

Article

Hemerocallis citrina Baroni Maturity Detection Method Integrating Lightweight Neural Network and Dual Attention Mechanism

Liang Zhang ^{1,†} , Ligang Wu ^{1,2,*,†} and Yaqing Liu ^{2,*}¹ College of Mechanical and Electrical Engineering, Shanxi Datong University, Datong 037003, China² School of Information Science and Technology, Dalian Maritime University, Dalian 116026, China

* Correspondence: wuligang@sxdtdx.edu.cn (L.W.); liuyaqing@dlnu.edu.cn (Y.L.)

† These authors contributed equally to this work.

Abstract: North of Shanxi, Datong Yunzhou District is the base for the cultivation of *Hemerocallis citrina Baroni*, which is the main production and marketing product driving the local economy. *Hemerocallis citrina Baroni* and other crops' picking rules are different: the picking cycle is shorter, the frequency is higher, and the picking conditions are harsh. Therefore, in order to reduce the difficulty and workload of picking *Hemerocallis citrina Baroni*, this paper proposes the GGSC YOLOv5 algorithm, a *Hemerocallis citrina Baroni* maturity detection method integrating a lightweight neural network and dual attention mechanism, based on a deep learning algorithm. First, Ghost Conv is used to decrease the model complexity and reduce the network layers, number of parameters, and Flops. Subsequently, combining the Ghost Bottleneck micro residual module to reduce the GPU utilization and compress the model size, feature extraction is achieved in a lightweight way. At last, the dual attention mechanism of Squeeze-and-Excitation (SE) and the Convolutional Block Attention Module (CBAM) is introduced to change the tendency of feature extraction and improve detection precision. The experimental results show that the improved GGSC YOLOv5 algorithm reduced the number of parameters and Flops by 63.58% and 68.95%, respectively, and reduced the number of network layers by about 33.12% in terms of model structure. In the case of hardware consumption, GPU utilization is reduced by 44.69%, and the model size was compressed by 63.43%. The detection precision is up to 84.9%, which is an improvement of about 2.55%, and the real-time detection speed increased from 64.16 FPS to 96.96 FPS, an improvement of about 51.13%.

Keywords: deep learning; lightweight neural networks; attentional mechanisms; *Hemerocallis citrina Baroni*; maturity detection



Citation: Zhang, L.; Wu, L.; Liu, Y. *Hemerocallis citrina Baroni* Maturity Detection Method Integrating Lightweight Neural Network and Dual Attention Mechanism. *Electronics* **2022**, *11*, 2743. <https://doi.org/10.3390/electronics11172743>

Academic Editor: Rashid Mehmood

Received: 2 August 2022

Accepted: 25 August 2022

Published: 31 August 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 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

In recent years, the policies of agricultural revitalization strategy and agricultural poverty alleviation have achieved many successes. Under the background of a rural revitalization strategy, facing the opportunities and challenges in the process of rapid development of agriculture, the Ministry of Agriculture and Rural Affairs has implemented science and technology to assist agriculture, accelerate the integration of rural industries and digital economy, solve the problems of low efficiency and quality, and actively encourage and promote the efficient development of smart agriculture.

In the process of sowing and growing [1–3], fertilizing and watering [4–6], pest monitoring [7–9], and fruit picking [10,11] of agricultural products [12], smart agriculture plays an irreplaceable role in improving the quality of agricultural products; it makes all the work more convenient and efficient, so smart agriculture has received a wide range of attention from researchers.

At present, the effective combination of artificial intelligence technology and smart agriculture has become a key research topic, whereas computer vision [13] and deep learning technology have become effective measures to promote rural revitalization and

agricultural poverty alleviation. Based on deep learning methods and computer vision techniques, in Ref. [14], an accurate quality assessment of different fruits was efficiently accomplished by using the Faster RCNN target detection algorithm. Similarly, in Ref. [15], the authors accomplished tomato ripening detection based on color and size differentiation using a two-stage target detection algorithm, Faster R-CNN, which has an accuracy of 98.7%. In addition, Wu [16] completed the detection of strawberries in the process of strawberry picking by a U²-Net network image segmentation technique, and applied it to the automated picking process.

Compared with two-stage target detection algorithms, single-stage target detection algorithms are more advantageous, and the YOLO algorithm is a typical representative. With the YOLOv3 target detection algorithm, Zhang et al. [17] precisely located the fruit and handle of a banana, which is convenient for intelligent picking and operation, and the average precision was as high as 88.45%. In Ref. [18], the detection of a palm oil bunch was accomplished using the YOLOv3 algorithm and its application in embedded devices through mobile and the Internet of Things. Wu et al. [19] accomplished the identification and localization of fig fruits in complex environments by using the YOLOv4 target detection algorithm, which distinguishes and discriminates whether the fig fruits are ripe or not. Zhou et al. [20] completed the ripening detection of tomatoes by K-means clustering and noise reduction processing based on the YOLOv4 algorithm, but the detection speed was only 5–6 FPS, which could not meet the needs of real-time detection.

In summary, it can be seen that deep learning methods and computer vision techniques are widely used in smart agriculture in previous studies [21,22]. However, with the improvement and update of the YOLO algorithm, the YOLOv5 algorithm was proposed, but, currently, there are less applications in agriculture related fields. Inspired by existing studies, we applied the YOLOv5 algorithm to the maturity detection process of *Hemerocallis citrina Baroni* because North of Shanxi, Datong Yunzhou District, has been known as the hometown of *Hemerocallis citrina Baroni* since ancient times, and is also the planting base of organic *Hemerocallis citrina Baroni*.

In recent years, the *Hemerocallis citrina Baroni* industry in Yunzhou District has entered the fast track of development. As the leading industry of “one county, one industry”, it has brought rapid economic development while promoting rural revitalization. However, at present, the picking of *Hemerocallis citrina Baroni* mainly relies on manual completion, and whether the *Hemerocallis citrina Baroni* is mature or not relies entirely on experience to distinguish. Therefore, the main work and contributions of this paper are as follows:

- a. Computer vision technology and deep learning algorithms are applied to the maturity detection of *Hemerocallis citrina Baroni*, and highly accurate maturity detection of whether the *Hemerocallis citrina Baroni* are mature and meet the picking standards, providing ideas for improving the picking method and reducing the cost of picking labor.
- b. The lightweight neural network is introduced to reduce the number of network layers and model complexity, compress the model volume, and lay the foundation for the embedded development of picking robots.
- c. Combined with the dual attention mechanism, it improves the tendency of feature extraction and enhances the detection precision and real-time detection efficiency.

The remainder of this paper is organized as follows. Section 2 offers YOLOv5 object detection algorithms. In Section 3, the GGSC YOLOv5 network structure and its constituent modules are presented. Section 4 introduces the model training parameters, advantages of the lightweight model, and analysis of experiments results. Finally, conclusions and future work are given in Section 5.

2. YOLOv5 Object Detection Algorithms

Currently, there are two types of deep learning target detection algorithms, one-stage and two-stage. The one-stage target detection algorithm performs feature extraction on the entire image to complete end-to-end training, and the detection process is faster, but less

precise. Common algorithms include YOLO, SSD, Retina Net, etc. The two-stage target detection algorithm selectively traverses the entire image by pre-selecting the boxes, which is a slower detection process, but has higher precision. Faster RCNN, Cascade RCNN, Mask RCNN, etc. are common algorithms.

With continuous improvements and updates [23], the YOLOv5 target detection algorithm has improved detection precision and model fluency. As shown in Figure 1, YOLOv5 mainly consists of four parts: input head, backbone, neck, and prediction head. The input head is used as the input of the convolutional neural network, which completes the cropping and data enhancement of the input image, and backbone and neck complete the feature extraction and feature fusion for the detected region, respectively. The prediction head is used as the output to complete the recognition, classification, and localization of the detected objects [24].

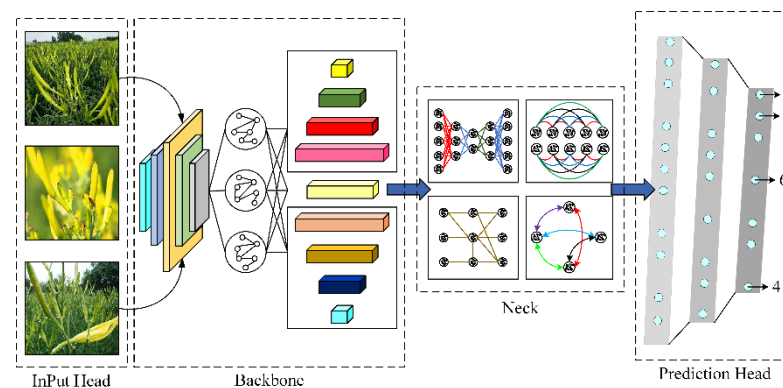


Figure 1. YOLOv5 algorithm process.

In the version 6.0 of the YOLOv5 algorithm, the focus module is replaced by a rectangular convolution with $stride = 2$, the CSP residual module [25] is replaced by the C3 module, and the size of the convolution kernel in the spatial pyramid pooling (SPP) [26] module is unified to 5. However, the problems of the complex model structure, redundant feature extraction during convolution, and large number of parameters and computation of the model still exist, which are not suitable for mobile and embedded devices.

To address the above problems, the GGSC YOLOv5 detection algorithm based on a lightweight and double attention mechanism is proposed in this paper, and applied to *Hemerocallis citrina Baroni* recognition, with the obvious advantages of the lightweight model and excellent detection performance in the detection process.

3. Deep Learning Detection Algorithm GGSC YOLOv5

3.1. Ghost Lightweight Convolution

In limited memory and computational resources, deploying efficient and lightweight neural networks is the future development direction of convolutional neural networks [27]. In the feature extraction process, the traditional convolution traverses the entire input image sequentially, with many similar feature maps generated by adjacent regions during the convolution process. Therefore, traditional convolutional feature extraction is computationally intensive, inefficient, and redundant in terms of information.

As shown in Figure 2, Ghost Conv [28] takes advantage of the redundancy characteristic of the feature map, and first generates m intrinsic feature maps by a few traditional convolutions. Then, the Φ_i cheap linear operation is performed on m intrinsic feature maps, such that each intrinsic feature map produces $s - 1$ new feature maps. Lastly, the m intrinsic feature maps and $s - 1$ new feature maps are spliced together to complete the lightweight convolution operation.

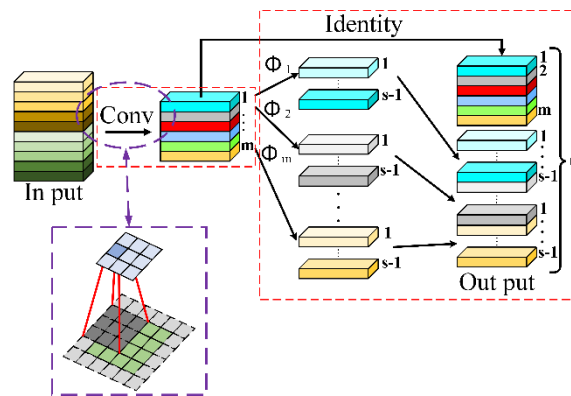


Figure 2. The Ghost Convolution process.

The Ghost Conv process is less computationally intensive and more lightweight than traditional convolution due to the cheap linear operations introduced in the process. Therefore, the theoretical speedup ratio (r_s) and model compression ratio (r_c) of Ghost Conv and traditional convolution are as follows, respectively:

$$r_s = \frac{C_T}{C_G} = \frac{c \times k \times k \times ms \times h' \times w'}{c \times k \times k \times m \times h' \times w' + m \times k \times k \times (s - 1) \times h' \times w'} = \frac{c \times s}{c + s - 1} \approx s \quad (1)$$

$$r_c = \frac{c \times k \times k \times ms}{c \times k \times k \times m + m \times k \times k \times (s - 1)} = \frac{c \times s}{c + s - 1} \approx s \quad (2)$$

where $h \times w$ and $h' \times w'$ are the height and width sizes of the input and output images. c is the number of input channels, ms is the number of output channels, and $k \times k$ is the custom convolution kernel size. C_T and C_G are the convolutional computations of traditional convolution and Ghost Conv, respectively.

In summary, the r_s and r_c of Ghost Conv are only $1/s$ of the traditional convolution due to the introduction of cheap linear operations. It can be seen that Ghost Conv has obvious advantages of being lightweight, with a lower number of parameters and computation compared with the traditional convolution.

3.2. Ghost Lightweight Bottleneck

Ghost Bottleneck [28] is a lightweight module consisting of Ghost Conv, Batch Normalization (BN) layers, down sampling, and activation functions. Its design method and model structure are similar to that of the Res Net residual network, which has the features of a simple model structure, easy application, and high operational efficiency. Since the number of channels remains constant before and after feature extraction, the module can be plug-and-play.

The structure of the Ghost Bottleneck model is shown in Figure 3. Ghost Bottleneck mainly consists of two Ghost Conv stacks, where the input image is passed through the first Ghost Conv to increase the number of channels, normalized by the BN layer, and the nonlinear properties of the neural network model are increased by the ReLU activation function. Subsequently, it goes through a second Ghost Conv to reduce the number of channels, thus ensuring that the number of output channels is the same as before the first Ghost Conv operation. Lastly, the output after twice Ghost Conv and the original input after down sampling are spliced and stacked by Add operation, which increases the amount of information of the desired features, while the number of channels remains the same.

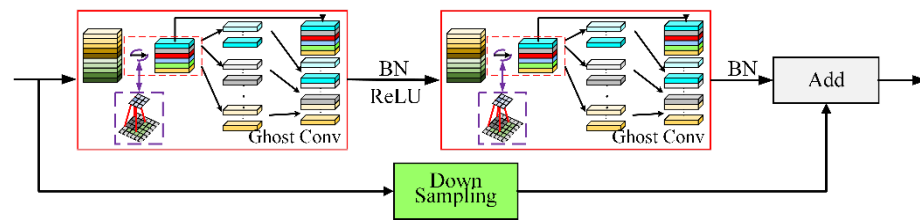


Figure 3. Ghost Bottleneck modules.

The structure of the Ghost Bottleneck model is similar to MobileNetv2. The BN layer is retained after compressing the channels without using the activation function, so the original information of feature extraction is retained to the maximum extent. Compared with other residual modules and cross-stage partial (CSP) network layers, Ghost Bottleneck uses fewer convolutional and BN layers, and the model structure is simpler. Therefore, using Ghost Bottleneck makes the number of model parameters and the Flops calculation lower, the number of network layers less, and the lightweight feature more obvious.

3.3. SE Attentional Mechanisms

The Squeeze-and-Excitation channel attention mechanism module (SE Module) [29] consists of two parts: Squeeze and Excitation. First, the SE Module performs the Squeeze operation on the feature map obtained by convolution to get the global features on the channel. Subsequently, the Excitation operation is performed on the global features, which learns the relationship between each channel and obtains the weight values of different channels. Lastly, the weight values of each channel are multiplied on the original feature map to obtain the final features after performing the SE Module.

The SE Module feature extraction process is shown in Figure 4. During the Squeeze operation, global average pooling is used to obtain global features, and the output z_c is obtained by the compression function F_{sq} according to the compression aggregation strategy. During the Excitation operation, the dimension is first reduced, and then, the dimension is increased. The output s after Excitation is obtained by the excitation function F_{ex} , and the relationship between channels is obtained by the feature capture mechanism of *Sigmoid* to complete the feature extraction.

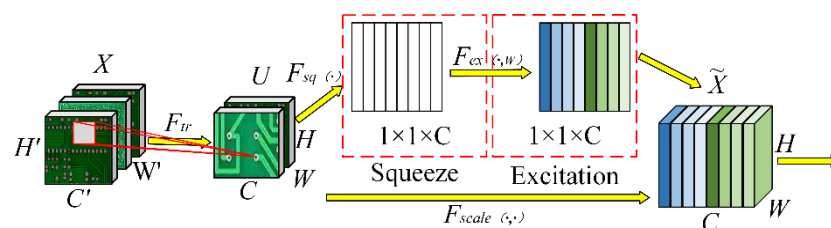


Figure 4. SE attentional mechanisms.

In the SE channel attention mechanism model, the Squeeze-Excitation function can be expressed as $x_c = F_{scale}(u_c, s_c) = s_c u_c$, where u_c represents the c -th feature map in the Squeeze-Excitation process, and s_c represents the weight of the c -th feature map.

The SE channel attention mechanism adaptively [30] accomplishes the adjustment of feature weights during feature extraction, which is more conducive to obtaining the required feature information. Therefore, the SE channel attention mechanism is introduced into the model structure, which can enhance the discrimination ability of the model, and improve the detection accuracy and maturity detection effect.

3.4. CBAM Attentional Mechanisms

The Convolutional Block Attention Module (CBAM) [31] attention mechanism is an efficient feed-forward convolutional attention model, which can perform the propensity

extraction of features sequentially in channel and spatial dimensions, and it consists of two sub-modules: Channel Attention Module (CAM) and Spatial Attention Module (SAM).

The CBAM feature extraction process is shown in Figure 5. First, compared with the SE Module, the channel attention mechanism in CBAM adds a parallel maximum pooling layer, which can obtain more comprehensive information. Second, the CAM and SAM modules are used sequentially to make the model recognition and classification more effective. Lastly, since CAM and SAM perform feature inference sequentially along two mutually independent dimensions, the combination of the two modules can enhance the expressive ability of the model.

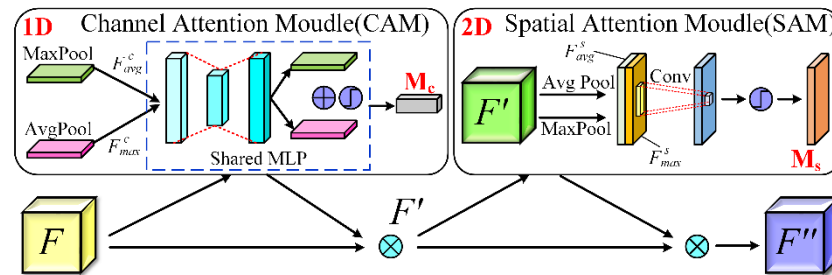


Figure 5. CBAM attentional mechanisms.

In the CAM module, with the input feature map performing maximum pooling and average pooling in parallel, the shared network in the multilayer perceptron (MLP) performs feature extraction based on the maximum pooling feature maps F_{max} and average pooling feature maps F_{avg} to produce a 1D channel attention map M_c . The CAM convolution calculation can be expressed as:

$$\begin{aligned}
 M_c(F) &= \sigma[MLP(AvgPool(F)) + MLP(MaxPool(F))] \\
 &= \sigma[W_1(W_0(F_{avg}^c)) + W_1(W_0(F_{max}^c))]
 \end{aligned}
 \tag{3}$$

where σ denotes the sigmoid function, and W_0 and W_1 denote the weights after pooling and sharing the network, respectively.

In the SAM module, the input feature map performs maximum pooling and average pooling in parallel, and a 2D spatial attention map M_s is generated by traditional convolution. The SAM convolution calculation can be expressed as:

$$\begin{aligned}
 M_s(F) &= \sigma[f^{k \times k}(AvgPool(F); MaxPool(F))] \\
 &= \sigma[f^{k \times k}(F_{avg}^s; F_{max}^s)]
 \end{aligned}
 \tag{4}$$

where $f^{k \times k}$ represents a traditional convolution operation with the filter size of $k \times k$.

The CBAM attention mechanism is an end-to-end training model with plug-and-play functionality, and, thus, can be seamlessly fused into any convolutional neural network. Combined with the YOLO algorithm, it can complete feature extraction more efficiently and obtain the required feature information without additional computational cost and operational pressure.

3.5. GGSC YOLOv5 Model Structure

The improved GGSC YOLOv5 algorithm model structure and module parameters are shown in Figure 6, which combines Ghost Conv and Ghost Bottleneck modules to achieve a light weight, and introduces the dual attention mechanism of SE and CBAM to improve detection precision (P) and real-time detection efficiency.

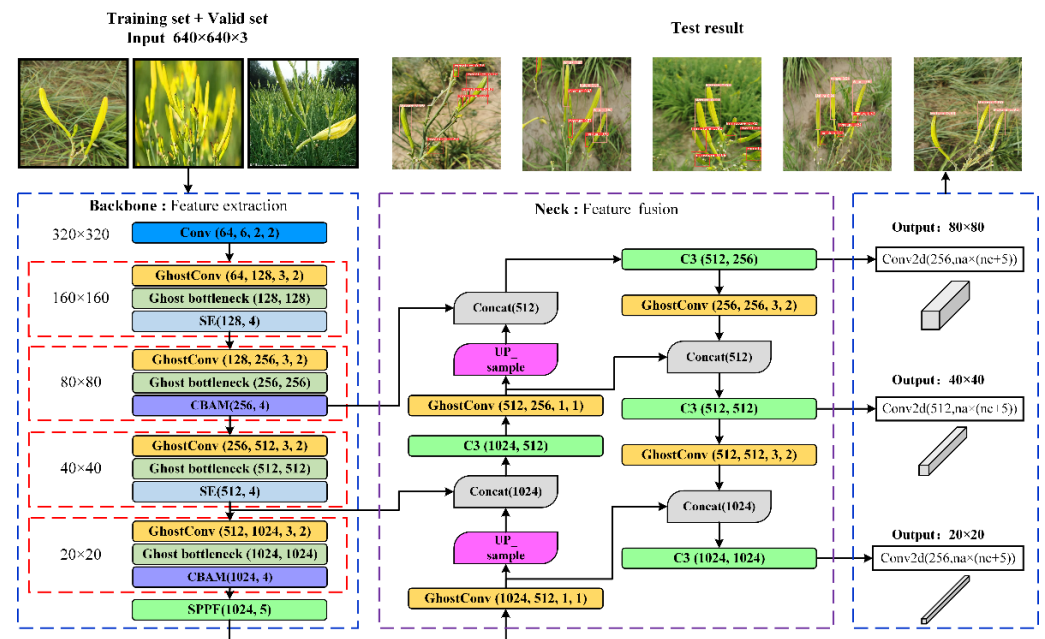


Figure 6. GGSC YOLOv5 model structure.

In the GGSC YOLOv5 algorithm feature extraction network backbone, Ghost Conv and Ghost Bottleneck module sets are used instead of traditional convolution and C3 modules, respectively, which reduces the consumption of memory and hardware resources in the convolution process. The SE and CBAM attention mechanisms are used alternately after each module group (Ghost Conv and Ghost Bottleneck) to enhance the tendency of feature extraction, enabling the underlying fine-grained information and the high-level semantic information to be extracted effectively. In the feature fusion network neck, images with different resolutions are fused by Concatenate and Up-sample, which makes the localization information, classification information, and confidence information of the feature map more accurate.

After the feature extraction and feature fusion, three different tensor are generated at the output prediction head by conventional convolution, Conv2d: $(256, na \times (nc + 5))$, $(512, na \times (nc + 5))$, and $(1024, na \times (nc + 5))$, corresponding to three sizes of output: 80×80 , 40×40 , and 20×20 , where 256, 512, and 1024 denote the number of channels, respectively. $na \times (nc + 5)$ represents the relevant parameters of the detected object; the number of anchors for each category and the number of categories of detected objects are denoted by na and nc , respectively. The four localization parameters and one confidence parameter of the anchor are represented by 5.

4. Experiments and Results Analysis

4.1. Model Training

The experiments in this paper were carried out using the Python 3.8.5 environment and CUDA 11.3, under Intel Core i9-10900k@3.7 GHz, NVidia GeForce RTX 3080 10G, and DDR4 3600 MHz dual memory hardware.

For the dataset, 800 images of *Hemerocallis citrina Baroni* were available after photographs on the spot, screening, dataset production, and classification. Among them, 597 images are used as the training set, 148 images are used as the validation set, and 55 images are used as the test set.

In this paper, the original YOLOv5 and the improved GGSC YOLOv5 algorithm use the same parameter settings, the image input is 640×640 , the learning rate is 0.01, the cosine annealing hyper-parameter is 0.1, the weight decay coefficient is 0.0005, and the momentum parameter in the gradient descent with momentum is 0.937. A total of 300 epochs and a batch size of 12 are used during training.

The GGSC YOLOv5 training process of the *Hemerocallis citrina Baroni* recognition method based on the lightweight neural network and dual attention mechanism is shown in Algorithm 1.

Algorithm 1: Training process of GGSC YOLOv5.

Determine: Parameters, Anchor, lr , L_{ciou} , IOU_{th} .
 InPut: Training dataset, Valid dataset, Label set.
 Loading: Train models, Valid models.
 Ensure: In Put, Backbone, Neck, OutPut. Algorithm environment.
 N iterations of training. i -th iteration training ($i \leq N$):
Feature extraction Net:
 a: Rectangular convolution
 b: i -th iteration ($i \leq 2$):
 Ghost Conv-Ghost Bottleneck-SE, feature extraction.
 Ghost Conv-Ghost Bottleneck-CBAM, feature extraction.
 c: Feature fusion.
 d: Predicted Head: classification c_i , confidence p_i .
 e: Positioning error, category error, confidence error.
 f: $\sum loss$.
Val Net:
 a: Test effect of model ω_i .
 b: Calculate P , R and $Val loss$.
 c: Adjust lr and update strategy.
 Save results of the i -th training: weight π_i , and model ω_i .
 Update: Weight: $\pi_{i+1} \leftarrow \pi_i$, Model: $\omega_{i+1} \leftarrow \omega_i$.
 Temporary storage model ω' .
 Plot: Result curve, Save best model ω' , Output.
End Train

4.2. Model Lightweight Analysis

The cultivation of *Hemerocallis citrina Baroni* has the characteristics of vast area, dense plants, and different growth. Therefore, recognition methods based on computer vision and deep learning are widely used in robotic picking, and the lightweight features are more in line with the practical needs and future development direction of embedded devices.

The GGSC YOLOv5 algorithm takes advantage of the redundancy of the feature maps to reduce the model complexity while improving the efficiency of feature extraction and the relationship between channels.

A comparison of the model parameters of GGSC YOLOv5 and the original YOLOv5 is shown in Figure 7. The improved algorithm has the obvious advantages of being lightweight. In terms of model structure, the number of network layers is reduced from 468 to 313, which is about 33.12% less. The number of parameters and the number of Flops operations decreased significantly, by about 63.58% and 68.95%, respectively. In terms of memory occupation and hardware consumption, the GPU utilization [32] was reduced from 6.9 G to 3.8 G, a reduction of about 44.69%. The volume of the model trained is reduced from 92.7 M to 33.5 M, a compression of about 63.43%. At the same time, the time required for training 300 epochs is reduced by 3.4%.

Neural network algorithms based on computer vision and deep learning have high requirements on the hardware and computing power of microcomputers. The memory and computational resources of the picking robot are limited in the recognition process of *Hemerocallis citrina Baroni*, so the GGSC YOLOv5 algorithm can show its advantages of being lightweight, and can reduce the demand of hardware equipment for the picking device.

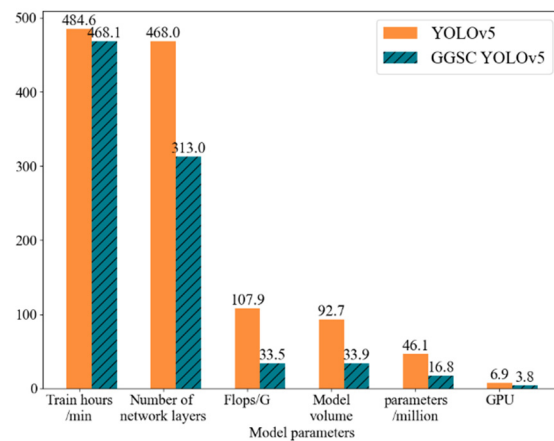


Figure 7. Comparison of the number of model parameters.

4.3. Model Training Process Analysis

The loss function convergence curves during model training are shown in Figure 8. As can be seen from the figure, the loss function curves of the training and validation sets of the training process show an obvious convergence trend, and the convergence speed of the validation set is faster, which proves that the model has excellent learning performance during the training process, so it shows more satisfactory results in the validation process.

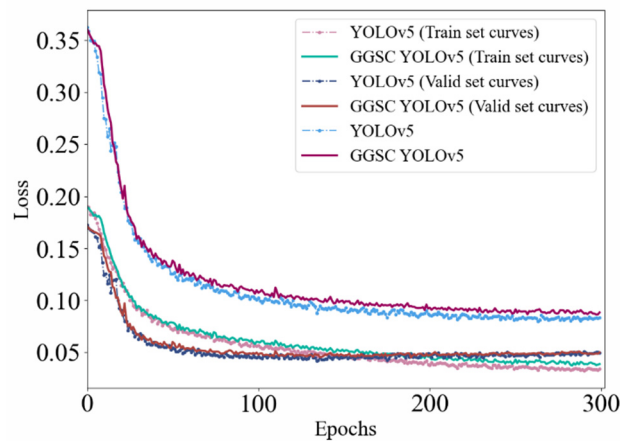


Figure 8. Loss value convergence curve with epoch times.

During the training process of the previous 50 times, the feature extraction is obvious, the learning efficiency is high, and the loss function continues to decline. After 100 iterations of training, the convergence trend of GGSC YOLOv5 and YOLOv5 algorithms is roughly the same, showing a gradual stabilization trend, and the loss function value does not decrease, the model converges, and the detection accuracy tends to be stable.

In the field of deep learning target detection, the reliability of the resulting model can be evaluated by calculating the precision (P), recall (R), and harmonic mean (F_1) based on the number of positive and negative samples.

The $R - P$ curve consists of R and P . It can show the variation trend of model P with R . The area under the $R - P$ curve line can indicate the average precision (AP) of the model, and the larger the area under the $R - P$ curve line, the higher the AP of the model, and the better the comprehensive performance.

The $R - P$ curves of the GGSC YOLOv5 and YOLOv5 algorithms are shown in Figure 9. In the figure, the trend and area under the line are approximately the same for both curves. During the model training, the AP value of GGSC YOLOv5 is 0.884, whereas the AP value of the YOLOv5 algorithm is 0.890, which is a very small difference. However, the model

structure of the GGSC YOLOv5 algorithm is simpler, lighter, and requires less hardware devices, memory size, and computer computing power.

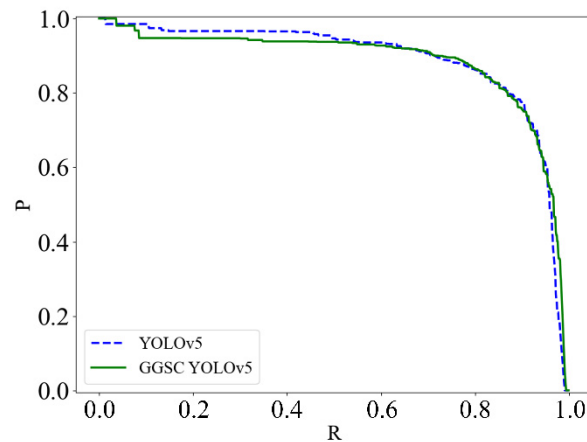


Figure 9. Comparison with $R - P$ curve.

The harmonic mean F_1 is influenced by P and R , which can reflect the comprehensive performance of the model. The value of F_1 is higher, and the model has better equilibrium performance for P and R , and vice versa. P and R indexes in the model can be effectively assessed by F_1 , which can determine whether there is a sharp increase in one and a sudden decrease in the other. Therefore, it is an important indicator to assess the reliability and comprehensiveness of the model.

The variation trend of YOLOv5 and GGSC YOLOv5 harmonic mean value curves F_1 with confidence is shown in Figure 10. After combining the lightweight network and the dual attention mechanism, the GGSC YOLOv5 algorithm has the same harmonic mean value as YOLOv5, both of which are 0.84. The results show that the harmonic performance of the improved algorithm is not affected on the basis of achieving lightweight.

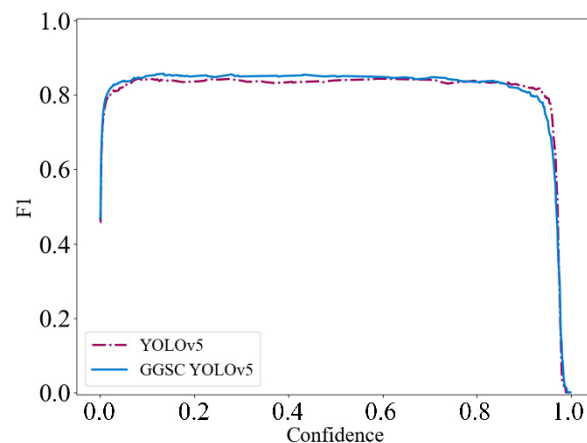


Figure 10. Comparison of the F_1 curves for before and after algorithm improvement.

In the process of identifying whether *Hemerocallis citrina Baroni* is mature, detection precision (P) is a key performance indicator that has a decisive impact on the picking results. In the process of picking *Hemerocallis citrina Baroni*, the higher the P , the more accurate the picking, and the lower the loss, the higher the income, and vice versa.

The precision curves of the YOLOv5 algorithm and the GGSC YOLOv5 algorithm, as well as the curve fitting, are shown in Figure 11. As can be seen from the figures, Figure 11a shows the original data and precision curves of the YOLOv5 and GGSC YOLOv5 algorithms. Figure 11b,c show the original and fitted curves of the YOLOv5 and GGSC

YOLOv5 algorithms, respectively. Figure 11d shows the fitted precision curves of the YOLOv5 and GGSC YOLOv5 algorithms.

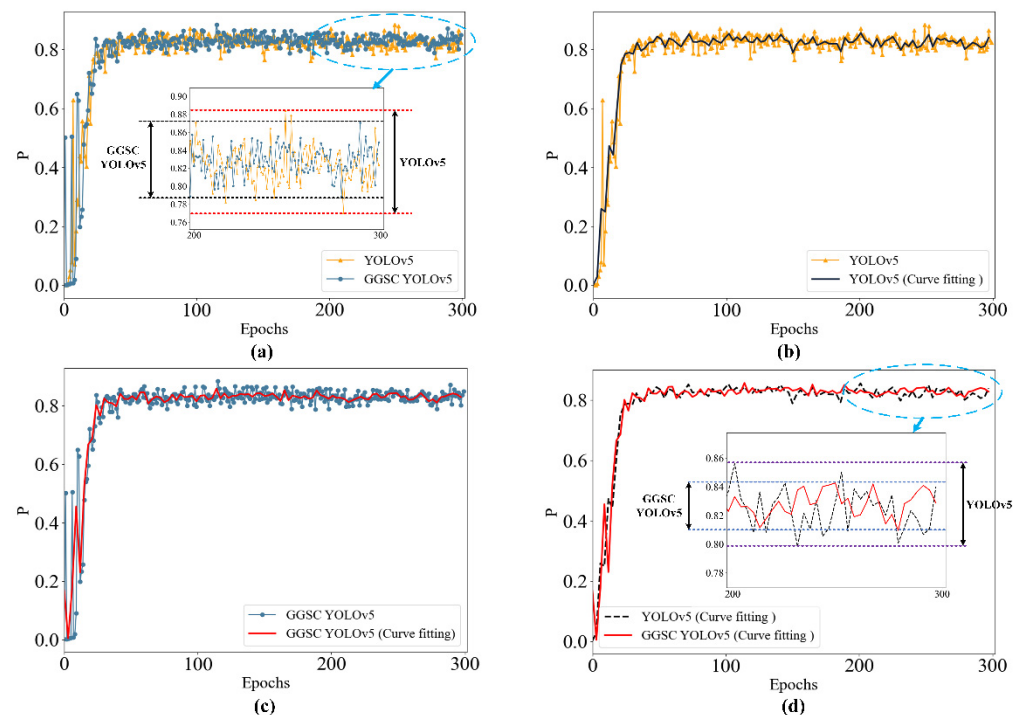


Figure 11. Model precision curve.

In the original precision curve and the fitted precision curve, GGSC YOLOv5 has less fluctuation range and higher precision compared with YOLOv5. During the training process, the final precision of YOLOv5 is 82.36%, whereas the final precision of GGSC YOLOv5 is 84.90%. After the introduction of Ghost Conv, Ghost Bottleneck, and the double attention mechanism, not only is a light weight achieved, but also the detection precision is improved by 2.55%.

The precision and fast picking of *Hemerocallis citrina Baroni* is a prerequisite to ensure picking efficiency. Therefore, for the maturity detection of *Hemerocallis citrina Baroni*, in addition to the detection precision, the real-time detection speed is also a crucial factor.

The real-time detection speed is determined by the number of image frames processed per second (*FPS*). The more frames processed per second, the faster the real-time detection speed and the better the real-time detection performance of the model, and vice versa. In the maturity detection process of *Hemerocallis citrina Baroni*, the real-time detection speed of the YOLOv5 algorithm is 64.14 *FPS*, whereas the GGSC YOLOv5 is 96.96 *FPS*, which exceeds the original algorithm by about 51.13%. It can be seen that based on computer vision technology and deep learning methods, the GGSC YOLOv5 algorithm can complete the recognition of *Hemerocallis citrina Baroni* with high accuracy and efficiency.

In summary, the average precision and harmonic mean performance of GGSC YOLOv5 and YOLOv5 algorithms are approximately the same. However, in model lightweight analysis, the GGSC YOLOv5 algorithm has more prominent advantages, which is in line with the future development direction of neural networks, and can also meet the needs of embedded devices in agricultural production. The experimental results of the training process show that GGSC YOLOv5 has higher detection precision and real-time detection speed, which can effectively improve the picking efficiency and meet the needs of *Hemerocallis citrina Baroni* picking.

Figure 12 compares the maturity detection results of the YOLOv5 algorithm and the improved GGSC YOLOv5 lightweight algorithm for *Hemerocallis citrina Baroni*. It can be seen from Figure 12a that the improved algorithm has higher coverage and detection precision

for yellow flower detection, with the same confidence threshold and intersection-over-union ratio threshold. In the multi-plant environment, GGSC YOLOv5 was more effective in detecting the overlap of *Hemerocallis citrina Baroni* fruits, whereas in the single-plant environment, the GGSC YOLOv5 algorithm gave a higher confidence in classification and maturity detection. In contrast, the GGSC YOLOv5 algorithm proposed in this paper has better maturity detection ability, and it can accurately identify highly dense, overlapping, and obscured *Hemerocallis citrina Baroni* fruits.

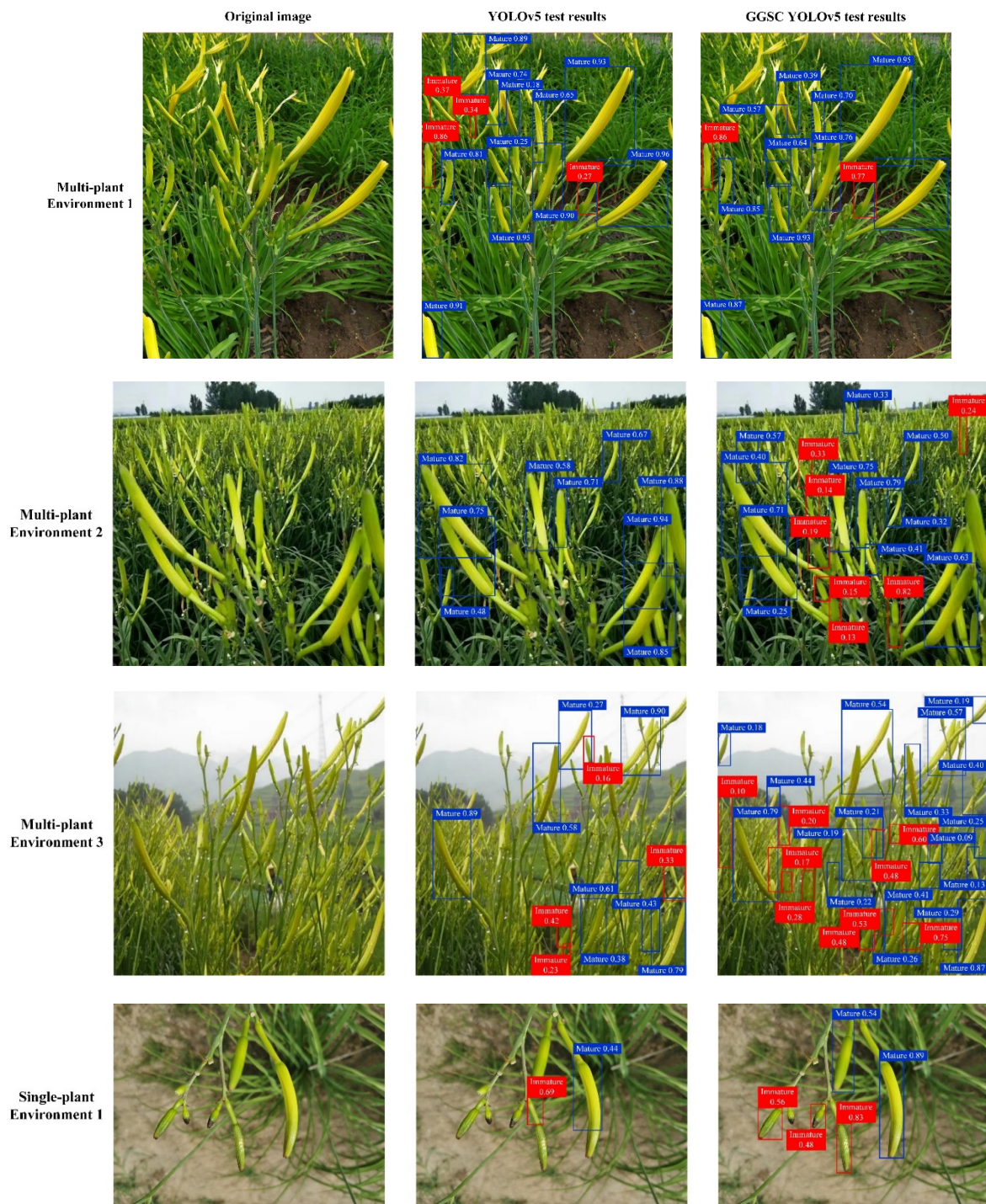


Figure 12. Cont.

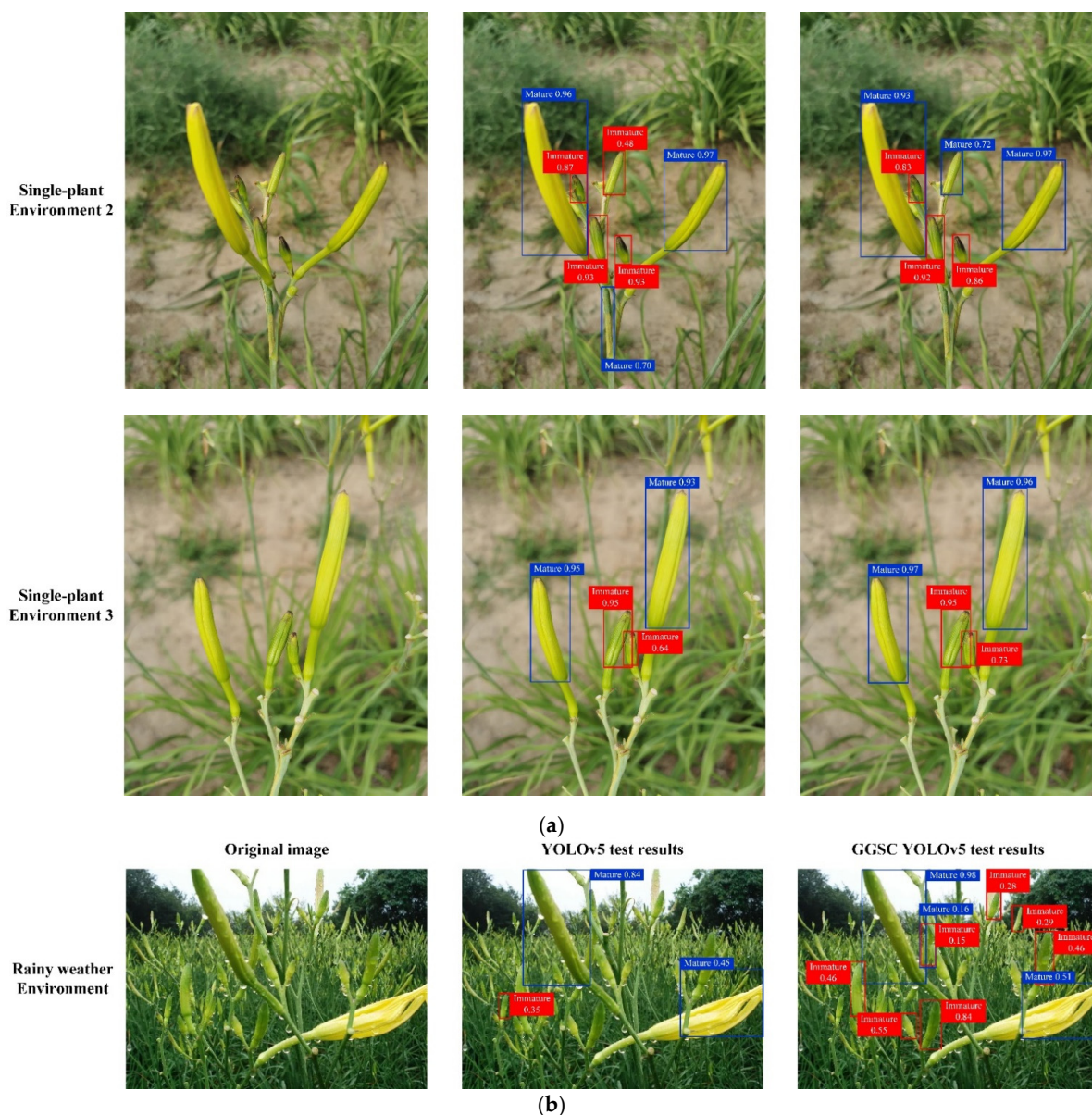


Figure 12. Maturity detection results of *Hemerocallis citrina Baroni* by different algorithms. (a) Test results of multi-plant and single-plant environment. (b) Test results of rainy weather environment.

In crop growing and picking, special environmental factors (e.g., rainy weather) can affect the normal picking work. Therefore, in order to verify the effectiveness of the proposed algorithm in this paper under multiple scenarios, the detection results of different algorithms in rainy weather environments are presented in Figure 12b. The experiments show that special factors such as rain and dew adhesion do not affect the effectiveness of the proposed algorithm, and it shows better maturity detection and detection results than the original algorithm, which shows that the proposed algorithm has good generalization and derivation ability.

5. Conclusions

In this paper, we propose a deep learning target detection algorithm, GGSC YOLOv5, based on a lightweight and dual attention mechanism, and apply it to the picking maturity detection process of *Hemerocallis citrina Baroni*. Ghost Conv and Ghost Bottleneck are used as the backbone networks to complete feature extraction, and reduce the complexity and redundancy of the model itself, and the dual attention mechanisms of SE and CBAM

are introduced to increase the tendency of the model feature extraction, and improve the detection precision and real-time detection efficiency. The experimental results show that the proposed algorithm achieves an improvement of detection precision and detection efficiency under the premise of being lightweight, and has strong discrimination and generalization ability, which can be widely applied in a multi-scene environment.

In future research and work, the multi-level classification of *Hemerocallis citrina Baroni* will be carried out. Through the accurate maturity detection of different maturity levels of *Hemerocallis citrina Baroni*, it will be able to play different edible and medicinal roles at different growth stages, and can then be fully exploited to enhance the economic benefits.

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

Funding: This research was funded by the Shanxi Provincial Philosophy and Social Science Planning Project, grant number 2021YY198; and Shanxi Datong University Scientific Research Yun-Gang Special Project (2020YGZX014 and 2021YGZX27).

Acknowledgments: The authors would like to thank the reviewers for their careful reading of our paper and for their valuable suggestions for revision, which make it possible to present our paper better.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Lin, Y.D.; Chen, T.T.; Liu, S.Y.; Cai, Y.L.; Shi, H.W.; Zheng, D.; Lan, Y.B.; Yue, X.J.; Zhang, L. Quick and accurate monitoring peanut seedlings emergence rate through UAV video and deep learning. *Comput. Electron. Agric.* **2022**, *197*, 106938. [[CrossRef](#)]
2. Perugachi-Diaz, Y.; Tomczak, J.M.; Bhulai, S. Deep learning for white cabbage seedling prediction. *Comput. Electron. Agric.* **2021**, *184*, 106059. [[CrossRef](#)]
3. Feng, A.; Zhou, J.; Vories, E.; Sudduth, K.A. Evaluation of cotton emergence using UAV-based imagery and deep learning. *Comput. Electron. Agric.* **2020**, *177*, 105711. [[CrossRef](#)]
4. Azimi, S.; Wadhawan, R.; Gandhi, T.K. Intelligent Monitoring of Stress Induced by Water Deficiency in Plants Using Deep Learning. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–13. [[CrossRef](#)]
5. Patel, A.; Lee, W.S.; Peres, N.A.; Fraisse, C.W. Strawberry plant wetness detection using computer vision and deep learning. *Smart Agric. Technol.* **2021**, *1*, 100013. [[CrossRef](#)]
6. Liu, W.; Wu, G.; Ren, F.; Kang, X. DFF-ResNet: An insect pest recognition model based on residual networks. *Big Data Min. Anal.* **2020**, *3*, 300–310. [[CrossRef](#)]
7. Wang, K.; Chen, K.; Du, H.; Liu, S.; Xu, J.; Zhao, J.; Chen, H.; Liu, Y.; Liu, Y. New image dataset and new negative sample judgment method for crop pest recognition based on deep learning models. *Ecol. Inf.* **2022**, *69*, 101620. [[CrossRef](#)]
8. Jiang, H.; Li, X.; Safara, F. IoT-based Agriculture: Deep Learning in Detecting Apple Fruit Diseases. *Microprocess. Microsyst.* **2021**, *91*, 104321. [[CrossRef](#)]
9. Orano, J.F.V.; Maravillas, E.A.; Aliac, C.J.G. Jackfruit Fruit Damage Classification using Convolutional Neural Network. In Proceedings of the 2019 IEEE 11th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM), Laoag, Philippines, 29 November–1 December 2019; pp. 1–6. [[CrossRef](#)]
10. Herman, H.; Cenggoro, T.W.; Susanto, A.; Pardamean, B. Deep Learning for Oil Palm Fruit Ripeness Classification with DenseNet. In Proceedings of the 2021 International Conference on Information Management and Technology (ICIMTech), Jakarta, Indonesia, 19–20 August 2021; pp. 116–119. [[CrossRef](#)]
11. Gayathri, S.; Ujwala, T.U.; Vinusha, C.V.; Pauline, N.R.; Tharunika, D.B. Detection of Papaya Ripeness Using Deep Learning Approach. In Proceedings of the 2021 3rd International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, India, 2–4 September 2021; pp. 1755–1758. [[CrossRef](#)]
12. Wu, D.; Wu, C. Research on the Time-Dependent Split Delivery Green Vehicle Routing Problem for Fresh Agricultural Products with Multiple Time Windows. *Agriculture* **2022**, *12*, 793. [[CrossRef](#)]
13. An, Z.; Wang, X.; Li, B.; Xiang, Z.; Zhang, B. Robust visual tracking for UAVs with dynamic feature weight selection. *Appl. Intell.* **2022**, *14*, 392–407. [[CrossRef](#)]

14. Kumar, A.; Joshi, R.C.; Dutta, M.K.; Jonak, M.; Burget, R. Fruit-CNN: An Efficient Deep learning-based Fruit Classification and Quality Assessment for Precision Agriculture. In Proceedings of the 2021 13th International Congress on Ultra-Modern Telecommunications and Control Systems and Workshops (ICUMT), Brno, Czech Republic, 25–27 October 2021; pp. 60–65. [[CrossRef](#)]
15. Widiyanto, S.; Wardani, D.T.; Wisnu Pranata, S. Image-Based Tomato Maturity Classification and Detection Using Faster R-CNN Method. In Proceedings of the 2021 5th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), Ankara, Turkey, 21–23 October 2021; pp. 130–134. [[CrossRef](#)]
16. Wu, H.; Cheng, Y.; Zeng, R.; Li, L. Strawberry Image Segmentation Based on U²-Net and Maturity Calculation. In Proceedings of the 2022 14th International Conference on Advanced Computational Intelligence (ICACI), Wuhan, China, 15–17 July 2022; pp. 74–78. [[CrossRef](#)]
17. Zhang, R.; Li, X.; Zhu, L.; Zhong, M.; Gao, Y. Target detection of banana string and fruit stalk based on YOLOv3 deep learning network. In Proceedings of the 2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE), Nanchang, China, 26–28 March 2021; pp. 346–349. [[CrossRef](#)]
18. Mohd Basir Selvam, N.A.; Ahmad, Z.; Mohtar, I.A. Real Time Ripe Palm Oil Bunch Detection using YOLO V3 Algorithm. In Proceedings of the 2021 IEEE 19th Student Conference on Research and Development (SCOREd), Kota Kinabalu, Malaysia, 23–25 November 2021; pp. 323–328. [[CrossRef](#)]
19. Wu, Y.J.; Yi, Y.; Wang, X.F.; Jian, C. Fig Fruit Recognition Method Based on YOLO v4 Deep Learning. In Proceedings of the 2021 18th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), Chiang Mai, Thailand, 19–22 May 2021; pp. 303–306. [[CrossRef](#)]
20. Zhou, X.; Wang, P.; Dai, G.; Yan, J.; Yang, Z. Tomato Fruit Maturity Detection Method Based on YOLOV4 and Statistical Color Model. In Proceedings of the 2021 IEEE 11th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER), Jiaxing, China, 27–31 July 2021; pp. 904–908. [[CrossRef](#)]
21. Jose, N.T.; Marco, M.; Claudio, F.; Andres, V. Disease and Defect Detection System for Raspberries Based on Convolutional Neural Networks. *Electronics* **2021**, *11*, 11868. [[CrossRef](#)]
22. Wang, J.; Wang, L.Q.; Han, Y.L.; Zhang, Y.; Zhou, R.Y. On Combining Deep Snake and Global Saliency for Detection of Orchard Apples. *Electronics* **2021**, *11*, 6269. [[CrossRef](#)]
23. Zhou, X.; Ma, H.; Gu, J.; Chen, H.; Deng, W. Parameter adaptation-based ant colony optimization with dynamic hybrid mechanism. *Eng. Appl. Artif. Intell.* **2022**, *114*, 105139. [[CrossRef](#)]
24. Chen, H.Y.; Miao, F.; Chen, Y.J.; Xiong, Y.J.; Chen, T. A Hyperspectral Image Classification Method Using Multifeature Vectors and Optimized KELM. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 2781–2795. [[CrossRef](#)]
25. Wang, C.Y.; Mark Liao, H.Y.; Wu, Y.H.; Chen, Y.H.; Hsieh, J.W.; Yeh, I.H. CSPNet: A New Backbone that can Enhance Learning Capability of CNN. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 14–19 June 2020; pp. 1571–1580. [[CrossRef](#)]
26. 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*, 346–361. [[CrossRef](#)] [[PubMed](#)]
27. Zhao, H.; Liu, J.; Chen, H.; Li, Y.; Xu, J.; Deng, W. Intelligent diagnosis using continuous wavelet transform and gauss convolutional deep belief network. *IEEE Trans. Reliab.* **2022**, *2022*, 1–11. [[CrossRef](#)]
28. Han, K.; Wang, Y.; Tian, Q.; Guo, J.; Xu, C. GhostNet: More Features from Cheap Operations. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 14–19 June 2020; pp. 1577–1586. [[CrossRef](#)]
29. Hu, J.; Shen, L.; Albanie, S.; Sun, G.; Wu, E. Squeeze-and-Excitation Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 2011–2023. [[CrossRef](#)] [[PubMed](#)]
30. Yao, R.; Guo, C.; Deng, W.; Zhao, H. A novel mathematical morphology spectrum entropy based on scale-adaptive techniques. *ISA Trans.* **2022**, *126*, 691–702. [[CrossRef](#)] [[PubMed](#)]
31. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the 15th European Conference on computer vision (ECCV), Munich, Germany, 8–14 September 2018; Springer: Cham, Switzerland, 2018; pp. 3–19.
32. Deng, W.; Ni, H.; Liu, Y.; Chen, H.; Zhao, H. An adaptive differential evolution algorithm based on belief space and generalized opposition-based learning for resource allocation. *Appl. Soft Comput.* **2022**, *127*, 109419. [[CrossRef](#)]