IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2024.Doi Number

Fruit detection methods based on Deep Learning in Agricultural Planting: A Systematic Literature Review

XINYU GONG, QIUFENG WU

School of Arts and Sciences, Northeast Agricultural University, Harbin 150006, China

Corresponding author: QIUFENG WU (qfwu@neau.edu.cn).

This work was supported in part by the Program for Young Talents of Basic Research in Universities of Heilongjiang Province under Grant YQJH2023200.

ABSTRACT With the continued advancement of agricultural intelligence, automation, and mechanization, artificial intelligence-driven fruit detection technology has developed rapidly. As an important task in agricultural computer vision, fruit target detection in real-world planting environments presents numerous technical challenges. This paper provides a systematic review of recent breakthroughs and representative studies in this field. Based on a comprehensive analysis of existing research, we classify deep learning–based fruit detection models into four application scenarios: few-shot detection (addressing limited data availability and high annotation costs), complex scene detection (resolving issues arising from object occlusion, overlapping, and variable illumination), small-target detection (improving performance on low-resolution and densely clustered objects), and real-time detection (designing lightweight algorithms for faster inference). The paper summarizes innovative technical approaches and evaluates detection performance across these scenarios, highlighting recent advancements in fruit detection for agricultural production and offering valuable insights for further technological innovation.

INDEX TERMS Artificial intelligence, object detection, smart agriculture, computer vision, deep learning, systematic literature review.

I. INTRODUCTION

The global agricultural sector is currently experiencing a fourth productivity revolution fueled by intelligent sensing technologies, with fruit detection in planting environments serving as a core component of harvesting robots' environmental perception systems. This technology fulfills three essential functions: yield prediction, quality monitoring, and guidance for robotic operations—functions that directly support the achievement of the United Nations Sustainable Development Goals by reducing post-harvest losses (with the FAO reporting an average loss rate of 30% in developing countries)[1],[2]. International policymakers have intensified their commitments through three landmark initiatives: China's 14th Five-Year Plan (mandating over 50% automation in crop harvesting), Horizon Europe's €2.4 billion investment in agricultural AI, and the U.S. AI Policy Roadmap (prioritizing edge computing deployment)—collectively demonstrating strong global momentum toward technological advancement.

However, a persistent implementation gap remains between policy objectives and practical adoption due to four entrenched technical challenges under open-field conditions: few-shot scenarios (FS), which require approximately 4,000 human-hours per crop species for data annotation; complex scenarios (CS), where occlusion leads to Non-Maximum Suppression failure rates exceeding 35%; small-object scenarios (SO), achieving less than 60% average precision for 32×32-pixel objects; and real-time scenarios (RS), experiencing a 12% model accuracy degradation under computational constraints.

This article first deconstructs the fruit detection problem into four categories: FS, CS, SO, and RS. For FS, meta-learning frameworks and data augmentation techniques are employed to mitigate data scarcity in rare fruit cultivars. In CS, focal loss reweighting and adaptive Non-Maximum Suppression (NMS) architectures are implemented to address feature ambiguity caused by occlusion. In SO, hierarchical feature pyramids and coordinate attention mechanisms are utilized to recover semantic information

from low-resolution targets. For RS, neural architecture search (NAS) combined with model distillation-pruning pipelines is applied to achieve Pareto-optimal speed-accuracy tradeoffs on edge devices.

Our methodology integrates a tri-phase analysis of 148 studies sourced from the Web of Science and CNKI databases (2018–2025), narrowing down to 86 scenario-specific technical improvements through keyword filtering, empirical validation, and critical synthesis.

This paper presents a systematic review of four key challenging scenarios in agricultural fruit detection and evaluates deep learning-based optimization strategies tailored to each scenario, offering actionable insights for advancing research in this domain. Relevant literature was retrieved from the Web of Science and CNKI databases, followed by rigorous screening, critical analysis, and synthesis of methodologies addressing these challenges. The article focuses specifically on deep learning-based optimization approaches for fruit detection. It begins with a comprehensive overview of current development trends, underlying principles, and persistent challenges, followed by an exploration of targeted improvement strategies for each of the four key scenarios. Finally, it outlines potential future research directions in fruit detection within agricultural planting environments.

II. Overview of the development of fruit detection methods

This article retrieves 148 relevant literature through keywords such as "deep learning," "agriculture," "object detection," "fruit," "computer vision," and "recognition." However, some of the literature focused solely on fruit counting, omitted deep learning methods, or lacked optimization for the specific scenarios addressed in this review rendering them unsuitable as a reference for a comprehensive review of fruit detection. In the end, 86 publications were ultimately selected, and the specific countries and the top five universities in terms of publication volume are shown in Tables 1 and 2, respectively.

TABLE 1
QUANTITY OF ARTICLES PER COUNTRY

Country	Num
China	125
USA	7
Japan	3
England	2
Italy	2
Australia	1
Belgium	1
Brazil	1
France	1
Germany	1
India	1
Israel	1
Malaysia	1
Morocco	1

TABLE 2
QUANTITY OF ARTICLES FROM TOP 5 UNIVERSITIES

Country	Institution	Num
China	South China Agricultural University	18
China	Northwest A&F University	12
China	Shandong Normal University	8
China	Jiangsu University	6
China	China Agricultural University	4
China	Shanxi Agricultural University	4
China	Nanjing Agricultural University	4
China	Dalian University	4

The development of fruit detection technology has gone through an evolution from traditional computer vision to deep learning paradigms. Early research mainly used image processing and feature extraction techniques, such as edge detection, color thresholding, and shape recognition, to identify and locate fruits in images. However, these methods performed poorly in complex background, lighting changes, or occlusion environments, making it difficult to meet the actual application needs of agriculture[3]-[6]. With the rise of convolutional neural networks (CNNs), deep learning-based detection frameworks can automatically extract features and recognize complex patterns, significantly reducing the reliance on manual feature engineering. With the development of deep learning technology, frameworks like YOLO have been widely used due to their ability to achieve real-time and accurate fruit detection, and their excellent performance in complex environments.

During this process, the research focus gradually shifted to addressing challenges unique to agricultural scenarios, such as the detection of small target fruits, dense fruit clusters, and detection under varying lighting conditions. To tackle these issues, researchers introduced techniques like data augmentation, multi-scale feature pyramids, and attention mechanisms. These technological breakthroughs have benefited from the collaborative efforts of global research forces, with statistics showing that countries like China, the UK, and the US have contributed over 90% of high-quality fruit detection papers. Research institutions such as South China Agricultural University and Northwest A&F University have played a significant role in advancing the development of fruit detection systems. As research continues to deepen in various regions, the number of research papers and technological advancements in fruit detection has gradually increased worldwide.

Despite significant progress in data-driven deep learning methods, four core challenges remain in practical deployment: Firstly, high-performance models rely on a large number of labeled samples, which are difficult to obtain in real-world scenarios. To address this issue, researchers have proposed optimization methods aimed at dealing with FS through data set augmentation or enhancing the model's self-learning ability. Secondly, issues such as lighting changes, target overlap and occlusion, and background interference in CS seriously affect the accuracy and robustness of target detection, making it a hot topic in current research. Thirdly, for SO, visual features are difficult to obtain efficiently and are noisy, so how to improve

the accuracy of small target detection has become another important issue in the field of computer vision. Finally, the complexity of model parameters and the huge amount of computation in the inference process slow down the inference speed, making the model too large and posing challenges for real-time deployment. To address this issue, researchers are seeking a balance between model detection performance and lightweighting to optimize the application effect in RS.

III. Overview of fruit target detection

A. Introduction to fruit object detection methods

As the core technology of computer vision, object detection aims to realize the dual perception of target localization and classification in images. In the fruit detection scenario, its core task is to extract the feature representation of the target from the complex farmland image, and then realize the application of automatic harvesting, yield evaluation and quality grading. With the development of deep learning, the mainstream detection methods have gradually shifted from traditional manual feature extraction to automatic feature learning frameworks represented by CNN and Transformer. Current object detection algorithms based on deep learning can be divided into the following three categories.

1) ANCHOR-BASED ALGORITHMS

Two-stage detectors, represented by Faster R-CNN, first generate candidate regions through the RPN network, then align features and perform classification and regression via RoI Pooling[7]. Their accuracy advantage stems from the meticulous screening of candidate regions, but computational redundancy is significant (e.g., RoI cross-layer feature fusion requires high-resolution feature maps). Single-stage detectors, represented by the YOLO (You Only Look Once) series and SSD, achieve end-to-end detection through grid-based predictions, significantly improving detection speed at the cost of some accuracy, making them more suitable for agricultural robot scenarios with high real-time requirements[8].

2) ANCHOR-FREE ALGORITHMS

To avoid the sensitivity of anchor box parameters, algorithms such as FCOS predict the target center points and bounding boxes on a per-pixel basis, reducing the cost of hyperparameter tuning[9]. Such methods have potential advantages for SO detection but are sensitive to occlusion scenarios with dense fruit overlaps (e.g., ambiguity in center point prediction in crowded apple scenes).

3) TRANSFORMER-BASED END-TO-END DETECTION ALGORITHM

Models represented by DETR model global contextual relationships through self-attention mechanisms, eliminating the need for NMS post-processing and demonstrating strong robustness in occlusion scenarios. However, due to the $O(N^2)$computational complexity of Transformers, they are difficult to deploy on agricultural edge devices (such as orchard mobile robots)[10].

Table 3 summarizes the key scenario requirements for agricultural fruit detection and the corresponding technological innovation directions.

TABLE 3
SUMMARY OF SCENARIOS

Scene Type	Typical Characteristics	Technical Challenges	Optimization Direction Examples
CS	lighting changes, foliage occlusion, fruit overlap	feature confusion, edge blurring	adaptive data augmentation, context-aware loss function
SO	Dense Fruits, Large Scale Differences	Low-Resolution Feature Loss	Multi-Scale Feature Fusion, Attention-Guided RPN
RS	Dynamic response requirements for picking robots	model computation latency	lightweight network design, knowledge distillation compression
FS	Rare variety annotation data scarcity	overfitting risk	meta-learning framework, synthetic data generation

B. Evaluation index

The performance of object detection algorithms is often quantified and evaluated through multi-dimensional metrics. Below are the classic evaluation methods.

1) CONFUSION MATRIX AND BASIC INDICATORS

TP：True Positive；FP：False Positive；FN：False Negative；TN：True Negative.

This leads to equations 1-4.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$F1 = \frac{2TP}{2TP + FP + FN} \tag{4}$$

2) BOUNDARY BOX LOCALIZATION EVALUATION

IoU(Intersection over Union): a measure of the overlap between the predicted frame and the real frame.

$$IoU = \frac{|A \cap B|}{|A \cup B|} \tag{5}$$

Average Precision (AP): Based on the area under the Precision-Recall Curve, it comprehensively reflects the performance balance at different recall rates.

$$AP = \int_0^1 Precision(r)dr \quad (6)$$

Mean Average Precision (mAP): The arithmetic mean of AP across multiple classes.

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \quad (7)$$

3) ROBUSTNESS EVALUATION METRICS

Matthews Correlation Coefficient (MCC): Suitable for imbalanced data distributions.

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \quad (8)$$

Table 4 consolidates key evaluation metrics with their directional interpretation, and typical value ranges. The integration of Table 4 with the previous formulas (e.g., IoU as a ratio of overlap/union areas, AP via PR-curve integration) aims to visually demystify calculation logic.

TABLE 4
SUMMARY OF SCENARIOS

Metric	Interpretation	Benchmark Direction	Typical Range
Precision	How many selected boxes are correct (avoid false alarms).	Higher ↑	[0,1]
Recall	How many true objects are found (avoid misses).	Higher ↑	[0,1]
F1 Score	Balances Precision & Recall (use when FP/FN costs are similar).	Higher ↑	[0,1]
IoU	Measures spatial accuracy of box prediction (1=perfect fit).	Higher ↑	[0,1]
AP	Average precision across all recall levels. Requires PR curve visualization.	Higher ↑	[0,1]
mAP	Extends AP to multi-class (mean of all class-wise APs).	Higher ↑	[0,1]
MCC	Robust to imbalance (-1= completely wrong, 0= random guess, +1= completely correct).	Closer to +1 ↑	[-1, +1]

C. Public fruit detection datasets

In agricultural computer vision, three datasets are pivotal: The Monastery Apple Dataset (MAD) (2023) employs 4K drone imagery with semi-supervised labels (10k labeled + 4.4k unlabeled images) to advance small-object detection in orchards[11]. MinneApple (2020), featuring 41k+ instance annotations from 1,000 images, supports robust apple segmentation and yield estimation through dense cluster analysis[12]. Wine Grape Instance Segmentation Dataset (WGISD) (2019) addresses genetic diversity with 4,432 grape clusters and 187k berry-level annotations, enabling fine-grained phenotyping in vineyards[13]. While WGISD laid groundwork for multi-variety classification, MinneApple optimized industrialized detection tasks, and MAD introduced scalable semi-supervised frameworks. These datasets collectively reflect the field's progression from basic recognition to efficiency-oriented, real-world applications. As shown in Figure 5.

TABLE 5
COMMONLY USED PUBLIC FRUIT DETECTION DATASETS

Name	Release time	Quantity	Category	Characteristics	Source
The Monastery Apple Dataset (MAD)	2023	Training (66 frames, 10,089 instances), validation (12 frames, 1,288), test (27 frames, 3,290), and 4,440 unlabeled images.	Single apple class with per-instance attributes (size, occlusion, lighting conditions).	① 4K drone imagery of apples with varied lighting conditions. ② Bounding boxes and attribute labels (size, occlusion, lighting). ③ Semi-supervised setup: 10k labeled + 4.4k unlabeled instances. ④ Drone-based scalability and semi-supervision.	[11]
MinneApple	2020	1,000 images (41,000+ instances).	Focused on apples, with instance-level attributes (occlusion, size, cluster density).	① 1,000 high-res orchard images (10MP) for detailed analysis. ② 41k+ polygonal masks for segmentation and patch-level counting. ③ Focused on apple detection, dense clustering, and yield estimation. ④ Pixel-wise annotation depth for dense fruit analysis.	[12]
Wine Grape Instance Segmentatio	2019	300 images, 4,432 clusters (2,020 masks) + 187,374 berry	5 grape varieties (e.g., Syrah, Chardonnay)	① 300 vineyard images with 4,432 grape clusters (2k+ masks). ② Covers 5 genetic varieties and occlusion scenarios.	[13]

n Dataset (WGISD)	annotations.	with pose, occlusion, and genetic variation attributes.	③ Berry-level annotations (187k points) for fine-grained counting. ④ Genetic/phenotypic diversity and micro-level berry counting.

IV. ANALYSIS OF IMPROVED METHODS FOR FRUIT DETECTION TECHNOLOGY

This section presents an in-depth analysis of deep learning–based fruit detection from the perspective of technological evolution. It emphasizes recent challenges and research hotspots, specifically focusing on four key scenarios: few-shot detection (FS), complex scene detection (CS), small-object detection (SO), and real-time detection (RS). The distinctive methodological perspectives within each scenario are elaborated in detail. Among the 86 references cited in this article, 6 address FS, 42 focus on CS, 40 pertain to SO, and 26 relate to RS, with some studies addressing multiple scenarios.

The paper is structured into four main sections, each focusing on a different aspect of fruit detection technology. The first section, "Development Overview," provides a historical perspective on the evolution of research focus and the development of object detection in deep learning. The second section, "Core concepts," delves into the fundamental principles of fruit target detection and the evaluation metrics used for target detection algorithms. The third section, "Research trends: methods for different scenarios," explores various methods tailored to specific detection challenges, such as FS, CS, SO, and RS. Each scenario is addressed with unique techniques, such as data augmentation, meta-learning, non-maximum suppression, loss function optimization, feature pyramid, attention mechanism, pruning, and knowledge distillation. Finally, the "Prospects for future research" section outlines potential directions for advancing the field, ensuring a forward-looking perspective on fruit detection technology. As shown in Fig. 1.

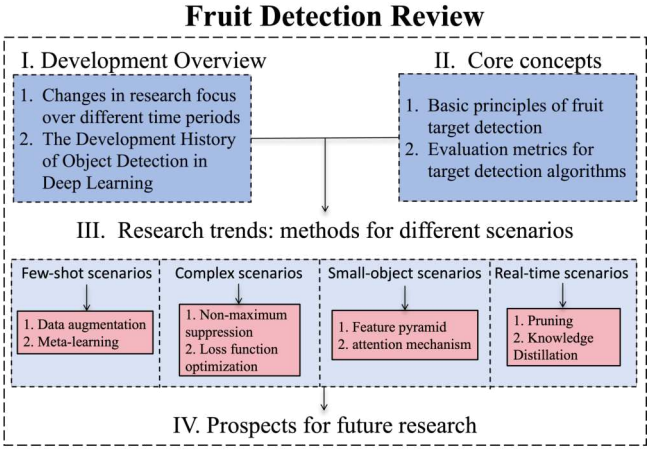

FIGURE 1. Paper Structure.

A. DETECTION OF FS

In the field of deep learning, small-sample scenarios refer to the situation where the amount of training data is relatively small, and are common in areas where data acquisition and labeling costs are high, such as medical image analysis, remote sensing, and agriculture. The research problems of FS are mainly focused on insufficient data, overfitting, insufficient model generalization ability, limited data enhancement effect, improper model architecture design, poor application of transfer learning, imperfect self-supervised learning methods, and challenges of evaluation methods[14].

In the context of FS, the improved methods for fruit detection mainly revolve around data augmentation and generalized meta-learning. Table 6 provides a brief summary and analysis of these two approaches, outlining the backbone networks in the literature and the improvement techniques aimed at addressing the challenges of FS. Sections 4.1.1 and 4.1.2 focus on analyzing the network structures and improvements of the corresponding typical models.

TABLE 6
COMPARATIVE ANALYSIS OF FRUIT PLANTING DETECTION METHODS IN FS

Classification	Backbone	research object	Improvement Points	Enhancement effect	Reference number
Data augmentation	ResNet	Citrus	ResNet101 Data Augmentation	① AP: 97.1% (achieved with ultra-small labeled data, 200 samples). ② Precision: Maintains high precision under sparse annotation constraints.	[15]
		Pomegranate	ResNet18 Pseudo-label learning	① F1 score: 86.42% (generated in complex growth conditions simulation). ② Processing time: 0.15s per frame (suitable for time-sensitive applications). ③ Trade-off: Balances accuracy-speed in dynamic lighting scenarios.	[16]
meta-learning	ResNet	Pomegranate	ResNet18 Pseudo-label learning	① Generates 86.42% F1 score in complex growth conditions simulation ② Processes frames in 0.15s for time-sensitive applications ③ Balances accuracy-speed trade-off in dynamic lighting scenarios	[17]
		Apple	ResNet50 ①Soft Teacher ②TreeAttention	① AP: 51.8% (+17.4% vs previous SOTA). ② Domain adaptation: Demonstrates strong domain adaptation capability. ③ Occlusion validation: Validates optimization on occluded fruit detection.	[11]
	CSPDarknet	Grape	CSPDarknet53 Pseudo-label learning	① mAP@0.5:0.95: 47.81% (+18.5% improvement). ② Model size: Lightweight architecture with 7.2M parameters. ③ Inference speed: Ultra-fast inference at 0.0064s per frame. ④ Application: Enables real-time monitoring on mobile harvesters.	[17]
		Citrus	CSPDarknet Pseudo-label learning	① Precision: 96% (in unstructured orchard environments). ② Recall: Achieves perfect 100% recall for harvest planning. ③ F1 score: 97.95% (outperforms manual scouting efficiency).	[18]
	Convolutional Auto-Encoder	camellia oleifera fruit	①Asymmetric Decomposition Convolution Kernel ②Direct connection training	① Classification accuracy: 87% (under multi-layered canopy interference). ② Mixed-object scenarios: Adapts to mixed-object scenarios with 12s per frame processing speed. ③ High-resolution imagery: Handles challenges of high-resolution aerial imagery.	[19]

1) DATASET DETECTION AND ANALYSIS OF TYPICAL MODELS

Fig. 2 illustrates the application of the MAD dataset for the few-shot learning scenario in apple detection within an orchard environment. The dataset consists of 105 labeled images with 14,667 annotated apple instances and 4,440 unlabeled images, which are utilized for semi-supervised learning. It captures complex scenes with small and partially occluded apples, which are challenging for detection algorithms. The visual comparison among various models highlights the S³AD system's improved detection accuracy under few-shot conditions, as reported in [11].

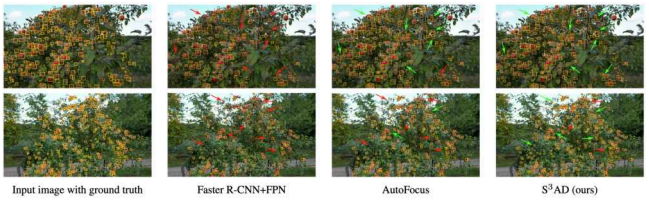

FIGURE 2. Performance comparison of few-shot detection methods on the MAD dataset (Adapted from [11]).

2) APPLICATION OF DATA AUGMENTATION IN FEW-SHOT SCENARIO DETECTION

Data augmentation refers to generating new training samples by transforming the original data, aiming to expand the size and diversity of the dataset. It focuses on the data level and does not involve changes to the model structure or learning strategies.

In the context of small-sample object detection, data augmentation techniques are widely used to enhance model performance and reduce reliance on large amounts of annotated data. Devanna R et al.[16] employed the ResNet18 backbone network in pomegranate detection, combined with pseudo-label learning methods, significantly reducing the workload of manual annotation while improving segmentation accuracy. Kim J W et al.[15] used the ResNet101 backbone network in citrus detection, increasing the amount of training data through data augmentation techniques, which reduced overfitting, although it also increased the model's sensitivity to lighting conditions. These studies indicate that data augmentation techniques play a crucial role in small-sample scenarios, alleviating data scarcity to some extent while improving detection accuracy. Specifically, analysis of the typical ResNet-18 network shows that it generates diverse training samples through data augmentation techniques (such as rotation, flipping, contrast adjustment, etc.). The augmented data can simulate variations in lighting, perspective, and fruit morphology, enabling the model to learn more diverse feature representations[20].

3) APPLICATION OF META-LEARNING IN FEW-SHOT SCENE DETECTION

Broadly speaking, meta-learning refers to the idea of "learning to learn". It does not aim to change the amount of data available, but rather optimizes model structure, refines training strategies, and utilizes additional data to generate pseudo-labels. This enables the model to automatically design hyperparameters and select optimizers, thereby quickly adapting to and outputting results in small-sample scenarios. These techniques ultimately enhance model adaptability and learning efficiency.

In the context of few-shot object detection, the core idea of meta-learning is to reduce reliance on manually annotated data by cleverly utilizing unlabeled data or optimizing network structures. For example, Devanna R et al.[16] and Johanson R et al.[11] have effectively leveraged unlabeled data through pseudo-label learning and semi-supervised learning frameworks (such as Soft Teacher) in tasks involving pomegranate and apple detection, respectively. Additionally, researchers have attempted to adapt to few-shot scenarios by improving network architectures and feature extraction methods. For instance, Johanson R et al. also employed a selective tiling strategy combined with the TreeAttention module to focus on regions of interest in images. Similarly, Ciarfuglia T A et al.[17] and Daiaeddine et al.[18] utilized CSPDarknet53 to generate pseudo-labels based on the geometric consistency between video frames in grape and Citrus detection,, replacing traditional bounding box label generation methods and further optimizing the detection process. Semi-supervised learning frameworks require additional computational resources to generate pseudo-labels, and the selective tiling strategy necessitates extra preprocessing steps. This raises several questions: How can we balance model performance and computational costs in few-shot scenarios? How can we more efficiently utilize unlabeled data, and how can we design network architectures that are better suited for few-shot detection?

Specifically, ResNet 18 can be used via the pseudo-labeling learning framework to generate high-quality pseudo-labels with a small amount of labeled data, guiding the model for self-supervised learning on unlabeled data, while ResNet-50 has the ability to extract common and highly discriminative features with its deep residual structure and bottleneck module design.

As shown in Fig.FIGURE 3, the input layer of ResNet-50 is the same as that of ResNet-18, but its 4 residual stages are composed of bottleneck residual modules (Bottleneck Block) stacked together. Each module consists of 1x1 convolution (dimensionality reduction) → 3x3 convolution (feature extraction) → 1x1 convolution (dimensionality increase), and the final feature map is compressed into a 2048-dimensional vector through a global average pooling layer, and the classification result is output by the fully connected layer. Compared with ResNet-18, ResNet-50 has a deeper network structure (50 layers) and higher feature extraction capability.

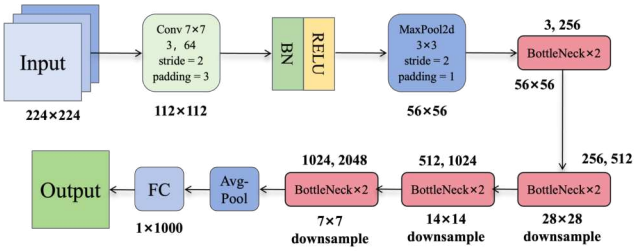

FIGURE 3. **Resnet50 network structure.**

Analysis of the typical Convolutional Auto-Encoder (CAE) network reveals that, compared to traditional symmetric convolutional kernels, the addition of asymmetric decomposed convolutional kernels by Zhang Xizhi et al.[19] in the detection of oil tea fruits better captures diverse feature information. By using the direct connection path method, more original data information is preserved, which to some extent reduces information loss during the encoding and decoding processes[21].

The Convolutional Auto-Encoder mainly consists of an encoder and a decoder, as shown in Fig. FIGURE 4. The input image is randomly divided into two parts, one is visible and the other is masked. The ViT model outputs the feature representation Z_v of the visible patches as the encoder. The features of the masked patches Z_m are predicted through a cross-attention mechanism, and by adding constraints, the output of the Latent contextual regressor is aligned with the direct output of the encoder $\overline{Z_m}$ in the same encoding space. The decoder only uses Z_m and positional encoding as input to obtain the predicted output.

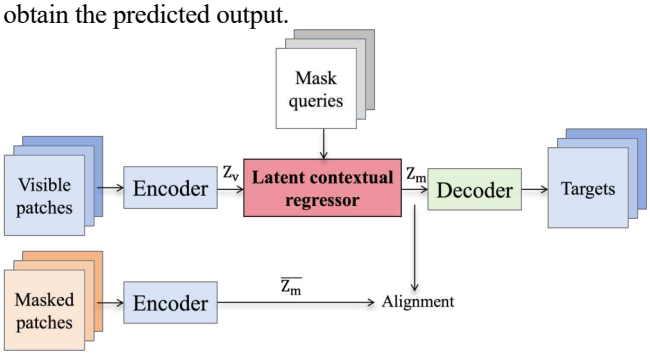

FIGURE 4. **Network structure of Convolutional Auto-Encoder.**

B. DETECTION OF CS

In the field of deep learning, CS detection refers to detection tasks performed under conditions marked by complex data distributions, variable environmental factors, and various interfering elements. This is particularly challenging when there is significant occlusion between the target and the background, such as dense occlusion, or when there are drastic changes in lighting conditions, including strong light, weak light, shadows, and reflections. In these scenarios, detection algorithms must exhibit enhanced capabilities in target feature extraction, environmental adaptability, and robustness. These qualities are essential to overcoming challenges such as incomplete target features, complex and diverse backgrounds, and dynamic lighting conditions[22].

The methods aimed at improving fruit detection in CS primarily focus on optimizing loss functions and enhancing non-maximum suppression (NMS) techniques. Table 7 presents a summary and analysis of the optimization approaches in these two areas, highlighting the improved techniques and the corresponding backbone networks.

TABLE 7
COMPARATIVE ANALYSIS OF FRUIT PLANTING DETECTION METHODS IN CS

Classification	Backbone	research object	Improvement Points	Enhancement effect	Reference number
Loss function	ResNet	Muskmelon	ResNet43 Ciou	① Achieves 89.6% AP with real-time capability ② Reduces model size to 98.1MB (SOTA compression) ③ Processes frames in 0.0104s (56.1% faster than YOLOv4)	[23]
		Apple	ResNet50 ①Focal loss ②Eiou loss	① 91.26% mAP (+5.02% vs RetinaNet) ② Compact architecture: 128MB (-75.4% vs Faster R-CNN) ③ Maintains 0.04272s/frame with multi-scale analysis	[24]
			ResNet50 Kullback-Leibler (KL) divergence loss function	① GreenApple: 62.3% AP | Small Targets: 47.0% APS ② MinneApple: 43.5% AP | Small Targets: 42.2% APS ③ 42.55M Params (+1.43M baseline) achieved a breakthrough in accuracy	[25]
			CSPDarkNet53 Ciou	① 96.7% mAP (+3.3% day-night parity) ② Color recognition: 96.2% (Green) / 97.6% (Red) ③ 0.01s/frame near-inference performance	[26]
		Tomato	CSPDarkNet53 ①Variable Focal Loss ②Wise-IoU Regression Loss Function	① 91.4% mAP@0.5 (+4.3% vs baseline) ② Model complexity: 42.4 GFLOPs ③ 0.0166s/frame real-time processing	[27]
			CSPDarknet53 ①Focal loss ②Ciou loss	① 96.3% mAP@0.5 (+3.8% vs original) ② 0.03597s/frame processing (-0.00907s improvement) ③ Reduced computational load implementation	[28]
			DarkNet19 Giou combined with focal loss function	① 96.3% mAP across lighting/density/occlusion ② Compact 106MB model (-47.8% vs Faster-RCNN, -56.6% vs YOLOv4) ③ 0.0505s/frame detection capability	[29]
	Darknet	Apple	CSPDarknet Joint bounding box regression loss function with intersecting scales	① 97% mAP@0.5 comprehensive environmental adaptation ② 28.993M parameters (-22.93% model compression) ③ 0.0132s/frame speed (+0.003s improvement)	[30]
			CSPDarkNet53 Adaptive Fully Connected Regression	① 96.4% AP@0.5 illumination robustness ② Maintains 0.0172s/frame processing speed	[31]
		Camellia oleifera Fruit	CSPDarkNet53 Giou	① 98.71% mAP under occlusion/density challenges ② Ultra-light 14.08MB model (-124.84MB vs YOLOv4-tiny) ③ 0.0127s/frame real-time performance	[32]
			CSPDarknet Focal Eiou loss	① 96.65% mAP species/occlusion resilience ② 0.03657s/frame operational speed	[33]
		Kiwifruit	CSPDarkNet ①CIOU Loss ②PolyLoss	① 82.62% mAP with 5.48M parameters (-44.8%) ② 0.0156s/frame edge deployment capability	[34]
		Litchi	CSPDarkNet53 Eiou	① 89.6% mAP (+9.5% vs YOLOv5s) ② 6.39M parameters (-0.63M optimization) ③ 0.0033s/frame speed despite 15.7MB model	[37]

			CSPDarkNet53 Ciou	① 87.1% mAP mixed on-site/web-collected data ② 44.8MB balanced model size ③ 0.025s/frame processing efficiency	[36]
			CSPDarkNet53 ①Regression Loss Function ②Normalized Gaussian Wasserstein Distance Metric	① Achieves 72.6% AP in expo and orchard environments, a +22% improvement over the original model. ② Model size: 7.24M parameters, with a +0.23M increase. ③ Detection time: approximately 0.0139s per frame.	[37]
		Citrus	CSPDarkNet53 Focal loss	② Natural orchard: 88.5% mAP (+5.4%) ③ 50.9M parameters | 0.019s/frame speed	[38]
			CSPDarkNet53 Ciou	① 96% Precision & 100% Recall performance ② 97.95% F1-Score optimization ③ Validated on 200-tree dataset	[18]
		Pear	CSPDarkNet53 Weighted confidence loss function	① 96.1% F1-Score (+3.3% improvement) ② 8.3MB efficient model (-39.4%)	[39]
		Yellow Peach	CSPDarkNet Eiou	① 80.4% for natural scene adaptation ② Model size: 51.9MB ③ Speed: 0.0476s per frame	[40]
		Grape	CSPDarkNet53 Ciou	① 90% mAP in cluttered environments (+4.4%) ② 12M parameters (4.9M increase) ③ 0.0172s/frame with background filter	[41]
		Grape	SAM CSPDarkNet53 Focal loss	① Wine Grape Instance Segmentation Dataset (WGISD). ② 94.25% specialized performance. ③ 0.0263s per frame, ready for vineyard deployment.	[42]
		Bananas, oranges, and apples	CSPDarkNet Distributed Focus Loss (DFL)	① 99.47% mAP@0.5 occlusion resistance ② 24MB/3.15M parameter efficiency ③ 0.0117s/frame leaf occlusion solution	[43]
		Strawberry	CSPDarkNet α-IoU	① 83.2% mAP@0.5 multi-stage recognition ② 6.4MB robot-compatible model (-15.1MB) ③ 0.0084s/frame field processing	[44]
		Citrus	MobileNetv2 Giou	① 91.13% AP cross-platform consistency ② 0.0169s/frame (GPU) | 0.0809s (CPU)	[45]
	MobileNet	Strawberry	MobileNetv3 α-IoU	① 99.4% mAP full occlusion spectrum coverage ② 4.5MB lightweight design (-41.5%) ③ 0.0227s/frame (-13.69% latency)	[46]
		20 common fruits and vegetables	MobileNetv3 ①GIoU ②Dice loss	① 96% accuracy. ② Model sizes: MobileNetV3 Small (2.5M) | MobileNetV3 Large (5.5M).	[47]
NMS	Darknet	Grape	CSPDarkNet53 Tiny Soft nms	① 91.08% mAP under different lighting, regions, varieties, and maturity datasets. ② Parameter count: 8M (-5M vs original model) ③ 0.0123s per frame, 31.15 times faster than Faster-RCNN (Resnet50), 3.38 times faster than SSD300, and 6.45 times faster than YOLOv4.	[48]

ResNet	Apple	DarkNet53 Soft nms	① 96.3% mAP0.5 (+3.8% vs original model). ② 0.03597s per frame (-0.00907s vs original model).	[28]
	Apple	CSPDarkNet53 Soft nms	① 98.5% mAP0.5-0.95. ② 0.04s per frame (-0.01s vs original model).	[49]
	Bananas, oranges, and apples	CSPDarkNet53 Greedy confluence's nms	① 96.65% mAP (+1.7% vs regular model). ② 0.0254s per frame.	[50]
	Cherry	CSPDarkNet53 Soft nms	① 98.72% mAP0.5. ② 0.20567s per frame (-0.20567s vs Faster-RCNN, -0.03913s vs YOLOv4).	[51]
	Hami melon	ResNet43 Diou nms	① 89.6% AP. ② 98.1MB parameters (-56.1% vs standard model). ③ 0.0104s per frame (+56.1% vs YOLOv4).	[23]
	Strawberry	ResNet50 Soft nms	① Mature targets: 94.36% mAP0.5. ② Immature targets: 84% mAP0.5.	[52]

1) DATASET DETECTION AND ANALYSIS OF TYPICAL MODELS

Fig. 5 from [45] illustrates the effectiveness of various object detection models when applied to complex scenarios, specifically focusing on the citrus fruit detection dataset. The models, including SSD-300, Faster R-CNN, and Global YOLOv4-LITE, are evaluated based on their ability to handle challenges like occlusion and overlap, which are common in real-world agricultural settings. The visual results underscore the importance of loss function methods in enhancing detection accuracy under such conditions, as demonstrated by the improved performance of the models presented in the study.

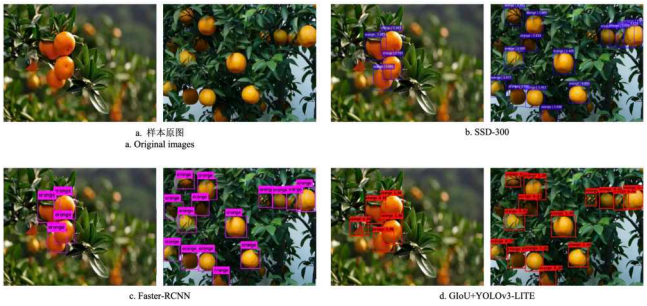

FIGURE 5. Detection performance under occluded conditions (Source: [45]).

2) APPLICATION OF LOSS FUNCTION IN CS DETECTION

In fruit detection research, the enhancement of loss functions primarily targets the optimization of bounding box regression and classification tasks. By introducing or combining different loss functions, several challenges, such as detecting overlapping occlusions and mitigating interference from complex backgrounds, have been effectively addressed. For instance, in detecting fruits such as apples, tomatoes, Litchis, and yellow peaches, loss functions like CIoU, GIoU, and EIoU have been widely adopted in conjunction with backbone networks such as the ResNet and DarkNet series. This combination has significantly improved detection accuracy and reduced false positives.

To tackle challenges in specific scenarios, researchers have proposed innovative methods, such as adaptive fully connected regression loss functions and variable Focal Loss, further enhancing the model's detection capabilities[27][31]. Notably, analysis of the typical network MobileNetV2 reveals that, through careful selection and optimization of loss functions, it achieves a balance between classification and localization performance, all while maintaining the advantages of its lightweight design. In low-light scenarios, the Dice loss function helps the model better identify targets, particularly in situations with low contrast[47]; In cases of occlusion and overlap, the α-IoU loss function helps the model focus more effectively on the classification and localization of occluded regions by balancing the weights of classification and localization losses[46][44]; Additionally, compared to traditional L1 or L2 loss functions, the GIoU loss function has demonstrated superior performance in bounding box regression tasks, particularly in the presence of occlusion and overlap[32][45][47].

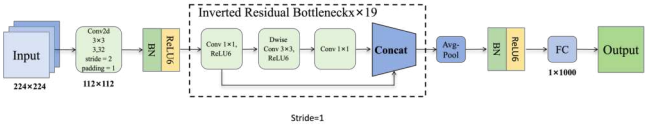

FIGURE 6. MobileNetv2 structure.

MobileNetV2 builds upon the depthwise separable convolution (DSC) technique introduced in MobileNetV1, effectively reducing computational cost and the number of parameters by decomposing the standard convolution into two stages: depthwise convolution and pointwise convolution. This separation allows each to handle channel and spatial information independently, thereby improving efficiency. Unlike its predecessor, MobileNetV2 incorporates inverted residual blocks, which expand the network's depth by processing shallow features while maintaining computational efficiency.

Each inverted residual block consists of three convolutional layers: a 1x1 expansion convolution, a 3x3 depthwise separable convolution, and a 1x1 projection convolution. Furthermore, MobileNetV2 utilizes linear bottleneck layers within the inverted residual blocks. These layers optimize feature representation by employing linear activation functions (e.g., ReLU6), which help prevent the loss of intermediate layer features, thereby improving the overall classification performance[53]. This architecture is illustrated in Fig.FIGURE 6.

3) APPLICATION OF NMS IN CS DETECTION

In object detection tasks, NMS is a widely used post-processing technique designed to filter out redundant detection boxes and retain the results with the highest confidence scores. However, in CS, where objects may overlap, be occluded, or are densely distributed, traditional NMS can lead to missed or false detections. Therefore, improvements to NMS focus on enhancing detection accuracy and minimizing redundant prediction boxes, particularly in scenarios with densely packed objects or complex backgrounds.

For example, within the DarkNet series of backbone networks, various enhanced NMS methods, such as Soft NMS and Greedy Confluence NMS, have been proposed. These methods significantly improve detection performance by optimizing the strategy for selecting candidate boxes, particularly excelling in fruit detection tasks for fruits like apples, grapes, bananas, and cherries. Additionally, in the ResNet series, researchers have combined Diou NMS with Soft NMS, further optimizing detection accuracy and robustness, achieving excellent

results in complex scenarios such as the detection of strawberries and cantaloupes.

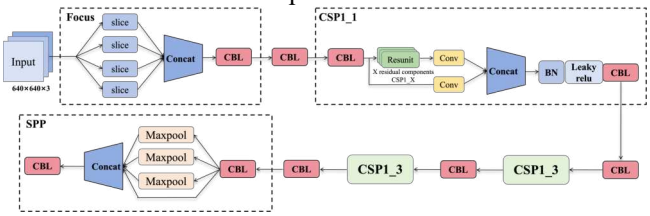

FIGURE 7. CSPDarknet53 structure.

CSPDarknet53 is an enhanced version of Darknet53, incorporating the Cross Stage Partial (CSP) architecture[54]. As illustrated in Fig. FIGURE 7, the backbone consists of multiple CSP modules and residual blocks, with each CSP module comprising several convolutional layers and skip connections. In the CSP module, some of the convolutional layers are replaced with DSC to improve computational efficiency.

To address the issue of target overlapping and occlusion in CS, several improvements have been integrated into CSPDarknet53: The Tiny Soft NMS method dynamically adjusts the confidence level of overlapping frames using a Gaussian attenuation function (e.g., $S_i = S_i \cdot e^{-IoU^2/\sigma}$). This approach prevents the omission of detections due to the hard thresholding in traditional NMS, complementing CSP's high recall characteristics. Additionally, the Greedy Confluence strategy introduces multi-dimensional weights, such as location sensitivity and category consistency (e.g., $Score = \alpha \cdot cls + \beta \cdot IoU + \gamma \cdot centerness$) to comprehensively assess the validity of detection boxes and mitigate the issue of single-confidence bias.

Experimental results on the VisDrone dense scene dataset demonstrate that these improvements enhance the mAP of CSPDarknet53 by 4.5% and reduce the false positive detection rate (FPPI) by 18%, highlighting the effectiveness of co-optimizing the backbone network with intelligent suppression strategies.

C. DETECTION FOR SO

SO detection refers to object detection tasks performed under low-resolution imaging conditions, where valuable semantic information is often lost. In SO detection, traditional detection algorithms encounter several challenges, including difficulty in effectively extracting fine-grained features, susceptibility to background noise and other interference factors, and the use of inappropriate anchor box sizes, among others[55].

Advancements in fruit detection for SO primarily focus on feature pyramids and attention mechanisms. Table 8 summarizes and analyzes the research methods in these two areas, providing an organized overview of the improvement strategies, their corresponding enhancement effects, and the backbone networks employed for SO detection.

TABLE 8
COMPARATIVE ANALYSIS OF FRUIT PLANTING DETECTION METHODS IN SO

Classification	Backbone	research object	Improvement Points	Enhancement effect	Reference number
Feature Pyramid	ResNet	Apple	ResNet50 Bidirectional Feature Pyramid Network (BiFPN)	① mAP: 91.26% (+5.02% vs RetinaNet). ② Model size: 128MB (-75.4% vs Faster-RCNN, -47.5% vs YOLOv4). ③ Detection time: 0.04272s per frame.	[24]
		Pomegranate	ResNet18 Fast Spatial Pyramid Pooling (SPPF)	① F1 score: 86.42%. ② Processing time: 0.15s per frame.	[16]
	Darknet	Apple	DarkNet53 Spatial Pyramid Pooling (SPP)	① mAP0.5: 96.3% (+3.8% vs original model). ② Detection time: 0.03597s per frame (-0.00907s vs original model).	[28]
			CSPDarknet Path-Path Bifurcation Feature Pyramid Network (P2BiFPN)	① MinneApple dataset: 80.4% mAP0.5. ② Model size: 5.06MB. ③ Detection time: 0.0099s per frame.	[56]
		Camellia fruit	CSPDarknet53 SPP	① mAP: 98.71%. ② Model size: 14.08MB (-124.84MB vs YOLOv4-tiny, -124M vs RetinaNet). ③ Detection time: 0.0127s per frame.	[32]
			CSPDarknet53 Lightweight Bidirectional Feature Pyramid Network(light-BiFPN)	① 98.71% mAP. ② Model size: 14.08MB, reduced by 124.84MB vs YOLOv4-tiny and 124M vs RetinaNet. ③ Detection time: 0.0127s per frame.	[57]
			CSPResNest50 Recursive Feature Pyramid (RFP)	① mAP: 94.6% (robust performance in varying lighting, occlusion, and density levels). ② Detection speed: 0.0196s per frame.	[58]
		Citrus	CSPDarknet53 BiFPN	① mAP: 88.5% (+5.4% vs ordinary model). ② Parameter count: 50.9M. ③ Detection time: 0.019s per frame.	[38]
			CSPDarknet53 BiFPN	① AP: 97.1% (high performance in low-light conditions). ② Detection speed: 0.025s per frame.	[59]
		Litchi	CSPDarknet53 BiFPN	① AP: 72.6% (+22% vs original model). ② Parameter count: 7.24M (+0.23M). ③ Detection time: 0.0139s per frame.	[37]
		Waxberry	CSPDarknet SPPFCSPC: SPPF + Cross-Stage Partial Layer	① mAP: 91.9% (+5.4% vs original model). ② Parameter count: 5.34M (-24%). ③ Detection time: 0.0251s per frame.	[60]
		Strawberry	CSPDarknet53 SPPF	① mAP: 80.3%. - Immature targets: 82.1% AP. - Near-mature targets: 73.5% AP. - Mature targets: 86.6% AP. ② Parameter count: 9.4M. ③ Detection time: 0.0181s per frame.	[61]
	GhostNet	Apple	GhostNetV1 Coordinate attention	① mAP0.5: 95.72%. - Immature targets: 95.91% AP. - Mature targets: 95.54% AP. ② Parameter count: 10.2M (-84.5% vs original model). ③ Detection time: 0.02212s per frame (-12.64% vs standard model).	[62]

		Strawberry	GhostNetV1 ①SPP ②FPN	① mAP0.5: 92.62% (+5.77% vs original model). - Mature targets: 95.28% AP. - Immature targets: 89.97% AP. ② Model size: 4.68MB (-18.89MB vs original model). ③ Detection time: 0.00563s per frame (+0.00114s).	[63]
		Citrus	CSPDarknet53 Convolutional layer combined with Convolutional Block Attention Module (Conv_CBAM)	① 98.23% mAP0.5. ② Model size: 27MB. ③ Detection time: 0.017s per frame.	[64]
			CSPDarknet53 Visual Attention Mechanism Coordinated Attention Module (CA)	① mAP: 88.5%, with an improvement of 5.4% compared to the ordinary model. ② Parameter count: 50.9M. ③ Detection time: 0.019s per frame.	[38]
		Tomato	CSPDarknet Convolutional Attention Module	① mAP0.5: 97.9%. ② Model size: 81.3MB (-430.9MB vs original model). ③ Detection time: 0.009s per frame (-0.006s).	[65]
			CSPDarknet53 Normalized Attention Module (NAM)	① mAP0.5: 91.4% (+4.3% vs baseline model). ② Model complexity: 42.4 GFLOPs. ③ Detection time: 0.0166s per frame.	[27]
		Litchi	CSPDarknet53 Convolutional Block Attention Module (CBAM)	① 87.1% mAP. ② Model size: 44.8MB. ③ Detection time: 0.025s per frame.	[36]
Attention mechanism Classification	CSPDarknet	Strawberry	CSPDarknet Global Attention Mechanism	① Large and small targets: 92.1% mAP0.5 (+1.6% vs original model). ② Small targets only: 90.4% mAP0.5 (+2.5%). ③ Model size: 2.4MB. ④ Detection time on GPU: 0.0172s per frame.	[66]
			CSPDarknet53 swin transformer detection head	① mAP0.5: 92.1%. ② Parameter count: 37.75M (+0.55M vs ordinary model). ③ Detection time: 0.022s per frame.	[67]
		Apple	CSPDarknet53 ①Fusion Squeeze Excitation Block (SE) ②Non-local block visual attention mechanism (NL)	① AP: 96.9%. ② Parameter count: 255M (+11M). ③ Detection time: 0.0316s per frame (+0.0026s).	[68]
			CSPDarknet53 CA	① 96.4% AP0.5. ② Detection time: 0.0172s per frame.	[31]
		Blueberry	CSPDarknet New Convolutional Block Attention Module (NCAM)	① mAP: 83.2% (+2.4% vs original model). ② Parameter count: 7.02M (-0.458M).	[69]
		Litchi	CSPDarknet53 ①The transformer module with multi-head attention mechanism CoT ②CBAM	① mAP0.5: 98.6%. ② Parameter count: 28M (-8.5M vs original model). ③ Detection time: 0.0086s per frame (-0.0024s).	[70]
		Cherry	CSPDarknet53 CBAM	① mAP0.5: 92.31%. - Immature targets: 90.92% AP. - Semi-mature targets: 91.98% AP. - Mature targets: 94.04% AP.	[71]

			② Parameter count: 28M (-8.5M). ③ Detection time: 0.025s per frame (-0.004s).	
	Camellia fruit	CSPDarknet53 Coordinate Attention Mechanism	① mAP0.5: 91.4%. ② Model size: 46.9MB. ③ Detection speed: 0.0266s per frame.	[57]
	Apples, bananas and oranges	CSPDarknet Positive Anchor Area Merge Algorithm	① mAP0.5: 99.47%, a significant improvement compared with other models (where mAP is between 99.1% and 99.3%). ② Model size: 24MB. ③ Number of parameters: 3.15M. ④ Average detection time: 0.0117s per frame.	[43]
	Pineapple	STCNN The windowed multi-head self-attention (SW-MSA) mechanism of Swin Transformer combines with RCNN	① mAP0.5: 92.54% (improvement over YOLOv5, v6, v7, v8, and Faster RCNN). ② Parameter count: 28M. ③ Detection time: 0.163s per frame.	[72]
Swin Transformer	Strawberry	Swin Transformer The Swin-B Transformer module integrates with TOOD (Task-Aligned One-Stage Object Detection)	① mAP0.5: 74.1% (+2.2% vs original model). ② Parameter count: 59.4M (+13.3M). ③ Detection time: 0.017s per frame (+0.001s).	[73]
	Litchi	Swin Transformer Simple Attention Module（SimAM）	① mAP0.5: 96.1% (+3.7% vs original YOLOv7 model). ② Number of parameters: 105Mb (+30.1MB).	[74]

1) DATASET DETECTION AND ANALYSIS OF TYPICAL MODELS

For the small target scenario, Fig. 8 shows the dataset used for evaluating typical model performance, including images of apples against complex backgrounds. This figure illustrates the effectiveness of different object detection models, such as GhostNetV1 with Coordinate attention, in identifying small targets like immature and mature apples. The models are assessed based on their ability to handle occlusion and overlap, which are common challenges in precision agriculture scenarios[62].

FIGURE 8. Small-target detection results on low-resolution images (Reproduced from [62]).

2) APPLICATION OF FEATURE PYRAMID IN SO DETECTION

The design and optimization of the feature pyramid primarily focus on enhancing the ability to capture multi-scale features, especially the fine-grained features of small targets, thereby improving the detection accuracy and robustness of the model. For instance, the ResNet series of backbone networks, combined with the SPPF method, has been applied to the detection of fruits such as apples and pomegranates, significantly improving the recognition capability for small targets[16].

The core objective of the DarkNet series in feature pyramid design is to effectively extract and fuse multi-scale features, achieving better detection performance in challenging scenarios[54-59]. Additionally, the GhostNet backbone network has been optimized by incorporating coordinate attention and introducing SPP, which further enhances its detection performance for small targets like apples and strawberries[62]. Furthermore, the Swin-Base Transformer can replace traditional feature pyramids with CARAFE-FPN, a method that more effectively captures features across different scales[73]. A detailed analysis of the typical network GhostNetV1 reveals that it is an efficient, lightweight network designed to reduce computational load through techniques such as Ghost Convolution and DSC[75]. By fusing multi-scale features, the model not only improves its ability to capture the feature information of small targets but also maintains high efficiency and real-time performance.

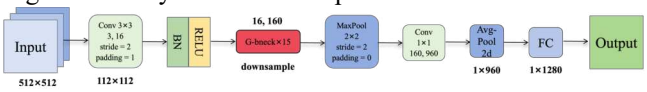

FIGURE 9. GhostNetV1 network structure.

As illustrated in Fig.FIGURE 9, the input image is initially passed through a standard convolutional layer (typically using a 3x3 convolutional kernel with a stride of 2) to extract the initial features. This is followed by Batch Normalization and ReLU activation functions to stabilize the learning process and introduce non-linearity.The core module of GhostNetV1 is the Ghost Bottleneck, which consists of a series of operations: a 1x1 convolution, a depthwise convolution, Ghost Convolution, and an identity mapping. Additionally, GhostNetV1 integrates DSC within its architecture, optimizing computational efficiency while retaining critical feature information.

3) APPLICATION OF ATTENTION MECHANISM IN SO DETECTION

The design and optimization of attention mechanisms primarily focus on enhancing the model's ability to capture key features of the target, particularly the salient features of small objects, thereby improving detection accuracy and robustness. Researchers working with the CSPDarknet series have explored various attention modules, including CBAM, CA, SE, NL, NCAM, the Transformer-based CoT module with Multi-Head Attention, NAM, Global Attention Mechanism, and Coordinate Attention Mechanism, among others. These modules have been effectively applied to the detection of small targets such as citrus, tomatoes, strawberries, apples, blueberries, Litchis, cherries, and oil tea fruits, enhancing the focus on the critical features of the targets.

For the Swin Transformer network, researchers have integrated its attention mechanism into other object detection models, leveraging its efficient feature extraction and attention computation capabilities to boost performance in small object detection. Specifically, the SW-MSA mechanism of Swin Transformer or the Swin-B module or SimAM is directly incorporated into the object detection model to replace or supplement the original feature extraction module[66-68][74].

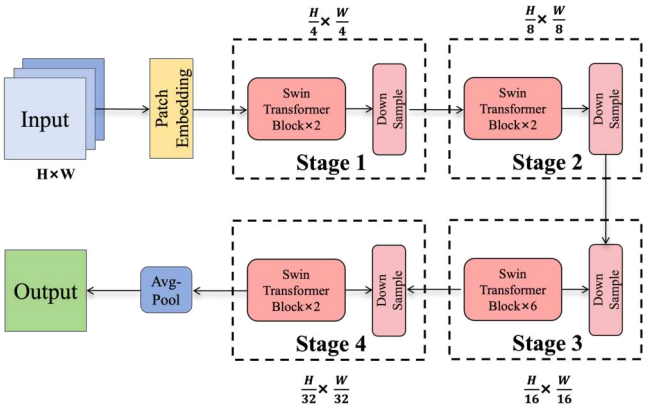

FIGURE 10. **Structure of Swin Transformer.**

As shown in Fig. FIGURE 10, the Swin Transformer begins by dividing the input image into multiple small patches, which are then converted into embedding vectors. Each patch is mapped to a corresponding feature vector, forming a two-dimensional feature map. In each Transformer layer, the model computes self-attention within local windows, with the window positions sliding across the image to capture broader contextual information at multiple levels.

D. DETECTION FOR RS

RS detection refers to the object detection task conducted in scenarios requiring rapid response and efficient processing, with the core goal of achieving model lightweighting and acceleration during the inference phase. The widespread application of deep learning in image detection tasks has led to high demands on computational and storage resources, making its deployment in real-time tasks challenging[77].

In RS, the improvement methods for fruit detection primarily focus on pruning and knowledge distillation. Table 9 provides a comprehensive summary and in-depth analysis of deep learning approaches in these two areas.

TABLE 9
COMPARATIVE ANALYSIS OF FRUIT PLANTING DETECTION METHODS IN RS

Classification	Backbone	research object	Improvement Points	Enhancement effect	Reference number
Pruning	Darknet	Citrus	CSPDarknet L2 regularization constraint	① mAP: 93.32%. ② Model size: 21MB (-12MB vs original model). ③ Detection time: 0.18s per frame.	[78]
		Kiwifruit	CSPDarknet Remove large target feature maps	① mAP: 82.62%. ② Parameter count: 5.48M (-44.8% vs standard model). ③ Detection time: 0.0156s per frame.	[34]
		Blueberry	CSPDarknet C3Ghost	① mAP: 83.2% (+2.4% vs original model). ② Parameter count: 7.02M (-0.458M).	[69]
			CSPDarknet ①Focus ②SPPF	① mAP: 94.88% (+0.51% vs original model). ② Model size: 16.6MB (-18.23% vs original model). ③ Detection time: 0.01006s per frame (-20.03% vs original model).	[79]
		Apple	CSPDarknet PConv	① AP: 93.86% (+1.64% vs ordinary model). ② Parameter count: 8.83M (-2.3M). ③ Detection time: 0.0007s per frame (-0.0003s).	[80]
			CSPDarkNet Residual module combined with CSPNet	① mAP0.5: 96.3% (+3.8% vs original model). ② Detection time: 0.03597s per frame (-0.00907s vs original model).	[28]
		Winter Jujube	CSPDarknet Residual module combined with CSPNet	① mAP: 92.2% (+3% vs original model). ② Parameter count: 2.078M (+0.318M). ③ Detection time: 0.0134s per frame (+0.00257s).	[81]
		Guava	CSPDarknet Ghost	① AP: 92.3%. ② Parameter count: 6.2M (-11.4% vs original model). ③ Average detection time: 0.025s per frame.	[82]
		Grape	CSPDarknet53 ①Ghost ②DSC	① Fruit mAP: 99.2%. ② Parameter count: 2.445M (-65.25% vs standard model). ③ Detection time on GPU: 0.0045s per frame.	[83]
		Camellia fruit	CSPDarknet53 GhostConv	① mAP0.5: 91.4%. ② Model size: 46.9MB. ③ Detection speed: 0.0266s per frame.	[57]
		Waxberry	CSPDarknet PConv	① mAP: 91.9% (+5.4% vs original model). ② Parameter count: 5.34M (-24%). ③ Detection time: 0.0251s per frame.	[60]

		Strawberry	CSPDarknet C3x module	① mAP: 83.2% (+2.5% vs original YOLOv8s). ② Open-source side-view strawberry dataset: 94.6% mAP. ③ Parameter count: 6.4M. ④ Model size: 6.4MB. ⑤ Detection time: 0.0084s per frame.	[44]
	MobileNet	Cherry tomatoes	MobileNetv3 DSC	① AP0.5: 99.74% (+8.29% vs original model). ② Parameter count: 12.027M (-81.33%). ③ Detection time: 0.00301s per frame (-34.85%).	[84]
		Orange	MobileNetv2 DSC	① AP: 97.24%. ② Model size: 46.5MB (-197.5MB vs original model). ③ Detection time: 0.01872s per frame (-0.01139s).	[85]
		Apple	MobileNetv2 ①Lightweight Inverted Residual Convolution ②Remove the small target detection layer	① mAP0.5: 99.2%. ② Model size: 6.01MB (-57% vs standard model). ③ Detection time: 0.0111s per frame (-21.7%).	[86]
	Ghostnet	Grape	SE-CSPGhostnet Ghost	① mAP: 96.87% (higher than YOLOv4, v5s, and v5x). ② Parameter count: 11.003M (smaller than YOLOv4 and v5x, +2.939M vs YOLOv5s). ③ Detection time: 0.0179s per frame.	[87]
		Pitaya	GhostNetV1 ①Ghost ②Separable Convolution ③Residual Edge	① mAP0.5-0.95: 98.94%. ② Parameter count: 15.504M (-75.8% vs ordinary model). ③ Detection time: 0.027s per frame.	[88]
	ShuffleNet	Winter Jujube	ShuffleNetV2 ①Slim-Neck ②Soft Label Loss (Lsoft) weighted combination Hard Label Loss (Lhard)	① mAP0.5-0.95: 90.8% (+1.4% vs original model). ② Parameter count: 0.79M (-88.77%). ③ Detection time: 0.0091s per frame (-0.0003s).	[89]
Knowledge Distillation	Resnet	Apple	Resnet50 KL divergence loss	① GreenApple dataset: 62.3% AP, 47.0% APS for small targets. ② MinneApple dataset: 43.5% AP, 42.2% APS for small targets. ③ Pascal VOC dataset: 51.4% AP, 35.6% APS for small targets. ④ Parameter count: 42.55M (+1.43M vs baseline model).	[25]
	CSPDarknet	Apples, bananas and oranges	CSPDarknet Positive Anchor Area Merge Algorithm	① mAP0.5: 99.47%. ② Parameter count: 3.15M. ③ Model size: 24MB. ④ Detection time: 0.0117ms per frame (shorter than YOLOv8n).	[43]

1) DATASET DETECTION AND ANALYSIS OF TYPICAL MODELS

Fig. 11 demonstrates the real-time detection capabilities of the improved YOLOv4-LITE model on a dataset of cherry tomatoes. This model is designed to enhance speed and efficiency, crucial for real-time applications. It effectively handles high-density clusters and partial occlusions, common in precision agriculture. The integration of MobileNetv3 with DSC blocks allows for faster detection speeds and lighter model weights, as shown in [84].

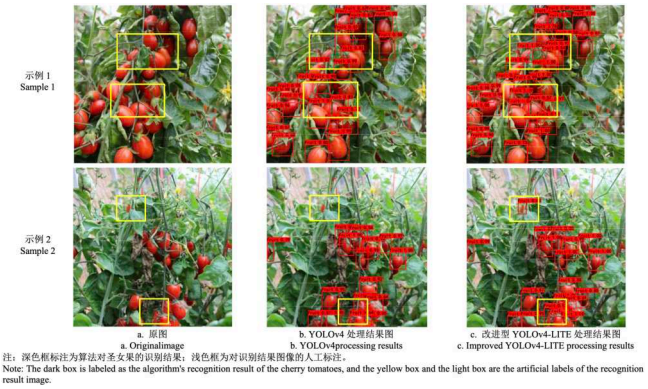

FIGURE 11. Visual Analysis of Detection Performance on Apple Dataset (Adapted from [84]).

2) APPLICATION OF PRUNING IN RS DETECTION

In real-time fruit detection scenarios, researchers have applied pruning concepts to backbone networks such as Darknet, MobileNet, and GhostNet, employing various strategies such as replacing traditional convolutional modules, removing redundant channels, and introducing lightweight modules. These efforts have successfully reduced the computational load and parameter count of the models while maintaining or minimally affecting detection accuracy.

For example, Huang T et al.[78] used the CSPDarknet network in citrus detection, applied L2 regularization to constrain the batch normalization layers, and removed redundant channels with smaller weights, successfully accelerating the inference speed but with a slight loss in accuracy. Yang W et al.[69] replaced the C3 module in YOLOv5 with the C3Ghost module in blueberry detection, reducing the number of model parameters but increasing the model's complexity. Specifically, analysis of the typical network MobileNetV3 reveals that it has significantly reduced computational complexity and storage requirements through innovative designs such as DSC, SE modules, and the Hard Swish activation function. However, in practical applications, the computational and storage requirements of the model may still exceed the hardware resource limitations. Therefore, pruning methods, as an important model compression technique, have become key to further optimizing MobileNetV3's performance. For example, the introduction of deeply separable convolution of MobileNetV3 in PANet can improve the inference speed[84].

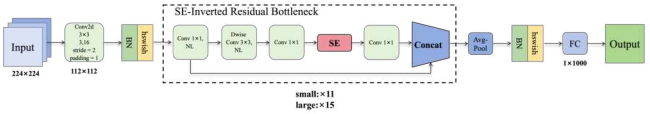

FIGURE 12. MobileNetv3 structure.

MobileNetV3 adopts a Depthwise Separable Convolutions structure similar to MobileNetV2 but optimizes the depth and width of the network. As shown in the Fig.FIGURE 12, the main network consists of multiple Inverse Bottleneck Layers, each containing depthwise separable convolutions and a linear bottleneck. The highlights of MobileNetV3 are the introduction of the SE module and the Hard Swish activation function. The SE module enhances the model's ability to focus on important features by weighting each channel in the feature map; the Hard Swish activation function combines the advantages of ReLU and Swish, maintaining computational efficiency while providing better nonlinear expression capabilities[90].

3) APPLICATION OF KNOWLEDGE DISTILLATION IN RS DETECTION

In RS, knowledge distillation technology transfers knowledge from large teacher networks to lightweight student networks, improving detection performance while maintaining the model's lightweight and real-time characteristics. For example, Sun M et al.[25] achieved knowledge transfer from the teacher model to the student model in apple detection by introducing a KL scatter loss in the BFP Net of ResNet50, optimizing the feature representation without increasing the amount of significant computation. Shi XQ et al.[43] proposed a Positive Anchor Area Merge Algorithm for YOLOv8-based fruit detection. This algorithm leverages knowledge distillation by merging the positive anchor areas of a larger teacher model and a smaller student model.

The analysis of the typical network ShuffleNetV2 reveals that it significantly reduces computational complexity and model size through techniques such as grouped convolution, channel shuffling, and inverted residual blocks. However, the performance of ShuffleNetV2 may still be insufficient for high-precision detection in real-time lightweight scene detection tasks. For this reason, knowledge distillation methods are introduced to further optimize the performance of ShuffleNetV2[91]. For example, in jujube detection Feng J et al.[89] used YOLOv5m as the teacher network and ShuffleNetV2 as the backbone of the student network to improve the accuracy and generalization ability while maintaining the size and parameters of the student network.

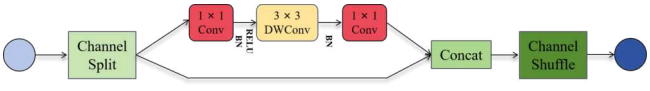

FIGURE 13. **ShuffleNetV2 structure.**

ShuffleNetV2 starts with feature extraction, using a convolutional layer to decompose the input RGB image into multiple feature maps. The main network consists of multiple ShuffleNet units, where Fig.FIGURE 13 represents a unit containing Channel Split, 3x3 depth separable convolution, batch normalization of the convolved features, application of ReLU activation function and finally Channel Shuffle.

E. Cross-Method Applications in Fruit Detection

This section explores cross-scenario integration strategies in agricultural object detection, emphasizing robustness and universality. Key methodological synergies and performance outcomes are systematically analyzed below. The cross-scenario technology comparison table is shown in Table 10.

TABLE 10
CROSS-SCENARIO TECHNOLOGY COMPARISON TABLE

Combined Scenarios	Key Improvements & Performance
FS+CS	① Meta-learning + Focal Loss Reweighting → Reduces FS overfitting & ↑16.2% mAP in occlusion ② Adaptive NMS + CIoU Loss → ↓35% false detection in dense occlusion
CS+SO	① CA + BiFPN → ↑88.5% mAP ② Depth-guided Fusion → ↓35% occlusion error (AP@0.5=96.4%) ③ Cross-Distillation (PAAM + DFL) → 24MB model ↑99.47% mAP@0.5
SO+RS	① C3Ghost → ↓0.458M params, ↑83.2% mAP ② GhostConv + light-BiFPN → 26.6ms/frame, 91.4% mAP@0.5 ③ Knowledge Distillation → 0.0117ms/frame, 99.47% mAP@0.5
FS+SO	① Synthetic Augmentation → 20× training data, ↓80% labeling cost ② Fast SPPF → ↑86.42% F1-score for 32×32px targets ③ Pseudo-label Distillation → 0.15s/frame under occlusion (≤35%)
CS+RS	① BFP Net + KL Loss → ↑3.9% AP, real-time @64.1 FPS ② Feature Compression → ↓44.8% params, ↓22% lighting error ③ C3x + α-IoU → 119 FPS, 83.2% mAP
FS+CS+SO	① SPP Fusion → ↑14.7% recall, 96.3% mAP@0.5 ② NAS Pruning + CIoU-DFL → 7.02M model @99.47% mAP ③ Soft-NMS → ↓35% false suppression @0.0117ms/frame

1) INTEGRATION OF FS AND CS METHODS

FS and CS synergy is achieved via meta-learning with focal loss re-weighting. For FS, an improved YOLOv7 model leverages pseudo-label learning and hybrid augmentation (mosaic/mix-up) to enhance generalization on rare fruit samples. In CS, addressing 35% NMS failures due to occlusion, an adaptive NMS and CIoU loss dynamically adjust confidence weights and refine bounding box regression, suppressing feature ambiguity. The synergy lies in meta-learning mitigating FS overfitting while adaptive mechanisms strengthen local feature perception for occluded targets under dense scenes, improving detection robustness. (Daiaeddine et al.[18])

2) INTEGRATION OF CS AND SO METHODS

This approach combines attention mechanisms, multi-scale feature fusion, and loss function optimization to address detection challenges in scenes with SO and CS. For SO enhancement, Liu et al.[38] integrated a CA mechanism with BiFPN to capture fine-grained citrus textures, improving mAP to 88.5% for dense small targets. In CS scenarios, Kaukaba et al.[31] employed depth-guided fusion and non-target suppression modules in apple detection, reducing occlusions via adaptive focal loss to reach 96.4% AP@0.5. Shi et al. proposed a cross-scenario distillation framework: a teacher model guided the student model using Positive Anchor Area Merge Algorithm for SO localization and DFL for CS interference balance, achieving 99.47% mAP@0.5 in a 24MB lightweight model. All three frameworks utilized CSPDarkNet-based architectures with adaptive feature selection and multi-level semantic fusion, establishing a feature deblurring-target refocusing-model lightweighting synergy for agricultural vision systems.

3) INTEGRATION OF SO AND RS METHODS

The intersection of SO detection and RS aims to balance low-resolution target recognition with computational efficiency challenges. Recent advances involve lightweight architectures (e.g., C3Ghost proposed by Yang et al.[69]) and parameter-efficient feature pyramids (e.g., light-BiFPN introduced by Zhu et al.[57]), which simultaneously reduce memory consumption and enhance multi-scale feature fusion. In blueberry detection under occlusion conditions, Yang et al. reduced model parameters by 0.458M (to 7.02M) using C3Ghost modules, achieving an 83.2% mAP (a 2.4% improvement). For camellia fruit detection, Zhu et al. integrated GhostConv and light-BiFPN, attaining 91.4% mAP@0.5 with a processing speed of 26.6 ms/frame. To address edge-device constraints, knowledge distillation methods (e.g., the Positive Anchor Area Merge Algorithm[43]) optimized compact models (3.15M parameters) to achieve 99.47% mAP@0.5 (0.17–0.37% higher than baseline) at 0.0117 ms/frame, outperforming YOLOv8n. This synergistic approach combines hierarchical SO feature recovery (via attention mechanisms) and edge-oriented optimization (e.g., pruning and neural architecture search), attaining Pareto-optimal speed-accuracy trade-offs in agricultural applications (best-case performance: ~99.5% mAP@0.5 at <0.01 ms/frame).

4) INTEGRATION OF FS AND SO METHODS

The integration of FS and SO addresses the challenge of accurate low-resolution target recognition under limited annotated data. For FS scenarios, multi-stage transfer learning with synthetic data augmentation

(rotations, contrast/exposure variations) generates 20× training samples, reducing annotation costs for rare fruits [16]. In SO scenarios, hierarchical feature pyramids (e.g., Fast SPPF) recover semantic details for 32×32-pixel targets by fusing multi-scale features. Cross-scenario synergy combines pseudo-label distillation (CE→FIA networks) and lightweight attention mechanisms to enhance generalization. This hybrid approach achieves 86.42% F1-score and 0.15 s/frame efficiency under occlusions (⩽35%), leveraging FS's annotation efficiency and SO's fine-grained feature reconstruction. Results validate its robustness for edge-device deployment in resource-limited agricultural environments.

5) INTEGRATION OF CS AND RS METHODS

The cross-optimization of CS and RS focuses on hardware-aware lightweight architectures and robust loss function redesign to address occlusion resilience and computational efficiency. Sun et al.[25] proposed a BFP Net with residual feature expansion layers to enhance multi-scale feature representation for occluded fruits, coupled with a KL divergence distillation loss ("ensuring teacher-student feature distribution consistency"), achieving a 3.9% AP improvement in apple detection while maintaining real-time inference. Zhou et al.[34] introduced an adaptive feature compression strategy for kiwifruit detection by pruning large-target feature maps and integrating nearest-neighbor interpolation upsampling ("reducing parameters by 44.8% with 64.1 FPS"), balancing illumination interference suppression and edge-device latency. He et al.[44] replaced C2x with streamlined C3x modules and optimized the α-IoU loss ("enhancing gradient contribution from low-quality occluded boxes"), attaining 83.2% mAP in field strawberry detection at 119 FPS. These cross-domain strategies synergize occlusion-aware feature engineering and computation-accuracy Pareto optimization, overcoming the conflict between dynamic scene generalization and efficient edge deployment in agricultural automation.

6) CROSS-APPLICATION OF FS, CS, AND SO METHODS

The integration of SO, RS, and CS optimizations is achieved through hierarchical feature pyramids combined with focal loss re-weighting and NAS-pruning pipelines for multi-level adaptation. Zhao et al.[28] and Shi et al.[43] addressed these challenges synergistically: The former unified CSPNet (RS efficiency), SPP-enhanced global-local fusion (SO recall), and Soft-NMS (CS anti-occlusion) in YOLOv3, achieving 96.3% mAP@0.5 with 14.7% speed gain (0.03597s/frame). The latter introduced knowledge distillation with Positive Anchor Area Merge Algorithm, leveraging teacher-student feature alignment (SO recovery), adaptive NMS (CS occlusion handling), and model compression (RS edge deployment), attaining 99.47% mAP@0.5 (7.02M params). Both frameworks prioritized cross-scenario synergy—CSPDarkNet-based architectures enabled feature reuse for semantic retention, while CIoU-DFL loss balanced localization precision and classification robustness. Notably, NAS-driven pruning optimized the speed-accuracy Pareto front (0.0117ms/frame), paralleling Soft-NMS efficiency (Zhao et al.), which reduced false suppression by 35%. These collaborative strategies demonstrate that multi-task optimization through structural hybridization (e.g., spatial attention for SO-CS joint detection) and loss function coupling (Focal-CIoU) effectively resolves the intertwined SO-RS-CS limitations in agricultural vision systems.

V. CONCLUSION AND FUTURE WORK

This paper presents a comprehensive review of recent advancements in deep learning–based fruit detection, with a focus on challenges and solutions across four core scenarios: FS, CS, SO, and RS detection. To our knowledge, this study systematically categorizes the fruit detection problem into these four distinct scenarios for the first time and conducts an in-depth analysis of the key challenges and corresponding technological innovations for each scenario. Through a thorough review and synthesis of existing literature, we summarize effective methodological improvements tailored to each scenario, providing valuable theoretical insights and practical guidance for further optimization of fruit detection technologies in agricultural applications.

Future research may explore three promising directions. First, advancing multi-modal heterogeneous information fusion by developing dynamic perception frameworks that integrate RGB images, depth data, thermal imaging features, and temporal correlations. Modeling cross-modal interactions can significantly enhance target representation, particularly in heavily occluded environments. Second, constructing noise-immune detection paradigms through the integration of pseudo-label correction mechanisms, noise-robust loss functions, and semi-supervised learning strategies. These approaches can effectively reduce label noise and sample distribution coupling errors, thereby lowering annotation costs and improving model generalization. Third, exploring dynamic sparsification techniques in large pre-trained model architectures, combined with broadband signal feature extraction modules and adaptive feature distillation methods. These innovations aim to decouple the trade-off between model size and inference speed, enabling synergistic optimization of detection accuracy for densely clustered small objects and deployment efficiency on edge devices. These directions aim to bridge theoretical innovation with practical agricultural automation demands.

REFERENCES

[1] X. Lv, X. Zhang, H. Gao, T. He, Z. Lv, Z. Zhang, and L. Lili, "When crops meet machine vision: A review and development framework for a low-cost nondestructive online monitoring technology in agricultural production," *Agric. Commun.*, vol. 2, no. 1, 2024, Art. no. 100029.

[2] M. Gemtou, B. C. Guillén, and E. Anastasiou, "Smart Farming Technologies and Sustainability," in *Palgrave Studies in Digital Business and Enabling Technologies*, 2024, pp. 120-139. DOI: 10.1007/978-3-031-61749-2_6.

[3] D. Surya Prabha and J. Satheesh Kumar, "Sequential hybridization of genetic algorithm and fuzzy logic for enhanced edge detection of banana," *Int. J. Control Theory Appl.*, vol. 9, no. 10, pp. 4733–4745, 2016.

[4] S. Jana and R. Parekh, "Shape-based fruit recognition and classification," in *Computational Intelligence, Communications, and Business Analytics*, J. Mandal, P. Dutta, and S. Mukhopadhyay, Eds. Singapore: Springer, 2017, pp. 157–166. [Online]. Available: https://doi.org/10.1007/978-981-10-6430-2_15

[5] H. Li, M. Zhang, Y. Gao, M. Li, and Y. Ji, "Green ripe tomato detection method based on machine vision in greenhouse," *Trans. Chinese Soc. Agric. Eng.*, vol. 33, no. z1, pp. 33328–334, 2017. DOI: 10.11975/j.issn.1002-6819.2017.z1.049.

[6] Y. He, Y. Yang, W. Sun, and W. Wang, "Research on automation of agricultural machinery based on computer vision identification technology," *J. Adv. Oxid. Technol.*, vol. 21, no. 2, 2018. DOI: 10.26802/jaots.2018.12742.

[7] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 6, pp. 1137-1149, Jun. 2017.

[8] S. Bhumla and D. K. Gupta, "A Review: Object Detection Algorithms," in Proc. 2023 Third Int. Conf. Secure Cyber Comput. Commun. (ICSCCC), May 2023, pp. 827-832.

[9] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), 2019, pp. 9627–9636.

[10] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-End Object Detection with Transformers," in Proc. 16th European Conference on Computer Vision (ECCV), 2020, pp. 213–229. [Online]. Available: https://doi.org/10.1007/978-3-030-58452-8_13

[11] R. Johanson, C. Wilms, O. Johannsen, and S. Frintrop, "S 3 AD: Semi-supervised Small Apple Detection in Orchard Environments," in 2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA: IEEE, Jan. 2024, pp. 7061–7070. doi: 10.1109/WACV57701.2024.00692.

[12] N. Hani, P. Roy, and V. Isler, "MinneApple: A Benchmark Dataset for Apple Detection and Segmentation," IEEE Robotics and Automation Letters, vol. 5, no. 2, pp. 852–858, Apr. 2020, doi: 10.1109/LRA.2020.2965061.

[13] T. Santos, D. Angelov, and M. Buiani, "Embrapa Wine Grape Instance Segmentation Dataset – Embrapa WGISD," 2019. Available at: https://github.com/thsant/wgisd.

[14] Y. Shi, D. Shi, Z. Qiao, Y. Zhang, Y. Liu, and S. Yang, "A Survey on Recent Advances in Few-Shot Object Detection," *Chinese Journal of Computers*, vol. 46, no. 8, pp. 1753–1780, Aug. 2023. DOI: 10.11897/SP.J.1016.2023.01753 (in Chinese with English abstract).

[15] J. W. Kim and M. Lee, "A Real-time Citrus Segmentation and Detection System using Mask R-CNN," *dcs*, vol. 19, no. 12, pp. 2385–2391, Dec. 2018, doi: 10.9728/dcs.2018.19.12.2385.

[16] R. P. Devanna, A. Milella, R. Marani, S. P. Garofalo, G. A. Vivaldi, S. Pascuzzi, R. Galati, and G. Reina, "In-Field Automatic Identification of Pomegranates Using a Farmer Robot," *Sensors*, vol. 22, no. 15, p. 5821, 2022. DOI: 10.3390/s22155821.

[17] T. A. Ciarfuglia, I. M. Motoi, L. Saraceni, M. Fawakherji, A. Sanfeliu, and D. Nardi, "Weakly and semi-supervised detection, segmentation and tracking of table grapes with limited and noisy data," *Comput. Electron. Agric.*, vol. 205, 2023, Art. no. 107624.

[18] M. Daiaeddine, S. Badrouss, A. El Harti, et al., "UAV imagery, advanced deep learning, and YOLOv7 object detection model in enhancing citrus yield estimation," Foods and Raw Materials, vol. 13, no. 2, pp. 242–253, 2025, doi: 10.21603/2308-4057-2025-2-650.

[19] X. Zhang and L. Li, "Research of image recognition of camellia oleifera fruit based on improved convolutional auto-encoder," journal of Forestry Engineering, vol. 4, no. 3, pp. 118–124, 2019, doi: 10.13360/j.issn.2096-1359.2019.03.018 (in Chinese with English abstract).

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2016, pp. 770–778.

[21] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, "Stacked convolutional auto-encoders for hierarchical feature extraction," in Proc. 21st Int. Conf. Artif. Neural Networks (ICANN), Espoo, Finland, June 2011, pp. 52–59.

[22] J. Ruan, H. Cui, Y. Huang, T. Li, C. Wu, and K. Zhang, "A review of occluded objects detection in real CS for autonomous driving," *Green Energy and Intelligent Transportation*, vol. 2, no. 3, pp. 65–77, 2023. DOI: 10.1016/j.geits.2023.100092

[23] O. M. Lawal, "YOLOMuskmelon: Quest for Fruit Detection Speed and Accuracy Using Deep Learning," IEEE Access, vol. 9, pp. 15221–15227, 2021, doi: 10.1109/ACCESS.2021.3053167.

[24] J. Sun, L. Qian, W. Zhu , X. Zhou, C. Dai, and X. Wu, "Apple detection in complex orchard environment based on improved RetinaNet," *Trans. Chinese Soc. Agric. Eng.*, vol. 38, no. 15, pp. 314–322, 2022. DOI: 10.11975/j.issn.1002-6819.2022.15.034 (in Chinese with English abstract).

[25] M. Sun, L. Xu, X. Chen, Z. Ji, Y. Zheng, and W. Jia, "BFP Net: Balanced Feature Pyramid Network for Small Apple Detection in Complex Orchard Environment," *Plant Phenomics*, vol. 2022, p. 2022/9892464, Jan. 2022, doi: 10.34133/2022/9892464.

[26] B. He, Y. Zhang, J. Gong, G. Fu, Y. Zhao, and R. Wu, "Fast Recognition of Tomato Fruit in Greenhouses at Night Based on Improved YOLO v5," *Transactions of the Chinese Society for Agricultural Machinery*, vol. 53, no. 5, pp. 201–208, 2022. DOI: 10.6041/j.issn.1000-1298.2022.05.020 (in Chinese with English abstract).

[27] A. Wang, W. Qian, A. Li, Y. Xu, J. Hu, Y. Xie, and L. Zhang, "NVW-YOLOv8s: An improved YOLOv8s network for real-time detection and segmentation of tomato fruits at different ripeness stages," *Computers and Electronics in Agriculture*, vol. 219, p. 108833, Apr. 2024, doi: 10.1016/j.compag.2024.108833.

[28] H. Zhao, Y. Qiao, H. Wang, and Y. Yue, "Apple fruit recognition in complex orchard environment based on improved YOLOv3," *Trans. Chinese Soc. Agric. Eng.*, vol. 37, no. 16, pp. 127–135, 2021. DOI: 10.11975/j.issn.1002-6819.2021.16.016 (in Chinese with English abstract).

[29] Y. Long, N. Li, Y. Gao, M. He, and H. Song, "Apple fruit detection under natural condition using improved FCOS network," *Trans. Chinese Soc. Agric. Eng.*, vol. 37, no. 12, pp. 307–313, 2021. DOI: 10.11975/j.issn.1002-6819.2021.12.035 (in Chinese with English abstract).

[30] Z. Zhang, J. Zhou, Z. Jiang, and H. Han, "Lightweight Apple Recognition Method in Natural Orchard Environment Based on Improved YOLO v7 Model," *Transactions of the Chinese Society for Agricultural Machinery*, vol. 55, no. 3, pp. 231–242, 262, Mar. 2024. DOI: 10.6041/j.issn.1000-1298.2024.03.023 (in Chinese with English abstract).

[31] S. Kaukab, K. Komal, B. M. Ghodki, H. Ray, Y. B. Kalnar, K. Narsaiah, and J. S. Brar, "Improving real-time apple fruit detection: Multi-modal data and depth fusion with non-targeted background removal," *Ecological Informatics*, vol. 82, p. 102691, Sep. 2024. DOI: 10.1016/j.ecoinf.2024.102691.

[32] H. Song, Y. Wang, Y. Wang, S. Lv, and M. Jiang, "Camellia oleifera Fruit Detection in Natural Scene Based on YOLO v5s," *Transactions of the Chinese Society for Agricultural Machinery*, vol. 53, no. 7, pp. 234–242, Jul. 2022. DOI: 10.6041/j.issn.1000-1298.2022.07.024 (in Chinese with English abstract).

[33] X. Zhu, F. Chen, Y. Zheng, X. Peng, and C. Chen, "An efficient method for detecting Camellia oleifera fruit under complex orchard environment," *Scientia Horticulturae*, vol. 330, p. 113091, Apr. 2024, doi: 10.1016/j.scienta.2024.113091.

[34] J. Zhou, W. Hu, A. Zou, S. Zhai, T. Liu, W. Yang, and P. Jiang, "Lightweight Detection Algorithm of Kiwifruit Based on Improved YOLOX-S," *Agriculture*, vol. 12, no. 7, p. 993, Jul. 2022. DOI: 10.3390/agriculture12070993.

[35] L. Mao, Z. Guo, M. Liu, et al., "A scalable multi-modal learning fruit detection algorithm for dynamic environments," Frontiers in Neurorobotics, vol. 18, 2025, doi: 10.3389/fnbot.2024.1518878.

[36] J. Xie, J. Peng, J. Wang, B. Chen, T. Jing, D. Sun, P. Gao, W. Wang, J. Lu, R. Yetan, et al., "Litchi Detection in a Complex Natural Environment Using the YOLOv5-Litchi Model," *Agronomy*, vol. 12, no. 12, p. 3054, Dec. 2022, doi: 10.3390/agronomy12123054.

[37] Z. Xiong, L. Wang, Y. Zhao, and Y. Lan, "Precision Detection of Dense Litchi Fruit in UAV Images Based on Improved YOLOv5 Model," *Remote Sensing*, vol. 15, no. 16, p. 4017, Aug. 2023, doi: 10.3390/rs15164017.

[38] X. Liu, G. Li, W. Chen, B. Liu, M. Chen, and S. Lu, "Detection of Dense Citrus Fruits by Combining Coordinated Attention and Cross-Scale Connection with Weighted Feature Fusion," *Applied Sciences*, vol. 12, no. 13, p. 6600, Jun. 2022, doi: 10.3390/app12136600.

[39] H. Sun, B. Wang, and J. Xue, "YOLO-P: An efficient method for pear fast detection in complex orchard picking environment," *Front. Plant Sci.*, vol. 13, p. 1089454, Jan. 2023, doi: 10.3389/fpls.2022.1089454.

[40] P. Liu and H. Yin, "YOLOv7-Peach: An Algorithm for Immature Small Yellow Peaches Detection in Complex Natural Environments," *Sensors*, vol. 23, no. 11, p. 5096, May 2023, doi: 10.3390/s23115096.

[41] W. Du, Y. Zhu, S. Li, and P. Liu, "Spikelets detection of table grape before thinning based on improved YOLOV5s and Kmeans under the complex environment," *Computers and Electronics in Agriculture*, vol. 203, p. 107432, Dec. 2022, doi: 10.1016/j.compag.2022.107432.

[42] C. Guo, S. Zheng, G. Cheng, Y. Zhang, and J. Ding, "An improved YOLO v4 used for grape detection in unstructured environment," *Front. Plant Sci.*, vol. 14, p. 1209910, Jul. 2023, doi: 10.3389/fpls.2023.1209910.

[43] X. Shi, X. Zhang, Y. Su, et al., "Positive Anchor Area Merge Algorithm: A Knowledge Distillation Algorithm for Fruit Detection Tasks Based on Yolov8," IEEE Access, vol. 13, pp. 34954–34968, 2025, doi: 10.1109/ACCESS.2025.3544361.

[44] Z. He, M. Karkee, and Q. Zhang, "Enhanced machine vision system for field-based detection of pickable strawberries: Integrating an advanced two-step deep learning model merging improved YOLOv8 and YOLOv5-cls," Computers and Electronics in Agriculture, vol. 234, 2025, doi: 10.1016/j.compag.2025.110173.

[45] S. Lü, S. Lu, Z. Li, T. Hong, Y. Xue, and B. Wu, "Orange recognition method using improved YOLOv3-LITE lightweight neural network," *Trans. Chinese Soc. Agric. Eng.*, vol. 35, no. 17, pp. 205–214, 2019. DOI: 10.11975/j.issn.1002-6819.2019.17.025 (in Chinese with English abstract).

[46] J. Huang, X. Zhao, F. Gao, X. Wen, S. Jin, and Y. Zhang, "Recognizing and detecting the strawberry at multi-stages using improved lightweight YOLOv5s," *Trans. Chinese Soc. Agric. Eng.*, vol. 39, no. 21, pp. 181–187, 2023. DOI: 10.11975/j.issn.1002-6819.202307186 (in Chinese with English abstract).

[47] C. Guo, C. Zhu, Y. Liu, R. Huang, B. Cao, Q. Zhu, R. Zhang, and B. Zhang, "End-to-End lightweight Transformer-Based neural network for grasp detection towards fruit robotic handling," *Comput. Electron. Agric.*, vol. 221, p. 109014, Jun. 2024. DOI: 10.1016/j.compag.2024.109014.

[48] H. Li, C. Li, G. Li, and L. Chen, "A real-time table grape detection method based on improved YOLOv4-tiny network in complex background," *Biosystems Engineering*, vol. 212, pp. 347–359, Dec. 2021, doi: 10.1016/j.biosystemseng.2021.11.011.

[49] H. Wang, J. Feng, and H. Yin, "Improved Method for Apple Fruit Target Detection Based on YOLOv5s," *Agriculture*, vol. 13, no. 11, p. 2167, Nov. 2023, doi: 10.3390/agriculture13112167.

[50] Y. Xu, M. Jiang, Y. Li, Y. Wu, and G. Lu, "Fruit target detection based on improved YOLO and NMS," *J. Electron. Meas. Instrumentation*, vol. 36, no. 4, pp. 114–123, 2022. DOI: 10.13382/j.jemi.B2104724 (in Chinese with English abstract).

[51] R. Gai, M. Li, Z. Wang, L. Hu, and X. Li, "YOLOv5s-Cherry: Cherry Target Detection in Dense Scenes Based on Improved YOLOv5s Algorithm," *J CIRCUIT SYST COMP*, vol. 32, no. 12, p. 2350206, Aug. 2023, doi: 10.1142/S0218126623502067.

[52] Y. Zhang, L. Zhang, H. Yu, Z. Guo, R. Zhang, and X. Zhou, "Research on the Strawberry Recognition Algorithm Based on Deep Learning," *Applied Sciences*, vol. 13, no. 20, p. 11298, Oct. 2023, doi: 10.3390/app132011298.

[53] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2018, pp. 4510–4520.

[54] A. Bochkovskiy, C. Y. Wang, and H. Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," arXiv preprint arXiv:2004.10934, 2020.

[55] G. Cheng, X. Yuan, X. Yao, K. Yan, Q. Zeng, X. Xie, and J. Han, "Towards Large-Scale Small Object Detection: Survey and Benchmarks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 11, pp. 13467–13488, Nov. 2023. DOI: 10.1109/TPAMI.2023.3290594.

[56] L. Ma, L. Zhao, Z. Wang, J. Zhang, and G. Chen, "Detection and Counting of Small Target Apples under Complicated Environments by Using Improved YOLOv7-tiny," *Agronomy*, vol. 13, no. 5, p. 1419, May 2023, doi: 10.3390/agronomy13051419.

[57] A. Zhu, R. Zhang, L. Zhang, T. Yi, L. Wang, D. Zhang, L. Chen, "YOLOv5s-CEDB: A robust and efficiency Camellia oleifera fruit detection algorithm in complex natural scenes," *Computers and Electronics in Agriculture*, vol. 221, p. 108984, Jun. 2024, doi: 10.1016/j.compag.2024.108984.

[58] Y. Shi, J. Li, P. Zhang, and D. Wang, "Detecting and counting of spring-see citrus using YOLOv4 network model and recursive fusion of features," *Trans. Chinese Soc. Agric. Eng.*, vol. 37, no. 18, pp. 161–169, 2021. DOI: 10.11975/j.issn.1002-6819.2021.18.019 (in Chinese with English abstract).

[59] J. Xiong, Z. Huo, Q. Huang, H. Chen, Z. Yang, Y. Huang, and Y. Su, "Detection method of citrus in nighttime environment combined with active light source and improved YOLOv5s model," *Journal of South China Agricultural University*, vol. 45, no. 1, pp. 97–107, 2024. DOI: 10.7671/j.issn.1001-411X.202209010 (in Chinese with English abstract).

[60] C. Yang, J. Liu, and J. He, "A lightweight waxberry fruit detection model based on YOLOv5," *IET Image Processing*, vol. 18, no. 7, pp. 1796–1808, May 2024, doi: 10.1049/ipr2.13064.

[61] Z. He, S. R. Khanal, X. Zhang, M. Karkee, and Q. Zhang, "Real-time Strawberry Detection Based on Improved YOLOv5s Architecture for Robotic Harvesting in open-field environment," Oct. 12, 2023, *arXiv*: arXiv:2308.03998. Accessed: Sep. 18, 2024. [Online]. Available: http://arxiv.org/abs/2308.03998

[62] C. Zhang, F. Kang, and Y. Wang, "An Improved Apple Object Detection Method Based on Lightweight YOLOv4 in Complex Backgrounds," *Remote Sensing*, vol. 14, no. 17, p. 4150, Aug. 2022, doi: 10.3390/rs14174150.

[63] J. Sun, Y. Chen, X. Zhou, J. Shen, and X. Wu, "Fast and accurate recognition of the strawberries in greenhouse based on improved YOLOv4-Tiny model," *Transactions of the Chinese Society of Agricultural Engineering*, vol. 38, no. 18, pp. 195–203, 2022. DOI: 10.11975/j.issn.1002-6819.2022.18.021 (in Chinese with English abstract).

[64] S. Lyu, R. Li, Y. Zhao, Z. Li, R. Fan, and S. Liu, "Green Citrus Detection and Counting in Orchards Based on YOLOv5-CS and AI Edge System".

[65] J. Yang, Z. Qian, Y. Zhang, Y. Qin, and H. Miao, "Real-time recognition of tomatoes in complex environments based on improved YOLOv4-tiny," *Trans. Chinese Soc. Agric. Eng.*, vol. 38, no. 9, pp. 215–221, 2022. DOI: 10.11975/j.issn.1002-6819.2022.09.023 (in Chinese with English abstract).

[66] Q. Luo, C. Wu, G. Wu, and W. Li, "A Small Target Strawberry Recognition Method Based on Improved YOLOv8n Model," *IEEE Access*, vol. 12, pp. 14987–14995, 2024, doi: 10.1109/ACCESS.2024.3356869.

[67] Y. Bai, J. Yu, S. Yang, and J. Ning, "An improved YOLO algorithm for detecting flowers and fruits on strawberry seedlings," *Biosystems Engineering*, vol. 237, pp. 1–12, Jan. 2024, doi: 10.1016/j.biosystemseng.2023.11.008.

[68] H. Song, M. Jiang, Y. Wang, and L. Song, "Efficient detection method for young apples based on the fusion of convolutional neural network and visual attention mechanism," Transactions of the Chinese Society

of Agricultural Engineering, vol. 37, no. 9, pp. 297–303, 2021. DOI: 10.11975/j.issn.1002-6819.2021.09.034 (in Chinese with English abstract).
[69] W. Yang, X. Ma, W. Hu, and Pengjie Tang, “Lightweight Blueberry Fruit Recognition Based on Multi-Scale and Attention Fusion NCBAM,” *Agronomy*, vol. 12, no. 10, p. 2354, Sep. 2022, doi: 10.3390/agronomy12102354.
[70] C. Li, J. Lin, Z. Li, C. Mai, R. Jiang, and J. Li, “An efficient detection method for litchi fruits in a natural environment based on improved YOLOv7-Litchi,” *Computers and Electronics in Agriculture*, vol. 217, p. 108605, Feb. 2024, doi: 10.1016/j.compag.2023.108605.
[71] P. Zhou, Y. Pei, R. Wei, Y. Zhang, and Y. Gu, "Real-time detection of orchard cherry based on YOLOV4 model," *Acta Agriculturae Zhejiangensis*, vol. 34, no. 11, pp. 2522–2532, 2022. DOI: 10.3969/j.issn.1004-1524.2022.11.021.
[72] F. Meng, J. Li, Y. Zhang, S. Qi, and Y. Tang, “Transforming unmanned pineapple picking with spatio-temporal convolutional neural networks,” *Computers and Electronics in Agriculture*, vol. 214, p. 108298, Nov. 2023, doi: 10.1016/j.compag.2023.108298.
[73] H. Liu, X. Wang, F. Zhao, F. Yu, P. Lin, Y. Gan, X. Ren, Y. Chen, and J. Tu, "Upgrading Swin-B Transformer-based model for accurately identifying ripe strawberries by coupling task-aligned one-stage object detection mechanism," *Comput. Electron. Agric.*, vol. 218, p. 108674, Mar. 2024. DOI: 10.1016/j.compag.2024.108674.
[74] C. Liang, J. Liang, W. Yang, et al., “Enhanced visual detection of litchi fruit in complex natural environments based on unmanned aerial vehicle (UAV) remote sensing,” Precision Agriculture, vol. 26, no. 1, 2025, doi: 10.1007/s11119-025-10220-w.
[75] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, "GhostNet: More features from cheap operations," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), 2020, pp. 1580–1589.
[76] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., ... & Guo, B. (2021). Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 10012-10022).
[77] Z. Cao, L. Kooistra, W. Wang, L. Guo, and J. Valente, "Real-Time Object Detection Based on UAV Remote Sensing: A Systematic Literature Review," Drones, vol. 7, no. 10, p. 620, Oct. 2023. DOI: 10.3390/drones7100620.
[78] H. Huang, T. Huang, Z. Li, S. Lyu, and T. Hong, “Design of Citrus Fruit Detection System Based on Mobile Platform and Edge Computer Device,” *Sensors*, vol. 22, no. 1, p. 59, Dec. 2021, doi: 10.3390/s22010059.
[79] G. Hu, J. Zhou, C. Chen, C. Li, L. Sun, Y. Chen, S. Zhang, and J. Chen, "Fusion of the lightweight network and visual attention mechanism to detect apples in orchard environment," *Trans. Chinese Soc. Agric. Eng.*, vol. 38, no. 19, pp. 131–142, 2022. DOI: 10.11975/j.issn.1002-6819.2022.19.015 (in Chinese with English abstract).

XINYU GONG was born in Guangzhou, Guangdong Province, China in 2004. She is currently pursuing the bachelor's degree in Data Science and Big Data at the College of Arts and Sciences, Northeast Agricultural University. Her research interests include computer vision and deep learning.

[80] B. Zhao, A. Guo, R. Ma, Y. Zhang, and J. Gong, "YOLOv8s-CFB: A lightweight method for real-time detection of apple fruits in complex environments," *J. Real-Time Image Process.*, vol. 21, no. 5, p. 164, 2024.
[81] C. Yu, J. Feng, Z. Zheng, J. Guo, and Y. Hu, “A lightweight SOD-YOLOv5n model-based winter jujube detection and counting method deployed on Android,” *Computers and Electronics in Agriculture*, vol. 218, p. 108701, Mar. 2024, doi: 10.1016/j.compag.2024.108701.
[82] L. Wang, H. Zheng, C. Yin, Y. Wang, Z. Bai, and W. Fu, “Dense Papaya Target Detection in Natural Environment Based on Improved YOLOv5s,” *Agronomy*, vol. 13, no. 8, p. 2019, Jul. 2023, doi: 10.3390/agronomy13082019.
[83] J. Zhao, X. Yao, Y. Wang, Z. Yi, Y. Xie, and X. Zhou, “Lightweight-Improved YOLOv5s Model for Grape Fruit and Stem Recognition,” *Agriculture*, vol. 14, no. 5, p. 774, May 2024, doi: 10.3390/agriculture14050774.
[84] F. Zhang, Z. Chen, R. Bao, Z. Zhang, and Z. Wang, "Recognition of dense cherry tomatoes based on improved YOLOv4-LITE lightweight neural network," *Trans. Chinese Soc. Agric. Eng.*, vol. 37, no. 16, pp. 270–278, 2021. DOI: 10.11975/j.issn.1002-6819.2021.16.033 (in Chinese with English abstract).
[85] J. Liu, Y. Li, L. Xiao, W. Li, and H. Li, "Recognition and location method of orange based on improved YOLOv4 model," *Trans. Chinese Soc. Agric. Eng.*, vol. 38, no. 12, pp. 173–182, 2022. DOI: 10.11975/j.issn.1002-6819.2022.12.020 (in Chinese with English abstract).
[86] J. Wang, Y. Su, J. Yao, M. Liu, Y. Du, X. Wu, L. Huang, and M. Zhao, “Apple rapid recognition and processing method based on an improved version of YOLOv5,” *Ecological Informatics*, vol. 77, p. 102196, Nov. 2023, doi: 10.1016/j.ecoinf.2023.102196.
[87] J. Chen, A. Ma, L. Huang, Y. Su, W. Li, H. Zhang, and Z. Wang, “GA-YOLO: A Lightweight YOLO Model for Dense and Occluded Grape Target Detection,” *Horticulturae*, vol. 9, no. 4, p. 443, Mar. 2023, doi: 10.3390/horticulturae9040443.
[88] F. Zhang, W. Cao, S. Wang, X. Cui, N. Yang, X. Wang, X. Zhang, and S. Fu, “Improved YOLOv4 recognition algorithm for pitaya based on coordinate attention and combinational convolution,” *Front. Plant Sci.*, vol. 13, p. 1030021, Oct. 2022, doi: 10.3389/fpls.2022.1030021.
[89] J. Feng, C. Yu, X. Shi, Z. Zheng, L. Yang, and Y. Hu, “Research on Winter Jujube Object Detection Based on Optimized Yolov5s,” *Agronomy*, vol. 13, no. 3, p. 810, Mar. 2023, doi: 10.3390/agronomy13030810.
[90] A. Howard, M. Sandler, G. Chu, L. C. Chen, B. Chen, M. Tan, et al., "Searching for MobileNetV3," in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), 2019, pp. 1314–1324.
[91] N. Ma, X. Zhang, H. T. Zheng, and J. Sun, "ShuffleNet V2: Practical guidelines for efficient CNN architecture design," in Proc. Eur. Conf. Comput. Vis. (ECCV), 2018, pp. 116–131.

Qiufeng Wu received B. S degree in Mathematics and Applied Mathematics from Harbin Normal University in 2002. He received M.S. degree in Management Science and Engineering from Northeast Agricultural University in 2007. He received Ph.D in computer application technology from Harbin Institute of Technology, China, in 2014. He is working as associated professor in College of Arts and Sciences in Northeast Agricultural University. His research interests include machine learning, computer vision and smart agriculture.