

Article

BHI-YOLO: A Lightweight Instance Segmentation Model for Strawberry Diseases

Haipeng Hu ^{1,2}, Mingxia Chen ^{1,2,*}, Luobin Huang ^{1,2}  and Chi Guo ^{1,2}

¹ Key Laboratory of Advanced Manufacturing and Automation Technology, Guilin University of Technology, Education Department of Guangxi Zhuang Autonomous Region, Guilin 541006, China; 2120221181@glut.edu.cn (H.H.); 2120221178@glut.edu.cn (C.G.)

² Guangxi Engineering Research Center of Intelligent Rubber Equipment, Guilin University of Technology, Guilin 541006, China

* Correspondence: wjunt@sohu.com

Abstract: In complex environments, strawberry disease segmentation models face challenges, such as segmentation difficulties, excessive parameters, and high computational loads, making it difficult for these models to run effectively on devices with limited computational resources. To address the need for efficient running on low-power devices while ensuring effective disease segmentation in complex scenarios, this paper proposes BHI-YOLO, a lightweight instance segmentation model based on YOLOv8n-seg. First, the Universal Inverted Bottleneck (UIB) module is integrated into the backbone network and merged with the C2f module to create the C2f_UIB module; this approach reduces the parameter count while expanding the receptive field. Second, the HS-FPN is introduced to further reduce the parameter count and enhance the model's ability to fuse features across different levels. Finally, by integrating the Inverted Residual Mobile Block (iRMB) with EMA to design the iRMA, the model is capable of efficiently combining global information to enhance local information. The experimental results demonstrate that the enhanced instance segmentation model for strawberry diseases achieved a mean average precision (mAP@50) of 93%. Compared to YOLOv8, which saw a 2.3% increase in mask mAP, the improved model reduced parameters by 47%, GFLOPs by 20%, and model size by 44.1%, achieving a relatively excellent lightweight effect. This study combines lightweight architecture with enhanced feature fusion, making the model more suitable for deployment on mobile devices, and provides a reference guide for strawberry disease segmentation applications in agricultural environments.



Citation: Hu, H.; Chen, M.; Huang, L.; Guo, C. BHI-YOLO: A Lightweight Instance Segmentation Model for Strawberry Diseases. *Appl. Sci.* **2024**, *14*, 9819. <https://doi.org/10.3390/app14219819>

Academic Editor: Piotr Gas

Received: 21 August 2024

Revised: 19 October 2024

Accepted: 25 October 2024

Published: 27 October 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/>).

Keywords: strawberry; disease; YOLOv8; UIB; HS-FPN; iRMA

1. Introduction

Strawberries originate from Europe and are beloved for their vibrant red color, rich fruity aroma, and sweet taste. According to statistics from the FAO [1], in 2022, the global area planted with strawberries was 397,603 hectares, with a production of 9,569,865 tons. Strawberries are among the most extensively grown fruit species worldwide. However, the growth of strawberries can be significantly threatened by biotic or abiotic factors, which pose a great risk to both the productivity and quality of strawberries. One of the most significant threats is the presence of various diseases affecting strawberries, such as gray mold, anthracnose fruit rot, leaf spot disease, etc. Therefore, effective identification and diagnosis of strawberry diseases are crucial aspects of disease prevention and control in strawberries. Traditional manual disease detection methods rely on professional technicians. However, with the rapid expansion of strawberry cultivation areas, the efficiency and real-time nature of this approach are gradually becoming insufficient to meet demands. In recent years, with advancements in artificial intelligence technology, especially the significant breakthroughs in convolutional neural networks [2,3], instance segmentation technology, as an important innovation that combines the functionalities of object detection and semantic segmentation,

is gradually making a significant impact in the field of computer vision. This technology not only accurately identifies and locates multiple similar objects within an image but also performs detailed analysis and labeling at the pixel level. Instance segmentation technology has achieved significant advancements in various fields, such as medical image analysis [4], industrial automation [5], autonomous driving [6], and agricultural disease control [7–10]. Compared to traditional neural networks, instance segmentation technology notably enhances the accuracy and efficiency of systems, providing robust support for precise identification and automated processing in complex scenarios.

Due to advancements in agricultural technology, computer vision is increasingly utilized in both research and practical applications within the agricultural sector. As early as 1985, in his PhD thesis, Sarkar [11] used computer vision to classify tomatoes based on size, shape, color, and surface defects. These days, numerous CNNs have found successful applications in the field of agriculture, such as the one-stage algorithm YOLO [12] series (YOLOv3 [13], YOLOv5 [14], YOLOv6 [15], YOLOv8, etc.); YOLACT [16]; SSD [17]; and the two-stage algorithms R-CNN [18], Faster R-CNN [19], Mask R-CNN [20], etc. Specifically, Jiang et al. [21] utilized an improved SSD model to detect five common apple diseases in complex scenes, achieving a mAP of 78.80% on the ALDD. This method has lower performance in multi-scale disease detection in complex scenes. Sapkota et al. [22] utilized the YOLOv8 algorithm and geometric shape fitting techniques to accurately measure the size of unripe green apples. The study demonstrated that the performance using the Azure Kinect sensor was significantly better than that of the Intel RealSense D435i (Intel, Santa Clara, CA, USA), showcasing the potential of this technology in orchard management. Fuentes et al. [23] utilized object detection algorithms to achieve recognition and localization of tomato diseases. Their proposed model effectively identifies nine different categories of tomato diseases in challenging scenarios but faces difficulties in meeting lightweight demands. Yao et al. [24] adopted a refined Mask R-CNN to segment and identify diseases affecting peach plants. They employed focal loss to address sample difficulty and imbalance issues, thereby improving the recognition rate and segmentation accuracy of peach diseases. This algorithm has many parameters, which makes it difficult to deploy in resource-limited environments. Afzaal et al. [25], utilizing enhanced learning strategies derived from the widely used Mask R-CNN method, conducted instance segmentation of seven types of diseases in strawberries, ultimately achieving an 82.4% mAP. This algorithm also suffers from a high number of parameters. Huang et al. [26] used YOLACT++ for image segmentation of northern corn leaf blight, achieving a comprehensive evaluation score of 98.36%. Although this method has high segmentation performance, its single-scale segmentation approach makes it challenging to guarantee the same effectiveness for multi-scale disease segmentation in complex scenes. Kumar et al. [27] proposed an improved YOLOv5 algorithm for segmenting different regions of rice leaves using DenseNet-201. They utilized a Bi-FAPN for feature extraction, achieving a mAP of 82.8% on the rice leaf disease dataset. Lu et al. [28] proposed a method that combines high-resolution RGB images (2048×1536 pixels) and NIR images, achieving an 87.18% mAP in strawberry defect detection. Tamrakar et al. [29] proposed a lightweight, improved YOLOv5s-CGHOSTnet model. By optimizing the network structure, this method significantly enhances both detection accuracy and efficiency, providing new insights into the classification of strawberry ripeness. Mi et al. [30] combined YOLOv9 with Swin Transformer to develop a complex model for strawberry ripeness detection, achieving an average accuracy of 87.3%. He et al. [31] proposed an automatic disease detection method for strawberry leaves based on an improved YOLOv8 model, achieving an accuracy of 89.7% in identifying diseases on strawberry leaves. Xie et al. [32] proposed an improved algorithm based on the YOLOv8 model, optimizing the Convolutional Block Attention Module (CBAM), an ultra-lightweight and effective dynamic upsampler (DySample), and Omni-Dimensional Dynamic Convolution (ODConv), significantly enhancing the real-time detection capability of strawberry diseases.

Although the aforementioned methods have shown good results for plant disease detection, various types of strawberry diseases, along with differences in pathogens and con-

texts, cause these diseases to exhibit varying sizes and characteristics, leading to multi-scale issues. When dealing with multi-scale issues of plant diseases in complex environments, these detection models typically have a high parameter count and large computational requirements. This results in significantly increased deployment and operational costs, thereby reducing the efficiency and feasibility of practical applications. This paper focuses on strawberry disease detection in complex environments, aiming to develop a deep learning detection model, design a lightweight architecture, and bolster the accuracy and robustness of disease segmentation. Based on YOLOv8n-seg, this paper introduces BHI-YOLO, a lightweight approach to instance segmentation. While ensuring high segmentation effectiveness, the model significantly reduces both the parameter count and the measured GFLOPs, thereby achieving efficient instance segmentation performance.

This paper designs the UIB [33] and HS-FPN [34] to make the YOLOv8 model more lightweight. Extensive comparative experiments with other modules are conducted to validate that these two modules enable the improved YOLOv8 model to achieve higher performance and a smaller model size. Specifically, UIB incorporates depthwise separable convolutions to reduce the number of parameters while retaining key features. HS-FPN enhances model robustness by integrating high-level and low-level features, further reducing the model's size. Finally, the design of the iRMA module effectively aggregates global information to complement local information, combining the advantages of iRMB [35] and EMA [36], and significantly improves system performance while maintaining the lightweight characteristics of the model. The BHI-YOLO model, which integrates these modules, demonstrates exceptional performance in strawberry disease segmentation tasks while minimizing model size, providing an effective solution for advancing lightweight model designs. This makes the model easier to deploy and reduces operational and maintenance costs.

2. Materials and Methods

2.1. Strawberry Dataset

This experiment selected a dataset of 2500 public images of strawberry diseases [25]. The dataset includes seven common diseases affecting strawberries. Additionally, 500 images of healthy strawberry fruits and leaves were added to enhance the recognition of strawberry health and disease in images. The strawberry disease pictures are all from natural fields, providing close-ups, long-range shots, and different angles of the strawberries. Examples of strawberry diseases are shown in Figure 1.

In this study, the distribution of seven strawberry diseases accurately reflects their occurrence in real-world environments. The ratio of healthy samples to diseased samples does not truly represent the actual occurrence in these environments. Healthy samples primarily serve as a control group, helping the model more effectively differentiate between healthy and diseased samples, thereby improving overall performance. However, an excessive number of healthy samples may result in model overfitting, which could reduce its generalization ability. Additionally, since the characteristics of healthy strawberries are relatively stable, this study selects an appropriate number of healthy and diseased samples to construct the experimental dataset. The dataset was annotated using the Labelme tool (v. 5.3.1), with each annotated image containing information on at least one type of strawberry disease. Therefore, the total number of strawberry diseases is slightly higher than 2500. The labeled data were divided based on a 7:2:1 ratio, a widely used method that effectively balances the data from each part and is able to reasonably assess the model's generalization ability. Specific details are provided in Table 1.

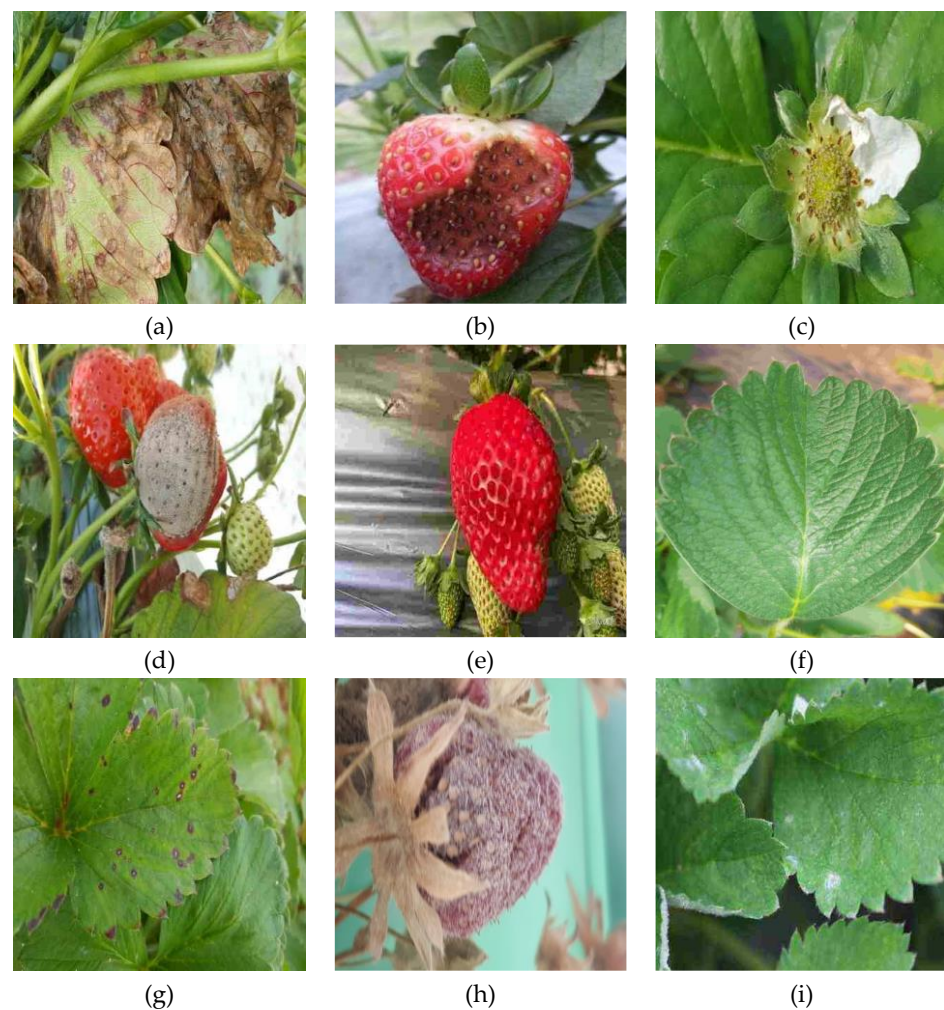


Figure 1. Nine images of healthy and diseased strawberries: (a) Angular Leafspot; (b) Anthracnose Fruit Rot; (c) Blossom Blight; (d) Gray Mold; (e) Healthy Fruit; (f) Healthy Leaf; (g) Leaf Spot; (h) Powdery Mildew Fruit; (i) Powdery Mildew Leaf.

Table 1. Dataset details.

Categories	Training Set	Validation Set	Test Set
Healthy Fruit	176	57	17
Healthy Leaf	178	50	22
Powdery Mildew Fruit	105	19	17
Angular Leafspot	302	93	40
Leaf Spot	442	107	66
Gray Mold	310	110	57
Powdery Mildew Leaf	367	115	51
Anthracnose Fruit Rot	78	13	9
Blossom Blight	150	37	21
Total	2108	601	300

2.2. YOLOv8 Algorithm

YOLOv8 is a real-time multitask model that was open-sourced by the Ultralytics team and numerous contributors in 2023. Compared to YOLOv5, it further enhances model performance and versatility, featuring superior capabilities for edge device deployment and industrial applications. The architecture of YOLOv8 includes three key components, namely, the backbone, neck, and head, as illustrated in Figure 2.

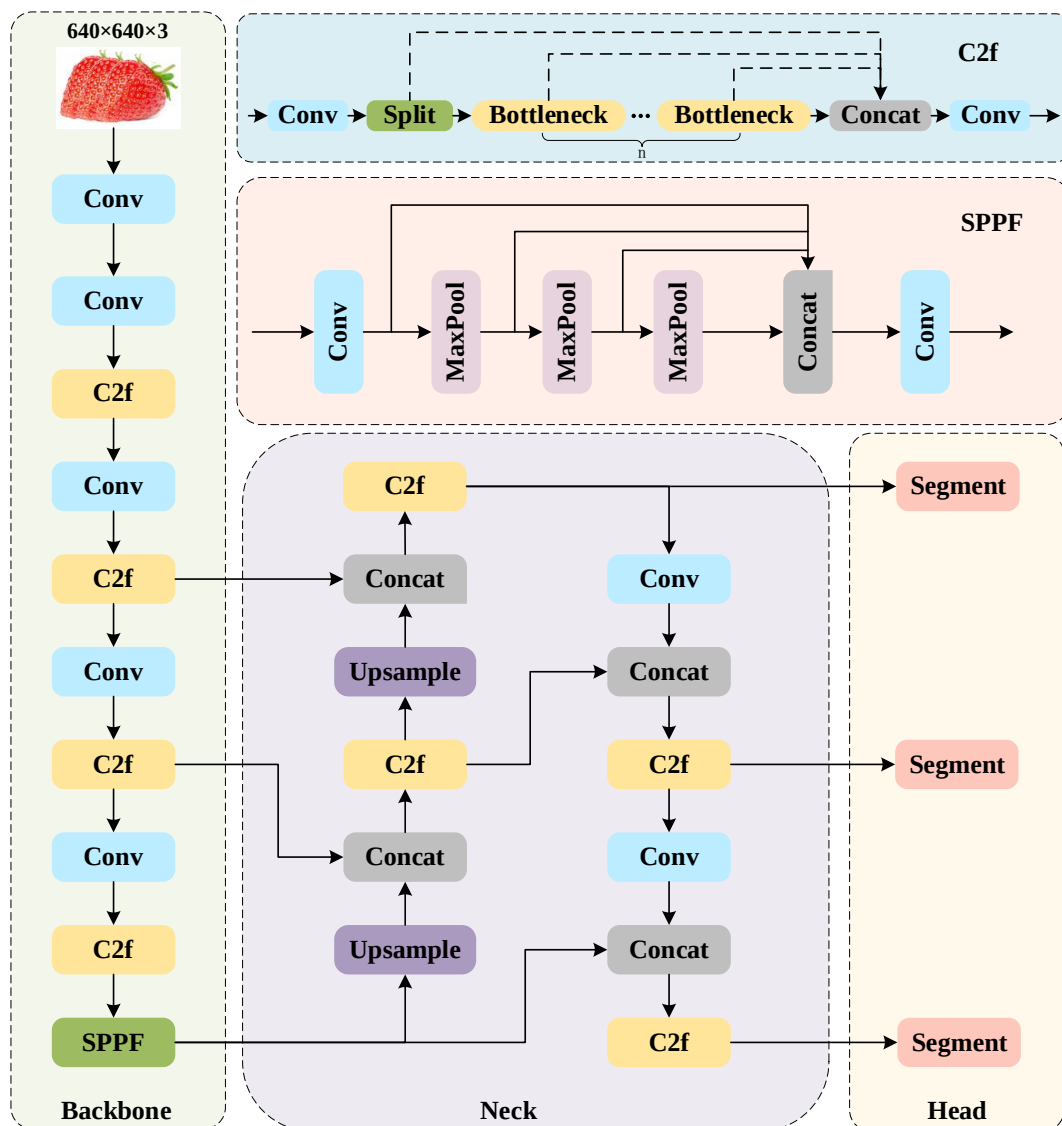


Figure 2. YOLOv8n-Seg structure diagram.

From the network structure of YOLOv8-Seg, we can see that it continues the network concept of YOLOv5. YOLOv8 offers optional mosaic data augmentation at the input, enriching the image backgrounds while reducing GPU dependency. The input data are processed by several convolutional and pooling layers that capture image features and progressively convert the input into a multi-level, rich feature representation. This process efficiently accomplishes the feature extraction task of the backbone network. The neck component processes and integrates feature maps from multiple levels of the backbone network to improve the model's capability in detecting objects of different sizes. Compared to YOLOv5, YOLOv8 replaces the C3 structures with the lighter C2f structures. The C2f module enhances gradient flow by utilizing various convolution modules and 'n' bottlenecks. In the head section, YOLOv8 adopts the idea from YOLACT for instance segmentation tasks by transitioning from a coupled head structure to a decoupled-head structure. Additionally, it utilizes the anchor-free method [37], thereby separating classification from detection. This reduces the model's computational load and enhances its generalization capability. Finally, the CIoU utilized in YOLOv8 is capable of effectively addressing segmentation tasks in intricate scenarios.

3. The Improved BHI-YOLO Instance Segmentation Model

Although YOLOv8-seg stands out as one of the more cutting-edge segmentation models at present, disease segmentation in complex environments tends to result in missed detections and false positives. The algorithm also requires numerous parameters in such environments, which leads to high computational costs. These costs impede its integration into resource-constrained mobile devices, thus limiting its practical applicability. To address the previously mentioned issues, this paper utilizes the general instance segmentation algorithm YOLOv8n-seg as the baseline model, improves and redesigns its network structure, and develops the new BHI-YOLO instance segmentation model. First, to tackle the problem of an excessive parameter count in YOLOv8, which complicates deployment on mobile devices, this study presents the C2f_UIB module as part of the backbone network. This reduces parameters and simplifies computational demands while also providing a more flexible balance between spatial and channel mixing, expanding the receptive field, and enriching semantic information. Second, to address the multi-scale issue of strawberry diseases in field greenhouses, the introduction of the High-level Screening-feature Fusion Pyramid Network (HSFPN) reduces the parameter volume while effectively filtering feature maps at various scales. It integrates disparate scale information synergistically, ultimately enhancing the model's detection capability. Finally, the iRMA structure, designed through the integration of iRMB and EMA, is fused with the C2f module of the neck network to establish the C2f_iRMA structure. This structure enhances the aggregation of global information, reduces interference, improves feature fusion, enhances model generalization capability, and significantly boosts segmentation performance. The BHI-YOLO structure is shown in Figure 3.

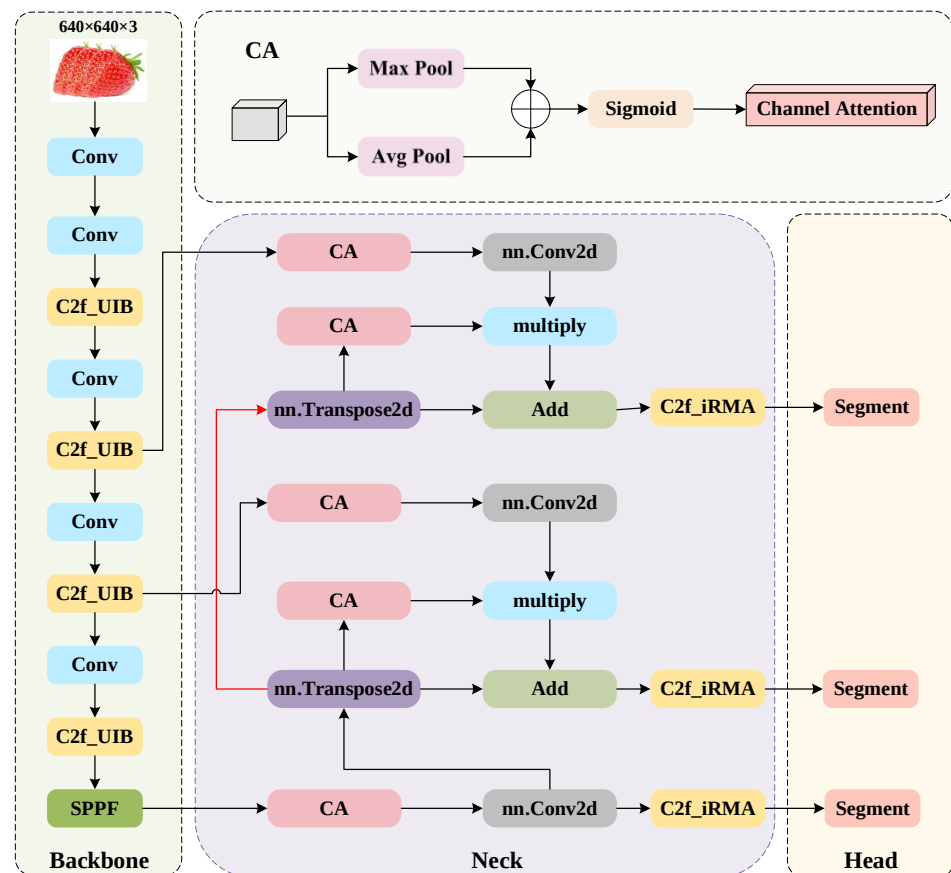


Figure 3. BHI-YOLO structure diagram.

3.1. Introduction of UIB

To make lightweight improvements to YOLOv8, this paper introduces the Universal Inverted Bottleneck (UIB) newly designed based on the Inverted Bottleneck [38] (IB) (as shown in Figure 4) module in the MobileNetV4 paper. This module (Figure 5) employs the ReLU activation function to prevent the vanishing gradient problem, thereby enabling more effective capture of complex features and enhancing the model's generalization capability. The ReLU formula is shown as follows.

$$\text{ReLU}(x_{in}) = \max(0, x_{in}) \quad (1)$$

where x_{in} is the input feature.

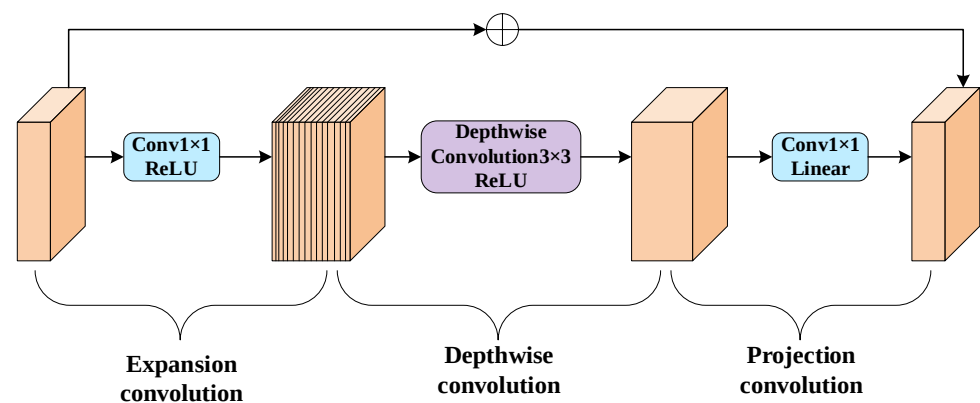


Figure 4. Inverted Bottleneck (IB).

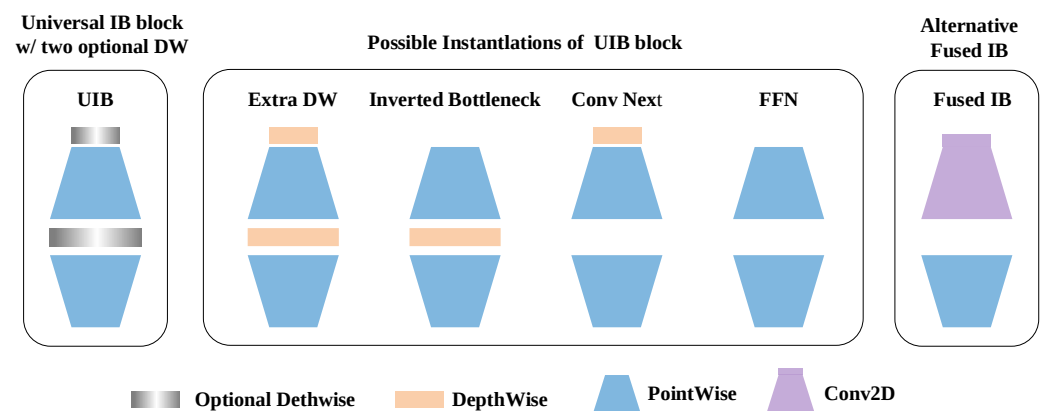


Figure 5. Universal Inverted Bottleneck (UIB) block.

According to Figure 4, it is evident that the IB module utilizes depthwise separable convolution, inverted residual, and linear bottleneck structures. The inverted residual structure employs residual connections and follows a pattern of first expanding the input, extracting features, and then reducing dimensions. This expansion introduces more channels to capture richer feature information. After expansion, depthwise separable convolution is employed to reduce parameter count and complexity. Additionally, linear activation is used instead of ReLU activation to minimize information loss. As shown in Figure 5, UIB introduces two optional depthwise convolutions (DW) within the Inverted Bottleneck block: one before the expansion layer and another between the expansion and projection layers. In addition, a new variant of ExtraDW is introduced, which effectively increases the network's depth as well as its receptive field. The UIB module is capable of trading off between spatial and channel mixing, expanding the receptive field, and maximizing computational utilization. Shi et al. [39] integrated the UIB module into YOLOv9; however, this paper differs from their work in the design of the UIB module. We use ReLU as the

activation function for this module to alleviate the vanishing gradient problem during the training process. Additionally, this paper employs depthwise separable convolutions to reduce the number of parameters and maximize the utilization of computational resources, which was not mentioned in the YOLOv9 publication. Furthermore, this paper focuses on how to modify the YOLOv8 backbone network using the UIB module to enhance its feature extraction capabilities. However, the YOLOv9 study did not provide a revised structure diagram for YOLOv9, which differs from our approach to improving the applicability and design goals of YOLOv8. These differences will lead to varying model performance. This paper constructs the C2f_UIB (Figure 6) module by replacing the bottleneck structure of the C2f module in the backbone network with the UIB module. The C2f_UIB module more effectively utilizes multi-channel information. This approach reduces module computational complexity and parameter count while obtaining more comprehensive strawberry disease feature information, thereby providing strong support for the improvement of lightweight models. Thus, the module is better suited for instance segmentation tasks overall. The UIB formula can be expressed as follows:

$$y_{out} = ReLU(W_d \times ReLU(W_e \times x_{in})) \quad (2)$$

where W_e is the weight of the expansion layer, W_d is the weight of the compression layer, and y_{out} is the output feature.

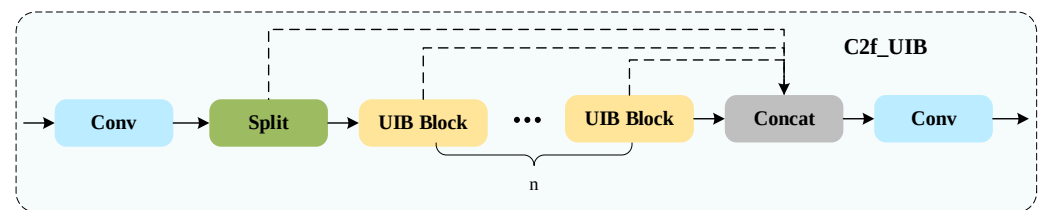


Figure 6. C2f_UIB structure diagram.

3.2. Introduction of HS-FPN

The High-level Screening-feature Fusion Pyramid Network (HS-FPN) first performs feature selection. As illustrated in Figure 7, the Channel Attention (CA) module processes the input feature maps, combining the feature maps generated through global average pooling and global max pooling. Afterward, the weight values for each channel are computed using the sigmoid activation function to obtain the final weights. The CA module uses this approach to accurately extract key information about strawberry diseases from each channel, while avoiding further loss of information. Subsequently, filtered feature maps are obtained by multiplying weight information with the corresponding proportion of feature maps. Before feature fusion, each proportional feature map's channel count is reduced to 256 using a 1×1 convolution, ensuring alignment across feature maps of different scales. CA is represented as follows:

$$f_{in} \in R^{C \times H \times W} \quad (3)$$

$$f_{CA} \in R^{C \times 1 \times 1} \quad (4)$$

where f_{in} symbolizes the input feature map; f_{CA} denotes the final weight for each channel; and C , H , and W refer to the number of channels and the height and width of the feature map, respectively.

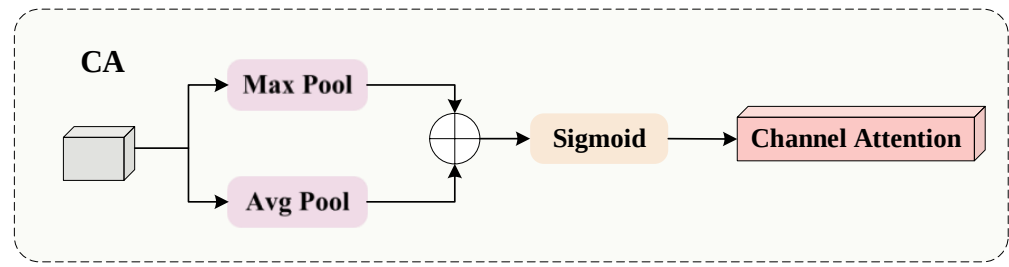


Figure 7. CA structure diagram.

After feature extraction from the feature maps, higher-level features are used as weights to eliminate basic semantic information contained in lower-level features, promoting targeted feature fusion. As shown in Figure 8, given high-level features f_{high} and low-level features f_{low} , features f_{high} are expanded using a transposed convolution with a stride of 2 and a 3×3 kernel, yielding $\widehat{f_{high}}$. To ensure dimensional alignment between high-level and low-level features, bilinear interpolation is used to upsample or downsample the high-level features, resulting in f_{att} . Finally, the low-level features filtered by the CA module are fused with the high-level features to obtain f_{out} . The formulaic process of the SSF module is as follows:

$$f_{low} \in \mathbb{R}^{C \times H_1 \times W_1} \quad (5)$$

$$f_{high} \in \mathbb{R}^{C \times H \times W} \quad (6)$$

$$\widehat{f_{high}} \in \mathbb{R}^{C \times 2H \times 2W} \quad (7)$$

$$f_{att} = BL(T - Conv(f_{high})) \quad (8)$$

$$f_{out} = f_{low} \times CA(f_{att}) + f_{att} \quad (9)$$

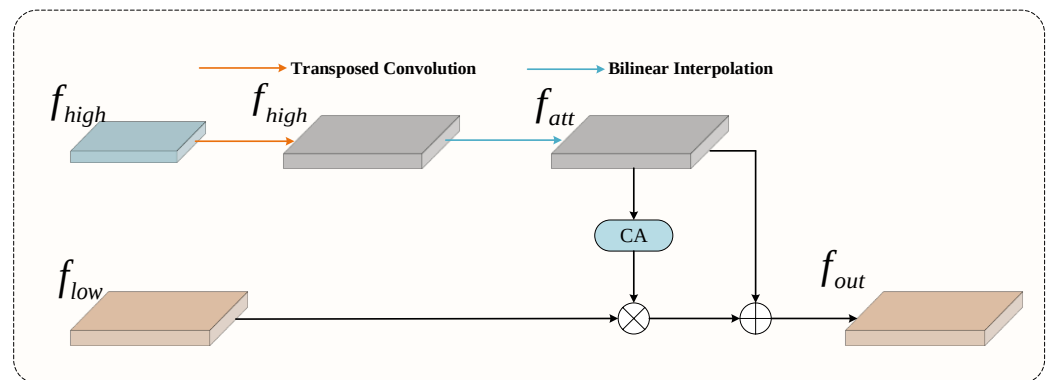


Figure 8. The framework of the SSF module.

In the strawberry disease dataset, the diverse types of strawberry diseases lead to a multi-scale issue, affecting the model's accuracy in disease identification. To address this, this experiment introduces HS-FPN (as depicted in Figure 3, neck structure) to achieve multi-scale feature fusion. The module utilizes CA and Dimension Matching (DM) mechanisms to screen feature maps of various diseases affecting strawberries. Through pooling operations and weight calculations, it effectively extracts important information from each channel. Subsequently, the Selective Feature Fusion (SSF) mechanism integrates the various strawberry disease features extracted earlier. The entire process not only preserves the information of small targets but also integrates other features of strawberries effectively, allowing the model to better adapt to complex strawberry disease issues and enhancing its segmentation capability. In addition, the simple structure of HSFPN significantly reduces computational redundancy, resulting in a smaller model size, which provides guidance for the lightweight improvements discussed in this paper.

3.3. Introduction of iRMA

The main idea of Inverted Residual Mobile Block (iRMB) is to enhance network performance on mobile devices by combining residual connections with mobile blocks. As presented in Figure 9, the iRMB module is made up of a depthwise separable convolution module and an interactive attention module. The first step involves feature expansion through 1×1 convolution to enhance the model's expressive capability. The second step utilizes Depthwise Separable Convolution (DW-Conv), applying 3×3 convolution kernels separately on each channel to obtain multiple feature maps. Subsequently, the feature maps derived from depthwise convolution are combined using 1×1 convolution kernels to produce the final feature map. This significantly reduces computation and the parameter count while maintaining model effectiveness. Residual connections allow input features to be directly added to output features, aiding the network in learning identity mappings or residual information more easily. This structure enhances the model's convergence speed and overall performance. Residual connections allow the network to more easily learn identity mappings or residual information by directly adding input features to output features. Ultimately, it achieved the goal of improving the model's convergence speed and performance.

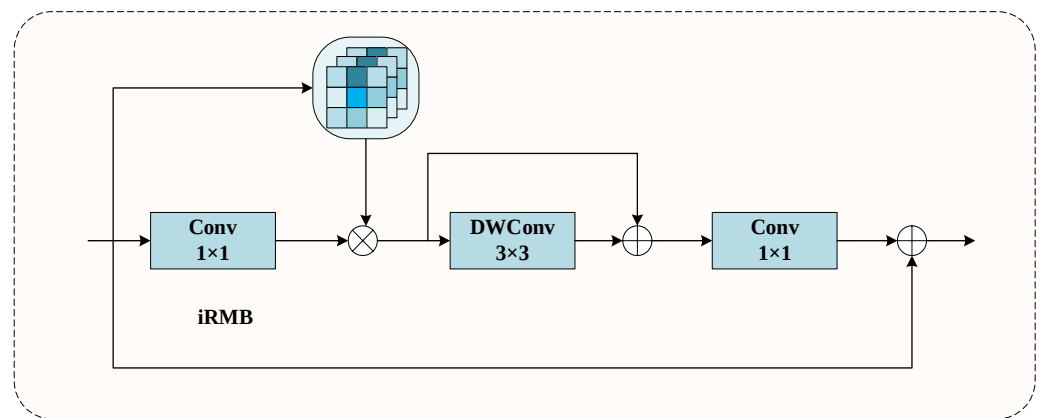


Figure 9. iRMB structure diagram.

Efficient Multi-Scale Attention (EMA) demonstrates excellent performance across various visual tasks and is particularly adept at handling complex tasks involving multi-scale information. It effectively enhances the model's perceptual skills across a range of scales while maintaining high efficiency. It avoids the impact of downscaling in the convolution process by transforming a segment of the channel dimension into the batch dimension and dividing the channel dimension into two types of sub-features: 1×1 and 3×3 . As shown in Figure 10, an EMA unit has input $X_{in} = [x_1, x_2, \dots, x_C] \in R^{C \times W \times H}$ and output $Y_{out} = [y_1, y_2, \dots, y_C] \in R^{C \times W \times H}$. First, separate X_{in} into G sub-features based on the channel dimension, i.e., $X_{in} = [X_0, X_1, \dots, X_{G-1}] \in R^{C/G \times W \times H}$, and then apply pooling operations with kernel sizes $(1, W)$ and $(H, 1)$ along the width and height dimensions, respectively. These operations generate two sets of one-dimensional features that encode vertical and horizontal contextual information. In the 3×3 branch, 3×3 convolutional kernels are used to capture multi-scale features. Concatenate two independent one-dimensional feature encodings and then decompose them into two vectors. Use the nonlinear sigmoid function to model the two-dimensional binary distribution of linear convolution. Then, implement different cross-spatial interaction features in the two branches. The outputs of the two branches are globally encoded using two-dimensional global average pooling operations to incorporate global information. The resulting outputs are then transformed into corresponding feature maps $R_1^{1 \times C/G} \times R_3^{C/G \times W \times H}$. In the end, each group's output feature map is calculated by combining two spatial attention weight values that have been produced. This aggregated result is applied to the input feature map X_{in} through

multiplication, yielding the output Y_{out} , where X_{in} and Y_{out} have the same dimensions. The two-dimensional global pooling operation is expressed as follows:

$$z_C = \frac{1}{W \times H} \sum_m^W \sum_n^H x_C(m, n) \quad (10)$$

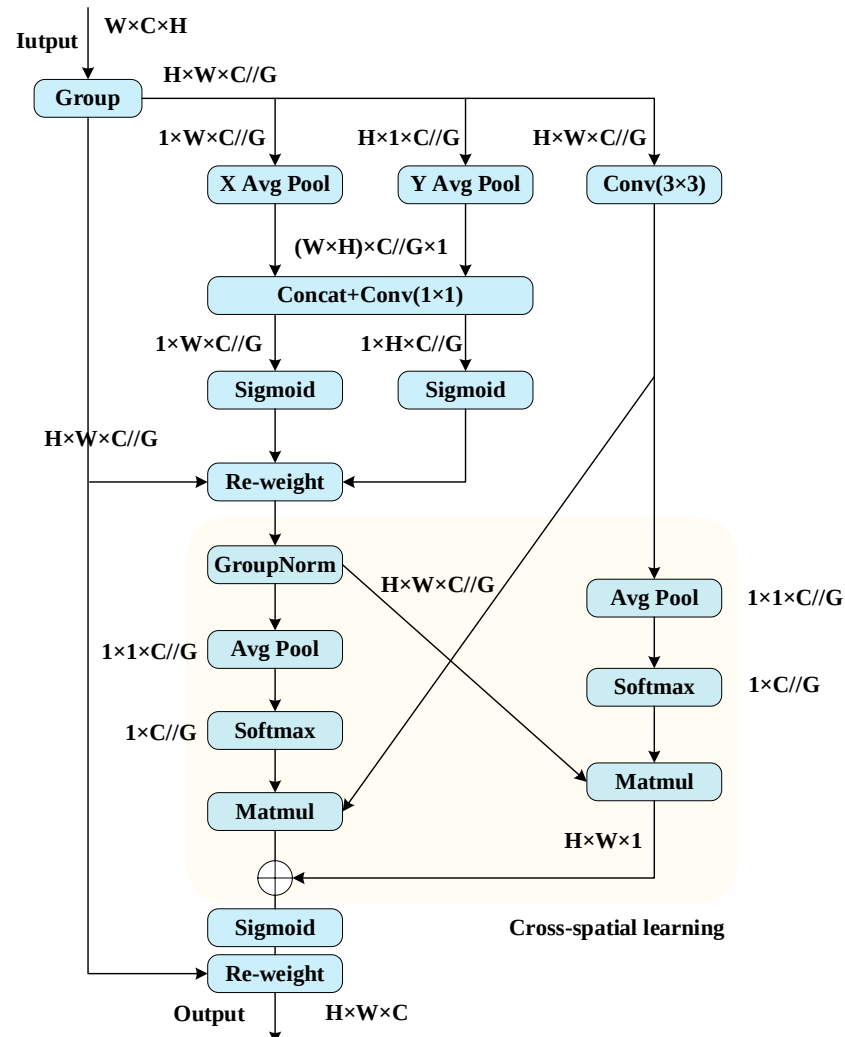


Figure 10. Structure diagram for EMA.

Here, x_C represents the input features at the C -th channel.

Attention mechanisms can flexibly capture various types of information from feature maps, inspired by the way human vision processes visual information. Various categories of strawberry disease in images exhibit distinct disease characteristics and diverse shapes. Therefore, introducing the attention mechanism helps to effectively integrate features across different scales. This paper designs the iRMA (Figure 11) mechanism by borrowing the inverted residual structure of iRMB and combining it with a plug-and-play EMA attention mechanism. The iRMA mechanism is incorporated into the C2f module, resulting in the C2f_iRMA structure (Figure 12). As shown in Figure 11, this module utilizes EMA for cross-space learning, aggregating information across different spatial dimensions. It captures spatial features through depthwise separable convolutions. Simultaneously, this component is connected by a residual edge to prevent gradient vanishing, thereby enhancing the stability and generalization effectiveness of model training. Finally, 1×1 convolutions are used to restore the channel number. Zhang et al. [40] improved YOLOv8's performance by designing the iRMA module. However, their paper lacks details on the specific structures

of iRMA and C2f_iRMA, resulting in design differences compared to this paper. While both studies employ C2f_iRMA for YOLOv8 enhancements, they differ in YOLOv8 locations. This paper's iRMA structure integrates all C2f modules in the neck network, whereas Zhang et al. utilized two C2f_iRMA modules. These design variations and distinct application scenarios lead to differences in model performance and applicability. Lastly, Zhang et al. did not provide experimental data to validate the performance of their iRMA module against EMA and iRMB, whereas this paper's experiments demonstrate that the iRMA module outperforms EMA and iRMB in strawberry segmentation tasks. The design of iRMA enhances segmentation performance while effectively controlling the increase in parameters, addressing the issues of the elevation of iRMB parameters and the limited improvement in EMA performance, and tackling the multi-scale challenges in strawberry disease recognition. This allows the research presented in this paper to achieve higher precision more easily during the development of lightweight models.

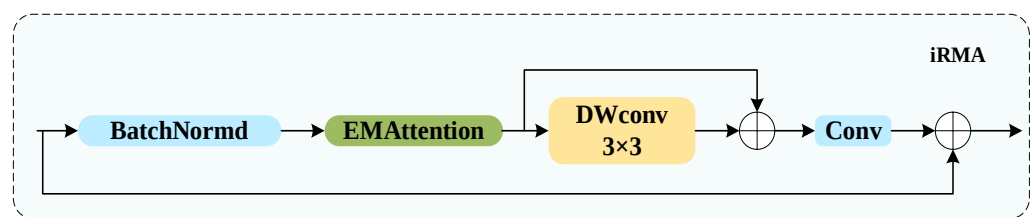


Figure 11. iRMA structure diagram.

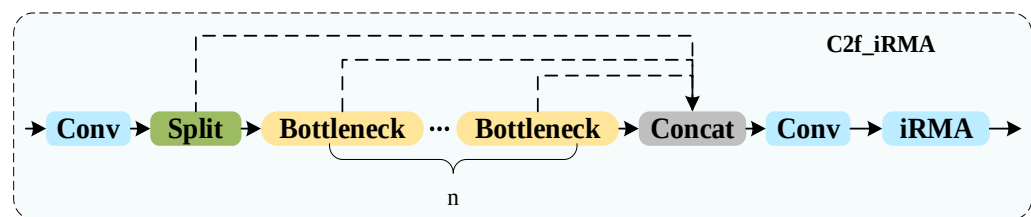


Figure 12. C2f_iRMA structure diagram.

3.4. Evaluation Criteria

This experiment uses Intersection over Union (IoU) and mean average precision with masks (Mask mAP50) to measure the performance of the model. Metrics including GFLOPs, parameters (M), and model size (Mb) are employed to evaluate the scale of the model. The formulas are provided as follows:

$$Precision = \frac{TP}{TP + FP} \quad (11)$$

$$Recall = \frac{TP}{TP + FN} \quad (12)$$

$$AP = \int_0^1 P(R) dR \quad (13)$$

$$mAP = \frac{1}{n} \sum_{i=1}^n AP_i \quad (14)$$

$$IoU = \frac{Intersection\ Area}{Union\ Area} \quad (15)$$

$$mIoU = \frac{1}{n} \sum_{i=1}^n IoU_i \quad (16)$$

Precision assesses the correctness of positive predictions, whereas recall evaluates the capability to identify true positive cases. “*n*” denotes the total number of categories. IoU

measures the overlap between the predicted area and the ground truth area, while mIoU is the average of IoUs across all categories. AP represents the accuracy of a strawberry disease, while mAP represents the average accuracy of nine different strawberry diseases.

4. Experiments and Analysis

4.1. Experimental Environment

The experimental platform parameters described in this paper are detailed in Table 2. Additionally, the image dimensions have been resized to (640, 640), and SGD [41] has been employed as the optimizer.

Table 2. Experimental configuration parameters.

Items	Types
Operating system	Windows 11
CPU	Intel Core i7-12650H
GPU	RTX 4060(8G)
Language	Python 3.8
Accelerating environment	CUDA11.8
framework	torch2.0
Batch Size	16
Learning Rate	0.01
Epochs	300

4.2. Segmentation Performance Analysis of Various Modules

To verify the performance of the improved C2f_UIB module, this paper simultaneously introduces modifications to the backbone network using other modules as control groups. To ensure the experimental rigor, modifications to the backbone network are consistent with the UIB. The results are shown in Table 3.

Table 3. A comparison of various module segmentation results.

Model	Parameters (M)	GFLOPs	Mask mAP50	Model Size (Mb)
YOLOv8n-seg (baseline)	3.26	12.0	90.7	6.8
UIB	2.83	10.8	91.7	6.0
Fast block	2.55	10.3	90.9	5.4
Context guided	2.80	10.6	91.0	5.9

According to the data in Table 3, the backbone models improved using UIB, fast block [42], and context-guided models [43], compared to the baseline model, exhibit increased mAP by 1.0%, 0.2%, and 0.3%, respectively. While the fast block module has relatively fewer parameters, GFLOPs, and model size compared to the UIB module, it only achieved a modest mAP improvement of 0.2%. The context-guided module has parameters, GFLOPs, and model size comparable to UIB, but it only achieved a modest 0.3% increase in mAP. Through comparison, the UIB module balances the lightweight nature of the model and accuracy, demonstrating higher effectiveness and superiority overall.

In addition to comparisons with the baseline model, this paper also selects current mainstream feature pyramid models, as shown in Table 4. Compared to YOLOv8n-seg, AFPN [44], GFPN [45], Bi-FPN [46], and HS-FPN achieved improvements in mAP of 0.7%, 1.0%, 0.9%, and 1.0%, respectively, showing similar enhancement effects. Meanwhile, in this paper, the HSFPN features fewer parameters, lower GFLOPs, and a smaller model size than the other models. The experimental results indicate that the improvement method used in this paper is more suitable for enhancing lightweight models.

Table 4. Comparison of results from different Feature Pyramid Networks.

Model	Parameters (M)	GFLOPs	Mask mAP50	Model Size (Mb)
YOLOv8n-seg (baseline)	3.26	12.0	90.7	6.8
AFPN	3.39	13.2	91.4	7.1
GFPN	3.54	12.3	91.7	7.4
Bi-FPN	3.03	12.0	91.6	6.4
HS-FPN	2.14	10.7	91.7	4.6

The iRMA module is a novel design in this paper, combining the inverted residual block concept from iRMB with the EMA. As shown in Table 5, iRMB, EMA, and iRMA improve mAP by 1.6%, 0.9%, and 1.7%, respectively. The improved model using the iRMB module shows varying degrees of growth in parameters, GFLOPs, and model size, with GFLOPs notably rising from 12.0 to 15.1, a significant increase. The model improved with EMA shows no differences in parameters, GFLOPs, and model size compared to YOLOv8n-seg, but the increase in mAP is relatively small. The designed iRMA mechanism in this study combines the advantages of two modules, achieving a higher increase in mAP while keeping the model more lightweight. Through comprehensive comparison, the iRMA module demonstrates superior applicability and effectiveness.

Table 5. Results of comparisons between various attention mechanisms.

Model	Parameters (M)	GFLOPs	Mask mAP50	Model Size (Mb)
YOLOv8n-seg (baseline)	3.26	12.0	90.7	6.8
iRMB	3.37	15.1	92.3	7.1
EMA	3.26	12.0	91.6	6.8
iRMA	3.29	12.1	92.4	6.9

4.3. Ablation Experiment

To ascertain whether integrating various modules results in a model with enhanced segmentation effectiveness, we conducted ablation experiments. Using YOLOv8n-seg as the baseline model, we sequentially incorporated improvements into the model training and validation process. The results from the validation ablation experiments are provided in Table 6. The experiment improves upon YOLOv8n-seg to enhance model accuracy while achieving a lightweight model. Afterwards, the C2f_UIB module was designed, which expanded the model's receptive field and helped the model obtain more comprehensive information about strawberry disease features. Compared to YOLOv8n-seg, the mAP increased by 1%, the mIoU increased by 1.2%, the model's parameter count was reduced by 13%, GFLOPs reduced by 10%, and the model size decreased by 12%. After integrating the High-level Selective Feature Pyramid Network (HSFPN), the model's capability to express strawberry disease features was enhanced. At this stage, the mAP increased by 1.7%, the mIoU increased by 2.2%, the model's parameter count was reduced by 48.6%, GFLOPs decreased by 22.5%, and the model size decreased by 45.5%. After finally introducing the iRMA embedding neck network into C2f, the effective aggregation of strawberry disease features was achieved. With minimal sacrifice in parameter count, the model's final mAP increased by 2.3%, the mIoU increased by 2.6%, the parameters were reduced by 47%, the GFLOPs decreased by 20%, and the model size decreased by 44.1%. Compared to YOLOv8n-seg, the enhanced model BHI-YOLO significantly improves performance while drastically reducing model size. This effectively addresses the issue of high computational costs.

Table 6. Ablation experiment results.

Model	Parameters (M)	GFLOPs	Mask mAP50	mIoU (%)	Model Size (Mb)
YOLOv8n-seg (baseline)	3.26	12.0	90.7	74.9	6.8
+C2f_UIB	2.83	10.8	91.7	76.1	6.0
+HSFPN	2.14	10.7	91.7	76.3	4.6
+C2f_iRMA	3.29	12.1	92.4	76.8	6.9
+C2f_UIB + HSFPN	1.68	9.3	92.4	77.1	3.7
+C2f_UIB + HSFPN + C2f_iRMA	1.72	9.6	93.0	77.5	3.8

4.4. Comparative Analysis of Algorithms

To better demonstrate the improvement in instance segmentation in this paper, in addition to the YOLOv8 baseline model, we selected current mainstream instance segmentation algorithm models for comparison, including YOLACT and the YOLO series, together with Mask R-CNN and Cascade-Mask R-CNN [47]. According to Table 7, the BHI-YOLO model from this experiment demonstrates an increase in mAP compared to the other seven instance segmentation models by 5%, 1.7%, 6.2%, 0.1%, 6.8%, 8.5%, and 10.4%, separately. The mAP achieves optimal results, with only YOLOv9 [48] demonstrating similar segmentation performance to the model improved in this paper. However, the parameters, GFLOPs, and model size of YOLOv9 are 4.23 times, 4.64 times, and 4.20 times those of YOLOv8, respectively, making it a very large model that is difficult to improve in terms of lightweight design. In contrast, YOLOv8 is inherently smaller and has become even more lightweight after optimization, making it extremely suitable for the research direction of this paper. Compared to other algorithms, BHI-YOLO in this paper exhibits outstanding performance in strawberry disease instance segmentation tasks, while minimizing parameter counts and computational overhead, making it easier to deploy on mobile devices.

Table 7. Comparison results of different algorithms.

Method	Parameters (M)	GFLOPs	Mask mAP50	Model Size (Mb)
YOLOv3tiny-seg	14.10	32.7	88.0	28.4
YOLOv5s-seg	7.42	25.8	91.3	15.2
YOLOv6-seg	4.40	15.2	86.8	9.1
YOLOv9c-seg	13.8	55.7	92.9	28.6
Mask R-CNN	44.0	190	86.2	370.2
Cascade-Mask R-CNN	77.1	1713	84.5	686.0
YOLACT	35.3	68.4	82.6	278.8
YOLOv8n-seg	3.26	12.0	90.7	6.8
BHI-YOLO	1.72	9.6	93.0	3.8

4.5. Comparison of Results

To validate the performance differences between BHI-YOLO and the baseline model across nine categories of strawberries, we used precision–recall (P–R) curves to compare the segmentation results of the two algorithms. The P–R curve for strawberry disease segmentation is presented in Figure 13. It can be noted that the AP for anthracnose fruit rot shows the most significant improvement, increasing by 9.4%. The significant improvement is mainly due to the relatively large-scale variation in anthracnose fruit rot, where the disease can affect a small part or even the entire fruit. Therefore, compared to other categories, segmenting it is more challenging. With the optimized model, it can better capture global contextual information by adjusting feature weights to effectively adapt to targets of different scales and positions, thus enhancing segmentation performance. In other categories, the improved model shows enhanced segmentation performance across the board. Overall, the improved model has shown an overall 2.3% increase in mAP, outperforming the baseline model.

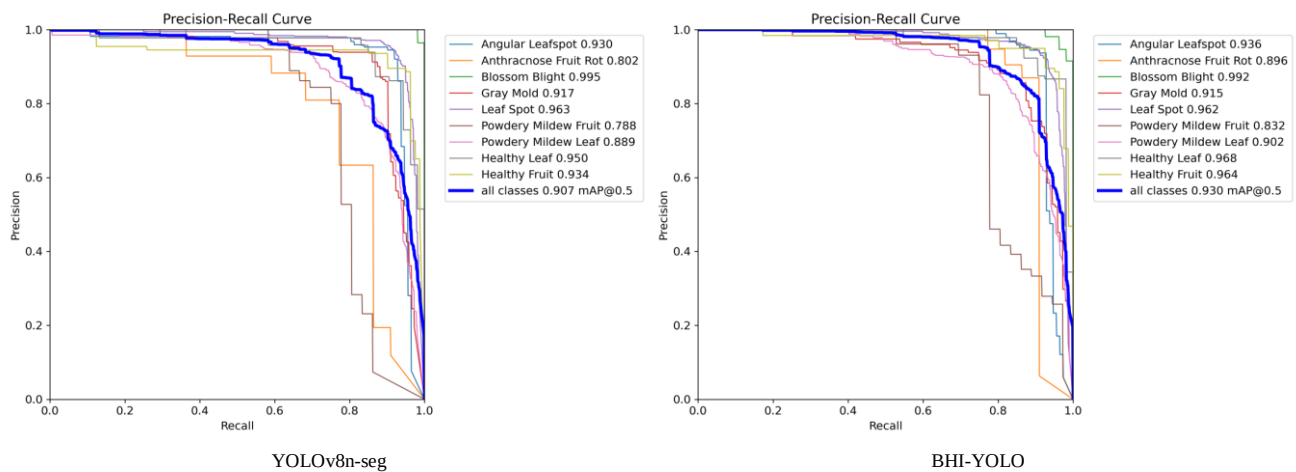


Figure 13. P–R curve comparison chart.

To visually compare the segmentation performance before and after model improvements, predictions were made on the test set employing both the baseline model and BHI-YOLO. Figure 14 depicts the instance segmentation results of the YOLOv8n-seg and BHI-YOLO models. The first row in the figure shows manually created truth labels. The instance segmentation outputs of YOLOv8 are presented in the second row, and the results of BHI-YOLO are displayed in the third row. Each column represents different disease characteristics of strawberries.

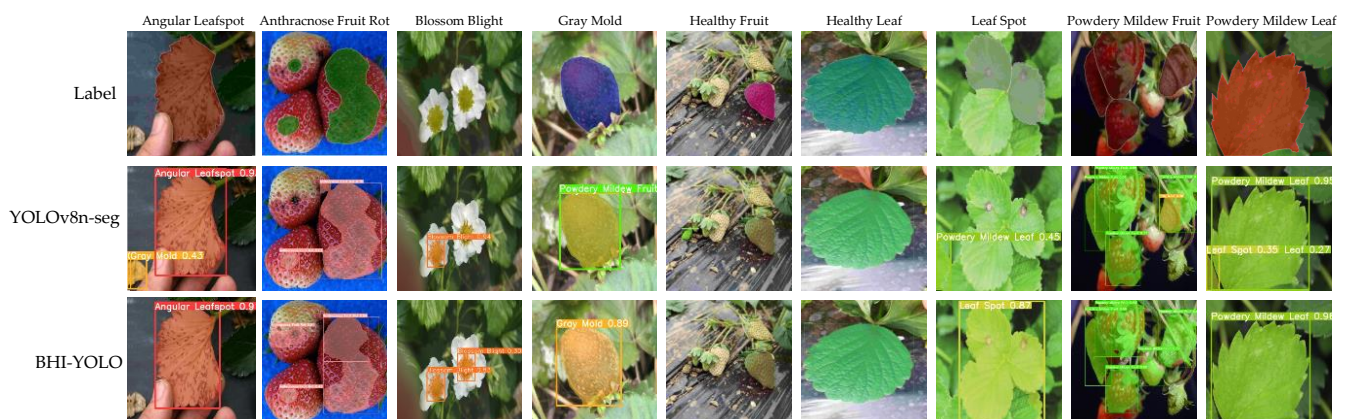


Figure 14. Comparison of segmentation effects between the YOLOv8n-seg and the BHI-YOLO.

Figure 14 shows that in the segmentation results of YOLOv8, columns 1, 5, 6, and 9 mistakenly detect the background of the images as strawberry diseases. In columns 4 and 8, strawberry diseases are incorrectly identified as another disease with similar features. Columns 2, 3, and 7 exhibit cases of missed detection. After improvement, the model can accurately identify various diseases in strawberries. This is because in complex backgrounds of strawberry disease images, the improved model has enhanced feature extraction capabilities and the ability to express features at different scales. As a result, the improved model exhibits superior adaptability to complex scenarios and performs better in multi-scale instance segmentation. The experimental findings indicate that the BHI-YOLO achieves ultimate lightweighting while possessing superior instance segmentation performance. This fully proves that the improved model not only exhibits stronger robustness but also effectively addresses the challenges of lightweight deployment on mobile terminals, thereby reducing maintenance costs. However, adverse weather conditions like rain and snow may lead to poor lighting, in which case YOLOv8 might perform better. This aspect requires further validation and resolution in future research.

5. Conclusions

This paper presents BHI-YOLO, a lightweight algorithm for strawberry disease segmentation that utilizes the YOLOv8 architecture, effectively addressing challenges such as the difficulty of detecting strawberry diseases in complex environments, high computational complexity of models, and large algorithm parameter counts. First, by embedding the UIB module within the C2f of the backbone network, the C2f_UIB module achieves a reduction in computational burden while maintaining excellent feature extraction capabilities. Additionally, the introduction of the HS-FPN (High-level Screening-feature Fusion Pyramid Network) further reduces parameter count while expanding the model's ability to perceive multi-scale targets. Finally, the iRMA module is designed to capture spatial features, aggregate global information, and enhance the segmentation accuracy of various diseases in strawberries. The experimental findings indicate that the enhanced model lowered the parameter count by 47%, decreased the computational workload by 20%, and reduced the model size by 44.1%. Moreover, it achieved a 2.3% increase over YOLOv8n-seg in mAP. This indicates that the improved model architecture achieves high performance while effectively reducing the algorithm's dependence on platform hardware, aiding the deployment on mobile devices that have restricted resources.

In the next phase of our research, we intend to apply model quantization and knowledge distillation in order to further lower the model's computational overhead and enhance its performance. Additionally, we aim to deploy and apply the model on mobile devices to expand its practical application and versatility, meeting users' demands for portability.

Author Contributions: Conceptualization, H.H. and M.C.; methodology, H.H.; software, H.H.; validation, H.H.; formal analysis, L.H. and M.C.; investigation, C.G.; resources, M.C.; data curation, H.H.; writing—original draft preparation, H.H.; writing—review and editing, H.H. and M.C.; visualization, H.H.; supervision, M.C.; project administration, M.C.; funding acquisition, M.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (Grant No. 61863009), the Guangxi Key R&D Program (Guike-AB22080093), and the Wuzhou Central Leading Local Science and Technology Development Fund Project (Grant No. 202201001).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

- Food and Agriculture Organization of the United Nations. Available online: <https://www.fao.org/faostat/zh/#data/QCL> (accessed on 26 May 2024).
- Gu, J.; Wang, Z.; Kuen, J.; Ma, L.; Shahroudy, A.; Shuai, B.; Liu, T.; Wang, X.; Wang, G.; Cai, J. Recent advances in convolutional neural networks. *Pattern Recogn.* **2018**, *77*, 354–377. [\[CrossRef\]](#)
- Park, J.; Shin, J.; Kim, R.; An, S.; Lee, S.; Kim, J.; Oh, J.; Jeong, Y.; Kim, S.; Jeong, Y.R. Accelerating Strawberry Ripeness Classification Using a Convolution-Based Feature Extractor along with an Edge AI Processor. *Electronics* **2024**, *13*, 344. [\[CrossRef\]](#)
- Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
- Raoofi, H.; Motamedi, A. Mask R-CNN deep learning-based approach to detect construction machinery on jobsites. In Proceedings of the 37th International Symposium on Automation and Robotics in Construction (ISARC 2020), Kitakyushu, Japan, 27–28 October 2020; pp. 1122–1127.
- Abu Alhaija, H.; Mustikovela, S.K.; Mescheder, L.; Geiger, A.; Rother, C. Augmented reality meets computer vision: Efficient data generation for urban driving scenes. *Int. J. Comput. Vis.* **2018**, *126*, 961–972. [\[CrossRef\]](#)
- Tassis, L.M.; de Souza, J.E.T.; Krohling, R.A. A deep learning approach combining instance and semantic segmentation to identify diseases and pests of coffee leaves from in-field images. *Comput. Electron. Agric.* **2021**, *186*, 106191. [\[CrossRef\]](#)
- Guo, Z.; Hu, X.; Zhao, B.; Wang, H.; Ma, X. StrawSnake: A Real-time strawberry instance segmentation network based on the contour learning approach. *Electronics* **2024**, *13*, 3103. [\[CrossRef\]](#)

9. Charisis, C.; Argyropoulos, D. Deep learning-based instance segmentation architectures in agriculture: A review of the scopes and challenges. *Smart Agr. Technol.* **2024**, *8*, 100448. [\[CrossRef\]](#)
10. Opasatian, I.; Ahamed, T. Driveway Detection for Weed Management in Cassava Plantation Fields in Thailand Using Ground Imagery Datasets and Deep Learning Models. *AgriEngineering* **2024**, *6*, 3408–3426. [\[CrossRef\]](#)
11. Sarkar, N.R. *Quality Sorting of Tomatoes Using Computer Vision*; Rutgers the State University of New Jersey, School of Graduate Studies: New Brunswick, NJ, USA, 1985.
12. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
13. Farhadi, A.; Redmon, J. YOLOv3: An incremental improvement. In *Computer Vision and Pattern Recognition*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 1–6.
14. Jocher, G.N.K.; Minerva, T.; Vilariño, R.J.A.M. YOLOv5. Available online: <https://github.com/ultralytics/yolov5> (accessed on 3 May 2024).
15. Gupta, C.; Gill, N.S.; Gulia, P.; Chatterjee, J.M. A novel finetuned YOLOv6 transfer learning model for real-time object detection. *J. Real-Time Image Process.* **2023**, *20*, 42. [\[CrossRef\]](#)
16. Bolya, D.; Zhou, C.; Xiao, F.; Lee, Y.J. Yolact: Real-time instance segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 9157–9166.
17. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In *Computer Vision—ECCV 2016, Proceedings of the 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016*; Proceedings, Part I 14; Springer: Berlin/Heidelberg, Germany, 2016; pp. 21–37.
18. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
19. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 1137–1149. [\[CrossRef\]](#)
20. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
21. Jiang, P.; Chen, Y.; Liu, B.; He, D.; Liang, C. Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks. *IEEE Access* **2019**, *7*, 59069–59080. [\[CrossRef\]](#)
22. Sapkota, R.; Ahmed, D.; Churuvija, M.; Karkee, M. Immature green apple detection and sizing in commercial orchards using YOLOv8 and shape fitting techniques. *IEEE Access* **2024**, *12*, 43436–43452. [\[CrossRef\]](#)
23. Fuentes, A.; Yoon, S.; Kim, S.C.; Park, D.S. A Robust Deep-Learning-Based Detector for Real-Time Tomato Plant Diseases and Pests Recognition. *Sensors* **2017**, *17*, 2022. [\[CrossRef\]](#) [\[PubMed\]](#)
24. Yao, N.; Ni, F.; Wu, M.; Wang, H.; Li, G.; Sung, W.K. Deep Learning-Based Segmentation of Peach Diseases Using Convolutional Neural Network. *Front. Plant Sci.* **2022**, *13*, 876357. [\[CrossRef\]](#) [\[PubMed\]](#)
25. Afzaal, U.; Bhattarai, B.; Pandeya, Y.R.; Lee, J. An Instance Segmentation Model for Strawberry Diseases Based on Mask R-CNN. *Sensors* **2021**, *21*, 6565. [\[CrossRef\]](#) [\[PubMed\]](#)
26. Huang, M.; Xu, G.; Li, J.; Huang, J. A method for segmenting disease lesions of maize leaves in real time using attention YOLACT++. *Agriculture* **2021**, *11*, 1216. [\[CrossRef\]](#)
27. Kumar, V.S.; Jaganathan, M.; Viswanathan, A.; Umamaheswari, M.; Vignesh, J. Rice leaf disease detection based on bidirectional feature attention pyramid network with YOLO v5 model. *Environ. Res. Commun.* **2023**, *5*, 065014. [\[CrossRef\]](#)
28. Lu, Y.; Gong, M.; Li, J.; Ma, J. Strawberry Defect Identification Using Deep Learning Infrared–Visible Image Fusion. *Agronomy* **2023**, *13*, 2217. [\[CrossRef\]](#)
29. Tamrakar, N.; Karki, S.; Kang, M.Y.; Deb, N.C.; Arulmozhi, E.; Kang, D.Y.; Kook, J.; Kim, H.T. Lightweight Improved YOLOv5s-CGhostnet for Detection of Strawberry Maturity Levels and Counting. *AgriEngineering* **2024**, *6*, 962–978. [\[CrossRef\]](#)
30. Mi, Z.; Yan, W.Q. Strawberry Ripeness Detection Using Deep Learning Models. *Big Data Cogn. Comput.* **2024**, *8*, 92. [\[CrossRef\]](#)
31. 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. [\[CrossRef\]](#)
32. Xie, D.; Yao, W.; Sun, W.; Song, Z. Real-Time Identification of Strawberry Pests and Diseases Using an Improved YOLOv8 Algorithm. *Symmetry* **2024**, *16*, 1280. [\[CrossRef\]](#)
33. 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.
34. Chen, Y.; Zhang, C.; Chen, B.; Huang, Y.; Sun, Y.; Wang, C.; Fu, X.; Dai, Y.; Qin, F.; Peng, Y.; et al. Accurate leukocyte detection based on deformable-DETR and multi-level feature fusion for aiding diagnosis of blood diseases. *Comput. Biol. Med.* **2024**, *170*, 107917. [\[CrossRef\]](#) [\[PubMed\]](#)
35. Zhang, J.; Li, X.; Li, J.; Liu, L.; Xue, Z.; Zhang, B.; Jiang, Z.; Huang, T.; Wang, Y.; Wang, C. Rethinking mobile block for efficient attention-based models. In Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 2–6 October 2023; pp. 1389–1400.

36. Ouyang, D.; He, S.; Zhang, G.; Luo, M.; Guo, H.; Zhan, J.; Huang, Z. Efficient multi-scale attention module with cross-spatial learning. In Proceedings of the ICASSP 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Rhodes Island, Greece, 4–10 June 2023; pp. 1–5.
37. Tian, Z.; Shen, C.; Chen, H.; He, T. Fcos: A simple and strong anchor-free object detector. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 1922–1933. [[CrossRef](#)] [[PubMed](#)]
38. Howard, A.; Sandler, M.; Chu, G.; Chen, L.-C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V. Searching for mobilenetv3. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1314–1324.
39. 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. [[CrossRef](#)]
40. Zhang, C.; Zhao, X.; Sun, D.; Wang, X.; Xu, G.; Zhao, R.; Gao, M.; Ma, H. Improved Real-Time Monitoring Lightweight Model for UAVs Based on YOLOv8. In Proceedings of the International Conference on Intelligent Computing, Singapore, 28–30 September 2024; pp. 278–288.
41. Bottou, L. Large-scale machine learning with stochastic gradient descent. In Proceedings of the COMPSTAT'2010: 19th International Conference on Computational Statistics, Paris, France, 22–27 August 2010; pp. 177–186.
42. Chen, J.; Kao, S.-H.; He, H.; Zhuo, W.; Wen, S.; Lee, C.-H.; Chan, S.-H.G. Run, don't walk: Chasing higher FLOPS for faster neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 12021–12031.
43. Wu, T.; Tang, S.; Zhang, R.; Cao, J.; Zhang, Y. CGNet: A Light-Weight Context Guided Network for Semantic Segmentation. *IEEE Trans. Image Process.* **2021**, *30*, 1169–1179. [[CrossRef](#)]
44. Yang, G.; Lei, J.; Zhu, Z.; Cheng, S.; Feng, Z.; Liang, R. AFPN: Asymptotic feature pyramid network for object detection. In Proceedings of the 2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Honolulu, Oahu, HI, USA, 1–4 October 2023; pp. 2184–2189.
45. Xie, X.; Liao, Q.; Ma, L.; Jin, X. Gated feature pyramid network for object detection. In Proceedings of the Pattern Recognition and Computer Vision: First Chinese Conference, Guangzhou, China, 23–26 November 2018; pp. 199–208.
46. Tan, M.; Pang, R.; Le, Q.V. Efficientdet: Scalable and efficient object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 10781–10790.
47. Cai, Z.; Vasconcelos, N. Cascade R-CNN: High quality object detection and instance segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *43*, 1483–1498. [[CrossRef](#)]
48. 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.

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.