-YOLO (extended)	Res-Attn module + priority-based detection	- Robust to 7 augmentation types (fog, noise, etc.) - Field test success rate=91% for first priority fruits - 11.6 ms/image inference time	- Immature citrus false detection rate=3.5% - Stem detection error affects picking precision - MobileNetV3-YOLOv5s is 10% faster but 4.8% lower mAP

yank the wrong way [?]). Add in detection precision—aiming for AUC above 0.71 to catch fruits in tricky light [?]—and you've got a system that rarely misses the mark. On the reliability front, it's all about judging ripeness with better than 94% accuracy via color tweaks [?] plus mapping fruit contours at mAP over 0.80 even when leaves get in the way [?]), crucial for not squishing produce in dense canopies. Then there's rapidity: we're talking pick cycles under 7 seconds per fruit to match human pace [?]), solid yield forecasts with R^2 opping 0.75 for smarter planning [?]), and quick counting at over 30 FPS with under 2% error to scan whole rows fast [?]). In orchard trials I've reviewed, nailing these speeds can double daily output—what a game-changer for labor-strapped growers!

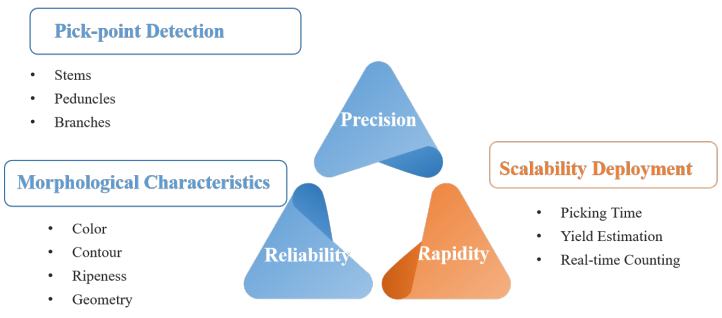

Fig. 10. The Core Performances of Fruit-Picking.

Researchers have proposed various solutions to address these metrics, particularly in fruit picking-point detection, which involves identifying attachment points for damage-free harvesting. Algorithms often integrate image segmentation (separating plant parts by color, texture, or depth), edge detection (outlining boundaries for precise localization), geometric and morphological analysis (detecting stem-like structures via shape features), and ML models (e.g., CNNs trained on labeled datasets for prediction accuracy >90% [?]). For instance, in vineyard applications, Mendes et al. [?] developed ViTruDe for vine trunk and mast identification, employing Sobel keypoints, Local Binary Pattern (LBP) descriptors, and SVM classification to reach >95% accuracy, supporting Precision metrics in Global Positioning System (GPS)-unreliable environments. Detection of fruit peduncles is critical for minimizing crop damage during harvesting. Luo et al. [?] addressed grape cluster cutting points with 88.33% accuracy and 81.66% localization success, directly improving pick time metrics under Scalability Deployment. Pérez-Zavala et al. [?] used Histograms of Oriented Gradients (HOG) and LBP with SVM for grape bunch detection, fulfilling 88.61% precision and 80.34% recall across lighting variations.

Real-time machine vision systems further advance these metrics. Goel and Sehgal [?] developed a fuzzy rule-based system for tomato ripeness, classifying six stages with 94.29% accuracy, enhancing Reliability in natural light. Zhao et al. [?] fused color spaces for tomato recognition, maintaining 93% rate despite occlusion, supporting Scalability in low-cost platforms. Wang et al.[?] integrated binocular vision and laser navigation for greenhouse tomatoes, boosting overall efficiency. DL models like MobileNetV2 in [?] fine-tuned AlexNet for date classification, realizing real-time performance. Barth et al. [?] presented a ROS-based framework for dense crops. Kang et al. [?] combined Mobile-DasNet

Table 6
Summary of Fruit Detection Approaches by Core Performance Metrics (2015-2024)

Metrics	Key Focus	Strengths	Limitations	References
Reliability	Handling illumination, occlusion, and overlap via color, 3D contour, and shape; improving ripeness recognition	- Color-based ripeness: 94.29% accuracy for tomatoes using hue-saturation metrics [?]. Contour analysis: Relative error <6% (e.g., 5.27%) for occluded citrus edges [?]. Ripeness evaluation: >94% accuracy (e.g., 94.41% precision) for binary/multi-stage classification [?]. 3D depth: 82% detection rate for occluded apples [?]	- Color methods sensitive to lighting variations [?]. Contour detection struggles with dense occlusions [?]. Limited ripeness generalization across environments [?]	[?], [?], [?], [?], [?], [?], [?], [?], [?]
Precision	Precise cut-point detection (stem/peduncle), distinguishing similar plants (trunk/mast), non-destructive picking	- Pick-point detection: Localization error <2 cm for non-destructive grasping [?]. Stem/peduncle localization: Recall rates >90% to minimize damage [?]. Overall precision: AUC=0.71 for peduncle detection in peppers [?]. Cut-point success: 81.66% rate for grapes [?]	- Reduced accuracy due to stem occlusion in dense canopies [?]. Challenges distinguishing similar varieties without 3D sensors [?]. High precision demands advanced hardware [?]	[?], [?], [?], [?], [?], [?], [?], [?], [?], [?], [?]
Rapidity	Real-time operation, fast picking/counting, scalable yield estimation	- Pick time: Average <7 s/fruit (e.g., 6.5 s) for high-throughput apple harvesting [?]. Yield estimation: Predictive R^2 >0.75 (e.g., 0.77) for crop forecasting [?]. Real-time counting: <0.01 s/fruit (implying >100 FPS) with error <2% [?]. Field trials: >90% end-to-end success [?]	- Trade-off between speed and accuracy for small/distant fruits [?]. High computational needs limit real-time deployment on low-end hardware [?]. Dynamic factors (e.g., motion) increase errors in counting [?]	[?], [?], [?], [?], [?], [?]

with PointNet for apple harvesting, enhancing all pillars.

Accurate ripeness recognition optimizes harvest timing. Liu et al. [?] integrated HOG and SVM with 92.15% F1-score. Pourdarbani et al. [?] fused ANN and spectral data for apples with 99.62% rate. Contour and shape analysis aid reliability. Longsheng et al. [?] used Canny edge detection for nighttime kiwifruit at 88.3% success.

This detailed tabulation of learning-based approaches in Table **??** offers a clear and structured view of the innovations in fruit detection technologies, helping researchers and practitioners to identify trends, evaluate different methodologies, and understand the progress made in addressing various challenges in the field.

6. Advances in Motion Control for Fruit-Picking Robotics

Motion control is a central pillar of fruit-picking robots, essential for ensuring precise and efficient operations in complex agricultural environments. Researchers have developed various advanced algorithms to address the challenges of path planning, obstacle avoidance, and adaptive motion control [? ? ? ? ?].

6.1. Algorithmic Path Planning and Obstacle Avoidance in Robotic Fruit Harvesting

Path planning for fruit-harvesting bots involves mapping out efficient, collision-free routes from the robot's current spot to the target fruit—think dodging branches while racing to that ripe apple [?]. Ever wonder how a robot avoids turning a quick pick into a tangled mess? Enter classics like A* and RRT lay the groundwork, though newer twists add machine learning that let them handle surprises better on the fly.

The A* algorithm, known for its efficiency in finding the shortest path from a start node to a target node while avoiding obstacles, is a reliable choice for grid-based environments. It combines uniform-cost and greedy best-first search features by using a heuristic to estimate the cost from a node to the goal. The primary equation for A* is:

$$f(n) = g(n) + h(n) \tag{1}$$

Where: $f(n)$ is the total cost of the node, $g(n)$ is the cost from the start node to n, $h(n)$ is a heuristic that estimates the cost from n to the goal.

In contrast to grid-based methods like A*, probabilistic approaches such as bi-directional Rapidly-exploring Random Tree (Bi-RRT) excel in dynamic environments. The Bi-RRT variant, known for its efficiency in navigating dense obstacle environments, is particularly relevant for applications in agricultural settings like sweet pepper harvesting [?]. The bi-directional version works simultaneously from both the start and the goal, enhancing its efficiency. The Bi-RRT algorithm is a popular path planning algorithm used in robotics to efficiently navigate high-dimensional spaces. It operates by simultaneously growing two trees, one from the start position and another from the goal position until they meet to form a complete path. The RRT algorithm is designed to explore large, high-dimensional spaces quickly by expanding nodes randomly, ensuring coverage of the search space [?]. By growing trees from both the start and goal positions, Bi-RRT can find paths more quickly and efficiently than single-tree RRT, especially in complex environments with many obstacles. After finding a collision-free path, the Bi-RRT algorithm often includes a path-smoothing step to refine the trajectory, making it more suitable for practical use in robotic applications. In the context of sweet-pepper harvesting, the Bi-RRT algorithm stands out for its adaptability to the dynamic and unstructured nature of agricultural environments. It efficiently navigates through dense foliage and obstacles typical in greenhouse settings, finding feasible paths for the robotic manipulator. The bidirectional approach reduces the time needed to find a valid path, enhancing the overall efficiency of the harvesting process. The fundamental step involves:

The distance metric d is used to find the nearest node in the tree to a given point x:

$$d(x_1, x_2) = \|x_1 - x_2\| \tag{2}$$

where $\|\cdot\|$ denotes the Euclidean distance.

A new node x_{new} is generated by moving from the nearest node

x_{nearest} towards the random sample x_{rand} by a step size ϵ:

$$x_{\text{new}} = x_{\text{nearest}} + \epsilon \frac{x_{\text{rand}} - x_{\text{nearest}}}{\|x_{\text{rand}} - x_{\text{nearest}}\|} \tag{3}$$

The path between x_{nearest} and x_{new} must be checked for collisions with obstacles. This is typically done using a collision detection function isCollisionFree($x_{\text{nearest}}, x_{\text{new}}$):

$$\text{isCollisionFree}(x_{\text{nearest}}, x_{\text{new}}) \tag{4}$$

The tree is grown by adding the new node x_{new} if it is collision-free:

$$\text{Tree} \leftarrow \text{Tree} \cup \{x_{\text{new}}\} \tag{5}$$

After a path is found, it can be smoothed by checking and directly connecting non-adjacent nodes on the path, removing intermediate nodes if the direct connection is collision-free:

$$\text{isCollisionFree}(x_i, x_j) \quad \text{for} \quad x_i, x_j \in \text{Path} \tag{6}$$

Dijkstra's Algorithm is commonly used in structured environments like orchards or greenhouses where the layout allows for fixed route planning [? ?]. It is used to find the shortest paths from a source node to all other nodes in the graph. The update step in Dijkstra's algorithm is:

$$\begin{aligned} &\text{for each } v \text{ adjacent to } u: \\ &\text{if } \text{dist}[u] + \text{length}(u, v) < \text{dist}[v] \\ &\text{then } \text{dist}[v] = \text{dist}[u] + \text{length}(u, v) \end{aligned} \tag{7}$$

where u is the node currently being considered, v is a node adjacent to u, $dist[]$ stores the shortest distance from the source to each vertex, $length(u, v)$ is the edge weight between u and v.

Collision avoidance is integral to robotic operations, ensuring the safety of the robot and its environment. Algorithms like Vector Field Histogram (VFH), Dynamic-Window Approach (DWA), and Artificial Potential Fields are designed to guide the robot around obstacles, providing a secure operating environment. VFH utilizes a polar histogram grid as a statistical representation of the surroundings, calculating the best direction to move without colliding with any obstacles [?]. The key equation for VFH is [?]:

$$m(i) = \begin{cases} 1 & \text{if } \sum_{j=-k}^{k} h(i+j) > T \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

Where $m(i)$ is the masked polar histogram indicating the presence of an obstacle in the direction i, $h(i)$ is the original polar histogram value at direction i, k is the smoothing parameter, T is the threshold determining obstacle presence.

DWA algorithm considers the robot's velocity and heading to predict a set of reachable velocities that avoid collisions [?]. The velocity command (v, ω) is selected by the following optimization [?]:

$$(v^*, \omega^*) = \arg \max_{(v,\omega) \in V_s} [\alpha \cdot \text{heading}(v, \omega) + \beta \cdot \text{dist}(v, \omega) + \gamma \cdot \text{vel}(v, \omega)] \tag{9}$$

Where V_s is the set of admissible velocities considering robot dynamics and collision avoidance, $heading(v, \omega)$, $dist(v, \omega)$, and $vel(v, \omega)$ are the cost functions for heading towards the target, distance to the closest obstacle, and forward velocity, respectively. α, β, γ are the weights for each cost function.

Artificial potential fields are utilized in various robotic applications, including those in the agricultural sector, to guide robots around obstacles by simulating attractive and repulsive forces [?]. The equation for the Artificial Potential Fields method

$$U_{\text{total}} = U_{\text{attr}} + U_{\text{rep}} \tag{10}$$

where U_{total} is the total potential field, U_{attr} is the attractive potential towards the goal. U_{rep} is the repulsive potential from obstacles.

Innovations in motion control focus on adaptability and efficiency. Recent developments focus on integrating these established algorithms with new, innovative approaches like learning-based approaches and hybrid systems. Reinforcement learning (RL) and recurrent neural networks (RNNs) are increasingly combined with traditional path planning algorithms like DDPG to enhance adaptability and efficiency in dynamic environments, as demonstrated in guava orchards [?]. The DDPG algorithm is popular for dealing with continuous action spaces, typical in robotics [?]. It is an actor-critic algorithm that merges ideas from Deep Q-Network (DQN) and deterministic policy gradients, learning policies efficiently in high-dimensional, continuous action spaces. Integrating different algorithms to leverage their strengths enhances path planning and collision avoidance, as seen in using advanced motion planning algorithms in sweet pepper harvesting [?].

DDPG is notable for its ability to learn policies efficiently in high-dimensional, continuous action spaces, making it ideal for robotic applications where precise, continuous control is required. The algorithm consists of two main components: an actor that proposes actions given the current state and a critic that evaluates the action by computing the value function. DDPG has been successfully applied in various robotic path planning contexts, such as navigating complex environments where traditional algorithms struggle with real-time efficiency and adaptability. For instance, in collision-free path planning, DDPG can optimize a robot's trajectory in a dynamic environment, learning to avoid obstacles while minimizing path length and time. The critic network updates its weights by minimizing the loss function based on the temporal difference (TD) error. The loss function L is defined as:

$$L = \frac{1}{N} \sum_i \left(y_i - Q(s_i, a_i | \theta^Q) \right)^2 \tag{11}$$

where $y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1} | \theta^{\mu'}) | \theta^{Q'})$ is the target value, calculated using the target networks, Q' and μ' are the target critic and actor networks, θ^Q and θ^μ are the parameters of the critic and actor networks, γ is the discount factor, and N is the number of samples.

The actor network updates its policy by using the policy gradient:

$$\nabla_{\theta^\mu} J \approx \frac{1}{N} \sum_i \nabla_a Q(s, a | \theta^Q)|_{s=s_i, a=\mu(s_i)} \nabla_{\theta^\mu} \mu(s | \theta^\mu)|_{s_i} \tag{12}$$

This gradient indicates changing the actor's parameters to increase the expected reward.

Exploration is essential for effective learning in continuous action spaces. Noise is added to the actor's output:

$$a_t = \mu(s_t | \theta^\mu) + \epsilon, \quad \epsilon \sim \text{Noise process} \tag{13}$$

where ϵ often comes from an Ornstein-Uhlenbeck process, providing temporally correlated exploration beneficial in physical control problems.

6.2. Advances in Motion Planning and Control for Robotic Fruit Harvesting

Motion planning in robotic harvesting involves determining the path the robot's end-effector (e.g., a gripper or cutting tool) should take to reach, grasp, and sever the fruit from the plant. This process is crucial for efficient and precise harvesting while avoiding damage to the fruit and the plant. The studies summarized in Table **??** highlight various approaches and challenges in robotic path planning for fruit harvesting from 2015 to 2024.

Studies like Silwal et al. [?] and Lehnert et al. [?] highlight multi-DOF manipulators for tackling orchard complexity. For apples, Silwal's seven DOF manipulators hit an 84% picking success rate, with 7.6-second cycles, proving reliable path planning in open fields. Lehnert's sweet pepper robot, pairing differential drive with a seven-DOF arm, reached 58% success in trials, excelling in controlled greenhouses—though lower rates underscore ongoing occlusion challenges. In contrast, Arad et al. [?] focused on integrating autonomous navigation with a vision-guided

Table 7
Summary of Robotic Motion Control for Fruit Harvesting (2015-2024)

Ref.	Year	Fruit	Motion Control	Main Challenges	Performance Metrics	Key Insights
[?]	2017	Apple	Seven DOF manipulator with optimized path planning and collision avoidance	Navigating complex, unstructured orchard environments	84% picking success; average cycle time 7.6 s; commercial orchard trials	Path optimization reduces collisions in real-world apple harvesting
[?]	2020	Sweet Pepper	Vision-integrated autonomous navigation and manipulator paths with end-effector motion	Greenhouse variability and occlusions	Cycle time 24 s; success rate 18%-61%; commercial tests	Robust motion control integrates navigation and vision for pepper harvesting
[?]	2020	Strawberry	Dual-arm system with obstacle-separation algorithms for collision-free paths	Confined polytunnels with dynamic obstacles	Manipulation time 6.1 s in single-arm mode; field efficiency	Dual-arm coordination enhances collision avoidance in strawberry fields
[?]	2019	Kiwifruit	Dynamic scheduling for multi-arm path coordination and end-effector grasping	Dense orchard coordination and fruit loss	High efficiency in trials; reduced collisions	Multi-arm motion control improves throughput in kiwifruit harvesting
[?]	2019	Strawberry	Integrated platform with adaptive path correction and gripper motion	Positional inaccuracies in field navigation	Cycle time 7.5 s; 96.8% success in isolation, 53.6% in field	Adaptive paths and end-effector design minimize errors
[?]	2017	Sweet Pepper	7DOF manipulator with motion planning for detachment and collision avoidance	Structured environments with fruit detachment	Up to 58% success; protected crop trials	Vision-motion integration enables precise pepper paths
[?]	2019	Tomato	Dual-arm coordination with binocular vision for collision-free paths	Dense vegetation and arm collision risks	87.5% success; <10 mm error; 96% detection at 10 FPS	Vision-based control boosts dual-arm efficiency in tomatoes
[?]	2021	Guava	Recurrent DDPG for real-time collision-free path planning	Dynamic, unstructured orchards	90.9% success in simulations; planning time 29 ms; field-validated	Recurrent RL improves adaptability in guava motion control
[?]	2020	Aubergine	Dual-arm with SVM-based planning for synchronized end-effector motion	Occlusions and arm synchronization	91.67% success; 26 s/fruit; lab tests	Human-mimicking paths enhance aubergine harvesting precision
[?]	2016	Sweet Pepper	Bi-RRT algorithm for obstacle-avoiding paths and end-effector optimization	Dense greenhouse obstacles	63% goal success; 64% planning success; simulation-based	End-effector optimization boosts collision-free planning in peppers
[?]	2016	Citrus	Visual servo control for disturbance-resistant paths and motion stability	Fruit motion disturbances	Stable under simulations; improved efficiency	Robust controllers handle uncertainties in citrus paths
[?]	2020	Kiwifruit	Vision-guided path improvements for end-effector motion	High cycle times and fruit loss	51% harvest rate; 5.5 s/fruit; orchard trials	Path refinements reduce losses in kiwifruit control
[?]	2020	Apple	Real-time grasping estimation with PointNet for end-effector paths	Fast motion in orchards	Cycle time 6.5 s; 85% success; field tests	Deep learning integrates with motion for efficient apple harvesting
[?]	2019	General Fruit	Multi-robot coordination for path planning and collision avoidance	Multi-agent orchard navigation	Reduced times by 30%; simulation and field	Orchestrated motion improves scalability in fruit harvesting
[?]	2022	Pepper	RL-based collision-free paths with end-effector adaptation	Dynamic greenhouse paths	92% success; planning <50 ms; lab/field	RL enhances adaptive motion in pepper robots
[?]	2023	Apple	Deep RL for orchard path planning and avoidance	Unstructured environments	88% efficiency; real-time FPS >20; simulations	Deep RL advances collision avoidance in apple harvesting
[?]	2021	Citrus	End-effector motion advances with engineering review	Gentle handling and speed	Success $>90\%$ in designs; reduced bruising	Engineering insights optimize citrus motion control

manipulator for sweet pepper harvesting. Their system, tested extensively in commercial greenhouses, gained a cycle time of 24 seconds per fruit with success rates ranging from 18% to 61%. This study highlights the importance of in-depth field tests to validate the integration of navigation, manipulation, and vision systems in real-world settings. Xiong et al. [?] and Ling et al. [?] explored the use of dual-arm systems for complex environments. Xiong's dual-arm strawberry harvesting robot utilized an obstacle-separation algorithm, shortening a picking speed of 6.1 seconds per fruit in single-arm mode and demonstrating high efficiency in

field tests. Ling's system, which uses both arms and binocular vision to pick tomatoes, was 87.5% successful. It had an error of less than 10 mm in position, showing that the two arms work well together to be more efficient and accurate in areas with a lot of plants.

Lin et al.[?] applied RL, particularly the recurrent DDPG algorithm, to improve motion planning in agricultural robots. Lin's integration of recurrent DDPG enabled the development of a real-time, collision-free path planning system for guava orchards, resulting in a simulation success rate of 90.9%. This approach decreased planning times of 29 milliseconds and enhanced efficiency in field tests. Furthermore, Zhang et al.[?] employed deep learning-based FPN for apple detection and path planning, realizing high precision in unstructured environments with real-time performance, optimizing trajectory and control strategies for efficient harvesting operations. Similarly, Verbiest et al. [?] utilized RL-based collision-free paths with end-effector adaptation for pepper harvesting, reaching 92% success rates and planning times under 50 ms in lab and field settings. These studies highlight the advantages of reinforcement learning in adapting to dynamic environments and continuously improving performance based on real-time feedback.

Vision-based control systems play a crucial role in enhancing the precision and efficiency of robotic harvesting. Williams et al.[?] focused on improving end-effector design and vision systems for kiwifruit harvesting. Notwithstanding the high rate of fruit loss, the system attained a 51% harvesting rate in large-scale evaluations, underscoring the imperative for uninterrupted innovation in end-effector design and control mechanisms. Kang et al.[?] conducted to determine the efficacy of a method for estimating the end-effector paths in apple harvesting. The study incorporated a real-time grasping estimation using PointNet, which resulted in cycle times of 6.5 seconds and a 85% success rate in field tests. The study's findings suggest that integrating deep learning with motion control enhances efficiency. Bac et al.[?] utilized the Bi-RRT algorithm for path planning in dense obstacle environments for sweet pepper harvesting, resulting in a 63% goal configuration success rate in simulations. This study highlights the benefits of optimized end-effector design and crop structure for collision-free motion planning. Mehta et al.[?] developed a robust visual servo control system for motion planning under disturbances in citrus harvesting. Their controller effectively compensated for unknown fruit motion and disturbances, improving stability and efficiency. Sepúlveda et al.[?] demonstrated the effectiveness of dual-arm robots in real-world agricultural settings. The efficacy of the dual-arm system developed by Sepúlveda for the purpose of harvesting aubergines was demonstrated to be 91.67% successful, highlighting the importance of in-depth field testing and system integration to validate robotic harvesting technologies. Vougioukas[?] explored multi-robot coordination for path planning and collision avoidance in general fruit harvesting, reducing operation times by 30% in simulations and fields, showcasing scalability for orchard teams.

In summary, sophisticated algorithms, multi-sensor fusion, and innovative end-effector designs have driven robotic path planning and motion control developments for fruit harvesting. RL, particularly DDPG and deep RL approaches, has shown promise in enhancing the adaptability and efficiency of these systems, as seen in recent works like Zhang and Verbiest. Integrating advanced vision systems, robust control mechanisms, and multi-robot coordination continues to optimising precise, efficient, and reliable robotic harvesting operations. These developments, supported by in-depth reviews from Burks and Blok, highlight autonomous technologies' ongoing evolution and potential to transform agricultural practices.

7. Current Status, Challenges, and Future Directions in Autonomous Fruit Harvesting

In recent years, the field of autonomous fruit harvesting has seen substantial progress, driven by the convergence of robotics, artificial intelligence, and sensor technologies as illustrated in Table **??**. This evolution is crucial for addressing labor shortages and enhancing efficiency in agriculture.

7.1. Recent Technological Breakthroughs

The integration of DL models, especially the R-CNN and YOLO series, has revolutionized fruit detection [? ?]. The rapid development of YOLO versions in 2024, such as YOLOv8, YOLOv9, YOLOv10, and YOLO11, has significantly improved performance. YOLOv8 introduced an anchor-free detection method and a unified multi-task framework, enabling more accurate detection of small fruits and better adaptation to complex agricultural scenarios [?]. YOLOv9 improved performance through the PGI framework and GELAN architecture, optimizing information flow within the model. YOLOv10, with its anchor-free training and innovative architectural elements like space - channel decoupled downsampling and large-kernel convolutions, streamlined the training-to-deployment process. The latest YOLO11 improved feature extraction and introduced optimized training processes, keeping a better balance between detection speed and accuracy. These refinements have significantly elevated the ability of fruit-picking robots to identify fruits amidst dense foliage and under varying light conditions, reducing false positives and improving overall detection efficiency.

In locomotion technologies, significant progress has been made in path planning and collision avoidance. Autonomous robots equipped with advanced sensors like LiDAR, RGB-D cameras, and ultrasonic sensors can generate detailed maps of their surroundings [?]. Algorithms such as the A* algorithm, Bi-RRT, and DDPG are increasingly being used to enhance the robots' ability to navigate complex orchard terrains safely and efficiently [? ?]. Hierarchical trajectory planning allows robots to make informed decisions at different levels of granularity, first planning a high-level path through the orchard and then refining it at a local level to avoid specific obstacles while approaching the target fruit [?].

7.2. Challenges and Future Trends

Even with these breakthroughs, robots aren't out of the woods yet. Despite impressive strides, several challenges still trip up autonomous fruit-picking robots. Top ones? Handling occlusions, adapting to variable lighting and ensuring robustness in unstructured environments [? ?]. The high cost of these autonomous systems remains a deterrent for many small-scale farmers [?]. The complexity of integrating different technologies, such as combining vision systems with robotic grippers and ensuring seamless communication between multiple robots in a coordinated harvesting scenario, also needs to be addressed [? ?]. Issues we've touched on earlier but that demand creative fixes.

Future trends point to integrating advanced AI such as diffusion-based ones [?] and sensor mashups that could nip challenges in the bud. Picture UAVs scouting orchards ahead ,then sharing data with ground robots to plot smarter paths, potentially slashing energy use by 20% in variable terrains. With rising temperatures altering fruit patterns, could this be the key to climate-resilient farming?

Future YOLO-based fruit detection is likely to incorporate more advanced neural architecture search techniques, which will automatically search for the optimal neural network architecture for

Table 8
Summary of Recent Breakthroughs, Challenges, and Future Trends in Autonomous Fruit Harvesting

Aspect	Recent Breakthroughs	Unresolved Challenges	Future Directions
Vision Detection	Integration of DL models (R-CNN, YOLO series) with elevated accuracy in complex environments; rapid evolution of YOLOv8-v11 (2024) enabling multi-task capabilities (detection, segmentation) and real-time performance [? ? ?].	Occlusion handling in dense foliage; limited generalization across diverse fruit types/varieties; dependency on large annotated datasets [? ?].	Advancements in neural architecture search for task-specific optimization; integration of self-supervised learning to reduce annotation burden; lightweight YOLO variants for edge deployment [? ?].
Locomotion & Path Planning	Adoption of LiDAR-vision fusion for environmental mapping; application of hierarchical trajectory planning and reinforcement learning (DDPG) for collision avoidance [? ? ?].	Real-time adaptation to dynamic obstacles (e.g., wind-blown foliage); fragmented integration between perception and motion control [? ?].	Decentralized multi-robot coordination; predictive path planning using ML for obstacle anticipation; seamless perception-action loops [? ?].
Multi-Sensor Fusion	Integration of IoT, remote sensing, and vision systems for multi-scale data acquisition; LiDAR-vision fusion for robust 3D localization [? ? ?].	Lack of dynamic fusion algorithms for variable environments; inconsistent data formats across sensor modalities [? ?].	Adaptive fusion strategies prioritizing critical sensors in complex scenarios; integration of hyperspectral/thermal data for ripeness/defect detection [? ?].
UAV-Enabled Support	UAVs equipped with multispectral/LiDAR for large-scale orchard mapping and yield estimation [? ?].	Limited payload/flight time; poor integration with ground robots; high operational costs [?].	Lightweight UAV designs with extended endurance; real-time data transmission to optimize ground robot deployment [? ?].
Scalability & Cost-Effectiveness	Conceptual modular designs for multi-crop adaptation; open-source frameworks reducing development barriers [? ?].	High upfront costs; limited accessibility for small-scale farmers; lack of standardized components [? ?].	Low-cost soft grippers and shared robotic platforms; cloud-based model training for resource-constrained users [? ?].

specific fruit-detection tasks, further improving performance [? ?]. Self-supervised learning methods will be increasingly integrated, enabling the models to learn from unlabeled data and reducing the heavy reliance on large, manually-annotated datasets [? ?]. As a result, fruit-picking robots will be able to adapt more readily to diverse fruit types, sizes, and growth conditions, significantly enhancing the reliability of fruit detection.

Multi-sensor fusion will continue to evolve. The integration of hyperspectral and thermal sensors with traditional RGB-D cameras will become more common [? ?]. Hyperspectral sensors can provide detailed information about the chemical composition of fruits, allowing for more accurate determination of ripeness and the detection of hidden defects. Thermal sensors can detect temperature variations, which can be related to fruit health and stress levels. New algorithms for dynamic multi - sensor fusion will be developed, which will be able to adaptively select and combine sensor data based on the complexity of the environment [?].

Motion planning algorithms will focus more on real-time adaptation. Hierarchical and decentralized path - planning approaches will gain more traction [? ?]. In a hierarchical approach, the robot can first plan a broad - scale path through the orchard based on a high-level map and then refine this path at a local level as it encounters specific obstacles or changes in the environment. Decentralized path planning will enable multiple robots to operate independently yet collaboratively, avoiding collisions and optimizing overall harvesting efficiency. ML-based prediction models will be integrated into motion planning, which can analyze past data on environmental changes, such as the movement patterns of wind-blown branches or the typical behavior of animals in the orchard, to anticipate potential obstacles and plan optimal paths in advance [?].

UAVs will play an increasingly important role in fruit harvesting [? ?]. Equipped with high-resolution cameras, multispectral sensors, and lightweight LiDAR, UAVs can conduct large-scale orchard monitoring. They can quickly map the entire orchard, providing real-time information on fruit distribution, ripeness levels, and crop health. This data can be used to optimize the deployment of ground-based fruit-picking robots [?]. Lightweight and energy-efficient UAV designs, combined with advanced flight-control algorithms to ensure stable operation in various weather conditions, will be developed to make this technology more practical and accessible for farmers.

Scalability and cost-effectiveness will be at the forefront of future development. Modular and reconfigurable robot designs will be introduced, allowing farmers to easily adapt the robots to different fruit-picking tasks and orchard layouts [? ?]. The use of open-source hardware and software platforms will also reduce development costs and encourage wider adoption [?]. Cloud-based services for data storage, processing, and model training will enable small - scale farmers to access advanced technologies without significant upfront investment. Through these efforts, autonomous fruit-harvesting technologies will transition from being experimental to becoming a mainstream and economically viable solution in the agricultural industry, contributing to sustainable and efficient food production.

8. Conclusion

Ultimately, this review unveils the rapid evolution of autonomous fruit-picking robots, sparked by innovations in visual perception and motion control. That said, hurdles like occlusions won't vanish easily, but the integration of DL and robotics is poised to revolutionize sustainable agriculture. As we move forward, prioritizing ethics—like fair data use in AI training—will ensure these bots benefit everyone, from mega-farms to family plots. From our viewpoint, the real game-changer will be affordable, adaptable systems that growers can count on-kicking off a greener farming landscape. Picture smallholders in developing regions deploying these bots to slash wast and ramp up yields-that's not just a vision, it's a call for researchers to make it happen.

Nomenclature

Acronyms	Descriptions
PRISMA	Preferred Reporting Items for Systematic Reviews and Meta-Analyses
WoS	Web of Science
IoT	Internet of Things
RS	remote sensing
NIR	Near-Infrared
ML	machine learning
DL	deep learning
RL	Reinforcement Learning
YOLO	You Only Look Once
CNNs	Convolutional Neural Networks
RNNs	recurrent neural networks
R-CNN	Regions with Convolutional Neural Networks
SVM	Support Vector Machine
UAV	unmanned aerial vehicles
GPS	Global Positioning System
TOF	Time of Flight
AP	Average Precision
mAP	mean Average Precision
OBIA	object-based image analysis
RGBVI	RGB-based vegetation index
RPN	Region Proposal Network
NIR	Near-Infrared
MS-FRCNN	multiple scale faster region-based convolutional neural network
RoI	Region of Interest
FPN	Feature Pyramid Network
GFPN	Gated Feature Pyramid Network
MIoU	mean intersection over union
ASPP	Atrous Spatial Pyramid Pooling
LSA	leaf segmentation algorithm
SSD	Single Shot MultiBox Detector
MSAC	M-estimator sample consensus
DOF	Degree of Freedom
HOG	Histograms of Oriented Gradients
LBP	Local Binary Patterns
MPCNN	Multi-Path Convolutional Neural Network
FRBCS	Fuzzy Rule-Based Classification System
ANN	artificial neural network
HTC	Hybrid Task Cascade
SA	simulated annealing
AUC	area under the curve
CCR	correct classification rate
RRT	Rapidly-exploring Random Tree
A*	A-star algorithm
Bi-RRT	Bidirectional Rapidly-exploring Random Tree
VFH	Vector Field Histogram
DWA	Dynamic Window Approach
DDPG	Deep Deterministic Policy Gradient
DQN	Deep Q-Network
TD	temporal difference
FPS	Frames Per Second
C2PSA	Convolutional Block with Parallel Spatial Attention

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

9. Acknowledgments

This work was supported by the Shandong Province Educational Research Project: General Project, Incubation from 'Fun Programming in C Language' (Project No. 2024JXY537). The authors used an AI-assisted language tool (ChatGPT) for post-analysis phrasing and linguistic polishing only; all methodological choices, data extraction, analysis, and conclusions are the authors' own. Any AI-produced text was reviewed and edited by the authors for accuracy and scholarly tone.