electronics

MDPI

Article

Real-Time Detection of Mango Based on Improved YOLOv4

Zhipeng Cao and Ruibo Yuan *

Faculty of Mechanical and Electrical Engineering, Kunming University of Science and Technology, Kunming 650031, China
* Correspondence: kmust_yrb@163.com

Abstract: Agricultural mechanization occupies a key position in modern agriculture. Aiming at the fruit recognition target detection part of the picking robot, a mango recognition method based on an improved YOLOv4 network structure is proposed, which can quickly and accurately identify and locate mangoes. The method improves the recognition accuracy of the width adjustment network, then reduces the ResNet (Residual Networks) module to adjust the neck network to improve the prediction speed, and finally adds CBAM (Convolutional Block Attention Module) to improve the prediction accuracy of the network. The newly improved network model is YOLOv4-LightC-CBAM. The training results show that the mAP (mean Average Precision) obtained by YOLOV4-LightC-CBAM is 95.12%, which is 3.93% higher than YOLOv4. Regarding detection speed, YOLOV4-LightC-CBAM is up to 45.4 frames, which is 85.3% higher than YOLOv4. The results show that the modified network can recognize mangoes better, faster, and more accurately.

Keywords: object detection; YOLOv4; width reduction; convolutional block attention module; feature extraction

Citation: Cao, Z.; Yuan, R. Real-Time Detection of Mango Based on Improved YOLOv4. *Electronics* **2022**, *11*, 3853. https://doi.org/10.3390/electronics11233853

Academic Editor: Silvia Liberata Ullo

Received: 27 September 2022
Accepted: 5 November 2022
Published: 23 November 2022

1. Introduction

China is a major fruit producer and consumer in the world. With the development of society, fewer and fewer people are engaged in the management and picking of orchards, and the labor force shortage will lead to a lack of productivity. However, robots can significantly reduce the labor force shortage. The research on picking robots was first carried out in the 1980s. Research has been conducted on machine vision, agricultural robots, remote sensing analysis, and fruit quality detection. Target detection and fruit recognition based on deep learning and computer image processing have the advantages of high efficiency, high precision, and low labor cost. In recent years, visual technology has been gradually applied to fruit identification and inspection in China. In other words, the image collected by the robot identifies and locates the fruit in the image through the object detection algorithm, and transmits the position information to the subsequent acquisition work.

With the development of technology, the performance of GPU has been dramatically improved, the neural network recognition technology has been iterated and updated, and the target recognition network has been updated continuously. More and more scholars also use a neural network to identify fruits. Faster R-CNN is a two-stage target recognition neural network proposed by Microsoft, which can achieve better recognition accuracy. Line et al. [1] applied Faster R-CNN to strawberry flower recognition, which could achieve a better recognition effect in different scenes of strawberry flowers. They could provide a reference for outdoor strawberry yield. Wan et al. [2] proposed an improved version of Faster R-CNN, which optimized the convolution layer and pooling layer structure, detected multiple kinds of fruits, and obtained a higher accuracy than the original algorithm. Parvathi et al. [3] proposed using ResNet-50 Faster R-CNN to see two critical ripening stages of coconuts in a complex background, which can identify young coconuts and mature coconuts, respectively. Zhao et al. [4] proposed a new Faster R-CNN to detect strawberry crop diseases and obtained an mAP of 92.18%.

The picking robot needs real-time picking. Since the recognition speed of Faster R-CNN still cannot reach real-time, the birth of a single-stage detection network can provide conditions for the real-time selection of the picking robot. Both single-stage detection networks are SSD (Single Shot MultiBox Detector) and YOLO (You Only Look Once). In 2016, Redmon et al. [5–7] proposed YOLO neural network, which can quickly identify objects, and then proposed YOLOv2 and YOLOv3 in 2017 and 2018, respectively. Using the neural network framework of the YOLO series, Liu et al. [8] proposed an improved version of Yolov3, which could improve the robustness of tomato detection, and the AP value reached 96.40%. Additionally for tomato recognition, the YOLOv3 deep learning model proposed by Wang et al. [9] estimates the yield of tomatoes in plant factories, and can also identify the red and green fruits of tomatoes. The mAP of this model is 2.7% higher than that of the traditional YOLOv3 model, reaching 99.3%. In terms of YOLOv3 improvement, Liu et al. [10] added DenseNet (Densely Connected Network) to the trunk network, optimized the feature layer, and finally calculated the three-dimensional position of the pineapple in the natural environment through binocular stereo vision. In 2020, Bochkovskiy et al. [11] proposed YOLOv4. In 2021, Gai et al. [12] proposed an improved version of YOLOv4, combined with DenseNet interlayer density, and offered a YOLOv4-dense model, which can effectively detect cherries and the maturity of cherries. Fan et al. [13] improved the detection accuracy of apple sorting through NIR (Near-Infrared) images and pruning YOLOv4. Wang et al. [14] proposed DSE-YOLO, which extracts rich details and semantic features through the DSE (Detail-Semantics Enhancement) module to improve the recognition accuracy of strawberries. SSD is a network proposed in 2016, with high precision and fast speed, and can be applied to fruit recognition. Liang et al. [15] suggested that SSD was used to detect mangoes on mango trees, and F1 (F1-Score) could reach 0.911 when FPS (Frames Per Second) was 35. Vasconez et al. [16] used SSD with MobileNet in 2020 to test three fruits: Hass avocado and lemon (both from Chile), and apple (from CA, USA), and achieved a 90% fruit count. Anagnostis et al. [17] using SSD to classify anthracnose infected trees in walnut orchards, could identify whether leaves were infected on the tree and validated datasets with 87% correct classification.

The computing power of some edge devices is insufficient, and the large network cannot achieve a real-time detection effect, so the lightweight network can be better applied to edge devices. The light network of the YOLO [18–22] series also has a good effect when applied to fruit identification and counting. Other neural networks improve SSD [23–25] with lightweight to improve detection speed.

The picking process of the picking robot first reads the photos from the camera, then uses the neural network to detect the target, and finally controls the picking mechanism to pick the fruit. Although YOLOv4 can reach 24.5 frames, the picture does not need to be read repeatedly; only the picture is detected, and there is no other calculation in the middle, so a detection network with higher, faster detection accuracy is required. In this paper, while reducing the amount of computation and improving the accuracy of neural network, the following modifications are made:

1. Adjust the width of the backbone network and reduce it by half.
2. Reduce the depth of the backbone network and modify the residual blocks in the CSP_Darknet module to layers 1, 2, 4, 4, 2.
3. The neck network is modified to remove the convolutional layer, leaving only one convolutional layer module.
4. Attention mechanism is added to the channel from backbone network to neck network to improve the detection accuracy of the whole neural network.

Experimental comparisons at each point were designed to evaluate performance.

2. Materials and Methods

The pictures were collected at the Mango Base in Panzhihua, Sichuan Province. Professional cameras (intel Realsense D435) were used to shoot and sample. In order to ensure the diversity of data, the following points should be paid attention to when shooting:

1. In order to ensure the diversity of data, the database consisted of many different individual hanging fruits from different trees.
2. The shooting time was at night. In order to ensure observable mango fruits, illuminating mangoes with LED lights.
3. In order to obtain more characteristic information, the mangoes images should include the presence of complicating factors, such as occlusion. The database should also be sufficiently large.

The data sample in this paper is 1700 pieces, and the image size is all scaled to the resolution of 612 × 512 by scaling. This paper divides 1700 images into three parts: the test set, accounting for 10%; the validation set, accounting for 9%; the training set, accounting for 81%. As shown in Figure 1, part of the picture sample.

Figure 1. A partial sample of the original image.

3. YOLOv4 and the Adjusted Network

3.1. YOLOv4 Model

As shown in Figure 2, YOLOv4 is an upgraded version of a more accurate target recognition network proposed after YOLOv3. YOLOv4 is divided into three parts: backbone network, neck network, and head network: CSP_Darknet53 is used in the backbone network; The neck network contains SPP (Spatial Pyramid Pooling) and PANet (Path Aggregation Network), which continue to obtain more information about mango shape, color, and other characteristics. The head network is the identification key of the YOLO series, and is used to detect the target and location of an image.

For some resource-constrained places, the computation amount and weight of YOLOv4 are still huge. Therefore, lightweight modification of YOLOv4 can reduce the computation amount and weight file size, save computing resources, and provide real-time picking basis for picking robots.

3.2. Network after Adjustment

3.2.1. Adjust the Width of the Trunk Network

The backbone network of YOLOv4 is CSP_Darknet53, and the maximum width of the backbone network is 1024. The width is the number of channels in the neural network, and the larger the number of channels, the more noise will be learned. Due to the single orchid background of the experimental data, too many channels will make the neural network learn more noise, resulting in the reduction of accuracy, so it is necessary to reduce the number of channels. Therefore, the first lightweight network is to reduce the width of the network, the backbone network CSP_Darknet53 channel number from 64, 128, 256, 512, 1024 to 32, 64, 128, 256, 512.

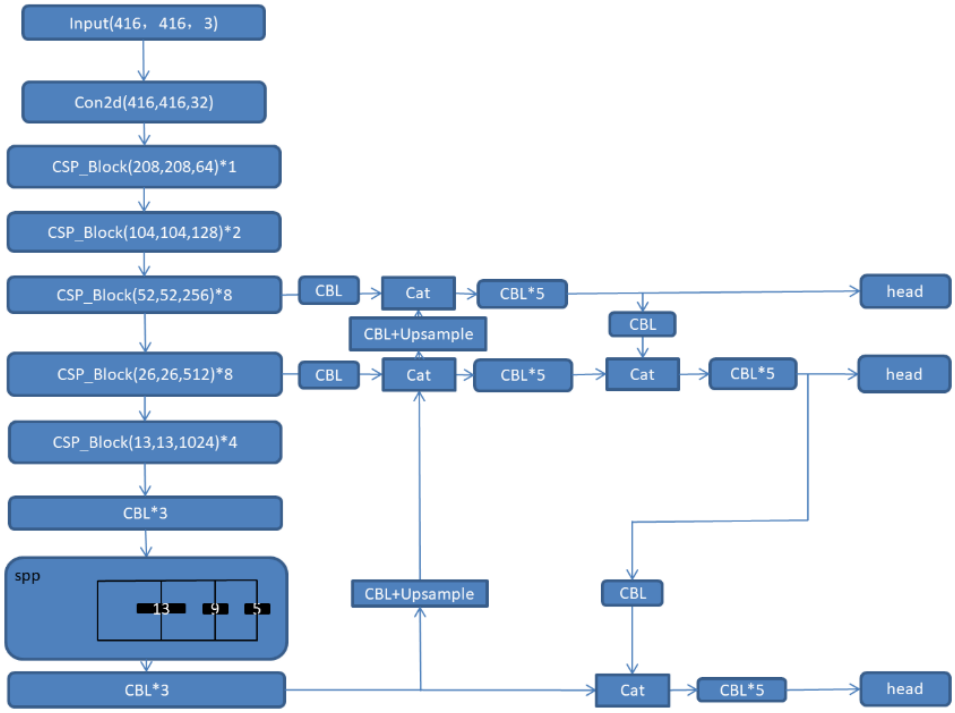

Figure 2. Block diagram of YOLOv4.

3.2.2. Backbone Network Depth

The backbone network is composed of five CSP_Darknet modules. The structure of the CSP_Darknet module is shown in Figure 3. It starts with the CBM module and then divides into two channels, one of which is conducted backward to the two CBM modules, there is a ResNet module in the middle of the two CBM modules, and the other one is conducted backward to the CBM module. Then the Cat module is used to merge the two channels together, and then the two channels are conducted to the CBM module. The CBM module is Conv + BN + Mish, and is composed of normal convolution Conv plus Batch Normalization and Mish activation functions. The ResNet module is divided into two branches, one consisting of two CBM modules, the other being direct backward conduction, with the two channels then being added.

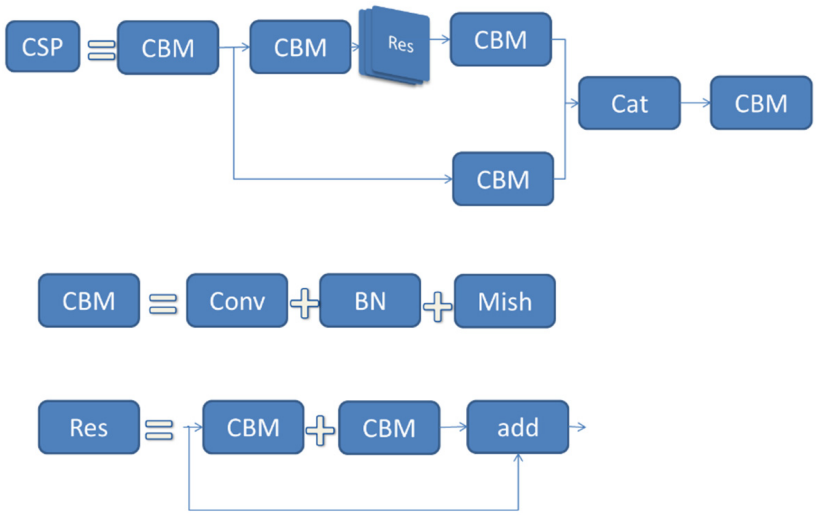

Figure 3. Structure diagram of CSP.

The deeper the network is, the better the fit will be, but it will fall into the local optimum and fail to obtain a better solution. When the number of layers of a neural network reaches a specific number, increasing the number of layers cannot improve the recognition accuracy of the neural network. Therefore, reducing the number of layers of neural networks can achieve higher detection speed without reducing the accuracy. The number of ResNet for each CSP_Darknet module will be different, which are 1, 2, 8, 8, 4 in YOLOv4. The number of ResNet is modified to 1, 2, 4, 4, 2 to ensure accuracy and speed.

3.2.3. Modifying the Neck Network

As shown in the red dashed box in Figure 4, in order to further reduce the computation amount and weight of the network, the convolution of the neck network is removed further to reduce the computation amount and weight of the network. In the 13 × 13 feature layer channel, a CBL module is used to replace the SPP and the convolution module before and after SPP. After the CBL module, two directions are generated: one order is directly transmitted to the back of the YOLO head, and the other direction is merged by the up-sampling channel and the channel of the 26 × 26 feature layer. After joining, the different orders are divided into two directions. The first direction is directly transmitted to the back of the YOLO head, successively transferred to the 52 × 52 feature layer.

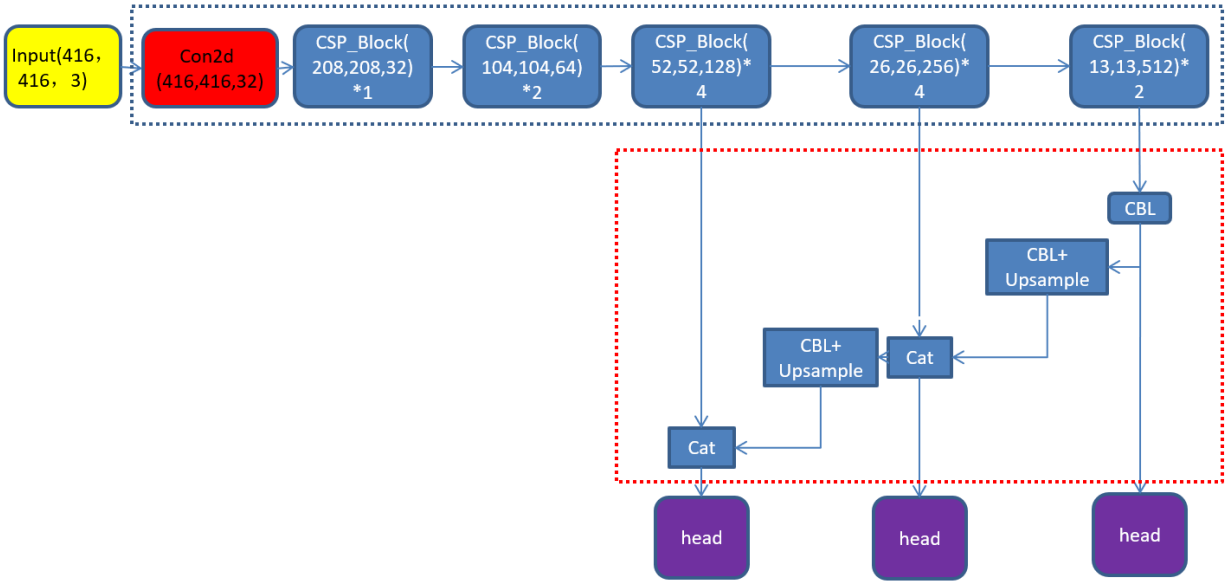

Figure 4. Network of neck modification.

3.2.4. Add Attention Mechanism

Because the computing power of computers is still insufficient, when the input information is too complex, more complex models are needed to identify the information.

The introduction of attention can reduce the size of the model and the amount of calculation. The function of concentration can optimize the overall network's performance and improve the network's detection accuracy with a small amount of computation. CBAM attention is used in this paper, as shown in Figure 5:

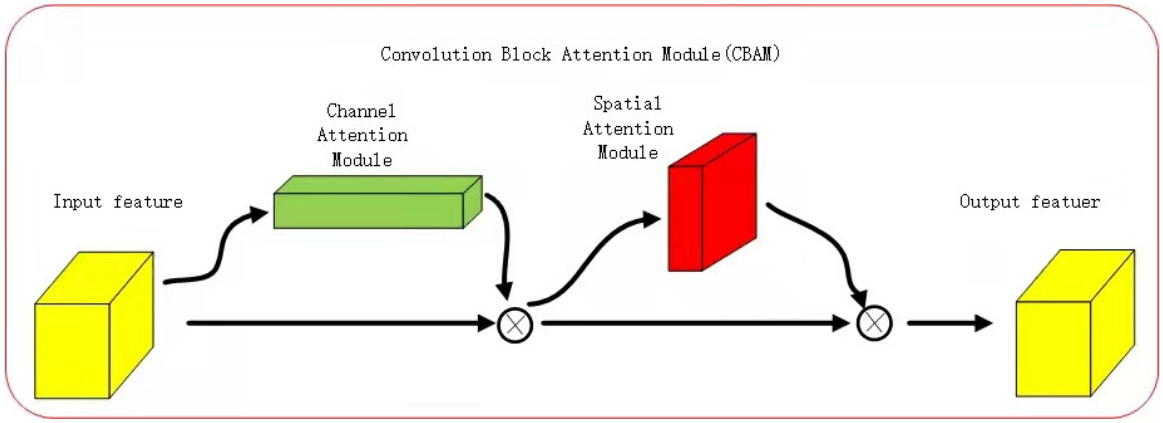

Figure 5. Structure diagram of the CBAM module.

3.2.5. New Network

As shown in Figure 6, attention is added to the three channels leading from the backbone network, and then an attention mechanism is added to the back of each upsampling module.

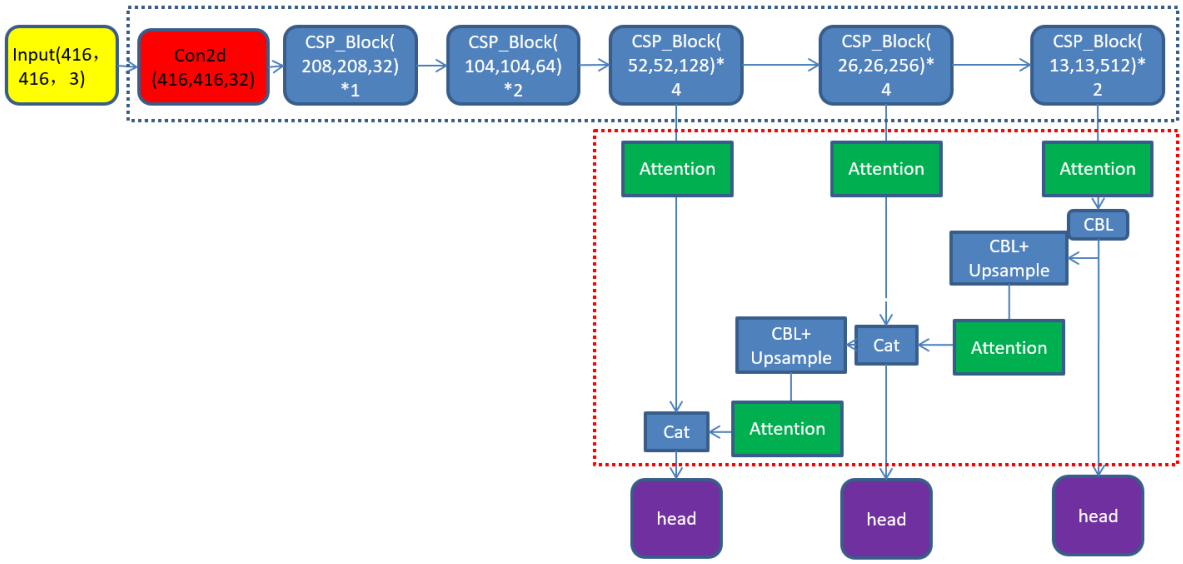

Figure 6. The network after the Add attention mechanism modification.

4. Experimental Results and Analysis

The training hardware used in this paper are: processor is Inter (R) Core (TM) I5-10400 CPU @ 2.90ghz, GPU is NVIDIA GeForce RTX 3060, and video memory 12GB. This software was compiled in Pycharm compiler, using Python3.7, Pytorch is 1.7.1, and CUDA is 11.3. The system is Win10.

The test uses a notebook. The PC is configured with Intel (R) Core (TM) I7-9750H CPU @ 2.60ghz 2.59ghz, and the GPU is NVIDIA GeForce GTX 1660 Ti. The system is WIN10. The network programming environment is Python3.7, and Pytorch1.7.1 version is used, confidence is 0.5, and ioU 0.5 is used as mAP0.5 calculation.

4.1. Model Training

In order to maintain consistency, the image size of YOLOv4 and YOLOv4's modification network input is kept as 416 × 416, batch size is 8, and the number of training rounds is 300. Cuda11.3 accelerate training, and each round's training set loss and validation set loss are recorded to save the training weight.

4.2. Evaluation Index

The performance of a neural network is evaluated by AP (Average Precision), mAP, F1, Recall, Precision, and FPS.

The expression formula of Recall and Precision means that three quantities should be introduced: TP (True Positives), FP (False Positives), and FN (False Negatives). TP is the quantity of positive samples detected as positive samples. FP is the number of negative samples detected as positive samples. FN is the number of negative samples detected as negative samples.

The Precision value can be calculated using the following formula:

$$\text{Precison} = \frac{TP}{TP + FP} \tag{1}$$

The calculation formula for Recall value is as follows:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2}$$

F1 is a relation between Recall and Precision, and the formula of F1 is as follows.

$$\text{F1} = \frac{2 \times \text{Precsion} \times \text{Recall}}{\text{Precsion} + \text{Recall}} \tag{3}$$

The AP value is the average of the precision values on the P-R curve. In the P-R curve, P is Precision on the vertical axis and R is Recall on the horizontal axis. In Equation (4), $p(r)$ is the functional relation between P and R in P-R curve, and r is the corresponding horizontal axis. mAP is the average of AP values of all targets. The formula is as follows:

AP value can be calculated by the following formula:

$$\text{AP} = \int_0^1 p(r)dr \tag{4}$$

mAP value can be calculated using the following formula:

$$\text{mAP} = \sum_{\text{i}=0}^{\text{n}} \text{AP}_\text{i} \tag{5}$$

This paper has only one objective, so mAP and AP are equal, and the following evaluation index uses mAP value. FPS refers to the number of pictures recognized by the neural network in one second. In this paper, the calculation time of recognizing 100 pictures is adopted, and then the FPS is calculated.

4.3. Experimental Results

4.3.1. Width Adjustment Experiment

Whether the width is appropriate will affect the accuracy of target recognition. In this experiment, the trunk network and the neck network are divided into two independent width networks, and the number of channels of the trunk network and the neck network is halved, respectively. The most appropriate width is selected by testing. A network with half the number of trunk network channels but the same number of neck network channels is called YOLOv4-bo. A network with the same number of trunk network channels but half the number of neck network channels is called YOLOv4-ne. A network with half the number of both trunk network and neck network channels is called YOLOv4-half. The experimental comparison results are shown in Table 1:

Table 1. Comparison table of width experiment.

Model	F1	Recall/%	Precision/%	mAP/%	FPS	Model Size/M
YOLOv4	0.93	91.64	95.18	91.32	24.5	244.4
YOLOv4-bo	0.95	96.42	92.97	95.92	29.2	166.9
YOLOv4-ne	0.94	94.69	93.21	94.24	29.5	138.1
YOLOv4-half	0.94	95.62	92.79	95.06	32.8	61.3

The mAP, F1 and Recall values of the network YOLOv4-bo and YOLO-bo after the number of trunk network channels is halved are the highest, and the mAP value is 4.6% higher than YOLOv4. According to the modified network above, reducing the number of channels can improve the mAP value. Reducing the number of channels in the backbone network can improve the mAP value more than the reduction of the number of channels in the neck network. According to the above modified network comparison, reducing the number of channels can improve the mAP value. Removing the number of channels in the backbone network can improve the mAP value more than reducing the number of channels in the neck network. Reducing the number of channels can improve the accuracy and increase the neural network's detection and inference speed. Therefore, in terms of overall performance evaluation, comprehensive detection accuracy, detection inference speed and weight value, YOLOv4-half has the best performance.

4.3.2. Adjustment Experiment of Trunk Depth

Based on the previous comparison experiment, the depth of the backbone network is modified. The deeper the backbone network is, the better it is. The deeper the neural network is, the better the accuracy, but the gradient will disappear during training. Therefore, the layers of the backbone network are adjusted in this paper. In order not to damage the structure of the CSP_Darknet module of the backbone network, only the layers of the residual network in the module are adjusted. The CSP_Darknet53 of YOLOv4's backbone network is divided into five modules, and the number of residual network layers of the modules is 1,2,8,8,4. For this reason, the last three are changed to (6,6,3), (4,4,2), (2,2,1). Based on the previous experiment, the adjusted network is named as YOLOv4-half-6, YOLOv4-half-4 and YOLOv4-half-2 respectively. From high to low and conduct comparative experiments, the experimental results are shown in Table 2:

Table 2. Comparison table of depth adjustment experiment.

Model	F1	Recall/%	Precision/%	mAP/%	FPS	Model Size/M
YOLOv4-half	0.94	95.62	92.79	95.06	32.8	61.3
YOLOv4-half-6	0.94	94.36	93.93	93.90	35.0	57.2
YOLOv4-half-4	0.94	95.62	93.15	95.08	38.0	53.1
YOLOv4-half-2	0.94	95.49	92.78	94.97	44.2	49.0

The highest overall mAP value is YOLOv4-half-4. In this experiment, the detection and inference speed FPS of the four models from high to low are 32.8, 35.0, 38.0, and 44.2, respectively. Among them, the YOLOv4-half-2 network model has the highest FPS, the YOLOv4-half-4 network model has the second fastest, and the YoloV4-half-2 network model has the smallest. In conclusion, the final modified network in this paper should have accuracy, detection, and inference speed, and the YOLOv4-half-4 model has the best performance.

4.3.3. Shear Comparison Experiment of Neck Network

It can be seen from the previous experiment that the optimal residual network layer is 1,2,4,4,2 in the backbone network. In order to further reduce the computational load and the size of the network, the neck network of YOLOv4 is modified to remove a large amount of convolution modules in the neck network, and the network that retains the SPP structure

is called YOLOv4-half-4-Ls. The network that removes SPP structure and replaces it with a CBL module is YOLOv4-half-4-Lc. Yolov4-half-4-L is the network except for the SPP structure. As shown in Table 3, the following comparative experiments were conducted:

Table 3. Comparison table of neck network shear comparison experiment.

Model	F1	Recall/%	Precision/%	mAP/%	FPS	Model Size/M
YOLOv4-half-4	0.94	95.62	93.15	95.08	38.0	53.1
YOLOv4-half-4-Ls	0.94	95.03	9281	94.46	48.1	38.0
YOLOv4-half-4-Lc	0.94	95.89	92.93	95.29	51.6	31.0
YOLOv4-half-4-L	0.94	95.42	93.08	94.79	53.0	35.6

In terms of detection speed, YOLOv4-half-4-Lc has the highest FPS value, but mAP is 0.5% lower than the highest one. In terms of model size, the four models have been reduced compared with the original YOLOv4 model, among which the model size of YOLOv4-half-4-Lc is the smallest. The overall performance of the YOLOv4-half-4-Lc model is better than that of other models.

4.3.4. Comparative Experiment of Attention Mechanism

The algorithm in this paper also hopes to improve neural networks' accuracy and recognition rate for target recognition with the minimum impact on speed. Therefore, attention mechanisms are added in the neck network layer, including Senet (Squeeze-and-Excitation Networks) [26], CAnet (Class-Agnostic Segmentation Networks With Iterative Refinement and Attentive Few-Shot Learning) [27], ECAnet (Efficient Channel Attention for Deep Convolutional Neural Networks) [28], CBAM [29], and FCAnet (Frequency Channel Attention Networks) [30]. The network models are YOLOv4-LightC-SE, YOLOv4-LightC-CA, YOLOv4-LightC-ECA, YOLOv4-LightC-CBAM, and YOLOv4-LightC-FCA. The five networks are compared, and the experimental results are shown in Table 4:

Table 4. Comparison table of attention mechanism.

Model	F1	Recall/%	Precision/%	mAP/%	FPS	Model Size/M
YOLOv4-half-4-Lc	0.94	95.89	92.93	95.29	51.6	31.0
YOLOv4-LightC-SE	0.94	95.89	92.87	95.30	48.1	31.1
YOLOv4-LightC-CA	0.94	95.36	92.48	94.72	44.9	31.1
YOLOv4-LightC-ECA	0.94	95.56	93.09	94.90	50.8	31.0
YOLOv4-LightC-CBAM	0.94	96.09	92.12	95.39	45.4	31.2
YOLOv4-LightC-FCA	0.94	94.89	93.53	94.34	49.9	31.1

The different algorithm of five attention mechanisms are compared in this paper, and there are five methods for comparison. CBAM, the best attention mechanism in this paper, has the highest mAP. Therefore, in terms of detection accuracy, the overall performance of YOLOv4-LightC-CBAM meets the requirements of good.

4.3.5. Comparison Experiment with Other Algorithms

As shown in Table 5, the modified network is compared with YOLOv4, Faster R-CNN, SSD, YOLOv3, and YOLOv4-Tiny.

Table 5. Comparison test table with other algorithms.

Model	F1	Recall/%	Precision/%	mAP/%	FPS	Model Size/M
YOLOv4	0.93	91.64	95.18	91.32	24.5	244.4
Faster R-CNN	0.81	97.28	69.72	94.48	1.9	108.1
SSD	0.87	79.84	95.40	79.33	40.2	90.6
YOLOv3	0.93	93.70	92.23	93.14	32.1	60.0
YOLOv4-Tiny	0.92	93.50	90.04	92.68	98.9	22.5
YOLOv4-LightC-CBAM	0.94	96.09	92.12	95.39	45.4	31.2

The detection speed of the improved network YOLOv4-LightC-CBAM in this paper can reach 45.4 frames, which is 85% higher than the 24.5 frames of YOLOv4. Based on the above analysis, YOLOv4-LightC-CBAM obtains the detection speed, and the overall performance is better than YOLOv4 and other models.

4.4. Result Analysis

This section will compare the actual detection effect of each network, and this paper will use two pictures for testing. As shown in Figure 7, the red box with the letter 'M' is the prediction box, the green circle is the missed target, the yellow triangle is the error detection target, and the blue pentagon is the error detection of multiple overlapping targets into one target. Combined with Section 4.3.5, the SSD with the lowest mAP is the one with the highest missed detection. The second missing model is the YOLOv4 model, followed by YOLOv3 and YOLOv4-Tiny turn. The disappeared model is Faster R-CNN and proposed in this paper. There is a point that needs to be pointed out. Faster R-CNN with no missed detection is incorrectly detected. It is framed in the yellow triangle in the figure, and the trunk is identified as a mango. YOLOv3 is less missed detection. YOLOv3 not only has error detection, but also detects multiple overlapping targets as one target. According to the effect comparison figure, the YOLOv4-LightC-CBAM proposed in this paper can better identify mangoes. Not only does it not miss detection, but it has a good discrimination degree for overlapping targets.

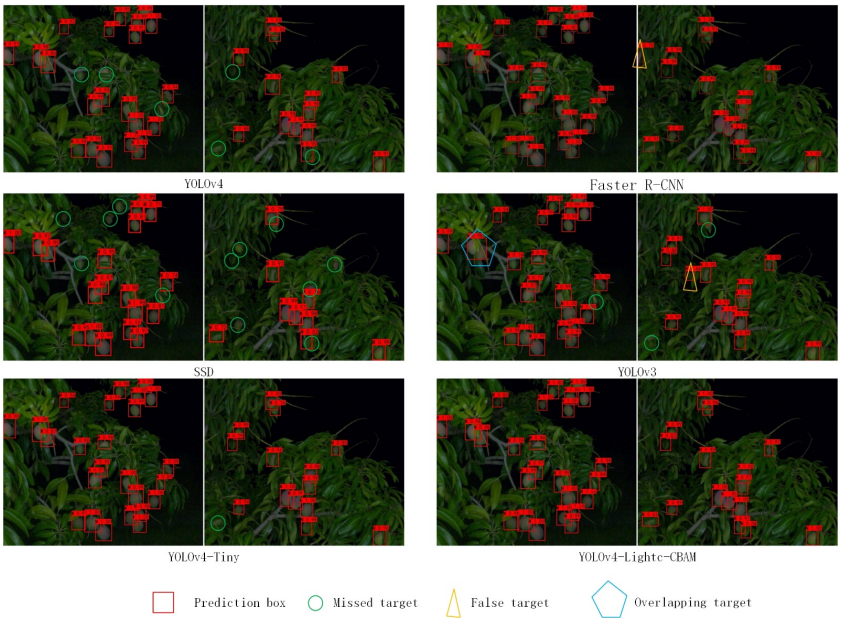

Figure 7. The experimental results of different algorithms have a total of 6 groups. The left side from the top to bottom is YOLOv4, SSD, YOLOV4-Tiny, respectively. The right side from top to bottom is Faster R-CNN, YOLOv3, YOLOV4-light-CBAM, respectively.

5. Conclusions

In this paper, a new object detection model, YOLOv4-LightC-CBAM, is proposed for real-time detection of picking robots. This model is improved on YOLOv4. The width of backbone network is reduced, the depth of backbone network is reduced, the neck network is changed, and the attention mechanism is increased. The results show that reducing the width of the backbone network can reduce the noise of the whole network learning, so that the dataset with a single background can obtain higher accuracy. The depth reduction of the backbone network can improve the detection speed of the network without reducing the mAP. The changes to the neck network result in a huge increase in the overall network detection speed and can improve the mAP. In this paper, the improved target detection model YOLOv4-LightC-CBAM is respectively compared with YOLOv4, Faster R-CNN, SSD, YOLOv3, and YOLOv4-Tiny five target detection networks. The five target detection models obtained were 91.32%, 94.55%, 79.33%, 93.14%, and 92.68% mAP, respectively. The model checking proposed in this paper obtained was 95.39% mAP, which is the highest mAP, and 4.07% higher than YOLOv4. The detection speed reached 45.4FPS, 85% higher than YOLOv4. The network model proposed in this paper achieves good results in terms of detection, accuracy and speed, and can provide a reliable and fast recognition algorithm for equipment with insufficient computing power, such as picking robots.

Author Contributions: Conceptualization, writing—review and editing, Z.C. and R.Y.; formal analysis, investigation, writing of the original draft, Z.C.; Funding Acquisition, R.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

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

1. Lin, P.; Lee, W.S.; Chen, Y.M.; Peres, N.; Fraisse, C. A deep-level region-based visual representation architecture for detecting strawberry flowers in an outdoor field. *Precis. Agric.* **2020**, *21*, 387–402. [CrossRef]
2. Wan, S.; Goudos, S. Faster R-CNN for multi-class fruit detection using a robotic vision system. *Comput. Netw.* **2020**, *168*, 107036. [CrossRef]
3. Parvathi, S.; Tamil Selvi, S. Detection of maturity stages of coconuts in complex background using Faster R-CNN model. *Biosyst. Eng.* **2021**, *202*, 119–132. [CrossRef]
4. Zhao, S.; Liu, J.; Wu, S. Multiple disease detection method for greenhouse-cultivated strawberry based on multiscale feature fusion Faster R_CNN. *Comput. Electron. Agric.* **2022**, *199*, 107176. [CrossRef]
5. 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.
6. 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.
7. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
8. Liu, G.; Nouaze, J.C.; Touko Mbouembe, P.L.; Kim, J.H. YOLO-Tomato: A Robust Algorithm for Tomato Detection Based on YOLOv3. *Sensors* **2020**, *20*, 2145. [CrossRef] [PubMed]
9. Wang, X.; Vladislav, Z.; Viktor, O.; Wu, Z.; Zhao, M. Online recognition and yield estimation of tomato in plant factory based on YOLOv3. *Sci. Rep.* **2022**, *12*, 8686. [CrossRef]
10. Liu, T.-H.; Nie, X.-N.; Wu, J.-M.; Zhang, D.; Liu, W.; Cheng, Y.-F.; Zheng, Y.; Qiu, J.; Qi, L. Pineapple (Ananas comosus) fruit detection and localization in natural environment based on binocular stereo vision and improved YOLOv3 model. *Precis. Agric.* **2022**. [CrossRef]
11. Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
12. Gai, R.; Chen, N.; Yuan, H. A detection algorithm for cherry fruits based on the improved YOLO-v4 model. *Neural Comput. Appl.* **2021**. [CrossRef]
13. Fan, S.; Liang, X.; Huang, W.; Jialong Zhang, V.; Pang, Q.; He, X.; Li, L.; Zhang, C. Real-time defects detection for apple sorting using NIR cameras with pruning-based YOLOv4 network. *Comput. Electron. Agric.* **2022**, *193*, 106715. [CrossRef]
14. Wang, Y.; Yan, G.; Meng, Q.; Yao, T.; Han, J.; Zhang, B. DSE-YOLO: Detail semantics enhancement YOLO for multi-stage strawberry detection. *Comput. Electron. Agric.* **2022**, *198*, 107057. [CrossRef]

15. Liang, Q.; Zhu, W.; Long, J.; Wang, Y.; Sun, W.; Wu, W. A Real-Time Detection Framework for On-Tree Mango Based on SSD Network. In *Proceedings of the Intelligent Robotics and Applications*; Springer: Cham, Switzerland, 2018; pp. 423–436.
16. Vasconez, J.P.; Delpiano, J.; Vougioukas, S.; Auat Cheein, F. Comparison of convolutional neural networks in fruit detection and counting: A comprehensive evaluation. *Comput. Electron. Agric.* **2020**, *173*, 105348. [CrossRef]
17. Anagnostis, A.; Tagarakis, A.C.; Asiminari, G.; Papageorgiou, E.; Kateris, D.; Moshou, D.; Bochtis, D. A deep learning approach for anthracnose infected trees classification in walnut orchards. *Comput. Electron. Agric.* **2021**, *182*, 105998. [CrossRef]
18. Fu, L.; Feng, Y.; Wu, J.; Liu, Z.; Gao, F.; Majeed, Y.; Al-Mallahi, A.; Zhang, Q.; Li, R.; Cui, Y. Fast and accurate detection of kiwifruit in orchard using improved YOLOv3-tiny model. *Precis. Agric.* **2021**, *22*, 754–776. [CrossRef]
19. Magalhães, S.A.; Castro, L.; Moreira, G.; dos Santos, F.N.; Cunha, M.; Dias, J.; Moreira, A.P. Evaluating the Single-Shot MultiBox Detector and YOLO Deep Learning Models for the Detection of Tomatoes in a Greenhouse. *Sensors* **2021**, *21*, 3569. [CrossRef]
20. Gao, F.; Fang, W.; Sun, X.; Wu, Z.; Zhao, G.; Li, G.; Li, R.; Fu, L.; Zhang, Q. A novel apple fruit detection and counting methodology based on deep learning and trunk tracking in modern orchard. *Comput. Electron. Agric.* **2022**, *197*, 107000. [CrossRef]
21. Li, D.; Sun, X.; Elkhouchlaa, H.; Jia, Y.; Yao, Z.; Lin, P.; Li, J.; Lu, H. Fast detection and location of longan fruits using UAV images. *Comput. Electron. Agric.* **2021**, *190*, 106465. [CrossRef]
22. Xu, Z.F.; Jia, R.S.; Liu, Y.B.; Zhao, C.Y.; Sun, H.M. Fast Method of Detecting Tomatoes in a Complex Scene for Picking Robots. *IEEE Access* **2020**, *8*, 55289–55299. [CrossRef]
23. Zhou, Z.; Song, Z.; Fu, L.; Gao, F.; Li, R.; Cui, Y. Real-time kiwifruit detection in orchard using deep learning on Android™ smartphones for yield estimation. *Comput. Electron. Agric.* **2020**, *179*, 105856. [CrossRef]
24. Xiao, D.; Li, H.; Liu, C.; He, Q. Large-Truck Safety Warning System Based on Lightweight SSD Model. *Comput. Intell. Neurosci.* **2019**, *2019*, 2180294. [CrossRef] [PubMed]
25. Jiang, L.; Nie, W.; Zhu, J.; Gao, X.; Lei, B. Lightweight object detection network model suitable for indoor mobile robots. *J. Mech. Sci. Technol.* **2022**, *36*, 907–920. [CrossRef]
26. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
27. Zhang, C.; Lin, G.; Liu, F.; Yao, R.; Shen, C. CANet: Class-Agnostic Segmentation Networks With Iterative Refinement and Attentive Few-Shot Learning. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 5212–5221.
28. 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.
29. Woo, S.; Park, J.; Lee, J.-Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In *Proceedings of the Computer Vision–ECCV 2018*; Springer: Cham, Switzerland, 2018; pp. 3–19.
30. Qin, Z.; Zhang, P.; Wu, F.; Li, X. FcaNet: Frequency Channel Attention Networks. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 763–772.