International Journal of Biology and Life Sciences
ISSN: 2957-9511 | Vol. 8, No. 2, 2024

Tomato Leaf Disease Detection based on the Improved YOLOv10n

Guogang Wang, Xu Jiang

School of Information and Control Engineering, Jilin Institute of Chemical Technology, Jilin, Jilin 132000, China

Abstract: In view of the difficulty in distinguishing tomato leaf lesions and the poor identification efficiency, an improved disease detection algorithm CGS-YOLOv10n based on YOLOv10n network was proposed. Firstly, the C2f in the global network is replaced by a multi-scale feature fusion C2f_DBB module, which can extract features more effectively. Secondly, SIOU-SoftNMS is partially used in the neck network to accurately deal with the differences between different targets, so as to improve the classification performance. Finally, the GDetect module was used to replace the detection head part of the original head network, so as to improve the accuracy and efficiency of recognition. The results show that compared with the traditional YOLOv10n, the improved CGS-YOLOv10n algorithm has an accuracy of 93.8%, a recall rate of 91.6%, and an average accuracy of 95.8%. The improved model has higher accuracy and stability in the disease detection of tomato leaf images, which can not only locate and identify targets more accurately, but also effectively reduce the rate of missed detection and false detection. The method proposed in this study provides a reliable technical means for the timely prevention and precise management of tomato diseases in agricultural production, which is helpful to promote the healthy development and sustainable development of agricultural industry.

Keywords: Tomato Leaf Disease; CGS-YOLOv10n; C2f_DBB; GDetect; SIOU-SoftNMS.

1. Introduction

Tomato is a very important cash crop in China's agricultural field, and its yield demand is increasing year by year. However, the frequent occurrence of tomato diseases has led to a significant decline in crop yield, which has caused great losses to our agricultural economy that cannot be ignored. Therefore, the early diagnosis and control of tomato diseases is a necessary condition to promote the high yield of tomato. In order to deal with the problems of large investment, long time, high cost and low accuracy of professional diagnostic training, crop diseases can be identified quickly and accurately with the help of image processing and recognition technology, thus reducing production costs.

In recent years, many domestic scholars have conducted research on plant disease detection methods. Liu et al. [1] improved the YOLOv3 algorithm and applied image pyramid technology to achieve multi-scale feature detection of tomato diseases and insect pests, realized rapid classification of diseases and insect pests and accurate identification of specific locations, and successfully conquered the key technologies of tomato diseases and insect pests image recognition in natural environments. Liu Jianhang et al. [2] proposed and verified an efficient tomato string detection model DC-YOLOv4, which can achieve sparsity of model parameters and improve the recognition accuracy of targets. Wang Na et al. [3] proposed an improved light-weight YOLOv5n method for tomato leaf disease recognition, which maintained a high recognition accuracy while reducing model size, parameter number and calculation amount. Xu Yue et al. [4] proposed an improved disease detection algorithm BKO-Yolov8s based on YOLOv8s network. Compared with the traditional YOLOv8s, its accuracy rate increased by 2.8 percentage points, recall rate increased by 3.0 percentage points, and mAP@50 increased by 2.8 percentage points, effectively reducing the rate of missed and false detection. There have been some achievements in the detection and recognition algorithm of tomato leaf diseases, but the recognition accuracy and accuracy need to be improved. YOLOv10n [5], as a single-stage detection algorithm recently launched by the YOLO series, achieves a good balance in terms of model size and performance. Therefore, the YOLOv10n network was selected in this study to optimize the algorithm for specific disease localization and recognition of tomato leaves, in order to improve the detection accuracy of tomato leaf disease recognition under the premise of reducing the amount of computation.

2. CGS-YOLOv10n Model Structure

2.1. YOLOv10n Network Model

As a relatively new generation algorithm in the YOLO series, YOLOv10n algorithm effectively improves detection performance and efficiency by eliminating non-maximum suppression (NMS) processing and optimizing model design [2], as shown in Fig. 1.

2.2. CGS-YOLOv10n Network Model

In this study, a new network model architecture, CGS-YOLOv10n, was proposed, as shown in Fig.2. The model is mainly based on YOLOv10n and optimized on it, including the use of C2f_DBB module with multi-scale fusion features in the global network[7], the partial use of SIOU-SoftNMS loss function in the neck network[8], and the use of GDetect header module [9] to reduce the calculation cost, Improve detection accuracy and improve efficiency.

2.3. C2f_DBB Model

In the original YOLOv10n network, the bottleneck module of C2f usually reduces the number of feature dimensions, but it may limit the ability of the model to express features. This is because part of the spatial information will be lost during the convolution operation. The DBB module enriches the function of a single convolution by combining multiple branches of different sizes and complexity to improve the performance of the model. As shown in Fig.3, the branches of the DBB module cover a series of convolution operations, multi-scale convolution operations, and average pooling operations. These operations cooperate with each other so that DBB module can handle feature information better, which to a certain extent solves the potential shortcomings of the bottleneck module, provides strong support for the improvement of model performance, and helps the model to achieve better performance in tasks such as target detection.

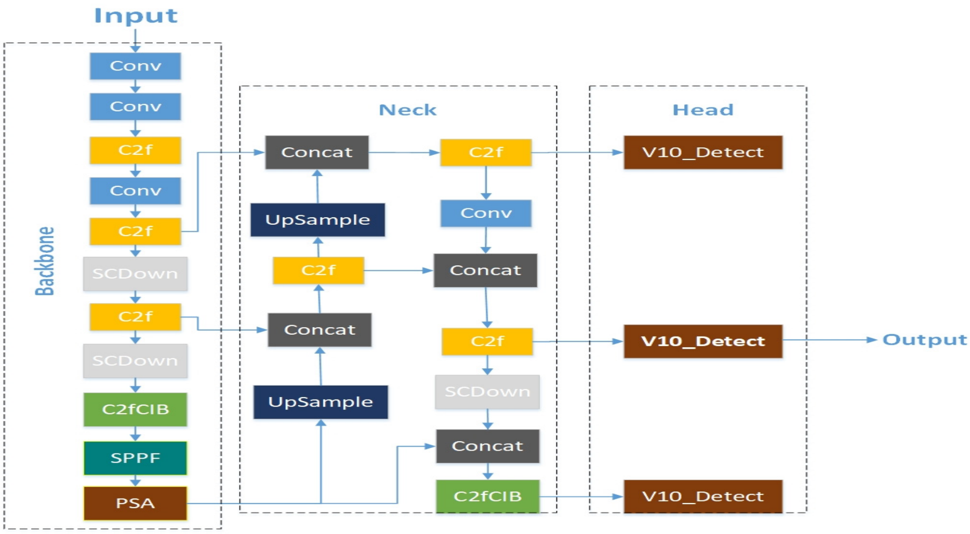


Fig 1. Structure of YOLOv10n Network Model

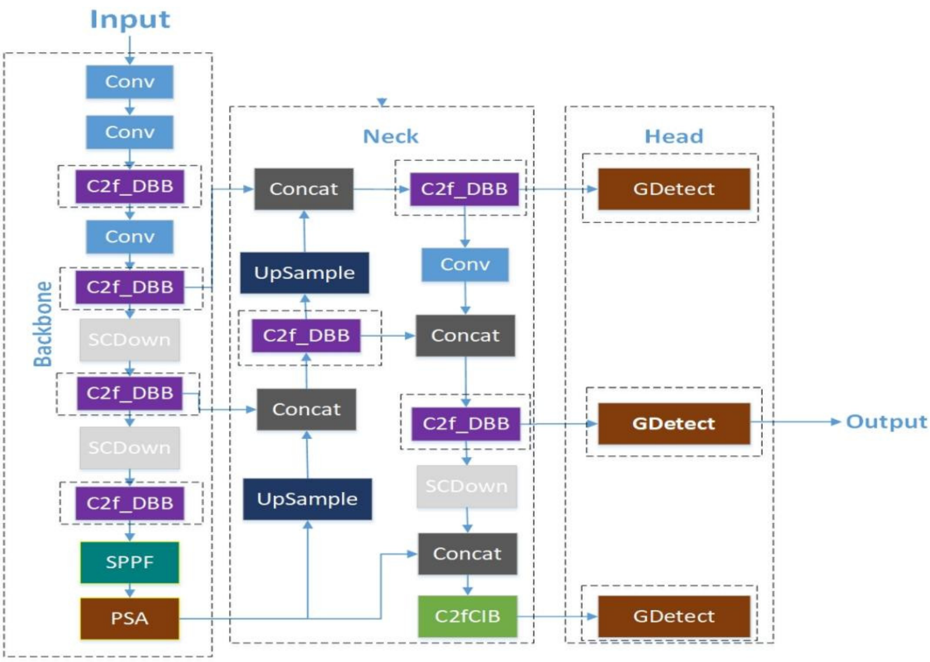


Fig 2. Structure of CGS-YOLOv10n Network Model

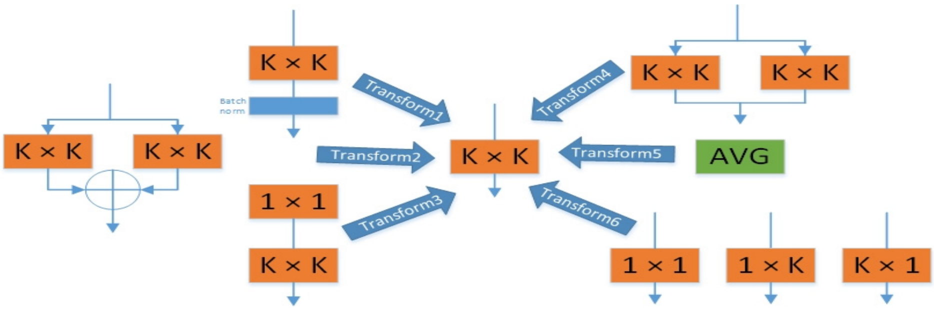


Fig 3. Structure of DBB Network Model

Branch 1 contains convolution and BN layers, which are used for channel normalization and prior scaling. Branchis an addition transformation. This addition property ensures that if the outputs of two or more convolution layers with the same configuration are summed, they can be combined into a single convolution. Branch 3 is a sequential convolution operation, which firstly combines channels linearly by 1×1 convolution, and then realizes spatial aggregation by K×K convolution.

Branch 4 uses deep joins to merge K×K convolution, and each K×K convolution can be operated by convolution of branch 3; Branch 5 is an average pooling operation, which performs downsampling when s>1 and smoothing when s = 1. Branch 6 is a multi-scale convolution operation, which can realize multi-scale feature extraction.

As shown in Fig.4, this paper replaces the bottleneck module in C2f with DBB module in order to enhance the surface feature extraction ability of tomato leaf diseases, and then improve the overall feature extraction ability of the model. This is because DBB module has unique structural and functional advantages, and the synergistic effect of its multiple branches can better process complex feature information in tomato leaf disease images, providing a richer and more accurate feature basis for subsequent model analysis and judgment.

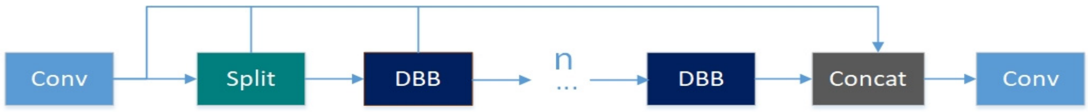


Fig 4. Structure of C2f_DBB Network Model

2.4. GDetect Model

In order to achieve the goal of reducing the number of parameters and calculation amount of YOLOv10n network and improving the accuracy and efficiency of detection at the same time, this paper carried out improvement work on the Detect module of the original YOLOv10n network. As shown in Fig.5, in a conventional convolution operation, the input and output channels are fully connected. However, packet convolution is different in that it divides the input channels into several groups evenly, then carries out convolution operations for the channels in each group separately, and finally combines the outputs of each group to form the final output result. This operation mode of grouping convolution can effectively reduce the number of parameters of the model, reduce the calculation cost to a certain extent, and then improve the calculation efficiency of the model. This improvement strategy is of great significance for optimizing network performance. By reasonably adjusting the convolutional mode, the model can run more efficiently while maintaining a certain detection accuracy, and provides more favorable technical support for the realization of relevant target detection tasks.

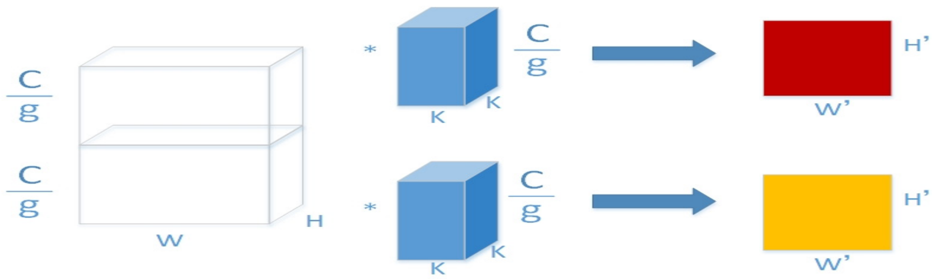


Fig 5. Structure of GConv Network Model

The parameters and number of Flops of ordinary convolutional layers are calculated as follows:

$$params = C_1 \times h_1 \times w_1 \times C_2 \quad (1)$$

$$FLOPs = C_1 \times C_2 \times W_2 \times H_2 \times w_1 \times h_1 \quad (2)$$

The parameters and number of GLOps of the grouping convolution are calculated as follows:

$$params = \frac{1}{g} \times C_1 \times h_1 \times w_1 \times C_2 \quad (3)$$

$$FLOPs = \frac{1}{g} \times C_1 \times C_2 \times W_2 \times H_2 \times w_1 \times h_1 \quad (4)$$

In this paper, GConv in the Detect part of the original YOLOv10n is replaouping structure reduces the number of parameters in each convolution kernel, thus reducing the computational complexity.

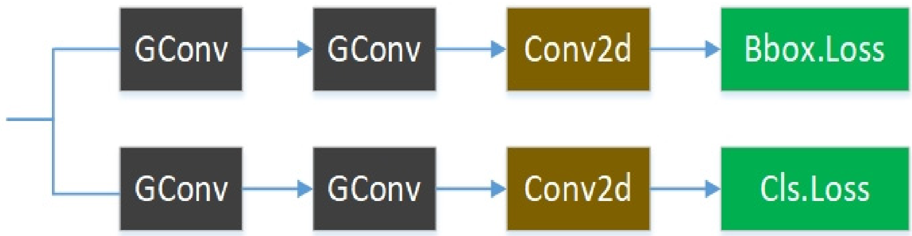


Fig 6. Structure of GDetect Network Model

2.5. SIOU-SoftNMS Loss Function

If the IOU of two objects is a true regression box, the network generates the IOU of the true regression box for the corresponding Bbox. The NMS method assumes that the first object has the highest score for a particular Bbox, and then records and calculates the IOU of that object with other Bboxes. If the IOU exceeds a predetermined threshold, the associated Bbox is removed, and the Bbox of an object whose IOU is larger than the object's IOU is removed, leaving a single Bbox. When the threshold is too low, some boxes are easily suppressed. When the threshold is too high, it is easy to cause false detection and cover up the suppression effect. In order to solve the problem of the NMS algorithm, SIOU-SoftNMS is used to multiply the obtained IOU by the Gaussian index as the weight before reordering and continuing the loop, as shown in formula (5)

$$s_i = s_i e^{-\frac{IOU(M, b_i)^2}{\sigma}} \tag{5}$$

Here, IOU(M,b_i) represents the intersection ratio of the prediction box M and the ith prediction box b_i with the maximum confidence score, and S_i represents the confidence of the ith prediction box.

Soft-NMS is suitable for handling the detection of dense areas in small targets, performing non-maximum suppression while taking into account the degree of overlap between scores and boundaries. Traditional IoU did not take into account the direction between the real frame and the predicted frame, resulting in a slow convergence speed. SIOU further considered the vector Angle between the real frame and the predicted frame, and redefined the relevant loss function, including Angle loss, distance loss, shape loss and IoU loss, as shown in Fig.7

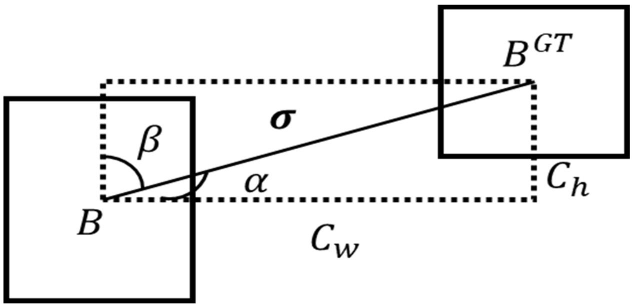


Fig 7. Structure of SIOU

Angle loss: Where c_h is the height difference between the center point of the real box and the prediction box. σ is the distance between the center point of the real box and the prediction box. In fact, $\arcsin(\frac{c_h}{\sigma})$ is equal to Angle α

$$\wedge = 1 - 2\sin^2(\arcsin(x) - \frac{\pi}{4}) \tag{6}$$

$$x = \frac{c_h}{\sigma} = \sin(\alpha) \tag{7}$$

$$\sigma = \sqrt{(b_{c_x}^{gt} - b_{c_x})^2 + (b_{c_y}^{gt} - b_{c_y})^2} \tag{8}$$

$$c_h = \max(b_{c_y}^{gt}, b_{c_y}) - \min(b_{c_y}^{gt}, b_{c_y}) \tag{9}$$

Distance loss: Where (c_w, c_h) are the width and height of the smallest external rectangle of the real box and the prediction box

$$\Delta = \sum_t = x, y^{(1-e^{-\gamma\rho_t})} = 2 - ex^{-\gamma\rho_x} - e^{-\gamma\rho_y} \tag{10}$$

$$\rho_x = (\frac{b_{c_x}^{g^t} - b_{c_x}}{W})^2, \rho_y = (\frac{b_{c_y}^{g^t} - b_{c_y}}{H})^2, \gamma = 2 - \Lambda \tag{11}$$

Shape loss: Where $(w, h), (w^{gt}, h^{gt})$ represents the width and height of the predicted box and the real box respectively. θ indicates how much attention is paid to controlling shape loss.

$$\Omega = (1 - e^{-w_w})^\theta + (1 - e^{-w_h})^\theta \tag{12}$$

IOU loss:

$$IOU = \frac{|B \cap B^{GT}|}{|B \cup B^{GT}|} \tag{13}$$

The final SIOU results are:

$$SIOU = 1 - IOU + \frac{1}{2}(\cos t_{distance} + \cos t_{shape}) \tag{14}$$

SIOU-SoftNMS solves the problem of missing detection when objects are very close and require very few additional hyperparameters. It only needs to multiply the resulting IOU by the Gaussian exponent, has no additional memory overhead, and its computational complexity is comparable to NMS.

3. Experimental Results and Analysis

3.1. Training Environment

In this study, Windows 10 operating system and Python3.8 programming language were used. In terms of hardware, the GPU model is NVIDIA GeForce RTX 4090, which runs 24GB of memory. In the software environment construction, the deep learning framework Pytorch 1.11 is used, and the GPU acceleration library of CUDA11.3 is used to improve the computing efficiency. For the setting of experimental parameters, the size of the input image was set to 640×640, the batch_size was set to 32, and the initial learning rate was 0.01. In the training process, after 300 cycles of iterative training, the best training weight file and the model with the highest accuracy were selected from the results of the training process.

3.2. Datasets Acquisition and Processing

The experimental datasets used in this study came from the Kaggle data set published online. 9 types of data were selected from tomato leaf diseases, including early blight, healthy blight, late blight, leaf potential disease, leaf mildew disease, Mosaic virus, Septoria, spider mite and yellow leaf roll virus, with a total of 3244 pieces. In addition, data sets of 3 types of citrus leaf pests and diseases were selected to enhance the data richness, including: citrus anthracnose, citrus leaf emery, and fatty spot macular disease, with a total of 767. The two datasets included a total of 12 leaf diseases, with a total of 4011 images. Then, Labelimg was used to annotate these 12 types of basic data sets in detail to ensure that each lesion was accurately labeled. Of this, 80% is used for dataset training and 20% for dataset validation. Part of the dataset sample is shown in Fig.8

Fig 8. Partial leaf disease data samples

3.3. Evaluation Index

In order to verify the robustness of the model and evaluate its performance, the accuracy rate P, recall rate R and average accuracy mean mAP were used as accuracy evaluation indexes [10]. Specifically, P represents the ratio of the prediction algorithm region to the actual detection region; R represents the proportion of accurately predicted categories to the total number of required categories; mAP calculates the accuracy of the overall sample in which the predicted box exceeds the actual box by 50%. The higher the accuracy, the less the model error check. The higher the recall rate, the less the model missed detection. Higher mAP values mean higher prediction accuracy. Among them, the calculation formula of accuracy, recall rate and average accuracy is as follows

$$mAP = \frac{\sum_{j=1}^{c}\left[\int_{0}^{1} P(R)\right]_{j}}{c} \tag{15}$$

$$\Pr ecision = \frac{TP}{TP + FP} \tag{16}$$

$$\operatorname{Re} call = \frac{TP}{TP + FN} \tag{17}$$

Where, TP represents the number of true positive samples in the test results; FP represents the number of false positive samples; FN indicates that the test result is not a true positive sample. In addition, c represents the total number of categories.

3.4. Optimized Structural Ablation Experiments of CGS-YOLOv10n Model

In order to verify the effectiveness of the CGS-YOLOv10n optimized in this paper, 8 groups of ablation experiments were conducted with different improvement points under the same conditions in the experimental environment. The results are shown in Table 1.

Table 1. Ablation results of YOLOv8n-HESS model

YOLOv10n	C2f_DBB	SIOU-SoftNMS	GDetect	P/%	R/%	mAP/%
√				91.5	90.1	94.7
√	√			92.2	90.0	95.5
√		√		91.8	89.8	94.9
√			√	91.6	89.7	95.0
√	√	√		90.7	89.1	95.1
√	√		√	92.7	90.4	94.8
√		√	√	89.3	91.2	95.2
√	√	√	√	93.8	91.6	95.8

The first set of experiments did not make changes to specific improvement points and served as baseline experiments. The second set of experiments added the C2f_DBB module, and the recall rate decreased slightly, and the accuracy and mAP both improved. The third group of experiments added the SIOU-SoftNMS loss function, and the three evaluation indexes were similar to the original model. The fourth group of experiments added the GDetect module, the recall rate decreased slightly, and the accuracy and mAP increased. A fifth set of experiments added C2f_DBB and SIOU-SoftNMS, and while accuracy and recall dropped by about 1%, mAP increased by 0.4%. In the sixth experiment, C2f_DBB and GDetect were added, and the three indexes increased slightly. The seventh experiment added SIOU-SoftNMS and GDetect in the first experiment, and the mAP value reached 95.2%. Compared with the original YOLOv10 algorithm model, the accurate value P increased by 2.3%, the recall rate increased by 1.5%, and the average precision increased by 1.1%, and the model achieved the best performance.

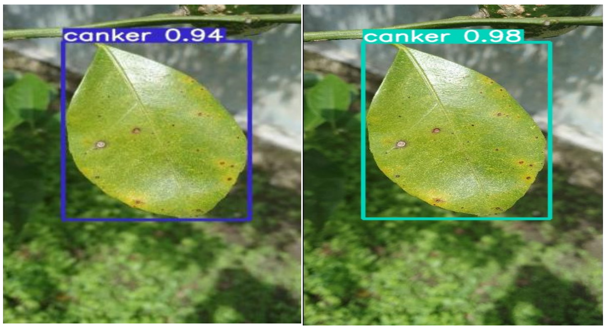
3.5. Comparative Analysis of Different Target Detection Models

In order to verify the effectiveness of the optimized CGS-YOLOv10n algorithm model, the improved model was compared with the YOLOv3 model, YOLOv5s model, YOLOv7 model, YOLOv8n model, YOLOv9-c model and YOLOv10n model for the leaf disease images in the test set. A comparative experiment was established as shown in Table 2.

Table 2. Performance comparison of different detection models

Algorithm model	P/%	R/%	mAP/%
YOLOv3	80.9	83.5	88.8
YOLOv5s	87.3	80.1	90.2
YOLOv7	88.6	93.1	85.0
YOLOv8n	89.7	88.2	93.5
YOLOv9-c	92.0	91.4	94.4
YOLOv10n	91.5	90.1	94.7
CGS-YOLOv10n	93.8	91.6	95.8

The experimental results show that by comparing multiple YOLO algorithm models with the improved algorithm proposed in this paper, the CGS-YOLOv10n algorithm model achieves better performance with higher accuracy and average accuracy. Fig.9 (a) is the inspection effect diagram of the original YOLOv10n algorithm model for leaf disease, and Fig.9(b) is the optimized model detection effect diagram of CGS-YOLOv10n algorithm. It can be seen that the algorithm model proposed in this paper has made progress on the basis of the good effect of the original model.



(a) (b)

Fig 9. Comparison of test results

4. Conclusion

Aiming at the problems of low accuracy and efficiency of various disease spot recognition on tomato and a few citrus leaves, this study proposed an optimized CGS-YOLOv10n model, which adopted C2f_DBB module in the global network, introduced multi-scale fusion loss function SIOU-SoftNMS, and utilized a new GDetect detection head module. Compared with the original YOLOv10n model, the accuracy rate P, recall rate R and average accuracy mean mAP are improved by 2.3 percentage points, 1.5 percentage points and 1.1 percentage points respectively. The algorithm in this paper can locate and identify the target more accurately, reduce the rate of missing and false detection effectively, and show superior performance. These positive results will provide strong support for the future development of target detection and provide more accurate and efficient solutions for practical applications. Future studies can continue to explore more effective model fusion strategies, data enhancement techniques, and more refined model parameter tuning to further improve the accuracy and efficiency of pest detection on crop leaf surfaces.

Acknowledgments

Doctoral Fund Initiation Project+222012312010.

References

[1] Jun L, Xuewei W .Tomato Diseases and Pests Detection Based on Improved Yolo V3 Convolutional Neural Network. [J]. Frontiers in plant science,2020,11898.

[2] Liu Jianhang, He Jianheng, Chen Haihua, et al. Tomato cluster detection Model based on improved YOLO v4 and ICNet [J]. Transactions of the Chinese Society for Agricultural Machinery, 2019,54(10):216-224+254.

[3] Wang Na, Chen Yong, Cui Yanrong, et al. Tomato leaf disease identification method based on improved lightweight YOLO v5n [J]. ,52 jiangsu agricultural science, 2024 (8) : 192-199. The DOI: 10.15889 / j.i SSN. 1002-1302.2024. 08.026.

[4] Xu Yue, Chen Lin. The tomato leaf disease detection algorithm based on improved YOLO v8 [J]. Journal of jiangsu agricultural science, 2024,52 (17): 192-200. The DOI: 10. 15889 / j.i SSN. 1002-1302.2024.17.026.

[5] JIN Xuemeng, Liang Xiyin, Deng Pengfei. Lightweight daylily classification detection model based on improved YOLOv10 [J/OL]. Wisdom agriculture (both in English and Chinese), 1-12 [2024-10-28]. http://kns.cnki.net/ kcms/detail/ 10. 1681. S. 20241016.0721.002.html.

[6] Liao J, He X ,Liang Y , et al. A Lightweight Cotton Verticillium Wilt Hazard Level Real-Time Assessment System Based on an Improved YOLOv10n Model[J].Agriculture, 2024, 14 (9):1617-1617.

[7] Xue P, Zhang Z .EBFF-YOLO: enhanced bimodal feature fusion network for UAV image object detection[J].Signal, Image and Video Processing,2024,18(10):6591-6600.

[8] Yanli S, Yi J ,Xianhe Z .FocusDet: an efficient object detector for small object[J].Scientific Reports,2024,14(1):10697-10697.

[9] Long Yang, Xiao Xiaoling. Improved Metal Surface Defect Detection Model of YOLOv8 [J]. Manufacturing Technology & Machine Tool,2024,(08):187-194.DOI:10.19287/ J.MTT. 1005-2402.2024.08.027.

[10] Sportelli M ,Apolo A E O ,Fontanelli M , et al. Evaluation of YOLO Object Detectors for Weed Detection in Different Turfgrass Scenarios[J].Applied Sciences,2023,13(14).