

Article

Not peer-reviewed version

---

# Improved YOLOv8 Algorithms for Agricultural Monitoring and Harvesting Tasks: A Comprehensive Review

---

[Mohieddine Jelali](#)\*

Posted Date: 25 November 2025

doi: 10.20944/preprints202511.1904.v1

Keywords: deep learning; convolutional neural networks; object detection in plants; Improved YOLOv8; tiny target detection layer; attention mechanisms



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a [Creative Commons CC BY 4.0 license](#), which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Review

# Improved YOLOv8 Algorithms for Agricultural Monitoring and Harvesting Tasks: A Comprehensive Review

Mohieddine Jelali

Cologne Laboratory of Artificial Intelligence and Smart Automation (CAISA), Institute of Product Development and Engineering Design (IPK), Technische Hochschule Köln—University of Applied Sciences, Betzdorfer Str. 2, 50679 Cologne, Germany; mohieddine.jelali@th-koeln.de

## Abstract

Accuracy and real-time performance are two major challenges in monitoring plant growth, detecting crops, and recognizing diseases in complex real-world agricultural environments. In such environments, object detection faces a series of challenges, including variable weather and lighting conditions, variable shooting distances, varying degrees of occlusion, and diverse morphological characteristics. The YOLO series of models, especially the prominent YOLOv8, are state-of-the-art models for object recognition that have revolutionized the field by achieving an optimal balance between speed and accuracy. Since YOLOv8 appeared two and a half years ago, many improvement measures or modifications have been proposed in the literature for different detection tasks and applications. This paper systematically reviews these Improved YOLOv8 algorithms, focusing on object detection in plants (e.g., crops, diseases, and growth stages), to evaluate the proposed changes or improvements. We surveyed over 376 papers, most of which were published between 2023 and 2025. Of those, 196 were directly relevant to the topic. Inspired by the reviewed architectures, our analysis provides recommendations and potential improvements for each YOLOv8 component that could be replaced or enhanced. Finally, the paper presents and evaluates selected Improved YOLOv8 algorithms to demonstrate their composition and complexity as prime examples of modular YOLOv8 architectures.

**Keywords:** deep learning; convolutional neural networks; object detection in plants; Improved YOLOv8; tiny target detection layer; attention mechanisms

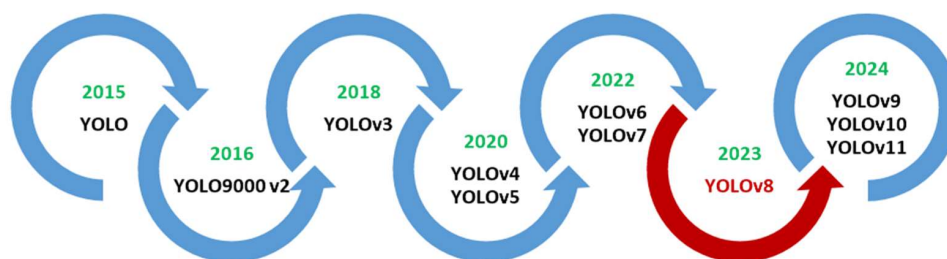
## 1. Introduction

Agriculture is widely regarded as one of the most critical sectors, playing a pivotal role in ensuring food security. However, as the global population continues to expand, so does the demand for food, thereby creating the need to transition from conventional farming methods to smart farming practices, also known as Agriculture 4.0 [1]. Agricultural monitoring and harvesting tasks play a critical role in modern precision agriculture to ensure optimal crop yield and quality. Key tasks such as crop maturity detection, disease or pest detection, and growth stage monitoring use advanced technologies such as computer vision, remote sensing, and machine learning (ML). Ripeness detection helps farmers determine the optimal time to harvest by analyzing the color, texture, and chemical properties of fruits and crops. Disease and pest detection uses image processing and AI models to detect early signs of plant diseases, enabling timely intervention and reducing crop losses. Growth stage monitoring tracks plant development using satellite imagery, drones, or IoT sensors, providing valuable insights into nutrient requirements and overall health. These automated techniques increase efficiency, reduce manual labor, and support sustainable agricultural practices.

Deep learning (DL), a subset of artificial intelligence (AI), has emerged as a powerful tool for image analysis and pattern recognition. Convolutional neural networks (CNNs), a prevalent deep learning (DL) architecture, have exhibited substantial success in a variety of image classification tasks,

including object detection, fruit counting, automated harvesting, and, notably, plant disease detection and diagnosis. In the realm of machine learning (ML) and deep learning (DL) algorithms, convolutional neural networks (CNNs) are frequently the preferred option for image detection and classification tasks. This preference can be attributed to the inherent capability of CNNs to autonomously extract relevant image features and to understand spatial hierarchies [2,3].

Two categories of object detection algorithms are distinguished: two-stage algorithms and one-stage algorithms. Two-stage algorithms, including ResNet, LeNet-5, AlexNet, GoogLeNet, and Faster R-CNN, delineate a two-step process. Conversely, one-stage algorithms, such as SSD, VGG, and YOLO, execute a single step. It has been demonstrated that two-step algorithms demonstrate relatively high accuracy. However, these algorithms are constrained by computational demands and real-time performance [3,4]. The advent of YOLO signified a substantial progression in the domain of object recognition, thereby fundamentally altering the methodologies employed in recognition tasks. The primary innovation of YOLO is the conceptualization of the object detection task as a single regression problem. In the solution approach, the prediction of bounding boxes and class probabilities occurs in a unified process [5]. Figure 1 illustrates the evolution of the YOLO family.



**Figure 1.** Historical timeline of the YOLO series.

Despite significant advances in object (weed, flower, crop, maturity, disease or pest) detection technology for plant/crop monitoring, diagnostics and harvesting, several formidable challenges remain to be overcome to realize its full potential. A significant challenge in this field pertains to occlusion, wherein objects become partially hidden or obscured, thereby complicating their accurate detection. Another challenge is the management of object scales, wherein objects manifest at varying stages or sizes due to alterations in type (leaf, flower, crop), shape, color, perspective, or distance, thereby impeding the processes of detection or classification. The need for real-time processing adds another layer of complexity, especially in real-world field environments that are diverse and unstructured, where lighting conditions, backgrounds, and object types can vary widely [5].

The big popularity and widespread application of YOLOv8 to a large number of fields have led to many advancements of its architecture and algorithms. A considerable number of researchers have continued to refine and extend the YOLOv8 framework, applying it to various tasks, including plant identification, disease detection and classification, and crop detection and maturity monitoring studies. These efforts have sought to attain consistent and reliable results in diverse and unstructured environments with varying lighting conditions, backgrounds, object types, scales, and growth stages. We identified over 200 enhanced versions of YOLOv8, though our analysis is constrained to case studies in the domain of agricultural object detection. However, a systematic review of these advancements is conspicuously absent from the extant literature. The majority of reviews offer a more expansive perspective on the evolution of YOLO, encompassing a range of primary versions and offering comparative analyses; for instance, YOLOv5–YOLOv10 in [6] and YOLOv8–YOLOv11 in [7]. Other surveys, such as those in [8,9], concern the macro framework of DL-based object detection.

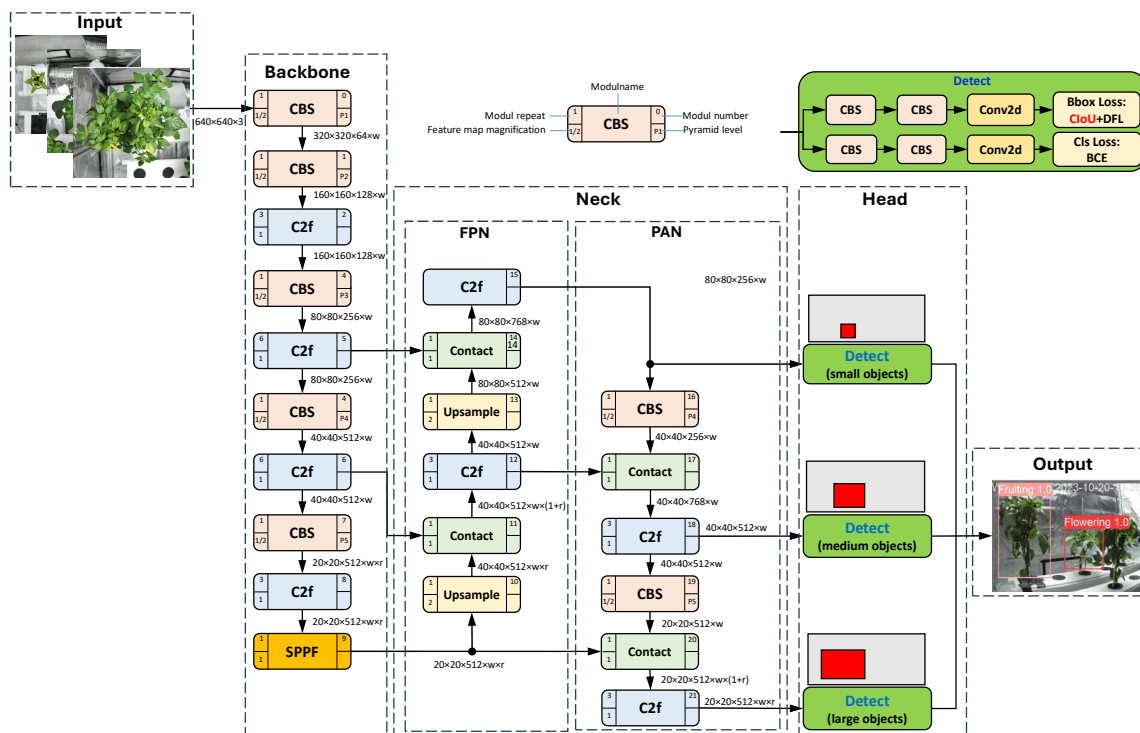
A notable exception is the work in [10], which approaches the subject in a somewhat analogous manner to our paper. However, the scope of Badgujar et al.'s work is considerably more limited, encompassing only a concise overview of the YOLO model development and modification.

The present work aims to provide a comprehensive, detailed, and systematic overview of Improved YOLOv8 algorithms published in the last two and a half years, since the appearance of YOLOv8,

focusing particularly on agricultural monitoring and harvesting tasks. This overview is presented in the form of tables, figures, and statistical analysis (pie diagrams). This paper can serve as a starting point for researchers and engineers seeking to expand their knowledge in this field, apply the reviewed algorithms, or develop further enhanced approaches. The main contributions of our work can be summarized as follows:

1. We present the first systematic review of its kind in this area, analyzing 196 selected research papers on the topic.
2. We assign the improvement measures, modifications, or extensions to the individual sections of YOLOv8 (Figure 2) (backbone, neck, head) and components (CBS, C2f, AM—Attention Mechanism, SPPF, Concat, Upsample, CIoU).
3. We provide recommendations and highlight potential improvements for each component of YOLOv8 that could be replaced or enhanced.
4. We recommend the most promising improvements and extensions of YOLOv8 for further development and application of specific detection methods, and outline future research directions in the field.

The present review focuses exclusively on the further development and improvement/modification of the YOLOv8 model. This paper does not cover topics like data acquisition, data processing, or YOLO integration with other algorithms for specific applications.



**Figure 2.** YOLOv8 network architecture (based on [11,12]). Legend: CBS—Convolution + Batch Normalisation + Sigmoid-weighted Linear Unit (SiLU); C2f—CSP (Cross Stage Partial) bottleneck with 2 convolutions, Fast version; SPPF—Spatial Pyramid Pooling, Fast version; Upsample—upsampling; w—Width multiple; r—Ratio; Concat—Concatenation operation; Detect: Detector; FPN—Feature Pyramid Network; PAN—Path Aggregation Network; Conv2d—2D Convolution; BBox Loss—Bounding Box Loss, IoU—Intersection over Union, CIoU—Complete IoU; DFL—Distributed Focal Loss; Cls Loss—Classification Loss, BCE—Binary Cross Entropy

## 2. Materials and Methods

This review was (partly) conducted following the guidelines of the Preferred Reporting Items for Systematic Reviews [13].

### 2.1. Review Question

The research question guiding this review is as follows: *Which module substitutions or extensions in the YOLOv8 architecture are revealed by the literature to improve detection performance in agricultural monitoring and harvesting tasks? What are the best module combinations for improving performance? What open issues warrant further investigation?*

### 2.2. Eligibility Criteria

We considered research articles, conference papers, and arXiv papers that explicitly propose and investigate *Improved YOLOv8* architectures, with case studies related to agricultural monitoring and harvesting tasks. Only works published in peer-reviewed journals or presented at reputable conferences were included. Additionally, the studies had to clearly describe their methodologies and applications and include an ablation study. They had to focus on at least one of the following five categories:

- Detection of plant diseases and pests.
- Detection of fruits/crops, their maturity, or picking points for harvesting.
- Detection of plant growth stages.
- Weed detection.
- Plant phenotyping.

The timeline for inclusion spanned publications from January 2023, the year YOLOv8 was released, to October 2025. This reflects YOLOv8's rapid evolution in recent years.

### 2.3. Exclusion Criteria

Studies were excluded if they were not directly related to one of the considered application fields. General discussions on AI and ML/DL, theoretical and basic works without practical implications, and duplicate publications were omitted. Additionally, non-peer-reviewed content, such as blog posts and opinion articles, as well as works lacking sufficient details on methodology, metric-based evaluations, and an ablation study, were excluded. Papers written in languages other than English were excluded.

### 2.4. Search Strategy

A simple search strategy was employed to identify relevant literature. Using a combination of keywords and Boolean operators ("AND", "OR"), we searched multiple academic databases, including Google Scholar, IEEE Xplore, and ScienceDirect. The keywords used were "Improved YOLOv8," "Disease/Pest Detection," "Pest Detection," "Fruit/Crop Detection," and "Maturity/Picking Point Detection," among others. We manually screened the reference lists of the included studies to identify additional relevant works and did not use tools such as Mendeley for deduplication and systematic screening.

### 2.5. Data Extraction and Data Synthesis

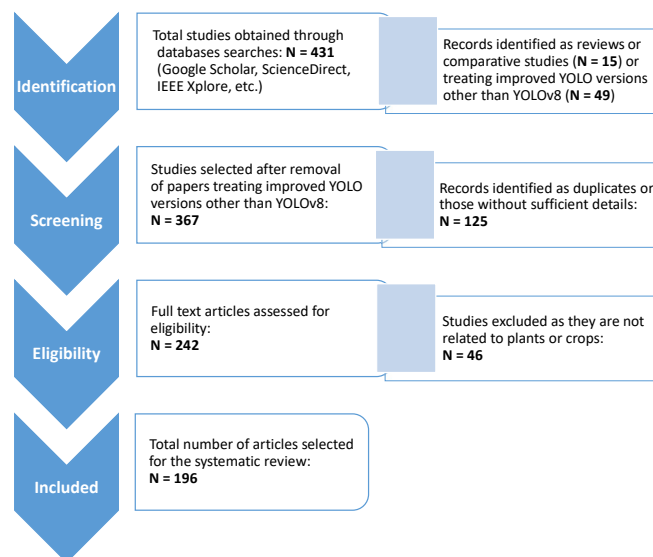
We used a thematic synthesis approach to organize the data into predefined YOLOv8 module categories (CBS, C2f, AM, SPPF, Concat, Upsample, and Ciou) and sections (Backbone, Neck, and Head), which are aligned with the study's focus areas, as defined in Section 2.2. The synthesis aimed to identify potential module combinations that lead to performance improvement or lightweighting. The visual representation in Table A1 was generated to summarize the findings and facilitate comparative analysis across studies.

## 3. Results

### 3.1. Selection of Sources

Our review involved analyzing the titles and abstracts of studies to identify matches with the searched keywords. A total of 431 records were initially identified based on these keywords. After eliminating duplicates and excluding overview articles or articles with unrelated content, the remaining

full-text studies were selected, screened, and evaluated for inclusion in the systematic review. The flow chart of PRISMA method is shown in Figure 3. After the first and second rounds of evaluation, we excluded those deemed irrelevant to the topic, non-English papers, and papers without sufficient details, selecting 196 articles for our survey.



**Figure 3.** PRISMA flow chart illustrating the selection process of the 196 studies included in the systematic review.

### 3.2. Synthesis of Results

The papers included in our review are listed in Table A1, summarizing the replacements or extensions introduced in the Improved YOLOv8 algorithms compared to the baseline YOLOv8. The improvement measures will be presented and discussed in Sections 4–sec:Head Improvements. The included studies are categorized to diverse agricultural monitoring and harvesting tasks, including detection of plant diseases and pests (Table 1), detection of fruit/crop maturity, pose, or picking point for harvesting (Table 2), detection of plant growth stages (Table 3), weed detection (Table 4), and plant phenotyping (Table 5).

**Table 1.** Overview of Improved YOLOv8-based algorithms/networks for the detection of plant diseases and pests.

Plant species	YOLO Algorithm [Ref.]
Rice	YOLOv8-Rice [14], Pyramid-YOLOv8 [15], Improved YOLOv8 [16], YOLOv8-AMD [17], RicePest-YOLO [18], TLI-YOLO [19], MFAC-YOLOv8 [20], RGC-YOLO [21], GA-YOLOv8 [22], YOLOv8-OW [23], YOLO-RMD [24], GhostConv+CA-YOLOv8n [25]
Tomato	YOLOv8s-Seg [26], BHC-YOLOv8 [27], LAMP-LYOLOv8n [28], YOLO-TGI [29], Improved YOLOv8 [30], YOLO-BSMamba [31], LT-YOLO [32], E-TomatoDet [33], CAPNet [34], YOLOv8n-CDSA-BiFPN [35], TomaFDNet [36]
Tea	YOLOv8-RCAA [37], YOLOv8-RMDA [38], YOLOv8-ASFF [39], YOLOv8n-WSE-Pest [40], SAHI-YOLOv8 [41], YOLOv8-TD [42], TLDDM [43]
Apple	YOLOv8-Leaf [44], YOLOv8n-GGi [45], Improved YOLOv8 [46], FSM-YOLO [47], YOLOv8-GSSW [48], YOLO-ACT [49]
Strawberry	KTD-YOLOv8 [50], Improved YOLOv8 [51], YOLO-Berry [52], YOLOv8s-Ghost-Trans [53], D-YOLO [54]
Maize/Corn	Maize-Rust model [55], Improved YOLOv8 [56], AMS-YOLO [57], YOLO-MSM [58]
Citrus	Light-SA YOLOv8 [59], YOLOv8-GABNet [60], Improved YOLOv8 [61]
Cucumber	DM-YOLOv8 [62], SEDCN-YOLOv8 [63]
Cotton	YOLO-UP [64], LCDDN-YOLO [65]
Wheat	YOLO-Wheat [66]
Grape	YOLOv8-ACCW [67]
Cauliflower	Cauli-Det [68]
Lemon	LSD-YOLO [69]
Sesame	Improved YOLOv8n [70]
Tobacco	Improved YOLOv8l [71]
Forest	Improved YOLOv8 [72]
Soybean	YOLOv8-DML [73]
Runner bean	YOLOv8-RBean [74]
Banana	BFWS [75]
Onion	YOLO-ODD [76]
Pepper	MSPB-YOLO [77]
Sugarcane	ADQ-YOLOv8m [78]
Multiple species	GLU-YOLOv8 [79], YOLOv8n-vegetable, [80], Improved YOLOv8 [81], Improved YOLOv8n/s [82], VegetableDet [83], SerpensGate-YOLOv8 [84], PYOLO [85], LWE-YOLOv8 [86], BGM-YOLOv8 [87]

**Table 2.** Overview of improved YOLOv8-based algorithms/networks for the detection of fruit/crop maturity, pose, or picking point for harvesting.

Plant species	YOLO Algorithm [Ref.]
Tomato	Improved YOLOv8s [88], MHSA-YOLOv8 [89], YOLOv8-Tomato [90], YOLOv8-EA [91], FastMLCA-YOLOv8 [92], S-YOLO [93], MTS-YOLO [94], NVW-YOLOv8s [95], RSR-YOLO [96], YOLOv8+ [97], YOLOv8-MR [98], YOLO-PP [99], GCSS-YOLO [100], YOLOv8-LBP [101]
Strawberry	Improved YOLOv8-Pose [102], CES-YOLOv8 [103], Improved YOLOv8n [104], LS-YOLOv8s [105], Two-Step-YOLO [106], RLK-YOLOv8 [107], STRAW-YOLO [108], GAM-YOLOv8s [109]
Apple	Faster-YOLO-AP [110], YOLOv8n-ShuffleNetv2-Ghost-SE [111], DNE-YOLO [112], SGW-YOLOv8n [113], Improved YOLOv8 [114], YOLOv8s-FCB [115], Improved YOLOv8n [116], Enhanced-YOLOv8+ [117]
Citrus	Improved YOLOv8s [118], YCCB-YOLO [119], AG-YOLO [120], YOLOv8-Scm [121], Improved YOLOv8n [122], Improved YOLOv8n-seg [123]
Pepper	Pepper-YOLO [124], YOLOv8n (Chili) [125], YOLOv8-BiFPN-EMSCP [126], PAE-YOLO [127], YOLOv8-CBSE [128]
Peach	Improved YOLOv8 [129], YOLO-PEM [130], PeachYOLO [131], EMA-YOLO [132], SYL-YOLOv8n [133]
Tea	Tea-YOLOv8s [134], Optimized YOLOv8 [135], Improved YOLOv8 [136], T-YOLO [137], RFA-YOLOv8 [138]
Mango	Improved YOLOv8 [139], Improved YOLOv8-seg [140]
Grape	TiGra-YOLOv8 [141], YOLOv8-GP [142]
Safflower	YOLO-SaFi [143], WED-YOLO [144]
Wolfberry	FEW-YOLO [145]
Persimmon	PerD-YOLOv8 [146]
Orange	Improved YOLOv8 [147]
Tobacco	Improved YOLOv8-seg [148]
Banana	YOLO-BRFB [149]
Wheat	YOLOv8-FDA [150]

**Table 3.** Overview of improved YOLOv8-based algorithms/networks for the detection of plant growth stages.

Plant species	YOLO Algorithm [Ref.]
Maize/Corn	Improved YOLOv8 [151], ST-YOLOv8s [152], PConv-YoloV8×6 [153], CSGD-YOLO [154]
Melon	YOLOv8-CML [155], Improved YOLOv8 [156]
Cabbage	YOLOv8-Improved [157], YOLOv8-cabbage [158]
Blueberry	TL-YOLOv8 [159]
Nectarine	YOLOv8n-CSD [160]
Plum	MAE-YOLOv8 [161]
Coffee	YOLOv8n-RFCACConv [162]
Lotus	YOLOv8-seg [163]
Litchi	YOLOv8-FPDW [164]
Walnut	OW-YOLO [165]
Coconut	YOLOv8n-obb [166]
Cabbage	YOLOv8n-Cabbage [167]
Pomegranate	PG-YOLO [168]
Palm	Improved Yolov8 [169]
Soybean	YOLOv8n-POD [170]
Strawberry	WCS-YOLOv8s [171]
Raspberry	YOLOv8n-day/-night [172]
Pineapple	DPD-YOLOv8 [173]
Multiple crops	DCGA-YOLOv8 [174]
Pepper	Improved YOLOv8 [175]
Pea	YOLOv8-Peas [176]
Wheat	Improved YOLOv8 [177]
Soybean	YOLOv8-VEW [178]
Oilpalmuav	HR-YOLOv8 [179]
Sugarcane	Sugarcane-YOLO [180]
Pitaya	YOLOv8n-CBN [181]
Tomato	YOLOv8-C [182]
Eggplant	EggYOLOPlant [183]
Apple	YO-AFD [184]
Mango	Improved YOLOv8 [185]
Rice	YOLO-ECO [186]

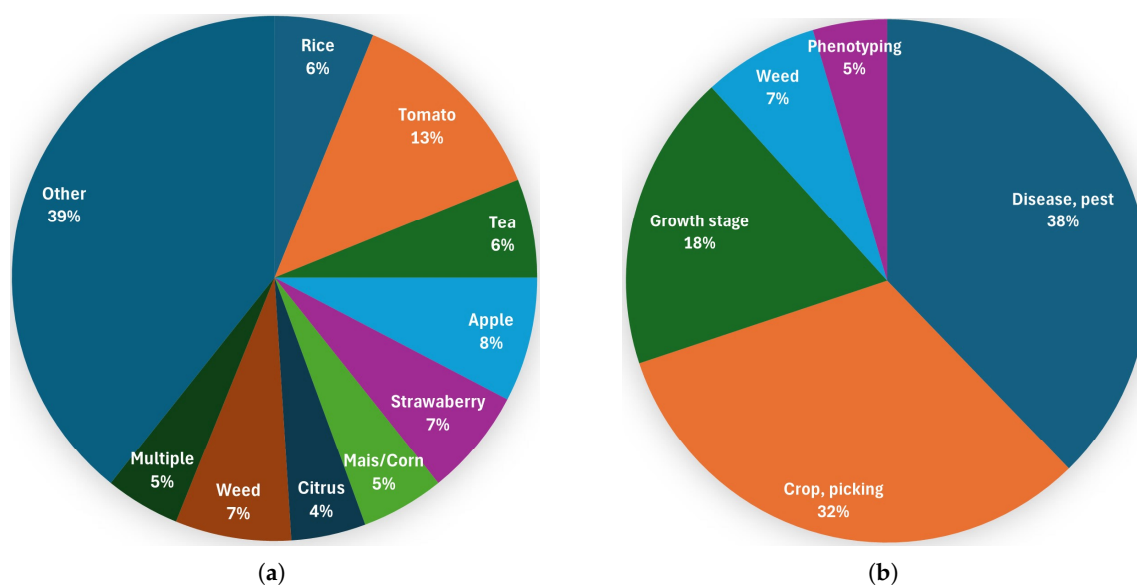
**Table 4.** Overview of improved YOLOv8-based algorithms/networks for weed detection.

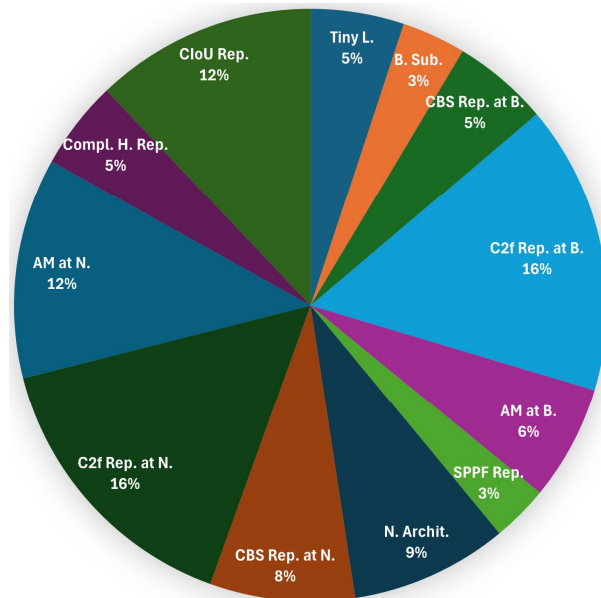
Plant species	YOLO Algorithm [Ref.]
Weed	BSS-YOLOv8 [187], YOLOv8-ECFS [188], EDS-YOLOv8 [189], RVDR-YOLOv8 [190], Improved YOLOv8 [191], ADL-YOLOv8 [192], BFFDC-YOLOv8-seg [193], YOLOv8-DMAS [194], EY8-MFEM [195], YOLOv8n-CBAM-C3Ghost [196], YOLOv8-MBM [197], YOLOv8-Weed Nano [198], YOLO-CWD [199], Star-YOLO [200]

**Table 5.** Overview of improved YOLOv8-based algorithms/networks for plant phenotyping.

Plant species	YOLO Algorithm [Ref.]
Maize/Corn	LCS-YOLOv8 [201], DBi-YOLOv8 [202]
Herbaceous	YOLOv8s-KDT [203]
Tree targets	ODN-Pro [204]
Apple	Improved YOLOv8 [205]
Asparagus	YOLOv8-CBAM [206]
Peanut	MS-YOLOv8 [207]
Watermelon	Improved YOLOv8 [208]
'Yuluxiang' pear	YOLO-RCS [209]

Figure 4 shows the distribution of the reviewed studies across agricultural monitoring and harvesting task categories. As can be seen, the most Improved YOLOv8 versions (38%) are related to disease and pest detection, which confirms its high relevance in agricultural object detection. The application of YOLOv8 variants for real-time identification of crops, picking points, and ripeness with efficiency and precision is similarly important, as demonstrated in 32% of the investigated papers. Next, accurately detecting crop growth stages is important for agricultural monitoring. This information, which was addressed in 18% of the reviewed studies, is essential for planning crops, predicting yields, and reducing fertilizer and workforce consumption. Lastly, weed detection is a traditional application field of computer vision and, thus, of YOLOv8.

**Figure 4.** Distribution of the reviewed studies (a) by plant type; (b) by categories of agricultural monitoring and harvesting tasks.



**Figure 5.** Frequency of replacement or extension of the respective YOLOv8 module or section in the reviewed studies (Table A1). Legend: Tiny L.—Addition of a Tiny object detection Layer at P2, B. Sub.—Substitution of the complete Backbone, CBS Rep. at B.—CBS Replacement at the Backbone, C2f Rep. at B.—C2f Replacement at the Backbone, AM at B.—Addition of an AM in the Backbone, SPPF Rep.—Replacement of the SPPF module, N. Archit.—Change of the Neck Architecture, CBS Rep. at N.—CBS Replacement at the Neck, C2f Rep. at N.—C2f Replacement at the Neck, AM at N.—Addition of an AM in the Neck, Compl. H. Rep.—Replacement of the complete Head, Clou Rep.—Replacement of the Clou loss function.

### 3.3. Improved YOLOv8 Taxonomy of this Paper

The YOLOv8 network structure (Figure 2) consists of three main sections: backbone, neck, and head. Accordingly, the improvement actions introduced relate to each of these sections. We have therefore analyzed the improvements and grouped them into the main categories in addition to pyramid modifications. Our Improved YOLOv8 Taxonomy presented in the paper is shown in Figure 6. In each category, there are corresponding component replacements or additions.

In the following subsections, we provide our overview and highlight the results of selected methods and/or comparative experiments on introduced improvement or extension measures in the reviewed literature. The presentations and discussions are based on the comprehensive overview of YOLOv8's modules given in Table A1. It is important to note that we accept the performance indicator values as stated in the papers without checking or reproducing them because doing so is not possible or would exceed the scope of this paper.

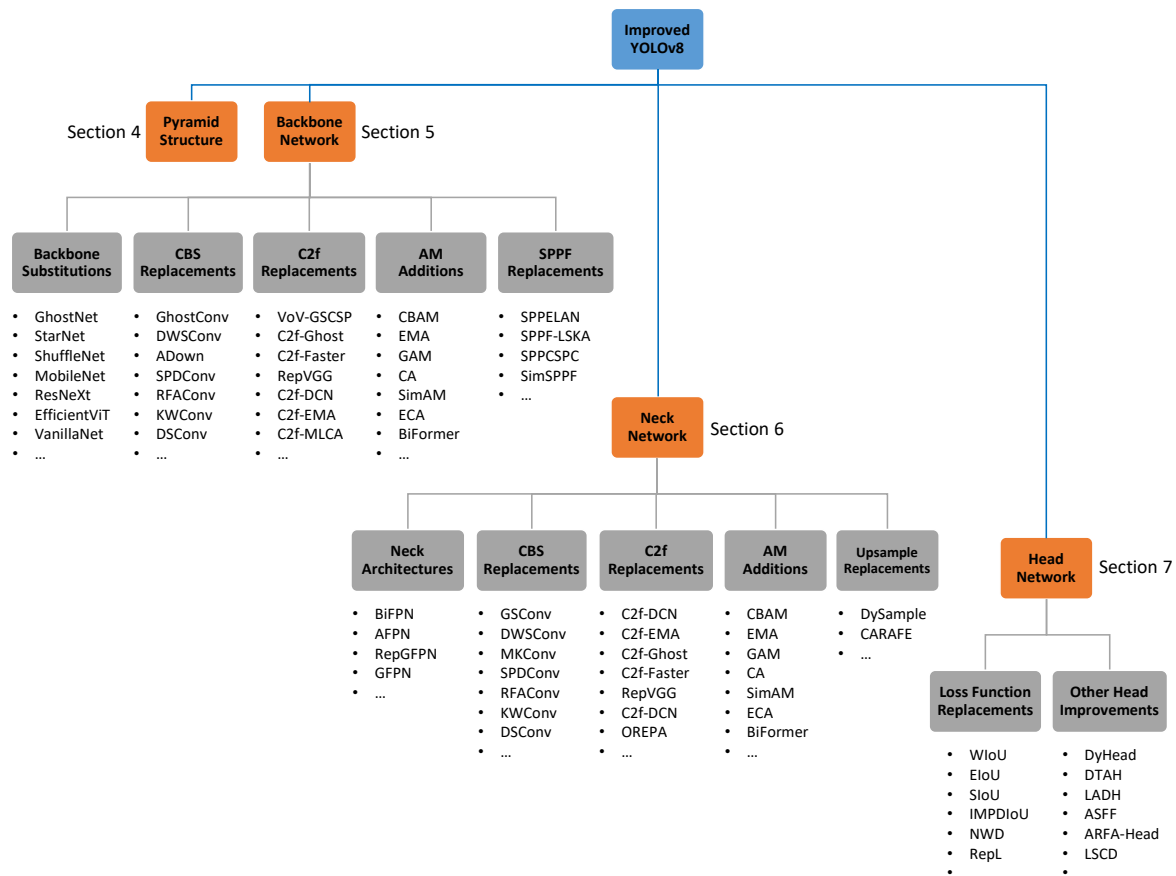


Figure 6. Improved YOLOv8 Taxonomy

#### 4. Pyramid Structure Modifications

Some improved versions of YOLOv8 extended the original feature pyramid structure at the  $160 \times 160$  scale (P2), working with an additional detection head to improve performance, as proposed by many researchers, e.g., in [15,80,157]; see Figure 7. An additional layer addresses the challenges associated with detecting small objects and the depletion of semantic knowledge due to varying scales [80]. The corresponding detection head uses high-resolution feature maps to accurately predict the location of small targets. This solves the problem of losing information about small targets during feature transfer. [15]. On the other hand, other researchers have suggested simplifying the complex structure of the standard YOLOv8 model to reduce the number of parameters. For instance, Fang and Yang [177] eliminated the large target detection modules and decreased the number of detection heads from three to two.

The addition of the fourth detection head is expected to have a large impact on the overall accuracy improvement. Yao et al. [66] demonstrated in their wheat disease detection case study that the addition of a small object detection layer resulted in a 3.38% increase in mAP and a 5.35 decrease in FPS. A remarkable 8.2% increase in mAP for small object detection was achieved in the work of Yue et al. [79] on the detection of crop pests and diseases. Adding a tiny object layer increased mAP by 9.0% and 6.3% for two different prediction models: YOLOv8n-day for daytime and YOLOv8n-night for nighttime in the Zhang et al. [172] study on raspberry and stalk detection. Improvements of 4% in mAP were achieved by introducing specialized small target detection in [23].

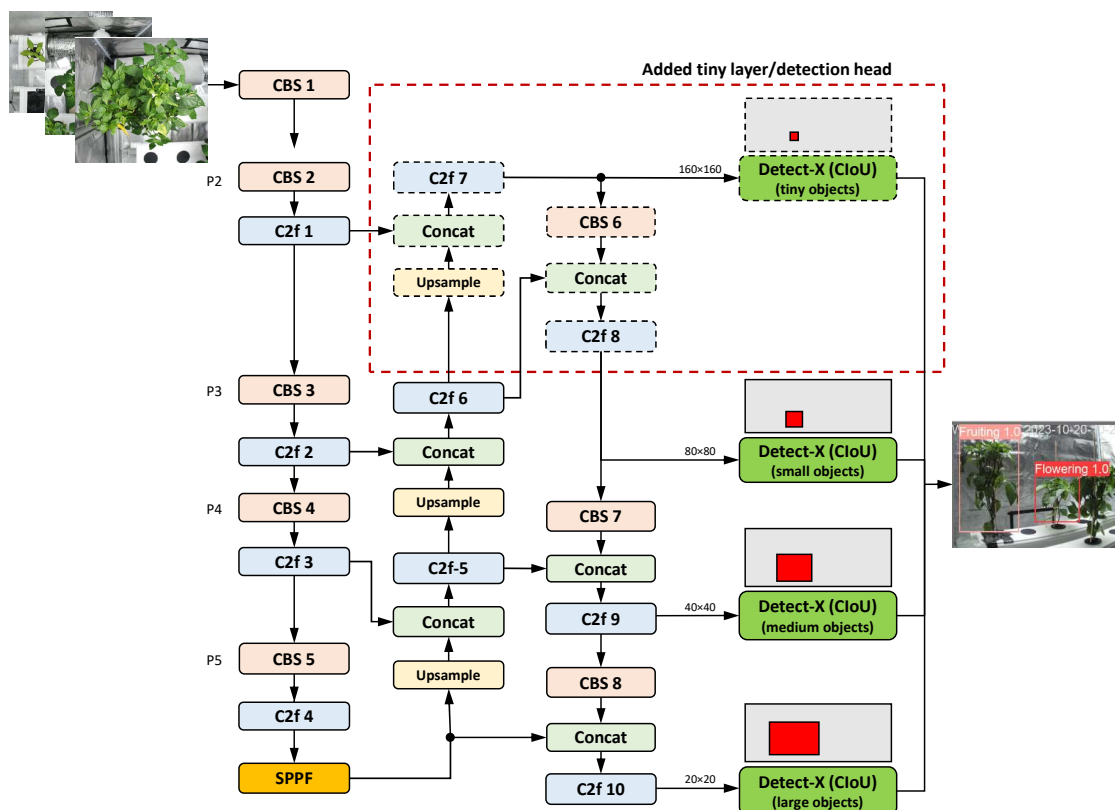


Figure 7. YOLOv8 network architecture with additional small-object pyramid layer at the  $160 \times 160$  scale (P2)

## 5. Backbone Improvements

The design of the backbone network structure is intended to efficiently extract multiscale feature information.

### 5.1. Backbone Substitutions

The complete substitution of the backbone network (except for the SPPF module) represents another structural improvement in the YOLOv8 algorithm. The use of EfficientViT as the backbone network was proposed in [91,189]. Other researchers replaced the backbone by StarNet [201] or RepViT [189]. Improved YOLOv8 networks, in which the backbone network has been replaced with either the Swin Transformer (SwinT) or the Deformable Attention Transformer (DAT), were presented in [152] and [83], respectively. Wang et al. [20] replaced the backbone with MobileNetV4, a new generation lightweight detection network introduced in April 2024. Other backbone replacements and transformer inclusions can be found in Table A1.

In [160], the performance of four different backbone networks (ShuffleNetV2, GhostNetV2, EfficientViT2, MobileNetV3) was compared with that of the C2f-Faster backbone proposed by Zhang et al. while keeping all other parameters constant. It was shown that the C2f-Faster improvement method and MobileNetV3 provided significant advantages in precision, recall, and average precision compared to the other networks. Wang et al. [37] provided comparative experiments to investigate the impact of different backbone networks on the performance of the YOLOv8 model for tea leaf disease detection. The backbone networks included the original YOLOv8 backbone, ResNeXt, MobileNetV2, ShuffleNetV2, EfficientNetV2, and RepVGG. The efficacy of replacing the original backbone with lightweight networks such as MobileNetV2, ShuffleNetV2, or EfficientNet was demonstrated, leading to an increase in detection speed, albeit at the expense of detection accuracy. In contrast, using the RepVGG backbone significantly reduced the number of parameters, floating-point operations, and memory consumption while maintaining good detection performance (mAP: +2.43%) [37]. The experimental results in [183] showed that replacing the backbone with an Mblock-based architecture

derived from the MobileNetV3 feature extraction module led to a 3.0% improvement in mAP50 over the baseline YOLOv8 model.

Replacing the backbone network with ConvNeXtV2, as proposed by Chen et al. [103], increased mAP by 2.09%. This significant improvement indicates that ConvNeXtV2 better captures image features and improves model performance compared to the original YOLOv8 structure.

In their study of weed detection in cotton fields, Zheng et al. [200] restructured the YOLOv8 backbone network and studied ten *lightweight* networks that have been widely used in recent years, including ConvNext, EfficientNet, FasterNet, MobileNeXtV3, MobileNetV3, PP-LCNet, ShuffleNetV2, VanillaNet, GhostNet, GhostNetV2, and StarNet. They found out that their improved algorithm using StarNet significantly outperforms other lightweight networks in terms of mAP (98.0%). Furthermore, a comparative analysis revealed that the performance of all other networks exhibited a range of degrees of deterioration relative to the baseline model. These results suggest that, while lightweight models can effectively reduce the number of parameters and computational requirements, excessive light weighting may reduce the feature extraction capabilities, resulting in the loss of semantic information and consequently degraded recognition accuracy [200]. For example, VanillaNet achieved the best results in terms of light weighting, but it had the lowest detection performance, with an mAP of only 89.0%, which means a decrease of 8.9% in mAP.

### 5.2. Inclusion of Transformer Networks

In addition to replacing the entire backbone network with a Vision Transformer Network (see Table A1), certain Transformer modules can be incorporated into the backbone in place of CBS, C2f or SPPF, or to bridge the gap between the backbone and the neck. In the Improved YOLOv8s network proposed by Diao et al. [152] for maize row segment detection, the backbone network was replaced with a SwinT network, resulting in a 4.03% improvement in mAP. Based on the analysis in [189], EfficientViT was proposed as the backbone network since it showed the highest performance improvement (+6% in mAP) compared to the original YOLOv8 backbone (baseline) and RepViT (+5.9% in mAP). Compared to the baseline model, Lin et al. [120] demonstrated that using the NextViT backbone network improved mAP for citrus fruit detection by 4.0%. In comparative experiments involving six CNN- and transformer-based detectors, Lin et al.'s AG-YOLO performed best across all metrics. According to the study by Fan et al. [129] on peach fruit detection, using FasterNet as the backbone resulted in a 5.2% increase in mAP. Fu et al. [91] presented an Improved YOLOv8 network structure utilizing EfficientViT as its backbone. They achieved an mAP increase of 6.9% using a self-constructed dataset of tomato fruit.

Do et al. [53] proposed an SC3T module as a replacement for the SPPF, combining the SPP and C3TR modules. As demonstrated by the conducted experiments, the mAP result was found to be promising, with a value of 78.1%, which represents a 5.8% increase over the baseline model. This was determined using a dataset focusing on strawberry diseases.

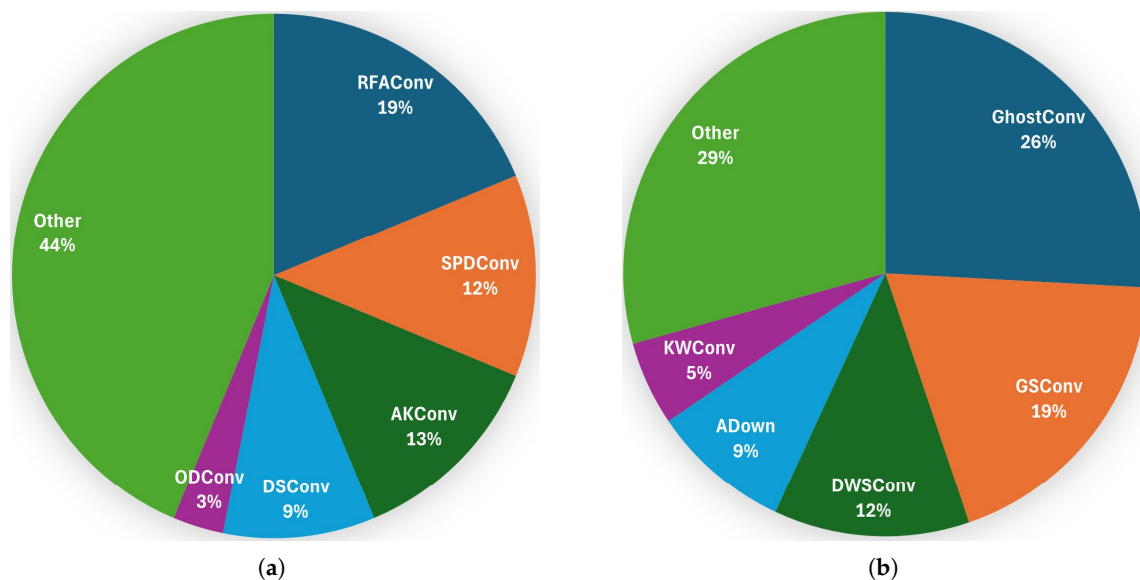
### 5.3. CBS Replacements

The regular convolution module, i.e., CBS (Conv2d + BatchNorm2d + SiLU), used in YOLOv8, applies a convolution kernel to each channel of the input feature map and adds the convolution results for each channel to create a single output feature map. This process is repeated on all channels of the input feature map to create multiple output feature maps [88]. To reduce the number of parameters and the computational complexity of this process, various CBS improvements have been proposed, such as DWConv, PDWConv, GSConv, and AKConv. Other CBS replacements for the Backbone (B) are listed in Table A1. The classification of a CBS replacement as either a model for higher-accuracy or a lightweight model is not clear-cut. In some cases, it is somewhat fluid.

Among the reviewed papers, the most commonly used CBS replacement is RFACConv, followed by SPDConv and AKConv, for higher-accuracy models; see Figure 8. For lightweight models, GhostConv is the most prevalent, followed by GSConv and DWConv. Furthermore, it can be observed that the replacement of CBS modules at the backbone for higher-accuracy models has been proposed

with a significantly higher frequency than at the neck. The underlying rationale is evident in the understanding that downsampling operations inherently entail a loss of information. This loss can be mitigated through the implementation of optimized convolution techniques in the early stage (i.e., at the backbone), thereby enhancing the efficacy of the feature extraction process.

Conversely, the replacement of CBS modules at the neck with lightweight models has been proposed at a significantly higher frequency than at the backbone. This objective is twofold: to reduce model complexity while preserving model accuracy.



**Figure 8.** Frequency of replacement options for the CBS module (a) for higher-accuracy YOLOv8 models ( $N = 54$ ); (b) for lightweight YOLOv8 models ( $N = 49$ ).

The primary goal of replacing CBS should be to reduce model complexity while maintaining detection accuracy, as confirmed in [192]. In this context, the Ghost Module proposed by Han et al. [210] plays a prominent role. It is a model compression technique that reduces redundant feature extraction across channels. This results in a reduction of the number of parameters and the computational cost, while ensuring that model accuracy and complexity are maintained [154].

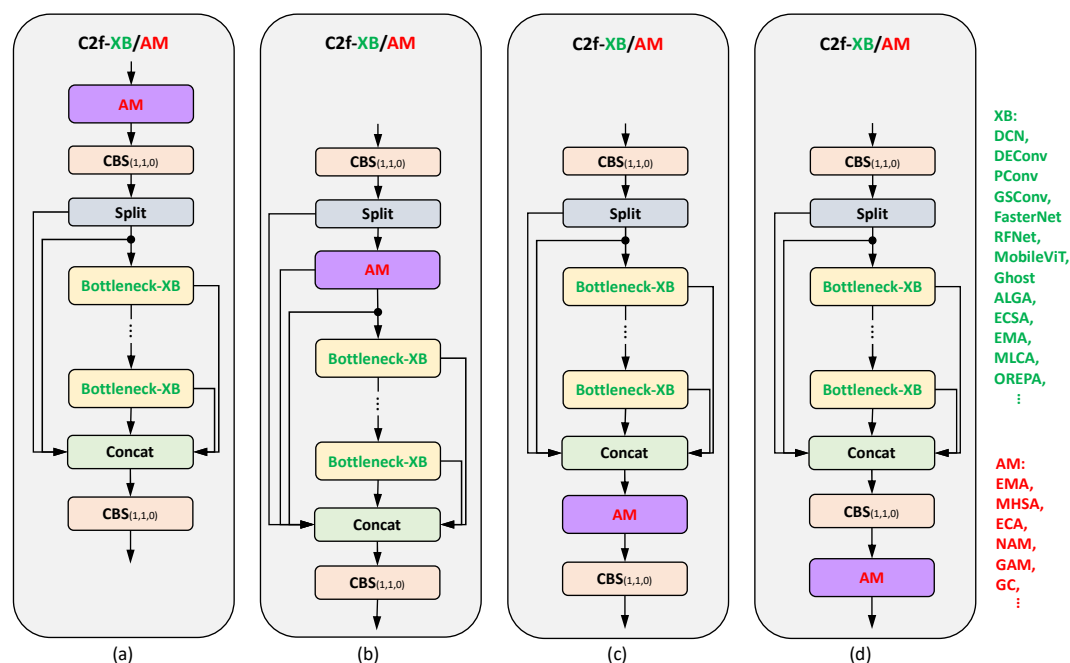
He et al. [50] compared the 3 different convolution modules DSCConv, SPDCConv, and KWConv, and found that SPDCConv achieved the highest mAP increase of 1.2% compared to the baseline YOLOv8 model. However, it significantly increases the computation time to 43 GFLOPS. Replacing the standard convolution with KWConv increases the mAP by 1.0%, but the computation time is much lower at 14.2 GFLOPS. As a suggestion, KWConv is the right option when detection speed is a priority, SPDCConv for applications where accuracy is the primary concern. Wang et al. [69] showed that replacing CBS with SAConv resulted in a significant improvement in mAP of 1.67%.

Another convolution option is to use GSCConv, a combination of Spatial Convolution (SC), DWSConv, and Shuffle operations, as suggested in [187]. Compared with the baseline network YOLOv8n, the model fused with the GSCConv module showed better detection performance, achieving a 1.9% increase in mAP. This finding suggests that the incorporation of GSCConv (included in the backbone) and VoV-GSCSP (included in the neck) modules within the network architecture can enhance the diversity and richness of feature extraction while concurrently simplifying the network structure for grassland weed detection. In the comparative analysis of YOLO models for coffee fruit maturity monitoring presented by Kazama et al. [162], the integration of the RFCACConv module into YOLOv8n achieved a mAP of 74.20%, outperforming the standard YOLOv8n by 1.90%. In their study on tomato leaf disease detection, Shen et al. [30] showed that replacing the second CBS in YOLOv8 with GDC resulted in a remarkable 3.4% increase in mAP.

#### 5.4. C2f Replacements

The C2f module is a pivotal component of the YOLOv8 network. Most of the improvements in the backbone thus focused on replacing C2f with various alternatives, such as C2f-DCN to improve the feature extraction ability of plant diseases [14,66], PDWConv to optimize the network efficiency [110], RepBlock to improve accuracy and allow structural re-parameterization for increased detection speed [37], OREPA module to accelerate the training process [159], and GELAN to streamline the backbone network structure, optimize its feature extraction capabilities, and achieve model lightweighting [41]. Novel options to strengthen the model's feature extraction capabilities and thus improve the model's detection accuracy are to integrate attention mechanisms into the C2f module, such as C2f-DCN, C2f-MLCA, and C2f-EMA. Other replacements of C2f are given in Table A1. As for the classification of CBS replacements, it is not clear-cut whether a C2f replacement is a model for higher-accuracy or a lightweight model.

**Note.** There are different ways to integrate submodules into C2f to get new C2f structures. Most of these replace the bottleneck in C2f with a specific submodule, such as the DCN block or the FasterNet-EMA block; see Figure 9.

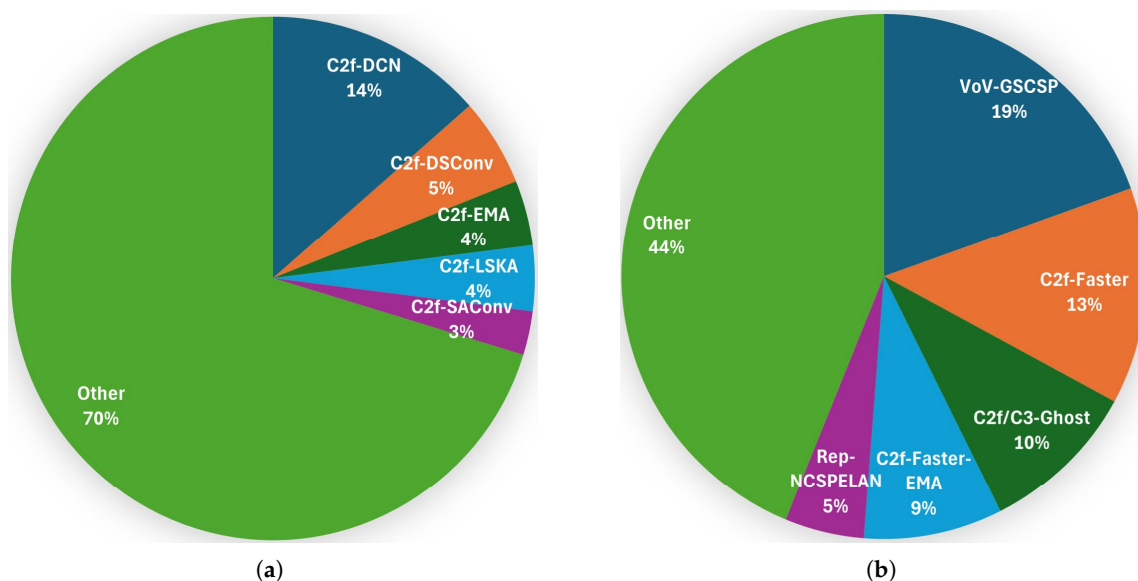


**Figure 9.** Structure diagrams of possible C2f configurations with Bottleneck replacement blocks (XB) and some options for integrating attention mechanisms (AM) into the C2f structure: whereas structures (a) and (d) can be merely serial connections of C2f and AM, structures (b) and (c) achieve real integration of C2f and AM. Furthermore, AM can be embedded within the C2f bottleneck.

Among the reviewed papers, the most commonly used C2f replacement is C2f-DCN, followed by C2f-DSCConv and C2f-EMA, for higher-accuracy models; see Figure 10. A substantial body of research has been dedicated to replacing C2f, with a significantly smaller number of studies focusing on replacing CBS. This observation underscores the heightened significance of C2f in the YOLOv8 architecture.

For lightweight models, VoV-GSCS is the most prevalent, followed by C2f-Faster and C2f/C3-Ghost. C2f-DCN structures, which are based on deformable convolution, are typically integrated into networks to improve their ability to adapt to target deformation. This improves the networks' ability to extract features of crops or crop diseases. Both VoV-GSCS (based on GSConv) and C2f-Ghost (based on GhostConv) are lightweight C2f structures that simplify the network architecture and reduce computational requirements.

The observations and conclusions concerning the frequency of C2f replacement are analogous to those concerning the CBS replacement when comparing the backbone and the neck.



**Figure 10.** Frequency of replacement options for the C2f module (a) for higher-accuracy YOLOv8 models ( $N = 74$ ); (b) for lightweight YOLOv8 models ( $N = 82$ ).

Recently, Jia et al. [170] investigated the performance effect of some C2f modifications (C2f-GOLDYOLO, C2f-EMSPConv, C2f-SCcConv, C2f-EMBC, C2f-DCNv3, C2f-DAM, and C2f-DBB) within the backbone network of YOLOv8. The experimental results presented there demonstrated that the inclusion of all these modules can improve the detection accuracy of YOLOv8 to a certain (small) extent (mAP: 77.8–80.4%). Relatively speaking, the C2f-DBB module showed greater potential in improving the network's feature extraction capabilities [170].

In [66], Yao et al. showed that the C2f-DCN module alone improved the detection accuracy (mAP) by 2.13% for wheat disease detection. Guan et al. [202] investigated replacing the C2f module with a C2f-DCN module at different positions in the backbone. When all four positions are replaced with C2f-DCN modules, the mAP of the model increases by 7.7% compared to the original model, indicating that the modified YOLOv8 network performs better in detecting maize plant organs.

Liu et al. [161] introduced and embedded four attention mechanisms (CA, SE, LSKA, and EMA) into the C2f structure of YOLOv8s-p2 for comparative experiments on the detection of green crisp plum and demonstrated that embedding EMA yields the best performance values, e.g., +1.1% for mAP, compared to the YOLOv8s-p2 model as baseline. When the C2f-ALGA module was integrated, the experimental results described in [195] showed a 2.9% increase in mAP compared to the YOLOv8s model. In the study of Jin et al. [23] on rice blast detection, substituting C2f with C2f-ODConv led to a substantial 4.2% increase in mAP.

Yan and Li [34] proposed replacing C2f with PFEM to introduce multilayer convolution operations within the feature maps. This achieves a richer feature representation, thereby enhancing the model's ability to recognize tomato leaf diseases against complex backgrounds. The case study presented showed that integrating the PFEM module significantly improves model performance, increasing mAP by 4.5%.

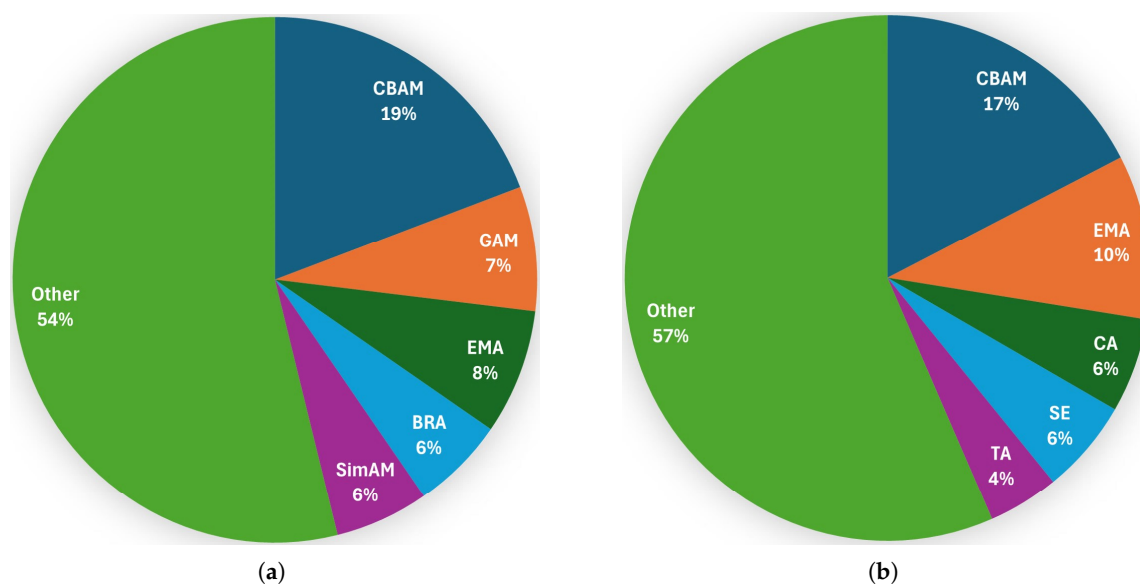
### 5.5. Additions of Attention Mechanisms

Basic YOLO models typically rely on global features or fixed receptive fields for object detection and localization. However, these are not optimal when dealing with complex scenes, occluded objects, or scale variations, such as those found in crop or disease detection tasks. To address these issues, researchers have initiated the exploration of the inclusion of attention mechanisms in the

YOVOv8 architecture. The fundamental premise of attention mechanisms is to empower the model to autonomously discern the regions or features that are of paramount importance for the object detection task. Attention mechanisms draw inspiration from the human vision system, which focuses on essential regions in complex scenes. Similarly, by introducing attention mechanisms in computer vision, the model can focus more on the target regions and ignore background distractors, thereby improving object detection accuracy [14]. However, it is important to note that including an attention mechanism in the network will inevitably increase computing costs.

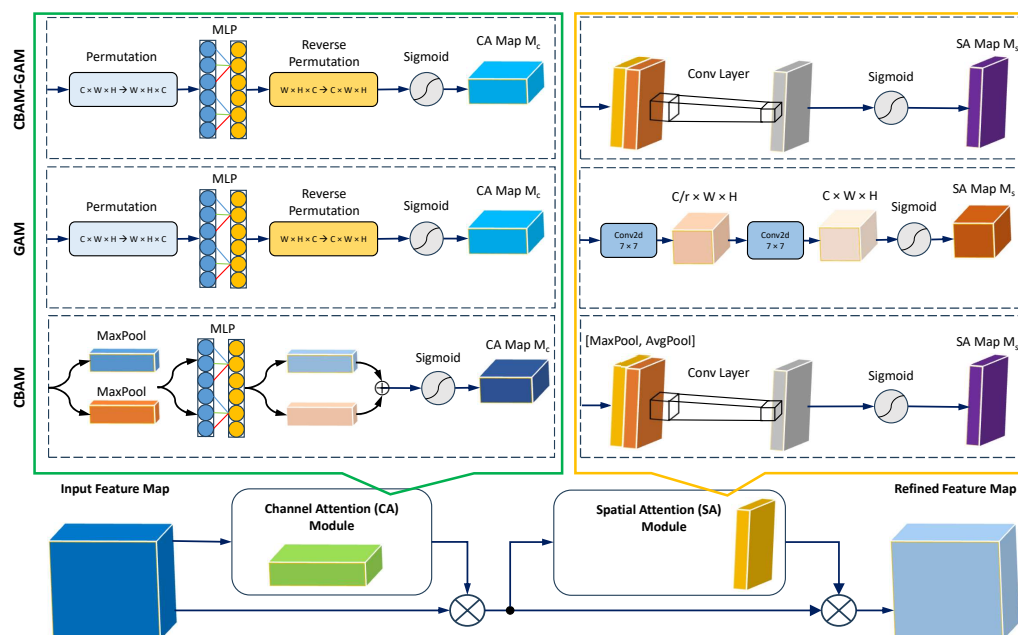
Attention mechanisms employed in visual recognition tasks can be broadly categorized into three distinct types: channel attention, spatial attention, and mixed-domain attention. In recent work, for example, CA, EMA, ECA, and GAM, have been included somewhere in the backbone, usually before or after the SPPF module, to enhance the feature extraction capability of the model at different spatial scales and further improve the accuracy. Other suggestions for including attention mechanisms in the backbone are given in Table A1. We do not categorize attention mechanisms because they are always integrated for higher accuracy, despite their varying levels of complexity.

Among the reviewed papers, the most commonly suggested attention mechanism in the backbone is CBAM, followed by GAM, then EMA; see Figure 11. Figure 12 illustrates the structures of CBAM and GAM, as well as their possible combination, as presented in [151]. Both CBAM and GAM have two submodules that are applied sequentially to the input feature map: the channel attention module (CAM) and the spatial attention module (SAM). Convolution is used to calculate spatial attention and establish spatial location information. However, such calculations can only capture local relationships and cannot capture long-distance dependencies [211]. Furthermore, attention mechanisms such as CBAM and GAM fail to consider the combined effect of both attentional mechanisms on the features. This effect is mixed after calculating one- and two-dimensional weights, respectively [189].



**Figure 11.** Frequency of AM extension options (a) for the backbone ( $N = 52$ ); (b) for the neck ( $N = 69$ ).

In contrast, EMA uses a parallel processing strategy that considers feature grouping and multiscale structures. Using the cross-spatial learning method, the parallelization of convolution kernels seems to be a more powerful structure for handling both short- and long-range dependencies [212]. Another promising attention mechanism that is simple yet effective is SimAM. It infers three-dimensional attention weights for intralayer feature maps without increasing the number of network parameters. This allows the model to learn from three-dimensional channels, improving its ability to recognize object features [70,189].



**Figure 12.** Diagrams illustrating the structure of the most notable attention mechanisms (AMs) as an improvement to YOLOv8: CBAM, GAM, and their combination. Legend: MLP—Multilayer Perceptron, Conv—Convolution, AvgPool—global average pooling, MaxPool—global max pooling,  $\otimes$ : broadcast element-wise multiplication,  $\oplus$ : broadcast element-wise summation. Feature maps are displayed as feature dimensions, e.g.,  $C \times H \times W$  refers to a feature map with channel number  $C$ , height  $H$  and width  $W$

In [14], a comparative performance analysis of 12 different attention mechanisms (CBAM, CA, GAM, ECA, TA, SE, SimAM, CoT, MHSA, PPS—Polarized Self-attention, SPS—Sequential Polarized Self-attention, and ESE—Effective SE) modules placed after the SPPF module led to the conclusion that TA yields a great performance improvement (mAP: +15.1%) for object detection of rice leaf diseases. Using SPS and GAM as attention modules also improved the YOLOv8 model's mAP values and detection accuracy for the four types of rice leaf diseases by 15.3% and 15.1%, respectively. However, the detector's performance was found to be highly unstable when detecting certain disease features, especially small targets like rice blast [14]. Another comparative study in [159] showed that the attention mechanism MPCA, placed after the SPPF module, has superior accuracy (+1.5% in mAP, compared to the baseline) for blueberry fruit detection compared to nine other attention mechanisms (SimAM, LSKBlock, TA, SE, Efficient Attention, BRA, BAMBlock, EMA, and CA). In addition, the introduction of two OREPA modules in the backbone and two OREPA modules in the neck resulted in a 2.7% improvement in mAP. The experiment in [50] proved that TA, when added after each C2f module in both the backbone and the neck, led to a 3.6% improvement in mAP, compared to the baseline model YOLOv8s, for strawberry disease detection.

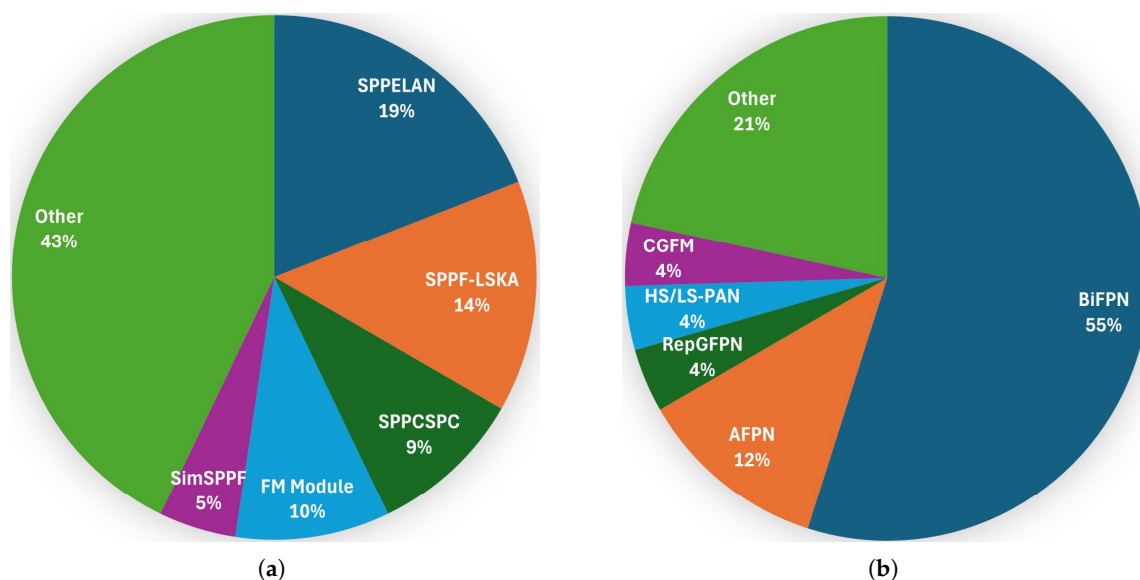
Under the EfficientViT backbone network, He et al. [189] showed that adding DSConv and SimAM integrated into the neck led to an improvement of 6.4% in mAP, beating structures like EMA and BiFormer. Therefore, He et al. suggested the combination of C2f-DSConv and SimAM as the optimal method for weed detection tasks. Another comparison of the detection results of four different attention modules (EMA, SE, CBAM, and CA) in [211] showed that under the premise of unchanged params and GFLOPs, EMA leads to the best results, i.e., an increase of mAP up to 3.55%, compared to SE, CBAM, and CA. Five different attention mechanisms (SE, CBAM, ECA, CAB, and BiFormer) were investigated in [27] on the YOLOv8n baseline model for detecting tea defects and pest damage. The findings indicated that the incorporation of BiFormer into the model's backbone network resulted in the best detection performance: +16.5% increase in mAP, followed by +11.1% when CAB was used to detect tea diseases and defects.

Yin et al. [16] showed that the inclusion of the Multibranch CBAM (M-CBAM) module improved the detection performance in terms of mAP by 1.8% over the baseline model in their rice pest detection

study. In the blueberry fruit detection study by Gai et al. [159], adding the MPCA module resulted in a 1.5% increase in mAP. A comparison of incorporating different attention mechanisms (MPCA, CPCA, BRA, SE, BAMBlock, and SEAM) into the backbone led to the conclusion that the SEAM attention mechanism is particularly advantageous for detecting small objects [170].

### 5.6. SPPF Replacements

YOLOv8's SPPF structure generates feature maps of different scales through multiple pooling operations, often containing similar information. In addition, the simple concatenation of these feature maps can lead to duplicate representations of features, making the model overly dependent on similar feature representations and increasing the risk of overfitting. This problem can occur with weeds and crops because they often have symmetrical morphologies and are very similar both horizontally and vertically [195]. SPPF employs three max-pooling methods to extract input features in series. Nonetheless, max pooling merely extracts the maximum value of the input feature. Consequently, it can only represent the local information of the input feature, while ignoring the global feature information of the input image [38]. Consequently, some researchers have proposed the combination of average pooling and maximum pooling to enhance the extraction of global information by SPPF. Mix-SPPF is one such module. Replacing the SPPF module was suggested, e.g., SimSPPF, SPPELAN, and MDPM. Other SPPF replacements are listed in Table A1. Among the reviewed papers, the most commonly used SPPF is SPPELAN, followed by SPPF-LSKA; see Figure 13.



**Figure 13.** Frequency of replacement options (a) for the SPPF module ( $N = 21$ ); (b) for the neck architecture ( $N = 51$ ).

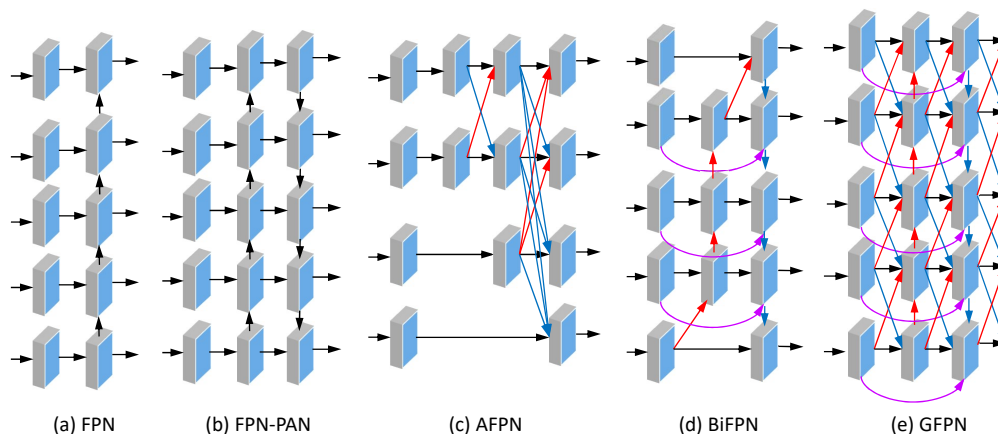
The ablation study by Wei et al. [151] showed that the replacement of SPPF with SPPELAN alone leads to a significant increase in performance: +2.5% in Precision and +2.2% in Recall. In [38], MixSPPF proposed by Ye et al. was compared with SPPF-LSA and SPPF-LSKA, and demonstrated superior performance in mAP, i.e., +3.89% compared to the baseline model with SPPF, for a case study of weed detection in cotton fields.

## 6. Neck Improvements

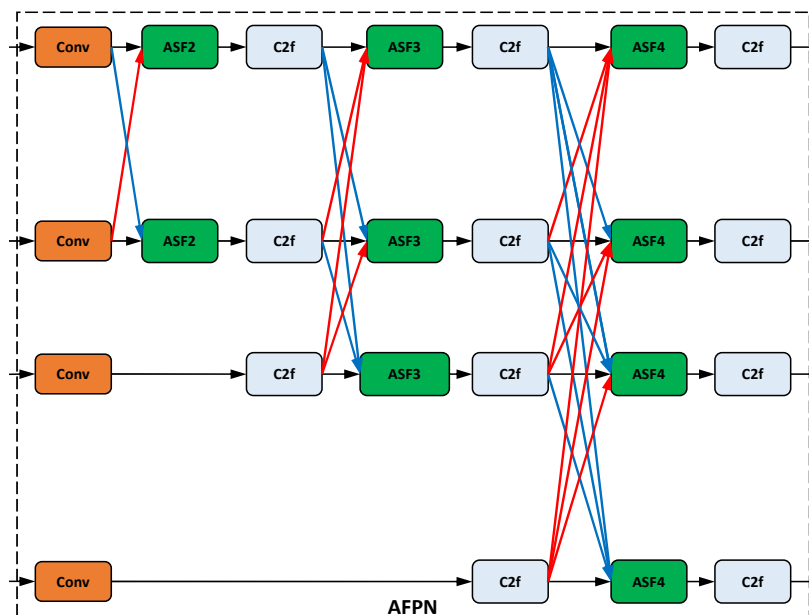
### 6.1. Neck Architectures

The most widely used architectures of YOLOv8's neck are the FPN and the PAN. Beyond the FPN/PAN, some cascade fusion strategies have been developed and shown to be potentially effective in multiscale feature fusion, such as the (weighted) BiFPN, the GFPN, the AFPN, and the GFPN. Other neck architectures are listed in Table A1. Some of these neck structures are illustrated in Figure 14.

Among the works examined, BiFPN is by far the most commonly used neck architecture, followed by AFPN; see Figure 13. The BiFPN's bidirectional architecture facilitates the acquisition of supplementary contextual information through two-way feature exchange. This enhancement in efficiency of feature propagation within the network, in turn, elevates the quality of feature representation [181]. The architecture of the AFPN is designed with the specific intention of addressing substantial semantic fusion gaps between adjacent layers of the backbone [71]. Figure 15 shows an AFPN configuration.



**Figure 14.** Some important FPN designs (based on [213,214]): (a) FPN utilizes a top-down strategy; (b) PAN adds a bottom-up pathway on top of FPN; (c) AFPN strengthens interlayer interactions; (d) BiFPN integrates cross-scale pathways bidirectionally; (e) GFPN (or GiraffeDet) includes a queen-fusion style and skip-layer connections. During this fusion process, arrows pointing diagonally upwards signify upsampling, while those pointing diagonally downwards indicate downsampling.



**Figure 15.** A configuration of AFPN for a neck network with four detection layers

In [97], it was shown that the introduction of the BiFPN in the neck enhanced the fusion capability of the model's multiscale features for tomato ripeness detection, resulting in a 3% improvement in mAP compared to the baseline YOLOv8n. According to [157], the mAP increased by 2.5% after replacing PAN as a neck network with BiFPN. According to the study by Fan et al. [129] on peach fruit detection, using BiFPN as the neck resulted in a high increase in mAP by 9.6%. A higher improvement (+10.9% in mAP) when replacing FPN-PAN with BiFPN was reported in [27] for detecting defects and pest damage in tea leaves.

Wu et al. [141] implemented six different extensions to the YOLOv8n model, YOLOv8n-CARAFE, YOLOv8n-EfficientRepBipan, YOLOv8n-GDFPN, YOLOv8n-GoldYolo, YOLOv8n-HSPAN, and YOLOv8n-ASF, for grape detection. The results showed that YOLOv8n-ASF achieved a mAP of 1.78% higher than the baseline model and demonstrated the best overall improvement in grape detection. Sun et al. [64] showed that replacing the FPN-PAN neck design of YOLOv8 with an AFPN structure resulted in a 1.92% improvement in mAP for detecting pest species in cotton fields.

In their study, Wu et al. [94] compared the performance of their MTS-YOLO algorithm with five other modifications of the YOLOv8 neck network: YOLOv8-EfficientRepBiPAN [215], YOLOv8-GDFPN [216], YOLOv8-GoldYOLO [217], YOLOv8-CGAFusion [218], and YOLOv8-HS-FPN [219]. It was shown that MTS-YOLO gave the best overall performance, e.g., +1.4% in mAP compared to the original YOLOv8n model. In the ablation study of Chen et al. [156] on melon ripeness detection, the integration of HS-PAN improved mAP by 2.6%.

### 6.2. CBS Replacements

As for the backbone, similar CBS replacements can be incorporated into the neck. Prominent options are KWConv, PDWConv, RFAConv. The lightweight subsampling convolution block, called ADown, was introduced in YOLOv9 with the specific purpose of enhancing object detection tasks, as described in [60]. Other CBS replacements for the Neck (N) are listed in Table A1. Among the reviewed papers, the most commonly used CBS replacement in the neck is GhostConv, followed by GSConv, then DWSCConv; see Figure 8.

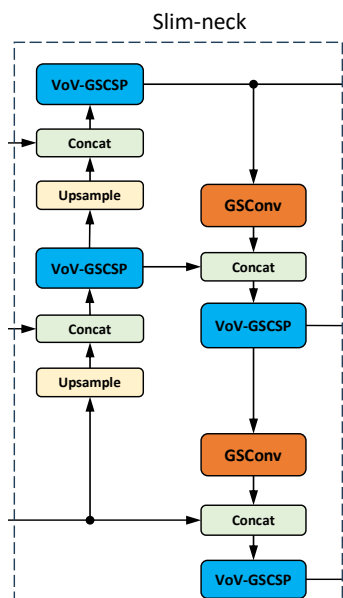
Wang et al. [220] showed that replacing standard convolution operations with RFAConv modules in the neck mainly reduces the computational cost, thereby increasing the computational speed. Note that the ALSA module relies on the structure of the RFAConv module [28]. Deng et al. [122] showed that replacing CBS with GhostConv resulted in a remarkably high 14% improvement in mAP over the original YOLOv8 model for citrus color identification. Replacing the standard CBS modules with ADown modules in both the backbone and neck improved the mAP by 1.3% in the leaf disease detection study by Wen et al. [86]. Including KWConv in both the backbone and the neck increased the mAP by 3.5% in the ablation study by Ma et al. [203] on the recognition of desert steppe plants.

### 6.3. C2f Replacements

In principle, the same C2f modules can be used for the neck as for the backbone. Special C2f replacements introduced in the neck are the lightweight C2f-Ghost [16,80,111] and C2f-Faster [92,130] to significantly reduce the number of parameters while improving detection performance. The integration of attention mechanisms, such as EMA and LSKA, was proposed to enhance the model's capacity to detect small targets and discern ambiguous characteristics. This integration resulted in the development of the C2f-EMA module and the C2f-LSKA module, as outlined in the works of Zhao et al. [178] and Fang and Yang [177], respectively. Other C2f replacements for the neck are listed in Table A1.

Among the works examined, VoV-GSCSP is the most commonly used C2f module in the neck, followed by C2f-Faster, for lightweight YOLOv8 models; see Figure 10. For higher-accuracy YOLOv8 models, most researchers used C2f-DCN in the neck.

Many studies [43,48,72,93,112,139,190,221] adopted the Slim-Neck design concept [222] for the neck components of YOLOv8, balancing feature extraction performance while significantly reducing model complexity and computational costs. The Slim-Neck architecture (Figure 16) consists of two components: a lightweight VoV-GSCSP module that replaces C2f and a GSConv module that takes the place of the standard Conv module. C2f-Faster is created by integrating the FasterNet module into the bottleneck, which makes the model more lightweight and reduces the network's computational load.



**Figure 16.** Slim-Neck architecture: a combination of VoVGSCSP (instead of C2f) modules and GSCConv (instead of CBS) modules is used to construct a lightweight model structure.

Lu et al. [14] showed that the C2f-DCNv2 module, when included in the backbone and the neck in place of C2f, helps to better align the irregular targets on the feature map. In the use case considered, this inclusion improved the mAP by 6% compared to the baseline YOLOv8n. Incorporating the EMA attention mechanism into the C2f within the backbone and neck of the base model (YOLOv8n-Pose) resulted in a 2% increase in mAP, with negligible impact on the number of parameters and computations, as shown in the experiments by Chen et al. [142] for grape and picking point detection. The results in [85] demonstrated that substituting C2f with C2f-ECA modules in the neck led to a substantial enhancement in plant disease detection through advanced spatial feature integration and characterization, resulting in a 1.9% increase in mAP. Replacing C2f with HCMamba resulted in a 2.7% increase in the mean average precision (mAP) in the study of Liu et al. [31] on tomato leaf disease detection.

#### 6.4. Additions of Attention Mechanisms

Attention mechanisms dynamically learn feature weights or attention distributions based on context and task requirements, allowing models to adaptively focus on key features [223]. Various attention mechanisms have been proposed for inclusion in the neck network, such as CA, CBAM, EMA, and SEAM. Other suggestions for including attention mechanisms in the neck are given in Table A1. The most commonly used attention mechanism in the neck is CBAM, followed by EMA, then CA; see Figure 11. The CA mechanism incorporates positional information into channel attention, thereby enhancing the importance of key features and achieving long-range dependencies [82,224].

Attention mechanisms can be introduced at different positions in the neck layer: as a bridge between the backbone and the neck [16,60,134], only to the FPN or only to the PAN [188], to both the FPN and the PAN [37], or at the exit of the neck (i.e., between the neck and the head) [160,191].

Deng et al. [122] showed that introducing the MCA mechanism to focus on capturing the local feature interactions between feature mapping channels in citrus color identification led to a 7.8% improvement in mAP over the original YOLOv8 model. In the study by He et al. [50], TA increased mAP by 2.4% when the Improved YOLOv8 algorithm was used to identify diseases in strawberry leaves.

In Lin et al.'s study [173] on pineapple fruit detection, five attention mechanisms (EMA, CPCA, SimAM, MLCA, and CA) were compared during the selection process. The results showed that CA led

to the highest level of attention toward the pineapple, increasing mAP by 4.8% and 5%, respectively, compared to EMA and MLCA.

Wei et al. [151] proposed a peculiar yet interesting approach: combining the GAM and CBAM attention mechanisms and integrating them at the neck. This option can compensate for the shortcomings of the single CBAM attention mechanism and improve the accuracy of the model to a greater extent (+1.6% in Precision, +1.8% in Recall), as shown for the detection of maize tassels [151].

The experimental results for mango inflorescence detection in [185] indicated that adding GAM in series with C2f-Faster (C2f-Faster-GAM) to both the backbone and the neck improved the mAP by 2.2%, outperforming other attention mechanisms such as SE, CA, and CBAM.

### 6.5. Upsampling Replacements

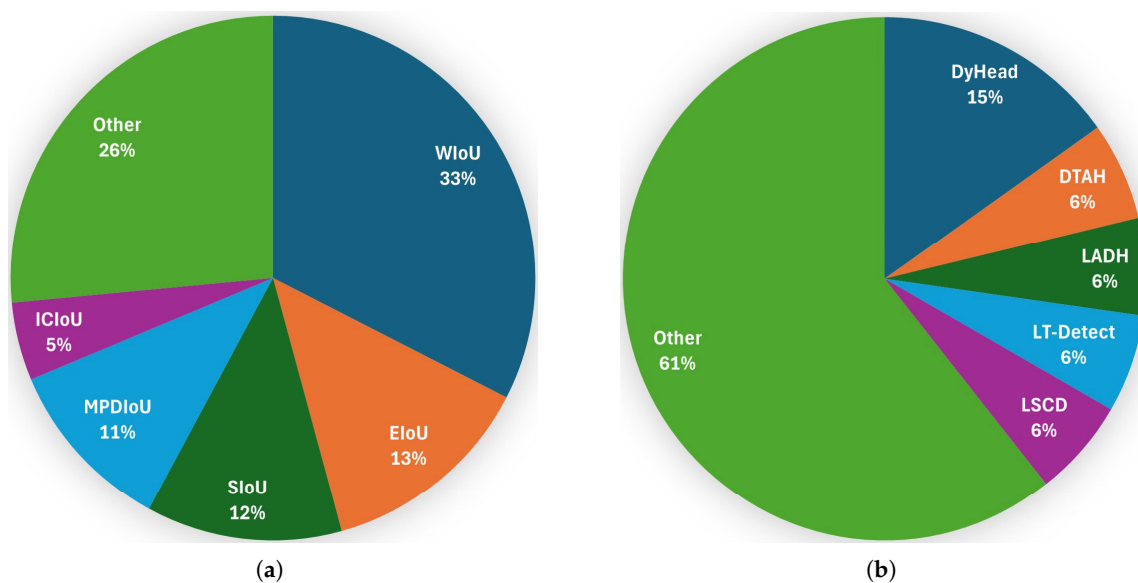
Even the upsampling module in YOLOv8 can be replaced with lightweight versions like DySample or CARAFE. These substitutions can improve the model's ability to detect important features during upsampling without adding additional parameters or computational overhead, thereby improving the network's ability to extract disease-related features [67].

In the use case in [192], replacing Usample with DySample improved all metrics, e.g., +2.78% in mAP, for the same amount of GFLOPS (8.2). Deng et al. [122] showed that using CARAFE led to a 3.9% improvement in mAP over the original YOLOv8 model for citrus color identification. In the study [195], using the CARAFE module instead of Upsample resulted in a 2.1% increase in mAP when the proposed Improved YOLOv8 algorithm was applied to the CottonWeedDet3 dataset.

## 7. Head Improvements

### 7.1. Loss Function Replacements

The design of an appropriate loss function is a critical step in enhancing the model's detection accuracy. The YOLOv8 loss function is comprised of two primary components: CIoU loss and DFL loss for object position regression and VFL loss for object class classification. The bounding box loss, in particular, has been identified as a pivotal element within the object detection loss function. The detection of small (disease/crop) features and higher convergence speed may require alternative loss functions. Options for replacing the CIoU loss function can be found in Table A1. Among the reviewed papers, the most commonly used CIoU loss replacement is WIoU, followed by EIoU and SIoU; see Figure 17.



**Figure 17.** Frequency of replacement options (a) for CIoU replacement ( $N = 83$ ); (b) for complete head substitution ( $N = 33$ ).

Zhang et al. [179] compared four different loss functions, including CIoU, Shape-IoU, Inner-IoU, and ISIoU, and showed that ISIoU leads to better performance in detecting crop growth states. As demonstrated in the comparison presented in Wu et al. [141], the YOLOv8n model employing the WIoU loss function exhibits a 3.11% higher level of detection accuracy compared to the original YOLOv8n model utilizing the CIoU loss function. This superior performance is also observed when benchmarked against all other loss functions considered.

Yao et al. [66] proposed the redesign of the detection head based on the concept of parameter sharing and an improved loss function called Normalized Wasserstein Distance (NWD). A comparison was made between the original YOLOv8s algorithm and the updated version using the same loss functions as the original model, specifically CIoU, EIou, SIou and WIou. It was found that WIou and NWD show strong performance, with NWD beating all other approaches and showing the largest improvement of 3.06% in mAP [66].

Wen et al. [86] proposed the WSIoU loss function, a combination of Wise-IoUv3 and SIoU, and compared it with eight other loss functions, including GIoU, SIoU, EIou, CIoU, DIoU, SIoU, WIouV3, and MPDIoU. The results showed that WSIoU achieved the highest detection accuracy, outperforming all other loss functions. Jin et al. [23] conducted trials on detecting rice blast and found that the YOLOv8 model performs optimally with WIouV3 as the loss function. This improved the mAP by 3.8% compared to CIoU.

### 7.2. Other Improvements in the Head

Several other head improvements have been proposed in the literature; see Table A1. The ASFF mechanism can be integrated into the original detection framework to improve the spatial consistency and detection accuracy of small-sized targets [194]. Other measures to improve the head are listed in Table A1. Among the reviewed papers, the most commonly suggested improvement to the head are the dynamic ones (DyHead, DTAH) to increase the flexibility in detecting plant parts (crops or diseased regions) of varying sizes; see Figure 17.

A significant number of researchers suggest integrating AM, such as ARFA and SEAM. On the other hand, many authors have proposed lightweight versions, which are useful for real-time critical applications. The standard YOLOv8 detection head accounts for approximately 25% of the model's parameters.

In the study of Jia et al. [170], it was concluded that DyHead contributes more significantly to the performance improvement of the soybean pod detection model. DyHead has been demonstrated to dynamically adjust the model's attention across different scales and occlusions, thereby enhancing its capacity to detect strawberries in complex, overlapping scenarios, as outlined in [107]. As with all lightweighting measures, LADH reduces the training time and the number of parameters. However, it usually increases the mAP, as shown in [208] for seedling growth prediction.

The experimental study in [63] showed that the incorporation of SEAM into the detection head resulted in a 1.1% improvement in mAP compared to the baseline model. This indicates that SEAM improves detection in complex natural scenarios with highly occluded diseased cucumber leaves [63]. The addition of an auxiliary detection head resulted in a 2.7% increase in mAP in the Fu et al. [91] study for the identification of tomato fruit. Integrating ARFA-Head optimized the performance in multiscale object detection. Specifically, it increased the mAP by 4.3% in tomato leaf disease detection, as demonstrated in Yan and Li's study [34].

## 8. Some Selected Improved YOLOv8 Architectures

We will now briefly review and assess some improved YOLOv8 architectures to demonstrate their composition and complexity as prime examples of modular YOLOv8 architectures:

- **Yao et al.'s YOLO-Wheat (Figure 18):** This wheat pest and disease identification algorithm replaces the C2f modules in the backbone with C2f-DCN modules, while also adding SCNet attention modules to the neck. Additionally, it uses the NWD loss function instead of CIoU in the

head to calculate the similarity between boxes and frames. The YOLO-Wheat model achieved an mAP of 93.28%, which is 12.47% higher than the baseline YOLOv8s model. The model's recall increased by 12.74%, while the FPS decreased by 8.99%. We expect higher performance values when the YOLO-Wheat structure is enhanced by integrating CBS replacements.

- **Shui et al.'s YOLOv8-Improved (Figure 19):** The YOLOv8-Improved model uses C2f-MLCA instead of C2f modules and BiFPN instead of PAN to enhance multiscale feature fusion. In addition, WIoU is incorporated in place of CIoU. The model performed well in the study [157] on detecting the maturity of flowering Chinese cabbage. It reached an mAP of 91.8% and a recall of 86.0%, which are 2.7% and 2.9% higher than the baseline YOLOv8 model, respectively. This model could benefit from the addition of attention modules in the neck and the replacement of CBS modules.
- **Yin et al.'s YOLO-RMD (Figure 20):** The novel YOLO-RMD model, as presented in [24] for rice pest detection, integrates C2f-RFAConv modules, replacing the C2f modules in the backbone, and adds MLCA modules between the neck and head. It also incorporates a dynamic head (DyHead), which integrates multiple types of attention across feature levels, spatial positions, and output channels [225]. Yin et al. achieved very high mAP and recall values of 98.2% and 96.2%, respectively. This represents a 3% increase in mAP and a 3.5% increase in recall compared to the baseline YOLOv8n model. However, this was accompanied by an increase in the model's parameter count and a corresponding decrease in detection speed [24]. Given this, it seems unlikely that replacing or adding more modules would significantly improve the accuracy. On the other hand, the level of accuracy of YOLO-RMD can be maintained while making it significantly lighter by integrating modules designed for this purpose. This would increase the detection speed, allowing for better deployment to edge computing devices.
- **Cao et al.'s Pyramid-YOLOv8 (Figure 21):** Pyramid-YOLOv8 was developed in [15] for rapid and accurate rice leaf blast disease detection. Throughout the entire network, the algorithm uses C2f-Pyramid modules instead of C2f modules. It also adds M-CBAM modules, which serve as a bridge between the backbone and the neck. Pyramid-YOLOv8 demonstrated good performance, achieving an mAP of 84.3% and a recall of 75.7%. This represents a 6.1% and 3.9% improvement, respectively, over the baseline YOLOv8x model. Therefore, there is still room for improvement in terms of accuracy, which can be achieved by modifying the head. On the other hand, the Pyramid-YOLOv8 model still requires significant computing resources, making it difficult to deploy on resource-constrained edge devices, as concluded in [15]. Therefore, improving the model's real-time performance and adopting a lighter network are still open areas of research.
- **Sun et al.'s Improved YOLOv8l (Figure 22):** In the enhanced YOLOv8 model by Sun et al. [71], the last two backbone C2f modules are replaced with VoV-GSCSP modules. Additionally, two SimAM attention modules are incorporated into the backbone and the neck of the original network is replaced with a BiFPN. All of this results in a complicated, nonhomogeneous model. Compared to the baseline YOLOv8l model, the Sun et al. model's parameters and GFLOPs are reduced by 52.66% and 19.9%, respectively. Meanwhile, the model's mAP improved by 1%, and its recall improved by 2.7% [71], achieving an mAP of 88.9% and a recall of 80.1%. It can be concluded that the enhanced YOLOv8 model effectively balances resource consumption and detection accuracy. However, there is still room for improvement in terms of accuracy, which can be achieved by modifying the head.
- **Wang et al.'s LSD-YOLO (Figure 23):** LSD-YOLO uses SAConv instead of CBS and C2f-SAConv instead of C2f in all positions of the backbone. Additionally, it incorporates CBAM modules throughout the entire network, including the backbone and neck. Among the reviewed architectures, it has the highest number of modules and thus builds the most complex structure. LSD-YOLO achieved an mAP of 92.89% and 88.36% for healthy and diseased lemons, respectively. Compared to the YOLOv8n baseline model, LSD-YOLO achieved an mAP of 90.62%, an improvement of 2.64%, with only a 0.34 million parameter increase [69]. Therefore, despite its

relative complexity, the LSD-YOLO model provides accurate recognition of healthy and diseased lemons without significantly increasing the computational burden. Nevertheless, there is room for improvement in the neck (more C2f replacements, CBS replacements) and head areas.

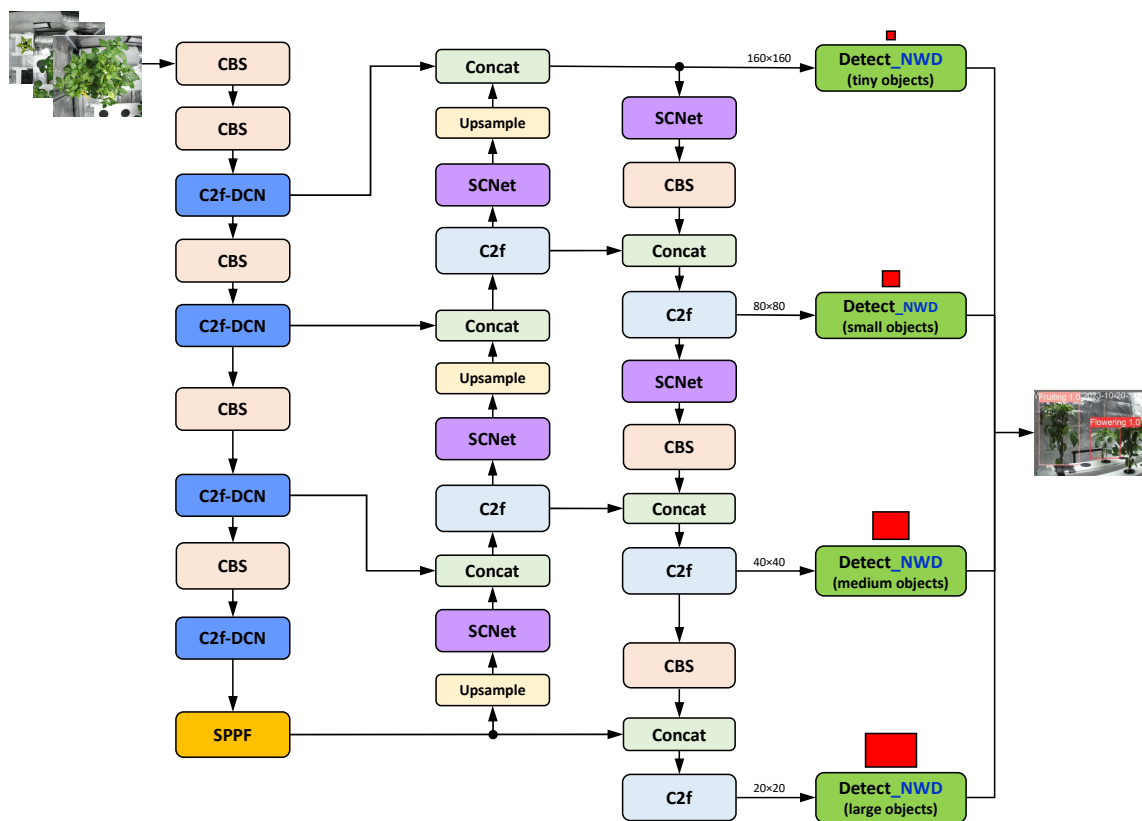


Figure 18. Structure of YOLO-Wheat proposed by Yao et al. [66]

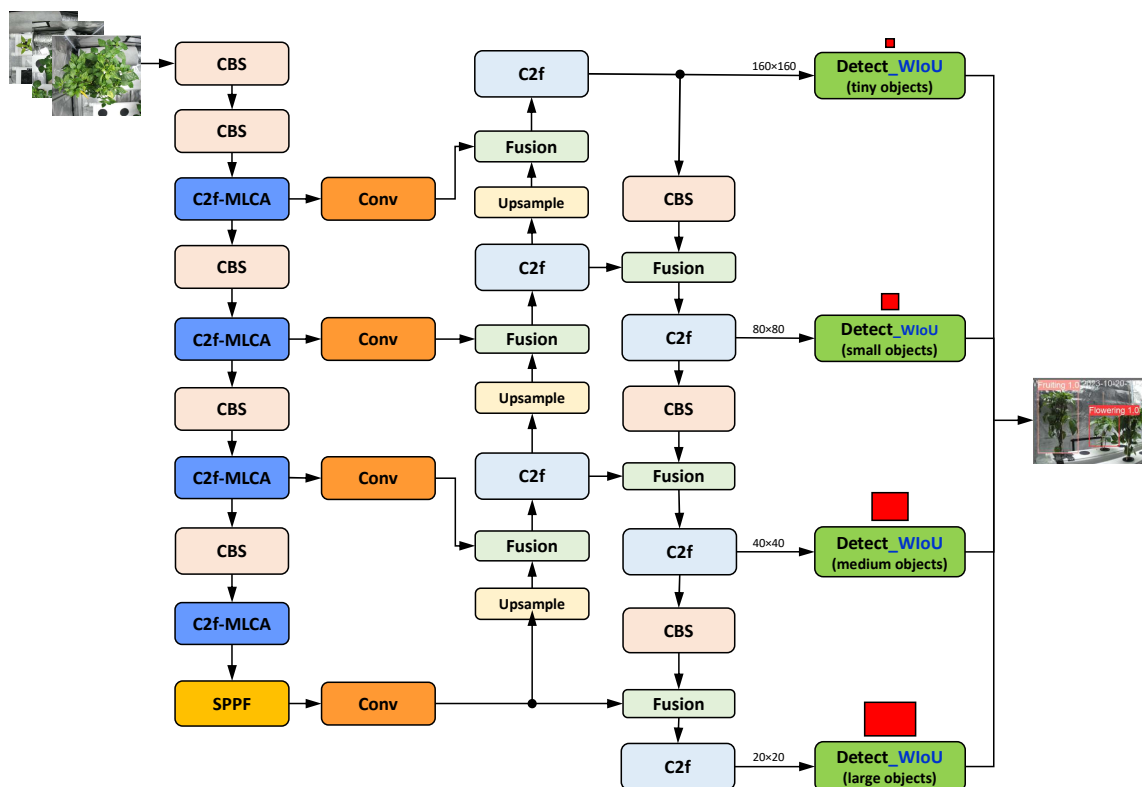


Figure 19. Structure of YOLOv8-Improved proposed by Shui et al. [157]

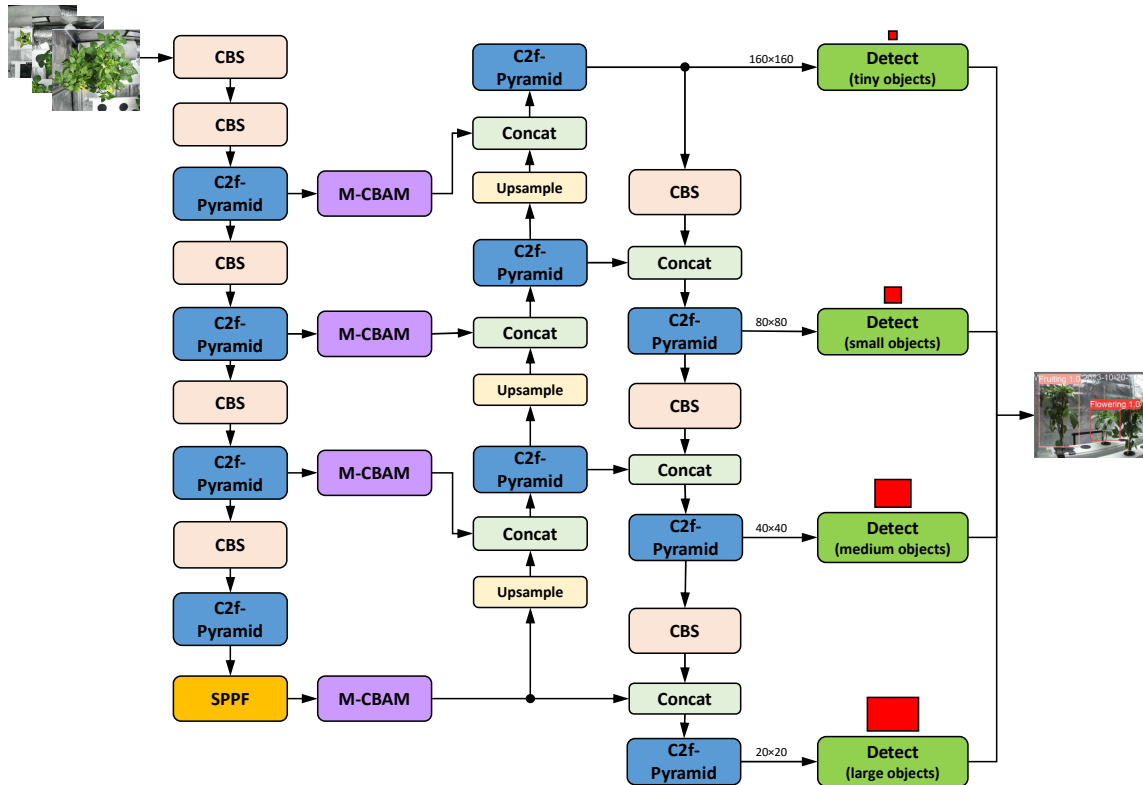


Figure 20. Structure of YOLO-RMD proposed by Yin et al. [24]

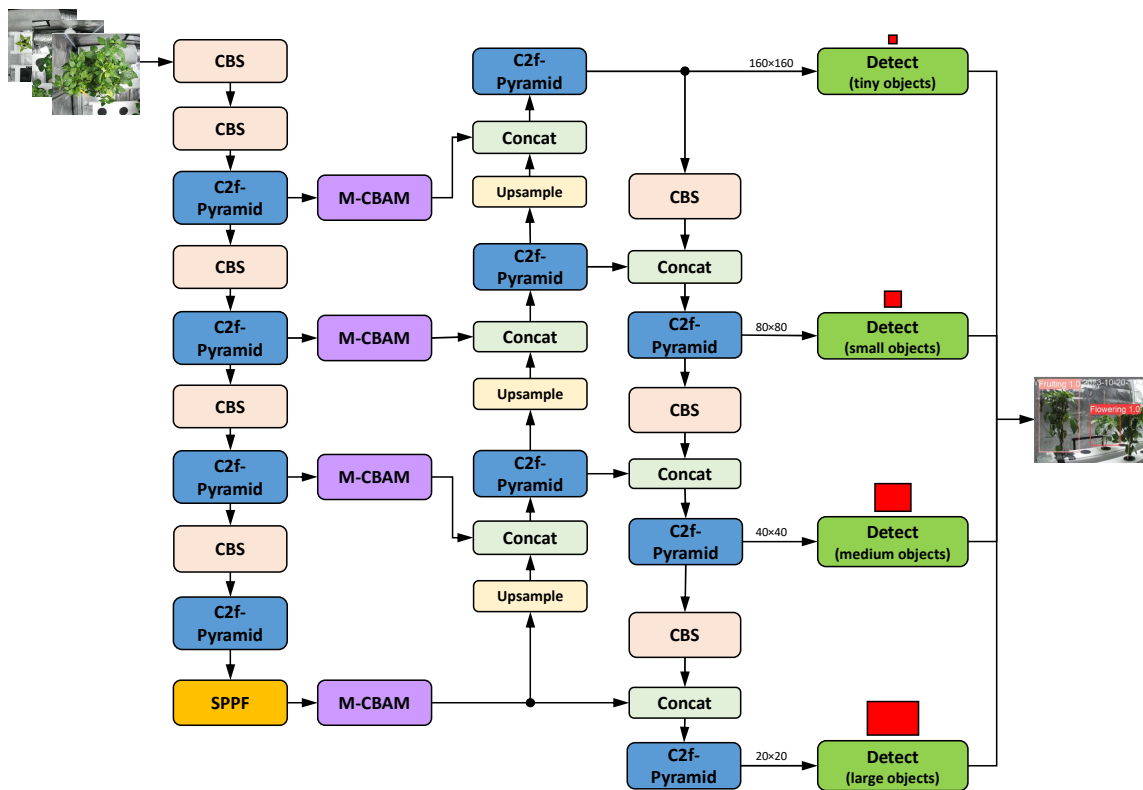


Figure 21. Structure of Pyramid-YOLOv8 proposed by Cao et al. [15]

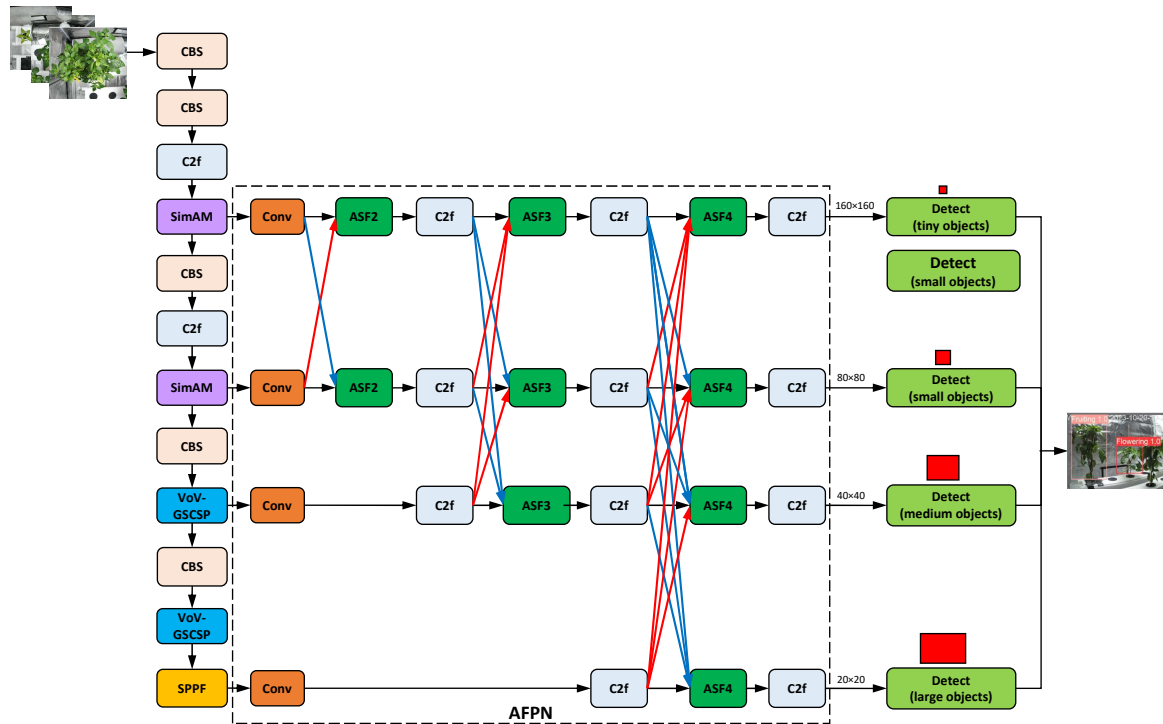


Figure 22. Structure of Improved YOLOv8l proposed by Sun et al. [71]

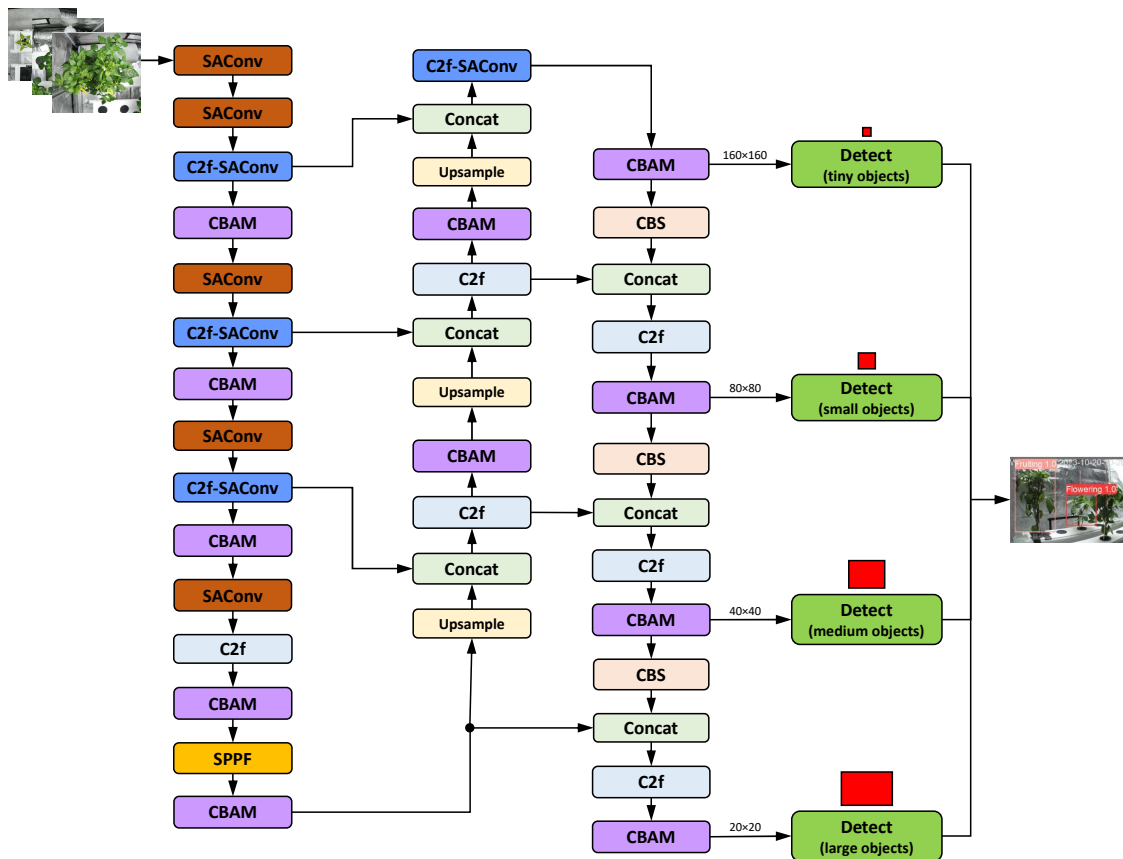


Figure 23. Structure of LSD-YOLO proposed by Wang et al. [69]

## 9. Discussion and Recommendations

Based on our comprehensive review and analysis, we offer the following recommendations for each YOLOv8 section/component that could be replaced or improved:

- **Pyramid structure:** In applications where small objects, such as berries, need to be detected, or when plant images have low resolution or contrast and need to be captured from a long or variable distance (e.g., by drones), the original YOLOv8 does not perform optimally. Small objects can easily be obscured by other objects or the background, particularly in dense scenes. Furthermore, background noise can resemble small objects, which increases the risk of false positives [226]. The incorporation of a tiny object detection layer at  $160 \times 160$  resolution (P2) has been demonstrated to augment the precision of the YOLOv8 network in detecting small and medium-sized object targets. Thus, we recommend the tiny object detection layer to be the default in modular YOLOv8 architectures.
- **Backbone/CBS:** The use of ordinary convolution operations to extract local features from an image by performing sliding operations on the input data through a convolution kernel has become standard practice for object detection tasks. However, standard convolution (Conv2d+BatchNorm2d+SiLU) presents some challenges, particularly when the receptive field size is fixed. This can limit the model's ability to capture information at different scales in the input data. Local connectivity may cause the model to overlook global information, negatively impacting performance [69]. Therefore, we suggest replacing the conventional CBS with convolution modules that adaptively adjust the sensing field according to changes in the target's scale and position within the image. These modules capture global information about the entire image and work efficiently in a wide range of contexts. SPDCConv and RFACConv are suitable options for applications where accuracy is the primary concern. In contrast, GhostConv and GSConv are the suggested options for applications where the main objective is model lightweighting that is required for embedded, edge, or mobile devices. Emerging options are SACConv and KWConv. SACConv improves object detection performance by applying different atrous rates to convolutions. It also incorporates a global context module and a weight-locking mechanism that enhances the model's ability to capture and utilize information from local and global contexts [69]. KWConv is an extension of dynamic convolution. It involves segmenting the convolutional kernel into nonoverlapping units and linearly mixing each unit using a predefined repository shared across multiple adjacent layers. KWConv replaces static convolutional kernels with sequential combinations of their respective blending results. This strengthens the dependency of convolutional parameters between the same and succeeding layers by dividing and sharing the kernels [203]. To ensure consistency and homogeneity, the CBS should be replaced by one module throughout the entire network. Alternatively, one module could be used for each section of the model (i.e., the backbone, neck, and head).
- **Backbone/C2f:** The standard C2f module in YOLOv8 uses a traditional, static convolutional computation. This method uses the same convolutional kernel to process different input information, which often leads to severe information loss [23]. However, the C2f module should be modified to adapt to the deformation of targets to be detected. This can be best achieved by replacing traditional convolution modules in C2f with dynamic ones. Based on our analysis of the reviewed papers, we recommend C2f-DCN and C2f-EMA as optimal choices for achieving higher-accuracy models. For lightweight models, good options are VoV-GSCSP, C2f-Ghost, and C2f-Faster. Emerging options are C2f-ODConv, PFEM, and OREPA. PFEM uses innovative convolution and multilevel feature fusion techniques to effectively distinguish disease spots from the background. This allows the model to maintain high detection accuracy in complex environments [34]. OREPA serves as a model structure for online convolutional reparameterization. The pipeline is composed of two stages, with the objective of reducing the training overhead by compressing complex training blocks into a single convolution [159].
- **Backbone/IM:** In the field of object detection for agricultural monitoring and harvesting tasks, deploying attention mechanisms enables networks to prioritize and allocate resources to critical areas. The analysis of the reviewed papers emphasizes the important role of these mechanisms. These mechanisms improve feature extraction, enhance model interpretability, and increase

overall detection accuracy. Beyond the prominent and widely used CBAM and GAM, we recommend EMA, and particularly SimAM. The latter is a complete 3D weight attention network that outperforms traditional 1D and 2D weight attention networks. On the other hand, CA is recommended for lightweight models to improve the accuracy of the network with minimal additional computational overhead.

- **Backbone/SPPF:** The main issue with the multi-scale feature extraction capability of SPPF is the redundancy of information in the fused feature maps [195]. Therefore, enhanced modules are needed to better fuse multiscale features and improve the ability to identify important features. We recommend using SPPELAN or SPPF-LSKA as replacement for SPPF. SPPELAN uses a combination of Spatial Pyramid Pooling (SPP) and an Efficient Local Aggregation Network (ELAN) to achieve superior feature extraction and fusion: While SPP can handle inputs of varying sizes through its multilevel pooling, which incorporates pooling operations of various sizes, ELAN improves feature utilization efficiency by establishing direct connections between different network layers[84]. The SPPF-LSKA module integrates the multiscale feature extraction functionality of the SPPF module with the long-range dependency capture capability of an LSKA mechanism. This integration enables the module to achieve more efficient and accurate feature extraction [201].
- **Neck/Architecture:** YOLOv8 uses a top-down FPN-PAN structure to fuse features at multiple scales. During the transmission of high-level semantic information, it disregards low-level details present in the feature fusion process. This approach does not take into account the unequal contributions of input features with different resolutions to the fused output features [31]. To improve the model's ability to detect objects at multiple scales, the neck network structure should be redesigned or extended so that it combines feature maps from different scales using cross-scale feature interaction mechanisms. At least, pathways need to be added from high-resolution features to low-resolution features. The most suitable and recommended options are BiFPN and AFPN, which are widely used in the reviewed literature. Emerging neck architectures are HS/LS-PAN and LFEP. The HS-FPN architecture integrates multiscale feature information to mitigate interference from complex backgrounds and enhance fruit detection accuracy. This phenomenon can be attributed to the fact that the color of immature fruit surfaces is similar to the color of the surrounding leaves and canes. The distinction between color-changing fruit and the complex background is accentuated by the disruption caused by elements such as fluctuating illumination and occlusion [156]. LFEP integrates four distinct scale feature maps through a series of upsampling and downsampling operations, thereby generating feature information at scales of 1/8, 1/16, and 1/32 for subsequent processing. During this process, the feature information is processed by a Comprehensive Multi-Kernel Module (CMKM), which contains large, medium, and small local feature processing branches [33].
- **Neck/CBS:** Beyond the prominent and widely used GhostConv, DWSCConv and GSConv for lightweight models, we recommend REAConv and DSConv for higher-accuracy models. Emerging options are MKConv and ODConv. MKConv is a multiscale convolution module that can be parametrized by a variety of configuration options. It can adapt flexibly to sampling shapes with specific data characteristics [58]. ODConv is a dynamic convolution approach that learns convolutional kernel features in parallel throughout all four dimensions of the convolutional kernel space using a unique attention mechanism. By introducing four types of attention to the accumulation kernel and applying them gradually to the respective convolution kernels, ODConv significantly improves the ability to extract features at each convolution layer [39].
- **Neck/C2f:** Based on our analysis of the reviewed papers, we recommend C2f-DCN and C2f-EMA (as well as C2f-DCN-EMA) as the best options for creating more accurate models. Good options for lightweight models are VoV-GSCSP (within Slim-Neck), C2f-Ghost, and C2f-Faster. Emerging options include OREPA, LW-C3STR, and HCMamba. In contrast to conventional transformer networks, which rely on the global attention mechanism and exhibit high computational complex-

ity, the SwinT network employs window-based attention mechanisms. This approach reduces computational complexity by confining attention computations to individual windows, thereby enhancing efficiency and reducing overheads. [152]. HCMamba is a hybrid convolutional module that integrates local detail information, extracted via convolution, with global context information provided by a state-space model-based approach. This design improves the model's ability to capture both global and detailed image features, thereby enhancing its performance in target localization and classification [31].

- **Neck/AM:** Attention mechanisms can be introduced at different positions in the neck layer: as a bridge between the backbone and the neck, only to the PAN, or at the exit of the neck, i.e., between the neck and the head. Beyond the prominent and widely used CBAM and EMA, we recommend GAM and CA. Emerging options are MLCA and TA. MLCA fully leverages the richness of features by integrating channel, spatial, local, and global information to improve feature representation and model performance. Additionally, MLCA uses a one-dimensional convolutional acceleration method to reduce computational demands and parameters [75]. TA is a new method that uses a three-branch structure (a series of rotations and arrangements) to capture cross-dimensional interactions and compute attention weights. Unlike other attention methods, TA emphasizes multidimensional interactions without reducing dimensionality and eliminates indirect correspondences between channels and weights [14].
- **Neck/Upsample:** Basic YOLOv8 uses nearest neighbor interpolation for upsampling. Although this method is simple and computationally efficient, it does not fully exploit the semantic information within the feature map for certain object detection tasks [195]. There are only two common options for Upsample replacements in improved versions of YOLOv8. We recommend using either DySample or CARAFE. Both are lightweight and efficient upsampling methods. DySample generates dynamic kernels and uses higher-resolution structures to guide the upsampling process. [192]. CARAFE considers not only the nearest neighbor points but also combines the content information of the feature map. This facilitates the attainment of more precise upsampling through the process of recombining feature map information [122].
- **Head/Detect:** The standard YOLOv8 detection head has a limited ability to handle multiscale disease features because it uses fixed receptive fields [34]. To address this problem, the introduction of dynamically adjustable receptive fields should improve detection accuracy and reduce errors caused by variations in the size of crops or disease spots. The module of choice is DyHead to substitute the original detection head module. We also suggest ARFA-Head as an emerging head version. ARFA-Head incorporates the RFACnv module, combining spatial attention with adaptive receptive fields. This allows the receptive field size to adjust dynamically and overcome the limitations of fixed receptive fields when handling complex, multiscale image inputs [34].
- **Head/CoU:** The CIoU is a key component of the bounding box prediction, as it quantifies the overlap between the predicted and ground truth boxes. The preferred function for replacing the CIoU loss function in the detection heads is the WIoU, which is widely used in the analyzed papers, followed by MPDIoU and EIoU.

These suggestions for selecting components serve as guidelines for implementing and applying advanced YOLOv8 algorithms for object detection in agricultural settings. The comprehensive analysis of numerous research papers reveals that the application of stacking modules does not yield substantial enhancements in model performance. Therefore, it is imperative to select network modules and sections that are tailored to the specific task at hand. To this end, combining and comparing different modules and subnetworks is always indispensable. To ensure consistency and homogeneity, the C2f or CBS should be replaced by one type of module throughout the entire network. Alternatively, one module type could be used for each section (i.e., backbone and neck).

## 10. Conclusions and Future Directions

In this paper, we presented the first detailed survey of the current state of object detection based on improved YOLOv8 algorithms, focusing on object detection in agricultural monitoring and harvesting tasks, and evaluated the proposed changes or improvements. We divided the improvement measures into individual sections (backbone, neck, and head) and components (CBS, C2f, AM, SPPF, Concat, Upsample, and Loss Function CIoU) of YOLOv8.

We highlighted the results of selected methods and/or comparative experiments regarding the introduced improvement or extension measures found in the reviewed literature. We provided recommendations for potential improvements to each component of YOLOv8 that could be replaced or enhanced. Based on these findings, we presented and evaluated improved YOLOv8 architectures from the reviewed literature to showcase their composition and complexity as prime examples of modular YOLOv8 architectures.

The detection of plant objects, such as crops, crop maturity, crop pose, picking points, plant growth stages, pests, and diseases in leaves or crops, is a constantly evolving area of research in agriculture and applied sciences. There are several promising future directions that researchers are exploring in this field:

1. **Automatic generation of Improved YOLOv8 configurations:** Tailored YOLOv8 configurations can be generated using optimization algorithms, such as grid search or genetic algorithms, based on objective functions to be defined for detection accuracy, parameter counts, or detection speed.
2. **Check of the performance results of the reviewed YOLOv8 algorithms:** The studies and results presented in the reviewed papers are all based on very different datasets and use partly divergent hyperparameters, etc. Therefore, the conclusions drawn in the papers are specific to the situation at hand and thus not transferable to other or similar situations. There is a great need for comparative studies of different models with different network structures on the same machine and computational environment, using the same datasets. This would provide fair and clear performance evaluations.  
Moreover, we found that some papers (marked with \*\* in Table A1) may contain incorrect or misleading DL procedures due to the fact that many groups and researchers around the world are applying AI techniques and sometimes overlooking fundamental aspects of data processing or modeling. For instance, data augmentation is a widely used technique in ML to enhance model generalization, yet its improper application—specifically, augmenting data before splitting them into training and test sets—can lead to significant data leakage and inflated performance metrics. We will address this problem in detail in an upcoming paper.
3. **Investigating recommended extensions of primary YOLOv8 architecture examples:** The further improvement suggestions for the examples of enhanced YOLOv8 architectures given in Section 8 can be implemented and investigated in real use cases and comparative studies.
4. **Investigation of transformer-based YOLOv8 algorithms:** The inclusion of transformer networks into YOLOv8 is very promising in the future. Of the studies examined, only a few consider transformer mechanisms to optimize detection performance; see Table A1. Transformer modules use the self-attention mechanism to capture global semantic information, such as long-range dependencies between fruits and stems. These modules can also effectively distinguish fruits from the background, preventing dense green leaves from being misidentified as fruits [99]. However, transformer-based models require more computational resources. We found contradictory results and conclusions regarding the integration of transformer networks and their achieved performance for YOLOv8. This motivates us to address this challenging topic in the near future.
5. **Benchmarking against other detectors:** Meanwhile, higher versions of YOLO, i.e., YOLOv9–YOLOv12 have appeared and some improvement measures have been proposed and evaluated in recent papers. Analyzing these studies can expand our survey, which focuses on YOLOv8. Claims could be strengthened by a broader benchmarking against other state-of-the-art detectors.

**Author Contributions:** Conceptualization, M.J.; Formal analysis, M.J.; Investigation, M.J.; Methodology, M.J.; Project administration, M.J.; Software Supervision, M.J.; Validation, M.J.; Visualization, M.J.; Writing - original draft, M.J.; Writing - review & editing, M.J.

**Funding:** Not applicable.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Data presented in the review are included in the article, further inquiries can be directed to the corresponding author.

**Acknowledgments:** Not applicable.

**Conflicts of Interest:** Not applicable.

## Abbreviations

This section contains the abbreviations used in the manuscript. They are listed according to the improvements or substitutions introduced in the YOLOv8 sections, as shown in the taxonomy in Figure 6.

### *Abbreviations Used for CBS Replacements Primarily Introduced for Lightweight YOLOv8 Models*

ADown	Subsampled convolution
ALSA	Attention Lightweight Subsampling
BConv	Multi-modal hybrid convolution structure combining DWConv and PConv
CBRM	Conv2d + BatchNorm2d + ReLu + MaxPool2d
DiSConv	Distributed Shift Convolution
DWConv	DepthWise Separable Convolution
EMSPConv	Efficient MultiScale PConv
GhostConv	Ghost Convolution
GSCConv	Group Shuffle Convolution
HWD	Haar Wavelet Downsampling
KWConv	KernelWarehouse dynamic convolution
PDWConv	Partial DepthWise Convolution
SCDown	Scale-Compressed Downsampling
ShNetV2-d	ShuffleNetV2-downsampling

### *Abbreviations Used for CBS Replacements Primarily Introduced for Higher-Accuracy YOLOv8 Models*

AKConv	Adaptive/Alterable Kernel Convolution
AMConv	Average Maxpooling Convolution
CGD	Context Guided Downsampling
DBD	Dual Branch Downsampling
DRFD	Deep Robust Feature Downsampling
DSConv	Dynamic Snake Convolution
FIConv	Fusion-Inception Convolution
GDC	Grouped Depthwise Convolution
GLUConv	Gated Linear Unit convolution
ODConv	Omnidimensional Dynamic Convolution
MKConv	MultiScale variable Kernel Convolution
NLAB	NonLocal Attention Block
RFAConv/RFCACConv	Receptive Field Attention Convolution/integrated with CA
RFCBAM	Receptive Field Concentration-Based Attention Module
SACConv	Switchable Atrous Convolution
SimConv	Conv2d + BatchNorm2d + ReLu
SPDConv	Spatial Depth Convolution
SRFD	Shallow Robust Feature Downsampling

*Abbreviations Used for C2f Replacements Primarily Introduced for Lightweight YOLOv8 Models*

C2f-Faster	C2f module with integrated (improved) FasterNet block
C2f/C3-Ghost	C2f/C3 with integrated Ghost convolution
C2f-Faster-EMA	C2f with integrated FasterNet block and EMA
C2f-DWSCConv	C2f with integrated Depthwise Separable Convolution
CSP	Cross-Stage Partial layer
C2f-KWConv	C2f with integrated KernelWarehouse convolution (KWConv)
C3-RepLK	C3 with integrated Large Kernel convolution network (RepLK)
C2f-RVB	C2f with integrated RepViT Block
C2f-RVB-EMA	C2f with integrated RepViT Block and EMA
C2f-Star	C2f with integrated StarBlock
C2f-DWR-DRB	C2f with integrated Dilation-Wise Residual-Dilated Reparam Block (DWR-DRB)
ConvNeXtV2	ConvNeXtV2 module
RepNCSPeLAN(4)	Re-parameterization-Net with Cross-Stage Partial (CSP) and Efficient Layer Aggregation Network (ELAN)
RepVGG	Reparameterizable VGG Block
VoV-GSCSP	VoV-Group Sparse and Cross-Stage Partial network
C2f-EPMSC	C2f with integrated Efficient Parallel MultiScale Convolutional module (EPMSC)
C2f-DRB	Enhanced C2f using Dilated Reparam Block (DRB)
CSPGNLayer	C2f with integrated Cross Stage Partial GhostNetV2 module
C2f-OREPA	C2f with integrated OREPA
GELAN	Generalized Efficient Layer Aggregation Network
C2f-LMSM	C2f with integrated Lightweight MultiScale Module (LMSM)
C2f-RFCBAM	C2f with integrated Receptive Field Concentration-Based Attention Module
CSP-PCConv	C2f with all standard convolutions replaced with PCConv blocks
C2f-SC3	C2f with custom-designed SC3 module in place of Bottleneck and GeLU in place of ReLU
OREPA	Online Convolutional Re-parameterization
PDW-Faster	C2f module with integrated FasterNet block and PDWConv

*Abbreviations Used for C2f Replacements Primarily Introduced for Higher-Accuracy YOLOv8 Models*

C2f-DCN	C2f with integrated Deformable Convolution Network (DCN)
C2f-DSCConv	C2f with integrated Dynamic Snake Convolution
C2f-EMA	C2f with integrated EMA
C2f-LSKA	C2f with integrated Large Selective Kernel Attention
C2f-SACConv	C2f with integrated Switchable Atrous Convolution
C2f-SimAM	C2f with integrated Similarity-based Attention Mechanism (SimAM)
C2f-MLCA	C2f with integrated Mixed Local Channel Attention
C2f-DWR	C2f with integrated Dilation-Wise Residual (DWR) module
C2f-RFACConv	C2f with integrated RFACConv
C2f-ODConv	C2f with integrated ODConv
C2f-UIB-IAFF	C2f with Universal Inverted Bottleneck (UIB) and Iterative Attention Feature Fusion (IAFF)
C2f-HDRAB	C2f with integrated Hybrid Dilated Residual Attention Block (HDRAB)
C2f-MobileViTv3	C2f with integrated lightweight visual converter MobileViTv3
C2f-NAM	C2f based on the Normalization-based Attention Module (NAM)
C2f-MSModule	C2f with integrated MultiScale adaptive convolution Module
C2f-Faster-GAM	C2f module with integrated FasterNet block and GAM
C2f-PDS	C2f with integrated DiSConv and PCConv
C2f-BL	C2f with integrated Balance Lightweight block
C2f-G-Ghost	C2f with integrated GPU-based Ghost convolution
C2f-DCN-EMA	C2f with integrated DCN block and EMA
C2f-Pyramid	C2f with Pyramid-Bottleneck

C3x	Simplified C2f module (used in YOLOv5)
CELAN	Condition Efficient Layer Aggregation Network
Faster-PGLU	Faster Partial Gated Linear Unit model
LW-C3STR	LW-Swin Transformer module
ShNetV2-b	ShuffleNetV2-basic
C2f-SC3	C2f with SC3 module in place of Bottleneck and GeLU in place of ReLU
C2f-ALGA	C2f with integrated Adaptive Local-Global Attention
C2f-RFEM	C2f with integrated Receptive Field Enhancement Module (RFEM)
C2f-AFEB	C2f with integrated Adaptive Feature Enhancement Block (AFEB)
C2f-ALGA	C2f with integrated Adaptive Local-Global Attention
C2f-SMCA	C2f with integrated Spatial-Multiscale Context Attention (SMCA)
C2f-MS Block	C2f with integrated MultiScale Block (MS Block)
C2f-MKMC	C2f with integrated Multi-Kernel Mixed Convolution (MKMC)
C2f-AD	C2f with integrated Attention mechanism and Deep separable convolution
C2f-GCB	C2f with integrated Global Context Block
C2f-MHSA	C2f with integrated MHSA mechanism
C2f-CDSA	C2f with integrated Cross-Domain Shared Attention (CDSA) mechanism
C2f-ConvFormer	C2f with integrated ConvFormer module
C2f-EViT	C2f with an added EfficientViT (Efficient Vision Transformer) branch
C2f-DBB	C2f with integrated Diverse Branch Block (DBB)
C2f-CAA	C2f with integrated Context Anchor Attention (CAA) mechanism
C2f-CDSA	C2f with integrated Cross-Domain Shared Attention (CDSA) mechanism
C2f-ECA	C2f with integrated ECA mechanism
C2f-AKConv	C2f with integrated AKConv
C2f-CSIB	C2f with integrated Cross-Stage Intersection Block
C2f-ECSA	C2f with integrated Efficient Channel and Spatial Attention (ECSA)
C2f-CA	C2f with integrated CA mechanism
C2f-ISAT	C2f with integrated ISAT, an attention mechanism integrating the Inverted Residual Mobile Block (IRMB) with the Spatial and Channel Synergistic Attention (SCSA) module
GCNet	Global Context Network
HCMamba	Hybrid Convolutional Mamba
PCMR	Partial Convolution Module improved with Reparameterization
PFEM	Precise Feature Enhancement Module
SwinT Block	Swin Transformer Block
ASF	Attentional Scale Fusion module
BiFormer/BRA	TransFormer with Bilevel Routing Attention (BRA)
CA	Coordinate Attention
CAA	Context Anchor Attention
CAB	Context Aggregation Block
CAFm	Convolution and Attention Fusion Module
CBAM	Convolutional Block Attention Mechanism

*Abbreviations Used for Additions of Multiscale Feature Extraction and Attention Mechanisms*

CAM	Cascaded Attention Mechanism
CoT	Contextual Transformer attention
DAM	Deformable Attention Mechanism
DCHRA	Dual Channel High-Resolution Attention
DPAG	Dual-Path Attention Gate
ECA	Efficient Channel Attention
EMA	Efficient Multiscale Attention
EPSANet	Efficient Pyramid Squeeze Attention Network

EVC	Explicit Visual Center
FEM	Feature Enhancement Module
GAM	Global Attention Mechanism
GCFM	Global Context Fusion Module
GLSA	Global to Local Spatial Aggregation
GMBAM	Gradient and Magnitude maps Block Attention Mechanism
LSKA	Large-Scale Kernel Attention
MCA	Multidimensional Collaborative Attention
MECS	Median-Enhanced Spatial and Channel attention
MHNCA	MultiHead Nonlocal Channel Attention
MHSA	MultiHead SelfAttention
MLCA	Mixed Local Channel Attention
MPCA	MultiPlexed Coordinate Attention
MS Block	MultiScale Block
MSDA	MultiScale Extended Attention
MSFA	Multi-Scale Fusion Attention
OAM	Occlusion perception Attention Module
ParNetA	Parallel Networks Attention
PSA	Pyramidal Scale Attention
RFB	Receptive Field Block
SCAA	Spatial Context-Aware Attention
SA	Shuffle Attention
SCC	SelfCalibrated Coordinate attention
SCConv	Spatial and Channel reconstruction Convolution
SCNet	Spatial and Channel attention Network
SE	Squeeze-and-Excitation attention
SEAM	Spatially Enhanced Attention Module
SE-MSDWA	SE-enhanced Multi-Scale Depthwise Attention (MSDWA)
SimAM	Similarity-Aware Attention Mechanism
TA	Triplet Attention

#### *Abbreviations Used for SPPF Replacements*

AIFI	Attention-based IntraScale Feature Interaction
BasicRFB	Basic Receptive Field Block
FM Module	Focal Modulation Module
GAM	Global Attention Mechanism
HWD-Pooling	Haar Wavelet Downsampling-based Pooling
L-SPPF	Lightweight SPPF
MDPM	MultiScale Directional Perception Module
MixSPPF	Mixed Pooling SPPF
SC3T	Fusion of the SPP and C3TR modules
SimSPPF	Simplified SPPF with CBR in place of CBS
$\alpha$ -SimSPPF	$\alpha$ -Simplified SPPF with GSConv in place of CBS
SimSPPF-LSKA	Simplified SPPF-LSKA
SPPCSPC	Spatial Pyramid Pooling Cross-Stage Partial Convolution
SPPELAN	SPPF module with ELAN
SPPF-LSKA	SPPF with LSKA
SPSP	Spatial Pyramid Squeeze Pooling

*Abbreviations Used for Neck Architectures*

AFF	Adjacent Feature Fusion
AFPN	Asymptotic FPN with Adaptive Spatial Feature Fusion (ASFF)
BFM	Bitemporal Feature Module
BiFPN	(Weighted) Bidirectional Feature Pyramid Network (FPN)
CGFM	Context-Guided Fusion Module
C-Neck	Lightweight multidimensional feature network
GD	Gather and Distribute mechanism
GFPN	Generalized FPN
HRFPN	High-Resolution FPN
HSFPN	High-level Selective FPN
HS/LS-PAN	High/Low-level (interactive) Screening PAN
LFEP	Local Feature Enhance Pyramid
MEFP	Multiscale Enhanced Feature Pyramid
MSFDNet	MultiScale Focus-Diffusion Network
MultiCat	Multiscale feature processing and fusion
RepGFPN	Reparameterized Generalized FPN

*Abbreviations Used for Upsample Replacements in the Neck Section*

DySample	Dynamic upSampling
CARAFE	Content-Aware Reassembly of Feature

*Abbreviations Used for Replacements of Loss Function (CIoU) in the Head Section*

EIoU	Efficient IoU loss
Focaler-IoU	More Focused IoU loss
FEIoU	Focal-EIoU: Combination of EIoU loss and focal loss
FSIoU	Focal SIoU
HIoU	Dynamically adjusted IoU loss
ICIoU	Inner-CIoU loss
IFCIoU	Inner-Focused CIoU loss
$\alpha$ -IoU	IoU loss based on the Box-Cox transform (Power IoU loss)
IFIoU	Inner-FocalerIoU loss
IMPDIoU	Introduction of IIoU into MPDIoU
ISIoU	Combination of Inner-IoU and Shape-IoU loss
ISIoU	Combination of Inner-IoU and Shape-IoU loss
MPDIoU	Minimum Points Distance IoU loss
NWD	Normalized Wasserstein Distance-based loss
PIoU	Point Intersection over Union loss
PolyLoss	Polynomial Loss
PIoUv2	Dynamically focused Powerful IoU loss
RepL	Repulsion Loss
SDL	Scale-based Dynamic Loss
SIoU	Shape IoU loss
WIoU	Wise-IoU (v1, v2, v3) loss
WSIoU	Combination of Wise-IoUv3 and SIoU loss

*Abbreviations Used for Head Improvement Measures*

ARFA-Head	Adaptive Receptive Field Attention (ARFA) Head
ASFF-Head	Adaptive Spatial Feature Fusion (ASFF) mechanism across all sensor heads
Aux-Head	Addition of an Auxiliary Detection Head module
Custom Head	Adding extra convolution (Conv) blocks to the head
DADet	Decoupled-Adaptive Training Sample Selection (ATSS) Detection
DBB-Head	Parameter-sharing Diverse Branch Block Head
Detect-EL	Design of a lightweight detection head, Efficient Lightweight
Detect-SEAM	Detection head including SEAM
DETR-Head	Real-Time Detection Transformer Head
DTAH	Dynamic Task-Aligned Head
DyHead	Dynamic Head
ECSA-Head	Introduction of the ECSA module into the Head
EPHead	Lightweight detection head using PConv
Ghost-Detect	Detection head with Ghost modules (Ghost-Detect)
LADH	Lightweight Asymmetric Detection Head
Lite-Detect	Lightweight Detection head
LSCD	Lightweight Shared Convolutional Detector
LSCSBD	Lightweight Shared Convolutional Separator Batch-normalization Detection head
LSDECD	Lightweight Shared Detail-Enhanced Convolution Detection head
LSK-Head	Integrating LSK attention mechanism prior to the Detect modules
LT-Detect	Partial introduction of PConv into the head structure
PDetect	Selfdesigned detector head based on PConv
SA-Detect	Self-Attention-based lightweight detection head
SGE-Head	Adding Spatial Group-wise Enhance (SGE) into the detection head
SPE-Head	Reconstructed detection SPE-Head structure with EMSPConv

## Appendix A

**Table A1.** Overview of the Improved YOLOv8 architectures—Replacements or extensions introduced compared to the baseline YOLOv8 are marked in blue; +mAP: improvement in mAP compared to the baseline model. \*: Only training and validation sets, but no test set are used. \*\*: Suspected data leakage because data augmentation is applied before the data are split into training, validation, and test sets.

Ref.	P-Layers	Backbone Improvements				Neck Improvements				Head/CIoU	+mAP
		CBS	C2f	AM	SPPF	Architec.	CBS	C2f	AM		
YOLOv8-based algorithms/networks for detection of plant diseases and pests.											
[14]	P3–P5	CBS	C2f-DCNv2	-	SPPF	BiFPN	CBS	C2f-DCNv2	-	Wiou	18.7%**
[15]	P2–P5	CBS	C2f-Pyramid	-	SPPF	FPN-PAN	CBS	C2f-Pyramid	M-CBAM	CIoU	6.1%
[16]	P3–P5	CBS	C2f-Ghost	-	SPPF	FPN-PAN	CBS	C2f-Ghost	M-CBAM	MPDIoU	2.6%
[17]	P3–P5	ADown	C2f-MLCA	-	SPPF	FPN-PAN	ADown	C2f	-	WSIoU	1.2%
[18]	P3–P5	ODConv	C2f	-	SPPF	BiFPN	CBS	C2f	-	CIoU	2.7%
[19]	P2–P5	CBS	C2f	-	SPPF	FPN-PAN	CBS	C2f-IRMB	-	CIoU	7.6%**
[20]	P3–P5	—	MobileNetV4	—	FM Mod.	FPN-PAN	CGD	C2f-AKConv	-	CIoU	3.4%
[21]	P3–P5	GhostConv	C2f-RepGhost	CBAM	SPPF	FPN-PAN	GhostConv	C2f-RepGhost	-	CIoU	4.8%
[22]	P3–P5	GhostConv	C3-Ghost	-	SPPF	FPN-PAN	GhostConv	C3-Ghost	EMA (CBAM/TA)	CIoU	2.3%
[23]	P2–P3	CBS	C2f-ODConv	-	SPPF	FPN-PAN	CBS	C2f	-	Wiou	5.2%
[24]	P2–P5	CBS	C2f-RFACConv	-	SPPF	FPN-PAN	CBS	C2f	MLCA	DyHead	3.0%**
[25]	P3–P5	GhostConv	C2f	-	SPPF	FPN-PAN	CBS	C2f	CAB	SIoU	7.4%**
[26]	P3–P5	CBS	C2f	-	SPPF	FPN-PAN	SimConv	RepVGG	-	CIoU	2.4%
[27]	P3–P5	CBS	C2f	BiFormer	HWD-P.	BiFPN	CBS	C2f	-	CIoU	22.3%*
[28]	P3–P5	CBS	C2f-LMSM	-	SPPF	FPN-PAN	CBS	C2f-LMSM	ALSA	MPDIoU	-0.6%
[29]	P3–P5	DWSCConv	C2f-Ghost	-	SPPF	FPN-PAN	DWSCConv	C2f	CBAM	CIoU	-2.0%
[30]	P3–P5	GDC	C2f	SE	SPPF	FPN-PAN	CBS	C2f	SE	CIoU	4.1%
[31]	P3–P5	CBS	C2f-SimAM/RVBE	-	SPPF	BiFPN	CBS	HC-Mamba	-	CIoU	4.8%
[32]	P3–P5	CBS	C2f-RVB	-	SPPF	FPN-PAN	CBS	C2f-RVB	-	LT-D.	-0.7%
[33]	P3–P5	—	CSWinT	—	SPPF	LFEP	ADown	C2f	-	CIoU	4.0%
[34]	P3–P5	CBS	PFEM	-	SPPF	CGFM	CBS	C2f	-	ARFA-D.	5.4%
Y[35]	P3–P5	CBS	C2f-CDSA	-	SPPF	BiFPN	CBS	C2f	-	ICIoU	0.5%
T[36]	P3–P5	CBS	C2f-EPMSC	-	SPPF	MFSNet	CBS	C2f-EPMSC	MSE	CIoU	4.2%
[37]	P3–P5	CBS	RepVGG	-	SPPF	FPN-PAN	CBS	RepVGG	CBAM	FEIoU	4.6%
[38]	P3–P5	RFCBAM	C2f-RFCBAM	-	MixSPPF	RepGFPN	AKConv	C2f	-	DyHead	5.97%

Table A1. Cont.

Ref.	P-Layers	Backbone Improvements				Architec.	Neck Improvements			Head/CIoU	+mAP
		CBS	C2f	AM	SPPF		CBS	C2f	AM		
[39]	P3-P5		— ResNet50 —		SPPF	AFPN	CBS	C2f-ODConv	-	CIoU	3.72%
[40]	P3-P5	CBS	C2f	SCConv	SPPF	FPN-PAN	CBS	C2f	EMA	CIoU	5.65%
[41]	P3-P5	CBS	GELAN	-	SPPF	FPN-PAN	CBS	MS Block	BiFormer	iMPDIoU	2.71%
[42]	P3-P5	CBS	CSPGNetLayer	-	SPPF	FPN-PAN	CBS	VoV-GSCSP	EMA	WIoU	5.08%
[43]	P3-P5	CBS	C2f-Faster-EMA	-	SPPF	FPN-PAN	GSCConv	VoV-GSCSP	DAM	EPHead	0.1%
[44]	P3-P5	DSCConv	CSP	-	SPPF	FPN-PAN	CBS	C2f	BiFormer	IF-CIoU	1.79%
[45]	P3-P5	GhostConv	C3Ghost	GAM	SPPF	BiFPN	CBS	C2f	-	CIoU	3.4%
[46]	P3-P6	CBS	C2f-RFEM	EMA	SPPF	FPN-PAN	CBS	C2f-DCN/-EMA	-	CIoU	3.0%
[47]	P3-P5	CBS	AFEB/MCB	-	SPPF	FPN-PAN	CBS	MCB	SCAA	CIoU	2.7%
[48]	P3-P5	GSCConv	C2f	-	SPPF	FPN-PAN	GSCConv	VoV-GSCSP	SA	WIoU	1.1%
[49]	P2-P5	CBS	C2f	-	SPPF	AFF	CBS	CAM	-	DATH	2.8%
[50]	P3-P5	KWConv	C2f-KWConv	TA	SPPF	FPN-PAN	KWConv	C2f-KWConv	TA	DBB-Head	2.8%
[51]	P3-P5	EMSPConv	C2f	-	SPPF	FPN-PAN	GSCConv	VoV-GSCSP	-	SPE-Head	1.3%
[52]	P3-P5	SCDown	C2f	-	SPPF	FPN-PAN	SCDown	C2f-CIB	SimAM	CIoU	4.0%
[53]	P3-P5	GhostConv	C2f	-	SC3T	FPN-PAN	GhostConv	C2f	-	SIoU	8.0%
[54]	P3-P5		— MobileNetV3 —		SPPF	BiFPN	CBS	C2f	CoT	WIoU	-1.8%
[55]	P3-P5	CBS	C2f	SimAM	SPPF	BiFPN	DWSCConv	C2f	-	CIoU	14.5%**
[56]	P3-P5	CBS	VoV-GSCSP	-	SPPF	FPN-PAN	CBS	C2f	GAM	ICIoU	3.56%
[57]	P3-P5	AMConv	C2f-SMCA	-	SPPF	FPN-PAN	AMConv	C2f-MS Block	-	CIoU	3.2%
[58]	P3-P5	MKConv	C2f-SKA	-	SPPF	FPN-PAN	MKConv	C2f	-	MPDIoU	0.83%
[59]	P3-P5	CBS	C2f-Faster	BRA	SPPF	BiFPN	CBS	C2f	-	CIoU	0.9%
[61]	P3-P5	AKConv	C2f-DWSCConv	-	SPPF	FPN-PAN	AKConv	C2f-DWSCConv	MSFA	WSIoU	1.2%
[60]	P3-P5	ADown	C2f	-	SPPF	BiFPN	ADown	C2f	GLSA	CIoU	4.3%
[62]	P3-P5	CBS	C2f-AD	-	SPPF	MultiCat	CBS	C2f-GCB	-	CIoU	3.2%
[63]	P3-P5	CBS	C2f-DCNv2	-	SPPF	FPN-PAN	CBS	C2f	-	Detect-SEAM	1.8%
[64]	P2-P5	CBS	C2f-SC3	-	SPPF	AFPN	CBS	C2f	LSKA	CIoU	3.46%
[65]	P3-P5	DiSConv	C2f	-	SPPF	BiFPN	CBS	C2f-PDS	CBAM	FEIoU	6.5%
[66]	P2-P5	CBS	C2f-DCN	-	SPPF	FPN-PAN	CBS	C2f	SCNet	NWD	12.0%
[67]	P3-P5	AKConv	C2f	-	SPPF	FPN-PAN	AKConv	C2f	CA	WIoU	3.1%
[68]	P3-P5	CBS	C2f-DCN	-	SPPF	FPN-PAN	CBS	C2f	SCNet	CIoU	7.0%
[69]	P2-P5	SACConv	C2f-SACConv	CBAM	SPPF	FPN-PAN	CBS	C2f-SACConv	CBAM	CIoU	2.64%

Table A1. Cont.

Ref.	P-Layers	Backbone Improvements				Neck Improvements				Head/CIoU	+mAP
		CBS	C2f	AM	SPPF	Architec.	CBS	C2f	AM		
[70]	P2-P5	CBS	C2f	SimAM	SPPF	BiFPN	CBS	C2f	-	EIoU	2.1%
[71]	P2-P5	CBS	C2f/VoV-GSCSP	SimAM	SPPF	AFPV	C2f	CBS	-	CIoU	1.0%
[72]	P3-P5	CBS	C2f	CBAM	SPPF	FPN-PAN	GSCConv	VoV-GSCSP	-	WIoUv3	1.0%
[73]	P3-P5	CBS	C2f-DWR	-	SPPF	MEFP	CBS	C2f-DWR	-	WIoUv3	1.8%
[74]	P3-P5	BConv	C2f	-	SPPF	FPN-PAN	CBS	C2f-BL	-	CIoU	2.3%
[75]	P3-P5	CBS	C2f-G-Ghost	MLCA	SPPF	FPN-PAN	CBS	C2f-G-Ghost	MLCA	CIoU	3.8%
[76]	P3-P5	CBS	C2f	CBAM	SPPF	FPN-PAN	CBS	C2f	-	DATH	4.4%
[45]	P3-P5	GhostConv	C3Ghost	GAM	SPPF	BiFPN	CBS	C2f	-	CIoU	3.4%
[46]	P3-P6	CBS	C2f-RFEM	EMA	SPPF	FPN-PAN	CBS	C2f-DCN/-EMA	-	CIoU	3.0%
[47]	P3-P5	CBS	AFEB/MCB	-	SPPF	FPN-PAN	CBS	MCB	SCAA	CIoU	2.7%
[48]	P3-P5	GSCConv	C2f	-	SPPF	FPN-PAN	GSCConv	VoV-GSCSP	SA	WIoU	1.1%
[49]	P2-P5	CBS	C2f	-	SPPF	AFF	CBS	CAM	-	DATH	2.8%
[50]	P3-P5	KWConv	C2f-KWConv	TA	SPPF	FPN-PAN	KWConv	C2f-KWConv	TA	DBB-Head	2.8%
[51]	P3-P5	EMSPConv	C2f	-	SPPF	FPN-PAN	GSCConv	VoV-GSCSP	-	SPE-Head	1.3%
[52]	P3-P5	SCDown	C2f	-	SPPF	FPN-PAN	SCDown	C2f-CIB	SimAM	CIoU	4.0%
[53]	P3-P5	GhostConv	C2f	-	SPPF	FPN-PAN	GhostConv	C2f	-	SIoU	8.0%
[54]	P3-P5	—	MobileNetV3	—	SPPF	BiFPN	CBS	C2f	CoT	WIoU	-1.8%
[55]	P3-P5	CBS	C2f	SimAM	SPPF	BiFPN	DWSCConv	C2f	-	CIoU	14.5%**
[56]	P3-P5	CBS	VoV-GSCSP	-	SPPF	FPN-PAN	CBS	C2f	GAM	ICIoU	3.56%
[57]	P3-P5	AMConv	C2f-SMCA	-	SPPF	FPN-PAN	AMConv	C2f-MS Block	-	CIoU	3.2%
[58]	P3-P5	MKConv	C2f-SKA	-	SPPF	FPN-PAN	MKConv	C2f	-	MPDIoU	0.83%
[59]	P3-P5	CBS	C2f-Faster	BRA	SPPF	BiFPN	CBS	C2f	-	CIoU	0.9%
[60]	P3-P5	ADown	C2f	-	SPPF	BiFPN	ADown	C2f	GLSA	CIoU	4.3%
[61]	P3-P5	AKConv	C2f-DWSCConv	-	SPPF	FPN-PAN	AKConv	C2f-DWSCConv	MSFA	WSIoU	1.2%
[62]	P3-P5	CBS	C2f-AD	-	SPPF	MultiCat	CBS	C2f-GCB	-	CIoU	3.2%
[63]	P3-P5	CBS	C2f-DCNv2	-	SPPF	FPN-PAN	CBS	C2f	-	Detect-SEAM	1.8%
[64]	P2-P5	CBS	C2f-SC3	-	SPPF	AFPV	CBS	C2f	LSKA	CIoU	3.46%
[65]	P3-P5	DiSConv	C2f	-	SPPF	BiFPN	CBS	C2f-PDS	CBAM	FEIoU	6.5%
[66]	P2-P5	CBS	C2f-DCN	-	SPPF	FPN-PAN	CBS	C2f	SCNet	NWD	12.0%
[67]	P3-P5	AKConv	C2f	-	SPPF	FPN-PAN	AKConv	C2f	CA	WIoU	3.1%
[68]	P3-P5	CBS	C2f-DCN	-	SPPF	FPN-PAN	CBS	C2f	SCNet	CIoU	7.0%
[69]	P2-P5	SACConv	C2f-SACConv	CBAM	SPPF	FPN-PAN	CBS	C2f-SACConv	CBAM	CIoU	2.64%

Table A1. Cont.

Ref.	P-Layers	Backbone Improvements				Neck Improvements				Head/CIoU	+mAP
		CBS	C2f	AM	SPPF	Architec.	CBS	C2f	AM		
[70]	P2-P5	CBS	C2f	SimAM	SPPF	BiFPN	CBS	C2f	-	ElIoU	2.1%
[71]	P2-P5	CBS	C2f/VoV-GSCSP	SimAM	SPPF	AFPN	C2f	CBS	-	CIoU	1.0%
[72]	P3-P5	CBS	C2f	CBAM	SPPF	FPN-PAN	GSCConv	VoV-GSCSP	-	WIoUv3	1.0%
[73]	P3-P5	CBS	C2f-DWR	-	SPPF	MEFP	CBS	C2f-DWR	-	WIoUv3	1.8%
[74]	P3-P5	BConv	C2f	-	SPPF	FPN-PAN	CBS	C2f-BL	-	CIoU	2.3%
[75]	P3-P5	CBS	C2f-G-Ghost	MLCA	SPPF	FPN-PAN	CBS	C2f-G-Ghost	MLCA	CIoU	3.8%
[76]	P3-P5	CBS	C2f	CBAM	SPPF	FPN-PAN	CBS	C2f	-	DATH	4.4%
[77]	P3-P5	CBS	C2f-RVB	-	SPPF	RepGFPN	CBS	CSPStage	-	CIoU	2.2%
[78]	P3-P5	CBS	C2f	-	SPPF	FPN-PAN	CBS	C2f	-	DyHead	4.1%
[79]	P2-P5	CLUConv	C2f	CBAM	SPPF	FPN-PAN	CBS	C2f	EMA	D.-LSK/SIoU	8.2%
[80]	P2-P5	CBS	C2f	-	SPPF	FPN-PAN	GhostCBS	C2fGhost	OAM	HIoU	6.46%
[81]	P2-P5	CBS	C2f	-	SPPF	FPN-PAN	CBS	C2f	SEnet	CIoU	2.1%
[82]	P3-P5	GhostConv	C2f	-	SPPF	FPN-PAN	GhostConv	C2f	CA	CIoU	3.9%
[83]	P3-P5		— DAT —	-	SPPF	FPN-PAN	CBS	C2f	CBAM	CIoU	7.47%*
[84]	P3-P5	CBS	C2f-DSCConv	-	SPPF	FPN-PAN	CBS	C2f	SViT	CIoU	3.3%
[227]	P3-P5	FIConv	C2f-SIS	-	SPPF-IS	FPN-PAN	CBS	C2f	-	CIoU	5.9%
[85]	P3-P5	CBS	C2f-MHSA	-	SPPF	BiFPN	CBS	C2f-ECA	-	CIoU	4.1%
[86]	P3-P5	ADown	GOCR-ELAN	-	SPPF	FPN-PAN	ADown	VoV-GSCSP	-	WSIoU	1.7%
[87]	P3-P5	CBS	C2f	MECS	SPPF	BFM	CBS	VoV-GSCSP	-	CIoU	3.9%
YOLOv8-based algorithms/networks for detection of fruit/crop maturity, pose, or picking point for harvesting.											
[88]	P3-P5	DWSCConv	C2f-DWSCConv	FEM	SPPF	FPN-PAN	DWSCConv	C2f-DWSCConv	DPAG	CIoU	1.5%
[89]	P3-P5	CBS	C2f	MHSA	SPPF	FPN-PAN	CBS	C2f	-	CIoU	0.5%
[90]	P3-P5	CBS	C2f	-	SPPF-LSKA	FPN-PAN	CBS	C2f	-	ICIoU	0.6%
[91]	P3-P5		— EfficientViT —	-	SPPF	FPN-PAN	CBS	C2f-Faster	-	AuxD./SIoU	7.2%
[92]	P3-P5	CBS	C2f-Faster	-	SPPF	FPN-PAN	CBS	C2f-Faster	MLCA	CIoU	2.8%
[93]	P3-P5	CBS	C2f	-	$\alpha$ -SimSPPF	FPN-PAN	GSCConv	VoV-GSCSP	SE	CIoU	2.1%
[94]	P3-P5	CBS	C2f	-	SPPF	HLIS-PAN	Conv2d	C2f	CAA/CA	CIoU	1.4%
[95]	P3-P5	CBS	C2f-NAM	-	SPPF	FPN-PAN	CBS	C2f-NAM	-	WIoU	4.3%
[96]	P3-P5	CBS	C2f-Faster	-	SPPF	GD	CBS	C2f-Faster	-	RepL	3.6%
[97]	P3-P5	CBS	C2f	-	SPPF	BiFPN	CBS	C2f	RCA-CBAM	IFIoU	5.1%

Table A1. Cont.

Ref.	P-Layers	Backbone Improvements				Architec.	Neck Improvements			Head/ CIoU	+mAP
		CBS	C2f	AM	SPPF		CBS	C2f	AM		
[98]	P3-P5	RFACConv	C2f	-	SPPF	FPN-PAN	CBS	RepNCSP-ELAN4	MHCNA	WIoU	1.7%
[99]	P3-P5	CBS	C2f-EViT	-	SPSP	FPN-PAN	CBS	C2f	-	CIoU	0.81%
[100]	P3-P5	CBS	RepNCSP-ELAN	-	SPPELAN	C-Neck	CBS	C2f	-	SGE-Head/ SloU	1.4%
[101]	P3-P5	CBS	C2f	LSKA	SPPF	BiFPN	CBS	C2f	-	CIoU	0.09%
[102]	P2-P5	—	MobileViTv3	—	SPPF	BiFPN	CBS	C2f	-	SloU	5.49%
[103]	P3-P5	CBS	ConvNeXtV2	ECA	SPPF	FPN-PAN	CBS	C2f	-	SloU	2.05%
[104]	P2-P4	SPDConv	C2f	-	SPPF	FPN-PAN	CBS	C2f	GAM	EIoU	1.6%
[105]	P3-P5	CBS	C2f	-	SPPF	FPN-PAN	CBS	LW-C3STR	-	CIoU	5.2%
[106]	P2-P5	CBS	C3	-	SPPF	FPN-PAN	CBS	C3	-	$\alpha$ -IoU	2.5%
[107]	P3-P5	CBS	C3-RepLK	-	SPPF	FPN-PAN	CBS	RepNCSP-ELAN4	-	DyHead/ PolyLoss	2.1%
[108]	P3-P5	CBS	C2f-OREPA/ C2f-DCNv3	EMA	SPPF	FPN-PAN	CBS	C2f-OREPA	-	CIoU	2.3%
[109]	P3-P5	CBS	C2f	-	GAM	FPN-PAN	CBS	C2f	-	CIoU	4.6%
[110]	P3-P5	DWConv	PDW-Faster	-	SPPF	FPN-PAN	DWConv	PDW-Faster	-	EIoU	-0.01%
[111]	P3-P5	ShNetV2-d	ShNetV2-b	SE	SPPF	FPN-PAN	GhostConv	C2f-Ghost	SE	CIoU	-4.0%
[112]	P3-P5	CBS	C2f	-	SPPF	FPN-PAN	GSCConv	VoV-GSCSP	CBAM	WIoU	1.3%
[113]	P3-P5	SPDConv	C2f	-	SPPF	FPN-PAN	CBS	C2f	GAM	WIoU	3.6%
[114]	P3-P5	CBS	C2f	SE	SPPF	FPN-PAN	CBS	C2f-DSCConv	-	CIoU	1.5%
[115]	P3-P5	CBS	C2f-Faster	-	FM Mod.	FPN-PAN	CBS	C2f	-	CIoU	1.64%
[116]	P2-P3	CBS	C2f	-	SPPF	FPN-PAN	CBS	PCMR	SCC	PolyLoss	2.9%
[117]	P3-P5	CBS	C2f	-	SPPF-LSKA	FPN-PAN	CBS	C2f	ECA	EIoU	3.0%
[118]	P2-P5	CBS	BiFormer	-	SPPF	FPN-PAN	CBS	C2f	-	EIoU	2.7%
[119]	P3-P5	CBS	C2f-Faster-EMA	-	SimSPPF-LSKA	FPN-PAN	CBS	C2f-Faster	TA	CIoU	1.73%
[120]	P3-P5	—	NextViT	—	SPPF	GCFM	ADown	C2f	-	CIoU	5.6%
[121]	P3-P5	DiSConv	C2f	GAM	SPPF	FPN-PAN	DiSConv	C2f	-	EIoU	2.0%
[122]	P3-P5	CBS	C2f	-	SPPF	FPN-PAN	GhostConv	C2f	MCA	CIoU	14.6%
[123]	P3-P5	GhostConv	C2f	CBAM	SPPF	FPN-PAN	GhostConv	C2f	-	CIoU	2.72%

Table A1. Cont.

Ref.	P-Layers	Backbone Improvements				Architec.	Neck Improvements			Head/CIoU	+mAP
		CBS	C2f	AM	SPPF		CBS	C2f	AM		
[124]	P3-P5		— RevColNet —		SPPF	FPN-PAN	CBS	RepNCSP-ELAN4/C2f-CIB	-	CIoU	5.8%
[125]	P3-P5		— HGNetV2 —		SPPF	FPN-PAN	CBS	C2f-DRB	-	D-SEAM	2.0%
[126]	P3-P5	CBS	C2f-EMSCP	-	SPPF	BiFPN	CBS	C2f	-	CIoU	1.6%
[127]	P3-P5	CBS	C2f-DCN	EMA	SPPF	FPN-PAN	CBS	C2f-DCN	EMA	CIoU	1.3%
[128]	P3-P5	DRFD	C2f-ConvFormer	-	SPPF	FPN-PAN	DRFD	C2f	-	CIoU	1.16%
[129]	P3-P5		— FasterNet —		SPPF	BiFPN	CBS	C2f	-	WIoU + MPDIoU	14.9%
[130]	P3-P5	CBS	C2f-Faster-EMA	-	SPPF	FPN-PAN	CBS	C2f-Faster	-	MPDIoU	1.85%
[131]	P3-P5	CBS	DCNv2	-	SPPF	FPN-PAN	CBS	C2f	CA	LT-D.	1.0%
[132]	P2-P5	CBS	C2f	-	SPPF	FPN-PAN	CBS	C2f	EMA	EIoU	4.2%
[133]	P2-P5	CBS	C2f-Faster-EMA	-	SPPF	BiFPN	CBS	C2f-Faster-EMA	-	MPDIoU	0.7%
[134]	P3-P5	CBS	C2f-DCNv2	-	SPPF	FPN-PAN	CBS	C2f	GAM	CIoU	3.59%
[135]	P3-P5	CBS	C2f	-	SPPCSPC	FPN-PAN	CBS	C2f	-	SIoU	7.3%
[136]	P3-P5	CBS	C2f	EPSANet	SPPF	FPN-PAN	CBS	C2f	SCCConv	ISIoU	3.4%
[137]	P3-P5	CBS	C2f	-	SPPF	BiFPN	CBS	C2f	CBAM	CIoU	4.7%
[138]	P3-P5	RFACConv	C2f	-	SPPCSPC	FPN-PAN	RFACConv	C2f-RFACConv	-	EIoU	3.6%
[139]	P3-P5	CBS	C2f	BRA	SPPF	FPN-PAN	GSCConv	VoV-GSCSP	-	DyHead	0.23%
[140]	P3-P5	SPDCConv	C2f	-	SPPF	BiFPN	CBS	C2f	-	CIoU	0.2%
[141]	P3-P5	CBS	C2f	-	SPPF	BiFPN	CBS	C2f	ASF	WIoU	3.33%
[142]	P3-P5	CBS	C2f-Faster-EMA	-	SPPF	BiFPN	CBS	C2f-Faster-EMA	-	CIoU	3.3%
[143]	P3-P5		— StarNet —		SPPF	FPN-PAN	CBS	C2f-ELC	-	D-EL/PIoUv2	0.3%
[144]	P2-P5	CBS	C2f-EMA	-	SPPF	FPN-PAN	CBS	C2f	-	WIoU	4.5%
[145]	P3-P5	CBS	C2f-Faster-EMA	-	SPPF	FPN-PAN	CBS	C2f-Faster-EMA	-	WIoU	1.89%
[146]	P2-P5		— FasterNet —		SPPF	BiFPN	CBS	C2f	-	WIoU	1.1%
[147]	P3-P5	CBS	-	-	SPPF	FPN-PAN	GSCConv	VoV-GSCSP	-	-	2.2/4.2%
[148]	P3-P5	CBS	C2f-HDRAB	-	SPPF	FPN-PAN	CBS	C2f-HDRAB	-	LSDECD	5.2%
[149]	P3-P5	CBS	C2f	-	BasicRFB	FPN-PAN	CBS	C2f	-	CIoU	0.6%
[150]	P3-P5	RFACConv	C2f	-	SPPF	FPN-PAN	HWD	C2f	-	SDL	2.6%

Table A1. Cont.

Ref.	P-Layers	Backbone Improvements				Neck Improvements				Head/ CIoU	+mAP
		CBS	C2f	AM	SPPF	Architec.	CBS	C2f	AM		
YOLOv8-based algorithms/networks for detection of plant growth stages.											
[151]	P2-P5	CBS	C2f	-	SPPELAN	FPN-PAN	CBS	C2f	GAM-CBAM	CIoU	3.6%
[152]	P3-P5		— SwinT —	-	SPPF	FPN-PAN	CBS	C2f	-	CIoU	4.03%
[153]	P3-P6	CBS	CS-PCConv	-	SPPF	FPN-PAN	CBS	CSP-PCConv	-	CIoU	-2.5%
[154]	P3-P5	RFACConv	C2f-UIB-iAFF	-	L-SPPF	FPN-PAN	CBS	C2f-UIB-iAFF	-	Ghost-D.	1.77%
[157]	P2-P5	CBS	C2f-MLCA	-	SPPF	BiFPN	CBS	C2f	-	WIoU	2.9%
[155]	P3-P5	CBS	C2f-Faster-EMA	-	SPPF	FPN-PAN	CBS	C2f-Faster	-	$\alpha$ -IoU	1.4%
[156]	P3-P5	CBS	C2f-DWR-DRB	-	SPPF	HS-PAN	Conv2d	C2f	-	CIoU	1.3%
[158]	P3-P5	CBS	C2f-DWS/SwinT Block	NLB	SPPF	FPN-PAN	CBS	C2f-DWS	-	CIoU	5.1%
[159]	P3-P5	CBS	OREPA	-	SPPF	FPN-PAN	CBS	OREPA	MPCA/SEAM	CIoU	3.4%
[160]	P2-P5	CBS	C2f-Faster	-	SPPF	FPN-PAN	CBS	C2f-Faster	SEAM	CIoU	0.4%
[161]	P2-P5	CBS	C2f-EMA	-	SPPF	AFPAN	CBS	CBS	-	CIoU	2.0%
[162]	P3-P5	RFACConv	C2f	-	SPPF	FPN-PAN	GhostConv	C2f	-	CIoU	1.9%
[163]	P3-P5	CBS	C2f	-	SPPF	FPN-PAN	CBS	C2f	CBAM	WIoUv3	0.6%
[164]	P3-P5	CBS	C2f-FasterNet	-	SPPF	FPN-PAN	CBS	C2f	ParNetA	WIoU	2.7%
[165]	P3-P4	CBS	C2f-DWR-DRB	-	SPPF	HSFPN	CBS	C2f	-	LADH	2.6%
[166]	P3-P5	CBS	GCNet	-	SPPF	FPN-PAN	DBD	C2f-SACConv	-	CIoU	2.5%
[168]	P2-P3		— ShuffleNetV2 —	-	SPPF	FPN-PAN	DWConv	C2f-DWConv	MHSA	CIoU	-1.2%
[167]	P3-P5		— ConvNeXtV2 —	-	SPPF	FPN-PAN	CBS	C2f	-	DyHead	0.6%
[169]	P3-P5	DWConv	Faster-PGLU	-	SPPF	FPN-PAN	DWConv	Faster-PGLU	-	WIoUv3	-0.7%
[170]	P3-P5	CBS	C2f-DBB	SEAM	SPPF	FPN-PAN	CBS	C2f-DBB	-	DyHead	4.9%
[171]	P3-P5	CBS	C2f	-	SPPF	CGFM	CBS	C2f	SE-MSDWA	CIoU	1.94%
[172]	P2-P5	CBS	C2f-CAA	-	SPPF	FPN-PAN	CBS	C2f	-	CIoU	22.2/ 7.7%**
[173]	P2-P5	CBS	C2f	-	SPPF	BiFPN	CBS	C2f	CA	DETR-H./Focaler-IoU	6.6%
[174]	P3-P5	CBS	C2f-DCNv2	-	SPPF	FPN-PAN	CBS	C2f-DCNv2	GAM	CIoU	1.3%
[175]	P2-P5	CBS	CELAN	GMBAM	SPPF	FPN-PAN	CBS	CSP	-	MPDIoU	3.1%

Table A1. Cont.

Ref.	P-Layers	Backbone Improvements				Architec.	Neck Improvements			Head/ CIoU	+mAP
		CBS	C2f	AM	SPPF		CBS	C2f	AM		
[176]	P3-P5	DWSCConv	C2f-Ghost	CA	SPPF	FPN-PAN	DWSCConv	C2f-Ghost	-	PDetect	0.0%
[177]	P3-P4	CBS	C2f	-	SimSPPF	FPN-PAN	CBS	C2f-LSKA	-	WIoU	1.3%
[178]	P3-P5	— VanillaNet —			SPPF	FPN-PAN	CBS	C2f-EMA	-	WIoU	2.4%
[179]	P3-P5	CBS	C2f	DCHRA	SPPF	HRRFPN	CBS	C2f	-	ISIoU	4.6%
[180]	P2-P4	SPDCConv	C2f	-	SPPF	FPN-PAN	CBS	C2f- SimAM	-	EIoU	0.36%
[181]	P3-P5	CBS	C2f	CBAM	SPPF	BiFPN	CBS	C2f- DCNv2	-	CIoU	2.1%
[182]	P5	CBS	C2f	-	SPPF	FPN-PAN	CBS	C2f	-	CIoU	0.9%
[183]	P3-P5	— MobileNetV3 —			SPPF	FPN-PAN	CBS	C2f-Faster	-	CIoU	3.8%
[184]	P3-P5	CBS	C2f	-	SPPF	FPN-PAN	CBS	C2f-ISAT	-	Focaler- IoU	2.0%
[185]	P3-P5	CBS	C2f-Faster- GAM	-	SPPF	FPN-PAN	CBS	C2f-Faster- GAM	-	WIoU	3.9%
[186]	P3-P5	CBS	C2f-Faster- EMA	-	SPPF	FPN-PAN	CBS	VoV- GSCSP	-	LSCD	2.0%
YOLOv8-based algorithms/networks for weed detection.											
[187]	P3-P5	GSCConv	C2f	-	SPPF	FPN-PAN	CBS	VoV- GSCSP	BoTNet	CIoU	3.4%
[188]	P3-P5	— EfficientNet-B0 —			SPPF	FPN-PAN	CBS	C2f	CA	FSIoU	2.5%
[189]	P3-P5	— RepViT/EfficientViT —			SPPF	FPN-PAN	CBS	C2f- DSCConv	EMA/ SimAM/ BiFormer	WIoU	6.4%**
[190]	P3-P5	— RevColNet —			SPPF	FPN-PAN	GSCConv	VoV- GSCSP	-	IMPDIoU	1.7%
[191]	P3-P5	CBS	C2f	-	SPPF	GFPN	CBS	CSP	SEAM	CIoU	1.5%
[192]	P3-P5	AKConv	C2f	-	SPPF	FPN-PAN	CBS	C2f-LSKA	-	CIoU	1.9%
[193]	P3-P5	CBS	C2f- DSCConv	-	SPPF	BiFPN	CBS	C2f	-	CIoU	2.8%
[194]	P2-P5	CBS	C2f-DWR	-	SPPF	FPN-PAN	CBS	C2f-DWR	-	ASFF-H.	2.1%
[195]	P3-P5	CBS	C2f-ALGA	-	MDPM	FPN-PAN	CBS	C2f	-	CIoU	5.1/1.3%
[196]	P3-P5	CBS	C2f	CBAM	SPPF	FPN-PAN	CBS	C2f	-	CIoU	0.3/0.1%
[197]	P3-P5	CBS	C2f- MobileViTv3	-	SPPF	BiFPN	CBS	C2f	-	MPDIoU	1.0%
[198]	P3-P5	— DS-HGNetV2 —			SPPF	BiFPN	CBS	C2f	-	Lite-D.	2.8%
[199]	P3-P5	CBS	C2f-ECSA	-	SPPF	FPN-PAN	CBS	C2f-ECSA	-	ECSA- H./ PIoU	0.0%
[200]	P3-P5	— StarNet —			SPPF	FPN-PAN	CBS	C2f-Star	LSKA	LSCSBD	0.1%

Table A1. Cont.

Ref.	P-Layers	Backbone Improvements				Architec.	Neck Improvements			Head/CIoU	+mAP
		CBS	C2f	AM	SPPF		CBS	C2f	AM		
YOLOv8-based algorithms/networks for phenotyping.											
[202]	P3-P5	CBS	C2f-DCN	-	SPPF	FPN-PAN	CBS	C2f	BRA	WIoU	9.6%
[203]	P3-P5	KWConv	C2f-KWConv	-	SPPF	FPN-PAN	KWConv	C2f-KWConv	-	CIoU	5.6%
[204]	P3-P5	CBS	C2f	-	SPPF	FPN-PAN	CBS	C2f	ECA/EVC	CIoU	1.2%
[205]	P3-P5	—	ConvNeXtV2	—	SPPF	FPN-PAN	DSCConv	C2f	MDSA	CIoU	3.95%
[206]	P3-P5	CBS	C2f	-	SPPF	FPN-PAN	CBS	C2f	CBAM	CIoU	1.77%
[207]	P3-P5	CBS	C2f-MSModule	-	SPPF	BiFPN	CBS	C2f	-	MPDIoU	6.1%**
[208]	P3-P5	CBS	C2f	-	SPPELAN	FPN-PAN	CBS	C2f	-	LADH/Focaler-IoU	0.3/0.4%
[209]	P3-P5	—	RepGhostNet	—	SPPF	FPN-PAN	CBS	C2f-CA	-	SIoU	1.0%

## References

1. Abbasi, R.; Martinez, P.; Ahmad, R. The digitization of agricultural industry – a systematic literature review on agriculture 4.0. *Smart Agricultural Technology* **2022**, *2*, 100042. <https://doi.org/10.1016/j.atech.2022.100042>.
2. Demilie, W.B. Plant disease detection and classification techniques: a comparative study of the performances. *Journal of Big Data* **2024**, *11*, 5. <https://doi.org/10.1186/s40537-023-00863-9>.
3. Jelali, M. Review of field dataset-based studies on deep learning for tomato disease and pest detection. *Frontiers Plant Science* **2024**, *15*, 1493322. <https://doi.org/10.3389/fpls.2024.1493322>.
4. Orchi, H.; Sadik, M.; Khaldoun, M.; Sabir, E. Real-Time Detection of Crop Leaf Diseases Using Enhanced YOLOv8 algorithm. *Proceedings of the International Wireless Communications and Mobile Computing (IWCMC) 2023*, pp. 1690–1696. <https://doi.org/10.1109/IWCMC58020.2023.10182573>.
5. Trigka, M.; Dritsas, E. A Comprehensive Survey of Machine Learning Techniques and Models for Object Detection. *Sensors* **2025**, *25*, 214. <https://doi.org/10.3390/s25010214>.
6. Yang, R.; Yuan, D.; Zhao, M.; Zhao, Z.; Zhang, L.; Fan, Y.; Liang, G.; Zhou, Y. *Camellia oleifera* Tree Detection and Counting Based on UAV RGB Image and YOLOv8. *Agriculture* **2024**, *14*, 1789. <https://doi.org/10.3390/agriculture14101789>.
7. Sapkota, R.; Meng, Z.; Churuvija, M.; Du, X.; Ma, Z.; Karkee, M. Comprehensive Performance Evaluation of YOLO11, YOLOv10, YOLOv9 and YOLOv8 on Detecting and Counting Fruitlet in Complex Orchard Environments. *arXiv:2407.12040v6* **2020**. <https://doi.org/10.48550/arXiv.2003.05689>.
8. Amjoud, A.B.; Amrouch, M. Object Detection Using Deep Learning, CNNs and Vision Transformers: A Review. *IEEE Access* **2023**, *11*, 35479–35516. <https://doi.org/10.1109/ACCESS.2023.3266093>.
9. Shoaib, M.; Shah, B.; El-Sappagh, S.; Ali, A.; Ullah, A.; Alenezi, F.; Gechev, T.; Hussain, T.; Ali, F. An advanced deep learning models-based plant disease detection: A review of recent research. *Frontiers Plant Science* **2023**, *14*, 1158933. <https://doi.org/10.3389/fpls.2023.1158933>.
10. Badgujar, C.M.; Poulouse, A.; Gan, H. Agricultural object detection with You Only Look Once (YOLO) Algorithm: A bibliometric and systematic literature review. *Computers and Electronics in Agriculture* **2024**, *223*, 109090. <https://doi.org/10.1016/j.compag.2024.109090>.
11. Ultralytics. YOLO by Ultralytics. <https://github.com/ultralytics/ultralytics>, 2023. visited on 1 February 2025.
12. Schneider, F.; Swiatek, J.; Jelali, M. Detection of Growth Stages of Chilli Plants in a Hydroponic Grower Using Machine Vision and YOLOv8 Deep Learning Algorithms. *Sustainability* **2024**, *16*, 6420. <https://doi.org/10.3390/su16156420>.
13. Page, M.J.; McKenzie, J.E.; Bossuyt, P.M.; Boutron, I.; Hoffmann, T.C.; Mulrow, C.D.; Shamseer, L.; Tetzlaff, J.M.; Akl, E.A.; Brennan, S.E.; et al. The Prisma 2020 Statement: An Updated Guideline for Reporting Systematic Reviews. *BMJ* **2021**, *372*, n71. <https://doi.org/10.1136/bmj.n71>.
14. Lu, Y.; Yu, J.; Zhu, X.; Zhang, B.; Sun, Z. YOLOv8-Rice: A rice leaf disease detection model based on YOLOv8. *Paddy and Water Environment* **2024**, *14*, 24492. <https://doi.org/10.1007/s10333-024-00990-w>.
15. Cao, Q.; Zhao, D.; Li, J.; Li, J.; Li, G.; Feng, S.; Xu, T. Pyramid-YOLOv8: A detection algorithm for precise detection of rice leaf blast. *Plant Methods* **2024**, *20*, 149. <https://doi.org/10.1186/s13007-024-01275-3>.
16. Yin, J.; Huang, P.; Xiao, D.; Zhang, B. A lightweight rice pest detection algorithm using improved attention mechanism and YOLOv8. *Agriculture* **2024**, *14*, 1052. <https://doi.org/10.3390/agriculture14071052>.
17. Fu, Y.; Zhang, Y. Lightweight Rice Leaf Disease Detection Method Based on Improved YOLOv8. *Engineering Letters* **2025**, *33*, 402–417.
18. Liu, G.; Di, J.; Wang, Q.; Zhao, Y.; Yang, Y. An Enhanced and Lightweight YOLOv8-Based Model for Accurate Rice Pest Detection. *IEEE Access* **2025**, *13*, 91046–91064. <https://doi.org/10.1109/ACCESS.2025.3569819>.
19. Li, Z.; Wu, W.; Wei, B.; Li, H.; Zhan, J.; Deng, S.; Wang, J. Rice Disease Detection: TLI-YOLO Innovative Approach for Enhanced Detection and Mobile Compatibility. *Sensors* **2025**, *25*, 2494. <https://doi.org/10.3390/s25082494>.
20. Wang, B.; Zhou, H.; Xie, H.; Chen, R. Identification of rice disease based on MFAC-YOLOv8. *Journal of Real-Time Image Processing* **2025**, *22*, 75. <https://doi.org/10.1007/s11554-025-01661-7>.
21. Wang, J.; Ma, S.; ZhentaoWang.; Ma, X.; Yang, C.; Chen, G.; Wang, Y. Improved Lightweight YOLOv8 Model for Rice Disease Detection in Multi-Scale Scenarios. *Agronomy* **2025**, *15*, 445. <https://doi.org/10.3390/agronomy15020445>.
22. Bui, T.D.; Le, T.M.D. Ghost-Attention-YOLOv8: Enhancing Rice Leaf Disease Detection with Lightweight Feature Extraction and Advanced Attention Mechanisms. *AgriEngineering* **2025**, *7*, 93. <https://doi.org/10.3390/agriengineering7040093>.

23. Jin, S.; Cao, Q.; Li, J.; Wang, X.; Li, J.; Fenga, S.; Xua, T. Study on lightweight rice blast detection method based on improved YOLOv8. *Pest Management Science* **2025**. <https://doi.org/10.1002/ps.8790>.
24. Yin, J.; Zhu, J.; Chen, G.; Jiang, L.; Zhan, H.; Deng, H.; Long, Y.; Lan, Y.; Wu, B.; Xu, H. An Intelligent Field Monitoring System Based on Enhanced YOLO-RMD Architecture for Real-Time Rice Pest Detection and Management. *Agriculture* **2025**, *15*, 798. <https://doi.org/10.3390/agriculture15080798>.
25. Li, F.; Lu, Y.; Ma, Q.; Yin, S.; Zhao, R. GhostConv+CA-YOLOv8n: a lightweight network for rice pest detection based on the aggregation of low-level features in real-world complex backgrounds. *Frontiers in Plant Science* **2025**, *16*, 1620339. <https://doi.org/10.3389/fpls.2025.1620339>.
26. Yue, X.; Qi, K.; Na, X.; Zhang, Y.; Liu, Y.; Liu, C. Improved YOLOv8-Seg Network for Instance Segmentation of Healthy and Diseased Tomato Plants in the Growth Stage. *Agriculture* **2023**, *13*, 1643. <https://doi.org/10.3390/agriculture13081643>.
27. Zhan, B.; Xiong, X.; Li, X.; Luo, W. BHC-YOLOV8: Improved YOLOv8-based BHC target detection model for tea leaf disease and defect in real-world scenarios. *Frontiers in Plant Science* **2024**, *15*, 1492504. <https://doi.org/10.3389/fpls.2024.1492504>.
28. Liu, W.; Bai, C.; Tang, W.; Xia, Y.; Kang, J. A lightweight real-time recognition algorithm for tomato leaf disease based on improved YOLOv8. *Agronomy* **2024**, *14*, 2069. <https://doi.org/10.3390/agronomy14092069>.
29. Kang, R.; Huang, J.; Zhou, X.; Ren, N.; Sun, S. Toward Real Scenery: A Lightweight Tomato Growth Inspection Algorithm for Leaf Disease Detection and Fruit Counting. *Plant Phenomics* **2024**, *6*, 0146. <https://doi.org/10.34133/plantphenomics.0174>.
30. Shen, Y.; Yang, Z.; Khan, Z.; Liu, H.; Chen, W.; Duan, S. Optimization of Improved YOLOv8 for Precision Tomato Leaf Disease Detection in Sustainable Agriculture. *Sensors* **2025**, *25*, 1398. <https://doi.org/10.3390/s25051398>.
31. Liu, Z.; Guo, X.; Zhao, T.; Liang, S. YOLO-BSMamba: A YOLOv8s-Based Model for Tomato Leaf Disease Detection in Complex Backgrounds. *Agronomy* **2025**, *15*, 870. <https://doi.org/10.3390/agronomy15040870>.
32. He, Z.; Tong, M. LT-YOLO: A Lightweight Network for Detecting Tomato Leaf Diseases. *Computers, Materials & Continua* **2025**, *82*, 4301–4317. <https://doi.org/10.32604/cmc.2025.060550>.
33. Sun, H.; Fu, R.; Wang, X.; Wu, Y.; Al-Absi, M.A.; Cheng, Z.; Chen, Q.; Sun, Y. Efficient deep learning-based tomato leaf disease detection through global and local feature fusion. *BMC Plant Biology* **2025**, *25*, 311. <https://doi.org/10.1186/s12870-025-06247-w>.
34. Yan, C.; Li, H. CAPNet: Tomato leaf disease detection network based on adaptive feature fusion and convolutional enhancement. *Multimedia Systems* **2025**, *31*, 178. <https://doi.org/https://doi.org/10.1007/s00530-025-01756-y>.
35. Mo, H.; Wei, L. Tomato yellow leaf curl virus detection based on cross-domain shared attention and enhanced BiFPN. *Ecological Informatics* **2025**, *85*, 102912. <https://doi.org/10.1016/j.ecoinf.2024.102912>.
36. Wang, R.; Chen, Y.; Liang, F.; Mou, X.; Zhang, G.; Jin, H. TomaFDNet: A multiscale focused diffusion-based model for tomato disease detection. *Frontiers in Plant Science* **2025**, *16*, 1530070. <https://doi.org/10.3389/fpls.2025.1530070>.
37. Wang, J.; Li, M.; Han, C.; Guo, X. YOLOv8-RCAA: A lightweight and high-performance network for tea leaf disease detection. *Agriculture* **2024**, *14*, 1240. <https://doi.org/10.3390/agriculture14081240>.
38. Ye, R.; Shao, G.; He, Y.; Gao, Q.; Li, T. YOLOv8-RMDA: Lightweight YOLOv8 Network for Early Detection of Small Target Diseases in Tea. *Sensors* **2024**, *24*, 2896. <https://doi.org/10.3390/s24092896>.
39. Ye, R.; Shao, G.; Yang, Z.; Sun, Y.; Gao, Q.; Li, T. Detection Model of Tea Disease Severity under Low Light Intensity Based on YOLOv8 and EnlightenGAN. *Plants* **2024**, *13*, 1377. <https://doi.org/10.3390/plants13101377>.
40. Li, H.; Yuan, W.; Xia, Y.; Wang, Z.; He, J.; Wang, Q.; Zhang, S.; Li, L.; Yang, F.; Wang, B. YOLOv8n-WSE-Pest: A Lightweight Deep Learning Model Based on YOLOv8n for Pest Identification in Tea Gardens. *Applied Sciences* **2024**, *14*, 8748. <https://doi.org/10.3390/app14198748>.
41. Ye, R.; Gao, Q.; Qian, Y.; Sun, J.; Li, T. Improved YOLOv8 and SAHI model for the collaborative detection of small targets at the micro scale: A case study of pest detection in tea. *Agronomy* **2024**, *14*, 1034. <https://doi.org/10.3390/agronomy14051034>.
42. Li, X.; Zhang, T.; Yu, M.; Yan, P.; HaoWang.; Dong, X.; Wen, T.; Xie, B. A YOLOv8-based method for detecting tea disease in natural environments. *Agronomy Journal* **2025**, *117*, e70043. <https://doi.org/10.1002/agj2.70043>.
43. Song, J.; Zhang, Y.; Lin, S.; Han, H.; Yu, X. TLDDM: An Enhanced Tea Leaf Pest and Disease Detection Model Based on YOLOv8. *Agronomy* **2025**, *15*, 727. <https://doi.org/10.3390/agronomy15030727>.

44. Li, T.; Zhang, L.; Lin, J. Precision agriculture with YOLO-Leaf: Advanced methods for detecting apple leaf diseases. *Frontiers Plant Science* **2024**, *15*, 1452502. <https://doi.org/10.3389/fpls.2024.1452502>.
45. Gao, L.; Zhao, X.; Yue, X.; Yue, Y.; Wang, X.; Wu, H.; Zhang, X. A Lightweight YOLOv8 Model for Apple Leaf Disease Detection. *Applied Sciences* **2024**, *14*, 6710. <https://doi.org/10.3390/app14156710>.
46. Huo, S.; Duan, N.; Xu, Z. An improved multi-scale YOLOv8 for apple leaf dense lesion detection and recognition. *IET Image Processing* **2024**, *18*, 4913–4927. <https://doi.org/10.1049/ipr2.13223>.
47. Yan, C.; Yang, K. FSM-YOLO: Apple leaf disease detection network based on adaptive feature capture and spatial context awareness. *Digital Signal Processing* **2024**, *155*, 104770. <https://doi.org/10.1016/j.dsp.2024.104770>.
48. Zeng, W.; Pang, J.; Ni, K.; Peng, P.; Hu, R. Apple Leaf Disease Detection Based on Lightweight YOLOv8-GSSW. *Applied Engineering in Agriculture* **2024**, *40*, 589–598. <https://doi.org/10.13031/aea.16022>.
49. Zhang, S.; Wang, J.; Yang, K.; Guan, M. YOLO-ACT: An adaptive crosslayer integration method for apple leaf disease detection. *Frontiers in Plant Science* **2024**, *15*, 1551794. <https://doi.org/10.3389/fpls.2024.1451078>.
50. He, Y.; Peng, Y.; Wei, C.; Zheng, Y.; Yang, C.; Zou, T. Automatic Disease Detection from Strawberry Leaf Based on Improved YOLOv8. *Plants* **2024**, *13*, 2556. <https://doi.org/10.3390/plants13182556>.
51. Wen, G.; Li, M.; Luo, Y.; Shi, C.; Tan, Y. The improved YOLOv8 algorithm based on EMSPConv and SPE-head modules. *Multimedia Tools and Applications* **2024**, *83*, 61007–61023. <https://doi.org/10.1007/s11042-023-17957-4>.
52. Chen, M.; Zou, W.; Niu, X.; Fan, P.; Liu, H.; Li, C.; Zhai, C. Improved YOLOv8-Based Segmentation Method for Strawberry Leaf and Powdery Mildew Lesions in Natural Backgrounds. *Agronomy* **2025**, *15*, 525. <https://doi.org/10.3390/agronomy15030525>.
53. Do, M.T.; Ha, M.H.; Nguyen, D.C.; Chen, O.T.C. Toward improving precision and complexity of transformer-based cost-sensitive learning models for plant disease detection. *Frontiers in Computer Science* **2025**, *6*, 148048. <https://doi.org/10.3389/fcomp.2024.1480481>.
54. Wu, E.; Ma, R.; Dong, D.; Zhao, X. D-YOLO: A Lightweight Model for Strawberry Health Detection. *Agriculture* **2025**, *15*, 570. <https://doi.org/10.3390/agriculture15060570>.
55. Wang, P.; Tan, J.; Yang, Y.; Zhang, T.; Wu, P.; Tang, X.; Li, H.; He, X.; Chen, X. Efficient and accurate identification of maize rust disease using deep learning model. *Frontiers in Plant Science* **2025**, *15*, 1490026. <https://doi.org/10.3389/fpls.2024.1490026>.
56. Yang, S.; Yao, J.; Teng, G. Corn Leaf Spot Disease Recognition Based on Improved YOLOv8. *Agriculture* **2024**, *14*, 666. <https://doi.org/10.3390/agriculture14050666>.
57. Deng, L.; Fang, D.; Ullah, A.; Hou, Q.; Yu, H. AMS-YOLO: multi-scale feature integration for intelligent plant protection against maize pests. *Frontiers in Plant Science* **2025**, *16*, 1640405. <https://doi.org/10.3389/fpls.2025.1640405>.
58. Meng, Y.; Zhan, J.; Li, K.; Yan, F.; Zhang, L. A rapid and precise algorithm for maize leaf disease detection based on YOLO MSM. *Scientific Reports* **2025**, *15*, 6016. <https://doi.org/10.1038/s41598-025-88399-1>.
59. Luo, D.; Xue, Y.; Deng, X.; Yang, B.; Chen, H.; Mo, Z. Citrus diseases and pests detection model based on self-attention YOLOv8. *IEEE Access* **2023**, *11*, 139872–139881. <https://doi.org/10.1109/ACCESS.2023.3340148>.
60. Dai, Q.; Xiao, Y.; Lv, S.; Song, S.; Xue, X.; Liang, S.; Huang, Y.; Li, Z. YOLOv8-GABNet: An enhanced lightweight network for the high-precision recognition of citrus diseases and nutrient deficiencies. *Agriculture* **2024**, *14*, 1964. <https://doi.org/10.3390/agriculture14111964>.
61. Zheng, Z.; Zhang, Y.; Sun, L. Improved YOLOv8-Based Algorithm for Citrus Leaf Disease Detection. *IEEE Access* **2025**, *13*, 105888–105900. <https://doi.org/10.1109/ACCESS.2025.3579074>.
62. Ding, J.Y.; Jeon, W.S.; Sang-YongRhee; Zou, C.M. An Improved YOLO Detection Approach for Pinpointing Cucumber Diseases and Pests. *Computers, Materials & Continua* **2024**, *81*, 3990–4013. <https://doi.org/10.32604/cmc.2024.057473>.
63. Xie, J.; Xie, X.; Xie, W.; Xie, Q. An Improved YOLOv8-Based Method for Detecting Pests and Diseases on Cucumber Leaves in Natural Backgrounds. *Sensors* **2025**, *25*, 1551. <https://doi.org/10.3390/s25051551>.
64. Sun, C.; Azman, A.; Wang, Z.; Gao, X.; Ding, K. YOLO-UP: A High-Throughput Pest Detection Model for Dense Cotton Crops Utilizing UAV-Captured Visible Light Imagery. *IEEE Access* **2024**, *11*, 19937–19945. <https://doi.org/10.1109/ACCESS.2025.3529873>.
65. Feng, H.; Chen, X.; Duan, Z. LCDDN-YOLO: Lightweight Cotton Disease Detection in Natural Environment, Based on Improved YOLOv8. *Agriculture* **2025**, *15*, 421. <https://doi.org/10.3390/agriculture15040421>.
66. Yao, X.; Yang, F.; Yao, J. YOLO-Wheat: A wheat disease detection algorithm improved by YOLOv8s. *IEEE Access* **2024**, *12*, 133877–133888. <https://doi.org/10.1109/ACCESS.2024.3460394>.

67. Chen, Z.; Feng, J.; Zhu, K.; Yang, Z.; Wang, Y.; Ren, M. YOLOv8-ACCW: Lightweight grape leaf disease detection method based on improved YOLOv8. *IEEE Access* **2024**, *12*, 123595–123608. <https://doi.org/10.1109/ACCESS.2024.3453379>.
68. Uddin, M.S.; Mazumder, M.K.A.; Prity, A.J.; Mridha, M.F.; Alfarhood, S.; Safran, M.; Che, D. Cauli-Det: Enhancing cauliflower disease detection with modified YOLOv8. *Frontiers in Plant Science* **2024**, *15*, 1373590. <https://doi.org/10.3389/fpls.2024.1373590>.
69. Wang, S.; Li, Q.; Yang, T.; Li, Z.; Bai, D.; Tang, C.; Pu, H. LSD-YOLO: Enhanced YOLOv8n Algorithm for Efficient Detection of Lemon Surface Diseases. *Plants* **2024**, *13*, 2069. <https://doi.org/10.3390/plants13152069>.
70. Zhong, M.; Li, Y.; Gao, Y. Research on Small-Target Detection of Flax Pests and Diseases in Natural Environment by Integrating Similarity-Aware Activation Module and Bidirectional Feature Pyramid Network Module Features. *Agronomy* **2025**, *15*, 187. <https://doi.org/10.3390/agronomy15010187>.
71. Sun, D.; Zhang, K.; Zhong, H.; Xie, J.; Xue, X.; Yan, M.; Wu, W.; Li, J. Efficient Tobacco Pest Detection in Complex Environments Using an Enhanced YOLOv8 Model. *Agriculture* **2024**, *14*, 353. <https://doi.org/10.3390/agriculture14030353>.
72. Jiang, T.; Chen, S. A Lightweight Forest Pest Image Recognition Model Based on Improved YOLOv8. *Applied Sciences* **2024**, *14*, 1941. <https://doi.org/10.3390/app14051941>.
73. Chen, C.; Lu, X.; He, L.; Xu, R.; Yang, Y.; Qiu, J. Research on soybean leaf disease recognition in natural environment based on improved Yolov8. *Frontiers in Plant Science* **2025**, *16*, 1523633. <https://doi.org/10.3389/fpls.2025.1523633>.
74. Chen, H.; Zhai, H.; Hu, J.; Chen, H.; Wen, C.; Feng, Y.; Wang, K.; Li, Z.; Wang, G. YOLOv8-RBean: Runner Bean Leaf Disease Detection Model Based on YOLOv8. *Agronomy* **2025**, *15*, 944. <https://doi.org/10.3390/agronomy15040944>.
75. Lin, S.; Ji, T.; Wang, J.; Li, K.; Lu, F.; Ma, C.; Gao, Z. BFWSD: A lightweight algorithm for banana fusarium wilt severity detection via UAV-Based large-scale monitoring. *Smart Agricultural Technology* **2025**, *11*, 101047. <https://doi.org/10.1016/j.atech.2025.101047>.
76. Raj, A.; Dawale, M.; Wayal, S.; Khandagale, K.; Bhangare, I.; Banerjee, S.; Gajarushi, A.; Velmurugan, R.; Baghini, M.S.; Gawande, S. YOLO-ODD: An improved YOLOv8s model for onion foliar disease detection. *Frontiers in Plant Science* **2025**, *16*, 1551794. <https://doi.org/10.3389/fpls.2025.1551794>.
77. Zheng, X.; Shao, Z.; Chen, Y.; Zeng, H.; Chen, J. MSPB-YOLO: High-Precision Detection Algorithm of Multi-Site Pepper Blight Disease Based on Improved YOLOv8. *Agronomy* **2025**, *15*, 839. <https://doi.org/10.3390/agronomy15040839>.
78. Li, Z.; Sun, J.; Yang, Y.; Chen, J.; Yu, Q.; Zhou, Z.; Yang, Y.; Yin, T.; Zhang, H.; Qian, Y. ADQ-YOLOv8m: a precise detection model of sugarcane disease in complex environment. *Frontiers in Plant Science* **2025**, *16*, 1669825. <https://doi.org/10.3389/fpls.2025.1669825>.
79. Yue, G.; Liu, Y.; Niu, T.; Liu, L.; An, L.; Wang, Z.; Duan, M. GLU-YOLOv8: An improved pest and disease target detection algorithm based on YOLOv8. *Forests* **2024**, *15*, 1486. <https://doi.org/10.3390/f15091486>.
80. Wang, X.; Liu, J. Vegetable disease detection using an improved YOLOv8 algorithm in the greenhouse plant environment. *Scientific Reports* **2024**, *14*, 4261. <https://doi.org/10.1038/s41598-024-54540-9>.
81. Han, R.; Shu, L.; Li, K. A Method for Plant Disease Enhance Detection Based on Improved YOLOv8. *EEE 33rd International Symposium on Industrial Electronics (ISIE)* **2024**, pp. 1–6. <https://doi.org/10.1109/ISIE54533.2024.10595696>.
82. Chin, P.W.; Ng, K.W.; Palanichamy, N. Plant Disease Detection and Classification Using Deep Learning Methods: A Comparison Study. *Journal of Informatics and Web Engineering* **2024**, *3*, 155–168. <https://doi.org/10.33093/jiwe.2024.3.1.10>.
83. Liu, J.; Wang, X.; Chen, Q. A Lightweight Framework for Protected Vegetable Disease Detection in Complex Scenes. *Food Science & Nutrition* **2025**, *13*, e70200. <https://doi.org/10.1002/fsn3.7020>.
84. Miao, Y.; Meng, W.; Zhou, X. SerpensGate-YOLOv8: An enhanced YOLOv8 model for accurate plant disease detection. *Frontiers in Plant Science* **2024**, *15*, 1514832. <https://doi.org/10.3389/fpls.2024.1514832>.
85. Wang, Y.; Wang, Y.; Mu, J.; Mustafa, G.R.; Wu, Q.; Wang, Y.; Zhao, B.; Zhao, S. Enhanced multiscale plant disease detection with the PYOLO model innovations. *Scientific Reports* **2025**, *15*, 5179. <https://doi.org/10.1038/s41598-025-89034-9>.
86. Wen, G.; Li, M.; Tan, Y.; Shi, C.; Luo, Y.; Luo, W. Enhanced YOLOv8 algorithm for leaf disease detection with lightweight GOCR-ELAN module and loss function: WSIoU. *Computers in Biology and Medicine* **2025**, *186*, 109630. <https://doi.org/10.1016/j.combiomed.2024.109630>.

87. Yu, C.; Xie, J.; Tony, F.J.A. BGM-YOLO: An accurate and efficient detector for detecting plant disease. *PLoS One* **2025**, *20*, e0322750. <https://doi.org/10.1371/journal.pone.0322750>.
88. Yang, G.; Wang, J.; Nie, Z.; Yang, H.; Yu, S. A lightweight YOLOv8 tomato detection algorithm combining feature enhancement and attention. *Agronomy* **2023**, *13*, 1824. <https://doi.org/10.3390/agronomy13071824>.
89. Li, P.; Zheng, J.; Li, P.; Long, H.; Li, M.; Gao, L. Tomato Maturity Detection and Counting Model Based on MHSA-YOLOv8. *Sensors* **2023**, *23*, 6701. <https://doi.org/10.3390/s23156701>.
90. Zheng, S.; Jia, X.; He, M.; Zheng, Z.; Lin, T.; Weng, W. Tomato Recognition Method Based on the YOLOv8-Tomato Model in Complex Greenhouse Environments. *Agronomy* **2024**, *14*, 1764. <https://doi.org/10.3390/agronomy14081764>.
91. Fu, Y.; Li, W.; Li, G.; Dong, Y.; Wang, S.; Zhang, Q.; Li, Y.; Dai, Z. Multi-stage tomato fruit recognition method based on improved YOLOv8. *Frontiers in Plant Science* **2024**, *15*, 1447263. <https://doi.org/10.3389/fpls.2024.1447263>.
92. Song, G.; Wang, J.; Ma, R.; Shi, Y.; Wang, Y. Study on the fusion of improved YOLOv8 and depth camera for bunch tomato stem picking point recognition and localization. *Frontiers in Plant Science* **2024**, *15*, 1447855. <https://doi.org/10.3389/fpls.2024.1447855>.
93. Sun, X. Enhanced tomato detection in greenhouse environments: A lightweight model based on S-YOLO with high accuracy. *Frontiers Plant Science* **2024**, *15*, 1451018. <https://doi.org/10.3389/fpls.2024.1451018>.
94. Wu, M.; Lin, H.; Shi, X.; Zhu, S.; Zheng, B. MTS-YOLO: A Multi-Task Lightweight and Efficient Model for Tomato Fruit Bunch Maturity and Stem Detection. *Horticulturae* **2024**, *10*, 1006. <https://doi.org/10.3390/horticulturae10091006>.
95. Wang, A.; Qian, W.; Li, A.; Xu, Y.; Hu, J.; Xie, Y.; Zhang, L. NVW-YOLOv8s: An improved YOLOv8s network for real-time detection and segmentation of tomato fruits at different ripeness stages. *Computers and Electronics in Agriculture* **2024**, *219*, 108833. <https://doi.org/10.1016/j.compag.2024.108833>.
96. Yue, X.; Qi, K.; Yang, F.; Na, X.; Liu, Y.; Liu, C. RSR-YOLO: A real-time method for small target tomato detection based on improved YOLOv8 network. *Discover Applied Sciences* **2024**, *6*, 268. <https://doi.org/10.1007/s42452-024-05914-1>.
97. Yang, Z.; Li, Y.; Han, Q.; Wang, H.; Li, C.; Wu, Z. A method for tomato ripeness recognition and detection based on an improved YOLOv8 model. *Horticulturae* **2025**, *11*, 15. <https://doi.org/10.3390/horticulturae11010015>.
98. Li, X.; Cai, C.; Song, B.; Yang, Y. YOLOv8-MR: An Improved Lightweight YOLOv8 Algorithm for Tomato Fruit Detection. *IEEE Access* **2025**, *13*. <https://doi.org/10.1109/ACCESS.2025.3533489>.
99. Qin, X.; Cao, J.; Zhang, Y.; Dong, T.; Cao, H. Development of an Optimized YOLO-PP-Based Cherry Tomato Detection System for Autonomous Precision Harvesting. *Processes* **2025**, *13*, 353. <https://doi.org/10.3390/pr13020353>.
100. Sun, H.; Zheng, Q.; Yao, W.; Wang, J.; Liu, C.; Yu, H.; Chen, C. An Improved YOLOv8 Model for Detecting Four Stages of Tomato Ripening and Its Application Deployment in a Greenhouse Environment. *Agriculture* **2025**, *15*, 936. <https://doi.org/10.3390/agriculture15090936>.
101. Yong, W.; Shunfa, X.; Konghao, C. YOLOv8-LBP: multi-scale attention enhanced YOLOv8 for ripe tomato detection and harvesting keypoint localization. *Frontiers in Plant Science* **2025**, *16*, 165638. <https://doi.org/10.3389/fpls.2025.1656381>.
102. Xia, C. Rapid strawberry ripeness detection and 3D localization of picking point based on improved YOLO V8-Pose with RGB-camera. *Journal of Electrical Systems* **2024**, *20-3s*, 2171–2181. <https://doi.org/10.52783/jes.1840>.
103. Chen, Y.; Xu, H.; Chang, P.; Huang, Y.; Zhong, F.; Jia, Q.; Chen, L.; Zhong, H.; Liu, S. CES-YOLOv8: Strawberry maturity detection based on the improved YOLOv8. *Agronomy* **2024**, *14*, 1353. <https://doi.org/10.3390/agronomy14071353>.
104. Luo, Q.; Wu, C.; Li, W. A Small Target Strawberry Recognition Method Based on Improved YOLOv8n Model. *IEEE Access* **2024**, *12*, 14987–14995. <https://doi.org/10.1109/ACCESS.2024.3356869>.
105. Yang, S.; Wang, W.; Gao, S.; Deng, Z. Strawberry ripeness detection based on YOLOv8 algorithm fused with LW-Swin Transformer. *Computers and Electronics in Agriculture* **2023**, *215*, 108360. <https://doi.org/10.1016/j.compag.2023.108360>.
106. He, Z.; Karkee, M.; Zhang, Q. Enhanced machine vision system for field-based detection of pickable strawberries: Integrating an advanced two-step deep learning model merging improved YOLOv8 and YOLOv5-cl. *Computers and Electronics in Agriculture* **2025**, *234*, 110173. <https://doi.org/10.1016/j.compag.2025.110173>.

107. He, L.; Wu, D.; Zheng, X.; Xu, F.; Lin, S.; Wang, S.; Ni, F.; Zheng, F. RLK-YOLOv8: Multi-stage detection of strawberry fruits throughout the full growth cycle in greenhouses based on large kernel convolutions and improved YOLOv8. *Frontiers in Plant Science* **2025**, *16*, 1552553. <https://doi.org/10.3389/fpls.2025.1552553>.
108. Ma, Z.; Dong, N.; Gu, J.; Cheng, H.; Meng, Z.; Du, X. STRAW-YOLO: A detection method for strawberry fruits targets and key points. *Computers and Electronics in Agriculture* **2025**, *230*, 109853. <https://doi.org/10.1016/j.compag.2024.109853>.
109. Zhao, S.; Fang, C.; Hua, T.; Jiang, Y. Detecting the Maturity of Red Strawberries Using Improved YOLOv8s Model. *Agriculture* **2025**, *15*, 2263. <https://doi.org/10.3390/agriculture15212263>.
110. Liu, Z.; Abeyrathna, R.M.R.D.; Sampurno, R.M.; Nakaguchi, V.M.; Ahamed, T. Faster-YOLO-AP: A lightweight apple detection algorithm based on improved YOLOv8 with a new efficient PDWConv in orchard. *Computers and Electronics in Agriculture* **2024**, *223*, 109118. <https://doi.org/10.1016/j.compag.2024.109118>.
111. Ma, B.; Hua, Z.; YuchenWen.; Deng, H.; Zhao, Y.; Pu, L.; Song, H. Using an improved lightweight YOLOv8 model for real-time detection of multi-stage apple fruit in complex orchard environments. *Artificial Intelligence in Agriculture* **2024**, *11*, 70–82. <https://doi.org/10.1016/j.aiaa.2024.02.001>.
112. Wu, H.; Mo, X.; Wen, S.; Wu, K.; Ye, Y.; Wang, Y.; Zhang, Y. DNE-YOLO: A method for apple fruit detection in Diverse Natural Environments. *Journal of King Saud University - Computer and Information Sciences* **2024**, *36*, 102220. <https://doi.org/10.1016/j.jksuci.2024.102220>.
113. Wu, T.; Miao, Z.; Huang, W.; Han, W.; Guo, Z.; Li, T. SGW-YOLOv8n: An Improved YOLOv8n-Based Model for Apple Detection and Segmentation in Complex Orchard Environments. *Agriculture* **2024**, *14*, 1958. <https://doi.org/10.3390/agriculture14111958>.
114. Yan, B.; Liu, Y.; Yan, W. A Novel Fusion Perception Algorithm of Tree Branch/Trunk and Apple for Harvesting Robot Based on Improved YOLOv8s. *Agronomy* **2024**, *14*, 1895. <https://doi.org/10.3390/agronomy14091895>.
115. Zhao, B.; Guo, A.; Ma, R.; Zhang, Y.; Gong, J. YOLOv8s-CFB: a lightweight method for real-time detection of apple fruits in complex environments. *Journal of Real-Time Image Processing* **2024**, *21*. <https://doi.org/10.1007/s11554-024-01543-4>.
116. Wang, M.; Li, F. Real-Time Accurate Apple Detection Based on Improved YOLOv8n in Complex Natural Environments. *Plants* **2025**, *14*, 365. <https://doi.org/10.3390/plants14030365>.
117. Li, Y.; Wang, Z.; Yang, A.; Yu, X. Integrating evolutionary algorithms and enhanced-YOLOv8 + for comprehensive apple ripeness prediction. *Scientific Reports* **2025**, *15*, 7307. <https://doi.org/10.1038/s41598-025-91939-4>.
118. Kong, D.; JiayiWang.; Zhang, Q.; Li, J.; Rong, J. Research on Fruit Spatial Coordinate Positioning by Combining Improved YOLOv8s and Adaptive Multi-Resolution Model. *Agronomy* **2023**, *13*, 2122. <https://doi.org/10.3390/agronomy13082122>.
119. Ang, G.; Zhiwei, T.; Wei, M.; Yuepeng, S.; Longlong, R.; Yuliang, F.; Jianping, Q.; Lijia, X. Fruits hidden by green: An improved YOLOv8n for detection of young citrus in lush citrus trees. *Frontiers in Plant Science* **2024**, *15*, 1375118. <https://doi.org/10.3389/fpls.2024.1375118>.
120. Lin, Y.; Huang, Z.; Liang, Y.; Liu, Y.; Jiang, W. AG-YOLO: A Rapid Citrus Fruit Detection Algorithm with Global Context Fusion. *Agriculture* **2024**, *14*, 114. <https://doi.org/10.3390/agriculture14010114>.
121. Cong, G.; Chen, X.; Bing, Z.; Liu, W.; Chen, X.; Wu, Q.; Guo, Z.; Zheng, Y. YOLOv8-Scm: an improved model for citrus fruit sunburn identification and classification in complex natural scenes. *Frontiers in Plant Science* **2025**, *16*, 1591989. <https://doi.org/10.3389/fpls.2025.1591989>.
122. Deng, F.; He, Z.; Fu, L.; Chen, J.; Li, N.; Chen, W.; Luo, J.; Qiao, W.; Hou, J.; Lu, Y. A new maturity recognition algorithm for Xinhui citrus based on improved YOLOv8. *Frontiers in Plant Science* **2025**, *16*, 1472230. <https://doi.org/10.3389/fpls.2025.1472230>.
123. Li, H.; Yin, Z.; Zuo, Z.; Pan, L.; Zhang, J. Precision citrus segmentation and stem picking point localization using improved YOLOv8n-seg algorithm. *Frontiers in Plant Science* **2025**, *16*, 1655093. <https://doi.org/10.3389/fpls.2025.1655093>.
124. Huang, Y.; Zhong, Y.; Zhong, D.; Yang, C.; Wei, L.; Zou, Z.; Chen, R. Pepper-YOLO: A lightweight model for green pepper detection and picking point localization in complex environments. *Frontiers in Plant Science* **2024**, *15*, 1508258. <https://doi.org/10.3389/fpls.2024.1508258>.
125. Ma, N.; Wu, Y.; Bo, Y.; Yan, H. Chili Pepper Object Detection Method Based on Improved YOLOv8n. *Plants* **2024**, *13*, 2402. <https://doi.org/10.3390/plants13172402>.
126. Duan, Y.; Li, J.; Zou, C. Research on Detection Method of Chaotian Pepper in Complex Field Environments Based on YOLOv8. *Sensors* **2024**, *24*, 5632. <https://doi.org/10.3390/s24175632>.

127. Wang, F.; Tang, Y.; Gong, Z.; Jiang, J.; Chen, Y.; Xu, Q.; Hu, P.; Zhu, H. A lightweight Yunnan Xiaomila detection and pose estimation based on improved YOLOv8. *Frontiers in Plant Science* **2024**, *15*, 1421381. <https://doi.org/10.3389/fpls.2024.1421381>.
128. Ma, Y.; Zhang, S. YOLOv8-CBSE: An Enhanced Computer Vision Model for Detecting the Maturity of Chili Pepper in the Natural Environment. *Agronomy* **2025**, *15*, 537. <https://doi.org/10.3390/agronomy15030537>.
129. Fan, J.; Salam, M.S.B.H.; Rong, X.; Han, Y.; Yang, J.; Zhang, J. Peach Fruit Thinning Image Detection Based on Improved YOLOv8 and Data Enhancement Techniques. *IEEE Access* **2024**, *12*, 191199–191218. <https://doi.org/10.1109/ACCESS.2024.3516484>.
130. Jing, J.; Zhang, S.; Sun, H.; Ren, R.; Cui, T. YOLO-PEM: A Lightweight Detection Method for Young “Okubo” Peaches in Complex Orchard Environments. *Agronomy* **2024**, *14*, 1757. <https://doi.org/10.3390/agronomy14081757>.
131. Li, T.; Chen, Q.; Zhang, X.; Ding, S.; Wang, X.; Mu, J. PeachYOLO: A Lightweight Algorithm for Peach Detection in Complex Orchard Environments. *IEEE Access* **2024**, *12*, 96220–96230. <https://doi.org/10.1109/ACCESS.2024.3411644>.
132. Xu, D.; Xiong, H.; Liao, Y.; Wang, H.; Yuan, Z.; Yin, H. EMA-YOLO: A Novel Target-Detection Algorithm for Immature Yellow Peach Based on YOLOv8. *Sensors* **2024**, *24*, 3783. <https://doi.org/10.3390/s24123783>.
133. Li, X.; Liu, H.; Lan, Y.; Chen, G.; Shan, C.; Jiao, L.; Liu, J.; Huizheng Wang, 1, 2, . SYL-YOLOv8n: A lightweight and robust detector for peach recognition and yield estimation in complex orchards. *Journal of Food Measurement and Characterization* **2025**. <https://doi.org/10.1007/s11694-025-03664-1>.
134. Xie, S.; Sun, H. Tea-YOLOv8s: A tea bud detection model based on deep learning and computer vision. *Sensors* **2023**, *23*, 6576. <https://doi.org/10.3390/s23146576>.
135. Zhou, C.; Zhu, Y.; Zhang, J.; Ding, Z.; Jiang, W.; Zhang, K. The tea buds detection and yield estimation method based on optimized YOLOv8. *Scientia Horticulturae* **2024**, *338*, 113730. <https://doi.org/10.1016/j.scienta.2024.113730>.
136. Wang, C.; Li, H.; Deng, X.; Liu, Y.; Wu, T.; Liu, W.; Xiao, R.; Wang, Z.; Wang, B. Improved You Only Look Once v.8 Model Based on Deep Learning: Precision Detection and Recognition of Fresh Leaves from Yunnan Large-Leaf Tea Tree. *Agriculture* **2024**, *14*, 2324. <https://doi.org/10.3390/agriculture14122324>.
137. Tang, X.; Tang, L.; Li, J.; Guo, X. Enhancing multilevel tea leaf recognition based on improved YOLOv8n. *Frontiers in Plant Science* **2025**, *16*, 1540670. <https://doi.org/10.3389/fpls.2025.1540670>.
138. Yang, Q.; Gu, J.; Xiong, T.; Wang, Q.; Huang, J.; Xi, Y.; Shen, Z. RFA-YOLOv8: A Robust Tea Bud Detection Model with Adaptive Illumination Enhancement for Complex Orchard Environments. *Agriculture* **2025**, *15*, 1982. <https://doi.org/10.3390/agriculture15181982>.
139. Gu, Z.; He, D.; Huang, J.; Chen, J.; Wu, X.; Huang, B.; Dong, T.; Yang, Q.; Li, H. Simultaneous detection of fruits and fruiting stems in mango using improved YOLOv8 model deployed by edge device. *Computers and Electronics in Agriculture* **2024**, *227*, 109512. <https://doi.org/10.1016/j.compag.2024.109512>.
140. Li, H.; Huang, J.; Gu, Z.; He, D.; Huang, J.; Wang, C. Positioning of mango picking point using an improved YOLOv8 architecture with object detection and instance segmentation. *Biosystems Engineering* **2024**, *247*, 202–220. <https://doi.org/10.1016/j.biosystemseng.2024.09.015>.
141. Wu, X.; Tang, R.; Mu, J.; Niu, Y.; Xu, Z.; Chen, Z. A lightweight grape detection model in natural environments based on an enhanced YOLOv8 framework. *Frontiers in Plant Science* **2024**, *15*, 1407839. <https://doi.org/10.3389/fpls.2024.1407839>.
142. Chen, J.; Ma, A.; Huang, L.; Li, H.; Zhang, H.; Huang, Y.; Zhu, T. Efficient and lightweight grape and picking point synchronous detection model based on key point detection. *Computers and Electronics in Agriculture* **2024**, *217*, 108612. <https://doi.org/10.1016/j.compag.2024.108612>.
143. Chen, B.; Ding, F.; Ma, B.; Wang, L.; Ning, S. A method for real-time recognition of safflower filaments in unstructured environments using the YOLO-SaFi model. *Sensors* **2024**, *24*, 4410. <https://doi.org/10.3390/s24134410>.
144. Zhang, Z.; Wang, Y.; Xu, P.; Shi, R.; Xing, Z.; Li, J. WED-YOLO: A detection model for safflower under complex unstructured environment. *Agriculture* **2025**, *15*, 205. <https://doi.org/10.3390/agriculture15020205>.
145. Chen, Y.; Liu, Q.; Jiang, X.; Wei, Y.; Zhou, X.; Zhou, J.; Wang, F.; Yan, L.; Fan, S.; Xing, H. FEW-YOLO: a lightweight ripe fruit detection algorithm in wolfberry based on improved YOLOv8. *Journal of Food Measurement and Characterization* **2025**, *19*, 4783–4795. <https://doi.org/10.1007/s11694-025-03290-x>.
146. Liu, H.; Gu, W.; Wang, W.; Zou, Y.; Yang, H.; Li, T. Persimmon fruit detection in complex scenes based on PerD-YOLOv8. *Journal of Food Measurement and Characterization* **2025**, *19*, 4543–4560. <https://doi.org/10.1007/s11694-025-03268-9>.

147. He, Y.; Li, Y.; Li, Z.; Song, R.; Xu, C. An improved YOLOv8-based lightweight approach for orange maturity detection. *Journal of Food Measurement and Characterization* **2025**, *19*, 4740–4754. <https://doi.org/10.1007/s11694-025-03286-7>.
148. Liu, Y.; Chen, D.; Zhang, Y.; Wang, X. An improved YOLOv8-seg-based method for key part segmentation of tobacco plants. *Frontiers in Plant Science* **2025**, *16*, 1673202. <https://doi.org/10.3389/fpls.2025.1673202>.
149. Wang, S.; Wei, L.; Zhang, D.; Chen, L.; Huang, W.; Du, D.; Lin, K.; Zheng, Z.; Duan, J. Real-time and resource-efficient banana bunch detection and localization with YOLO-BRFB on edge devices. *Frontiers in Plant Science* **2025**, *16*, 1650012. <https://doi.org/10.3389/fpls.2025.1650012>.
150. Lin, Y.; Xiao, X.; Lin, H. YOLOv8-FDA: lightweight wheat ear detection and counting in drone images based on improved YOLOv8. *Frontiers in Plant Science* **2025**, *16*, 1682243. <https://doi.org/10.3389/fpls.2025.1682243>.
151. Wei, J.; Wang, R.; Wei, S.; Wang, X.; Xu, S. Recognition of maize tassels based on improved YOLOv8 and unmanned aerial vehicles RGB images. *Drones* **2024**, *8*, 691. <https://doi.org/10.3390/drones8110691>.
152. Diao, Z.; Ma, S.; Zhang, D.; Zhang, J.; Guo, P.; He, Z.; Zhao, S.; Zhang, B. Algorithm for corn crop row recognition during different growth stages based on ST-YOLOv8s network. *Agronomy* **2024**, *14*, 1466. <https://doi.org/10.3390/agronomy14071466>.
153. Yu, X.; Yin, D.; Xu, H.; Espinosa, F.P.; Schmidhalter, U.; Nie, C.; Bai, Y.; Sankaran, S.; Ming, B.; Cui, N.; et al. Maize tassel number and tasseling stage monitoring based on near-ground and UAV RGB images by improved YOLOv8. *Precision Agriculture* **2024**, *25*, 1800–1838. <https://doi.org/10.1007/s11119-024-10135-y>.
154. Sun, W.; Xu, M.; Xu, K.; Chen, D.; Wang, J.; Yang, R.; Chen, Q.; Yang, S. CSGD-YOLO: A Corn Seed Germination Status Detection Model Based on YOLOv8n. *Agronomy* **2025**, *15*, 128. <https://doi.org/10.3390/agronomy15010128>.
155. Chen, G.; Hou, Y.; Cui, T.; Li, H.; Shangguan, F.; Cao, L. YOLOv8-CML: A lightweight target detection method for color-changing melon ripening in intelligent agriculture. *Scientific Reports* **2024**, *14*, 14400. <https://doi.org/10.1038/s41598-024-65293-w>.
156. Chen, G.; Hou, Y.; Chen, H.; Cao, L.; Yuan, J. A lightweight color-changing melon ripeness detection algorithm based on model pruning and knowledge distillation: leveraging dilated residual and multi-screening path aggregation. *Frontiers in Plant Science* **2024**, *15*, 1406593. <https://doi.org/10.3389/fpls.2024.1406593>.
157. Shui, Y.; Yuan, K.; Wu, M.; Zhao, Z. Improved multi-size, multi-target and 3D position detection network for flowering Chinese cabbage based on YOLOv8. *Plants* **2024**, *13*, 2808. <https://doi.org/10.3390/plants13192808>.
158. Jiang, P.; Qi, A.; Zhong, J.; Luo, Y.; Hu, W.; Shi, Y.; Liu, T. Field cabbage detection and positioning system based on improved YOLOv8n. *Plant Methods* **2024**, *20*, 96. <https://doi.org/10.1186/s13007-024-01226-y>.
159. Gai, R.; Liu, Y.; Xu, G. TL-YOLOv8: A Blueberry fruit detection algorithm based on improved YOLOv8 and transfer learning. *IEEE Access* **2024**, *12*, 86378–86390. <https://doi.org/10.1109/ACCESS.2024.3416332>.
160. Zhang, G.; Yang, X.; Lv, D.; Zhao, Y.; Liu, P. YOLOv8n-CSD: A lightweight detection method for nectarines in complex environments. *Agronomy* **2024**, *14*, 2427. <https://doi.org/10.3390/agronomy14102427>.
161. Liu, Q.; Lv, J.; Zhang, C. MAE-YOLOv8-based small object detection of green crisp plum in real complex orchard environments. *Computers and Electronics in Agriculture* **2024**, *226*, 109458. <https://doi.org/10.1016/j.compag.2024.109458>.
162. Kazama, E.H.; Tedesco, D.; dos Santos Carreira, V.; Júnior, M.R.B.; de Oliveira, M.F.; Ferreira, F.M.; Junior, W.M.; da Silva, R.P. Monitoring coffee fruit maturity using an enhanced convolutional neural network under different image acquisition settings. *Scientia Horticulturae* **2024**, *328*, 112957. <https://doi.org/10.1016/j.scienta.2024.112957>.
163. Ma, J.; Zhao, Y.; Fan, W.; Liu, J. An Improved YOLOv8 Model for Lotus Seedpod Instance Segmentation in the Lotus Pond Environment. *Agronomy* **2024**, *14*, 1325. <https://doi.org/10.3390/agronomy14061325>.
164. Liang, C.; Liu, D.; Ge, W.; Huang, W.; Lan, Y.; Long, Y. Detection of litchi fruit maturity states based on unmanned aerial vehicle remote sensing and improved YOLOv8 model. *Frontiers in Plant Science* **2025**, *16*, 1568237. <https://doi.org/10.3389/fpls.2025.1568237>.
165. Wang, H.; Yun, L.; Yang, C.; Wu, M.; Wang, Y.; Chen, Z. OW-YOLO: An Improved YOLOv8s Lightweight Detection Method for Obstructed Walnuts. *Agriculture* **2025**, *15*, 159. <https://doi.org/10.3390/agriculture15020159>.

166. Fu, Y.; Wang, Z.; Zheng, H.; Yin, X.; Fu, W.; Gu, Y. Integrated detection of coconut clusters and oriented leaves using improved YOLOv8n-obb for robotic harvesting. *Computers and Electronics in Agriculture* **2025**, *231*, 109979. <https://doi.org/10.1016/j.compag.2025.109979>.
167. Tian, Y.; Zhao, C.; Zhang, T.; Wu, H.; Zhao, Y. Recognition Method of Cabbage Heads at Harvest Stage under Complex Background Based on Improved YOLOv8n. *Agriculture* **2024**, *14*, 1125. <https://doi.org/10.3390/agriculture14071125>.
168. Wang, J.; Liu, M.; Du, Y.; Zhao, M.; Jia, H.; Guo, Z.; Su, Y.; Lu, D.; Liu, Y. PG-YOLO: An efficient detection algorithm for pomegranate before fruit thinning. *Engineering Applications of Artificial Intelligence* **2024**, *134*, 108700. <https://doi.org/10.1016/j.engappai.2024.108700>.
169. Li, J.; Zhang, T.; Luo, Q.; Zeng, S.; Luo, X.; Chen, P.; Yang, C. A lightweight palm fruit detection network for harvesting equipment integrates binocular depth matching. *Computers and Electronics in Agriculture* **2025**, *233*, 110061. <https://doi.org/10.1016/j.compag.2025.110061>.
170. Jia, X.; Hua, Z.; Shi, H.; Zhu, D.; Han, Z.; Wu, G.; Deng, L. A Soybean Pod Accuracy Detection and Counting Model Based on Improved YOLOv8. *Agriculture* **2025**, *15*, 617. <https://doi.org/10.3390/agriculture15060617>.
171. Gao, S.; Cui, G.; Wang, Q. WCS-YOLOv8s: an improved YOLOv8s model for target identification and localization throughout the strawberry growth process. *Frontiers in Plant Science* **2025**, *16*, 1579335. <https://doi.org/10.3389/fpls.2025.1579335>.
172. Zhang, X.; Zhang, N.; Xu, X.; Wang, H.; Cao, J. Optimal cutting point determination for robotic raspberry harvesting based on computer vision strategy. *Multimedia Tools and Applications* **2025**. <https://doi.org/10.1007/s11042-025-20808-z>.
173. Lin, C.; Jiang, W.; Zhao, W.; Zou, L.; Xue, Z. DPD-YOLO: Dense pineapple fruit target detection algorithm in complex environments based on YOLOv8 combined with attention mechanism. *Frontiers Plant Science* **2025**, *16*, 1523552. <https://doi.org/10.3389/fpls.2025.1523552>.
174. Shi, J.; Bai, Y.; Zhou, J.; Zhang, B. Multi-Crop Navigation Line Extraction Based on Improved YOLO-v8 and Threshold-DBSCAN under Complex Agricultural Environments. *Agriculture* **2023**, *14*, 45. <https://doi.org/10.3390/agriculture14010045>.
175. Wang, Z.; Zhang, C. An improved chilli pepper flower detection approach based on YOLOv8. *Plant Methods* **2025**, *21*, 71. <https://doi.org/10.1186/s13007-025-01390-9>.
176. Jiang, H.; Hu, F.; Fu, X.; Chen, C.; Wang, C.; Tian, L.; Shi, Y. YOLOv8-Peas: A lightweight drought tolerance method for peas based on seed germination vigor. *Frontiers Plant Science* **2023**, *14*, 1257947. <https://doi.org/10.3389/fpls.2023.1257947>.
177. Fang, C.; Yang, X. Lightweight YOLOv8 for wheat head detection. *IEEE Access* **2024**, *12*, 66214–66222. <https://doi.org/10.1109/ACCESS.2024.3397556>.
178. Zhao, K.; Li, J.; Shi, W.; Qi, L.; Yu, C.; Zhang, W. Field-based soybean flower and pod detection using an improved YOLOv8-VEW method. *Agriculture* **2024**, *14*, 1423. <https://doi.org/10.3390/agriculture14081423>.
179. Zhang, J.; Yang, W.; Lu, Z.; Chen, D. HR-YOLOv8: A crop growth status object detection method based on YOLOv8. *Electronics* **2024**, *13*, 1620. <https://doi.org/10.3390/electronics13091620>.
180. Zhang, F.; Dong, D.; Jia, X.; Guo, J.; Yu, X. Sugarcane-YOLO: An improved YOLOv8 model for accurate identification of sugarcane seed sprouts. *Agronomy* **2024**, *14*, 2412. <https://doi.org/10.3390/agronomy14102412>.
181. Qiu, Z.; Zhuo, S.; Li, M.; Huang, F.; Mo, D.; Tian, X.; Tian, X. An Efficient Detection of the Pitaya Growth Status Based on the YOLOv8n-CBN Model. *Horticulturae* **2024**, *10*, 899. <https://doi.org/10.3390/horticulturae10090899>.
182. Hemamalini, P.; Chandraprakash, M.K.; Laxman, R.H.; Rathinakumari, C.; Kumaran, G.S.; Suneetha, K. Thermal canopy segmentation in tomato plants: A novel approach with integration of YOLOv8-C and FastSAM. *Smart Agricultural Technology* **2025**, *10*, 100806. <https://doi.org/10.1016/j.atech.2025.100806>.
183. Song, H.; Zeng, Y.; Wen, T.; Li, X.; Liu, Y. EggYOLOPlant: Optimized YOLOv8 for real-time eggplant seedling center detection. *Journal of Real-Time Image Processing* **2025**, *22*, 100. <https://doi.org/10.1007/s11554-025-01676-0>.
184. Wang, D.; Song, H.; Wang, B. YO-AFD: an improved YOLOv8-based deep learning approach for rapid and accurate apple flower detection. *Frontiers Plant Science* **2025**, *16*, 1541266. <https://doi.org/10.3389/fpls.2025.1541266>.

185. Wang, L.; Xiao, J.; Peng, X.; Tan, Y.; Zhou, Z.; Chen, L.; Tang, Q.; Cheng, W.; Liang, X. Mango Inflorescence Detection Based on Improved YOLOv8 and UAVs-RGB Images. *Forests* **2025**, *16*, 896. <https://doi.org/10.3390/f16060896>.
186. Zheng, H.; Liu, C.; Zhong, L.; Wang, J.; Huang, J.; Lin, F.; Ma, X.; Tan, S. An android-smartphone application for rice panicle detection and rice growth stage recognition using a lightweight YOLO network. *Frontiers in Plant Science* **2025**, *16*, 1561632. <https://doi.org/10.3389/fpls.2025.1561632>.
187. Guo, B.; Ling, S.; Tan, H.; Wang, S.; Wu, C.; Yang, D. Detection of the Grassland Weed *Phlomis umbrosa* Using Multi-Source Imagery and an Improved YOLOv8 Network. *Agronomy* **2023**, *13*, 3001. <https://doi.org/10.3390/agronomy13123001>.
188. Niu, W.; Lei, X.; Li, H.; Wu, H.; Hu, F.; Wen, X.; Zheng, D.; Song, H. YOLOv8-ECFS: A lightweight model for weed species detection in soybean fields. *Crop Protection* **2024**, *184*, 106847. <https://doi.org/10.1016/j.cropro.2024.106847>.
189. He, C.; Wan, F.; Ma, G.; Mou, X.; Zhang, K.; Wu, X.; Huang, X. Analysis of the impact of different improvement methods based on YOLOv8 for weed detection. *Agriculture* **2024**, *14*, 674. <https://doi.org/10.3390/agriculture14050674>.
190. Ding, Y.; Jiang, C.; Song, L.; Liu, F.; Tao, Y. RVDR-YOLOv8: A Weed Target Detection Model Based on Improved YOLOv8. *Electronics* **2024**, *13*, 2182. <https://doi.org/10.3390/electronics13112182>.
191. Liu, H.; Hou, Y.; Zhang, J.; Zheng, P.; Hou, S. Research on weed reverse detection methods based on improved You Only Look Once (YOLO) v8: Preliminary Results. *Agronomy* **2024**, *14*, 1667. <https://doi.org/10.3390/agronomy14081667>.
192. Jia, Z.; Zhang, M.; Yuan, C.; Liu, Q.; Liu, H.; Qiu, X.; Zhao, W.; Shi, J. ADL-YOLOv8: A field crop weed detection model based on improved YOLOv8. *Agronomy* **2024**, *14*, 2355. <https://doi.org/10.3390/agronomy14102355>.
193. Lyu, Z.; Lu, A.; Ma, Y. Improved YOLOv8-Seg Based on Multiscale Feature Fusion and Deformable Convolution for Weed Precision Segmentation. *Applied Sciences* **2024**, *14*, 5002. <https://doi.org/10.3390/app14125002>.
194. Zheng, L.; Yi, J.; He, P.; Tie, J.; Zhang, Y.; Wu, W.; Long, L. Improvement of the YOLOv8 model in the optimization of the weed recognition algorithm in cotton field. *Plants* **2024**, *13*, 1843. <https://doi.org/10.3390/plants13131843>.
195. Ren, D.; Yang, W.; Lu, Z.; Chen, D.; Shi, H. Improved Weed Detection in Cotton Fields Using Enhanced YOLOv8s with Modified Feature Extraction Modules. *Symmetry* **2024**, *16*, 450. <https://doi.org/10.3390/sym16040450>.
196. Karim, M.J.; Nahiduzzaman, M.; Ahsan, M.; Haider, J. Development of an early detection and automatic targeting system for cotton weeds using an improved lightweight YOLOv8 architecture on an edge device. *Knowledge-Based Systems* **2024**, *300*, 112204. <https://doi.org/10.1016/j.knsys.2024.112204>.
197. Liu, Y.; Zeng, F.; Diao, H.; Zhu, J.; Ji, D.; Liao, X.; Zhao, Z. YOLOv8 Model for Weed Detection in Wheat Fields Based on a Visual Converter and Multi-Scale Feature Fusion. *Sensors* **2024**, *24*, 4379. <https://doi.org/10.3390/s24134379>.
198. Wang, J.; Qi, Z.; Wang, Y.; Liu, Y. A lightweight weed detection model for cotton fields based on an improved YOLOv8n. *Scientific Reports* **2025**, *15*, 57. <https://doi.org/10.1038/s41598-024-84748-8>.
199. Ma, C.; Chi, G.; Ju, X.; Zhang, J.; Yan, C. YOLO-CWD: A novel model for crop and weed detection based on improved YOLOv8. *Crop Protection* **2025**, *192*, 107169. <https://doi.org/10.1016/j.cropro.2025.107169>.
200. Zheng, L.; Zhu, C.; Liu, L.; Yang, Y.; Wang, J.; Xia, W.; Xu, K.; Tie, J. Star-YOLO: A lightweight and efficient model for weed detection in cotton fields using advanced YOLOv8 improvements. *Computers and Electronics in Agriculture* **2025**, *235*, 110306. <https://doi.org/10.1016/j.compag.2025.110306>.
201. Ning, S.; Tan, F.; Chen, X.; Li, X.; Shi, H.; Qiu, J. Lightweight corn leaf detection and counting using improved YOLOv8. *Sensors* **2024**, *24*, 5279. <https://doi.org/10.3390/s24165279>.
202. Guan, H.; Deng, H.; Ma, X.; Zhang, T.; Zhang, Y.; Zhu, T.; Zhou, H.; Gu, Z.; Lu, Y. A corn canopy organs detection method based on improved DBi-YOLOv8 network. *European Journal of Agronomy* **2024**, *154*, 127076. <https://doi.org/10.1016/j.eja.2023.127076>.
203. Ma, H.; Sheng, T.; Ma, Y.; Gou, J. An Improved Ningxia Desert Herbaceous Plant Classification Algorithm Based on YOLOv8. *Sensors* **2024**, *24*, 3834. <https://doi.org/10.3390/s24123834>.

204. Pan, Y.; Xiao, X.; Hu, K.; Kang, H.; Jin, Y.; Chen, Y.; Zou, X. ODN-Pro: An improved model based on YOLOv8 for enhanced instance detection in orchard point clouds. *Agronomy* **2024**, *4*, 697. <https://doi.org/10.3390/agronomy14040697>.
205. Chen, J.; Ji, C.; Zhang, J.; Feng, Q.; Li, Y.; Ma, B. A method for multi-target segmentation of bud-stage apple trees based on improved YOLOv8. *Computers and Electronics in Agriculture* **2024**, *220*, 108876. <https://doi.org/10.1016/j.compag.2024.108876>.
206. Ji, S.; Sun, J.; Zhang, C. Phenotypic Image Recognition of Asparagus Stem Blight Based on Improved YOLOv8. *Computers, Materials & Continua* **2024**, *80*, 4017–4029. <https://doi.org/10.32604/cmc.2024.055038>.
207. Zhang, F.; Zhao, L.; Wang, D.; Wang, J.; Smirnov, I.; Li, J. MS-YOLOv8: Multi-scale adaptive recognition and counting model for peanut seedlings under salt-alkali stress from remote sensing. *Frontiers in Plant Science* **2024**, *15*, 1434968. <https://doi.org/10.3389/fpls.2024.1434968>.
208. Song, Y.; Yang, L.; Li, S.; Yang, X.; Ma, C.; Huang, Y.; Hussain, A. Improved YOLOv8 Model for Phenotype Detection of Horticultural Seedling Growth Based on Digital Cousin. *Agriculture* **2025**, *15*, 28. <https://doi.org/10.3390/agriculture15010028>.
209. Ren, R.; Zhang, S.; Sun, H.; Wang, N.; Yang, S.; Zhao, H.; Xin, M. YOLO-RCS: A method for detecting phenological period of 'Yuluxiang' pear in unstructured environment. *Computers and Electronics in Agriculture* **2025**, *229*, 109819. <https://doi.org/10.1016/j.compag.2024.109819>.
210. Han, K.; Wang, Y.; Tian, Q.; Guo, J.; Xu, C.; Xu, C. GhostNet: More Features from Cheap Operations. *arXiv:1911.11907v2* **2020**. <https://doi.org/10.48550/arXiv.1911.11907>.
211. Xie, W.; Feng, F.; Zhang, H. A detection algorithm for citrus Huanglongbing disease based on an improved YOLOv8n. *Sensors* **2024**, *24*, 4448. <https://doi.org/10.3390/s24144448>.
212. with Cross-Spatial Learning, E.M.S.A.M. HS-FPN: High Frequency and Spatial Perception FPN for Tiny Object Detection. *arXiv:2305.13563* **2023**. <https://doi.org/10.48550/arXiv.2305.13563>.
213. Khalili, B.; Smyth, A.W. SOD-YOLOv8—Enhancing YOLOv8 for Small Object Detection in Aerial Imagery and Traffic Scenes. *Sensors* **2024**, *24*, 6209. <https://doi.org/10.3390/s24196209>.
214. Wang, S.; Li, Y.; Qiao, S. ALF-YOLO: Enhanced YOLOv8 Based on Multiscale Attention Feature Fusion for Ship Detection. *Ocean Engineering* **2024**, *308*, 118233. <https://doi.org/10.1016/j.oceaneng.2024.118233>.
215. Li, C.; Li, L.; Geng, Y.; Jiang, H.; Zhang, M.C.B.; Ke, Z.; Xu, X.; Chu, X. YOLOv6 v3.0: A Full-Scale Reloading. *arXiv:2301.05586v1* **2023**. <https://doi.org/10.48550/arXiv.2301.05586>.
216. Jiang, Y.; Tan, Z.; Wang, J.; Sun, X.; Lin, M.; Li, H. GiraffeDet: A Heavy-Neck Paradigm for Object Detection. *arXiv:2202.04256v2* **2022**. <https://doi.org/10.48550/arXiv.2202.04256>.
217. Wang, C.; He, W.; Nie, Y.; Guo, J.; Liu, C.; Han, K.; Wang, Y. Gold-YOLO: Efficient Object Detector via Gather-and-Distribute Mechanism. *arXiv:2309.11331v5* **2023**. <https://doi.org/10.48550/arXiv.2309.11331>.
218. Chen, Z.; He, Z.; Lu, Z.M. DEA-Net: Single Image Dehazing Based on Detail-Enhanced Convolution and Content-Guided Attention. *IEEE Transactions on Image Processing* **2024**, *33*, 1002–1015. <https://doi.org/10.1109/TIP.2024.3354108>.
219. Shi, Z.; Hu, J.; Ren, J.; Ye, H.; Yuan, X.; Ouyang, Y.; He, J.; Ji, B.; Guo, J. HS-FPN: High Frequency and Spatial Perception FPN for Tiny Object Detection. *arXiv:2412.10116v3* **2024**. <https://doi.org/10.48550/arXiv.2412.10116>.
220. Wang, Y.; Yi, C.; Huang, T.; Liu, J. Research on intelligent recognition for plant pests and diseases based on improved YOLOv8 model. *Applied Sciences* **2024**, *14*, 5353. <https://doi.org/10.3390/app14125353>.
221. Zou, H.; Lv, P.; Zhao, M. Detection of Apple Leaf Diseases Based on LightYOLO-AppleLeafDx. *Plants* **2025**, *14*, 599. <https://doi.org/10.3390/plants14040599>.
222. Li, H.; Li, J.; Wei, H.; Liu, Z.; Zhan, Z.; Ren, Q. Slim-neck by GSConv: A lightweight-design for real-time detector architectures. *arXiv:2206.02424v3* **2022**. <https://doi.org/10.48550/arXiv.2206.02424>.
223. Shen, L.; Lang, B.; Song, Z. DS-YOLOv8-Based Object Detection Method for Remote Sensing Images. *IEEE Access* **2023**, *11*, 125122–125137. <https://doi.org/10.1109/ACCESS.2023.3330844>.
224. Hou, Q.; Zhou, D.; Feng, J. Coordinate Attention for Efficient Mobile Network Design. *arXiv:2103.02907v1* **2021**. <https://doi.org/10.48550/arXiv.2103.02907>.
225. Dai, X.; Chen, Y.; Xiao, B.; Chen, D.; Liu, M.; Yuan, L.; Zhang, L. Dynamic Head: Unifying Object Detection Heads with Attentions. *arXiv:2106.08322v1* **2021**. <https://doi.org/10.48550/arXiv.2106.08322>.

226. Zhao, Z.; Chen, S.; Ge, Y.; Yang, P.; Wang, Y.; Song, Y. RT-DETR-Tomato: Tomato Target Detection Algorithm Based on Improved RT-DETR for Agricultural Safety Production. *Applied Sciences* **2024**, *14*, 6287. <https://doi.org/10.3390/app14146287>.
227. Qin, R.; Wang, Y.; Liu, X.; Yu, H. Advancing precision agriculture with deep learning enhanced SIS-YOLOv8 for Solanaceae crop monitoring. *Frontiers Plant Science* **2025**, *15*, 1485903. <https://doi.org/10.3389/fpls.2024.1485903>.

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.