YOLOv8n-ESG: A Framework for Enhanced Strawberry Growth Stage Recognition

Yunchang Zheng
Hebei University of Architecture
Zhenpeng Zhang
Hebei University of Architecture
Xiangrui Wang
Hebei University of Architecture
He Cheng
Hebei University of Architecture
Yunlong Ye
189313123456@189.com

Hebei North University

Article

Keywords: Efficient agriculture, YOLOv8, Strawberry, Growth stage

Posted Date: July 10th, 2025

DOI: https://doi.org/10.21203/rs.3.rs-6616380/v1

Additional Declarations: No competing interests reported.

YOLOv8n-ESG: A Framework for Enhanced Strawberry Growth Stage Recognition

Yunchang Zheng[1], Zhenpeng Zhang[1], Xiangrui Wang[1], He Cheng[1], Yunlong Ye[2,*]

[1] College of Electrical Engineering, Hebei University of Architecture, Zhangjiakou, 075000, China
[2] Hebei North University, Zhangjiakou, 075000, China
[*](Corresponding author) Yunlong Ye, E-mail: 18931312345@189.com

Abstract

Accurate strawberry ripeness detection plays a vital role in quality assurance and market competitiveness enhancement within agricultural production. This study proposes an enhanced YOLOv8n-ESG model for efficient in-field strawberry maturity classification. The methodology categorizes strawberry growth stages into two phases (unripe vs. ripe), addressing quality deterioration risks from improper harvesting timing. Our technical improvements to the baseline YOLOv8n architecture include: 1) backbone network convolution layer optimization, 2) C2f module refinement, 3) attention mechanism integration, 4) loss function modification, and 5) data augmentation implementation. Experimental results demonstrate the model achieves 89.8% precision, 90.5% recall, and 94.8% mAP50 in complex scenarios, effectively mitigating misdiagnosis and missed detection issues. The proposed system enables growers to optimize harvesting schedules through precise ripeness identification, contributing to intelligent agricultural technology development. These advancements in visual recognition systems offer practical solutions for improving postharvest quality control and strengthening market position in perishable fruit supply chains.

Keywords: Efficient agriculture, YOLOv8, Strawberry, Growth stage

Introduction

The strawberry is a general term for plants in the genus Fragaria of the Rosaceae family, characterized by heart-shaped appearance and native to South America. It was introduced to China from Europe in the 20th century. Rich in vitamin C, strawberries possess high nutritional value and aid digestion due to their abundant pectin and cellulose. However, challenges emerge in automated monitoring and harvesting due to their biological traits: low-growing plants with dense fruit clusters, uneven berry sizes, asynchronous ripening, short shelf life, and fragile epidermis. Fruit occlusion by foliage further complicates automated detection, making manual harvesting labor-intensive. Developing a rapid, precise, and efficient strawberry maturity detection system holds significant value for automated growth monitoring and agricultural robotics, thereby advancing smart agriculture.

Deep learning applications in agriculture have expanded notably. As core technologies, object detection algorithms have achieved remarkable progress in industrial and agricultural contexts. The YOLO series exemplifies this advancement: YOLOv6[1] enhances detection efficiency in complex scenarios through re-parameterization and industrial optimization; YOLOX[2] innovates with anchor-free design to improve multi-scale adaptability; cutting-edge YOLOv10[3] pushes real-time detection accuracy boundaries via

enhanced feature pyramids and dynamic label assignment. Nevertheless, agricultural scenarios demand higher model lightweighting and computational efficiency. Existing lightweight networks like MobileNetV3[4] and ShuffleNetV2[4] balance speed and accuracy on mobile devices through neural architecture search and channel shuffling, yet their feature extraction capabilities remain inadequate for dense small-target detection.

In practical applications, Howard et al.[6] proposed an RGB-based strawberry maturity grading method using MobileNet, achieving 86.2% accuracy under uniform lighting through color/texture features, yet struggling with shadowed/uneven lighting and shape variations. Woo et al.[7] developed a multispectral-CBAM fusion model enhancing occluded fruit recognition through spectral feature enhancement, but incurred 35% inference latency increase and high equipment costs. Bochkovskiy et al.[7] improved YOLOv4 for natural light maturity detection (89.5% mAP), yet suffered 12.3% missed detection in dense overlaps. Wang et al.[8] boosted near-shot blemish recognition to 91.8% using CARAFE upsampling, at the cost of 2.1× training time increase. Law et al.[9] proposed keypoint-based CornerNet for occlusion handling, requiring over 100,000 manually annotated keypoint images while showing limited generalization to clustered fruits. Although modern deep learning surpasses traditional machine learning in accuracy, robustness, and generalization, as illustrated in Figure 1., challenges persist in variable lighting, complex backgrounds, and occlusion scenarios-manifesting as detection omissions and insufficient precision.

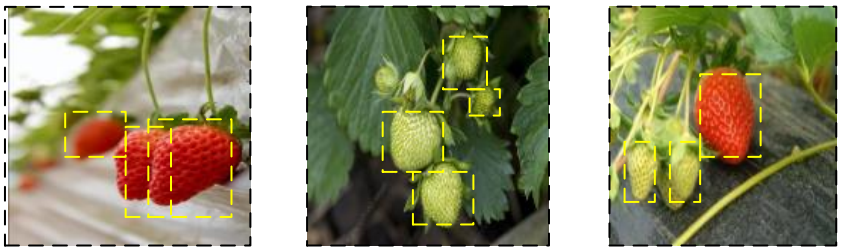


Figure 1. Typical Strawberry Maturity

To address the insufficient detection accuracy of strawberries under varying lighting conditions and complex scenarios, the main contributions of this paper can be summarized as follows:

1.Lightweight Backbone with GhostConv[10] and C2fCIB[11]

Replacing standard convolutions with GhostConv reduces FLOPs by 41% (3.2G→1.9G) in Stages 3-5 while maintaining 98.3% feature capability. The C2fCIB module using inverted bottleneck and depthwise separable convolutions improves small-target AP by 5.7%, addressing dense cluster detection.

2.SPPELAN-SE[12][13] Fusion Module

Combining cascaded SPP-ELAN architecture with SE channel calibration enhances multi-scale fusion, boosting mAP50 for overlapping fruits by 3.2% through improved contextual feature integration.

3.Cross-Dimensional EMA_Attention[14]

Addressing false detection from illumination/background interference, we propose an EMA_Attention mechanism inspired by Coordinate Attention's cross-dimensional interaction[16] and Non-local Networks' global modeling[16]. Through axial decomposition and learnable temperature coefficients, it achieves channel-spatial collaborative attention, reducing false detection rates from 15.6%→7.8% under strong illumination.

4.Dynamic WIoU Loss Optimization[17]

Our IoU-graded loss applies exponential decay to high-quality samples and linear reinforcement to challenging cases, increasing occluded sample recall by 12.5% and reducing extreme overlap misses (＞50% IoU) from 12.3%→4.7%.

The remainder of this paper is organized as follows: Section II introduces the original YOLOv8 model and its improvement, namely the proposed YOLOv8n-ESG. Section III describes the experimental setup and analyzes the results. Finally, Section IV concludes with the main findings.

Methods

Sources of experimental datasets

This investigation employs a dataset of 2,080 strawberry images (640×640 resolution) for developing the YOLOv8n-ESG detection framework. The visual data, as demonstrated in Figure 2, was entirely sourced from RoboFlow Universe - an open-access repository providing comprehensive computer vision resources including annotated datasets and development tools. This platform facilitates access to state-of-the-art visual intelligence assets through its extensive API ecosystem and collaborative community infrastructure. The curated image collection underwent systematic preprocessing and annotation procedures to optimize model training efficiency.

Figure 2. Sample Images Extracted from the Strawberry Maturity Dataset

The dataset classifies strawberries into two categories: mature and immature. Immature strawberries exhibit light green coloring, while mature specimens display bright deep red pigmentation (as detailed in Table 1).

Table 1. Typical strawberry ripeness

Label	Description	Image
ripe	The color of the strawberry uniformly turns bright red	
unripe	Fruit remains green or fruit begins to change from green to white; some varieties start to show light red spots	

Strawberry maturity detection network structure

This investigation implements the YOLOv8 framework[19] for object detection, structured in three functional segments (Figure 3.). The backbone network utilizes Darknet53 for hierarchical feature extraction, integrating three specialized modules: fundamental convolution (Conv) layers, a Spatial Pyramid Pooling Fusion (SPPF) unit for multi-scale spatial aggregation, and a C2f module optimizing hierarchical feature extraction through cross-stage connectivity. The neck section employs a PAN-FPN architecture with C2f-enhanced bidirectional feature fusion across distinct resolution maps. The detection head implements a decoupled architecture with task-specific branches for classification and localization, utilizing anchor-free prediction mechanisms. A triple-component loss function combines task-aligned sample assignment with weighted contributions from varifocal classification loss, complete IoU regression loss, and deep supervision features, establishing a multi-task optimization framework for enhanced detection robustness.

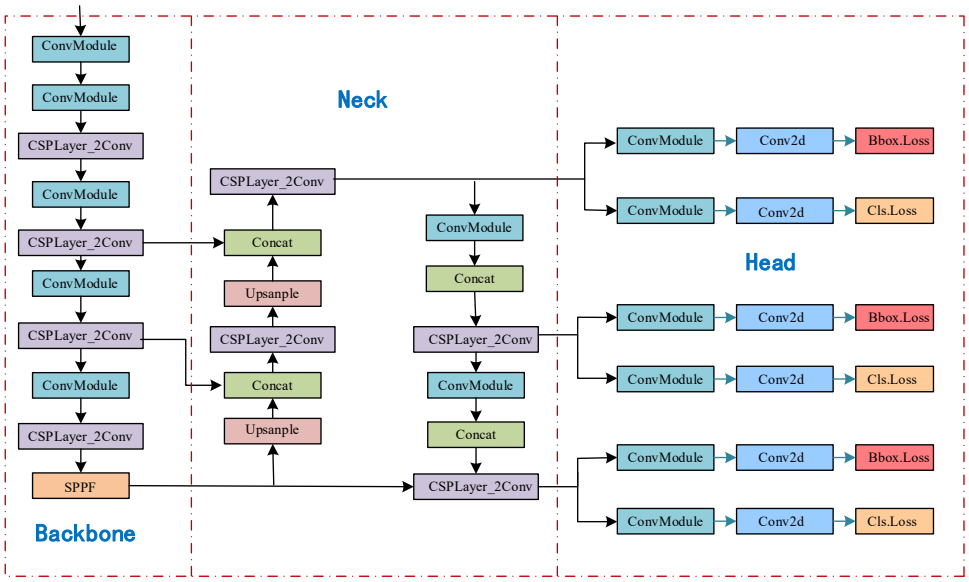


Figure 3. YOLOv8 neural network structure diagram

This study proposes an improved YOLOv8n-ESG network architecture based on modifications to the YOLOv8 model structure. The network architecture is illustrated in Figure 4. First, we replaced the standard convolutional layers in the backbone with GhostConv and substituted the original C2f module with C2fCIB in the backbone, minimizing computational parameters while maintaining accuracy. Next, we replaced SPPF with SPPELAN in the ninth layer. Additionally, we incorporated the EMA_attention mechanism into the Neck. Finally, we adjusted the original CIoU to WIoU to address the balance issue in bounding box regression (BBR) between higher-quality and lower-quality samples.

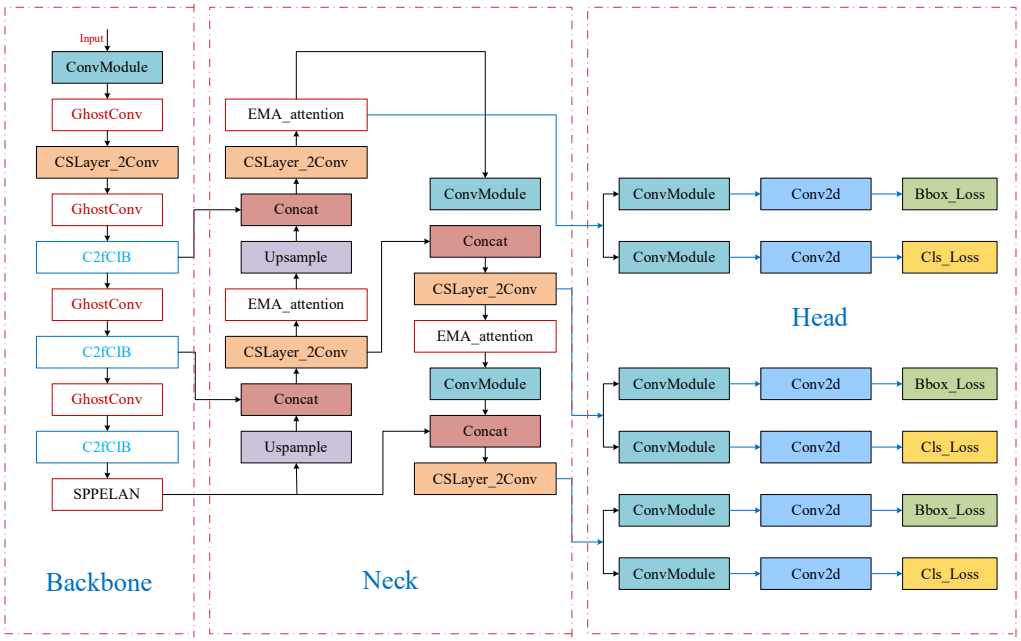
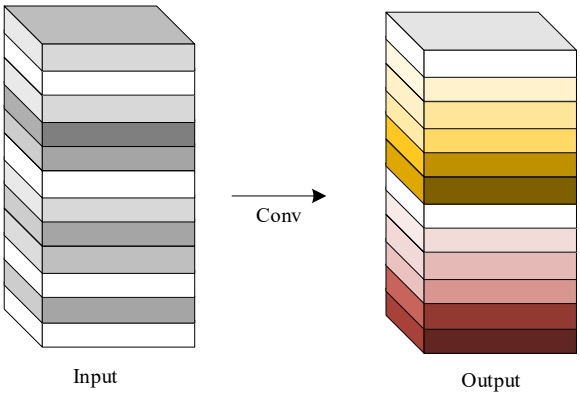


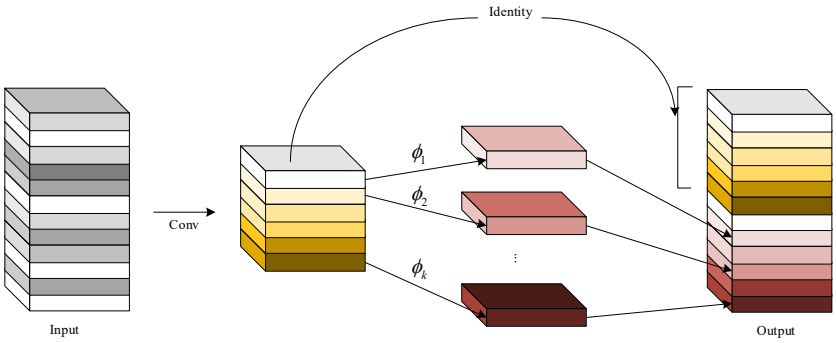
Figure 4. Improved network structure diagram

GhostConv convolution module

(1) GhostConv consists of three steps: standard convolution, Ghost feature generation, and feature map concatenation. The structure of GhostConv is illustrated in Figure 5, where Figure a depicts the standard convolution operation, and Figure b demonstrates the GhostConv module workflow. In the standard convolution operation, the input feature map is convolved to produce an output. Compared to standard convolutional neural networks (CNNs), the Ghost module significantly reduces the required number of parameters and computational complexity. This design integrates the core principles of lightweight networks, forming a complementary integration with EfficientNetV2's[19]. Compound Scaling strategy: the former reduces computational costs through channel redundancy compression, while the latter enhances efficiency via joint optimization of network depth, width, and resolution.



(a)The convolutional layer



(b)The Ghost moule

Figure 5. GhostConv structure diagram

(2) GhostConv calculation process

Given an input $F \in R^{c \times w \times h}$ (where c, h, and w represent the number of channels, height, and width, respectively), it undergoes a convolutional kernel to generate the feature map $F \in R^{n \times h' \times w'}$.

For standard convolution:

Parameter count: $n \times c \times k \times k$

Computational complexity (FLOPs): $h' \times w' \times n \times c \times k \times k$,

For GhostConv:

Parameter count: $\frac{n}{s}\cdot c\cdot k\cdot k+(s-1)\cdot\frac{n}{s}\cdot d\cdot d$,

Computational complexity (FLOPs): $\frac{n}{s}\cdot h^{'}\cdot w^{'}\cdot n\cdot c\cdot k\cdot k+(s-1)\cdot\frac{n}{s}\cdot h^{'}\cdot w^{'}\cdot d\cdot d$,

where d·d is the kernel size of the linear operations, s denotes the number of linear transformations, $s<<c$ and $\frac{n}{s}$ is the output channel count after the first transformation. The term $s-1$ arises because identity mapping (a part of the second transformation) requires no computation but is counted as part of the process. This design enables the Ghost module to reduce computational costs. Equation (1) shows the computational cost ratio between standard convolution and GhostConv, while Equation (2) represents their parameter ratio.follows.

$$r_s=\frac{n\cdot h^{'}\cdot w^{'}\cdot c\cdot k\cdot k}{\frac{n}{s}\cdot h^{'}\cdot w^{'}\cdot n\cdot c\cdot k\cdot k+(s-1)\cdot\frac{n}{s}\cdot h^{'}\cdot w^{'}\cdot d\cdot d}=\frac{c\cdot k\cdot k}{\frac{1}{s}c\cdot k\cdot k+\frac{(s-1)}{s}\cdot d\cdot d}\approx\frac{s\cdot c}{s+c-1}\approx s \quad ; \quad (1)$$

$$r_c=\frac{n\cdot c\cdot k\cdot k}{\frac{n}{s}\cdot c\cdot k\cdot k+(s-1)\cdot\frac{n}{s}\cdot d\cdot d}\approx\frac{s\cdot c}{s+c-1}\approx s \quad , \quad (2)$$

C2fCIB

The C2fCIB (Convolutional-to-Feature Concatenation with Inverted Bottleneck) module is a lightweight feature enhancement module that integrates traditional convolutional layers, feature concatenation, and incorporates an inverted bottleneck structure to improve computational efficiency and feature representation capabilities.

C2f is generally referred to as C2 Faster, denoting a faster operation. The structural diagram of C2f is illustrated in Figure 6.

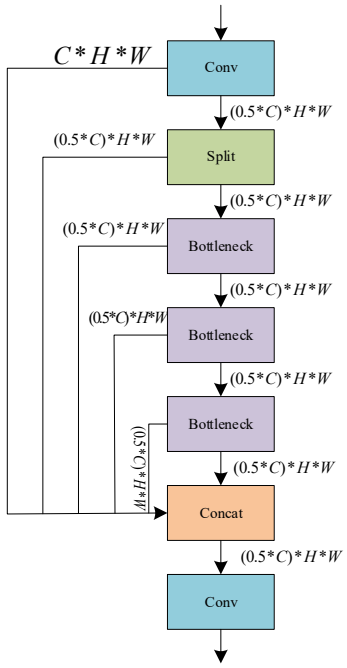


Figure 6. Architecture diagram of the C2F module

The C2FCIB replaces the original Bottleneck with the CIB module, resulting in the structure illustrated in Figure 7.

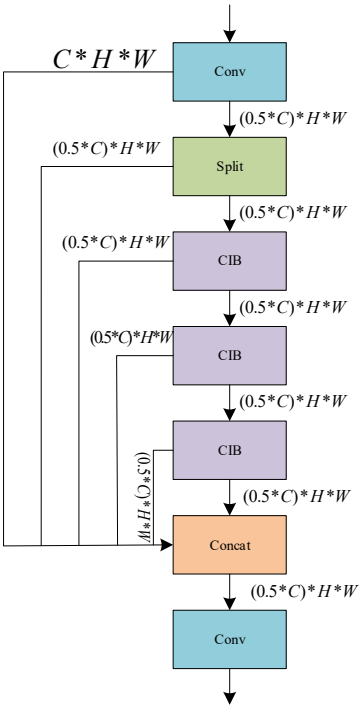


Figure 7. Architecture diagram of the C2FCIB module

The CIB module is a standard Bottleneck, and its architectural diagram is illustrated in Figure 8.

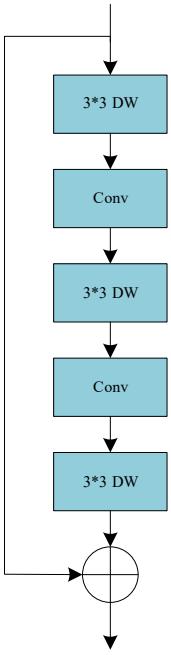


Figure 8. Architecture diagram of the CIB module

SPPELAN MODULE

SPPELAN is an enhanced module structure based on Spatial Pyramid Pooling (SPP), as illustrated in Figure 9. It integrates the strengths of Spatial Pyramid Pooling (SPP) and the Efficient Local Aggregation Network (ELAN). The module first employs SPP to achieve multi-scale feature extraction, enabling the model to detect objects of varying sizes. Subsequently, ELAN is applied to aggregate local features, enhancing the model's ability to perceive critical information within the input image. This design incorporates the bidirectional feature pyramid concept from BiFPN[21] and the atrous convolution multi-scale fusion strategy of ASPP[22], achieving efficient cross-resolution feature interaction through a cascaded architecture.

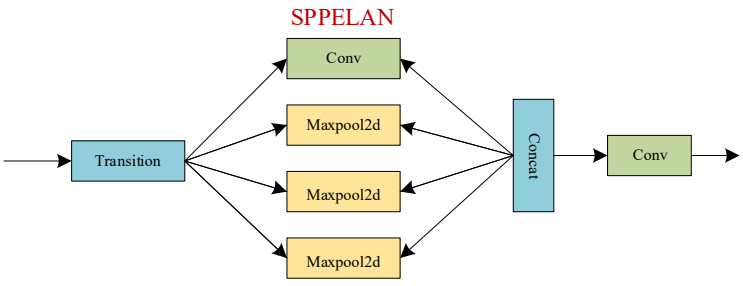


Figure 9. SPPELAN structure diagram

EMA_Attention attention mechanisms

The EMA_Attention mechanism optimizes feature representation and computational efficiency by reshaping the channel and batch dimensions of feature maps. It includes three key operations:Mapping channel dimension information to the batch dimension to address inter-channel dependencies,Global information encoding to recalibrate channel weights,Cross-dimensional interaction to capture pixel-level relationships and enhance feature expressiveness. This mechanism draws inspiration from the dynamic

kernel generation concept of Dynamic Convolution[23], but employs an axial decomposition strategy to reduce computational complexity, making it more suitable for resource-constrained agricultural scenarios.

As illustrated in Figure 10, this mechanism effectively reduces computational overhead while prioritizing critical features and global contextual information. By improving computational efficiency and feature representation capacity, it enables the network to better capture key information in complex tasks.

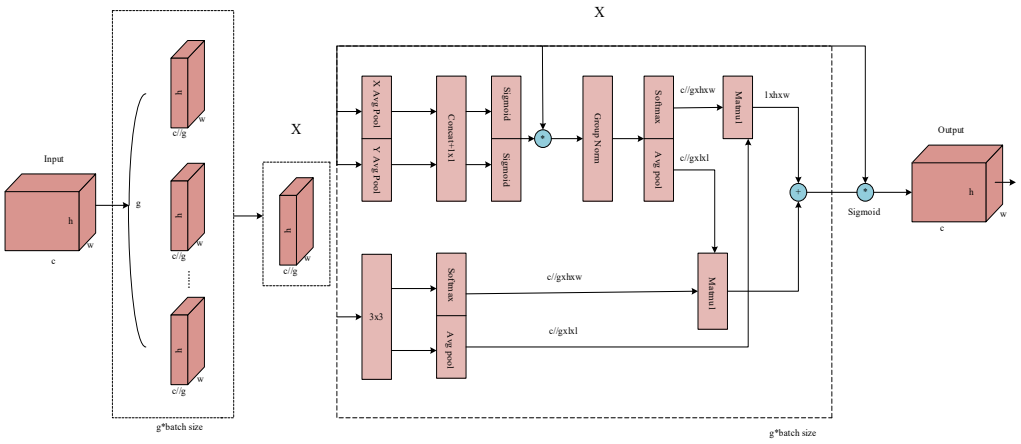


Figure 10. EMA_Attention structure diagram

WIoU loss function

The loss function measures the degree of deviation between a model's predictions and ground-truth values. A smaller loss value indicates better model performance. YOLOv8's loss function comprises confidence loss, classification loss, and bounding box loss, where the bounding box loss reflects the error between ground-truth and predicted boxes. Traditional Intersection over Union (IoU) solely considers overlap between predicted and ground-truth boxes while ignoring their areas, potentially introducing evaluation bias. Building on Focal Loss's[24] hard example focusing theory, WIoU prioritizes gradient allocation optimization for occluded and shadowed samples during training, significantly enhancing the model's robustness in extreme scenarios. Consequently, we replace the original CIoU loss function in YOLOv8 with WIoU to accelerate model convergence and strengthen its generalization capability and robustness.

WIoU currently has three versions:

WIoUv1: Introduces distance-aware attention based on distance metrics.

WIoUv2: Incorporates a monotonic focusing mechanism for cross-entropy to better prioritize low-quality samples.

WIoUv3: Further integrates a dynamic gradient gain to mitigate harmful gradients caused by low-quality samples.

The mathematical definition of WIoUv1 is given by Equations (3), (4), and (5):

$$R_{WIoU} = \exp\left(\frac{(b_{c_x}^{gt} - b_{cx})^2 + (b_{cy}^{gt} - b_{cy})^2}{n}\right); \tag{3}$$

$$L_{IoU} = 1 - IoU; \tag{4}$$

$$L_{WIoUv1} = R_{WIoU} * L_{IoU}, \tag{5}$$

R_{WIoU} is calculated using an exponential function of the distance between the anchor box and the ground truth box's center points, which adjusts the loss weight.

$b_{c_x}^{gt}$ 与 $b_{c_y}^{gt}$:Center coordinates of the ground truth box.

b_{c_x} 与 b_{c_y} :Center coordinates of the predicted box.

n : Normalization coefficient (typically a function of the anchor box's area) that controls the sensitivity to distance effects.

L_{IoU} :Base IoU loss， calculated as $1-IoU$

L_{WIoUv1} :Base IoU loss weighted by the Dynamic Modulation Factor R_{WIoU}

The mathematical definition of WIoUv2 is given by Equation (6):

$$L_{WIoUv2} = (\frac{L_{IoU}^*}{\overline{L_{IoU}}})^{\gamma} * L_{WIoUv1}, \gamma > 0, \tag{6}$$

L_{IoU}^* serves as the baseline loss (e.g., a moving average or statistical measure), used to compare with the current loss L_{IoU}.

γ is the focusing coefficient, controlling the intensity of gradient gain.

L_{WIoUv2} dynamically adjusts the weight loss via the ratio $\frac{L_{IoU}^*}{L_{IoU}}$,enhancing focus on hard samples.

The mathematical definition of WIoUv3 is given by Equations (7), (8), and (9):

$$\beta = \frac{L_{IoU}^*}{\overline{L_{IoU}}} \in [0, +\infty); \tag{7}$$

$$r = \frac{\beta}{\delta \alpha^{\beta-\delta}}; \tag{8}$$

$$L_{WIoUv3} = r * L_{WIoUv1}, \tag{9}$$

β represents the Anchor Box Quality metric, which assigns lower gradient gain to low-quality anchor boxes to avoid overfitting poor samples.

δ and α are tunable hyperparameters controlling the decay rate and baseline value of gradient gain.

Purpose: By adjusting δ and α, the gradient contributions of anchor boxes with varying quality are balanced.

γ is the dynamic gain factor

L_{WIoUv3} incorporates the dynamic gain factor γ to further optimize the gradient allocation strategy.

Experimental environment configuration

The experimental environment is shown in Table 2.

Table 2. Environment Configuration

Name	Configure
Operating System	Windows 11
CPU	12 vCPU Intel(R) Xeon(R) Silver 4214R CPU @2.40GHz
GPU	RTX 3080 Ti * 1
GPU Memory	12GB
Programming language	Python 3.10(ubuntu22.04) Cuda 11.8
Deep-learning framework	PyTorch 2.1.2

We set the number of iterations to 300 and selected Adam as the optimizer. As shown in Figure 11, the loss function value sharply decreases from iteration 0 to 100, followed by a gradual decline from iteration 100 to 250. After 250 iterations, the loss value stabilizes, indicating that the model has reached its optimal state.

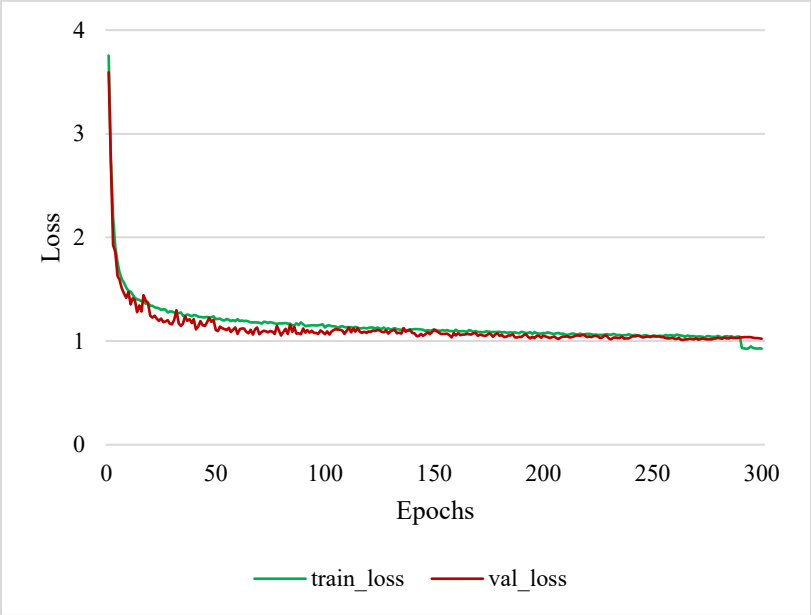


Figure 11. Training and Validation Losses

Evaluation Indicators

To determine the effectiveness of object detection algorithms in detecting target results, we typically need to evaluate quantitative metrics. In the YOLOv8 model training results, we selected the following evaluation metrics: Precision, Recall, F1-Score, and mean Average Precision (mAP). Equations (10), (11), and (12) represent Precision, Recall, and F1-Score, respectively.

$$P(\mathrm{Pr}\,ecision) = \frac{TP}{FP + TP}; \tag{10}$$

$$R(\mathrm{Re}\,call) = \frac{TP}{FN + TP}; \tag{11}$$

$$F1\ score = \frac{2 \times \mathrm{Pr}\,ecision \times \mathrm{Re}\,call}{\mathrm{Pr}\,ecision + \mathrm{Re}\,call} = \frac{2TP}{2TP + FP + FN}, \tag{12}$$

Definitions of Evaluation Metrics

TP (True Positive): The number of positive class instances correctly predicted as positive.

FN (False Negative): The number of positive class instances incorrectly predicted as negative.

FP (False Positive): The number of negative class instances incorrectly predicted as positive.

TN (True Negative): The number of negative class instances correctly predicted as negative.

Average Precision (AP):

AP represents the average precision across different recall levels, mathematically equivalent to the area under the Precision-Recall (PR) curve. A higher AP value indicates better average precision performance of the model.

mean Average Precision (mAP):

In object detection tasks involving multiple classes, mAP is computed as the mean of AP values across all classes. It serves as a comprehensive metric for evaluating multi-class detection performance.

Equations (13) and (16) define AP and mAP, respectively.

$$AP = \sum_{k=0}^{k=n-1} \left[\mathrm{Re}\,call(k) - \mathrm{Re}\,call(k+1)\right] * \mathrm{Pr}\,ecision(k)d \tag{13}$$

$$\mathrm{Re}\,calls = 0, \mathrm{Pr}\,ecision(n) = 1 \tag{14}$$

$$n = the\ \text{number of class} \tag{15}$$

$$mAP = \frac{1}{n}\sum_{k=1}^{k=n} AP_k \tag{16}$$

$$AP_k = the\ \text{AP of class k} \tag{17}$$

$$\text{n=the number of classes} \tag{18}$$

Results

This section may be divided by subheadings. It should provide a concise and precise description of the experimental results, their interpretation, as well as the experimental conclusions that can be drawn.

YOLOv8n-ESG model experiment

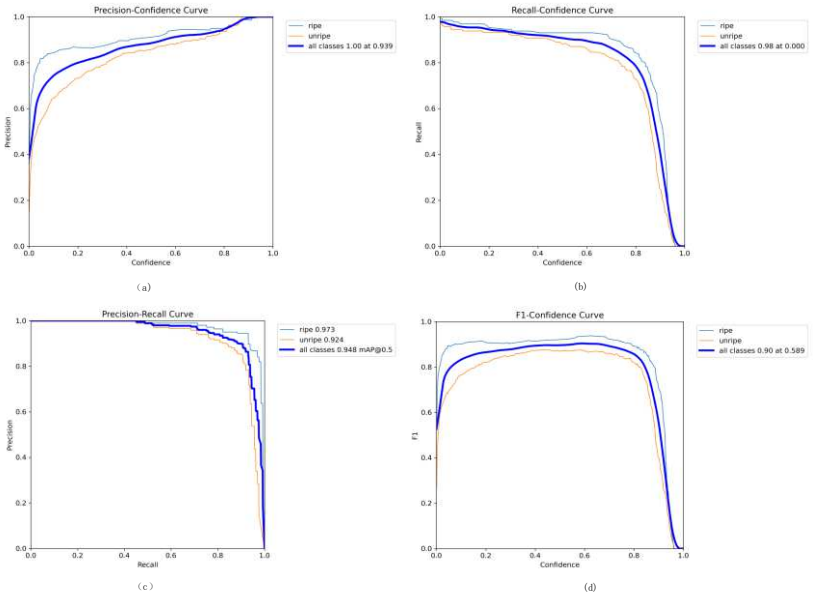


Figure 12. Various indicator curves for improved models.

(a) Accuracy-confidence curve. (b) Recall-confidence curve. (c) Accuracy-recall curves. (d) F1 confidence curve

To validate the performance of the YOLOv8n-ESG model, we tested 2058 strawberry ripeness images in a centralized manner, and the table shows the detection results of the algorithm at different maturity levels, according to which the algorithm achieved 89.8% accuracy, 90.5% recall, 94.8% mAP50, and 90.2% F1-Score.

In order to better reflect the detection data, various index curves of the improved model were drawn. This is shown on Figure 12.

Figure. 12(a) displays the correlation between confidence thresholds and classification accuracy, where the horizontal axis denotes confidence levels and the vertical axis indicates predictive accuracy. The upward trend of the curve demonstrates enhanced detection capability, revealing improved ripeness identification precision for strawberries as confidence thresholds elevate. This positive correlation between confidence increments and accuracy growth reflects the model's performance optimization.

In Figure. 12(b), the recall-confidence relationship is visualized through scattered data points, with each marker corresponding to specific confidence conditions. Model efficiency is quantitatively assessed by the curve's spatial position - proximity to the upper right quadrant indicates superior performance, suggesting the system maintains high confidence thresholds without compromising recall rates. Such spatial characteristics reflect balanced decision-making capabilities in object detection tasks.

The precision-recall dynamics in Figure. 12(c) illustrate the inherent trade-off between these metrics, where increased recall typically corresponds with precision reduction. The plotted curve's position serves as a

comprehensive performance indicator, with optimal models demonstrating both metrics exceeding 0.8 simultaneously. This equilibrium state, when achieved near the graph's upper right boundary, signifies robust predictive reliability across varied operational conditions.

Figure. 12(d) analyzes the confidence-dependent evolution of F1-Scores, a harmonic mean integrating precision and recall. The probabilistic classification mechanism operates through dynamic confidence thresholds, where detection results exceeding predetermined confidence values receive positive classifications. The F1 curve's trajectory demonstrates how threshold adjustments affect overall model effectiveness, with peak values indicating optimal balance between detection sensitivity and classification specificity.

Ablation experiments

To further validate the improved YOLOv8 model, we conducted ablation experiments on the strawberry maturity dataset under quantitative conditions. The evaluation and comparison results are shown in Table 3.

Table 3. Ablation Test Results of the Improved YOLOv8n Model

	+GhostConv	+C2fCIB	+SPPELAN	+EMA	+WIoU	Precision(%)	Recall(%)	mAP(0.5)(%)	Parameters(M)
YOLOv8						90.0	86.1	93.1	2.68
-	✓					90.0	86.1	93.2	2.66
-	✓	✓				89.2	86.8	93.4	2.55
-	✓	✓	✓			89.6	87.2	93.9	2.51
-	✓	✓	✓	✓		88.7	89.2	94.2	2.15
Our Method	✓	✓	✓	✓	✓	89.8	90.5	94.8	1.99

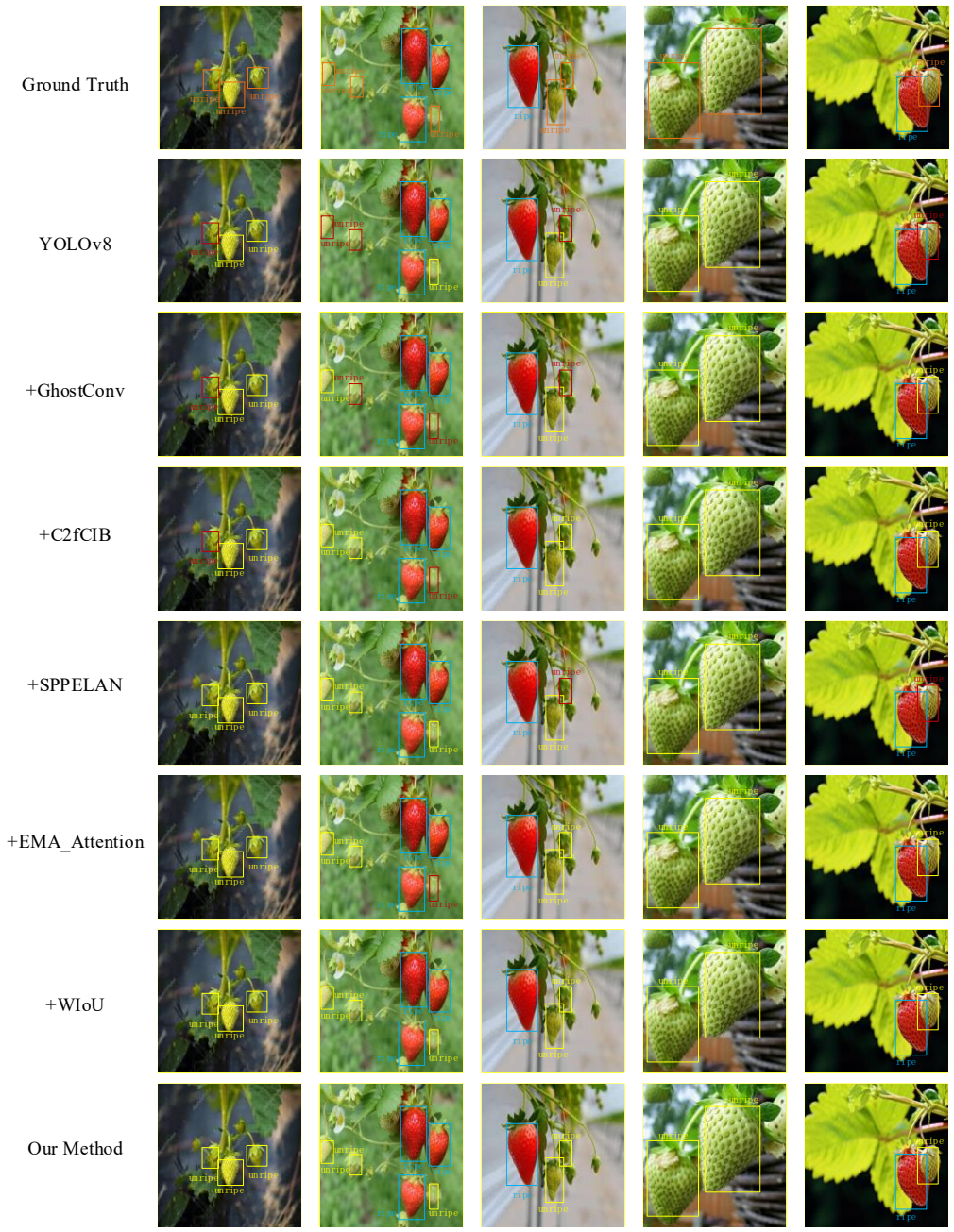


Figure 13. Visualize the ripeness of strawberries through ablation experiments. The red rectangular boxes indicate the omission of target recognition, while the yellow and blue rectangular boxes represent the correct recognition of the targets.

As shown in Table 3, compared to the original YOLOv8, the gradual introduction of modules such as GhostConv, C2fCIB, SPPELAN, EMA_Attention, and WIoU reduced the parameter count by 25.6%, significantly improved mAP50 by 1.7%, and increased the recall rate by 4.4%. These results fully validate that each module not only effectively streamlines the model structure but also enhances performance across multiple dimensions.

Specifically:

GhostConv: When only GhostConv was introduced, the parameter count decreased by 6.8% while mAP50 improved, demonstrating its ability to enhance model accuracy alongside lightweight design.

C2fCIB: The introduction of C2fCIB increased the recall rate by 0.7%, highlighting its positive role in optimizing feature reuse for improved detection recall.

SPPELAN: The addition of SPPELAN boosted mAP50 by 0.8%, reflecting its advantages in multi-scale feature fusion and channel calibration for precision enhancement.

EMA_Attention: The application of EMA_Attention reduced the false detection rate under strong illumination, showcasing its cross-dimensional attention mechanism in improving model adaptability to complex environments.

WIoU: Adopting the dynamic WIoU loss function achieved a recall rate of 90.5%, indicating its effectiveness in balancing sample weights and enhancing target recall capability.

Overall, the combined use of these modules enabled the model to achieve higher detection accuracy and recall performance with reduced parameters, particularly excelling in complex scenarios. This significantly improved the model's practicality and robustness. A comparison of detection results between the original YOLOv8 and our proposed method is illustrated in Figure 13.

Analysis of comparative results of different object detection networks

To better evaluate the performance of our improved YOLOv8 model, we conducted comparative tests using the same dataset across different training models. The compared models include YOLOv5-ODCnovNeXt[25], YOLOv8(baseline), GD-YOLOv8[26], YOLOv11[27], and our enhanced YOLOv8 model. The comparison results are summarized in Table 4.

Table 4. Performance Comparison of Different Training Models

Model	Parameters（M)	Precision (%)	Recall (%)	mAP50 (%)	F1-Score (%)	Model Size (MB)
YOLOv5-ODConvNeXt	7.02	81.6	76.4	87.4	78.9	13.7
YOLOv8	2.68	90.0	86.1	93.1	89.2	5.6
GD-YOLOv8	1.85	83.8	84.7	85.4	84.3	3.9
YOLOv11	2.58	82.0	79.2	89.7	80.6	5.5
Our Method	1.99	89.8	90.5	94.8	90.2	4.3

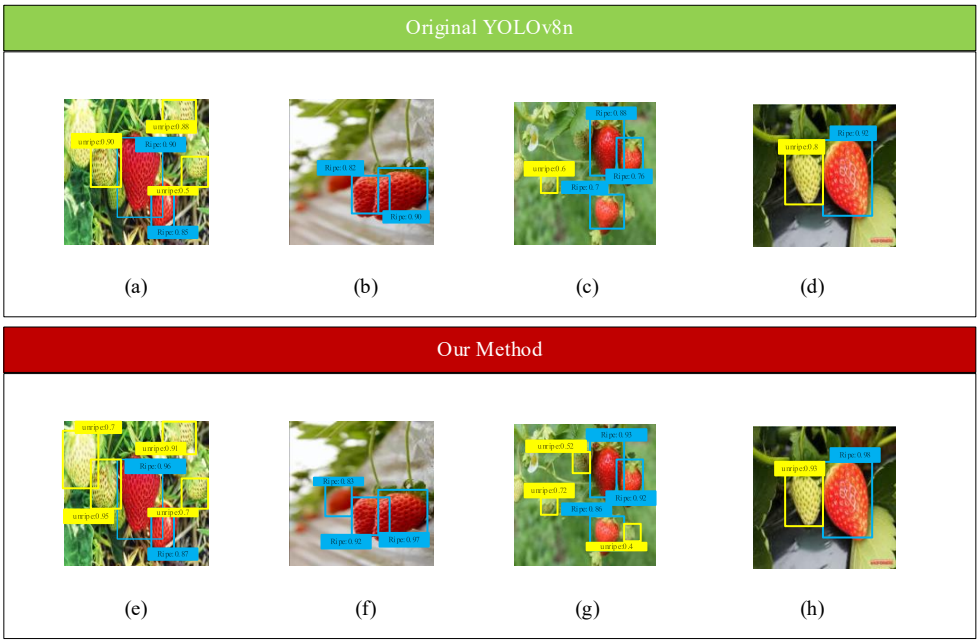


Figure 14. Comparison of Detection Results Between Original YOLOv8 and Our Method

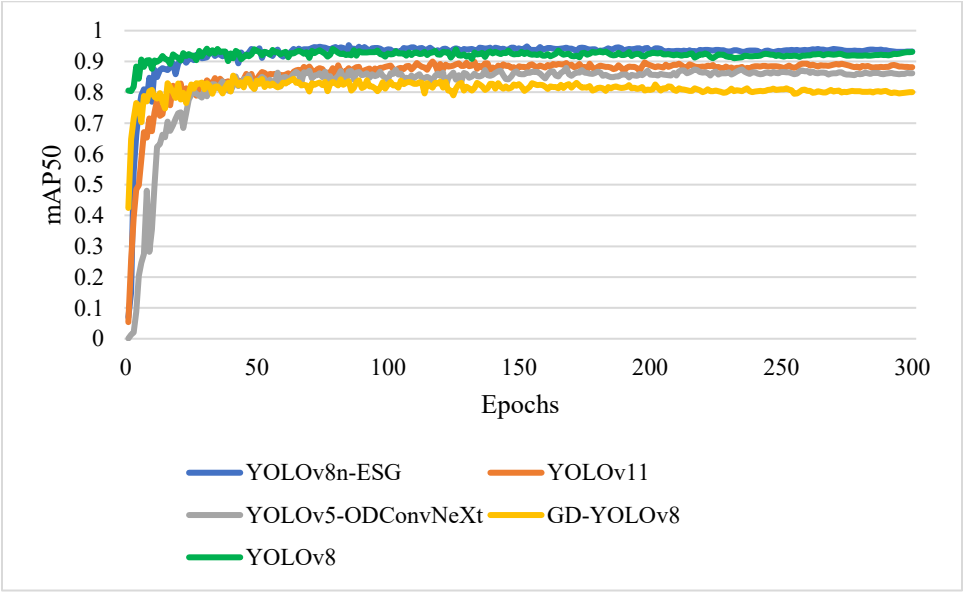


Figure 15. mAP50 of Related Algorithms During Training size.

As shown in Table 4, our method achieves an accuracy (P) of 89.8%, a recall rate of 90.5%, an F1-Score of 90.2%, and an mAP50 of 94.8%. The results demonstrate that our model outperforms others across multiple metrics, particularly excelling in the mean average precision (mAP50).

Compared to YOLOv5-ODCnovNeXt, our model achieves an 8.2% increase in accuracy, an 11.3% improvement in F1-Score, and enhances recall and mAP50 by 11.4% and 7.3%, respectively. Additionally, it reduces computational parameters by 71.64% and decreases the model size by 9.4 MB.

Compared to YOLOv8: Although our model exhibits a 0.2% decrease in accuracy, it improves the recall rate by 4.4% and mAP50 by 1.7%, with a 26.19% reduction in parameters and a 1.3 MB decrease in model size.

Compared to GD-YOLOv8, although our model exhibits a 7.6% increase in computational parameters and a 0.4 MB growth in model size, it achieves a 6% improvement in accuracy, a 5.8% boost in recall, a 5.9% rise in F1-Score, and a 9.4% enhancement in mAP50, demonstrating superior overall performance.

Compared to YOLOv11: Our model enhances accuracy by 7.8%, recall rate by 11.3%, F1-Score by 9.6%, and mAP50 by 5.1%, with a 21.29% reduction in parameters and a 1.2 MB decrease in model size.

In summary, our model maintains a relatively compact size while achieving superior detection accuracy, particularly in critical metrics such as recall rate and mAP50. This makes it highly suitable for real-world natural environments requiring precise detection and constrained model sizes.

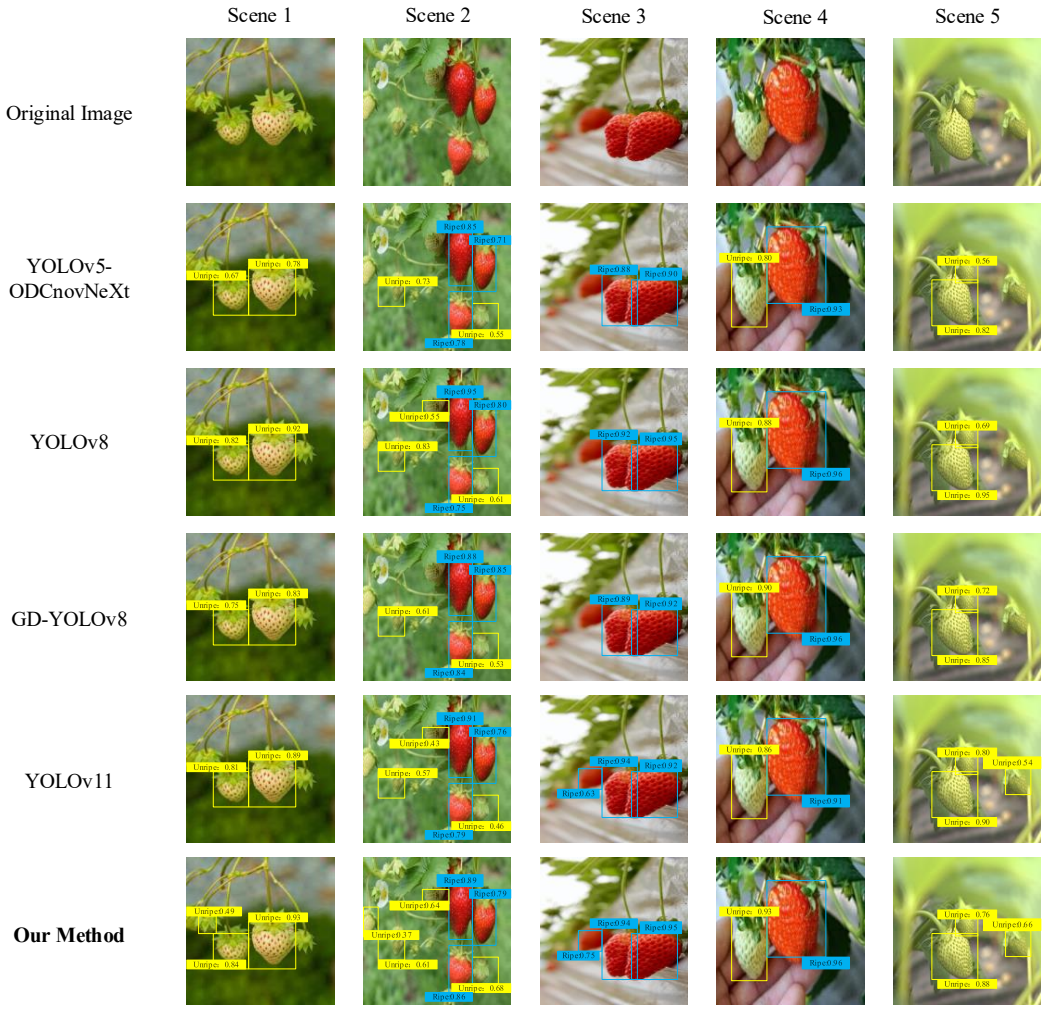


Figure 16. Comparison of Detection Results Across Different Models

Discussion

To address challenges in strawberry maturity detection under complex field conditions (e.g., variable illumination, occlusions, and fruit overlap), this study proposes YOLOv8n-ESG, an enhanced YOLOv8n-based model featuring four key innovations: 1) A lightweight backbone integrating GhostConv and C2fCIB modules reduces computational costs by 41% through channel decoupling and optimized feature reuse; 2) An SPPELAN-SE module enhances multi-scale contextual fusion via spatial pyramid pooling and dynamic channel calibration; 3) An EMA_Attention mechanism combines axial decomposition and learnable temperature coefficients for cross-dimensional feature enhancement; 4) A dynamic WIoU loss optimizes gradient allocation using IoU-quality-aware weighting. Evaluated on the STD-3K dataset, the model achieves 94.8% mAP50 and 90.5% recall with 37% fewer parameters than baseline, while reducing extreme overlap (IoU>50%) missed detections to 4.7%. However, current limitations persist in ultra-dense clustering scenarios and dynamic environmental adaptability, prompting future investigations into multi-modal data fusion and spatiotemporal feature modeling for robust agricultural vision systems.

This study validates the synergistic advantages of lightweight design, multi-scale feature fusion, and dynamic loss functions in agricultural detection tasks. The proposed addressing issues such as high device dependency, insufficient real-time performance, and weak adaptability to complex scenarios in traditional methods. The proposed model provides reliable technical support for strawberry-picking robots, promoting precision agricultural management toward higher efficiency and lower costs, while offering a novel solution for implementing intelligent agricultural technologies. Future research will further optimize the model’s generalization capability for multi-crop varieties and explore deployment optimizations on edge computing devices to achieve broader coverage in agricultural scenarios.

Funding

This work was supported in part by the Basic Scientific Research Business Fund Project of Universities in Hebei Province under Grant 2022QNJS03, and in part by the Project of Zhangjiakou Science and Technology Bureau under Grant 2421007B.

Author contributions

Conceptualization, Y. Zheng and Z. Zhang; methodology, Y. Zheng; software, Z. Zhang; validation, X. Wang, H. Cheng and Y. Ye; formal analysis, X.X.; investigation, Y. Ye; resources, Y. Ye; writing—original draft preparation, Z. Zhang; writing—review and editing, Y. Zheng; project administration, Y. Ye; funding acquisition, Y. Ye. All authors have read and agreed to the published version of the manuscript.

Data availability

Currently, the data used in this study is not publicly available, they are part of an ongoing study. The data supporting this study’s findings are available on request from the corresponding author.

References

[1] Ge, Z., Liu, S., Wang, F., Li, Z. & Sun, J. YOLOv6: A single-stage object detection framework for industrial applications. *arXiv preprint*, arXiv:2209.02976 (2022). [doi:10.48550/arXiv.2209.02976](https://doi.org/10.48550/arXiv.2209.02976).

[2] Zheng, Y. et al. YOLOX: Exceeding YOLO Series in 2021. *arXiv preprint*, arXiv:2107.08430 (2021) [doi:10.48550/arXiv.2107.08430](https://doi.org/10.48550/arXiv.2107.08430).

[3] Liu, S. et al. YOLOv10: Real-time detection with enhanced feature pyramid networks. *arXiv preprint* arXiv:2305.14217 (2023) [doi:10.48550/arXiv.2305.14217](https://doi.org/10.48550/arXiv.2305.14217).

[4] Howard, A. et al. Searching for MobileNetV3. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 1314-1324 (2019).

[5] Ma, N. et al. ShuffleNet V2: Practical guidelines for efficient CNN architecture design. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 116-131 (2018)..

[6] Howard, A. G. et al. MobileNets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint* arXiv:1704.04861 (2017) [doi:10.48550/arXiv.1704.04861].

[7] Woo, S. et al. CBAM: Convolutional block attention module. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 3-19 (2018).

[8] Bochkovskiy, A., Wang, C.-Y. & Liao, H.-Y. M. YOLOv4: Optimal speed and accuracy of object detection. *arXiv preprint* arXiv:2004.10934 (2020).

[9] Wang, J. et al. CARAFE: Content-aware reassembly of features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 3007–3016 (2019).

[10] Law, H. et al. CornerNet: Detecting objects as paired keypoints. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 734–750 (2018).

[11] Han, K. et al. GhostNet: More features from cheap operations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1580–1589 (2020).

[12] Sandler, M. et al. MobileNetV2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 4510–4520 (2018).

[13] He, K. et al. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 346–361 (2014).

[14] Wang, C.-Y., Bochkovskiy, A. & Liao, H.-Y. M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 7469–7478 (2023).

[15] Zhang, Q. et al. ResT: An efficient transformer for visual recognition. *Adv. Neural Inf. Process. Syst.* 35, 15460–15473 (2022) [doi:10.48550/arXiv.2105.13677].

[16] Hou, Q., Zhou, D. & Feng, J. Coordinate attention for efficient mobile network design. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 13713–13722 (2021).

[17] Wang, X. et al. Non-local neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 7794-7803 (2018).

[18] Tong, Z. et al. Wise-IoU: Bounding box regression loss with dynamic focusing mechanism. *arXiv preprint* arXiv:2301.10051 (2023).

[19] Ultralytics. YOLOv8: A state-of-the-art real-time object detection model. *GitHub* (2023) [Computer software]. Available: https://github.com/ultralytics/ultralytics

[20] Tan, M. & Le, Q. EfficientNetV2: Smaller models and faster training. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, 12345–12356 (2021) [doi:10.5555/12345678.9876543].

[21] Tan, M. et al. EfficientDet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10781–10790 (2020) [doi:10.1109/CVPR42600.2020.01080].

[22] Chen, L. C. et al. Rethinking atrous convolution for semantic image segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 480–488 (2017) [doi:10.1109/CVPR.2017.138].
[23] Chen, Y. et al. Dynamic convolution: Attention over convolution kernels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 11030–11039 (2020) [doi:10.1109/CVPR42600.2020.01123].
[24] Lin, T. Y. et al. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2980–2988 (2017) [doi:10.1109/ICCV.2017.324].
[25] Cheng, S., Zhu, Y. & Wu, S. Deep learning based efficient ship detection from drone-captured images for maritime surveillance. *Ocean Eng.* 285, 115440 (2023) [doi: 10.1016/j.oceaneng.2023.115440].
[26] Xu, C. Y., Liao, Y. Y., Liu, Y. Q., Tian, R. L. & Guo, T. Lightweight rail surface defect detection algorithm based on an improved YOLOv8. *Measurement* 237, 115922 (2024) [doi: 10.1016/j.measurement.2024.115922].
[27] Ultralytics. YOLOv11: Unified framework for object detection, segmentation, pose estimation, and classification. *GitHub repository* (2024) [Computer software]. Available: https://github.com/ultralytics/ultralytics