

The Lightweight Cherry Tomato Fruit Maturity Detection Method Based on Improved YOLOv10n

Ji Cai

Shanxi Agricultural University

Shuaishuai Cui

Shanxi Agricultural University

Baofan Chen

Shanxi Agricultural University

Yuhao Hao

Shanxi Agricultural University

Kun Wang

Shanxi Agricultural University

Guozhu Song

`songguozhu@sxau.edu.cn`

Shanxi Agricultural University

Article

Keywords:

Posted Date: June 25th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-7952907/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.
[Read Full License](#)

Additional Declarations: No competing interests reported.

The Lightweight Cherry Tomato Fruit Maturity Detection Method Based on Improved YOLOv10n

Ji Cai^{1,+}, Shuaishuai Cui^{1,+}, Baofan Chen¹, Yuhao Hao¹, Kun Wang¹, and Guozhu Song^{1,*}

¹College of Software, Shanxi Agricultural University, Taigu, China

*corresponding.songguozhu@sxau.edu.cn

+these authors contributed equally to this work

ABSTRACT

In the detection and recognition algorithms for the maturity of cherry tomatoes, it is difficult to simultaneously achieve high precision, fast recognition speed, and a small model size. To tackle this problem, this paper presents a lightweight method for detecting the maturity of cherry tomato fruits. With YOLOv10n as the baseline model, we propose the YOLOv10n-FBD model. Firstly, the CCFM (Channel Cross Feature Modulation) feature fusion technique is adopted to enhance information interaction between different feature layers, thereby improving feature representation capabilities. Secondly, the PSA-BiFormer attention mechanism is introduced to effectively strengthen the model's focus on key features, further enhancing detection accuracy. Finally, a C2f-Dual lightweight strategy is implemented to significantly reduce computational complexity and model size without compromising performance. Experimental results show that the improved YOLOv10n-FBD algorithm achieves a mAP@.5:95 of 94.3%, which is an improvement of 2.2% compared to the traditional YOLOv10n model. Meanwhile, the parameter count is reduced by 37.5%, and the model size is compressed to 3.5 MB, a decrease of 38.6%. With only 6.0 GFLOPs, the algorithm achieves a high frame rate of 369.0 frames per second. Compared with models such as Faster R-CNN, SSD, YOLOv5, YOLOv7, YOLOv8 and YOLOv11, the YOLOv10n-FBD model demonstrates improvements of 15%, 10.8%, 2%, 5.7%, 3.9%, and 2.2% respectively in the mAP@.5:95 metric. Additionally, it shows significant optimizations in terms of parameter count, FLOPs, and model size. The YOLOv10n-FBD model achieves high-precision detection of maturity stages for cherry tomato fruits while effectively reducing the model's parameter count and complexity.

Introduction

Tomatoes are a widely cultivated and economically important crop in China, with their high yield significantly contributing to farmer incomes and public nutrition¹. However, traditional harvesting methods remain heavily labor-intensive, resulting in high costs, low efficiency, and fruit damage, which collectively constrain agricultural productivity². Consequently, advancing the mechanization and intelligent automation of tomato harvesting has become an urgent priority for modern agriculture^{3,4}. Intelligent harvesting systems are valued for automation and efficiency, yet practical deployment remains difficult. The chief obstacle is insufficient detection/recognition accuracy and high miss rates, which hinder reliable assessment of tomato maturity and reduce picking precision and efficiency. In addition, high computational complexity and large model size impose substantial operational burdens—raising processing and storage demands and degrading device performance—thereby limiting real-world deployment prospects⁵. In recent years, the rapid development of deep learning in computer vision has offered new pathways for tomato fruit maturity detection. Owing to strong feature extraction and pattern recognition capabilities, deep learning has achieved notable results across domains, and agriculture is no exception. Many scholars are improving algorithms to boost the accuracy and reliability of tomato maturity detection. Miao Ronghui⁶ introduced MobileNetV3 as the backbone in YOLOv7 to reduce parameters and added a Global Attention Mechanism (GAM) in the fusion network to enhance feature representation; the improved model reached 98.6% precision, 98.1% recall, and 98.2% mAP, with 82 ms per image and 66.5 MB memory. Hou Guolong⁷ proposed DSP-YOLOv7-CA for detecting occluded cherry tomatoes, achieving 98.86% mAP and superior performance under natural occlusion. Yang Xinting⁸ fused bilateral images to obtain the global red proportion, built datasets across maturity stages, and optimized Swin Transformer initialization via transfer learning, achieving 95.783% accuracy, surpassing VGG16, AlexNet, and ResNet50 by 2.83%, 3.35%, and 12.34%, respectively. Chang Wenlong⁹ incorporated growth-environment characteristics, introduced MobileNetV3 and ECANet, and improved YOLOv5s; compared with the original YOLOv5, accuracy, recall, and mAP increased by 3.4%, 2.4%, and 2.3%, while model weight decreased by 48.6% and detection speed improved by 52.9%. Wang Aichen¹⁰ designed a C2f-N module based on NAM within NVW-YOLOv8s to enhance feature extraction in complex scenes; with an enhanced balanced dataset, detection performance improved by 4.8%, reaching 91.4% mAP@.5 and 85.4% F1. Huang Wenyuan¹¹ proposed AITP-YOLO, an improved YOLOv10s-based model for tomato ripeness detection that improves accuracy and efficiency under complex conditions; Variable Focus Loss

(VFL) and a Wise-IoU (WIoU)-based regression loss were adopted to address positive–negative imbalance and small/occluded targets. Zeng Taiheng¹² developed a lightweight improved YOLOv5 for real-time localization and maturity detection by replacing the focus layer, reconstructing the backbone, pruning channels, and tuning hyperparameters via genetic algorithms; parameters and FLOPs dropped by 78% and 84.15%, with 96.9% mAP@.5. NCNN quantization enabled an Android app to reach 26.5 FPS, a 268% speedup, 51.1% size reduction, and 93% real-world detection rate. Zhu Xueyan¹³ proposed YOLO-LM (based on YOLOv7-tiny) that introduces three Criss-Cross Attention (CCA) modules in the backbone, adopts ASFF as the head, and replaces standard convolutions with GSConv in the neck; it achieves 93.96% precision, 93.32% recall, 93.18% mAP@.5, 10.17M parameters, 19.46 GFLOPs, and 19.82 MB model size, improving mAP@.5 over YOLOv3-tiny, YOLOv4-tiny, YOLOv5s, YOLOv7-tiny, and YOLOv8s by 6.24%, 6.11%, 2.52%, 2.45%, and 0.57%, respectively, albeit with slightly higher parameters and size than several mainstream models. Although the above methods have achieved certain effectiveness in identifying research targets, they often struggle to find an appropriate balance between speed, accuracy, and the number of parameters. In view of this, this study takes YOLOv10n as the model foundation and constructs the YOLOv10n-FBD model by introducing the CCFM feature fusion technology, the BiFormer attention mechanism, and C2f-Dual lightweight strategy, thus achieving strong performance in terms of speed, accuracy, and the number of parameters under our evaluated settings. The specific work is as follows: (1)The CCFM (Cross-Scale Feature Fusion Module) feature fusion technology is introduced. This technology provides the model with more comprehensive and rich feature information on tomato fruits by fusing features of different scales and layers. Such feature fusion not only enhances the model's ability to understand fruit features but also lays a solid foundation for subsequent feature screening and optimization. (2)Integration of the BiFormer attention mechanism. This mechanism can further screen and highlight important features, enabling the model to focus more on the key feature regions critical for fruit recognition within the comprehensive feature information already acquired. This improvement significantly enhances the model's recognition accuracy, providing robust support for the precise detection of tomato fruit maturity. (3) The C2f-Dual lightweight strategy is adopted to optimize the model. This strategy effectively reduces model complexity while maintaining favorable performance, thereby significantly enhancing the model's generalization ability. Such lightweight processing not only enables the model to operate on resource-constrained devices but also avoids the degradation of generalization ability caused by data overfitting.

Results

To address the challenges of low accuracy, slow recognition speed, and large model size in cluster tomato fruit maturity recognition, this study constructs the YOLOv10n-FBD model based on the traditional YOLOv10n model. Validated through ablation studies and comparative experiments, the improved model maintains the same mAP@.5 as the traditional YOLOv10n while achieving a 2.2% performance gain in the higher-standard mAP@.5:.95. It also reduces model size, parameter count, and computational complexity (FLOPs) by 38.6%, 37.5%, and 28.6%, respectively, significantly optimizing resource consumption. Compared with mainstream object detection models such as Faster R-CNN, SSD, and YOLO series (v5, v7, v8, v11, and original v10n), the improved model demonstrates significant improvements in mAP@.5:.95, reaching up to 15%. With a compact model size of only 3.5MB and a low computational cost of 6.0 GFLOPs, it achieves a processing speed of 369.0 FPS. This achievement not only substantially improves the processing efficiency of global information and multi-scale objects but also effectively alleviates the computational pressure during large-scale dataset processing and reduces storage requirements. The study successfully constructs an accurate and efficient cherry tomato recognition network that perfectly balances precision and efficiency, making it suitable for complex and variable real-world application environments.

Ablation experiments to improve the model

To demonstrate the performance improvements of the proposed algorithm on the YOLOv10n model, we conducted an ablation study. This study validates the effectiveness of each module by progressively adding different improvement components to the model using the same dataset and training protocol. The specific model configurations used in the experiments are as follows: Traditional Model: The baseline YOLOv10n algorithm without any improvements. Method1: Starting with the traditional model, we add the lightweight Cross-Scale Feature Fusion Module (CCFM), forming the YOLOv10n-CCFM model. Method2: Starting with the traditional model, we optimize the PSA mechanism by incorporating the BiFormer attention mechanism, creating the YOLOv10n-BiFormer model. Method3 Starting with the traditional model, we add the lightweight C2f module improved with DualConv, forming the YOLOv10n-C2f-DualConv model. Building on these three basic methods, we further examined their combinations in an exploratory manner to assess potential interactions. The results of the ablation study are presented in Table 1. In the ablation study, it was observed that the combination of all three modules (Method 1+2+3) achieved an mAP@.5 of 97.9%, which is slightly lower than the result of Method 1+2 (98.4%). This indicates that the introduction of C2f-Dual may lead to a minor decline in precision at the looser IoU threshold of 0.5. A possible explanation is that C2f-Dual emphasizes lightweight design by reducing parameters and channel dimensions, which may limit the feature richness required by the PSA-BiFormer attention mechanism. As a result, the ability of the model to fully

leverage attention-based feature enhancement might be compromised, leading to a slight decrease in mAP@.5. Nevertheless, it is worth noting that the stricter metric mAP@.5:.95 remains unchanged (94.3%), which suggests that the combined model achieves better robustness and localization accuracy under rigorous evaluation criteria. Future work will further explore the compatibility among modules, especially the potential trade-offs between PSA-BiFormer and C2f-Dual, to achieve an optimal balance between efficiency and accuracy. Analysis of Table 1 reveals that when the CCFM module is used alone, although the mAP@.5 slightly decreases, the mAP@.5:.95 significantly improves (from 0.921 to 0.931). Meanwhile, the model size is reduced by nearly 33% (from 5.7 to 3.8 MB), and the computational complexity is decreased by approximately 23% (FLOPs from 8.4 G to 6.5 G), achieving remarkable lightweight optimization. In contrast, adopting BiFormer to optimize the PSA mechanism significantly enhances the model's performance across a broader range of IoU thresholds while maintaining low computational complexity. The DualConv-improved C2f module demonstrates improvements in all performance metrics compared to the traditional YOLOv10n model, while also slightly reducing the model size and parameter count. Furthermore, when these improvement modules are combined, the combination of CCFM and BiFormer-optimized PSA mechanism produces a significant synergistic effect, greatly boosting the model's performance. The model size and computational complexity are also superior to those achieved by any individual module alone. When all improvement modules are applied simultaneously, the YOLOv10n-FBD model's mAP@.5:.95 increases by 2.2%, while the model size, parameter count, and FLOPs decrease by 38.6%, 37.5%. Figure 1 provides a detailed comparison of performance metrics for various improved YOLOv10n algorithms during 200 training epochs. Observations show that compared to the original YOLOv10n model without any modifications, integrating only the lightweight cross-scale feature fusion module CCFM (Cross-Scale Feature Fusion Module) enables the model to exhibit significantly higher precision than the original model in most stages of training. However, when separately adding the BiFormer module or C2f-Dual module, although both achieve certain improvements in the mAP@.5:.95 performance metric, the training stability of the model declines to varying degrees. Notably, when applying these three methods in pairwise combinations to the model, the synergistic effects are particularly pronounced. Specifically, the combination of the CCFM module and BiFormer module not only achieves substantial improvements in precision, recall, and mAP but also demonstrates extremely high training stability. Furthermore, when integrating the CCFM module, BiFormer module, and C2f-Dual module, although the precision is slightly inferior to that of the original model, this combination optimizes training performance under more rigorous mAP evaluation criteria while ensuring training stability and achieving the optimal training effect.

Improved model recognition performance

We employed the YOLOv10n-FBD model to classify the maturity of cherry tomato fruits, dividing maturity into three categories: ripe (R), unripe (G), and semi-ripe (RG). After 200 training epochs, the improved YOLOv10n model demonstrated exceptional performance. On the validation set, the model achieved a precision of 93.1%, a recall of 96.6%, an average precision at an IoU threshold of 0.5 (mAP@.5) of 97.9%, and an average precision across the broader IoU threshold range of 0.5 to 0.95 (mAP@.5:.95) of 94.3%. Additionally, while maintaining high performance, the model achieves efficient computational efficiency, with a parameter count of only 1,693,497, an FLOPs value of 6.0 G, an FPS of 369.0, and a compact model size occupying only 3.5MB of storage space. The evaluation metrics for the three maturity categories are detailed in Table 2. As presented in Figure 2 (a normalized confusion matrix, where normalization is performed by the true class), the overall separability of the classes is favorable, yet the primary classification errors occur between the semi-ripe (RG) and ripe (R) classes—approximately 7% of the true R samples are incorrectly predicted as RG, while roughly 4% of the true RG samples are misclassified as R; by contrast, the unripe (G) class is the most stable, with a correct classification rate of approximately 96% (corresponding to the diagonal element of the confusion matrix). For background images, false detections are more frequently labeled as R (accounting for approximately 50%) and RG (around 41%) than as G (only about 9%), a phenomenon that inflates the false positive rates for the R and RG classes and helps explain the relatively lower precision (89.2%) and F1-score (92.1%) of the RG class reported in Table 2. The potential causes of these errors include fruit occlusion, small or distant fruit targets, challenging illumination conditions (such as direct light, glare, and color casts), and ambiguity in borderline labeling between the RG and R classes, and corresponding mitigation strategies involve enriching the dataset with boundary cases and hard-to-classify samples while moderately upsampling the RG class to balance class distribution, applying data augmentation techniques (including HSV color space jitter, random occlusion/shadow addition, and exposure variation) to enhance the model's robustness to diverse scenarios, tuning class-specific confidence thresholds for the R and RG classes to maximize the per-class F1-score on the validation set, increasing the input image resolution or strengthening the extraction of mid-level and high-level features where computational resources permit, and adopting class weighting or focal loss combined with label smoothing to alleviate the negative impact of boundary-label noise on classification performance. This study implemented the classification and recognition of cherry tomato ripeness. As shown in Figure 3, the model displays confidence scores as numbers and uses differently colored bounding boxes to distinguish ripeness categories, demonstrating excellent recognition capability across all ripeness stages. To validate the effectiveness of the YOLOv10n-FBD model in real-world cherry tomato detection scenarios, we conducted video-based validation experiments using greenhouse-captured videos (untrained on the model) as

image input sources to simulate practical conditions. The test videos were recorded with a Huawei P40 smartphone camera (50 MP, f/1.9 aperture) under natural daylight in the same orchard, with the recording distance maintained at approximately 2–4 meters and the shooting angle ranging from 30° to 60° relative to the horizontal plane—parameters chosen to mimic realistic fruit-harvesting scenarios. These videos had a frame rate of 30 frames per second (FPS) and a resolution of 2160×3840 pixels, and differed from the training dataset (described in Section 2.1, which primarily comprises high-resolution still images collected under controlled angles and distances) by introducing additional challenges, including camera motion, temporal continuity, and slight variations in lighting and viewpoint; such differences enabled a more comprehensive assessment of the model’s robustness in dynamic, practical application environments. The dynamic detection results of cherry tomatoes are presented in Figure 4: as observed from the figure, for the same frame, the YOLOv10n-FBD model exhibits superior detection performance, detecting a greater number of cherry tomatoes with higher confidence scores. Furthermore, the model achieves a frame rate of 369.0 FPS, fully meeting the real-time detection requirements for cherry tomatoes.

Comparison with mainstream models

Under consistent experimental conditions, we conducted a comprehensive performance comparison between the YOLOv10n-FBD model and several mainstream object detection algorithms, including the two-stage detection algorithm Faster R-CNN¹⁴, the single-stage detection algorithm SSD¹⁵, and other models from the YOLO series. The comparison results are presented in Table 3. Analysis of Table 3 demonstrates that compared with Faster R-CNN, the paradigm of two-stage detection algorithms, although the YOLOv10n-FBD model shows a slight decrease in recall, it achieves a significant 28.84% increase in precision. Meanwhile, the model improves mAP@.5, mAP@.5:.95, and F1-score by 1.46%, 15%, and 16.72%, respectively. Notably, the YOLOv10n model substantially outperforms Faster R-CNN in two computational performance metrics (i.e., computational efficiency and resource consumption) and model size. Compared with the SSD algorithm, the YOLOv10n-FBD model also demonstrates comprehensive performance advantages. Specifically, the model achieves improvements of 6.13%, 5.63%, 2.29%, 11%, and 6.39% in precision, recall, mAP@.5, mAP@.5:.95, and F1-score, respectively. Additionally, the YOLOv10n model substantially outperforms the SSD model in terms of computational performance and model size. In comparison with other YOLO-series models, although the YOLOv10n-FBD model shows moderate performance in precision, it significantly outperforms other YOLO models in three key metrics: recall, mAP@.5, and mAP@.5:.95. More importantly, the model achieves substantial reductions in parameter count and FLOPs, ranging from 37.89% to 84.99% and 28.57% to 94.18%, respectively. Furthermore, with a model size of only 3.5MB, it demonstrates a reduction of 38.95% to 95.29% compared to other YOLO models, highlighting its exceptional model compression efficiency and lightweight design advantages. In addition, the proposed YOLOv10n-FBD model achieves 369 FPS, corresponding to an average inference latency of approximately 2.7 ms per frame when evaluated on an NVIDIA RTX 3090 GPU, thereby underscoring its strong suitability for real-time deployment. Although the YOLOv10n-FBD model shows a slightly lower F1 score (94.72%) compared to the original YOLOv10n (95.11%), it achieves a 2.2% improvement in the stricter mAP@.5:.95 metric. This discrepancy mainly arises from the different emphases of the evaluation metrics: the F1 score is affected by threshold settings and class imbalance, focusing more on classification stability, while mAP@.5:.95 comprehensively reflects localization accuracy and robustness across multiple IoU thresholds. Therefore, the improved model demonstrates stronger advantages in localization precision, even if slight fluctuations occur in the classification performance of certain categories.

Methods

Dataset collection and processing

The Cherry Tomato Image Dataset was collected from the Shanxi Agricultural Valley Tomato Town, located in Gezitou Village, Fancun Town, Taigu District, Jinzhong City, Shanxi Province, China (geographical coordinates: 112.78074°E, 37.49854°N). Data collection was conducted on July 23–24, 2022, and February 28, 2024. To thoroughly analyze the growth characteristics of cherry tomatoes and enhance the robustness of machine learning models, this study employed a multi-device collaborative strategy, capturing images across multiple time periods, angles, and orientations. This comprehensive data acquisition approach was performed under diverse and complex environmental conditions. The image resolutions and the number of images collected by each device are summarized in Table 4. Using the aforementioned smartphone models, data were collected under two weather conditions (sunny and cloudy) across multiple time periods, including morning, noon, afternoon, and dusk, with both front and back lighting. A multi-angle, multi-orientation systematic imaging approach was employed for cherry tomato acquisition. The specific imaging protocol is designed as follows: (1) The baseline reference plane was defined as the plane parallel to the tomato row and perpendicular to the ground. (2) For each cherry tomato, five images were captured at angles of 10°, 45°, 90°, 135°, and 170°, as illustrated in Figure 5(a). (3) Additionally, three images were captured for each cherry tomato: one from a 45° downward view, one from a 45° front-rotated view, and one from a 45° back-rotated view, all relative to the reference plane, as shown in Figure 5(b). Through this shooting scheme, a complete dataset consisting of 8 images was captured for each cherry tomato. After removing blurred and highly similar images (similarity > 95%), a total of 4,046

images were retained. The statistical summary of the dataset is presented in Table 5. The open-source software LabelImg was employed to annotate cherry tomato fruits in images, with annotation information stored in XML document format. The annotation scheme is detailed as follows: (1) According to fruit color, cherry tomatoes are classified into three categories: Fruits that are fully red or contain 1–2 yellowish (non-green) fruits are annotated as R (ripe). Fruits with no red components are annotated as G (unripe). All other cases are annotated as RG (semi-ripe). (2) If cherry tomatoes are occluded by vines or leaves, they are not annotated. (3) Only cherry tomatoes and their pedicels on the row closest to the camera in the image are annotated, while more distant or overly small cherry tomatoes are ignored. The annotated dataset of 4,046 images was randomly divided into training, validation, and test sets at a ratio of 8:1:1, containing 3,236, 405, and 405 images, respectively. Although the annotation strategy adopted in Section 2.1 excluded occluded, distant, and very small fruits, which may lead to a dataset that does not fully reflect the complexity of real-world orchard environments, this choice was made to ensure annotation consistency and to reduce labeling ambiguity. As a result, the current dataset emphasizes clear fruit instances, facilitating stable model training and reliable baseline comparisons. Nevertheless, we acknowledge that this strategy might introduce a generalization gap when applying the model to complex scenarios with frequent occlusion and scale variation. To address this limitation, future work will consider three directions: (1) relaxing the annotation rules to include partially occluded and small fruits in order to improve robustness; (2) augmenting the dataset with synthetic samples or domain adaptation techniques to better simulate real orchard environments; and (3) validating the model across multiple orchards and diverse growing conditions to comprehensively assess its generalization ability.

Experimental environment and parameter settings

The training and testing of the tomato ripeness dataset were conducted on the Ubuntu 22.04 operating system with the following hardware and software configurations: hardware included an Intel(R) Core(TM) i7-13700K (13th Gen) CPU and an NVIDIA GeForce RTX 3090 GPU; software specifications comprised CUDA 11.1, Python 3.8.18, and the PyTorch 1.9.0 deep learning framework. Training parameters were set as follows: 200 epochs (to prevent overfitting), a batch size of 16, and an input image size of 640×640. The optimizer used was Stochastic Gradient Descent (SGD) with a momentum of 0.937, weight decay of 0.0005, and an initial learning rate of 0.01. A cosine annealing learning rate schedule was adopted to gradually reduce the learning rate during training, which helped stabilize convergence and improve performance. For data augmentation, a combination of techniques was employed to enhance dataset diversity and model generalization, including random horizontal flipping ($p=0.5$), Mosaic augmentation, MixUp augmentation, random scaling (0.5–1.5 range), color jittering (hue ± 0.015 , saturation ± 0.7 , value ± 0.4), random cropping with padding, and random affine transformations (rotation $\pm 10^\circ$, translation $\pm 10\%$, shear $\pm 2^\circ$) to simulate variations in object appearance and spatial layout. This configuration ensured stable training while balancing accuracy and lightweight deployment efficiency.

YOLOv10n network model

In the field of object detection, the YOLO (You Only Look Once) series of algorithms is widely recognized for its efficient real-time performance and high detection accuracy¹⁶. As a one-stage object detection algorithm, YOLO predicts all object categories and their bounding boxes in an image simultaneously through a single forward pass. This study adopts YOLOv10n¹⁷ as the training model. The model reduces inference latency by introducing a consistent bipartite assignment method for non-maximum suppression (NMS)-free training while maintaining competitive detection performance. Additionally, under the design philosophy of holistic efficiency-and-accuracy-driven optimization, YOLOv10n comprehensively optimizes each component, thereby achieving a more efficient and accurate real-time end-to-end object detection solution. Conv represents ordinary convolution operation; C2f is a feature fusion module for aggregating and refining features; SCDown denotes spatial-channel downsampling, reducing spatial dimensions and processing channels; C2fCIB is a C2f module integrated with Channel Interaction Block to enhance channel-wise information exchange; SPPF stands for Spatial Pyramid Pooling - Fast, aggregating multi-scale features efficiently; PSA represents Position-Sensitive Attention, focusing on position-related feature enhancement; Upsample refers to upsampling, increasing spatial resolution; Concat denotes concatenation operation, fusing feature maps along the channel dimension; One-to-many Head and One-to-one Head are detection heads for multi-target and single-target detection tasks respectively. The network architecture of YOLOv10n is shown in Figure 6, composed of three major components: Backbone, Neck, and Head, designed to enable efficient object detection. In the Backbone, besides the C2f module and SPPF (Spatial Pyramid Pooling Fast) module inherited from YOLOv8, a lightweight classification head, Spatial-Channel Separate Downsampling (SCDown), and rank-based block design (C2fCIB) are proposed to boost efficiency. For accuracy enhancement, large-kernel convolutions are explored, and an effective Partial Self-Attention (PSA) module is introduced to strengthen model capabilities, achieving performance improvements at low cost. The Neck network incorporates a Path Aggregation Network (PANet), which effectively fuses feature maps across different scales, enhancing the model's detection capability for multi-scale objects. The Head component designs a consistent bipartite assignment strategy, integrating the advantages of one-to-many and one-to-one assignments, which are respectively applied during training and inference. During the training phase, the one-to-many assignment strategy is employed to assign multiple positive samples to each

ground-truth bounding box, providing abundant supervisory signals to help the model learn better. Meanwhile, the one-to-one assignment strategy allocates one positive sample per ground-truth bounding box, ensuring the model learns to select the optimal predicted bounding box during training. Whereas during the inference phase, the one-to-one head generates a single optimal prediction for each object, eliminating the need for non-maximum suppression (NMS), effectively reducing inference latency, and enhancing real-time performance. For the loss function, the YOLOv10n model adopts CIOU Loss (Complete Intersection over Union Loss), which comprehensively considers the overlap area, center distance, and aspect ratio between the predicted and ground-truth boxes, thereby further improving the model's localization accuracy and robustness. Overall, YOLOv10n achieves optimization of efficiency and accuracy in structural design and algorithmic implementation, meeting the requirements of complex scenarios in practical applications.

Improvements to the YOLOv10n model

CCFM feature fusion

In the traditional YOLOv10n model, the detection of small-scale objects often suffers from information loss due to the limited number of pixels and repeated convolution and pooling operations, which ultimately reduces detection accuracy. To address this issue, this study integrates the Cross-Scale Feature Fusion Module (CCFM) from RT-DETR into YOLOv10n, replacing its original feature fusion component. By fusing features across multiple scales, the CCFM module enhances the adaptability of the model to scale variations and improves the detection performance for small objects. This design effectively combines fine-grained local details with global contextual information, thereby improving the overall detection capability of the model¹⁸. As shown in Figure 7(a), the CCFM module adopts a structure similar to PANet (Path Aggregation Network) but incorporates Fusion Blocks along the fusion paths to merge features from adjacent scales. Each Fusion Block is composed of convolutional layers and additional operations designed to refine and integrate features. The detailed structure of the Fusion Block is illustrated in Figure 7(b). RepBlock represents a re-parameterization block. It combines multiple computational branches during training and simplifies the structure for efficient inference while maintaining performance. The input to the CCFM module typically consists of multi-scale feature maps from the backbone network, denoted as S3, S4, and S5, which represent image information at different resolutions. The Fusion Block serves as the core component of the CCFM module, responsible for fusing these multi-scale features. Its computational process can be abstractly expressed by equations (1)–(3):

$$Q = K = V = \text{Flatten}(S_5) \quad (1)$$

$$F_5 = \text{Reshape}(\text{Attn}(Q, K, V)) \quad (2)$$

$$\text{Output} = \text{CCFM}(\{S_3, S_4, F_5\}) \quad (3)$$

Here, "Q", "K", and "V" correspond to the query, key, and value matrices within the attention mechanism (Attn), which are the fundamental components for computing attention-based feature weighting. "Flatten" denotes the operation that reshapes the feature map "S5", derived from the backbone, into a one-dimensional sequence. After the attention operation is applied, "Reshape" restores the feature representation to match the original spatial structure of "S5". "S3" and "S4" represent feature maps at other scales extracted from the backbone network, each carrying complementary information. "F5" denotes the feature map obtained by applying the attention operation and shape restoration to "S5". Finally, the output of the CCFM module is obtained by fusing S3, S4, F5 through specific operations such as addition or concatenation, thus achieving effective multi-scale feature integration for subsequent tasks like object detection.

PSA-BiFormer attention mechanism

The PSA (Position Sensitive Attention) mechanism in the YOLOv10n model faces efficiency challenges when processing global information and multi-scale objects. Moreover, the global attention mechanism's high computational complexity and memory demands constrain the model's real-time performance and deployment flexibility. To address these issues, this study introduces the BiFormer attention mechanism in the YOLOv10n model to replace the original PSA module. The BiFormer is a vision Transformer model integrating Bi-level Routing Attention, and inherits the core advantages of Transformer architectures while implementing unique structural optimizations¹⁹. Compared with convolutional neural networks (ConvNets) that primarily rely on local operations, Transformer models can more effectively capture long-range dependencies in images through their attention mechanisms with global receptive fields. However, such global attention mechanisms also introduce issues of high computational complexity and large memory requirements, thereby limiting their efficiency in practical applications. To address this challenge, this study borrows the region-to-region routing approach, aiming to globally locate and focus on valuable

key-value pairs in a more efficient manner. The applicability of BiFormer in agricultural vision has also been verified. Li et al.²⁰ integrated BiFormer attention into the YOLO-Leaf model for apple leaf disease detection and reported enhanced feature extraction performance under complex backgrounds, which further supports its integration into our improved YOLOv10n model. Specifically, a regional-level affinity graph is first constructed and pruned to retain only the top-k connections for each node, thereby achieving regional-level filtering. This step ensures that each region only focuses on its top-k most relevant regions. Subsequently, a token-to-token attention mechanism is introduced. Aiming at the possible spatial dispersibility of filtered key-value pairs, a hardware-friendly solution is proposed: collecting key/value tokens and processing them using dense matrix multiplication to avoid sparse matrix multiplication, which is less efficient on modern GPUs²¹. This approach, termed Bi-level Routing Attention (BRA), combines region routing with token attention, optimizing object detection performance. Model architecture diagram based on BiFormer. Input denotes the initial input with dimensions $H \times W \times 3$ (height $\times$ width $\times$ 3 RGB channels); Stage represents sequential feature extraction phases with varying spatial resolutions and channel capacities; Patch Embedding is a module that segments the input into patches and projects them to a high - dimensional space for initial feature encoding; BiFormer Block is the core component integrating bidirectional self - attention for advanced feature interaction; MLP stands for Multi - Layer Perceptron, used for feature transformation; LN refers to Layer Normalization, which normalizes features within a layer to stabilize training; Bilevel Routing Attention is a hierarchical attention mechanism that adaptively focuses on spatial regions at two levels; DWConv 3×3 denotes depthwise convolution with a 3×3 kernel, capturing local spatial patterns per channel; (N1, N2, N3, N4) indicate the number of repetitions of the BiFormer Block in each stage; (C, 2C, 4C, 8C) represent channel dimensions at different stages, reflecting progressive feature capacity expansion. The attention mechanism architecture of BiFormer, as shown in Figure 8, is constructed by stacking multiple BiFormer blocks and can be flexibly configured according to specific tasks and requirements. The core innovation of this architecture lies in the introduction of a bi-level routing attention mechanism, which achieves content-aware sparsity within each BiFormer block, thereby significantly enhancing computational efficiency and task performance. The structure of the BiFormer block, serving as the fundamental building unit of BiFormer, is detailed on the right side of Figure 8. Comprising multiple key sublayers, this block includes a self-attention sublayer and a feed-forward neural network sublayer. Specifically, the self-attention sublayer ingeniously implements the bi-level routing attention mechanism, which enables the system to adaptively focus on the most relevant key-value pairs based on query content, thus achieving efficient information filtering and integration. The feed-forward neural network sublayer, conversely, performs nonlinear transformations and feature extraction on the self-attention sublayer's output via a multilayer perceptron (MLP), further enhancing the model's representational capability and adaptability. The overall architecture and BiFormer blocks, designed in this manner, not only effectively leverage the bi-level routing attention mechanism to achieve content-aware sparsity but also offer high flexibility and robust representational power. In attention computation, Q (Query) encodes the location/features to retrieve relevant information. Kg (Key) serves as both the aggregation key and sampling key, interacting with the Query to facilitate information filtering. Vg (Value) denotes the collected information, corresponding to the Key to generate the final output. A (Attention Map) encodes the interaction weights between Queries and Keys, while O (Output) represents the refined feature map after attention integration. The efficient computation process of the BiFormer attention mechanism, as shown in Figure 9, aims to optimize computational efficiency and resource utilization. Its core lies in the Bi-level Routing Attention mechanism, which precisely focuses on the top-k window key-value pairs most relevant to the query, significantly reducing computational overhead. Compared to the global processing of traditional attention mechanisms, BiFormer avoids high computational complexity and resource consumption through a two-level routing strategy. First, the first-layer routing mechanism preliminarily filters key-value pairs to exclude irrelevant regions. Then, the second-layer routing mechanism refines the selection, ensuring that only the top-k most relevant window key-value pairs participate in subsequent computations. This approach not only leverages data sparsity to avoid redundant calculations, thereby improving computational efficiency but also ensures that only relevant key-value pairs undergo GPU-intensive matrix multiplication, reducing both computation and memory consumption to enhance performance. Additionally, the bi-level routing attention mechanism endows the model with high adaptability, enabling it to intelligently adjust its focus, accurately capture semantic correlations in input data, enhance representational power, and improve the flexibility and robustness of task processing. In summary, the BiFormer attention mechanism achieves dual optimization of computational efficiency and resource utilization through a bi-level routing strategy, strongly supporting large-scale data processing.

Lightweight realization via C2f-Dual

In the YOLOv10n model, the C2f module, as a component of the backbone network, integrates low-level and high-level feature maps to facilitate gradient information flow. However, the standard convolutional layers within the C2f structure can impose a high computational burden when processing large-scale datasets, potentially impacting the model's real-time performance²². Additionally, the large number of model parameters not only increases storage requirements but may also lead to overfitting during training. To address these challenges, this paper introduces a lightweight Dual-C2f (Dual Convolution C2f-Dual Design) improved by DualConv. DualConv is an innovative convolutional network structure designed to construct lightweight deep neural networks, with its specific architecture illustrated in Figure 10. The core of DualConv lies in the intelligent combination

of 3×3 and 1×1 convolutional kernels, integrated with group convolution technology and fusion strategies²³. Specifically, the DualConv design splits the convolution operation into two parallel sublayers: one sublayer employs a 3×3 convolutional kernel to primarily capture spatial features of the feature maps, extracting image details through a local receptive field; the other sublayer uses a 1×1 convolutional kernel to integrate features, enabling inter-channel interactions and information fusion while reducing parameters and computational complexity. Additionally, this network structure fully leverages group convolution technology, processing input, and output feature maps in groups to further reduce model complexity and optimize information flow and feature extraction efficiency. Finally, a fusion strategy combines the features extracted by the two sublayers to form the final feature representation, which maintains the model's diversity and expressive power while decreasing computational complexity and model size.

Constructing the YOLOv10n-FBD model

The YOLOv10n-FBD model is developed to address the challenges faced by the traditional YOLOv10n model in cherry tomato maturity detection. Key improvements are introduced in three major areas: feature fusion, attention mechanism optimization, and computational complexity reduction. These optimizations significantly enhance the model's detection accuracy in complex environments, while ensuring real-time detection performance. The first major enhancement is the introduction of CCFM (Cross-Scale Feature Fusion Module), which improves the feature fusion process. This module is based on a PANet-like bidirectional feature flow architecture, integrating Fusion Blocks into the multi-scale feature map fusion paths. Each Fusion Block consists of 3×3 convolution and RepBlock components, which enhance feature representations through spatial-dimensional resampling and channel-dimensional attention modulation. The module effectively handles situations such as detecting semi-ripe cherry tomatoes that are partially occluded by branches and leaves. The CCFM module enhances detection by fusing high-resolution edge features with global color semantics from higher layers, addressing the challenge of small object detection. By flattening high-level feature maps, applying multi-head self-attention to recalibrate channel weights, and fusing features across scales, this module significantly improves the model's multi-scale representation capabilities. The second improvement integrates the PSA-BiFormer attention mechanism, which replaces the original PSA mechanism to enhance global feature processing. The BiFormer utilizes a two-layer routing strategy: the first layer filters key regions using a region-level affinity map, and the second layer accelerates computations through dense matrix multiplication. This approach reduces unnecessary computations while focusing on relevant regions of interest, such as red fruits, and suppresses interference from less important areas like green leaves or stems. This attention mechanism ensures better detection, even under challenging lighting conditions, such as backlighting, by improving the semantic distinction between fruits and their background. The third significant optimization involves the C2f-Dual hybrid convolution strategy, which restructures the traditional C2f module in the backbone network. This strategy divides the standard convolution operation into two parallel branches: a 3×3 spatial convolution branch for capturing the contour and texture features of cherry tomatoes, and a 1×1 channel convolution branch for integrating color semantic information. The use of group convolution technology further reduces the parameter count, while residual connections maintain gradient flow integrity, leading to a more lightweight yet effective model. The network architecture of YOLOv10n is shown in Figure 11, demonstrates the integration of the key components mentioned above. The convolution layers (Conv) are initially applied to process the input image and extract basic features. The CCFM block is introduced to fuse features across multiple scales, followed by the C2f-Dual module to enhance feature processing while reducing the model size. SCDwn and upsample operations are used to manage the resolution of feature maps as they pass through different network layers. The final output layers include One-to-many and One-to-one heads, which facilitate precise detection and classification of the cherry tomato maturity stages (ripe, semi-ripe, unripe) by generating bounding boxes and confidence scores. This structure allows the model to maintain a balance between high performance and lightweight design, making it suitable for real-time applications in resource-constrained environments.

Experimental evaluation metrics

This study employs seven metrics to evaluate the experimental results: Precision(P), Recall(R), F1 score, mAP@.5, mAP@.5:.95, parameter count, and model memory footprint. Precision(P) is used to measure the accuracy of the classifier in extracting positive examples, defined as the proportion of true positives among predicted positive examples. Its mathematical expression is shown in Equation (4), where TP (True Positive) denotes the number of true positives, meaning the number of samples that are actually positive and correctly predicted as positive. FP (False Positive) denotes the number of false positives, meaning the number of samples that are actually negative but incorrectly predicted as positive.

$$\text{Precision}(P) = \frac{TP}{TP + FP} \quad (4)$$

Recall(R) is used to assess the classifier's ability to predict positive examples, defined as the proportion of actual positive examples that are correctly predicted. Its calculation formula is shown in Equation (5), where FN (False Negative) denotes the

number of false negatives, meaning the number of samples that are actually positive but incorrectly predicted as negative.

$$\text{Recall}(R) = \frac{TP}{TP + FN} \quad (5)$$

Given the trade-off between Precision and Recall, the F1 score integrates their performance via a harmonic mean to assess model stability. When Precision and Recall are equally weighted, the F1 score is defined by Equation (6). When the weight parameter $\beta \neq 1$, the calculation is given by Equation (7), where β adjusts the relative importance of Precision and Recall.

$$\text{F1 score} = \frac{2 \times P \times R}{P + R} = \frac{2TP}{2TP + FP + FN} \quad (6)$$

$$F_\beta = \frac{1}{1 + \beta^2} \left(\frac{1}{P} + \frac{\beta^2}{R} \right) = \frac{(1 + \beta^2)PR}{\beta^2 P + R} \quad (7)$$

The mean Average Precision (mAP) comprehensively evaluates the overall detection performance of a model by calculating the mean of the average precision for all categories, as shown in Formula (5). Here, k represents the total number of target categories, and AP_i denotes the average precision of the i -th category.

$$\text{mean Average Precision (mAP)} = \frac{\sum_{i=1}^k AP_i}{k} \quad (8)$$

The mAP@.5 refers to the mAP value when the Intersection over Union (IoU) threshold is 0.5, where a prediction is considered correct only if the IoU between the predicted bounding box and the annotated bounding box exceeds 0.5. mAP@.5:.95 calculates the mean of mAP values across 10 IoU thresholds (stepping by 0.05) within the interval [0.5, 0.95]. This metric covers detection requirements of varying strictness levels and a higher value indicates a higher positioning accuracy of the predicted bounding boxes. In addition to recognition performance metrics, two computational performance metrics are adopted to evaluate the model. One is the number of parameters, a crucial indicator of the model's spatial complexity, which denotes the total count of all trainable parameters in the model. These parameters are optimized during training to minimize the loss function. The other metric is FLOPs (floating-point operations per second), which refers to the number of billion floating-point operations per second. Fewer parameters and lower FLOPs indicate that the model can operate more efficiently under the same computational resources while reducing memory usage and enhancing computational speed. Model memory footprint refers to the total memory occupied by a model during inference or training. This encompasses the memory usage of the model's weights, biases, and input data, as well as any intermediate results and temporary variables that may be generated. A smaller memory footprint enables the model to run on resource-constrained devices without causing memory overflow or performance degradation. Additionally, it helps reduce operational costs and improves the overall system performance.

Limitations

This study exhibits notable limitations in dataset construction and scenario adaptability. For the purpose of ensuring annotation consistency, occluded cherry tomatoes as well as those that are distant or extremely small were excluded from the dataset, with only samples featuring clear features retained. This renders the dataset incapable of reflecting complex scenarios in real orchards, such as dense clustering of fruits, extensive occlusion, or tiny targets, potentially leading to an underestimation of the model's detection errors under such circumstances. Although the data collection covered multiple devices and time periods, it was confined to preset angles, specific natural lighting conditions, and the planting environment of Tomato Town in Shanxi Agricultural Valley. It did not include scenarios involving extreme lighting (e.g., glare under strong light, low light), different geographical regions, or various planting patterns, making it difficult to generalize the model to environments beyond the established data collection settings. Additionally, the ablation experiments revealed a compatibility trade-off between the lightweight design of the C2f-Dual module and the BiFormer attention mechanism. When all three modules (CCFM, BiFormer, and C2f-Dual) were integrated simultaneously, the mAP@.5 slightly decreased (from 98.4% of the CCFM + BiFormer combination to 97.9%), indicating that there is still room for improvement in the collaborative optimization among the modules.

Data Availability Statement

We confirm that we have obtained appropriate permissions for the collection of the Cherry Tomato Image Dataset. The dataset was captured at the Tomato Town facility in the Shanxi Agriculture Valley. We secured explicit permission from the management of this facility prior to our visit, which granted us authorization to enter the premises and conduct photography of the cherry tomato plants for our research purposes.

The data provided in this study are available upon request from the corresponding author.

References

1. Chu, X., Zheng, L., Ye, G. & Chen, J. Comparison of greenhouse gas emissions of open-field and facility-grown tomato production systems in china. *J. Agro-Environment Sci.* **42**, 1870–1881, DOI: [10.11654/jaes.2022-1252](https://doi.org/10.11654/jaes.2022-1252) (2023).
2. Gu, W., Wei, J., Yin, Y., Liu, X. & Ding, C. Multi-category segmentation method of tomato image based on improved deeplabv3+. *Transactions Chin. Soc. for Agric. Mach.* **54**, 261–271, DOI: [10.6041/j.issn.1000-1298.2023.12.025](https://doi.org/10.6041/j.issn.1000-1298.2023.12.025) (2023).
3. Dou, H. *et al.* Research progress on autonomous navigation technology for orchard intelligent equipment. *Transactions Chin. Soc. for Agric. Mach.* **55**, 1–22, DOI: [10.6041/j.issn.1000-1298.2024.04.001](https://doi.org/10.6041/j.issn.1000-1298.2024.04.001) (2024).
4. Mao, S. & Zhang, C. Promote the construction of an agricultural power through the development of new quality agricultural productivity. *Issues Agric. Econ.* 36–46, DOI: [10.13246/j.cnki.iae.2024.04.005](https://doi.org/10.13246/j.cnki.iae.2024.04.005) (2024).
5. Liu, J. & Jiang, Y. Industrialization trends and multi-arm technology direction of harvesting robots. *Transactions Chin. Soc. for Agric. Mach.* **55**, 1–17, DOI: [10.6041/j.issn.1000-1298.2024.10.001](https://doi.org/10.6041/j.issn.1000-1298.2024.10.001) (2024).
6. Miao, R., Li, Z. & Wu, J. Lightweight maturity detection of cherry tomato based on improved yolo v7. *Transactions Chin. Soc. for Agric. Mach.* **54**, 225–233, DOI: [10.6041/j.issn.1000-1298.2023.10.022](https://doi.org/10.6041/j.issn.1000-1298.2023.10.022) (2023).
7. Hou, G. *et al.* An occluded cherry tomato recognition model based on improved yolov7. *Front. Plant Sci.* **14**, DOI: [10.3389/fpls.2023.1260808](https://doi.org/10.3389/fpls.2023.1260808) (2023).
8. Yang, X., Liu, T., Han, J., Guo, X. & Yang, L. Low temperature storage tomato maturity recognition and time series prediction based on swin transformer-gru. *Transactions Chin. Soc. for Agric. Mach.* **55**, 213–220, DOI: [10.6041/j.issn.1000-1298.2024.03.021](https://doi.org/10.6041/j.issn.1000-1298.2024.03.021) (2024).
9. Chang, W., Tan, Y., Zhou, L. & Yang, Q. Tomato ripening detection in natural environment based on improved yolov5s. *Acta Agric. Univ. Jiangxiensis(Natural Sci. Ed.* **46**, 1025–1036, DOI: [10.3724/aauj.2024091](https://doi.org/10.3724/aauj.2024091) (2024).
10. Wang, A. *et al.* Nvw-yolov8s: an improved yolov8s network for real-time detection and segmentation of tomato fruits at different ripeness stages. *Comput. Electron. Agric.* **219**, DOI: [10.1016/j.compag.2024.108833](https://doi.org/10.1016/j.compag.2024.108833) (2024).
11. Huang, W. *et al.* Aitp-yolo: improved tomato ripeness detection model based on multiple strategies. *Front. Plant Sci.* **16**, DOI: [10.3389/fpls.2025.1596739](https://doi.org/10.3389/fpls.2025.1596739) (2025).
12. Zeng, T., Li, S., Song, Q., Zhong, F. & Wei, X. Lightweight tomato real-time detection method based on improved yolo and mobile deployment. *Comput. Electron. Agric.* **205**, DOI: [10.1016/j.compag.2023.107625](https://doi.org/10.1016/j.compag.2023.107625) (2023).
13. Zhu, X., Chen, F., Zheng, Y., Chen, C. & Peng, X. Detection of camellia oleifera fruit maturity in orchards based on modified lightweight yolo. *Comput. Electron. Agric.* **226**, DOI: [10.1016/j.compag.2024.109471](https://doi.org/10.1016/j.compag.2024.109471) (2024).
14. Wen, J., Wang, Y., Li, J. & Zhang, Y. Faster r-cnn object detection algorithm based on multi-head self-attention mechanism. *Mod. Electron. Tech.* **47**, 8–16, DOI: [10.16652/j.issn.1004-373x.2024.07.002](https://doi.org/10.16652/j.issn.1004-373x.2024.07.002) (2024).
15. Paymode, A. & Malode, V. Transfer learning for multi-crop leaf disease image classification using convolutional neural network vgg. *Artif. Intell. Agric.* **6**, 23–33, DOI: [10.1016/j.aiia.2021.12.002](https://doi.org/10.1016/j.aiia.2021.12.002) (2022).
16. Xu, Y., Li, J., Dong, Y. & Zhang, X. Survey of development of yolo object detection algorithms. *J. Front. Comput. Sci. & Technol.* **18**, 2221–2238, DOI: [10.3778/j.issn.1673-9418.2402044](https://doi.org/10.3778/j.issn.1673-9418.2402044) (2024).
17. Wang, Q., Yan, N., Qin, Y., Zhang, X. & Li, X. Bed-yolo: an enhanced yolov10n-based tomato leaf disease detection algorithm. *Sensors* **25**, DOI: [10.3390/s25092882](https://doi.org/10.3390/s25092882) (2025).
18. Xin, D. & Li, T. Revolutionizing tomato disease detection in complex environments. *Front. Plant Sci.* **15**, DOI: [10.3389/fpls.2024.1409544](https://doi.org/10.3389/fpls.2024.1409544) (2024).
19. Li, X., Zhang, T., Zhang, Z., Wei, H. & Qian, Y. Survey of transformer research in computer vision. *Comput. Eng. Appl.* **59**, 1–14, DOI: [10.3778/j.issn.1002-8331.2204-0207](https://doi.org/10.3778/j.issn.1002-8331.2204-0207) (2023).
20. Li, T., Zhang, L. & Lin, J. Precision agriculture with yolo-leaf: advanced methods for detecting apple leaf diseases. *Front. Plant Sci.* **15**, DOI: [10.3389/fpls.2024.1452502](https://doi.org/10.3389/fpls.2024.1452502) (2024).
21. Wang, C., Han, Q., Zhang, T., Li, C. & Sun, X. Litchi picking points localization in natural environment based on the litchi-yoso model and branch morphology reconstruction algorithm. *Comput. Electron. Agric.* **226**, DOI: [10.1016/j.compag.2024.109473](https://doi.org/10.1016/j.compag.2024.109473) (2024).
22. Dong, Y., Ceng, H. & Hou, S. Lmuav-yolov8: Lightweight network for object detection in low-altitude uav vision. *Comput. Eng. Appl.* **61**, 94–110, DOI: [10.3778/j.issn.1002-8331.2407-012](https://doi.org/10.3778/j.issn.1002-8331.2407-012) (2025).
23. Gang, S., Liu, P. & Guo, X. Lightweight improved yolov8n model for steel defect detection features. *Electron. Meas. Technol.* **48**, 74–82, DOI: [10.19651/j.cnki.emt.2417230](https://doi.org/10.19651/j.cnki.emt.2417230) (2025).

Acknowledgements

We're truly grateful to Shanxi Agricultural University for offering the experimental environment and research resources that were crucial for the seamless progress of this study. A special shout - out goes to our supervisor, Song Guozhu, for his invaluable suggestions and patient guidance throughout the entire research process. We also want to express our appreciation to all co - authors for their hard work and support in bringing this work to completion.

Author contributions statement

Conceptualization, J.C. and S.C.; methodology, J.C. and S.C.; software, J.C. and S.C.; validation, J.C. and S.C.; formal analysis, J.C. and S.C.; data curation, J.C., S.C. and B.C.; writing—original draft, J.C. and B.C.; writing—review and editing, S.C., B.C. and Y.H.; visualization, B.C. and Y.H.; supervision, K.W. and G.S.; project administration, K.W. and G.S.; funding acquisition, K.W. and G.S.. All authors have read and agreed to the published version of the manuscript.

Conflict of Interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Ethical Statement

This study did not involve animal experiments or human clinical trials, and thus did not require additional ethics committee approval. All procedures related to human participation in this study were conducted in accordance with the ethical principles outlined in the Declaration of Helsinki. Prior to their involvement, all participants were fully informed about the purpose, procedures, and potential implications of the study. Informed consent was obtained from each participant, ensuring they understood their rights and could withdraw from the study at any time without repercussions.

Funding

This work was supported by the Key Research and Development Project of the Shanxi Provincial Department of Science and Technology (202102140601015), the 2023 Scientific and Technological Innovation Project of Jinzhong National Agricultural High-tech Zone (Taigu National Science and Technology Innovation Center), and the Shanxi Provincial Basic Research Program (202403021222112).

Model	mAP@.5 (%)	mAP@.5:.95 (%)	Model Size (MB)	Parameters	FLOPs (G)
YOLOv10n	97.9	92.1	5.7	2708210	8.4
Method 1	97.8	93.1	3.8	1846969	6.5
Method 2	97.6	93.4	5.5	2712994	8.3
Method 3	97.3	92.8	5.1	2489634	7.6
Method 1 + 2	98.4	94.3	3.8	1864377	6.6
Method 1 + 3	97.2	93.7	3.5	1681033	6.1
Method 2 + 3	97.4	93.4	5.3	2507042	7.6
Method 1 + 2 + 3	97.9	94.3	3.5	1693497	6.0

Table 1. Ablation study of algorithms on the cherry tomato dataset.

Ripeness	P (%)	R (%)	mAP@.5 (%)	mAP@.5:.95 (%)	F1-score (%)
Ripe (R)	94	94.6	98	95.6	94.3
Unripe (G)	96	100	98.4	94.6	98
Semi-ripe (RG)	89.2	95.2	97.4	92.6	92.1

Table 2. Evaluation metrics for each ripeness category.

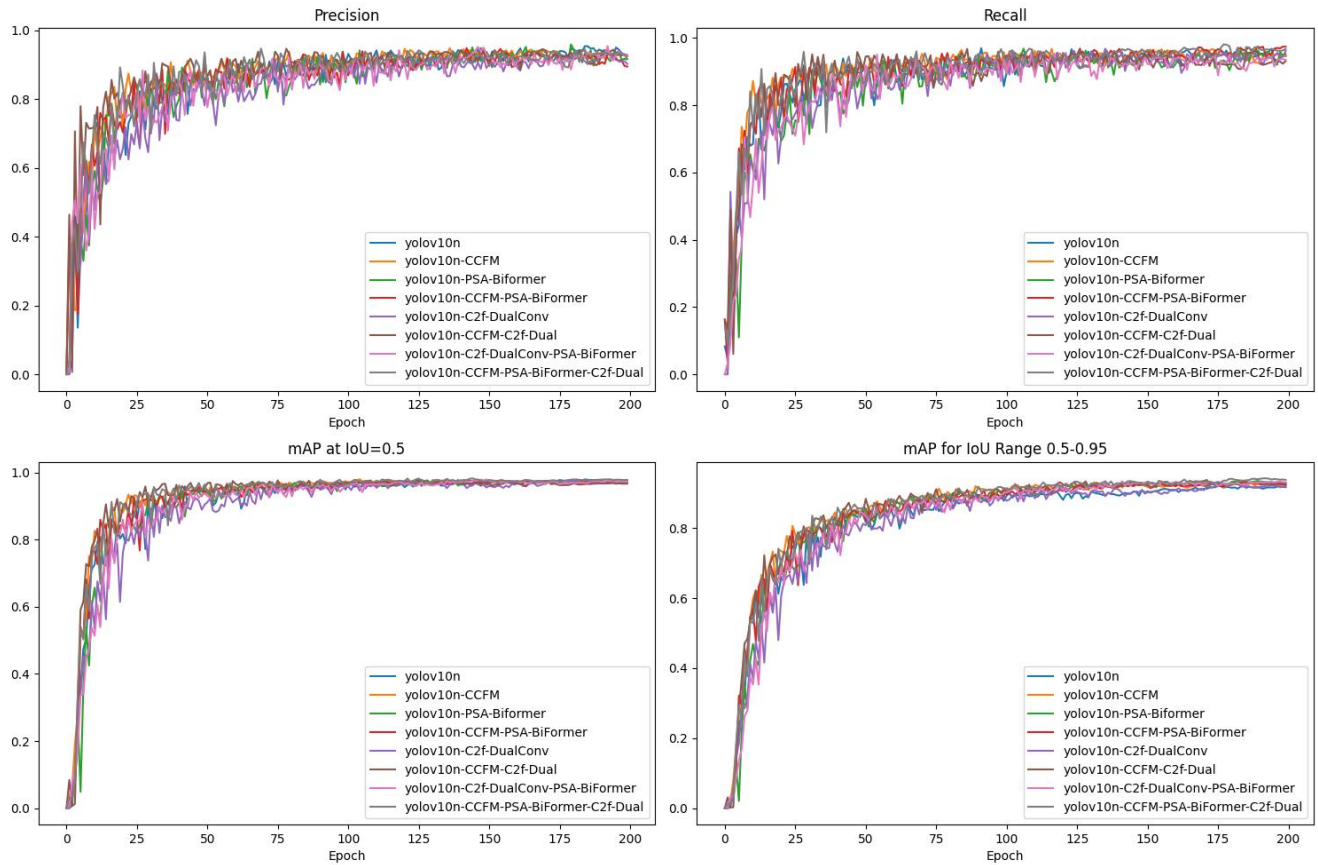


Figure 1. Performance results of different improved YOLOv10n algorithms during training.

Model	P (%)	R (%)	mAP@.5 (%)	mAP@.5:.95 (%)	F1 (%)	Parameters	FLOPs (G)	Model Size (MB)
Faster R-CNN	64.26	98.34	96.44	79.3	78	136729994	369.768	521.58
SSD	86.37	91.03	95.71	83.5	88.33	23879570	60.955	91.09
YOLOv5	94.8	95.8	97.5	91.3	95	9112697	23.8	18.5
YOLOv7	92.4	94.6	97.4	88.6	93	6020400	103.5	74.8
YOLOv8	93	93.1	97.6	90.4	93	11126745	28.4	21.4
YOLOv10	95.4	94.6	97.9	92.1	95.1	2708210	8.4	5.7
YOLOv11	90.5	95.0	97.3	91.9	92.7	2582737	6.3	5.5
YOLOv10n-FBD	93.1	96.6	97.9	94.3	94.72	1693497	6	3.5

Table 3. Evaluation Metrics of Detection Performance for Different Models.

Smartphone Model	Image Resolution	Number of Images
iPhone 13 Pro Max	3024×4032	1200
iPhone 14	3072×4096	1100
Huawei P40	2736×3648	960
vivo X60	3000×4000	1000

Table 4. Number of images captured and resolution specifications for each device.

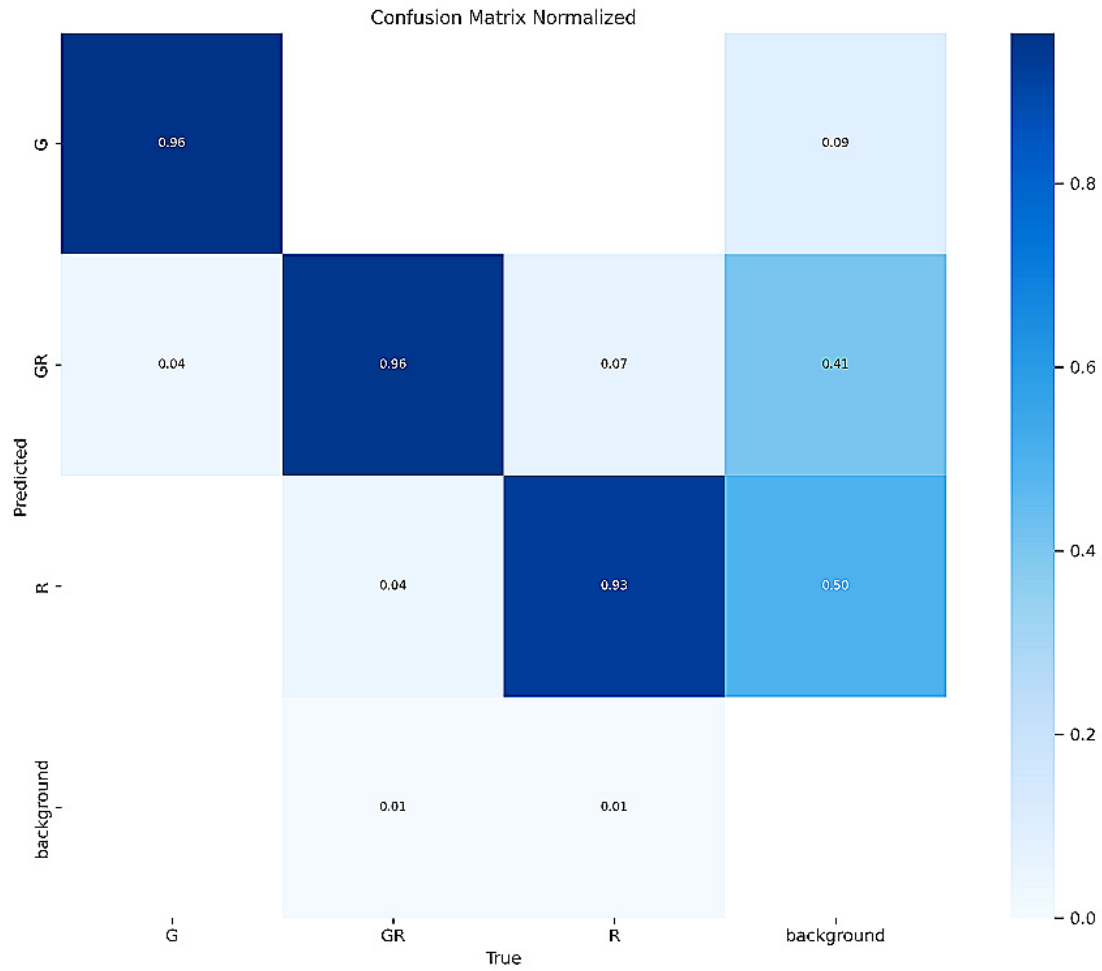


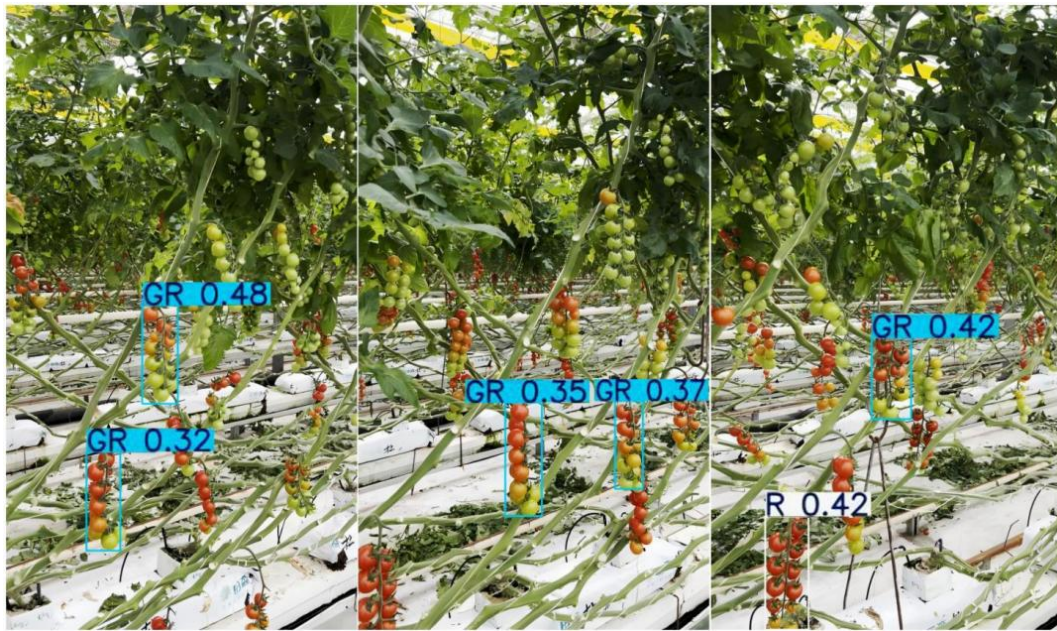
Figure 2. Normalized Confusion Matrix.

Dataset	Target Object	Number of Classes	Total Number of Images	Number of Training Images	Number of validation Images	Number of Testing Images
Custom Dataset	Cherry Tomato Fruits	3	4046	3236	405	405

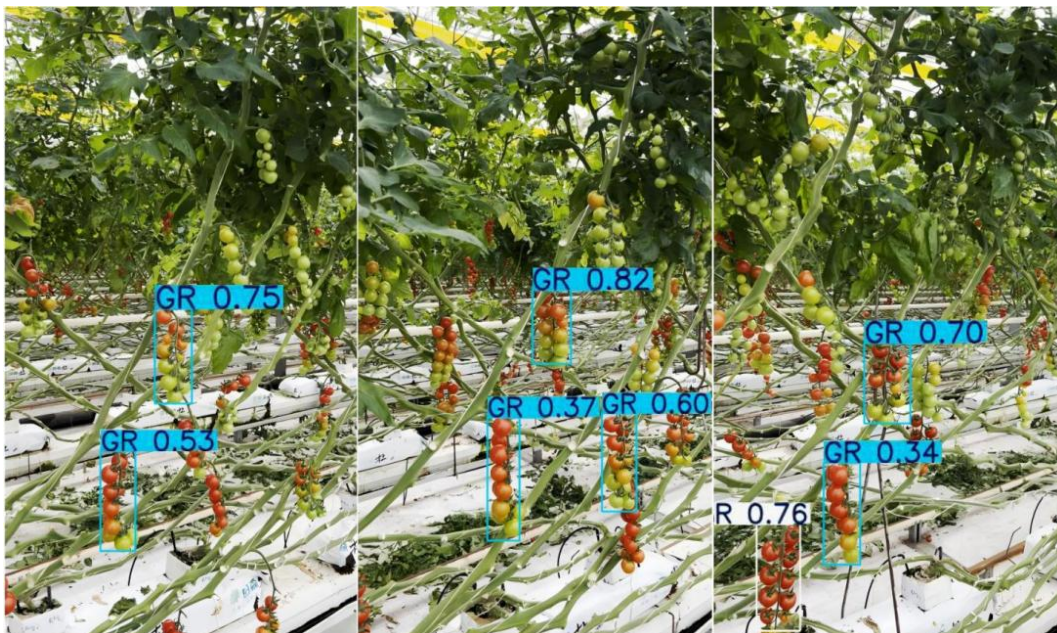
Table 5. Statistical summary of the dataset.



Figure 3. The YOLOv10n-FBD model performance for cherry tomato ripeness recognition.

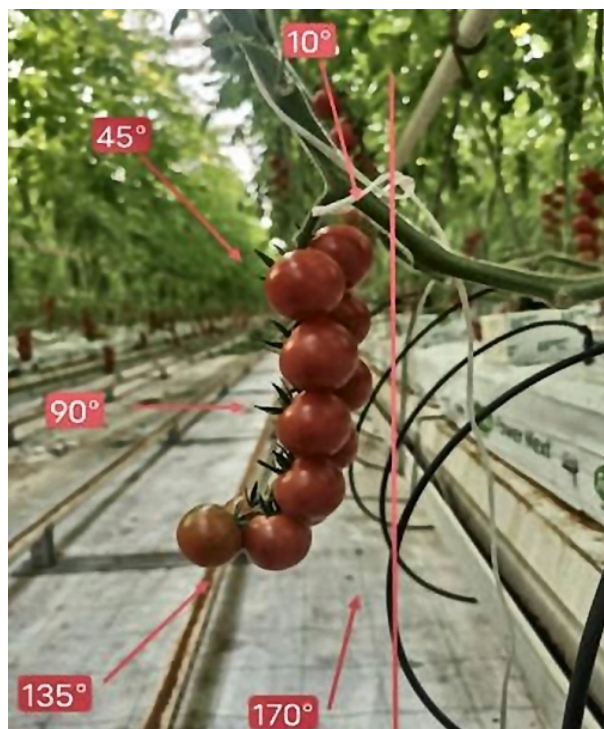


(a) YOLOv10n

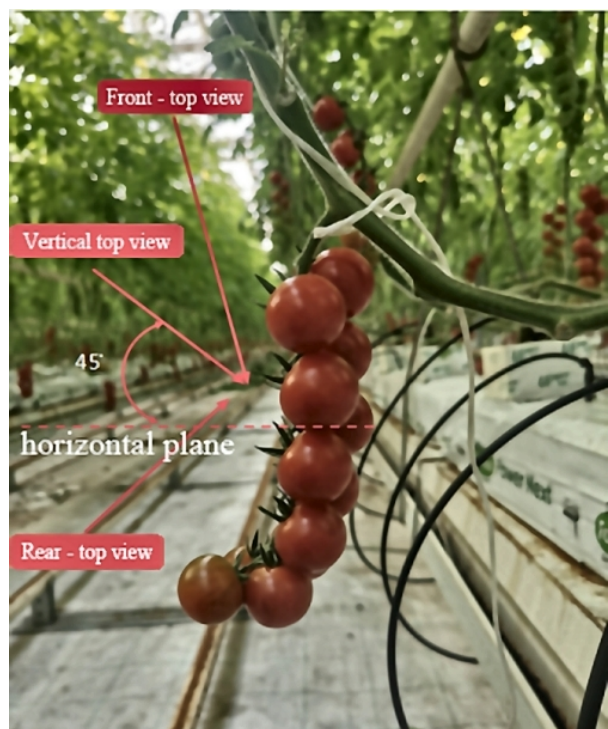


(b) YOLOv10n-FBD

Figure 4. Comparison of video detection results.



(a) Shooting angle



(b) 45° shooting angle change

Figure 5. Schematic diagram of the imaging setup.

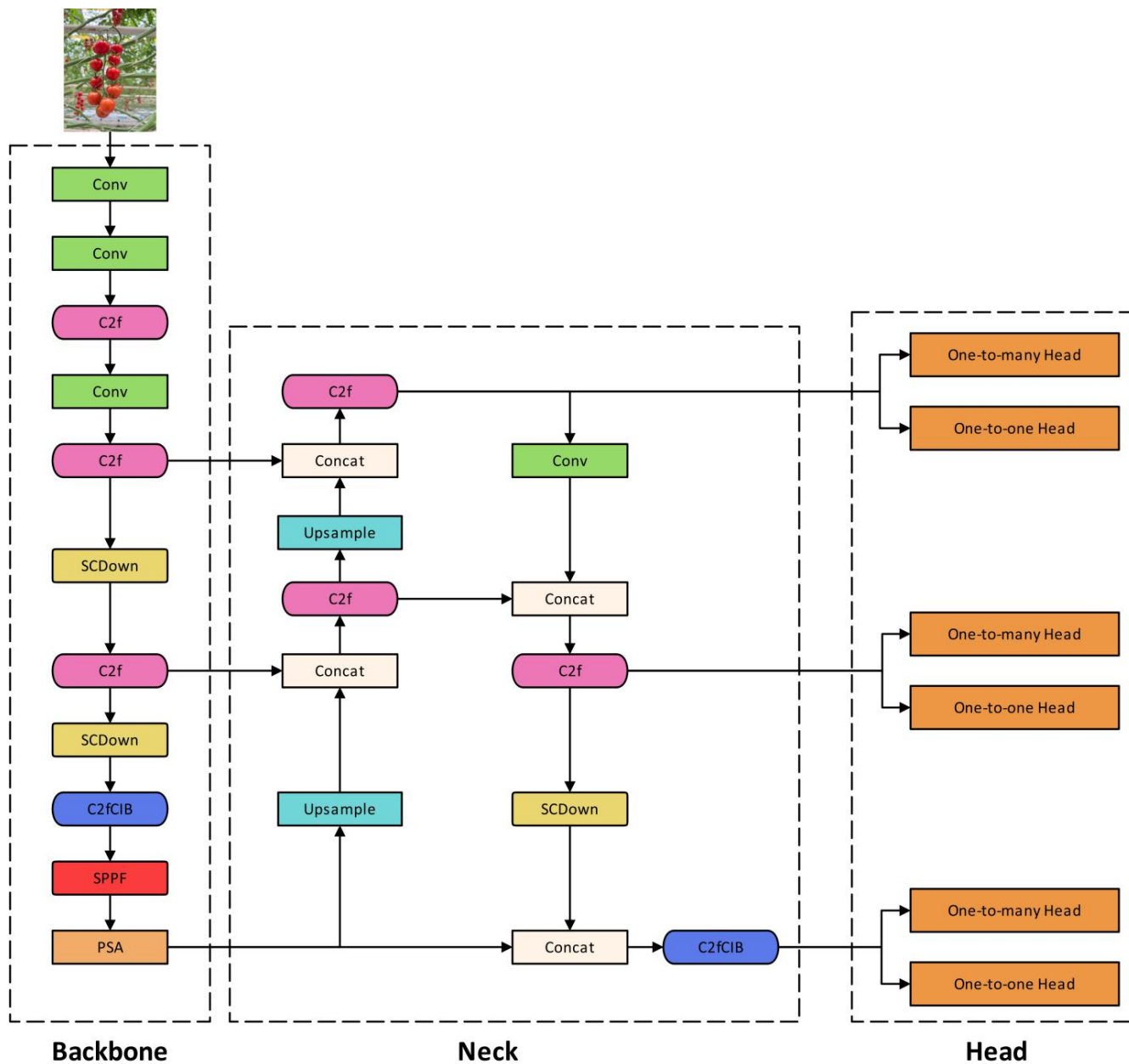
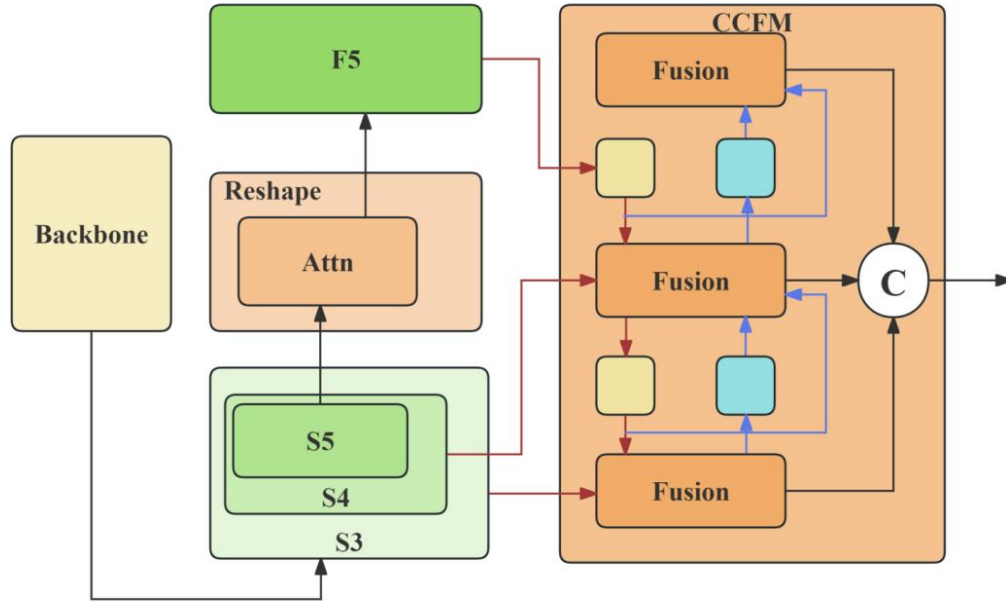
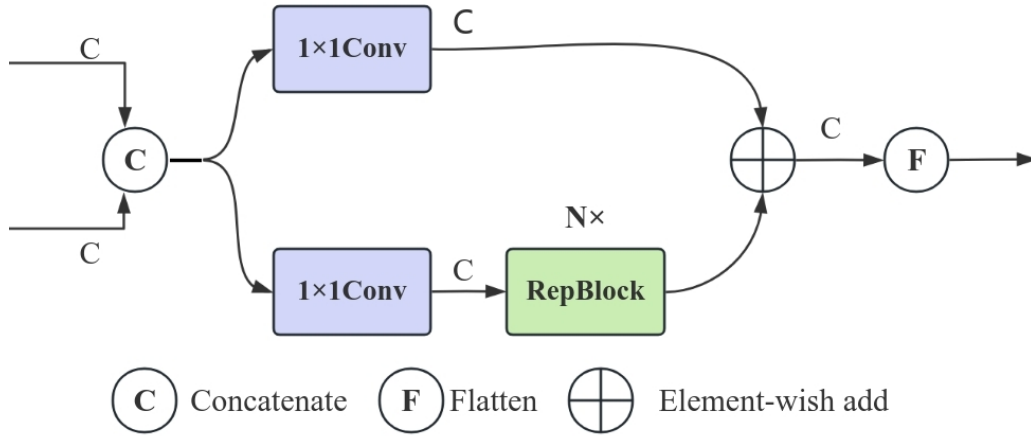


Figure 6. Architecture of the YOLOv10n network.



(a) The CCFM module structure



(b) Structure of the Fusion Block

Figure 7. Detailed structure of the CCFM module.

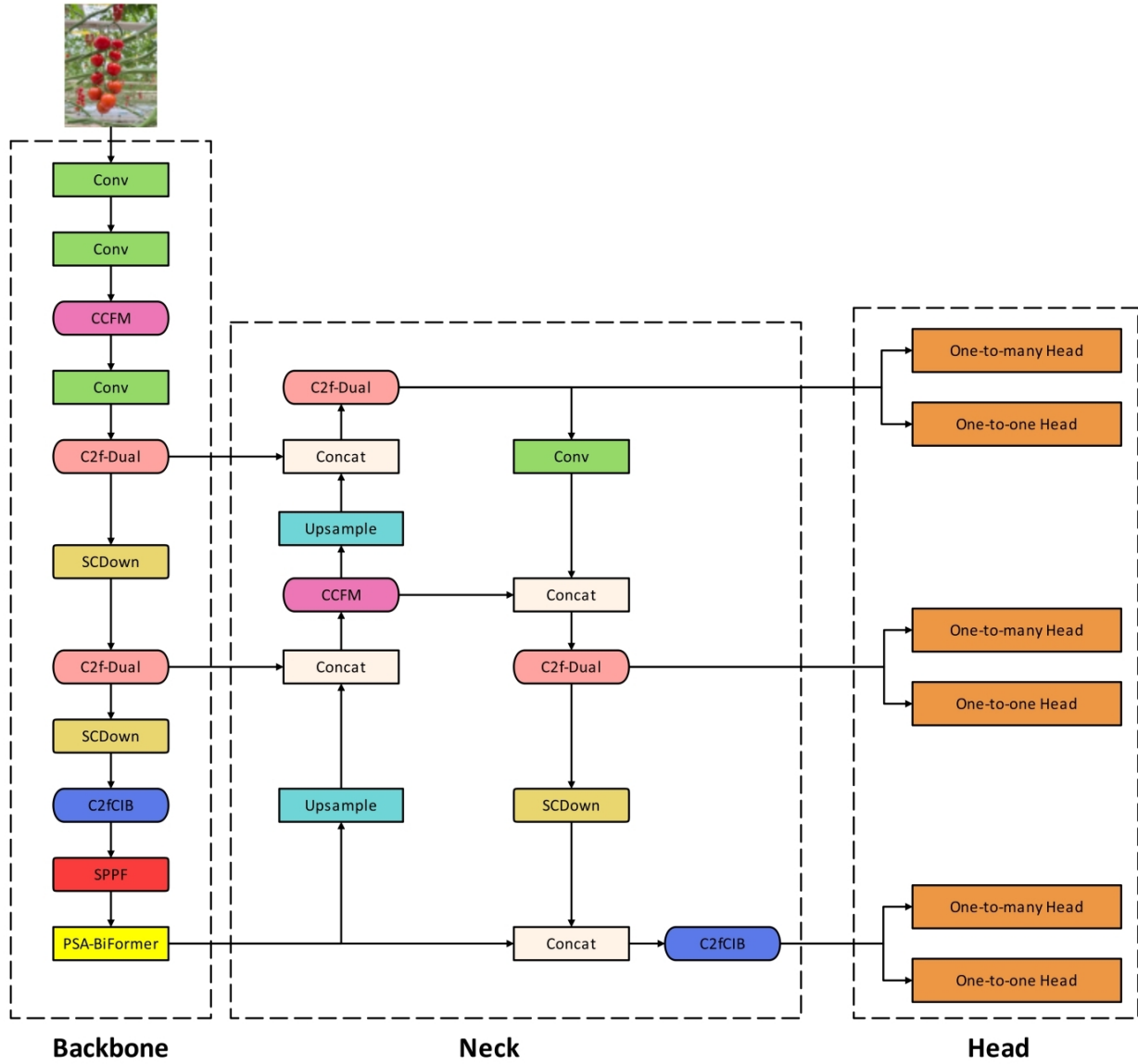


Figure 11. Architecture of the YOLOv10n-FBD network.