



Article

YOLOv9s-Pear: A Lightweight YOLOv9s-Based Improved Model for Young Red Pear Small-Target Recognition

Yi Shi ^{1,2}, Zhen Duan ^{2,*}, Shunhao Qing ^{1,*} , Long Zhao ³ , Fei Wang ¹ and Xingcan Yuwen ¹

¹ College of Agricultural Equipment Engineering, Henan University of Science and Technology, Luoyang 471000, China; shiyigongteng@163.com (Y.S.); wf01082021@163.com (F.W.); 15227166478@163.com (X.Y.)

² Academy of Agricultural Planning and Engineering, Ministry of Agriculture and Rural Affairs, Beijing 100125, China

³ College of Horticulture and Plant Protection, Henan University of Science and Technology, Luoyang 471000, China; hkdzhaolong@haust.edu.cn

* Correspondence: aape_duanzhen@163.com (Z.D.); 220320261805@stu.haust.edu.cn (S.Q.)

Abstract: With the advancement of computer vision technology, the demand for fruit recognition in agricultural automation is increasing. To improve the accuracy and efficiency of recognizing young *red pears*, this study proposes an improved model based on the lightweight YOLOv9s, termed YOLOv9s-Pear. By constructing a feature-rich and diverse image dataset of young *red pears* and introducing spatial-channel decoupled downsampling (SCDown), C2FUIBELAN, and the YOLOv10 detection head (v10detect) modules, the YOLOv9s model was enhanced to achieve efficient recognition of small targets in resource-constrained agricultural environments. Images of young *red pears* were captured at different times and locations and underwent preprocessing to establish a high-quality dataset. For model improvements, this study integrated the general inverted bottleneck blocks from C2f and MobileNetV4 with the RepNCSPPELAN4 module from the YOLOv9s model to form the new C2FUIBELAN module, enhancing the model's accuracy and training speed for small-scale object detection. Additionally, the SCDown and v10detect modules replaced the original AConv and detection head structures of the YOLOv9s model, further improving performance. The experimental results demonstrated that the YOLOv9s-Pear model achieved high detection accuracy in recognizing young *red pears*, while reducing computational costs and parameters. The detection accuracy, recall, mean precision, and extended mean precision were 0.971, 0.970, 0.991, and 0.848, respectively. These results confirm the efficiency of the SCDown, C2FUIBELAN, and v10detect modules in young red pear recognition tasks. The findings of this study not only provide a fast and accurate technique for recognizing young *red pears* but also offer a reference for detecting young fruits of other fruit trees, significantly contributing to the advancement of agricultural automation technology.

Keywords: YOLOv9s; young *red pears*; object recognition; computer vision; lightweight model



Citation: Shi, Y.; Duan, Z.; Qing, S.; Zhao, L.; Wang, F.; Yuwen, X. YOLOv9s-Pear: A Lightweight YOLOv9s-Based Improved Model for Young Red Pear Small-Target Recognition. *Agronomy* **2024**, *14*, 2086. <https://doi.org/10.3390/agronomy14092086>

Academic Editor: Chengming Sun

Received: 9 August 2024

Revised: 8 September 2024

Accepted: 11 September 2024

Published: 12 September 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With the continuous development of automation and intelligent technologies, automated fruit detection has become a key factor in driving economic growth in the fruit industry and improving agricultural production efficiency [1–3]. *Red pears*, a fruit with high economic value, require early identification during the young fruit stage, which is critical for tasks such as fruit thinning, growth monitoring, and yield prediction [4,5]. However, during this stage, the fruits are small in size and varied in shape, and the complexity of their growing environment often makes it difficult to accurately and efficiently detect them. Factors such as lighting conditions and complex backgrounds can significantly impact the recognition process, leading to lower accuracy and efficiency [6]. Therefore, developing a precise and efficient method for recognizing young fruits is crucial. This not only enhances

the level of intelligent agricultural production but also helps reduce production costs and increase economic benefits.

Currently, orchard fruit detection methods can be categorized into three main types: manual detection, detection based on machine learning and image processing, and detection based on deep learning [7–10]. Although manual detection is highly accurate due to human expertise, it is labor-intensive, time-consuming, and inefficient [11]. Machine learning and image processing technologies provide automated solutions by using algorithms to identify fruit features; however, they often require extensive feature engineering and may struggle with the variability of fruit appearance and environmental conditions. Saranya, et al. [12] conducted classification experiments on fruits like apples, bananas, oranges, and pomegranates using the Fruit-360 dataset. They explored traditional machine learning algorithms such as KNN and SVM and compared them with CNN, a deep learning algorithm. The results showed that CNN outperformed traditional algorithms in fruit image recognition, providing technical support for the development of intelligent fresh parks. Yamamoto, et al. [13] developed a method based on traditional RGB cameras and machine learning to detect tomatoes through image segmentation and classification models based on color, shape, texture, and size. The experimental results showed that this method achieved a recall rate of 0.80 and an accuracy rate of 0.88 in fruit detection on test images, with recall rates for mature fruits, immature fruits, and young fruits being 1.00, 0.80, and 0.78, respectively. Although the combination of image processing technology and machine learning has yielded good results in fruit detection, the feature extraction and parameter settings require significant technical and professional expertise, and feature extraction can be challenging in complex scenarios [14,15]. By contrast, detection methods based on deep learning, which use convolutional neural networks (CNNs) to automatically learn and extract relevant features from images, are significantly superior to the other two methods. These methods improve accuracy and efficiency, especially in complex and dynamic orchard environments [16].

Significant progress has been made in image recognition using deep learning technology, which employs multi-layer neural network structures to learn and extract image features, leading to efficient recognition and detection [17,18]. Yang, et al. [19] proposed an improved YOLOv5 model for the recognition and detection of mature fruits in orchards. This model, based on YOLOv5, introduced a bidirectional cross-attention mechanism, achieving an average precision of 97.70%. Bai, et al. [20] addressed the challenge of detecting small flowers and fruits with similar colors in strawberry seedlings by proposing an improved YOLO algorithm with a Swin Transformer prediction head for fast and accurate detection in greenhouses. The study showed that this model achieved 92.6%, 89.6%, and 92.1% in precision, recall, and mean average precision, respectively. Mazzia, et al. [21] implemented the YOLOv3-tiny architecture on affordable, energy-efficient embedded hardware, achieving a high detection accuracy of 83.64% and a frame rate of 30 fps for small objects, despite challenges like occlusion and complex backgrounds. Zhao, et al. [22] proposed an HDMNet target detection network for automated pear picking based on YOLOv8. Experimental results showed that HDMNet had advantages in parameter count, computational cost, and detection speed while maintaining high precision and accurate localization, with a parameter count as low as 12.9M, GFLOPs of 41.1, mAP of 75.7%, mAP50 of 93.6%, mAP75 of 70.2%, and FPS of 73.0. de Moraes, et al. [23] have utilized a YoloV7 detector and CBAM attention mechanism-based model, Yolo-Papaya, to create a dataset of 23,158 papaya fruit disease images, achieving an overall mAP of 86.2%, effectively enhancing the accuracy and practical applicability of fruit disease detection. Sun, et al. [24] developed a Focal Bottleneck Transformer Network (FBoT-Net) specifically for detecting small green apples. This network combines high-level semantic information and coarse-grained global context features with fine-grained local region features through a focal transformer layer, specifically targeting small apples that are similar in color to the background. Experimental results showed that FBoT-Net optimized small apple detection accuracy on the small apple dataset and demonstrated good generalization capability on

the Pascal VOC dataset. Sun, et al. [25] proposed an improved YOLOv5 model, called YOLO-P, designed for fast and accurate pear detection in complex orchard environments. This model optimizes feature extraction by introducing Shuffle blocks and inverted shuffle blocks and enhances the capture of key pear features using convolutional block attention modules (CBAM). Additionally, YOLO-P employs the Hard-Swish activation function and a weighted confidence loss function to improve small-object detection. Experimental results showed that YOLO-P outperformed other lightweight networks in comparative experiments, achieving an average precision (AP) of 97.6%, which is 1.8% higher than the original YOLOv5s, while reducing the model size by 39.4%, from 13.7MB to 8.3MB. Among the many deep learning-based object detection algorithms, the YOLO series is favored for its excellent performance and real-time capability [26,27]. Considering the YOLO series' high efficiency in image recognition and robustness in various complex environments, we selected the YOLO series algorithm for detecting young *red pears* to achieve fast and accurate detection results.

To address the issues such as small-target detection challenges, complex agricultural environments, limitations of existing models, and the lack of targeted optimizations in existing young fruit recognition and detection technologies, this study proposes a method for recognizing young *red pears* based on a lightweight YOLOv9s model. The model introduces the YOLOv10 model's detection head (v10detect) and combines the C2f module with the universal inverted bottleneck (UIB) block structure from MobileNetV4 to create a new C2fUIBELAN4 module. Additionally, it replaces the AConv module in YOLOv9s with the spatial-channel decoupled downsampling (SCDown) module, significantly enhancing the model's performance. These improvements increase the accuracy of the YOLOv9s model in small-object detection and provide robust technical support for the fast and accurate recognition of small targets, such as young *red pears*.

2. Materials and Methods

2.1. Dataset Construction

To develop an efficient and accurate young red pear recognition technology, we collected and annotated a dataset of young *red pears*. The images were collected from a pear orchard in Luoyang, Henan Province, using a Redmi K40 smartphone (Xiaomi Corporation, Beijing, China) with a camera resolution of 3000×3000 pixels. The images for our dataset were collected on 3 May 2024, starting from noon and continuing through to the early evening as the light conditions changed. This high-resolution setup ensured detailed capture of young *red pears* under various natural lighting conditions, contributing to the diversity of lighting conditions. Prior to data augmentation, a total of 395 high-quality images were collected. The collected images were preprocessed, including scaling, cropping, and normalization, to improve model training efficiency and enhance the model's adaptability to different conditions. Furthermore, we used specialized image annotation software (<https://pypi.org/project/labelImg/>, accessed on 8 August 2024) to accurately label the young *red pears* according to uniform standards, ensuring consistency and accuracy in data annotation. To further enhance the complexity of the dataset, we employed various data augmentation techniques, including flipping, scaling, and noise addition, to expand the dataset's diversity and improve the model's robustness. Ultimately, we constructed a dataset of 1580 images of young *red pears*, divided into training, validation, and test sets in an 8:1:1 ratio for comprehensive model evaluation and validation. Figure 1 displays sample images from the dataset, demonstrating the application of data augmentation techniques such as flipping, scaling, and noise addition.



Figure 1. Sample images of the red pear dataset.

2.2. Construction of the Young Red Pear Recognition Model

2.2.1. Analysis of the Original YOLOv9s Network Structure

The YOLOv9 model inherits the efficient object detection capabilities of the YOLO series, utilizing innovative network architecture and training strategies to significantly enhance its performance in image recognition tasks [28]. The YOLOv9 model introduces the programmable gradient information (PGI) framework, which provides detailed input information, ensuring accurate gradient information during the calculation of the objective function, thus effectively updating network weights [29]. The YOLOv9 model also employs a new lightweight network architecture, the Generalized Efficient Layer Aggregation Network (GELAN), which integrates CSPNet and ELAN neural network structures. This architecture achieves effective integration and circulation of feature information through gradient path planning, maintaining high-precision detection while reducing model computational complexity and optimizing parameter efficiency [30,31]. YOLOv9s, a variant of the YOLOv9 series, retains high-precision detection capabilities while reducing computational resource consumption and improving operational efficiency. Figure 2 illustrates the network structure of the YOLOv9s model.

In the YOLOv9s model, the RepNCSPeLAN4 module serves as the feature extraction and fusion module [32]. This module, designed based on the GELAN architecture, combines the strengths of CSPNet and ELAN structures to improve detection accuracy and enhance the model's ability to recognize small targets through multi-scale feature fusion. The structure of RepNCSPeLAN4 is shown in Figure 3. The AConv module is a key component in YOLOv9s for feature aggregation and channel expansion, allowing the network to capture multi-scale features while maintaining computational efficiency, which is crucial for improving small-object detection accuracy. The AConv module design not only enhances feature richness but also reduces model complexity by decreasing the number of parameters.

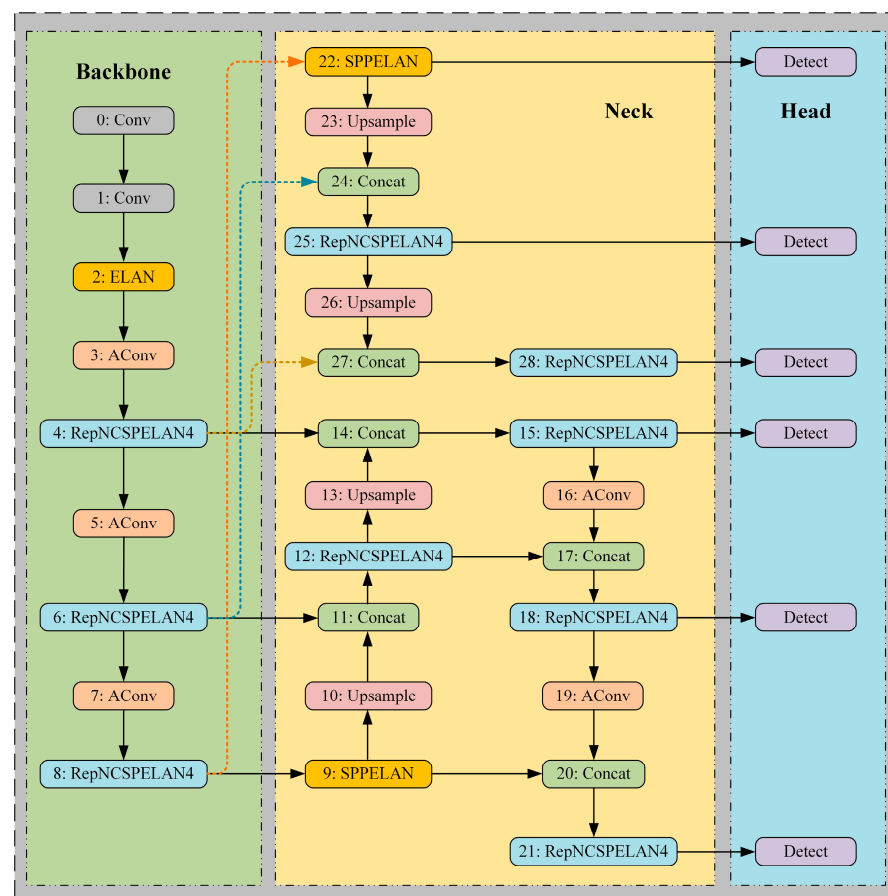


Figure 2. Structure of the YOLOv9s model. (Note: Conv is a convolution operation, ELAN is the efficient layer aggregation network module, RepNCSP4 is the reparametrized net with cross-stage partial connections and efficient layer aggregation network, AConv is the simplified downsampling convolution module, SPPELAN is the spatial pyramid pooling with enhanced local attention network, Upsample is the upsampling module, Concat is the feature connection module, and Detect is the detection head).

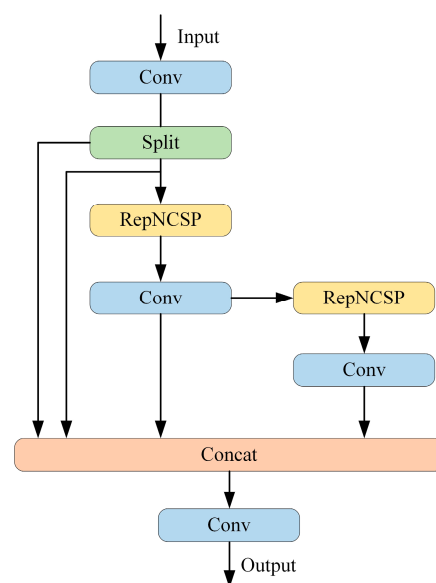


Figure 3. Structure of the RepNCSP4 module (RepNCSP is the reparametrized net with cross-stage partial connections).

2.2.2. Construction of the C2fUIBELAN4 Module

This study conducted an in-depth analysis of the YOLOv9s model, particularly its RepNCSPeLAN4 module. Despite its excellent performance in feature extraction and multi-scale fusion, the module requires a significant number of parameters, resulting in relatively high computational resource consumption, which is particularly limiting in small-object detection tasks. To address this issue, we first introduced the C2f module, as shown in Figure 4. The C2f module enhances the model's ability to capture rich feature information through feature fusion operations and multi-scale model channel adjustments while maintaining a lightweight structure, effectively improving detection accuracy and overall performance [33,34]. Additionally, we incorporated the UIB structure from MobileNetV4, which significantly reduces the number of model parameters and computational complexity while enhancing the ability to capture small-scale features by adjusting the configuration of convolution kernels, achieving lightweight and efficient performance [35]. Ultimately, we integrated the C2f module and UIB block to create the C2f_UIB module, as shown in Figure 5, replacing the RepNCSP structure in the RepNCSPeLAN4 module to form the new C2fUIBELAN4 module. This new module inherits the original YOLOv9s multi-scale feature fusion and path aggregation advantages while significantly reducing computational resource requirements through its lightweight design.

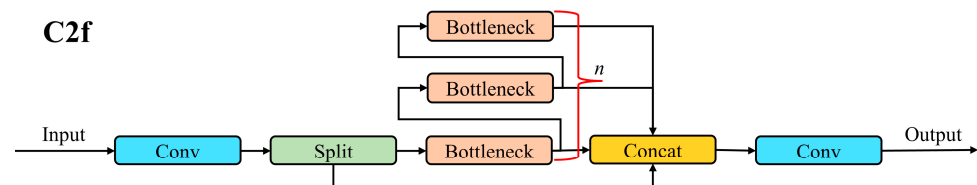


Figure 4. Structure of the C2f module.

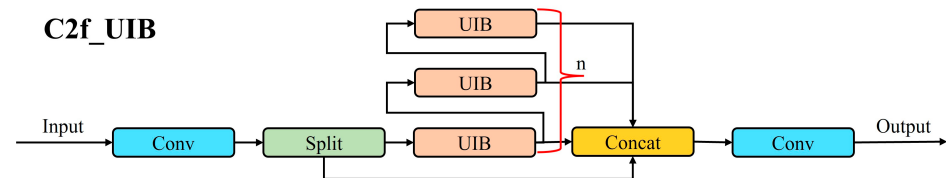


Figure 5. Structure of the C2f_UIB module. (Note: UIB is the universal inverted bottleneck module).

The introduction of the C2f module and the UIB structure not only enhances the performance of small-object detection but also promotes mutual enhancement through their design principles, enhancing the overall performance of the model. The C2f module, through feature fusion and channel adjustment, allows the model to capture richer feature information while maintaining a lightweight structure. The UIB structure, on the other hand, focuses on enhancing the capture of small-scale features by reducing the number of parameters and computational complexity. The complementary nature of these design principles makes the C2f_UIB module exceptional in small-object detection tasks while maintaining model efficiency.

Building upon the YOLOv9s model, the introduction of the C2f and UIB structures not only improves the performance of small-object detection but also reduces the model's computational resource requirements through lightweight design. This design strategy enhancement provides a new solution for the field of real-time object detection, especially in resource-constrained environments such as mobile devices or embedded systems, which has significant application value. With these improvements, the YOLOv9s model maintains high precision while also gaining better practicality and deployment flexibility.

2.2.3. SCDowN Module

Although the original AConv module in the YOLOv9s model played a crucial role, its relatively high computational cost limited the model's application in resource-constrained environments. In this study, we replaced the AConv module with the spatial-channel decoupled downsampling (SCDown) module from YOLOv10 to achieve more efficient feature capture and channel dimension expansion. The SCDown module enhances model efficiency and performance through spatial and channel decoupling operations, which helps to handle feature mapping more efficiently with reduced computational requirements. It first uses pointwise convolution to adjust the channel dimension, followed by depthwise convolution for spatial downsampling. This design not only reduces the number of parameters but also prevents excessive information loss, enabling the model to capture local features more accurately while maintaining low computational costs. In addition, the architecture of the SCDown module is designed to preserve information during downsampling, which is critical for the accurate detection of small objects. The preservation of local feature details allows for a more accurate representation of small objects within the detection framework. Integrating the SCDown module into our model also allows for multi-scale feature fusion, a fusion process that is critical. Through multi-scale feature fusion, the details of small targets are preserved, which enables the model to utilize high-resolution details and semantic information, and enhances the model's adaptability and generalization ability to various target scales, thus improving the detection accuracy of small objects at different scales [36]. These improvements make the model more suitable for small-object recognition applications. The structures of the AConv, SCDown, and Conv modules are shown in Figure 6.

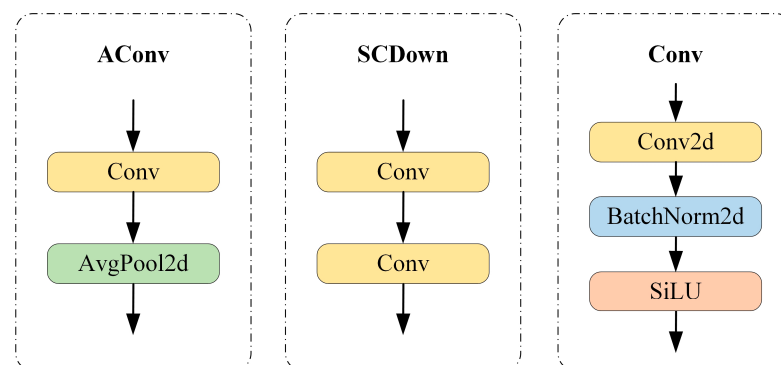


Figure 6. Structure of the AConv, SCDown, and Conv modules. (Note: AvgPool2d is a 2D average pooling operation, BatchNorm2d is a batch normalization operation, and SiLU is the activation function).

2.2.4. Improvement of the Detection Head

This study improved the detection head of the YOLOv9s model to enhance the detection accuracy of small targets by replacing it with the v10detect module. This improvement resulted in significant performance gains. The v10detect structure introduces a one-to-many and one-to-one detection strategy. During the training phase, the model receives abundant supervision signals through the one-to-many approach, while the one-to-one approach provides accurate predictions during inference, effectively avoiding reliance on non-maximum suppression and significantly improving detection efficiency [37]. Moreover, the v10detect structure adopts a consistent matching metric method, ensuring consistency across different allocation strategies, thereby enhancing model performance. The v10detect module captures rich feature information, particularly for small-target representation, achieving high-precision detection results while maintaining faster running speeds.

2.2.5. Young Red Pear Recognition Model

In this study, we developed a young red pear recognition model based on the YOLOv9s architecture, named YOLOv9s-Pear, specifically designed for the task of recognizing young *red pears*. By replacing the original AConv module with the SCDown module, the model not only achieves efficient feature capture and channel dimension expansion but also effectively reduces computational costs and improves the model's generalization ability across targets of different scales. Furthermore, we innovatively integrated the C2f module with the UIB block from MobileNetV4 to construct the C2fUIBELAN4 module, enhancing the model's accuracy in small-object detection tasks while maintaining efficient processing speed. Additionally, by incorporating the advanced detection head from YOLOv10, we further improved the model's performance. These innovations make the YOLOv9s-Pear model highly suitable for resource-constrained agricultural automation environments, providing robust technical support for the fast and accurate recognition of small targets such as young *red pears*.

2.3. Evaluation Metrics

To comprehensively evaluate the performance of different models in the task of recognizing young *red pears*, this study employs precision (P), recall (R), mean average precision (mAP50), and extended mean average precision (mAP50-95) as evaluation metrics. Precision measures the proportion of true positive samples among those predicted as positive by the model, directly reflecting the model's ability to accurately identify positive classes. Recall indicates the proportion of true positive samples correctly predicted among all actual positive samples, showing the model's capability to identify all positive classes. mAP50, as an indicator of the model's overall performance across different categories, is obtained by calculating the average of the average precisions (APs) for each category, helping to assess the model's overall recognition effectiveness. Meanwhile, mAP50-95 provides a more detailed performance evaluation, offering a comprehensive reflection of the model's adaptability to various conditions and accurately measuring the model's performance under different levels of strictness in matching. The specific calculation formulas are as follows:

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

$$mAP_{50} = \frac{1}{n} \sum_{i=1}^n AP_i \quad (3)$$

$$mAP_{50-95} = \frac{1}{n} \sum_{i=1}^n \frac{1}{91} \sum_{j=1}^{91} AP_{i,j} \quad (4)$$

where, TP represents the number of true positives, FP represents the number of false positives, and FN represents the number of false negatives. n is the number of categories, AP_i is the average precision for category i at an IoU threshold of 0.5, and $AP_{i,j}$ is the average precision for category i at an IoU threshold of $0.5 + 0.005 \times (j - 1)$. The number 91 represents the number of equally spaced points calculated for AP as the IoU threshold varies from 0.5 to 0.95, in intervals of 0.05.

3. Results

3.1. Experimental Environment

The experiments in this study were conducted on an Ubuntu 20.04 operating system, using version 2.0.1 of the PyTorch deep learning framework, CUDA version 11.8, Python version 3.10 as the development language, and Jupyter as our IDE. The central processing

unit (CPU) used in the experiments was an Intel (R) Xeon (R) Gold 5318Y CPU @ 2.10 GHz. The GPU used was an NVIDIA A16 with 15 GB of memory.

3.2. Ablation Experiments

During the ablation experiments on the YOLOv9s model, we found that replacing certain modules significantly enhanced the model's performance in recognizing young *red pears*. The results of the ablation experiments are shown in Table 1, and Figure 7 presents the parameter count and GFLOPs for the experiments.

Table 1. Accuracy results of ablation experiments.

Number	SCDown	C2FUIBELAN	v10detect	P	R	mAP50	mAP50-95
1	-	-	-	0.948	0.923	0.970	0.747
2	✓	-	-	0.964	0.911	0.977	0.764
3	-	✓	-	0.954	0.911	0.970	0.751
4	-	-	✓	0.962	0.958	0.989	0.833
5	✓	✓	-	0.964	0.971	0.991	0.834
6	✓	✓	✓	0.971	0.970	0.991	0.848

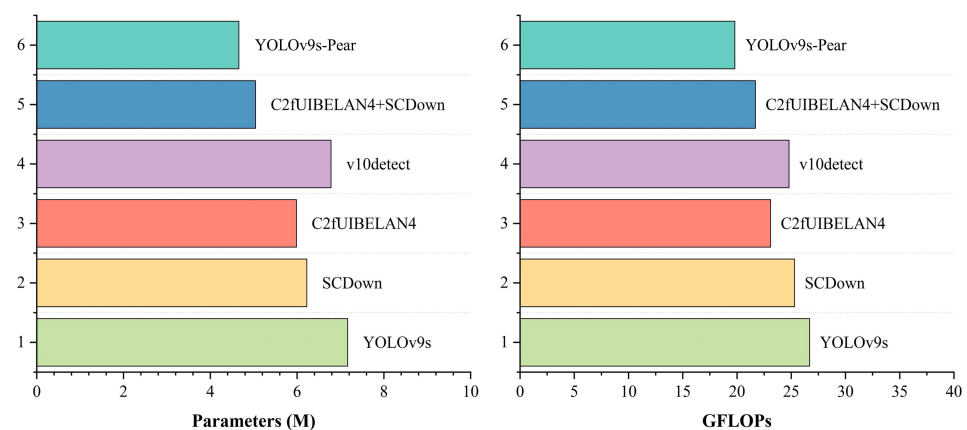


Figure 7. Number of parameters and GFLOPs for ablation experiment results.

As illustrated in Figure 7 and Table 1, replacing the AConv module with the SCDown module increased the model's precision (P) from 0.948 to 0.964, while the recall (R) remained stable. The mAP50 and mAP50-95 also improved, from 0.970 to 0.977 and from 0.747 to 0.764, respectively. The parameter count decreased to 6,225,203, and GFLOPs decreased to 25.3, indicating that the SCDown module played a crucial role in improving precision and effectively reducing the required parameter count and computational complexity. Additionally, when we replaced the RepNCSPeLan4 module in YOLOv9s with the C2FUIBELAN module, the improvement in accuracy was modest, but the parameter count decreased to 5,984,947 and GFLOPs decreased to 23.1, demonstrating that the C2FUIBELAN module positively impacted model performance by significantly reducing the parameter count and computational complexity. Replacing the YOLOv9s detection head with the v10detect module significantly increased the model's P and R, reaching 0.962 and 0.958, respectively. The mAP50 and mAP50-95 also showed significant improvements, increasing to 0.989 and 0.833, respectively, with a slight decrease in parameter count and GFLOPs. This suggests that the v10detect detection head greatly enhanced detection performance.

Furthermore, combining the SCDown, C2FUIBELAN, and v10detect improvements resulted in additional performance gains. In the combined experiment of SCDown and C2FUIBELAN, P and R increased to 0.964 and 0.971, respectively, while mAP50 and mAP50-95 rose to 0.991 and 0.834, respectively. The parameter count was 5,042,675, and GFLOPs were 21.7. In the combined experiment involving SCDown, C2FUIBELAN, and v10detect,

the evaluation metrics improved further, reaching 0.971, 0.970, 0.991, and 0.848, respectively, with the parameter count decreasing to 4,658,102 and GFLOPs decreasing to 19.8. These results demonstrate the significant positive impact of the combined improvements on model performance.

Figure 8 shows the accuracy trends from the ablation experiments, indicating that the accuracy curves for all models exhibit an upward trend, with the YOLOv9s-Pear model performing the best. During the early stages of training, the YOLOv9s-Pear model rapidly increased its detection accuracy. As training progressed, the rate of accuracy improvement slowed, and the accuracy stabilized at a high level. Compared to other models, the YOLOv9s-Pear model demonstrated a clear advantage in overall accuracy, showcasing excellent performance in the task of recognizing young *red pears*.

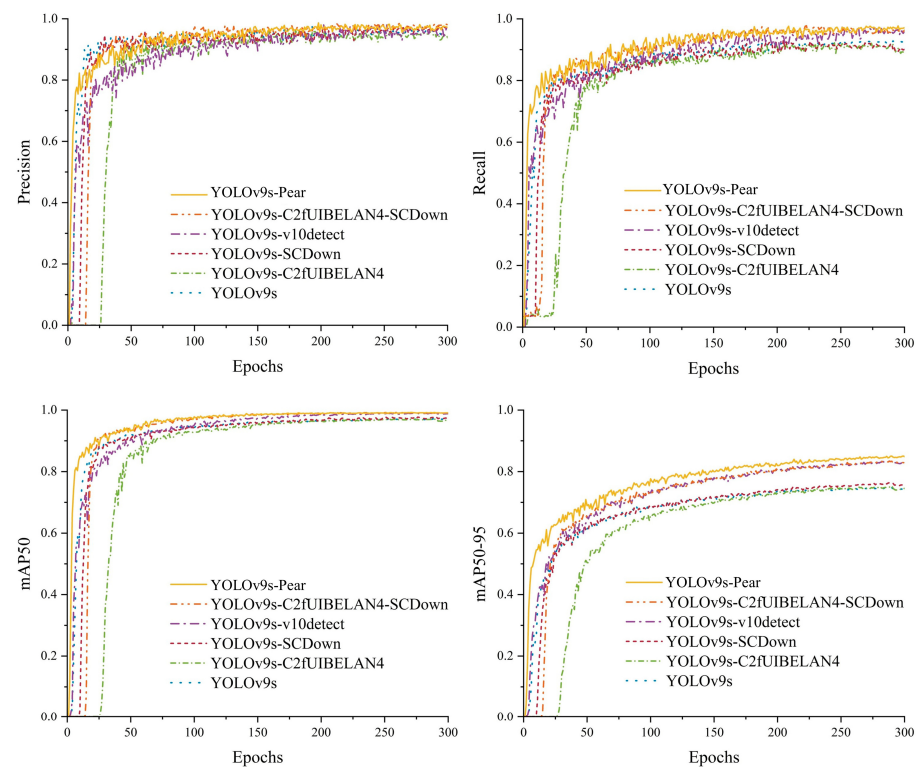


Figure 8. Precision change curves for ablation experiments.

Overall, the YOLOv9s-Pear model exhibited the best performance across all evaluation metrics. These improvements not only enhanced P and R but also maintained high mAP50 and mAP50-95, reduced complexity, increased detection speed, and improved the model's overall performance. This is crucial for the task of recognizing young *red pears*.

3.3. Comparison of Different Models

This study evaluated the performance of various YOLO series models in the task of recognizing young *red pears*. Table 2 presents the training accuracy results for different models, with the parameter count and GFLOPs shown in Figure 9 and the recognition results displayed in Figure 10. Table 3 shows the time required for training different models. The comprehensive evaluation of the YOLO series models on the task of young red pear recognition revealed that the YOLOv9s-Pear model achieved significant improvements in key performance metrics compared to SSD-ResNet18, RTDETR-ResNet18, YOLOv5s, YOLOv6s, YOLOv8s, and YOLOv9s models.

Specifically, compared to SSD-ResNet18, YOLOv9s-Pear improved precision (P) by 3.1%, from 0.942 to 0.971, and recall (R) by 3.5%, from 0.937 to 0.970. Compared to RTDETR-ResNet18, the improvements were 1.5% and 8.1%, respectively. Compared to YOLOv5s, the

improvements were 3.4% and 3.6%, respectively. Compared to YOLOv6s, the improvements were 2.8% and 2.9%, respectively. Compared to YOLOv8s, the improvements were 2.3% and 3.4%, respectively. And compared to YOLOv9s, the improvements were 2.4% and 5.1%, respectively. In terms of mAP50 and mAP50-95, YOLOv9s-Pear achieved 0.991 and 0.848, representing increases of 1.2% and 1.3% compared to SSD-ResNet18, 1.5% and 12.8% compared to RTDETR-ResNet18, 1.3% and 14.1% compared to YOLOv5s, 1.2% and 9.7% compared to YOLOv6s, 1.2% and 12.8% compared to YOLOv8s, and 2.2% and 13.5% compared to YOLOv9s.

Regarding model parameters, computational resource consumption, and total computational time, YOLOv9s-Pear had a parameter count of 4,658,102, which is approximately 29.1% lower than SSD-ResNet18, 85.6% lower than RTDETR-ResNet18, 49.3% lower than YOLOv5s, 71.3% lower than YOLOv6s, 57.9% lower than YOLOv8s, and 35.1% lower than YOLOv9s. Additionally, its GFLOPs were 19.8, which is 40.3% lower than SSD-ResNet18, 79.6% lower than RTDETR-ResNet18, 16.2% lower than YOLOv5s, 54.5% lower than YOLOv6s, 30.3% lower than YOLOv8s, and 26.2% lower than YOLOv9s. The total computation time of the YOLOv9s-Pear model was 12.1 ms, which was the fastest training speed among all the models; it was 5.4 ms faster than SSD-ResNet18, 29.1 ms faster than RTDETR-ResNet18, 1.0 ms faster than YOLOv5s, 1.1 ms faster than YOLOv6s, 0.7 ms faster than YOLOv8s, and 4.5 ms faster than YOLOv9s. These data indicate that YOLOv9s-Pear not only maintains high-efficiency target detection performance but also consumes significantly fewer resources, which is particularly important for applications that require model deployment in resource-constrained environments.

Table 2. Accuracy of training results for different models.

Model	P	R	mAP50	mAP50-95
SSD-ResNet18	0.942	0.937	0.979	0.748
RTDETR-ResNet18	0.957	0.897	0.976	0.752
YOLOv5s	0.938	0.936	0.978	0.743
YOLOv6s	0.945	0.942	0.979	0.773
YOLOv8s	0.949	0.938	0.979	0.752
YOLOv9s	0.948	0.923	0.970	0.747
YOLOv9s-Pear	0.971	0.970	0.991	0.848

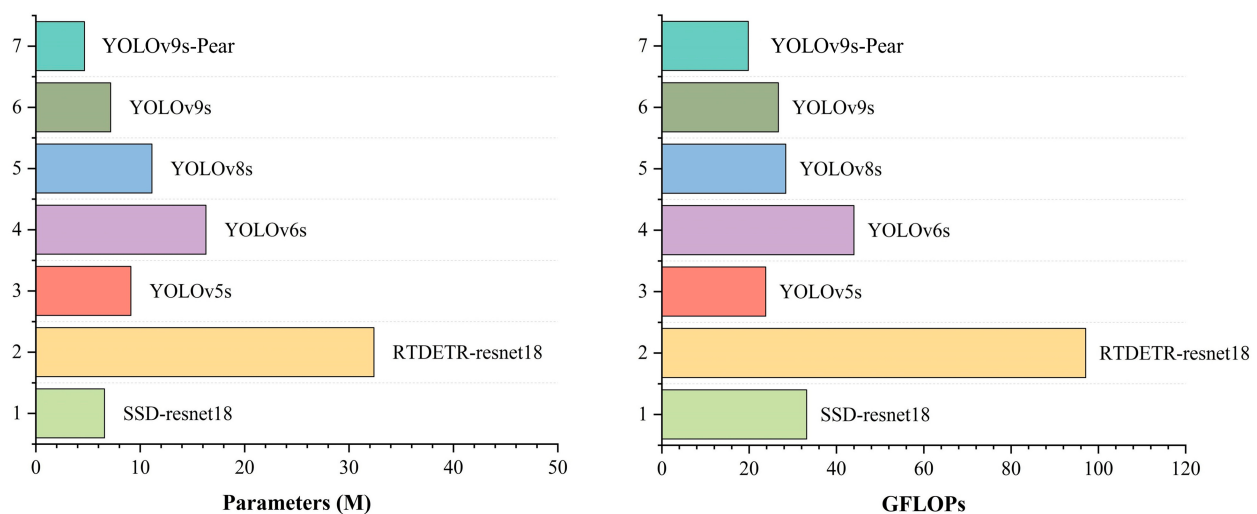
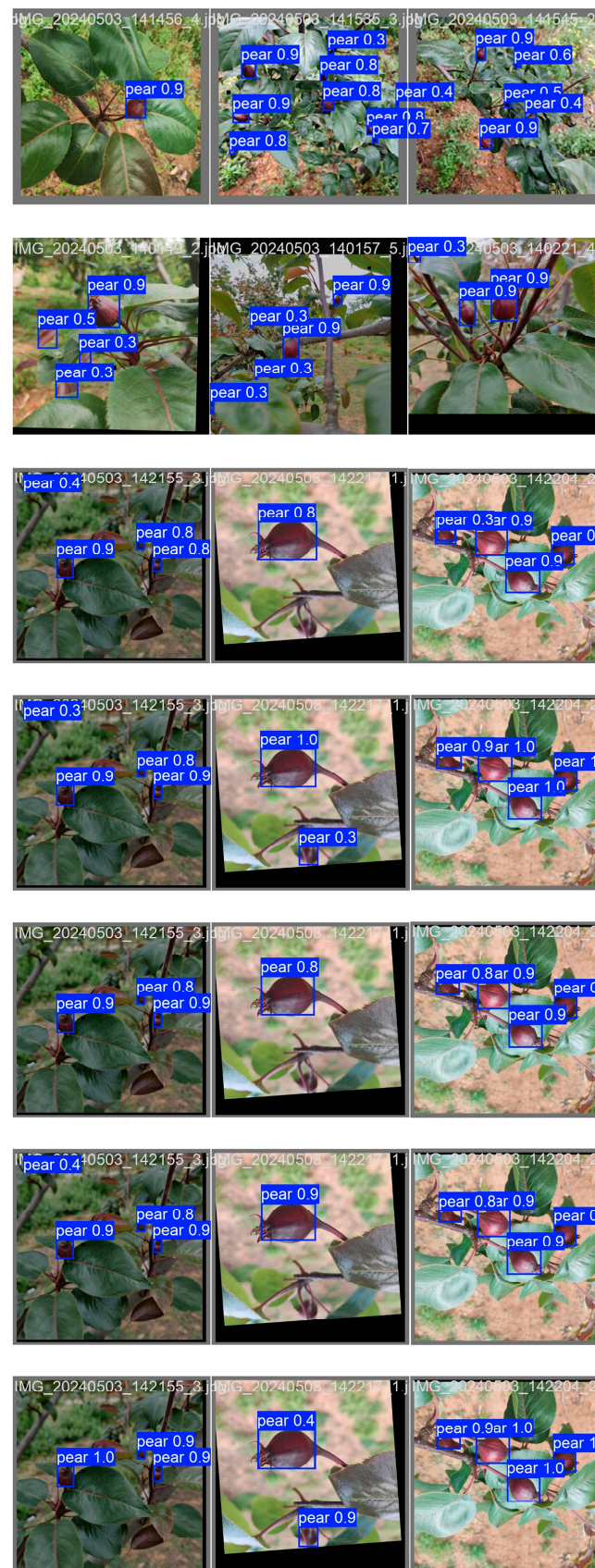


Figure 9. Number of parameters and GFLOPs for different model training results.



SSD-resnet18

RTDETR-resnet18

YOLOv5s

YOLOv6s

YOLOv8s

YOLOv9s

YOLOv9s-Pear

Figure 10. Plot of detection results for different models.

Table 3. Time required for training different models.

Model	Preprocess Time (ms)	Inference Time (ms)	Postprocess Time (ms)	Total Computational Time (ms)
SSD-ResNet18	0.8	15.6	1.1	17.5
RTDETR-ResNet18	0.5	40.5	0.2	41.2
YOLOv5s	0.8	10.4	1.9	13.1
YOLOv6s	0.8	11.4	1.0	13.2
YOLOv8s	0.8	11.2	0.8	12.8
YOLOv9s	0.8	15.1	0.7	16.6
YOLOv9s-Pear	0.3	11.6	0.2	12.1

In summary, the YOLOv9s-Pear model demonstrated superior performance across all evaluation metrics compared to the SSD-ResNet18, RTDETR-ResNet18, YOLOv5s, YOLOv6s, YOLOv8s, and YOLOv9s models, while also achieving significant reductions in parameter count and computational resource consumption. These improvements not only enhance the model's detection accuracy but also set a new technical benchmark for real-time object detection, providing strong technical support for agricultural automation and precision agriculture.

4. Discussion

In this ablation study, we conducted a series of improvements on the YOLOv9s model by replacing key modules and observing their impact on model performance. The experimental results demonstrate that these improvements significantly enhanced the model's performance in the task of young red pear recognition, while also reducing its computational complexity and parameter count.

First, the introduction of the SCDown module was one of the key improvements in this study. By decoupling spatial and channel dimensions during downsampling, this module effectively reduced the computational burden of the model while preserving critical spatial information. This design not only improved the model's accuracy but also decreased the parameter count and GFLOPs, which is particularly important for applications where the model needs to be deployed on resource-constrained devices. Second, while the replacement of the C2FUIBELAN module offered a limited increase in model accuracy, its effectiveness in reducing the model's parameter count and computational complexity is noteworthy. This improvement helps enhance the model's generalization capability, reduces the risk of overfitting, and contributes to the overall model lightweighting. Furthermore, the replacement of the detection head with v10detect brought about significant performance gains. The new detection head played a crucial role in improving the model's precision and recall, underscoring the importance of detection head design for overall model performance. By optimizing the detection head, we can achieve a more accurate localization and recognition of targets, which is critical for improving the model's reliability and accuracy in practical applications. Additionally, the combined application of SCDown, C2FUIBELAN, and v10detect not only further enhanced the model's performance but also demonstrated the synergistic effects between different improvements. This combination strategy improved model accuracy while reducing complexity, leading to a comprehensive enhancement of model performance.

In the cross-model comparison experiments, the YOLOv9s-Pear model demonstrated not only significant improvements in precision (P) and recall (R) but also notable practical implications for real-world deployments. Compared to the SSD-ResNet18, RTDETR-ResNet18, YOLOv5s, YOLOv6s, YOLOv8s, and YOLOv9s models, YOLOv9s-Pear enhanced P by 3.4% and R by 3.6% over YOLOv5s, which translates to a more robust detection capability, crucial for applications where high accuracy is paramount. The model also achieved substantial gains in mAP50 and mAP50-95, reaching 0.991 and 0.848, respectively, indicating a broader and more reliable detection range.

Moreover, the practical implications of these improvements are profound, especially in resource-constrained environments. YOLOv9s-Pear's parameter count is approximately 49.3% lower than YOLOv5s, and its GFLOPs are 16.2% lower, which directly translates to reduced computational demands. This is particularly significant for edge devices and mobile platforms where computational resources are limited. The lower parameter counts and GFLOPs of YOLOv9s-Pear suggest that it can be deployed more efficiently in scenarios with limited processing power, such as in IoT devices or in regions with poor infrastructure, without compromising on detection performance.

In this study, the enhanced YOLOv9s model demonstrates the capacity for precise identification of young red pear fruits, thereby aiding orchard managers in conducting efficient thinning operations. This ensures that the fruits receive adequate growth space and nutrients, consequently enhancing the overall quality of the produce. Furthermore, the model's accurate estimation of the number of young fruits provides robust data support for yield forecasting. This enables farmers to implement more refined water and fertilizer management strategies based on the projected outcomes. Particularly in agricultural environments where resources are constrained, such precision management is instrumental in reducing costs and increasing the operational efficiency of the orchard.

The overall evaluation results indicate that the YOLOv9s-Pear model excels in all key performance metrics. These improvements not only enhanced the model's precision and recall but also maintained high mAP50 and mAP50-95, while reducing complexity, increasing detection speed, and boosting overall performance. However, a deeper analysis of the errors reveals that the model's performance is slightly hindered in scenarios where young *red pears* are partially occluded. This limitation is likely due to a relative scarcity of occlusion cases in our current dataset, which we acknowledge requires further expansion. Additionally, while we have tailored the model to better recognize young *red pears*, there is room for further refinement to enhance its robustness against occlusions. This is crucial for the task of young red pear recognition and provides strong technical support for agricultural automation and precision agriculture.

In future research, we are committed to testing the applicability and generalization capability of our YOLOv9s model improvements on additional datasets, beyond the young red pear dataset we have constructed. This will involve collaboration with other research institutions and the use of public datasets to ensure a comprehensive evaluation of the model's performance across various conditions and crops. We will also focus on optimizing the model for a wider range of application scenarios, including different environmental conditions and agricultural contexts. Real-time performance evaluation and computational resource optimization will remain important areas of investigation to ensure the model's efficiency and practicality for on-field applications. Through these efforts, we aim to contribute to the advancement of real-time object detection technology, with the potential for broader applications in agricultural automation and precision farming.

5. Conclusions

This study proposed an improved YOLOv9s model for the recognition and detection of young *red pears*, achieving efficient recognition of small targets. The proposed YOLOv9s-Pear model can efficiently recognize young *red pears* with strong generalization capability and low computational cost. The conclusions of this study are as follows:

1. The introduction of SCDwn, C2FUIBELAN, and v10detect modules all contributed to improving the model's detection performance, with the v10detect module having the most significant impact on accuracy improvement and the C2FUIBELAN module being the most effective in reducing the model's parameter count and computational resource consumption.
2. The combination of improvement strategies significantly enhanced the detection performance of the YOLOv9s model. When applying the SCDwn, C2FUIBELAN, and v10detect modules together, the model's performance further improved. The YOLOv9s-Pear model achieved $P = 0.971$, $R = 0.970$, $mAP50 = 0.991$, and $mAP50-95 = 0.848$.

3. Compared to the SSD-ResNet18, RTDETR-ResNet18, YOLOv5s, YOLOv6s, YOLOv8s, and YOLOv9s models, the YOLOv9s-Pear model demonstrated superior performance, achieving higher precision while maintaining lower parameter counts and computational complexity.

The results of this study prove the efficiency of the SCDown, C2FUIBELAN, and v10detect modules. The SCDown module significantly enhanced the model's adaptability to multi-scale targets and effectively reduced computational cost. The introduction of the C2FUIBELAN module further optimized the model's ability to capture small-scale features, achieving higher detection accuracy and a lightweight structure. The integration of the v10detect module maintained high precision while significantly reducing the computational resources and parameter count required by the model. The combined application of these three modules significantly reduced the computational complexity and parameter count of the YOLOv9s-Pear model while enhancing its ability to detect small targets. This made the model more adaptable and flexible while maintaining a lightweight structure, providing a new solution for agricultural automation. Future research could explore the model's application in a broader dataset to enhance its practicality and further optimize the model structure to achieve higher precision and lower complexity in object detection, strengthening its performance in practical applications.

Author Contributions: Conceptualization, Y.S. and Z.D.; methodology, S.Q.; software, S.Q.; validation, L.Z.; formal analysis, F.W.; investigation, L.Z.; resources, Y.S.; data curation, X.Y.; writing—original draft, S.Q.; writing—review and editing, Y.S. and F.W.; visualization, X.Y.; supervision, Z.D.; project administration, Z.D.; funding acquisition, Z.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Ministry of Agriculture and Rural Affairs Academy of Agricultural Planning and Engineering Independent Research and Development Project (No. QX202412), National Natural Science Foundation of China (No. 52309050), Key R&D and Promotion Projects in Henan Province (Science and Technology Development) (No. 232102110264, No.222102110452), and Key Scientific Research Projects of Colleges and Universities in Henan Province (No. 24B416001, No. 22B416002).

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Acknowledgments: We would like to express our gratitude to the orchards of Luoyang City, Henan Province for their support of our experimental data collection. Thank you to the Academy of Agricultural Planning and Engineering, Ministry of Agriculture and Rural Affairs, where Shi Yi works, for providing assistance.

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

References

1. Onishi, Y.; Yoshida, T.; Kurita, H.; Fukao, T.; Arihara, H.; Iwai, A. An automated fruit harvesting robot by using deep learning. *Robomech J.* **2019**, *6*, 13. [\[CrossRef\]](#)
2. Duong, L.T.; Nguyen, P.T.; Di Sipio, C.; Di Ruscio, D. Automated fruit recognition using EfficientNet and MixNet. *Comput. Electron. Agric.* **2020**, *171*, 105326. [\[CrossRef\]](#)
3. Gené-Mola, J.; Ferrer-Ferrer, M.; Gregorio, E.; Blok, P.M.; Hemming, J.; Morros, J.-R.; Rosell-Polo, J.R.; Vilaplana, V.; Ruiz-Hidalgo, J. Looking behind occlusions: A study on amodal segmentation for robust on-tree apple fruit size estimation. *Comput. Electron. Agric.* **2023**, *209*, 107854. [\[CrossRef\]](#)
4. Shi, Y.; Qing, S.; Zhao, L.; Wang, F.; Yuwen, X.; Qu, M. YOLO-Peach: A High-Performance Lightweight YOLOv8s-Based Model for Accurate Recognition and Enumeration of Peach Seedling Fruits. *Agronomy* **2024**, *14*, 1628. [\[CrossRef\]](#)
5. Kukunda, C.B.; Duque-Lazo, J.; González-Ferreiro, E.; Thaden, H.; Kleinn, C. Ensemble classification of individual Pinus crowns from multispectral satellite imagery and airborne LiDAR. *Int. J. Appl. Earth Obs. Geoinf.* **2018**, *65*, 12–23. [\[CrossRef\]](#)
6. Jiang, M.; Song, L.; Wang, Y.; Li, Z.; Song, H. Fusion of the YOLOv4 network model and visual attention mechanism to detect low-quality young apples in a complex environment. *Precis. Agric.* **2022**, *23*, 559–577. [\[CrossRef\]](#)
7. Dorj, U.-O.; Lee, M.; Yun, S.-S. An yield estimation in citrus orchards via fruit detection and counting using image processing. *Comput. Electron. Agric.* **2017**, *140*, 103–112. [\[CrossRef\]](#)

8. Gao, F.; Fang, W.; Sun, X.; Wu, Z.; Zhao, G.; Li, G.; Li, R.; Fu, L.; Zhang, Q. A novel apple fruit detection and counting methodology based on deep learning and trunk tracking in modern orchard. *Comput. Electron. Agric.* **2022**, *197*, 107000. [\[CrossRef\]](#)
9. Villacrés, J.; Viscaino, M.; Delpiano, J.; Vougioukas, S.; Cheein, F.A. Apple orchard production estimation using deep learning strategies: A comparison of tracking-by-detection algorithms. *Comput. Electron. Agric.* **2023**, *204*, 107513. [\[CrossRef\]](#)
10. Dubey, S.R.; Jalal, A.S. Apple disease classification using color, texture and shape features from images. *Signal Image Video Process.* **2016**, *10*, 819–826. [\[CrossRef\]](#)
11. Zhang, Y.; Shi, N.; Zhang, H.; Zhang, J.; Fan, X.; Suo, X. Appearance quality classification method of Huangguan pear under complex background based on instance segmentation and semantic segmentation. *Front. Plant Sci.* **2022**, *13*, 914829. [\[CrossRef\]](#) [\[PubMed\]](#)
12. Saranya, N.; Srinivasan, K.; Pravin Kumar, S.; Rukkumani, V.; Ramya, R. Fruit classification using traditional machine learning and deep learning approach. In Proceedings of the Computational Vision and Bio-Inspired Computing: ICCVIB 2019, Coimbatore, India, 25–26 September 2019; pp. 79–89.
13. Yamamoto, K.; Guo, W.; Yoshioka, Y.; Ninomiya, S. On plant detection of intact tomato fruits using image analysis and machine learning methods. *Sensors* **2014**, *14*, 12191–12206. [\[CrossRef\]](#) [\[PubMed\]](#)
14. Archana, R.; Jeevaraj, P.E. Deep learning models for digital image processing: A review. *Artif. Intell. Rev.* **2024**, *57*, 11. [\[CrossRef\]](#)
15. Bargoti, S.; Underwood, J.P. Image segmentation for fruit detection and yield estimation in apple orchards. *J. Field Robot.* **2017**, *34*, 1039–1060. [\[CrossRef\]](#)
16. Vishnoi, V.K.; Kumar, K.; Kumar, B.; Mohan, S.; Khan, A.A. Detection of apple plant diseases using leaf images through convolutional neural network. *IEEE Access* **2022**, *11*, 6594–6609. [\[CrossRef\]](#)
17. Gai, R.; Chen, N.; Yuan, H. A detection algorithm for cherry fruits based on the improved YOLO-v4 model. *Neural Comput. Appl.* **2023**, *35*, 13895–13906. [\[CrossRef\]](#)
18. Xiong, J.; Yu, D.; Liu, S.; Shu, L.; Wang, X.; Liu, Z. A review of plant phenotypic image recognition technology based on deep learning. *Electronics* **2021**, *10*, 81. [\[CrossRef\]](#)
19. Yang, R.; Hu, Y.; Yao, Y.; Gao, M.; Liu, R. Fruit Target Detection Based on BCo-YOLOv5 Model. *Mob. Inf. Syst.* **2022**, *2022*, 8457173. [\[CrossRef\]](#)
20. Bai, Y.; Yu, J.; Yang, S.; Ning, J. An improved YOLO algorithm for detecting flowers and fruits on strawberry seedlings. *Biosyst. Eng.* **2024**, *237*, 1–12. [\[CrossRef\]](#)
21. Mazzia, V.; Khaliq, A.; Salvetti, F.; Chiaberge, M. Real-time apple detection system using embedded systems with hardware accelerators: An edge AI application. *IEEE Access* **2020**, *8*, 9102–9114. [\[CrossRef\]](#)
22. Zhao, P.; Zhou, W.; Na, L. High-precision object detection network for automate pear picking. *Sci. Rep.* **2024**, *14*, 14965. [\[CrossRef\]](#)
23. De Moraes, J.L.; de Oliveira Neto, J.; Badue, C.; Oliveira-Santos, T.; de Souza, A.F. Yolo-papaya: A papaya fruit disease detector and classifier using cnns and convolutional block attention modules. *Electronics* **2023**, *12*, 2202. [\[CrossRef\]](#)
24. Sun, M.; Zhao, R.; Yin, X.; Xu, L.; Ruan, C.; Jia, W. FBoT-Net: Focal bottleneck transformer network for small green apple detection. *Comput. Electron. Agric.* **2023**, *205*, 107609. [\[CrossRef\]](#)
25. Sun, H.; Wang, B.; Xue, J. YOLO-P: An efficient method for pear fast detection in complex orchard picking environment. *Front. Plant Sci.* **2023**, *13*, 1089454. [\[CrossRef\]](#)
26. Xue, C.; Xia, Y.; Wu, M.; Chen, Z.; Cheng, F.; Yun, L. EL-YOLO: An efficient and lightweight low-altitude aerial objects detector for onboard applications. *Expert Syst. Appl.* **2024**, *256*, 124848. [\[CrossRef\]](#)
27. Magalhães, S.A.; Castro, L.; Moreira, G.; Dos Santos, F.N.; Cunha, M.; Dias, J.; Moreira, A.P. Evaluating the single-shot multibox detector and YOLO deep learning models for the detection of tomatoes in a greenhouse. *Sensors* **2021**, *21*, 3569. [\[CrossRef\]](#)
28. Wang, C.-Y.; Yeh, I.-H.; Liao, H.-Y.M. Yolov9: Learning what you want to learn using programmable gradient information. *arXiv* **2024**, arXiv:2402.13616.
29. An, R.; Zhang, X.; Sun, M.; Wang, G. GC-YOLOv9: Innovative smart city traffic monitoring solution. *Alex. Eng. J.* **2024**, *106*, 277–287. [\[CrossRef\]](#)
30. Shi, Y.; Li, S.; Liu, Z.; Zhou, Z.; Zhou, X. MTP-YOLO: You only look once based maritime tiny person detector for emergency rescue. *J. Mar. Sci. Eng.* **2024**, *12*, 669. [\[CrossRef\]](#)
31. Vo, H.-T.; Mui, K.C.; Thien, N.N.; Tien, P.P. Automating Tomato Ripeness Classification and Counting with YOLOv9. *Int. J. Adv. Comput. Sci. Appl.* **2024**, *15*, 1892–1905. [\[CrossRef\]](#)
32. Li, J.; Feng, Y.; Shao, Y.; Liu, F. IDP-YOLOV9: Improvement of Object Detection Model in Severe Weather Scenarios from Drone Perspective. *Appl. Sci.* **2024**, *14*, 5277. [\[CrossRef\]](#)
33. Chen, Y.; Zhan, S.; Cao, G.; Li, J.; Wu, Z.; Chen, X. C2f-Enhanced YOLOv5 for Lightweight Concrete Surface Crack Detection. In Proceedings of the 2023 International Conference on Advances in Artificial Intelligence and Applications, Wuhan, China, 18–20 November 2023; pp. 60–64.
34. Zhu, Q.; Ma, K.; Wang, Z.; Shi, P. YOLOv7-CSAW for maritime target detection. *Front. Neurobotics* **2023**, *17*, 1210470. [\[CrossRef\]](#)
35. Qin, D.; Lechner, C.; Delakis, M.; Fornoni, M.; Luo, S.; Yang, F.; Wang, W.; Banbury, C.; Ye, C.; Akin, B. MobileNetV4-Universal Models for the Mobile Ecosystem. *arXiv* **2024**, arXiv:2404.10518.

36. Hussain, M. YOLOv5, YOLOv8 and YOLOv10: The Go-To Detectors for Real-time Vision. *arXiv* **2024**, arXiv:2407.02988.
37. Sundaresan Geetha, A.; Alif, M.A.R.; Hussain, M.; Allen, P. Comparative Analysis of YOLOv8 and YOLOv10 in Vehicle Detection: Performance Metrics and Model Efficacy. *Vehicles* **2024**, *6*, 1364–1382. [[CrossRef](#)]

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.