



Article

Real-Time Detection of Apple Leaf Diseases in Natural Scenes Based on YOLOv5

Huishan Li, Lei Shi , Siwen Fang and Fei Yin * 

College of Information and Management Science, Henan Agricultural University, Zhengzhou 450046, China

* Correspondence: yin.feih@henau.edu.cn

Abstract: Aiming at the problem of accurately locating and identifying multi-scale and differently shaped apple leaf diseases from a complex background in natural scenes, this study proposed an apple leaf disease detection method based on an improved YOLOv5s model. Firstly, the model utilized the bidirectional feature pyramid network (BiFPN) to achieve multi-scale feature fusion efficiently. Then, the transformer and convolutional block attention module (CBAM) attention mechanisms were added to reduce the interference from invalid background information, improving disease characteristics' expression ability and increasing the accuracy and recall of the model. Experimental results showed that the proposed BTC-YOLOv5s model (with a model size of 15.8M) can effectively detect four types of apple leaf diseases in natural scenes, with 84.3% mean average precision (mAP). With an octa-core CPU, the model could process 8.7 leaf images per second on average. Compared with classic detection models of SSD, Faster R-CNN, YOLOv4-tiny, and YOLOx, the mAP of the proposed model was increased by 12.74%, 48.84%, 24.44%, and 4.2%, respectively, and offered higher detection accuracy and faster detection speed. Furthermore, the proposed model demonstrated strong robustness and mAP exceeding 80% under strong noise conditions, such as exposure to bright lights, dim lights, and fuzzy images. In conclusion, the new BTC-YOLOv5s was found to be lightweight, accurate, and efficient, making it suitable for application on mobile devices. The proposed method could provide technical support for early intervention and treatment of apple leaf diseases.

Keywords: smart agriculture; detection of apple leaf diseases; YOLOv5; transformer; CBAM



Citation: Li, H.; Shi, L.; Fang, S.; Yin, F. Real-Time Detection of Apple Leaf Diseases in Natural Scenes Based on YOLOv5. *Agriculture* **2023**, *13*, 878. <https://doi.org/10.3390/agriculture13040878>

Academic Editor: Filipe Neves Dos Santos

Received: 15 March 2023

Revised: 10 April 2023

Accepted: 14 April 2023

Published: 15 April 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

As one of the top four popular fruits in the world, apple is highly nutritious and provides significant medicinal value [1]. In China, apple production has expanded, making it the world's largest apple producer. However, a variety of diseases hamper the healthy growth of apple, seriously affecting the quality and yield of apple and causing significant economic losses. According to statistics, there are approximately 200 types of apple diseases, most of which occur in apple leaf areas. Therefore, to ensure the healthy development of the apple planting industry, accurate and efficient leaf disease identification and control measures are needed [2].

In traditional disease identification techniques, fruit farmers and experts rely on visual examination based on their experience, a method which is inefficient and highly subjective. With the advance of computer and information technology, image recognition technology has been gradually applied in agriculture. Many researchers have applied machine vision algorithms to extract features such as color, shape, and texture from disease images and input them into specific classifiers to accomplish plant disease recognition tasks [3]. Zhang et al. [4] processed apple disease images using HSI, YUV, and gray models; then, the authors extracted features using genetic algorithms and correlation based-feature selection, and ultimately discriminated apple powdery mildew, mosaic, and rust diseases using an SVM classifier with an identification accuracy of more than 90%. However, the complex image background and the feature extraction, dominated by strong experience,

make the labor and time costs much higher, as well as making the system difficult to promote and popularize.

In recent years, deep learning convolutional neural networks have been widely used in agricultural intelligent detection, with faster detection speeds and higher accuracy compared to traditional machine vision techniques [5]. There are two types of target detection models; the first is the two-stage detection algorithm represented by R-CNN [6] and Faster R-CNN [7]. Xie et al. [8] used an improved Faster R-CNN detection model for real-time detection of grape leaf diseases, introducing three modules (Inception v1, Inception-ResNet-v2, and SE) in the model, and mean average precision (mAP) achieved 81.1%. Deng et al. [9] proposed a method for large-scale detection and localization of pine wilt disease using unmanned remote sensing and artificial intelligence technology, and a series of optimizations to improve detection accuracy to 89.1%. Zhang et al. [10] designed a Faster R-CNN (MF³R-CNN) model with multiple feature fusion for soybean leaf disease detection, achieving an average accuracy of 83.34%. Wang et al. [11] used the RFCN ResNet101 model to detect potato surface defects and achieved an accuracy of 95.6%. This two-stage detection model was capable of identifying crop diseases, but its large network model and slow detection speed made it difficult to apply in real planting industry.

Another type of target detection algorithm is the one-stage algorithm represented by SSD [12] and YOLO [13–16] series. Unlike the two-stage detection algorithm, it does not require the generation of candidate frames. By converting the boundary problem into a regression problem, features extracted from the network are used to predict the location and class of lesions. Due to its high accuracy, fast speed, short training time, and low computational requirement, it is more suitable for agricultural applications. Wang et al. [17] used the SSD-MobileNet V2 model for the detection of scratches and cracks on the surface of litchi, which eventually achieved 91.81% mAP and 102 frame per second (FPS). In the experiments of Chang-Hwan et al. [18], a new attention-enhanced YOLO model was proposed for identifying and detecting plant foliar diseases. Li et al. [19] improved the CSP, feature pyramid networks (FPN), and non-maximum suppression (NMS) modules in YOLOv5 to detect five vegetable diseases and obtained 93.1% mAP, effectively reducing missing and false detections caused by complex background. In complex orchard environments, Jiang et al. [20] proposed an improved YOLOX model to detect sweet cherry fruit ripeness. In improving the model, mAP and recall were both improved by 4.12% and 4.6%, respectively, which effectively solved the interference caused by fruit overlaps and shaded branches and leaves. Li et al. [21] used the improved YOLOv5n model to detect cucumber diseases in natural scenes and achieved higher detection accuracy and speed. While the development of intelligent crop disease detection using one-stage detection algorithms has matured, less research has been carried out for apple leaf disease detection. Small datasets and simple image backgrounds pose problems for most existing studies. Consequently, it is crucial to develop an apple leaf disease detection model with high recognition accuracy and fast detection speed for mobile devices with limited computing power.

Considering the complex planting environment in apple orchards and the various shapes of lesions, this study proposed the use of an improved target detection algorithm based on YOLOv5s. The proposed algorithm aimed to reduce false detections caused by multi-scale lesions, dense lesions, and inconspicuous features in apple leaf disease detection tasks. As a result, the accuracy and efficiency of the model could be enhanced to provide essential technical support for apple leaf disease identification and intelligent orchard management.

2. Materials and Methods

2.1. Materials

2.1.1. Data Acquisition and Annotation

In this study, three datasets were used to train and evaluate the proposed model: the Plant Pathology Challenge 2020 (FGVC7) [22] dataset, the Plant Pathology Challenge 2021 (FGVC8) [23] dataset, and the PlantDoc [24] dataset.

FGVC7 and FGVC8 [22,23] consist of apple leaf disease images used in the Plant Pathology Fine-Grained Visual Categorization competition hosted by Kaggle. The images were captured by Cornell AgriTech using Canon Rebel T5i DSLR and smartphones, with a resolution of 4000×2672 pixels for each image. There are four kinds of apple leaf diseases, namely rust, frog-eye leaf spot, powdery mildew, and scab. These diseases occur frequently and cause significant losses in the quality and yield of apples. Sample images of the dataset are shown in Figure 1.



Figure 1. FGVC7 and FGVC8 disease images. (a) Frog-eye leaf spot; (b) Rust; (c) Scab; (d) Powdery mildew.

PlantDoc [24] is a dataset of non-laboratory images constructed by Davinder Singh et al. in 2020 for visual plant disease detection. It contains 2598 images of plant diseases in natural scenes, involving 13 species of plants and as many as 17 diseases. Most of the images in PlantDoc have low resolution, large noise, and an insufficient number of samples, making detection more difficult. In this study, apple rust and scab images were used to enhance and validate the generalization of the proposed model. Examples of disease images are shown in Figure 2.



Figure 2. PlantDoc disease images. (a) Rust; (b) Scab.

From the collected datasets, we selected (1) images with light intensity varying with the time of day, (2) images capture using different shooting angles, (3) images with different disease intensities, and (4) images from different disease stages to ensure the richness and diversity of the dataset. Finally, a total of 2099 apple leaf disease images were selected. Labelling software was used to label the images with categories including disease type, center coordinates, width, and height of each disease spots. In total, we annotated 10,727 lesion instances, and annotations are shown in Table 1. The labeled dataset was randomly divided into training and test sets at a ratio of 8:2. This dataset was called ALDD (apple leaf disease data) and was used to train and test the model.

Table 1. Label distribution of ALDD.

Disease Type	Number of Images	Number of Labeled Instances
Scab	498	4722
Frog-eye leaf spot	600	3091
Rust	502	2166
Powdery mildew	499	748
Total number	2099	10,727

2.1.2. Data Enhancement

The actual apple orchard in a complex environment contains many disturbances and the currently selected data is far from sufficient. To enrich the image dataset, mosaic image enhancement [16] and online data enhancement were chosen to expand the dataset. Mosaic image enhancement involves a random selection of 4 images from the training set, which are finally combined into one image after rotation, scaling, and hue adjustment. This approach not only enriches the image background and increases the number of instances, but also indirectly boosts the batch size. This accelerates model training and is favorable to improving small target detection performance. Online augmentation is the use of data augmentation in model training, which ensures the invariance of the sample size and the diversity of the overall sample and improves the model's robustness by continuously expanding the sample space. Mainly includes alterations to hue, saturation, brightness transformation, translation, rotation, flip, and other operations. The total number of the dataset is constant; however, the amount of data input to each epoch is changing, and it is more conducive to fast convergence of the model. Examples of enhanced images are shown in Figure 3.

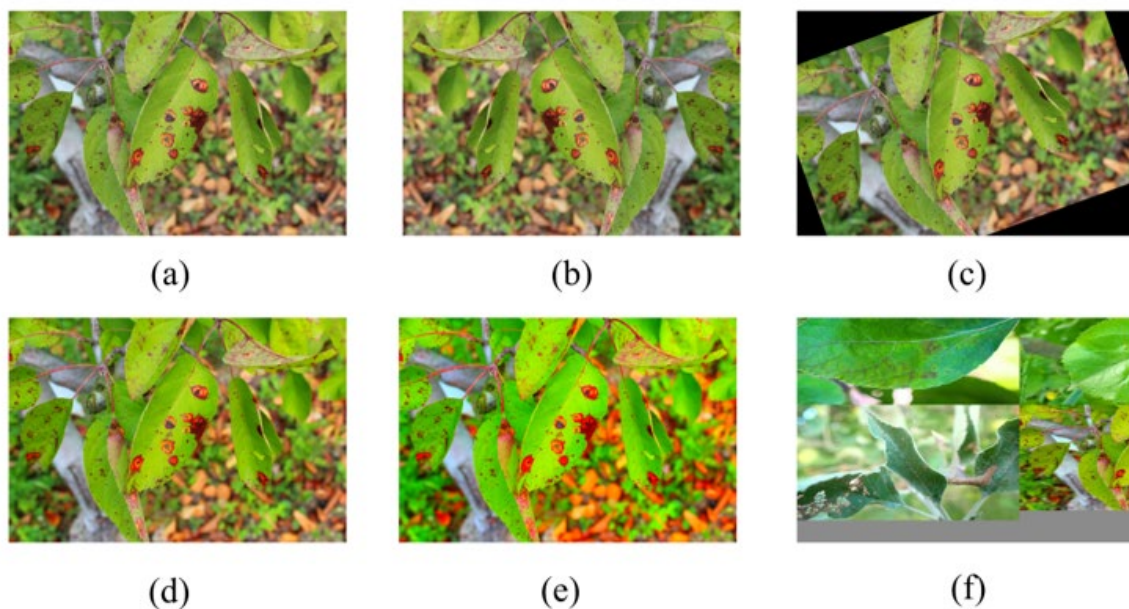


Figure 3. Original and enhanced images. (a) Original; (b) Flip horizontal; (c) Rotation transformation; (d) Hue enhancement; (e) Saturation enhancement; (f) Mosaic enhancement.

2.2. Methods

2.2.1. YOLOv5s Model

Depending on the network depth and feature map width, YOLOv5 can be divided into YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x [25]. As the depth and width increase, the number of layers of the network increases as well as the structure becomes more complex. In order to meet the requirements of lightweight deployment and real-time detection, reduce storage space occupied by the model and improve the identification speed, YOLOv5s was selected as the baseline model in this study.

The YOLOv5s was composed of four parts: input, backbone, neck, and prediction. The input section included mosaic data enhancement, adaptive calculation of the anchor box, and adaptive scaling of images. The backbone module performed feature extraction and consisted of four parts: focus, CBS, C3, and spatial pyramid pooling (SPP). There were two types of C3 [26] modules in YOLOv5s for backbone and neck, as shown in Figure 4. The first one used the residual units at the backbone layer, while the second one did not. SPP [27] performed the maximum pooling of feature maps using convolutional kernels of

different sizes in order to fuse multiple sense fields and generate semantic information. The neck layer used a combination of (FPN) [28] and path aggregation networks (PANet) [29] to fuse the image features. The prediction included three detection layers, corresponding to 20×20 , 40×40 , and 80×80 feature maps, respectively, for detecting large, medium, and small targets. Finally, the distance between the predicted boxes and the true boxes was calculated using the complete intersection over union (CIoU) [30] loss function, and the NMS was applied to remove the redundant boxes and retain the detection boxes with the highest confidence. The YOLOv5s network model is shown in Figure 4.

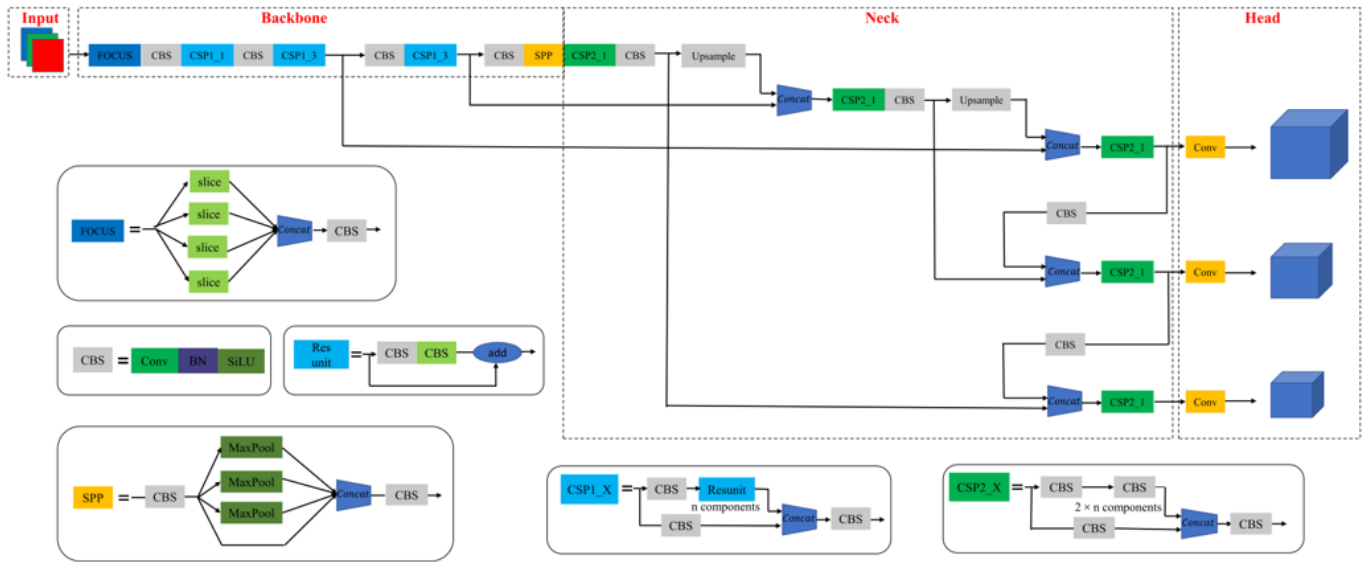


Figure 4. YOLOv5s method architecture diagram.

2.2.2. Bidirectional Feature Pyramid Network

The YOLOv5s combines FPN and PANet for multi-scale feature fusion, with FPN enhancing semantic information in a top-down fashion and PANet enhancing location information from the bottom up. This combination enhances the feature fusion capability of the neck layer. However, when fusing input features at different resolutions, the features are simply summed and their contributions to the fused output features are usually inequitable. To address this problem, Tan et al. [31] developed the BiFPN based on efficient bidirectional cross-scale connections and weighted multiscale feature fusion. The BiFPN introduced learnable weights in order to learn the importance of different input features, while top-down and bottom-up multi-scale feature fusion was applied iteratively. The structure of BiFPN is shown in Figure 5.

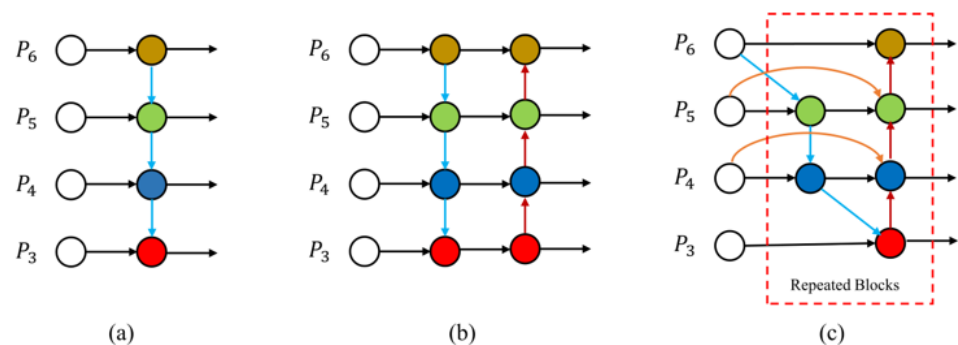


Figure 5. BiFPN network structure diagram, where (a) FPN introduces a top-down path to fuse multi-scale features from P_3 to P_6 ; (b) PANet adds an additional bottom-up path on top of the FPN; (c) BiFPN removes redundant nodes and adds additional connections on top of PANet.

The BiFPN removes the node with only one input edge because it does not perform feature fusion. The contribution to the network aim of fusing different features is minimal, and so it is removed and the bidirectional network is simplified. Additionally, an extra edge is added between the input and output nodes that are at the same layer to obtain higher-level fusion features through iterative stacking. The BiFPN introduces a simple and efficient weighted feature fusion mechanism by adding a learnable weight that assigns different degrees of importance to feature maps of different resolutions. The formulas are shown in (1) and (2):

$$P_i^{td} = Conv \left(\frac{w_1 \cdot P_i^{in} + w_2 \cdot Resize(P_{i+1}^{in})}{w_1 + w_2 + \epsilon} \right) \quad (1)$$

$$P_i^{out} = Conv \left(\frac{w'_1 \cdot P_i^{in} + w'_2 \cdot P_i^{td} + w'_3 \cdot Resize(P_{i-1}^{out})}{w'_1 + w'_2 + w'_3 + \epsilon} \right) \quad (2)$$

where P_i^{in} is the input feature of layer i , P_i^{td} is the intermediate feature on the top-down pathway of layer i , P_i^{out} is the output feature on the bottom-up pathway of layer i , w is the learnable weight, $\epsilon = 0.0001$ is a small value to avoid numerical instability, *Resize* is a downsampling or upsampling operation, and *Conv* is a convolution operation.

The neck layer with BiFPN added a fusion of multi-scale features to provide powerful semantic information to the network. It helped to detect apple leaf diseases of different sizes and alleviated the network's inaccurate identification of overlapping and fuzzy targets.

2.2.3. Transformer Encoder Block

There was a high density of lesions on apple leaves. In order to avoid the problem that the number of lesions and background information increased after mosaic data enhancement, which caused the inability to accurately locate the area where the diseases, the transformer [32] attention mechanism was added to the end of the backbone layer. The transformer module was employed to capture global contextual information and establish long-range dependencies between feature channels and disease targets. The transformer encoder module used a self-attentive mechanism to explore the feature representation capability and had excellent performance in highly dense scenarios [33]. The self-attention mechanism was designed based on the principles of human vision and allocated resources according to the importance of visual objects. The self-attentive mechanism had a global sensory field, which modeled long-range contextual information, captured rich global semantic information, and assigned different weights to different semantic information to make the network focus more on key information [34]. It was calculated as (3), and contained three basic elements: query, key, and value, denoted by Q , K , and V , respectively.

$$Attention(Q, K, V) = softmax \left(\frac{QK^T}{\sqrt{d_k}} \right) V \quad (3)$$

where d_k is the number of input feature map channel sequences, using normalized data to avoid gradient increment.

Each transformer encoder is composed of a multi-head attention and a feed-forward neural network. The structure of multi-head attention mechanism is shown in Figure 6. It differs from the self-attentive mechanism in that the self-attentive mechanism uses only one set of Q , K , and V values, while it uses multiple sets of Q , K , and V values to compute and stitch multiple matrices together. The different linear transformations feature different vector spaces, which can help the current code to focus on the current pixels and acquire semantic information about the context [35]. The multi-head attention mechanism enhances the ability to extract disease features by capturing long-distance dependent information without increasing the computational complexity and improves the model's detection performance.

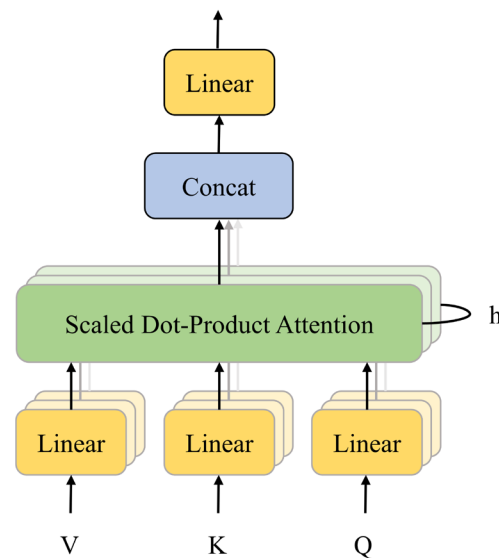


Figure 6. Structure of multi-headed attention mechanism.

2.2.4. Convolutional Block Attention Module

Determining the disease species relies more on local information in the feature map, while the localization of lesions is more concerned with the location information. This model used the CBAM [36] attention mechanism in the improved YOLOv5s to weight the features in space and channels and enhance the model’s attention to local and spatial information.

As shown in Figure 7, the CBAM contained two sub-modules: the channel attention module (CAM) and the spatial attention module (SAM), for spatial and channel attention, respectively. The input feature map $F \in \mathbb{R}^{C \times H \times W}$ was first passed through the one-dimensional convolution operation $M_c \in \mathbb{R}^{C \times 1 \times 1}$ of the CAM, and the convolution result was multiplied with the input features. The output result of CAM was then used as input, the two-dimensional convolution operation $M_s \in \mathbb{R}^{1 \times H \times W}$ of the SAM was performed, and then the result was multiplied with the CAM output to obtain the final result. The calculation formulas are as (4) and (5).

$$F' = M_c(F) \otimes F \tag{4}$$

$$F'' = M_s(F') \otimes F' \tag{5}$$

where F denotes the input feature map, M_c denotes the one-dimensional convolution operation of CAM, M_s denotes the two-dimensional convolution operation of SAM, and \otimes denotes element multiplication.

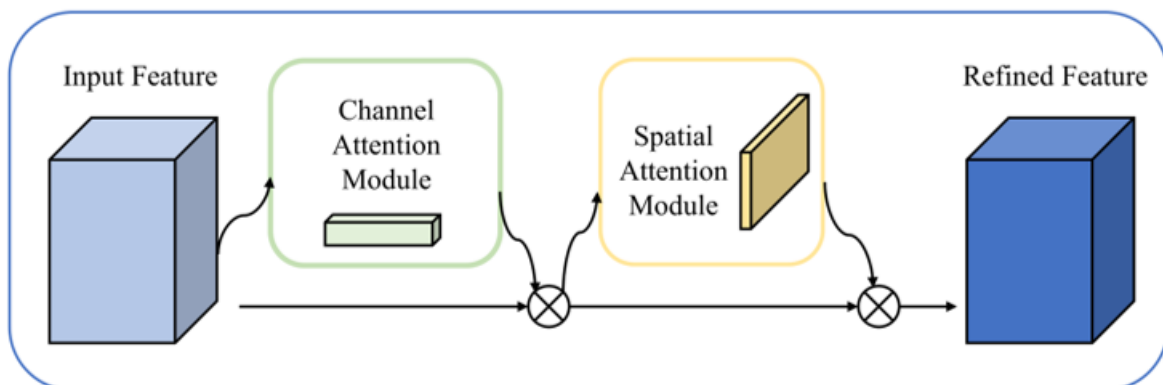


Figure 7. Convolutional block attention module (CBAM).

The CAM in CBAM focused on the weights of different channels and multiplied the channels with the corresponding weights to increase attention to important channels. The feature map F of size $H \times W \times C$ was averaged and maximally pooled to obtain two $1 \times 1 \times C$ channel mappings, respectively, and then a two-layer shared multi-layer perception (MLP) operation was performed. The two outputs were summed element by element, and then a sigmoid activation function was applied to output the final result. The calculation process is shown in Equation (6).

$$M_c(F) = \sigma(MLP(AvgPool(F)) + MLP(MaxPool(F))) \tag{6}$$

As shown in Equation (7), the SAM was more concerned with the location information of the lesions. The CAM output was averaged and maximally pooled to obtain two $H' \times W' \times 1$ channel maps. The final result was obtained by concatenating the two feature maps, followed by a 7×7 convolution operation and a Sigmoid activation function.

$$M_s(F) = \sigma(f^{7 \times 7}([AvgPool(F); MaxPool(F)])) \tag{7}$$

2.2.5. BTC-YOLOv5s Detection Model

Based on the original advantages of the YOLOv5s model, this study proposed using an improved BTC-YOLOv5s algorithm for detecting apple leaf diseases. While ensuring the speed of the procedure, it improved the accuracy of identifying apple leaf diseases in a complex environment. The proposed algorithm was improved mainly in three parts: the BiFPN, transformer, and CBAM attention mechanism. Firstly, the CBAM module was added in front of the SPP in the YOLOv5s backbone layer to highlight useful information and suppress useless information in the disease detection task, thereby improving the model’s detection accuracy. Secondly, the C3 was replaced with the C3TR module with transformer and improved the ability to extract apple leaf disease features. Thirdly, we replaced the concat layer with the BiFPN layer, and a path from the 6th layer was added to the 20th layer. The features generated by the backbone at the same layer were bidirectionally connected with the features generated by the FPN and the PANet to provide stronger information representation capability. Figure 8 shows the overall framework of the BTC-YOLOv5s model for this study.

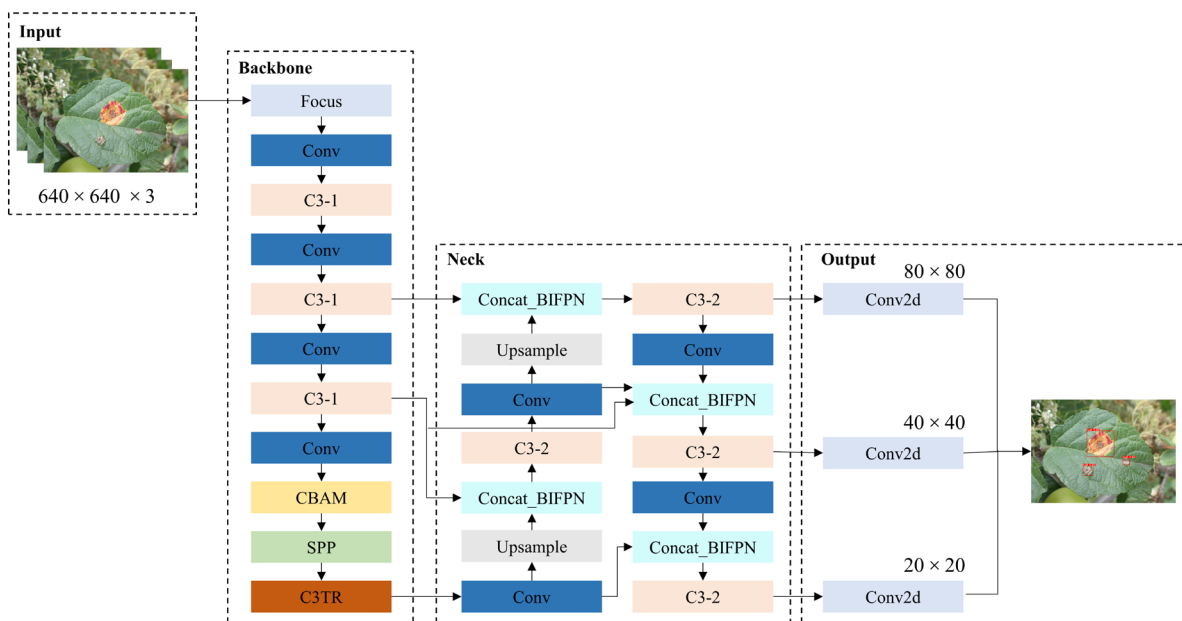


Figure 8. BTC-YOLOv5s model structure diagram.

2.3. Experimental Equipment and Parameter Settings

The model was trained and tested on a Linux system running under the PyTorch 1.10.0 deep learning framework, using the following device specifications: Intel(R) Xeon(R) E5-2686 v4 @ 2.30 GHz processor, 64 GB of memory, and NVIDIA GeForce RTX3090 graphics card with 24 GB of video memory. The software was executed on cuda 11.3, cudnn 8.2.1, and python 3.8.

During training, the initial learning rate was set to 0.01, and the cosine annealing strategy was employed to decrease the learning rate. Additionally, the neural network parameters were optimized using the stochastic gradient descent (SGD) method, with a momentum value of 0.937 and a weight decay index score of 0.0005. The training epoch was 150, the image batch size was set to 32, and the input image resolution was uniformly adjusted to 640×640 . Table 2 shows the tuned training parameters.

Table 2. Model training parameters.

Parameters	Values
Input size	640×640
Batch size	32
Epoch	150
Initial learning rate	0.01
Optimizer	SGD
Momentum	0.937
Weight decay	0.0005

2.4. Model Evaluation Metrics

The evaluation metrics are divided into two aspects: performance assessment and complexity assessment. The model performance evaluation metrics include precision, recall, mAP, and F1 score. The model complexity evaluation metrics include model size, floating point operations (FLOPs), and FPS, which evaluate the computational efficiency and image processing speed of the model.

Precision is the ratio of the correctly predicted positive samples to the total number of samples predicted as positive and is used to measure the classification ability of a model, while the recall measures the ratio of the correctly predicted positive samples to the total number of positive samples. The AP is the integral of precision and recall, and the mAP is the average of AP, which reflects the overall performance of the model for target detection and classification. F1 score is the harmonic mean of precision and recall, and it uses both precision and recall to evaluate the performance of the model. The calculation formulas are shown in Equations (8)–(12).

$$Precision = \frac{TP}{TP + FP} \quad (8)$$

$$Recall = \frac{TP}{TP + FN} \quad (9)$$

where TP is the number of positive samples with correct detection, FP is the number of positive samples with incorrect detection, and FN is the number of negative samples with incorrect detection.

$$AP = \int_0^1 P(R) dR \quad (10)$$

$$mAP = \frac{\sum_{i=1}^n AP_i}{n} \quad (11)$$

where n is the number of disease species.

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (12)$$

The model size refers to the amount of memory required for storing the model. FLOPs is used to measure the complexity of the model, which is the total number of multiplication and addition operations performed by the model. The lower the FLOPs value, the less computation is required for model inference, and the faster model computation will be. The formula for FLOPs is shown in Equations (13) and (14). The FPS indicates the number of pictures processed per second by the model, which can assess the processing speed and is crucial for real-time disease detection. Considering that the model can be implemented on mobile devices with low computational cost, an octa-core CPU without a graphics card was selected to run the test.

$$\text{FLOPs(Conv)} = (2 \times C_{in} \times K^2 - 1) \times W_{out} \times H_{out} \times C_{out} \quad (13)$$

$$\text{FLOPs(Liner)} = (2 \times C_{in} - 1) \times C_{out} \quad (14)$$

where C_{in} represents the input channel, C_{out} represents the output channel, K represents the convolution kernel size, and W_{out} and H_{out} represent the width and height of the output feature map, respectively.

3. Results

3.1. Performance Evaluation

The proposed BTC-YOLOv5s model was validated using the constructed ALDD test set. Additionally, the same optimized parameters were used to compare results with YOLOv5s baseline model. As shown in Table 3, the improved model achieved similar AP scores for frog eye leaf spots as the original model, while significantly improving the detection performance for the other three diseases. Notably, scab disease, with its irregular lesion shape, was the most issue to detect, and the improved model achieved a 3.3% increase in AP, which was the largest improvement. These results indicated that the proposed model effectively detected all four diseases with improved accuracy.

Table 3. Comparison of detection results of YOLOv5s and BTC-YOLOv5s.

Models	AP(%)				mAP@0.5(%)	
	Frog	Scab	Powdery	Rust	Spare	Dense
YOLOv5s	93	60.3	88.8	88.7	85.6	80.7
BTC-YOLOv5s	92.9	63.6	90.2	90.3	87.3	81.4

Figure 9 shows evaluation results of precision, recall, mAP@0.5, and mAP@0.5:0.95 for the baseline model YOLOv5s and the improved model BTC-YOLOv5s trained with 150 epochs.

In Figure 9, it is displayed that the precision and recall curves fluctuated within a narrow range after 50 epochs, but that the BTC-YOLOv5s curve remained consistently above the baseline model curve. From the mAP@0.5 curve, it can be seen that the mAP@0.5 curve of the improved model intersected with the baseline model at around 60 epochs. Although the mAP@0.5 of the baseline model increased rapidly in the early stage, the BTC-YOLOv5s model improved steadily in the later stage and showed better results. The mAP@0.5:0.95 curve also demonstrated a similar behavior.

As apple leaf diseases were small and densely distributed, for further verification of the BTC-YOLOv5s model's accuracy, the test sets were divided into two groups based on lesion density, namely sparse distribution and dense distribution of lesions. We compared the detection results of the baseline model and the improved model. The mAP@0.5 of BTC-YOLOv5s model for sparse and dense lesions images was 87.3% and 81.4%, respectively, which was 1.7% and 0.7% higher than that of the baseline model.

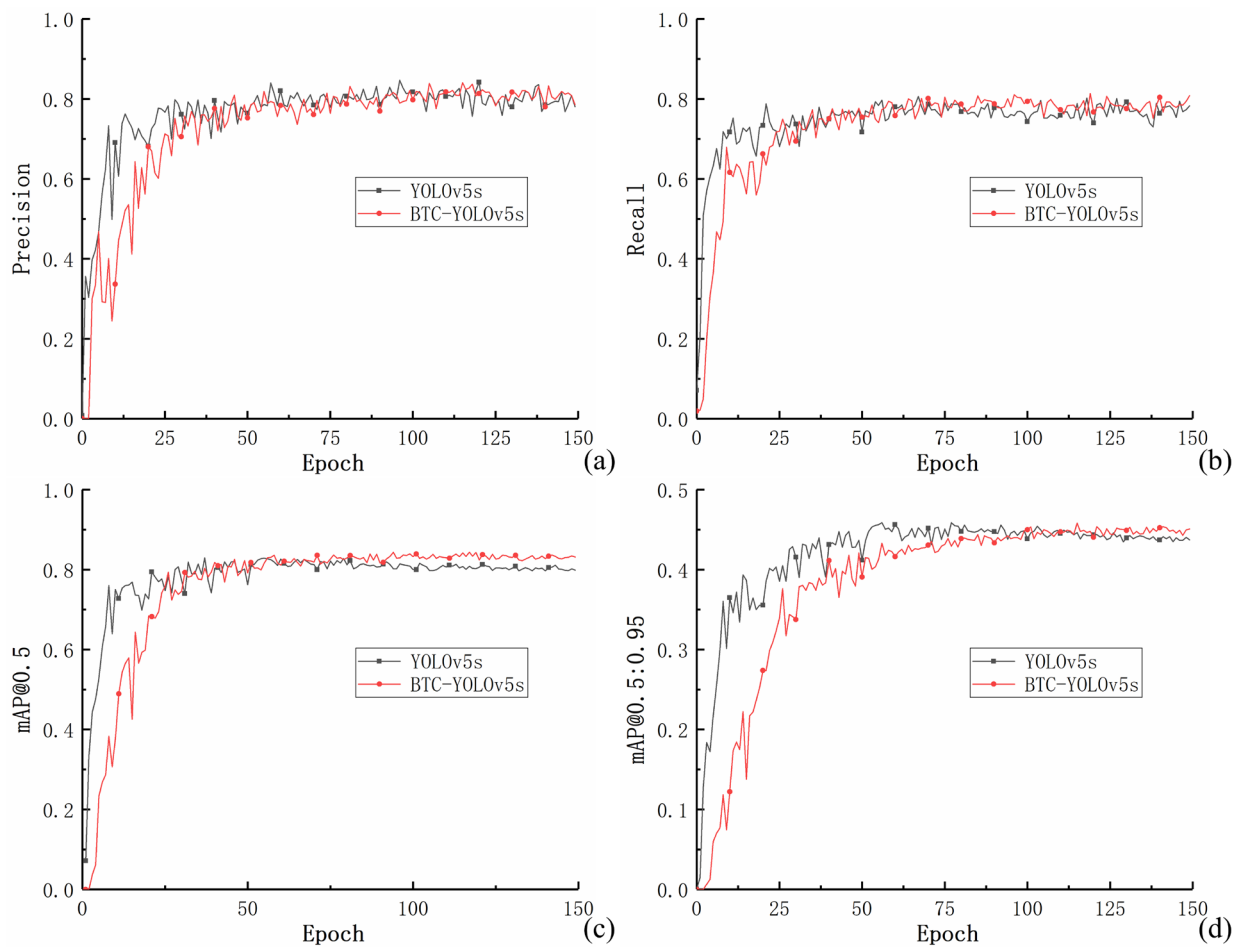


Figure 9. Evaluation metrics of different models, where (a) is a comparison of precision curves before and after model improvement; (b) comparison of recall curves before and after model improvement; (c) comparison of mAP@0.5 curves before and after model improvement; (d) comparison of mAP@0.5:0.95 curves before and after model improvement.

As shown in Figure 10, yellow circles represent missed detections and red circles represent false detections. It can be seen that, irrespective of whether the disease is sparse or dense, the baseline model YOLOv5s missed small or blurred lesions (the first row of images in Figure 10a,b). However, the improved model resolved this issue and detected small lesions or diseases on the leaves that were not in the focus range (the second row of images in Figure 10a,b). Additionally, the BTC-YOLOv5s model had higher confidence levels. The baseline model also mistakenly detected the non-diseased parts such as apples, background, and other irrelevant objects (Figure 10(a3,b1)), and there was a false detection whereby the scab was mistakenly detected as rust (Figure 10(b5)). The improved model could concentrate more on diseases and extract the gap characteristics between different diseases at a deeper level to avoid the above errors. Furthermore, the lesions of frog-eye leaf spot, scab, and rust were small, dense, and distributed in different parts of the leaves, while powdery mildew typically affected the whole leaf. This led to the scale of the model detection box changing from large to small, and the proposed model was able to adapt well to the scale changes of different diseases.

Therefore, the BTC-YOLOv5s model could not only adapt to the detection of different disease distributions but could also adapt to the changes in apple leaf diseases with different scales and characteristics, showing excellent detection results.

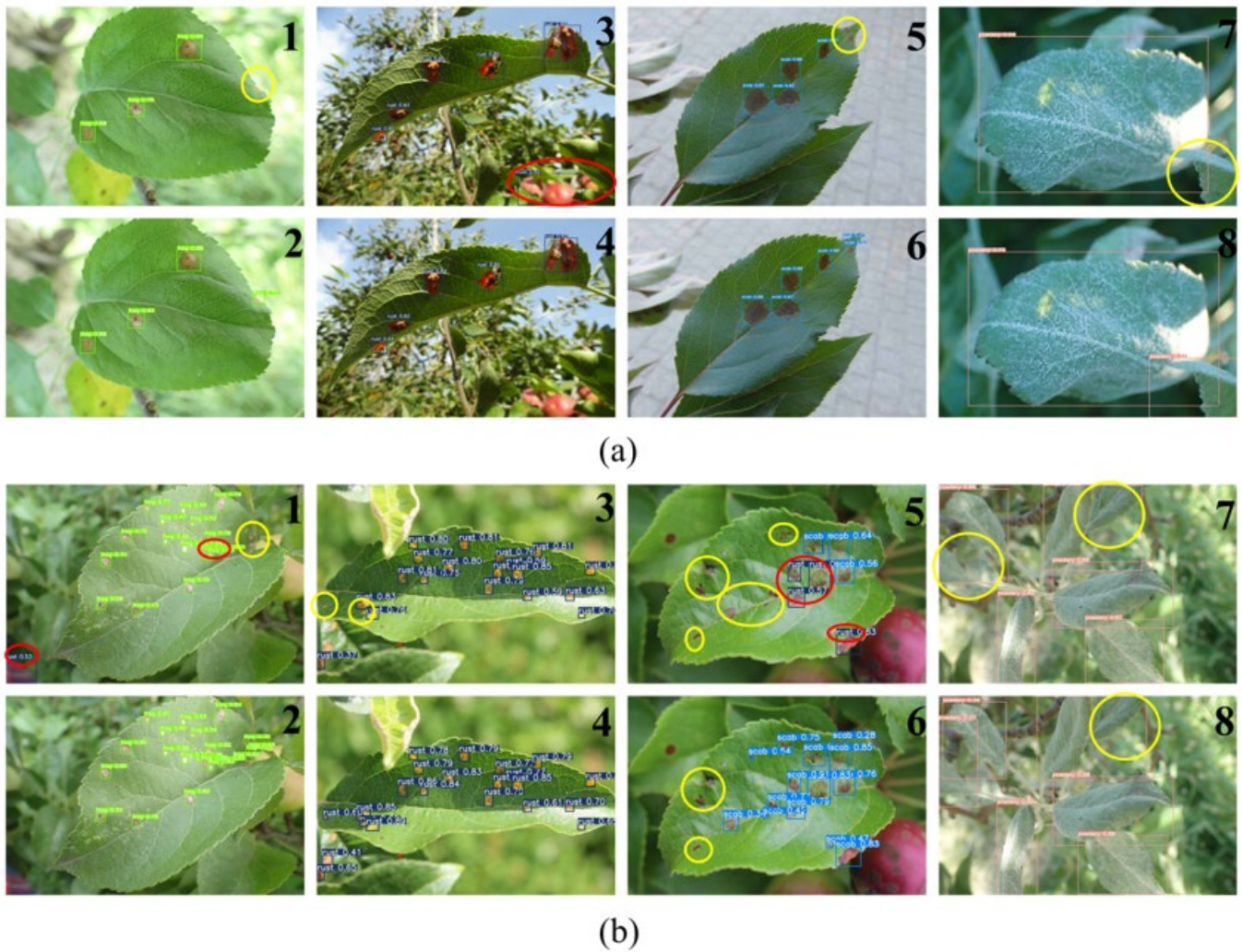


Figure 10. Comparison of detection effect of lesion (sparse and dense) before and after model improvement. (a) Sparse distribution; (b) Dense distribution. Where yellow circles represent missed detections and red circles represent false detections. Lines 1 and 3 are the YOLOv5s baseline model, and lines 2 and 4 are the improved BTC-YOLOv5s model. Numbers 1 and 2 are frog-eye leaf spot, numbers 3 and 4 are rust, numbers 5 and 6 are scab, and numbers 7 and 8 are powdery mildew.

3.2. Results of Ablation Experiments

This study verified the effectiveness of different optimization modules via ablation experiments. We constructed several improved models by adding the BiFPN module (BF), transformer module (TR), and CBAM attention module sequentially to the baseline model YOLOv5s and compared the results on the same test data. The experimental results are shown in Table 4.

In Table 4, the precision and mAP@0.5 of the baseline model YOLOv5s were 78.4% and 82.7%. By adding three optimization modules, namely the BiFPN module, transformer module, and CBAM attention module, both precision and mAP@0.5 were improved compared to the baseline model. Specifically, the precision increased by 3.3%, 3.3%, and 1.1%, respectively, and the mAP@0.5 increased by 0.5%, 1%, and 0.2%, respectively. The final combination of all three optimization modules achieved the best results, with precision, mAP@0.5 and mAP@0.5:0.95 all reaching the highest values, which were 5.7%, 1.6%, and 0.1% higher than those of the baseline model, respectively. By fusing cross-channel information with spatial information, the CBAM attention mechanism focused on important features while suppressing irrelevant ones. Additionally, the transformer module used the

self-attention mechanism to establish a long-range feature channel with the disease features. The BiFPN module fused the above features across scales to improve the identification of overlapping and fuzzy targets. As a result of the combination of three modules, the BTC-YOLOv5s model achieved the best performance.

Table 4. Results of ablation experiments.

Models	Precision (%)	Recall (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)
YOLOv5s	78.4	79.7	82.7	45.8
YOLOv5s + BF	81.7	78.4	83.2	45.3
YOLOv5s + CBAM	81.7	79.7	83.7	45.7
YOLOv5s + TR	79.5	78.9	82.9	45.6
YOLOv5s + BF + CBAM	81	81	84.3	44.9
YOLOv5s + BF + TR	83.5	77.6	83	45.1
YOLOv5s + BF + TR + CBAM (proposed)	84.1	77.3	84.3	45.9

Where BF and TR represent the BiFPN module and transformer module, respectively.

3.3. Analysis of Attention Mechanisms

In order to assess the effectiveness of the CBAM attention mechanism module, other structures of the BTC-YOLOv5s model were retained as experimental parameter settings, and only the CBAM module was replaced with other mainstream attention mechanism modules, such as SE [37], CA [38], and ECA [39] modules, for comparison purposes.

Table 5 shows that the attention mechanism could significantly improve the accuracy of the model. The mAP@0.5 of SE, CA, ECA, and CBAM models reached 83.4%, 83.6%, 83.6%, and 84.3%, respectively, which was 0.4%, 0.6%, 0.6%, and 1.3% higher than that of YOLOv5s + BF + TR model. Each attention mechanism improved the mAP@0.5 to varying degrees, with the CBAM model performing the best and reaching 84.3%, which was 0.9%, 0.7%, and 0.7% higher than that of SE, CA, and ECA models, respectively, and the mAP @ 0.5: 0.95 was also the highest among the four attention mechanisms. The SE and ECA attention mechanisms only took into account the channel information in the feature map, while the CA attentional mechanism encoded the channel relations using the location information. In contrast, the CBAM attention mechanism combined spatial and channel attention, emphasizing the information on disease features in the feature map, which was more conducive to disease identification and localization.

Table 5. Performance comparison of different attention mechanisms.

Attention Mechanisms	mAP@0.5 (%)	mAP@0.5:0.95 (%)	Model Size (MB)	FLOPs (G)
SE	83.4	45.3	15.7	17.5
CA	83.6	45.1	15.8	17.5
ECA	83.6	44.8	15.7	17.5
CBAM	84.3	45.9	15.8	17.5

Moreover, the attention module did not increase the model size or FLOPs, indicating that it was a lightweight module. The BTC-YOLOv5s model with the CBAM module achieved improved recognition accuracy while maintaining the same model size and computational cost.

3.4. Comparison of State-of-the-Art Models

The current mainstream two-stage detection model Faster R-CNN and the one-stage detection models SSD, YOLOv4-tiny, and YOLOx-s were selected for comparison experi-

ments. The ALDD dataset was used for training and testing, with the same experimental parameters across all models. The experimental results are shown in Table 6.

Table 6. Performance comparison of mainstream detection models.

Models	mAP@0.5 (%)	F1 (%)	Model Size (MB)	FLOPs (G)	FPS
SSD	71.56	60.77	92.1	274.70	1.15
Faster R-CNN	35.46	35.83	108	401.76	0.16
YOLOv4-tiny	59.86	55.79	22.4	16.19	8.21
YOLOx-s	80.10	77.36	34.3	26.64	4.08
YOLOv5s	82.70	79.04	13.7	16.40	9.80
BTC-YOLOv5s	84.30	80.56	15.8	17.50	8.70

Among all models, the mAP@0.5 and F1 score of Faster R-CNN were lower than 50%, with a large model size and computational effort, resulting in only 0.16 FPS, making it unsuitable for real-time detection of apple leaf diseases. The one-stage detection model SSD had an mAP@0.5 value of 71.56% and a model size of 92.1 MB, which did not meet the detection requirements in terms of model accuracy and complexity. In the YOLO model series, YOLOv4-tiny had an mAP@0.5 of only 59.86%, and the accuracy was too low. The YOLOx-s achieved 80.1% mAP@0.5, but the FLOPs were 26.64 G, and there were only 4.08 pictures per second. Neither of them was not conducive to mobile deployment. The proposed BTC-YOLOv5s model had the highest mAP@0.5 and F1 score among all models, exceeding SSD, Faster R-CNN, YOLOv4-tiny, YOLOx-s, and YOLOv5s by 12.74%, 48.84%, 24.44%, 4.2%, and 1.6%, respectively. The model size and FLOPs were similar to the baseline model, and FPS reached 8.7 frames per second to meet real-time detection of apple leaf diseases in real scenarios.

As seen in Figure 11, the BTC-YOLOv5s model outperformed the other five models in terms of detection accuracy. Additionally, the BTC-YOLOv5s model exhibited comparable model size, computational effort, and detection speed to the other lightweight models. In summary, the overall performance of the BTC-YOLOv5s model was excellent and could accomplish accurate and efficient apple leaf disease detection tasks in real-world scenarios.

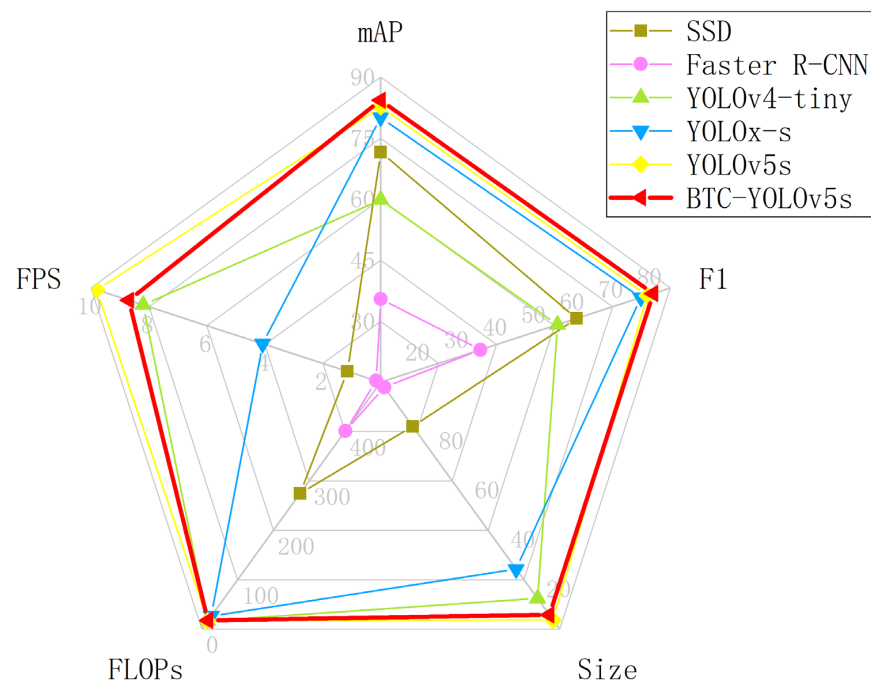


Figure 11. Performance comparison of different detection algorithms.

3.5. Robustness Testing

In the actual production, the detection of apple leaf diseases may be interfered with by various objective environmental factors such as overexposure, dim light, and low-resolution images. In this study, the test set images were simulated by enhancing brightness, reducing brightness, and adding Gaussian noise, resulting in a total of 1191 images (397 images per case). We evaluated the robustness of the optimized BTC-YOLOv5s model under a variety of interference environments to determine its detection effectiveness. Additionally, we tested the model’s ability to detect concurrent diseases by adding 50 images containing multiple diseases. Experimental results are shown in Figure 12.

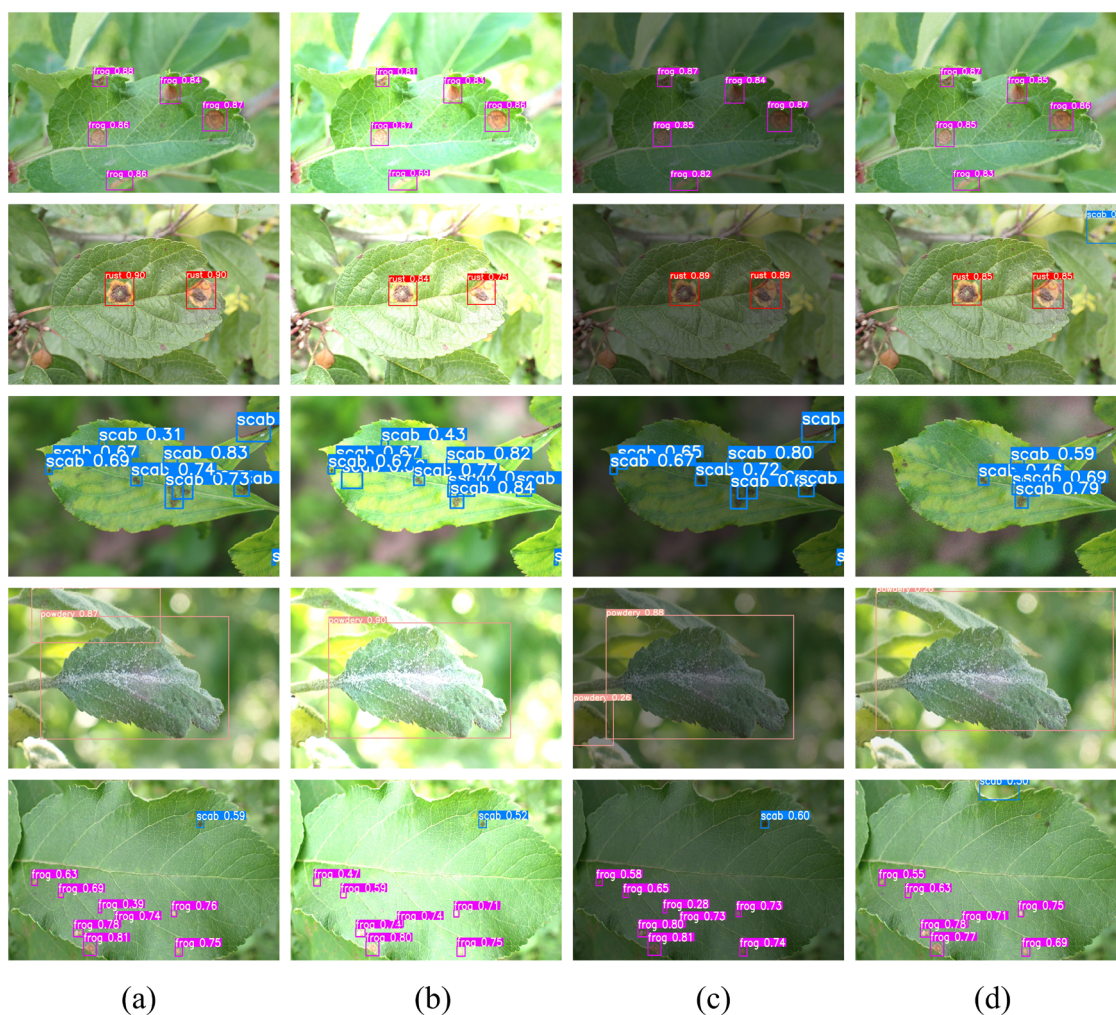


Figure 12. Robustness test results under three extreme conditions. (a) Original; (b) Bright light; (c) Dim light; (d) Blurry. Where first to fifth rows show results for apple frog eye leaf spot, rust, scab, powdery mildew, and multiple diseases, respectively.

From the detection results, the model could accurately detect frog eye leaf spot, rust, and powdery mildew images under all three noise conditions (bright light, dim light, and blurry), with few missing detections. The scab disease was also correctly identified, but a certain degree of missing detections occurred in dim light and blurry conditions. This is mainly because the scab lesions appeared to be black, the overall background of the image has similar color to the lesions under dim light conditions. As shown in the fifth row of Figure 12, the model also demonstrated detection capabilities for images with concurrent onset, although a few missing detections occurred in the blurry condition. The experimental

results achieved more than 80% of mAP. Overall, the BTC-YOLOv5s model still exhibited strong robustness under extreme conditions, such as blurred images and insufficient light.

4. Discussion

4.1. Multi-Scale Detection

Multi-scale detection is a challenging task in apple leaf disease detection due to the varying sizes of the lesions. In this study, frog-eye leaf spot, scab, and rust lesions are typically small and dense, while powdery mildew is a whole lesion distributed over the leaf. The size of the spots that need to be detected relative to the proportion of the whole image can vary widely between images or even within the same image. To address this issue, this study introduced the BiFPN into YOLOv5s based on the idea of multi-scale feature fusion to improve the model's ability. The BiFPN stacks the entire feature pyramid framework multiple times, providing the network with strong feature representation capabilities. It also performs weighted feature fusion, allowing the network to learn the significance of different input features. In the field of agricultural detection, multi-scale detection has been a popular research topic. For example, Li et al. [21] accomplished multi-scale cucumber disease detection by adding a set of anchors matching small instances. Cui et al. [40] used a squeeze-and-excitation feature pyramid network to fuse multi-scale information, retaining only the 26×26 detection head for pinecone detection. However, the current study still faces the challenge of significantly degraded detection accuracy for very large- or very small-scale targets. Future studies will focus on exploring how models can be applied to different scales of disease spots.

4.2. Attentional Mechanisms

The attention mechanism assigns weight to the image features extracted by the model, enabling the network to focus on target regions with important information, while suppressing other irrelevant information and reducing interference caused by irrelevant backgrounds on detection results. The introduction of the attention mechanism can effectively enhance the detection model's feature learning ability, and many researchers have incorporated it to improve model performance. For example, Liu et al. [41] added the SE attention module to YOLOX to enhance the extraction of the cotton boll feature details. Bao et al. [42] added a dual-dimensional mixed attention (DDMA) to the detection model Neck, which parallelizes coordinate attention with channel and spatial attention to reduce missed and false detections caused by dense blade distribution. This study used the CBAM attention mechanism to enhance the BTC-YOLOv5s model's feature extraction ability. CBAM comprised two modules, SAM and CAM, and using the two submodules alone yielded an accuracy of 83.2% and 83.1%, respectively, inferior to the performance of the model using CBAM. As SAM and CAM are only spatial and channel attention modules alone, whereas CBAM combines both, it considers useful information from both feature channels and spatial dimensions, making it more beneficial for the model to locate and identify lesions.

4.3. Outlook

Although the proposed model can accurately identify apple leaf diseases, there are still some issues that deserve attention and further study. Firstly, the dataset used in this study only contains images of four disease types, whereas there are approximately 200 apple diseases in total. Therefore, future research will include images of more species and different disease stages. Secondly, the accuracy of model is not good in case of dense disease and decreases significantly compared to the performance in the sparse case. The detection results showed that scab had the highest error rate, mainly due to its irregular lesion shape and non-obvious border which interfered with the model detection. In the future, scab disease will be considered as a separate research topic to improve the model's detection accuracy.

5. Conclusions

This study proposed an improved detection model BTC-YOLOv5s based on YOLOv5s aimed at addressing the issues of missing and false detection caused by different shapes of diseased spots, multi-scale, and dense distribution of apple leaf lesions. To enhance the overall detection performance of the original YOLOv5s model, the study introduced the BiFPN module, which increases the fusion of multi-scale features and provides more semantic information. Additionally, the transformer and CBAM attention modules were added to improve the ability to extract disease features. Results indicated that the BTC-YOLOv5s model achieved an mAP@0.5 of 84.3% on the ALDD test set, with a model size of 15.8 M and detection speed of 8.7 FPS on an octa-core CPU device. Additionally, it still maintained good performance and robustness under extreme conditions. The improved model has high detection accuracy, fast detection speed and low computational requirements, making it suitable for deployment on mobile devices for real-time monitoring and the intelligent control of apple diseases.

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

Funding: This research was funded by Natural Science Foundation of Henan Province (No.222300420463); Henan Provincial Science and Technology Research and Development Plan Joint Fund (No.222301420113); the Collaborative Innovation Center of Henan Grain Crops, Zhengzhou and by the National Key Research and Development Program of China (No. 2017YFD0301105).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

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

References

- Zhong, Y.; Zhao, M. Research on deep learning in apple leaf disease recognition. *Comput. Electron. Agric.* **2020**, *168*, 105146. [[CrossRef](#)]
- Bi, C.; Wang, J.; Duan, Y.; Fu, B.; Kang, J.-R.; Shi, Y. MobileNet Based Apple Leaf Diseases Identification. *Mob. Netw. Appl.* **2022**, *27*, 172–180. [[CrossRef](#)]
- Abbaspour-Gilandeh, Y.; Aghabara, A.; Davari, M.; Maja, J.M. Feasibility of Using Computer Vision and Artificial Intelligence Techniques in Detection of Some Apple Pests and Diseases. *Appl. Sci.* **2022**, *12*, 906. [[CrossRef](#)]
- Zhang, C.; Zhang, S.; Yang, J.; Shi, Y.; Chen, J. Apple leaf disease identification using genetic algorithm and correlation based feature selection method. *Int. J. Agric. Biol. Eng.* **2017**, *10*, 74–83. [[CrossRef](#)]
- Liu, Y.; Lv, Z.; Hu, Y.; Dai, F.; Zhang, H. Improved Cotton Seed Breakage Detection Based on YOLOv5s. *Agriculture* **2022**, *12*, 1630. [[CrossRef](#)]
- Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
- Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [[CrossRef](#)] [[PubMed](#)]
- Xie, X.; Ma, Y.; Liu, B.; He, J.; Li, S.; Wang, H. A Deep-Learning-Based Real-Time Detector for Grape Leaf Diseases Using Improved Convolutional Neural Networks. *Front. Plant Sci.* **2020**, *11*, 751. [[CrossRef](#)]
- Deng, X.; Tong, Z.; Lan, Y.; Huang, Z. Detection and Location of Dead Trees with Pine Wilt Disease Based on Deep Learning and UAV Remote Sensing. *AgriEngineering* **2020**, *2*, 294–307. [[CrossRef](#)]
- Zhang, K.; Wu, Q.; Chen, Y. Detecting soybean leaf disease from synthetic image using multi-feature fusion faster R-CNN. *Comput. Electron. Agric.* **2021**, *183*, 106064. [[CrossRef](#)]
- Wang, C.; Xiao, Z. Potato Surface Defect Detection Based on Deep Transfer Learning. *Agriculture* **2021**, *11*, 863. [[CrossRef](#)]
- Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single shot MultiBox detector. In Proceedings of the Computer Vision—ECCV 2016, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Cham, Switzerland; pp. 21–37.

13. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
14. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6517–6525.
15. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
16. Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
17. Wang, C.; Xiao, Z. Lychee Surface Defect Detection Based on Deep Convolutional Neural Networks with GAN-Based Data Augmentation. *Agronomy* **2021**, *11*, 1500. [[CrossRef](#)]
18. Son, C.-H. Leaf Spot Attention Networks Based on Spot Feature Encoding for Leaf Disease Identification and Detection. *Appl. Sci.* **2021**, *11*, 7960. [[CrossRef](#)]
19. Li, J.; Qiao, Y.; Liu, S.; Zhang, J.; Yang, Z.; Wang, M. An improved YOLOv5-based vegetable disease detection method. *Comput. Electron. Agric.* **2022**, *202*, 107345. [[CrossRef](#)]
20. Li, Z.; Jiang, X.; Shuai, L.; Zhang, B.; Yang, Y.; Mu, J. A Real-Time Detection Algorithm for Sweet Cherry Fruit Maturity Based on YOLOX in the Natural Environment. *Agronomy* **2022**, *12*, 2482. [[CrossRef](#)]
21. Li, S.; Li, K.; Qiao, Y.; Zhang, L. A multi-scale cucumber disease detection method in natural scenes based on YOLOv5. *Comput. Electron. Agric.* **2022**, *202*, 107363. [[CrossRef](#)]
22. Thapa, R.; Zhang, K.; Snavely, N.; Belongie, S.; Khan, A. The Plant Pathology Challenge 2020 data set to classify foliar disease of apples. *Appl. Plant Sci.* **2020**, *8*, e11390. [[CrossRef](#)]
23. Plant Pathology 2021-FGVC8. Available online: <https://www.kaggle.com/competitions/plant-pathology-2021-fgvc8> (accessed on 14 March 2023).
24. Singh, D.; Jain, N.; Jain, P.; Kayal, P.; Kumawat, S.; Batra, N. PlantDoc: A dataset for visual plant disease detection. In Proceedings of the 7th ACM IKDD CoDS and 25th COMAD, Hyderabad, India, 5–7 January 2020; pp. 249–253.
25. Dong, X.; Yan, S.; Duan, C. A lightweight vehicles detection network model based on YOLOv5. *Eng. Appl. Artif. Intell.* **2022**, *113*, 104914. [[CrossRef](#)]
26. Park, H.; Yoo, Y.; Seo, G.; Han, D.; Yun, S.; Kwak, N. C3: Concentrated-comprehensive convolution and its application to semantic segmentation. *arXiv* **2018**, arXiv:1812.04920.
27. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [[CrossRef](#)] [[PubMed](#)]
28. Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
29. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path aggregation network for instance segmentation. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8759–8768.
30. Zheng, Z.; Wang, P.; Liu, W.; Li, J.; Ye, R.; Ren, D. Distance-iou loss: Faster and better learning for bounding box regression. In Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI-20), New York, NY, USA, 7–12 February 2020; Volume 34, pp. 12993–13000.
31. Tan, M.; Pang, R.; Le, Q.V. Efficientdet: Scalable and efficient object detection. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 10781–10790.
32. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S. An image is worth 16 × 16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
33. Zhu, X.; Lyu, S.; Wang, X.; Zhao, Q. TPH-YOLOv5: Improved YOLOv5 Based on Transformer Prediction Head for Object Detection on Drone-captured Scenarios. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 2778–2788.
34. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-end object detection with transformers. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; pp. 213–229.
35. Nediyanachath, A.; Paramasivam, P.; Yenigalla, P. Multi-head attention for speech emotion recognition with auxiliary learning of gender recognition. In Proceedings of the ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 7179–7183.
36. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
37. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
38. Hou, Q.; Zhou, D.; Feng, J. Coordinate attention for efficient mobile network design. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 13713–13722.
39. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 11531–11539.
40. Cui, M.; Lou, Y.; Ge, Y.; Wang, K. LES-YOLO: A lightweight pinecone detection algorithm based on improved YOLOv4-Tiny network. *Comput. Electron. Agric.* **2023**, *205*, 107613. [[CrossRef](#)]

41. Liu, Q.; Zhang, Y.; Yang, G. Small unopened cotton boll counting by detection with MRF-YOLO in the wild. *Comput. Electron. Agric.* **2023**, *204*, 107576. [[CrossRef](#)]
42. Bao, W.; Zhu, Z.; Hu, G.; Zhou, X.; Zhang, D.; Yang, X. UAV remote sensing detection of tea leaf blight based on DDMA-YOLO. *Comput. Electron. Agric.* **2023**, *205*, 107637. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.