Journal of Combinatorial Mathematics
and Combinatorial Computing
www.combinatorialpress.com/jcmcc

Research on Citrus Target Recognition Based on Improved YOLOv8 Algorithm

Danyi Zhang[1,✉], Jun Li[2], Zhengshun Fei[1]

[1] *School of Automation and Electrical Engineering, Zhejiang University of Science & Technology, Hangzhou, Zhejiang, 310000, China*
[2] *Taizhou University, School of Aeronautical Engineering, Taizhou, Zhejiang, 318000, China*

ABSTRACT
In natural orchard environments, tangerines are susceptible to being shaded by foliage and to overlapping with multiple fruits. Varying weather conditions can cause inconsistent levels of illumination, and these unstable factors combined with complex backgrounds can diminish the efficiency of tangerine recognition and localization. Consequently, this paper utilizes images of tangerines captured under various weather conditions within a tangerine orchard as a dataset, and a method based on the YOLOv8n object detection algorithm is proposed. The dataset was trained using BiFPN, MCA attention mechanism, and PConv. An improvement in the algorithm resulted in an accuracy rate of 94.4% for tangerine target detection, a recall rate of 92.7%, an F1 score of 93.5%, and a mAP of 98.3%, with each metric showing an increase of 0.7%, 0.6%, 0.7%, and 1.3% respectively over the original model.

Keywords: Target recognition, YOLOv8n, BiFPN, MCA attention mechanism, PConv, Tangerine

1. Introduction

Yolov8 is an excellent target detection algorithm, which is based on the idea of deep learning, and has high accuracy and real-time performance by training deep convolutional neural network for feature extraction and target detection [1-2]. The implementation details of the algorithm are rich, including network structure, dataset selection, training process, optimization method and so on. With the continuous optimization and improvement of the algorithm, Yolov8 has been widely used in the fields of security, surveillance and unmanned vehicles, and has also received attention in citrus target recognition [3-6].

Citrus target recognition is the basis of automatic picking, a variety of classification and clustering algorithms to design the target recognition model in citrus target recognition in the better target

✉ Corresponding author.
E-mail address: 19818518892@163.com (D. Zhang).
Received 10 January 2024; Revised 15 March 2024; Accepted 25 December 2024; Published Online 16 April 2025.
DOI: 10.61091/jcmcc127b-488

detection results obtained [7-8]. However, the basis of this method is to obtain image features from the fruit itself, when there are a variety of common disturbing factors in the natural picking environment such as light changes, shadow coverage, uneven coloring, branch and leaf shading and fruit overlap, the fruit features change significantly [9-12], which makes the features used to describe the fruit also appear significantly different, so the image feature-based citrus recognition method for the detection of citrus in the natural environment Therefore, image feature-based citrus recognition methods are not satisfactory in natural environment[13-14]. The features of citrus images in natural environment have obvious differences under different interference factors. As the interference factors in the natural environment are more and more varied, it is difficult to obtain citrus target features that cover all the above mentioned interference situations, so it is difficult for the citrus target recognition methods based on image analysis to cope with the simultaneous existence of a variety of interference factors in the natural environment [15-18]. The YOLOv8 algorithm, however, is characterized by deep model hierarchy and strong feature expression ability, which is of great practical significance in citrus target recognition as it can adaptively learn the feature expression needed for the current task from large-scale datasets [19-21].

Although a lot of research has been conducted on target detection techniques, most of the experimental studies have not considered the actual environment. tangerines in the natural environment is easily obscured by branches and leaves, and the fruits are very easy to overlap; different weather conditions, different light intensity, and different degrees of tangerines maturity will cause interference in tangerines recognition; the highest average accuracy of tangerines recognition is 96.9%, and the highest F1 is 89.56%, which is still a lot of room for improvement. Therefore, in this paper, with the understanding of the current research situation, based on the YOLOv8n target recognition algorithm, we use BiFPN to replace FPN in the Head network part, add the MCA attention mechanism in the backbone network, and use PConv to merge into C2f to improve the recognition rate of tangerines in the natural environment. The accuracy of the improved algorithm is 94.4%, the recall is 92.7%, the F1 is 93.5%, and the average precision mean is 98.3%, providing a fast and accurate tangerines recognition model for tangerines picking robots with limited computational power.

2. Theory of improved algorithms for tangerines identification

YOLOv8 is an anchorless frame detection model, which means that instead of predicting the offset of a target from a known anchor frame, it directly predicts the center of the target. Anchorless detection reduces the number of prediction frames as a means of speeding up non-extreme value suppression. In order to meet the needs of real mobile operation, the number of parameters of the model should not be too large and should meet the real-time detection. Among the YOLOv8 series models, YOLOv8n has the smallest model size, which meets the practical application conditions of tangerines picking, but in the task of recognizing tangerines, YOLOv8n still suffers from the problem of missed detection when the recognition target is heavily occluded. Therefore, this study chose to design and improve the model of YOLOv8n. The improved network model diagram is shown in Fig. 1:

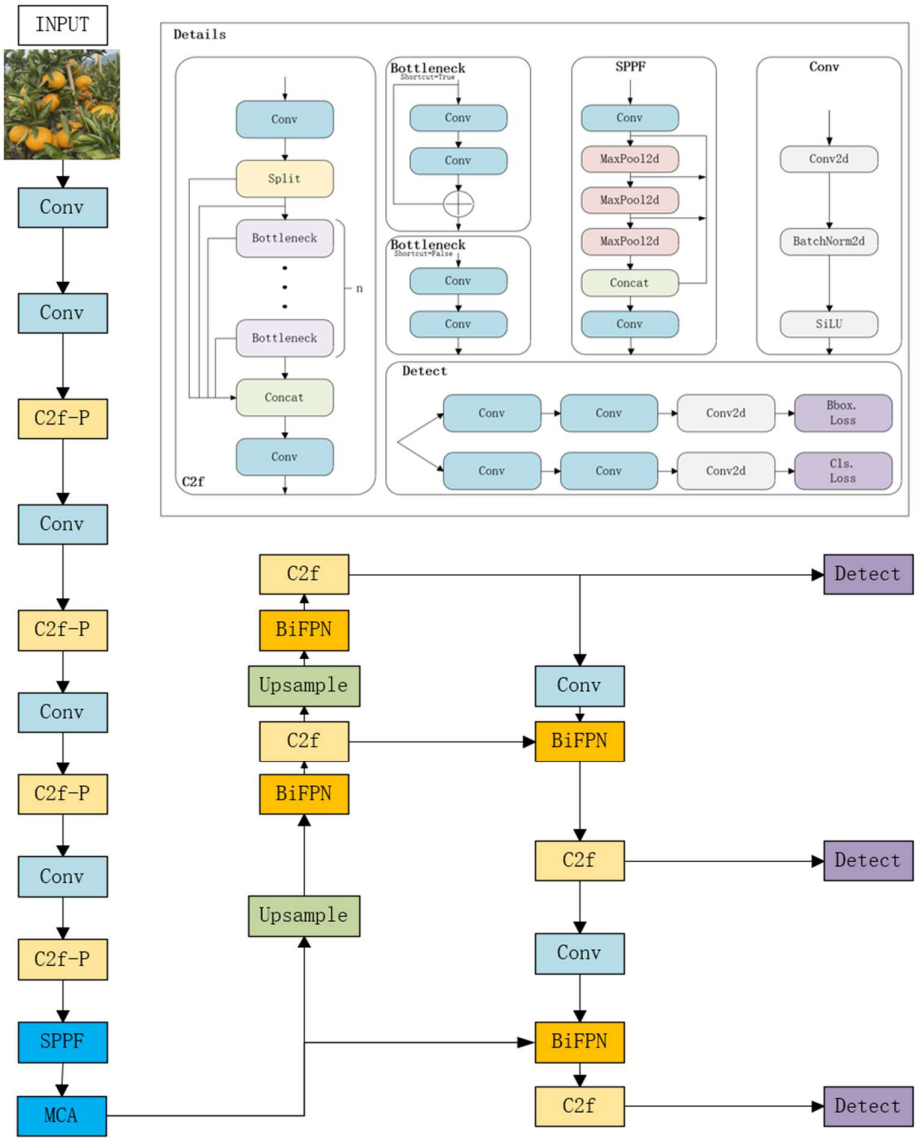

Fig. 1. Improved network model diagram

2.1. BiFPN (Weighted Bidirectional Feature Pyramid Network)

Information exchange and fusion between tangerines image features for different scales is important for enhancing feature extraction and recognition. The feature pyramid network (FPN) shows its superior multi-scale feature fusion capability when dealing with targets with significant scale differences. The design concept of FPN is to enhance the semantic richness of features by transferring high-level semantic information to low-level feature maps through a top-down path. However, FPNs may encounter the problem of diminishing information in deep networks, i.e., important information may be gradually lost as the network level deepens.

To overcome this challenge, YOLOv8 introduces the PAN-FPN structure, which aims to retain more detailed information and enhance feature representativeness by enhancing information flow and fusion between high-level (low-resolution) features and low-level (high-resolution) features. Although PAN-FPN improves the efficiency of information fusion, there are still challenges in achieving finer information capture and parametric efficiency. To further optimize this process, this paper employs BiFPN (Bidirectional Feature Pyramid Network), an efficient multilevel feature fusion strategy. BiFPN achieves more detailed and efficient feature fusion by combining top-down and bottom-up feature flow paths and introducing weighted contextual information edges. This bi-directional feature fusion

path not only increases the semantic depth of the features, but also effectively optimizes the contribution between different features through the weighting mechanism, reduces the information loss, and improves the model performance without significantly increasing the parameters. The detailed structures of PAN and BiFPN are shown in Fig. 2 and Fig. 3.

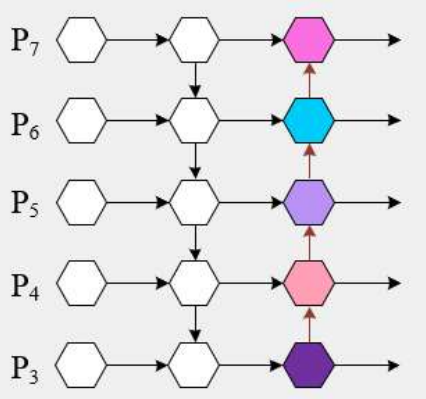

Fig. 2. Detail structure diagram of PAN

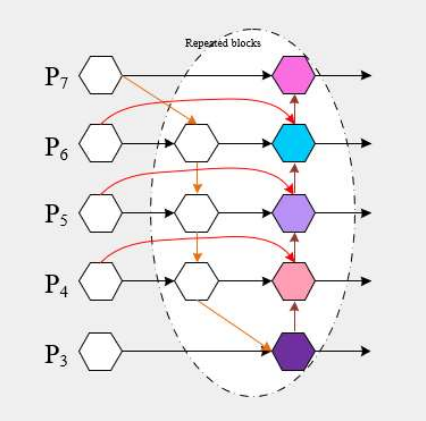

Fig. 3. Detail structure diagram of BiFPN

BiFPN effectively establishes top-down and bottom-up feature fusion channels through its unique two-way feature fusion strategy based on path enhancement, realizing two-way cross-scale feature connectivity. This approach realizes balanced processing of feature information by dynamically adjusting the weights of the importance of input features with different resolutions when fusing multi-scale feature maps, thus effectively improving the detection accuracy of the model. BiFPN removes redundant single-input nodes by simplifying the network structure, while enhancing the connectivity between nodes in the same layer, and this improvement allows the network to more effectively integrate and utilize the abundant feature information. In addition, the design of BiFPN includes a duplicate feature layer utilizing bi-directional paths, which further deepens the depth of feature integration.

In implementing the feature fusion process, considering the significant differences in the importance of different features in network learning, BiFPN adopts the strategy of assigning learnable weights to each input feature, and adaptively learns the importance of features through fast normalization. This approach effectively copes with the problem of ignoring feature resolution differences, which is common in traditional feature fusion methods. The weighting process involves applying a softmax function to normalize each feature weight, ensuring that the weights of different features in the fusion process are both balanced and reasonable, as shown in Equation (1).

$$O = \sum_i \frac{w_i}{\varepsilon + \sum_j w_j} \cdot I_i \tag{1}$$

where, O represents the output features, w_i represents the node weights, I_i represents the input features and the learning rate is set to $\varepsilon = 0.0001$ to ensure numerical stability.

2.2. MCA Attention Mechanism

The actual tangerines collection environment is exceptionally complex, and each image collected contains a large amount of extraneous information, such as leaves and branches. Therefore, it is crucial to extract effective information from this complex background quickly and accurately. Although the traditional attention mechanism SENet is simple and effective to improve the performance of the model, it has some problems, such as increasing the number of parameters, leading to the loss of global average pooled information, limiting the local relevance restriction, and high training overhead. Therefore, this paper adopts the MCA (Multidimensional Collaboration Attention) attention mechanism, which significantly improves the model performance by using efficient modeling spatial features and channel features. Its specific structure is shown in Figure 4 below:

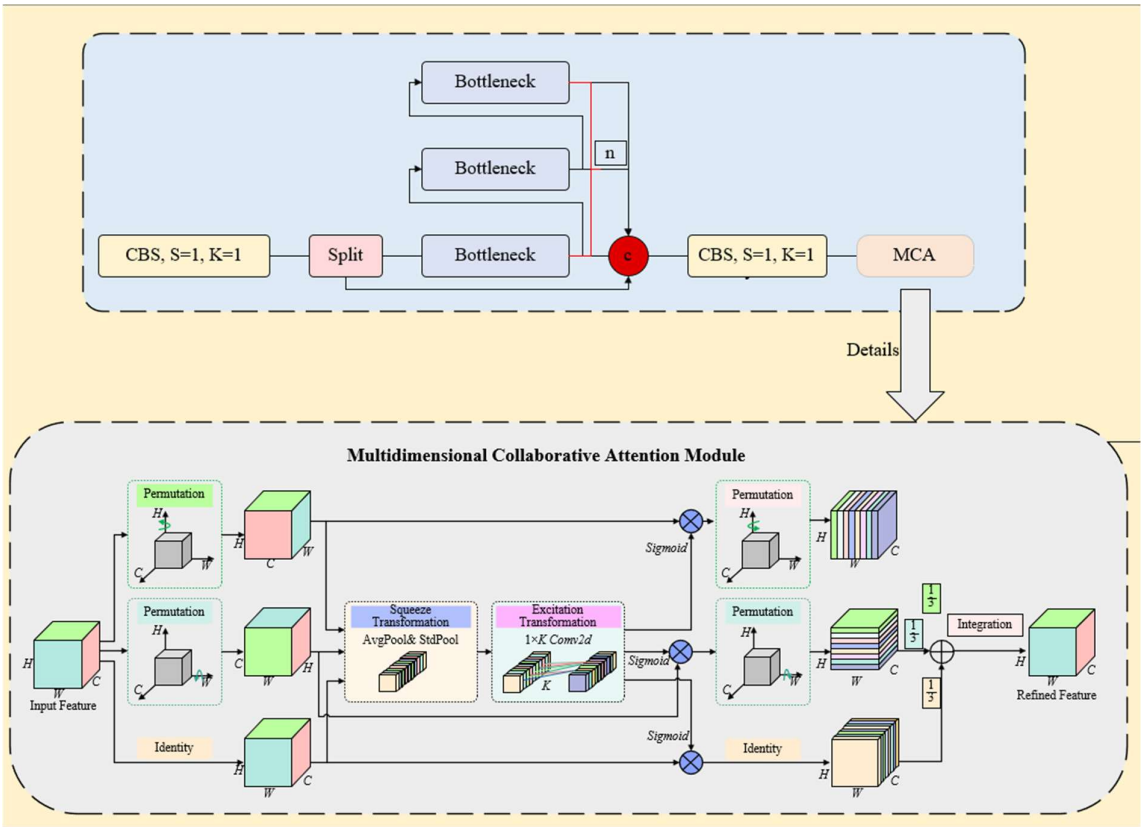

Fig. 4. MCA attention mechanism structure

MCA consists of three main parallel branches, the first two focusing on capturing spatial features dependent features W and H, and the last branch focusing on capturing interactions between channels.

The first branch first performs a H -axis counterclockwise rotation of 90° on the output feature map F of the convolutional layer to generate the rotated feature map $\hat{F}_W$. $\hat{F}_W$ is then fed into the Squeeze Transformation module to model the channel (C) and spatial (H) dimensions to generate the aggregated feature map $\hat{F}_W$. The aggregated feature map $\hat{F}_W$ is then processed through the Excitation Transformation module to capture the interactions between the spatial dimensions and generate the

attention weights via a sigmoid function. These attention weights are applied to the rotated feature map to obtain the enhanced feature map by element-by-element multiplication $F_W^{'}$. Finally, the enhanced feature map is rotated 90° clockwise to restore it to its original shape. This process can be summarized as the following equation 2:

$$\begin{gathered} \hat{F}_W = PM_H\left(F\right) \\ \hat{F}_W = T_{sq}\left(\hat{F}_W\right), \tilde{F}_W = T_{ex}\left(\hat{F}_W\right) \\ \mathrm{A}_W = \sigma\left(\tilde{F}_W\right), F_W^{'} = \mathrm{A}_W \otimes \hat{F}_W, F_W^{''} = PM_H^{-1}\left(F_W^{'}\right) \end{gathered} \tag{2}$$

where $PM_H\left(\cdot\right)$ represents a 90° counterclockwise rotation along the H-axis and $PM_H^{-1}\left(\cdot\right)$ represents its inverse process. $\sigma\left(\cdot\right)$ denotes the sigmoid activation function, and $T_{sq}\left(\cdot\right)$ and $T_{ex}\left(\cdot\right)$ represent the squeezing and excitation transformations, respectively.

Similarly, the second branch is first F rotated 90° counterclockwise along the W-axis, resulting in a rotated feature map denoted as $\hat{F}_H$. In order to model the channel dimension C as well as the spatial dimension H and further capture the interactions between the heights, squeeze transform and excitation transform operations are performed sequentially on $\hat{F}_H$. From this, an aggregated feature map $\hat{F}_H$ and height feature weights $\tilde{F}_H$ are obtained. Subsequently, the attention weights obtained by the sigmoid function are subjected to an element-by-element multiplication operation with the initial features to obtain the enhanced feature map $F_H^{'}$. Finally, the same shape as the original input is obtained by rotating 90° clockwise along the direction of the W-axis. This process can be summarized as the following equation 3:

$$\begin{gathered} \hat{F}_H = PM_W\left(F\right) \\ \hat{F}_H = T_{sq}\left(\hat{F}_H\right), \tilde{F}_H = T_{ex}\left(\hat{F}_H\right) \\ \mathrm{A}_H = \sigma\left(\tilde{F}_H\right), F_H^{'} = \mathrm{A}_H \otimes \hat{F}_H, F_H^{''} = PM_W^{-1}\left(F_H^{'}\right) \end{gathered} \tag{3}$$

The branching design at the bottom is mainly responsible for modeling spatial dependencies on each other and capturing interactions between channels. It first goes through identity mapping to generate its identical feature maps. Sequentially feeding them into the two modules Squeeze Transformation and Excitation Transformation, the aggregated feature maps and channel feature weights can be sequentially inferred. After that, the attention weights are derived by sigmoid function. The original features are rescaled by the attention weights C to generate the enhanced feature map. Finally, the mapping is performed through the feature mapping function. Again, the process is given in Eq. 4 below:

$$\begin{gathered} \hat{F}_C = T_{sq}\left(\hat{F}_C\right), \tilde{F}_C = T_{ex}\left(\hat{F}_C\right) \\ \mathrm{A}_C = \sigma\left(\tilde{F}_C\right), F_C^{'} = \mathrm{A}_C \otimes \hat{F}_C, F_C^{''} = IM\left(F_C^{'}\right) \end{gathered} \tag{4}$$

where $IM\left(\cdot\right)$ refers to the unit mapping feature.

Finally, the branches generated by the three branches need to be integrated, so that the final features obtained can more accurately locate the feature details of interest and improve the discriminatory properties, the overall steps are shown in Equation 5 below:

$$F'' = \frac{1}{3} \otimes \left(F_W^{''} \oplus F_H^{''} \oplus F_C^{''}\right) \tag{5}$$

Here, $F_W^{''}, F_H^{''}$ *and* $F_C^{''}$ can be represented by the following equation 6, respectively:

$$
\begin{aligned}
F_W^{''} &= PM_H^{-1}\left(\sigma\left(T_{ex}\left(T_{sq}\left(PM_H\left(F\right)\right)\right)\right)\otimes PM_H\left(F\right)\right)\\
F_H^{''} &= PM_W^{-1}\left(\sigma\left(T_{ex}\left(T_{sq}\left(PM_W\left(F\right)\right)\right)\right)\otimes PM_W\left(F\right)\right)\\
F_C^{''} &= IM\left(\sigma\left(T_{ex}\left(T_{sq}\left(IM\left(F\right)\right)\right)\right)\otimes IM\left(F\right)\right)
\end{aligned}
\tag{6}
$$

2.3. PConv (partial convolution)

PConv extracts spatial features by performing regular convolution on a portion of the input channel while keeping the rest of the channels unchanged, thus efficiently utilizing the redundant information in the feature map and reducing the amount of computation and computational cost, whose structure is shown in Fig. 5.

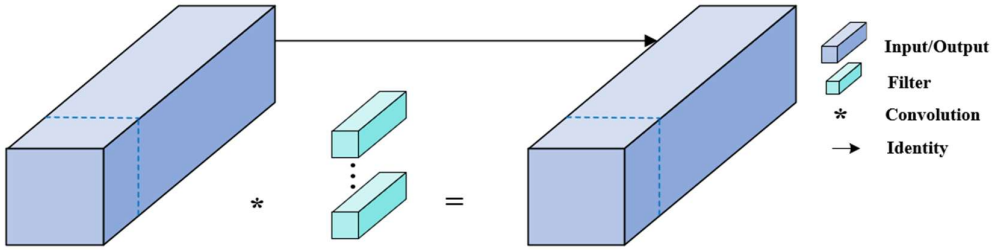

Fig. 5. Pconv module structure

PConv focuses on computing the first C_p or second C_p consecutive channels of the feature map, which serve as the main representatives of the computation for continuous and efficient memory accesses. This design significantly reduces the number of floating-point operations, which can be represented by Equation 7.

$$h\times w\times k^2\times c_p^2 \tag{7}$$

where h and w are the height and width of the feature map, k is the convolution size, and C_p is the selected channel. For a typical partial scale $r=\frac{c_p}{c}=\frac{1}{4}$, the computational effort of PConv is only 1/16th of that of conventional convolution.In addition, PConv has fewer memory accesses, which can be expressed by Equation 8.

$$h\times w\times 2c_p + k^2\times c_p^2 \le h\times w\times 2c_p \tag{8}$$

PConv effectively optimizes the efficiency of parameter usage and reduces the computational cost by localizing the convolution strategy, ensuring that complex tasks can be processed more quickly.

3. Experimental results and analysis

3.1. Data sets and evaluation indicators

The tangerines dataset used in this study was collected manually, and was taken by different means (e.g., camera, cell phone) to meet the diversity and richness of the pictures of the more complex orchard scenes, and then to complete the training and recognition of the more complex scenes in the orchard. In order to make the dataset reflect the real characteristics of tangerines targets in the natural environment, the tangerines samples collected in the image collection were selected from the tangerines produced in Yongquan Town, Linhai District, Taizhou, Zhejiang Province, and many photos with different characteristics were taken, including: the fruit tree branches and trunks of the leaves cover the tangerines, and the tangerines fruits overlap each other's characteristics, which makes the

tangerines fruits not completely exposed from the flat image; the tangerines fruits ripen faster and slower than the other tangerines fruits in the natural environment. In addition to ripening tangerines, we also photographed unripe tangerines, and took multiple shots in different weather conditions, such as sunny days (strong light), after rain (water droplets on the tangerines fruits), and in the evening (dark surrounding colors). Image acquisition work was carried out on different tangerines at different times using different angles, and a total of more than 3000 high-quality images were captured, as shown in Fig. 6. In addition, the data-enhanced images were accurately labeled using VoTT software in this study. In order to meet the experimental requirements, we randomly assigned the images in the dataset to the training set, validation set and test set in the ratio of 8:1:1.

Fig. 6. Sample dataset

The environment for this experiment is configured as follows: operating system is Linux, CPU is Intel Xeon E5-2680v3, GPU is NVIDIA GeForce RTX 4090, and memory is 11 GB VRAM. pyTorch 1.7.0 framework and Python version 3.8. The experimental setup contains hyperparameters such as an initial learning rate of 0.01, 300 integrated training cycles, and a batch size of 32.

In order to fully validate the performance of the improved YOLOv8 algorithm for tangerines detection, four basic evaluation metrics were used in this study as precision (P), recall (R), average precision (mAP), and $F1$. The corresponding formulas are given below in Equation 9:

$$\begin{aligned} P &= \frac{TP}{TP+FP} \\ R &= \frac{TP}{TP+FN} \\ F1 &= \frac{2\times P\times R}{P+R} \\ mAP &= \frac{\sum_{q=1}^{Q} AP(q)}{Q} \end{aligned} \tag{9}$$

where precision is used to measure the accuracy of positive case prediction in a binary classification problem and quantifies the proportion of samples predicted by the model that are actually positive cases. Recall, on the other hand, calculates the proportion of samples that are actually positive examples that are correctly predicted as positive examples. $F1$ is the weighted summed average of precision and recall, which is used to assess the accuracy and recall performance of the model as a

whole, and takes a value ranging from 0 to 1. The closer it is to 1, the better the overall performance of the model. Mean precision (mAP) represents the average of the mean precision of multiple categories, which is used to evaluate the performance of multiple categories in the target detection task.

3.2. Comparison before and after improvement

In order to verify a superiority of the present model before and after improvement, the optimized model is compared with YOLOv8n using the same dataset while ensuring that the relevant configurations remain unchanged. The specific results are shown in Table 1, and the tangerines detection results in different scenarios are shown in Figure 7.

Table 1. Comparison between YOLOv8n and Improved Model

Model	Precision (%)	Recall (%)	F1 (%)	mAP (%)
YOLOv8n	93.7	92.1	92.8	97.0
Improvements	94.4	92.7	93.5	98.3

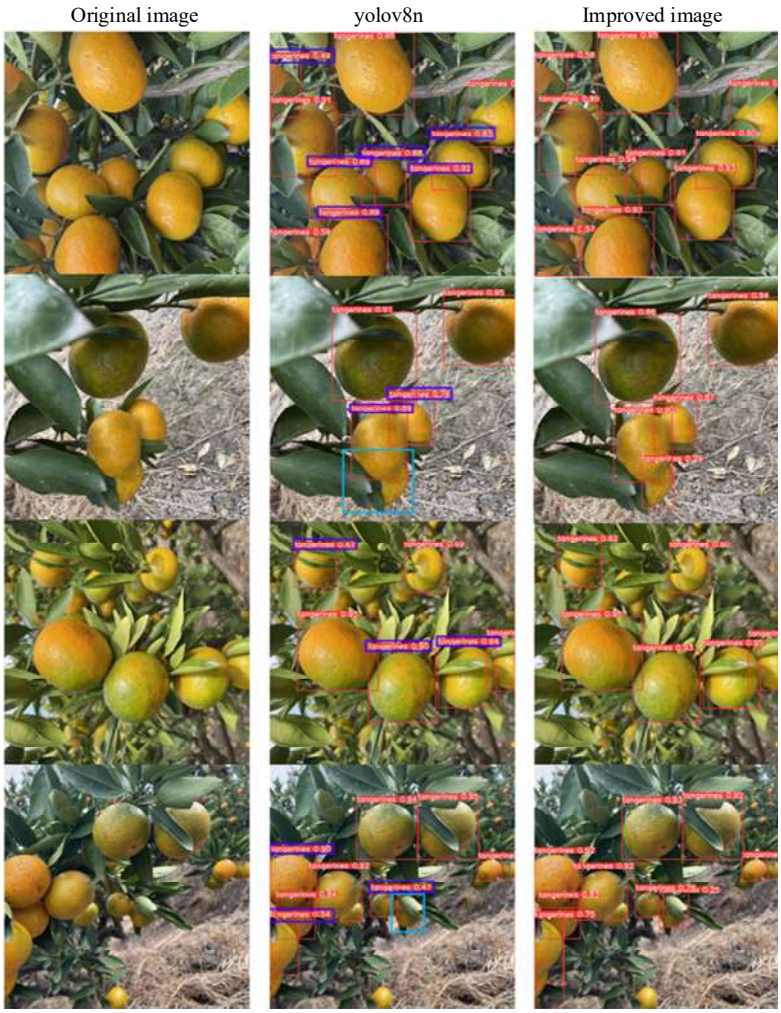

Fig. 7. Comparison of Tangerines Detection Results between YOLOv8n and Improved Model

The data in Table 1 show that the optimized model significantly improves the performance over the YOLOv8n model in four metrics: precision, recall, mAP and $F1$, by 0.7%, 0.6%, 0.7% and 1.3%, respectively; according to Fig. 7, the optimized model is more stable. In Fig. 7, the blue box is the tangerines missed by YOLOv8n, and the purple box is the data for which the confidence level of

YOLOv8n is lower than that of the improved network. For example, in the first figure, the confidence level of YOLOv8n algorithm for the tangerines in the upper-left corner is 0.49, while that of the improved network is 0.58. In the third figure, the confidence level of YOLOv8n algorithm for the tangerines in the upper-left corner is 0.47, while that of the improved network is 0.62. 0.62; in the fourth figure, the YOLOv8n algorithm has a confidence level of 0.54 for the tangerines in the lower left corner, while the improved network has a confidence level of 0.75. This is in part related to the introduction of the BiFPN architecture, which has better accuracy and efficiency trade-offs and optimizes the fusion of targets at different scales. Secondly, the use of MCA can effectively extract the tangerines features in the image and improve the accuracy of target detection. In addition, the reference of PConv module makes the classification model more focused on tangerines-related regions, which improves the computational efficiency and generalization ability and enhances the model stability while maintaining a high accuracy. These performance improvements are mainly achieved by focusing on useful feature information in a targeted manner and reducing the computational cost, thus enabling the model to run efficiently. In conclusion, the improved YOLOv8n model has higher accuracy for tangerines detection and meets the demand for real-time accurate tangerines detection.

3.3. Ablation experiments

In order to assess the effectiveness of the YOLOv8n model design, this study uses an ablation experiment to provide an in-depth exploration of the novel network structure. The experiment aims to explicitly reveal the practical value embedded in each enhancement strategy and provide a more intuitive demonstration of the model performance. The relevant experimental results have been presented in detail in Table 2.

Table 2. YOLOv8n ablation experiment

Model	Precision (%)	Recall (%)	F1 (%)	mAP (%)
YOLOv8n	93.7	92.1	92.8	97.0
YOLOv8n+ BiFPN	93.9	92.3	93.0	97.6
YOLOv8n+ BiFPN +MCA	94.1	92.5	93.3	98.0
YOLOv8n+ BiFPN +MCA+PConv	94.4	92.7	93.5	98.3

According to the data in Table 2, the mAP value of the model achieves a significant improvement of 0.6% after introducing the enhanced BiFPN. This optimization is mainly due to the fact that the bidirectional information flow structure in BiFPN plays an important role in effectively integrating multi-scale feature information, which in turn improves the precision and recall of detection and ultimately enhances the mAP value. In addition, BiFPN introduces learnable weights to determine the importance of different input features, which improves the ability of model feature fusion.

After integrating the MCA attention mechanism, a 0.4% increase on the mAP value of the model is achieved. This significant improvement is mainly attributed to MCA's ability to enhance the perceptual capabilities of YOLOv8 using efficiently modeled spatial and channel features, allowing it to pay more attention to the tangerines region when processing images.

The introduction of the enhanced PConv results in a significant improvement of 0.3% in the model's mAP value. This improvement is mainly due to the introduced PConv mechanism, which enables the model to process the input data more efficiently by means of information rearrangement, reduces the amount of unnecessary computation, accelerates the training and inference process of the model, and improves the efficiency of the model.

3.4. Other comparative experiments

In order to evaluate the advantages of this paper's model in terms of detection performance, this study conducted a series of experiments to compare and analyze it with the mainstream target detection models of single-stage and two-stage, and the results are shown in Table 3.

Table 3. Other comparative experiments

Model	Precision (%)	Recall (%)	F1 (%)	mAP (%)
Faster-RCNN	87.6	86.1	86.8	91.2
SSD	88.1	87.3	87.7	92.6
YOLOv5	90.8	89.9	90.3	95.2
YOLOv7	92.0	91.2	91.6	96.1
Improvements	94.4	92.7	93.5	98.3

As shown in Table 3, when analyzing the limitations of the Faster R-CNN model in terms of detection accuracy, it can be found that its lack of performance mainly stems from the fact that it relies on predefined anchor frames to generate the candidate regions, and this design problem may lead to the model not being able to cover all the tangerines regions, which may result in leakage or misdetection, which is a significant shortcoming in comparison with other two-stage detection models. In addition, the complex network structure and large number of parameters of Faster R-CNN also increase the computational cost and cause difficulties in the deployment of end devices. With the increase of network depth, the feature information of small targets is easily lost in the propagation process due to the lower resolution of the feature maps in the lower layers, thus weakening the performance of SSD in detecting smaller tangerines. In contrast, YOLOv3 successfully improves the accuracy of tangerines detection for multiple scale sizes while maintaining fast and lightweight through finer scale division and more effective feature fusion.YOLOv5, as an upgraded version of YOLOv3, demonstrates higher accuracy regardless of the target detection scale size and complexity. In addition, more efficient training methods, such as data augmentation and loss function optimization, enable faster convergence and better results, which also improves the generalization ability of this model. Although the lightweight YOLOv7 has certain detection performance and generalization ability, however, it may need further optimization and improvement in complex scenes and small target detection.

In contrast, the proposed YOLOv8 model uses a more advanced feature extraction network and introduces an improved multi-scale prediction technique to achieve high real-time performance and high accuracy for tangerines detection at multi-scale sizes, enabling the tangerines detection process to be completed quickly and improving recognition efficiency and detection speed.

4. Conclusion

tangerines in the natural orchard environment is easy to be shaded by branches and leaves and multiple fruits overlap, different weather conditions will lead to different light and darkness, these unstable factors will cause interference in the identification and localization of tangerines. Therefore, the dataset is based on the photos taken in the natural environment of tangerines orchards, and the photos of tangerines with different brightness and darkness in different weather conditions are selected. In this paper, based on the YOLOv8n target recognition algorithm, BiFPN is used to replace FPN in the head network part, which reduces the information loss and improves the performance of the model; MCA attention mechanism is added to the backbone network, which improves the network's characterization ability and recognition accuracy; and PConv is fused into C2f, which makes use of the redundancy information of the feature map, and reduces the amount of computation and the cost of operation. This finally resulted in an accuracy of 94.4%, a recall of 92.7%, an F1 of 93.5%, and a mAP of 98.3%, all of which were improved compared to the rest of the network, providing a theoretical basis for automatic tangerines picking.

References

[1] Wang, F., Wang, H., Qin, Z., & Tang, J. (2023). UAV target detection algorithm based on improved YOLOv8. IEEE Access.

[2] Wang, J., & Zhao, H. (2024). Improved YOLOv8 Algorithm for Water Surface Object Detection. Sensors, 24(15), 5059.

[3] Varghese, R., & Sambath, M. (2024, April). YOLOv8: A Novel Object Detection Algorithm with Enhanced Performance and Robustness. In 2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS) (pp. 1-6). IEEE.

[4] Wang, H., Yang, H., Chen, H., Wang, J., Zhou, X., & Xu, Y. (2024). A Remote Sensing Image Target Detection Algorithm Based on Improved YOLOv8. Applied Sciences, 14(4), 1557.

[5] Liu, L., Li, P., Wang, D., & Zhu, S. (2024). A wind turbine damage detection algorithm designed based on YOLOv8. Applied Soft Computing, 154, 111364.

[6] Jiang, W., Han, D., Han, B., & Wu, Z. (2024). YOLOV8-FDF: a small target detection algorithm in complex scenes. IEEE Access.

[7] Qin, X., Huang, R., & Hua, B. (2021, March). Research and implementation of yield recognition of Citrus reticulata based on target detection. In Journal of physics: conference series (Vol. 1820, No. 1, p. 012128). IOP Publishing.

[8] Jia, X., Jiang, X., Li, Z., Mu, J., Wang, Y., & Niu, Y. (2023). Application of deep learning in image recognition of citrus pests. Agriculture, 13(5), 1023.

[9] Wang, C., Luo, Q., Chen, X., Yi, B., & Wang, H. (2021, March). Citrus recognition based on YOLOv4 neural network. In Journal of Physics: Conference Series (Vol. 1820, No. 1, p. 012163). IOP Publishing.

[10] Xu, L., Wang, Y., Shi, X., Tang, Z., Chen, X., Wang, Y., ... & Zhao, Y. (2023). Real-time and accurate detection of citrus in complex scenes based on HPL-YOLOv4. Computers and Electronics in Agriculture, 205, 107590.

[11] Xu, L., Zhu, S., Chen, X., Wang, Y., Kang, Z., Huang, P., & Peng, Y. (2020). Citrus recognition in real scenarios based on machine vision. DYNA-Ingeniería e Industria, 95(1).

[12] Xiao, X., Huang, J., Li, M., Xu, Y., Zhang, H., Wen, C., & Dai, S. (2022). Fast recognition method for citrus under complex environments based on improved YOLOv3. The Journal of Engineering, 2022(2), 148-159.

[13] Changhui, Y. A. N. G., Yanping, L. I. U., Yi, W. A. N. G., Longye, X. I. O. N. G., Hongbin, X. U., & Wanhua, Z. H. A. O. (2019). Research and experiment on recognition and location system for citrus picking robot in natural environment. Nongye Jixie Xuebao/Transactions of the Chinese Society of Agricultural Machinery, 50(12).

[14] Chen, G., Hua, C., & Zhang, Z. (2023, July). Recognition of Citrus on Trees in Different Growth Periods. In 2023 42nd Chinese Control Conference (CCC) (pp. 7321-7326). IEEE.

[15] Hua, Y., Wu, S., Liang, L., Jiang, M., Fan, Y., & Xie, W. (2024, July). Citrus objects recognition in orchard scene using HSV methods. In International Conference on Image Processing and Artificial Intelligence (ICIPAl 2024) (Vol. 13213, pp. 630-635). SPIE.

[16] Xiuyan, G. A. O., & ZHANG, Y. (2023). Detection of Fruit using YOLOv8-based Single Stage Detectors. International Journal of Advanced Computer Science & Applications, 14(12).

[17] Tian, Y., Duan, H., Luo, R., Zhang, Y., Jia, W., Lian, J., ... & Li, C. (2019). Fast recognition and location of target fruit based on depth information. IEEE Access, 7, 170553-170563.

[18] Yasmeen, U., Khan, M. A., Tariq, U., Khan, J. A., Yar, M. A. E., Hanif, C. A., ... & Nam, Y. (2021). Citrus diseases recognition using deep improved genetic algorithm. Comput. Mater. Contin, 71(2).

[19] YUE, K., ZHANG, P., WANG, L., GUO, Z., & ZHANG, J. (2024). Recognizing citrus in complex environment using improved YOLOv8n. Transactions of the Chinese Society of Agricultural Engineering, 40(8), 152-158.

[20] Xie, W., Feng, F., & Zhang, H. (2024). A Detection Algorithm for Citrus Huanglongbing Disease Based on an Improved YOLOv8n. Sensors, 24(14), 4448.

[21] Ang, G., Zhiwei, T., Wei, M., Yuepeng, S., Longlong, R., Yuliang, F., ... & Lijia, X. (2024). Fruits hidden by green: an improved YOLOV8n for detection of young citrus in lush citrus trees. Frontiers in Plant Science, 15, 1375118.