agronomy

MDPI

Article

Dense Papaya Target Detection in Natural Environment Based on Improved YOLOv5s

Lei Wang [1], Hongcheng Zheng [2], Chenghai Yin [1,2], Yong Wang [2], Zongxiu Bai [1,*] and Wei Fu [2,*]

[1] College of Mechanical and Electrical Engineering, Shihezi University, Shihezi 832003, China; wl_mac@shzu.edu.cn (L.W.); 20202109042@stu.shzu.edu.cn (C.Y.)
[2] Mechanical and Electrical Engineering College, Hainan University, Haikou 570228, China; 22220854060042@hainanu.edu.cn (H.Z.); 21210802000018@hainanu.edu.cn (Y.W.)
* Correspondence: baizongxiu@stu.shzu.edu.cn (Z.B.); 994026@hainanu.edu.cn (W.F.)

Abstract: Due to the fact that the green features of papaya skin are the same colour as the leaves, the dense growth of fruits causes serious overlapping occlusion phenomenon between them, which increases the difficulty of target detection by the robot during the picking process. This study proposes an improved YOLOv5s-Papaya deep convolutional neural network for achieving dense multitarget papaya detection in natural orchard environments. The model is based on the YOLOv5s network architecture and incorporates the Ghost module to enhance its lightweight characteristics. The Ghost module employs a strategy of grouped convolutional layers and weighted fusion, allowing for more efficient feature representation and improved model performance. A coordinate attention module is introduced to improve the accuracy of identifying dense multitarget papayas. The fusion of bidirectional weighted feature pyramid networks in the PANet structure of the feature fusion layer enhances the performance of papaya detection at different scales. Moreover, the scaled intersection over union bounding box regression loss function is used rather than the complete intersection over union bounding box regression loss function to enhance the localisation accuracy of dense targets and expedite the convergence of the network model training. Experimental results show that the YOLOv5s-Papaya model achieves detection average precision, precision, and recall rates of 92.3%, 90.4%, and 83.4%, respectively. The model's size, number of parameters, and floating-point operations are 11.5 MB, 6.2 M, and 12.8 G, respectively. Compared to the original YOLOv5s network model, the model detection average precision is improved by 3.6 percentage points, the precision is improved by 4.3 percentage points, the number of parameters is reduced by 11.4%, and the floating-point operations are decreased by 18.9%. The improved model has a lighter structure and better detection performance. This study provides the theoretical basis and technical support for intelligent picking recognition of overlapping and occluded dense papayas in natural environments.

Keywords: papaya; target detection; YOLOv5s; lightweighting; coordinate attention

Citation: Wang, L.; Zheng, H.; Yin, C.; Wang, Y.; Bai, Z.; Fu, W. Dense Papaya Target Detection in Natural Environment Based on Improved YOLOv5s. *Agronomy* **2023**, *13*, 2019. https://doi.org/10.3390/agronomy13082019

Academic Editor: Roberto Marani

Received: 27 June 2023
Revised: 22 July 2023
Accepted: 28 July 2023
Published: 29 July 2023

1. Introduction

Papaya, known as the "king of all fruits" is extensively cultivated in tropical and warmer subtropical regions [1]. Manual picking of papaya is labour-intensive and costly, needing an autonomous picking robot to enhance picking efficiency, minimise fruit damage, and reduce labour costs. The accuracy of target recognition and positioning is crucial for determining the picking efficiency of a robotic system.

In recent years, artificial intelligence and machine vision have gained effectiveness and popularity in agricultural machinery due to advancements in computers and information technology [2]. Target recognition methods rely on colour feature segmentation to identify fruits in images containing background objects, such as leaves. These methods utilise pixel point information and incorporate colour spaces, such as RGB, HSI, and HSV. Colour-based recognition methods are suitable when the target fruit exhibits distinct colour

characteristics from the background object colour [3]. For example, Li et al. used colour features to distinguish pineapples from the background, achieving a correct recognition rate of 90% on sunny days and 60% on cloudy days [4]. To improve the accuracy of fruit recognition segmentation, Wei et al. proposed an automatic fruit object extraction method for vision systems in complex agricultural contexts. This method utilises an improved Otsu thresholding algorithm and a new function in the OHTA colour space to extract ripe fruits from complex agricultural backgrounds, achieving an extraction accuracy of over 95% [5]. Lv et al. [6] introduced a method for obtaining fruit, branch, and leaf regions from red apple images by extracting R-G images from colour images and applying threshold segmentation to identify fruit regions. Lin et al. [7] proposed a segmentation method that combines an AdaBoost classifier with texture-colour features. The proposed method achieved an accuracy of 0.867 and a recall of 0.768 for identifying citrus fruits. Wu et al. [8] presented a fruit point cloud segmentation method that incorporates colour and 3D geometric features. The viewpoint feature histogram of each point cloud cluster was used to remove nonfruit regions. Experimental results showed that the proposed method achieved a segmentation accuracy of 98.99% and a precision of 80.09%, surpassing traditional colour separation methods. The green features of papaya epidermis are the same colour as the leaves and the overlapping occlusion between the fruits. Based on the above traditional machine vision for recognition of papaya in the natural environment of the orchard there are limitations, for example, the method based on colour and texture features is constrained by the lighting conditions as well as the colour and shape of the fruits, and the geometric features are affected by the occlusion between the fruits.

Deep learning has emerged as a promising approach to overcome the limitations of traditional target recognition in image analysis. It is a subfield of machine learning that has witness the development of various architectures [9]. Gill et al. [10] proposed a scheme that combines type II fuzzy, teacher–learner-based optimisation and deep learning techniques—such as convolutional neural networks (CNN), recurrent neural networks, and long short-term memory (LSTM) networks—to enhance, segment, recognise, and classify fruit images. The proposed scheme demonstrates improved classification performance in feature selection, extraction, and classification compared with existing methods. Li et al. [11] proposed a method that integrates the improved YOLOv5s, improved DeepLabv3+ model and depth image information for the 3D localisation of longan picking points in complex natural environments. The experimental results showed that the accuracy of detecting longan bunches and main fruit branches was 85.50%, and the accuracy of semantic segmentation of main fruit branches was 94.52%. Zhang et al. [12] introduced a target identification and localisation scheme called GNPD-YOLOv5s, which is based on an improved version of YOLOv5s, to automatically identify obscured and unobscured peppers. The scheme incorporates lightweight optimisation of the Ghost module to prune and refine the model. The Ghost module uses a strategy of grouped convolutional layers and weighted fusion reduces the computational effort and generates most of the feature information. It enables the network structure to have multi-scale detection capability while maintaining depth. Experimental data demonstrated that the GNPD-YOLOv5s scheme reduces the number of floating-point operations by 40.9%, the model size by 46.6%, and the inference speed from 29 ms/frame to 14 ms/frame compared with the YOLOv5s model.

Although many studies solely utilise existing CNNs and ignore the distinctive features of fruit images, Min et al. addressed this problem by employing a multiscale attention network [13]. This network captures attention from different levels of the CNN and aggregates various visual attention features from different levels into a final integrated representation. Evaluations performed on the Fruits dataset show that MSANet has top-1 accuracies of 99.99% and 99.69% on both the Fruits-360 and FruitVeg-81 datasets, respectively. Pan et al. used a 3D stereo camera in combination with masked region convolutional neural network (Mask R-CNN) deep learning techniques to identify pears in complex orchard environments. The mean average precision (mAP) for pear fruit identification was 95.22% for Mask R-CNN and 99.45% for the test set [14]. Gai et al. [15] proposed a YOLOV4-DenseNet model,

which includes a densely connected cherry detection network, to address the challenge of recognising small target fruits. The YOLOV4-DenseNet model uses DenseNet rather than the CSPDarknet53 used in the original YOLOV4 and employs the Leaky ReLU activation function as the loss function to facilitate feature reuse and fusion, thereby improving the network's performance in cherry detection. The experimental results on cherry images demonstrate that the YOLOV4-DenseNet model achieves superior detection performance, enabling intelligent picking and improving the efficiency of picking robots. Wu et al. [16] proposed a method enhancing the YOLOv4 model using a channel-pruning algorithm for the detection of apple blossoms. The improved model achieved excellent lightweighting results, with a reduction of 96.74% in the number of parameters whilst maintaining accuracy. The comparison results showed that the mAP of apple blossom detection using the proposed method was 97.31%.

However, the complexity of orchard presents challenges for detection, including leaf shading, overlapping fruits, and insufficient lighting, which can affect the results. The RetinaNet model uses a focal loss function during the prediction process to address these issues. This function successfully solves the problem of imbalanced positive and negative samples during model training by adjusting the classification weights of different samples. It has been used to identify apples [17,18] and camellia oleifera fruit [19] in real orchard environments. The YOLO family of algorithms are end-to-end single-stage detection algorithms with high model accuracy that are easy to deploy on a picking robot mobile. Typical studies are: Wang et al. [20] proposed a lightweight and improved SSD model specifically for detecting Lingwu dates in the complex operating environment of a Lingwu-date-picking robot. The experimental results showed that the improved SSD model achieved a mAP of 96.6% in detecting Lingwu long dates. Zhang et al. [21] investigated the method of locating picking points under partial occlusion and proposed the grape cluster detection algorithm YOLO v5-GAP based on YOLO v5. The experimental results showed that YOLOv5-GAP achieved an average accuracy of 95.13%, which was 16.13%, 4.34%, and 2.35% higher than the algorithms YOLOv4, YOLOv5, and YOLOv7, respectively. This study contributes to the localisation of picking points in sheltered situations. Abeyrathna et al. developed a recognition system for apple orchards using an enhanced, complex training dataset. The system was evaluated using a deep learning algorithm built with CNNs. For counting apples, YOLOv5 and YOLOv7 showed a higher number of detections in outdoor dynamic conditions, reaching 86.6% accuracy [22]. Zhou et al. [23]. proposed the PSP-ellipse method for detecting dragon fruit endpoints. This approach involves localising and classifying dragon fruit using YOLOv7, segmenting dragon fruit with PSPNet, localising endpoints with an ellipse-fitting algorithm, and classifiying endpoints with ResNet. In dragon fruit detection, YOLOv7 achieved precision, recall, and average accuracy values of 0.844, 0.924, and 0.932, respectively. For endpoint detection, the accuracy of ResNet-based endpoint classification was 0.92 Guo et al. [24] proposed a peppercorn detection network based on the YOLOv5 target detection model. This model addressed challenges such as irregular shape and overlapping branches and leaves of peppercorns. The improved models achieved 5.4% and 4.7% higher accuracy than the original model, respectively. The proposed model accurately identifies mature peppercorns in their natural environment at a detection speed of approximately 89.3 frames/s. The above study provides research ideas for the detection of overlapping and occluded papayas in complex environments. At present, the YOLOV5 algorithm is mainly used for the aforementioned fruits, such as apples [25], lychees [26], dragon fruits [27], and other fruits. However, the green features of papaya epidermis are the same colour as the leaves and there is overlapping and occlusion. There are fewer studies on automatic detection of papayas.

In summary, this study improves the YOLOv5s network model for the problem of papaya epidermal green features being the same colour as leaves and occlusion between densely growing papayas. The main contributions are as follows: (1) incorporating the Ghost module in the backbone network to achieve a lightweight network and facilitate the deployment of mobile; (2) accurately detecting dense papayas, coordinate attention

modules are added to the network; (3) improving the performance of papaya detection at different scales by fusing a bidirectional weighted feature pyramid network in the PANet structure to the feature fusion layer; (4) the SIoU bounding box regression loss function is used to speed up the convergence of network model training and enhance the detection accuracy of dense papaya. This approach provides a theoretical basis and technical support for the intelligent picking and recognition of dense multitarget papayas.

2. Materials and Methods

2.1. Construction of the Papaya Dataset

In this study, images of papayas in modern standardised orchards were collected from a papaya plantation in Nanbin Farm, Yazhou District, Sanya City, Hainan Province. The images were captured in sunny and cloudy weather, during morning, midday, and afternoon, using a Nikon DSLR camera (resolution:4288 × 2848) and a smartphone (resolution: 4032 × 3024) at various shooting distances (0.5–1.5 m). In order to avoid duplication of image information, only one image was taken for each tree, and a total of 1000 papaya images were collected which contained information about the scene at different shooting distances (near, middle, and far) and different lighting (smooth, backlight, and shadow). Manual annotation of the data was performed using the LabelImg annotation tool before model training. All labelled objects are named papaya. The smallest outer rectangle of the entire papaya is used as the true box when labelling as a way to reduce the number of interfering pixels on the background inside the box. The labelling requirements for shaded papayas are to ensure that the size of the box matches the width of the target fruit to include the shaded fruit with maximum accuracy. However, papayas are not labelled when they are shaded by more than 2/3 of their area. After completing the annotation, an annotation file in .xml format was generated. The total number of papaya tags labelled in 1000 images was 8126. The random sampling method was used to divide the original dataset into training set, validation set and test set in the ratio of 8:1:1. The training and validation sets were used for model training and evaluation during a single training session, and the test set was employed to evaluate the final model's detection performance.

Training a deep learning model requires a large amount of data because a small training set can lead to overfitting of the neural network. Therefore, data augmentation techniques were employed to expand the sample size of the training set. In this paper, as shown in Figure 1, Figure 1a shows the original image, and Figure 1b–h shows the data enhancement using translation transformation, random rotation, random scaling, horizontal mirroring, splash noise, defocus blur, and motion blur. A total of 5600 images were obtained as the final training set after data augmentation.

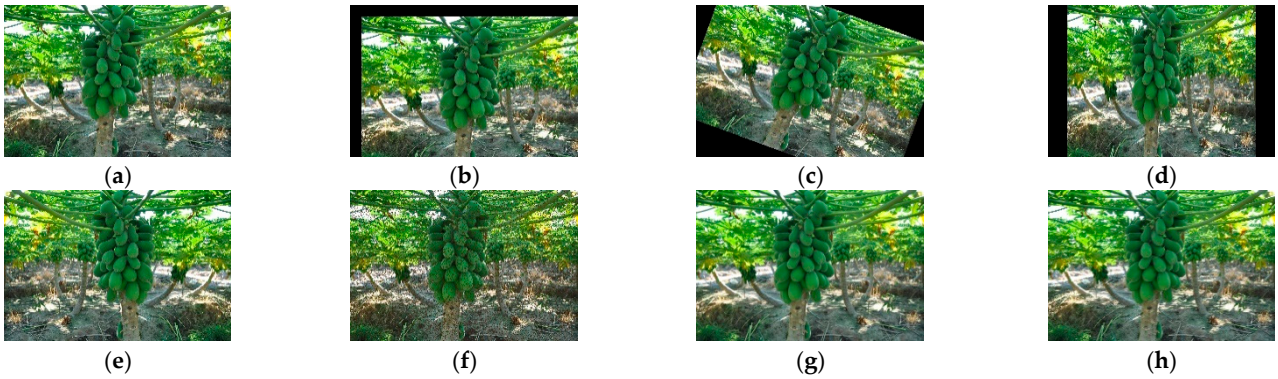


Figure 1. Papaya image data enhancement method. (**a**) Origina. (**b**) Translational transformation. (**c**) Random rotation. (**d**) Random scaling. (**e**) Horizontal mirroring. (**f**) Splash noise. (**g**) Defocus blur. (**h**) Motion blur.

2.2. Papaya Target Detection Network Model Construction

The you only look once (YOLO) algorithm is a representative of the first stage of target detection algorithms, known for its high detection accuracy and fast running speed. The YOLO algorithm has undergone several iterations, resulting in different architectures with varying numbers of network feature extraction modules and convolutional kernels. These iterations progressively increased the number of parameters and model size. In this study, the YOLOv5s architecture is used to improve the design of the papaya picking target detection network, considering accuracy, efficiency, and model complexity.

2.2.1. YOLOv5s Network Architecture

The YOLOv5s architecture comprises the input, backbone network, neck network and prediction head. The input side uses mosaic data augmentation, adaptive anchor frame calculation and adaptive image scaling. The backbone network utilises various feature extraction modules, such as Conv, C3, and SPPF. The SPPF role is to achieve the fusion of local and global features at the featherMap level. The neck network employs the PANet structure [28] for multiscale feature fusion and enhanced feature extraction. The prediction head uses complete intersection over union (CIOU) loss [29] to calculate the bounding box loss, and BCE loss to calculate the confidence loss and classification loss. The structure of the original YOLOv5s network model is shown in Figure 2.

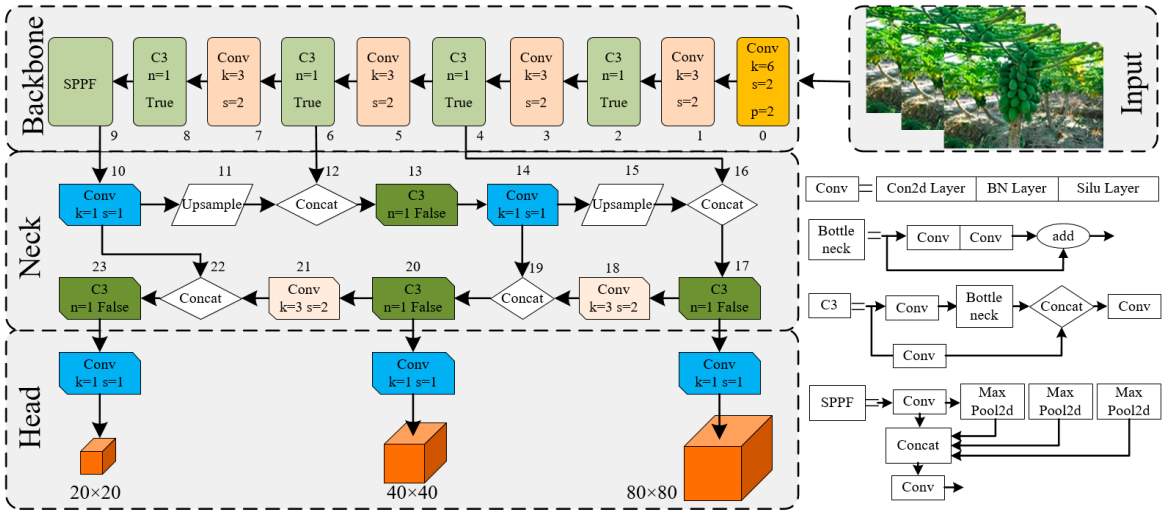


Figure 2. YOLOv5s network model structure.

2.2.2. Model Lightweighting Improvements

The Ghost module is a technique that can replace convolutional operations in traditional CNN networks, providing a means of implementing lightweight neural networks [30]. The Ghost module reduces the number of network model parameters and floating-point operations (FLOPs) whilst maintaining the algorithm's performance capability.

In the Ghost module, a ghost graph is generated by a linear operation Φ rather than ordinary convolution. As shown in Figure 3, assuming the input feature map is $h \times w \times c$, ordinary convolution is performed with n sets of $k \times k$ convolution kernels to obtain an output feature map of size $h' \times w' \times n$. In the Ghost model, convolution is performed with m sets of $k \times k$ convolution kernels to generate the $m \times h' \times w'$ eigenmap intrinsic, after which the eigenmap is linearly transformed Φ to produce the ghost graph, and the results of the splicing of the intrinsic and ghost graphs are output together. The use of the Ghost module reduces the computational and parameter volume compared with ordinary

convolution. The model acceleration ratio r_s and compression ratio r_c can be calculated as shown in Equations (1) and (2), respectively.

$$r_s = \frac{n \cdot h' \cdot w' \cdot c \cdot k \cdot k}{\frac{n}{s} \cdot h \cdot w' \cdot c \cdot k \cdot k + (s-1) \cdot \frac{n}{s} \cdot h' \cdot w' \cdot c \cdot d \cdot d} \approx s \tag{1}$$

$$r_c = \frac{n \cdot c \cdot k \cdot k}{\frac{n}{s} \cdot c \cdot k \cdot k + (s-1) \cdot \frac{n}{s} \cdot c \cdot d \cdot d} \approx s \tag{2}$$

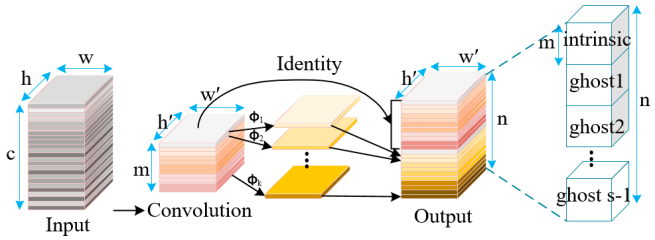
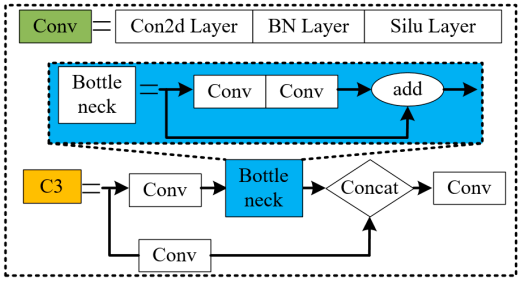
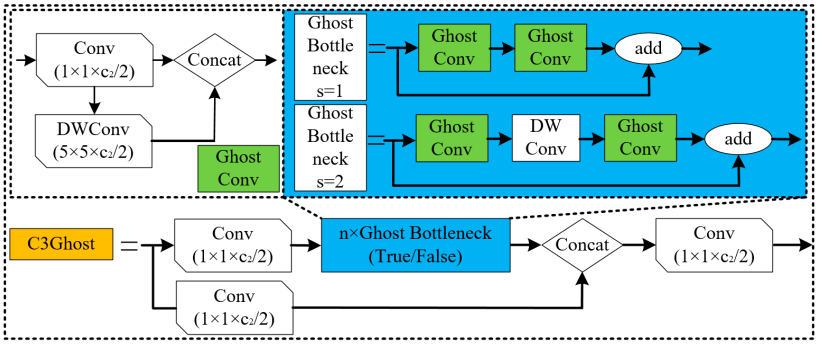


Figure 3. Schematic of the Ghost module structure.

From Equations (1) and (2), the Ghost module reduces the computational and parametric volume of the convolution process compared with normal convolution. In the original YOLOv5 network model, a large number of Conv and C3 modules are found in the backbone network layer, which involve a large number of operations and parameters. The model becomes lighter by using the Ghost module to improve the Conv and C3 modules in the backbone network layer of the YOLOv5s network model. The structure of the model before and after the improvement is shown in Figure 4 and is named GhostConv and C3Ghost, respectively.



(**a**)



(**b**)

Figure 4. Structure of C3 module before and after improvement. (**a**) C3 module. (**b**) GhostConv module with C3Ghost module. Notes: 1 × 1, 5 × 5 denotes the size of convolution kernel, $c_2/2$ denotes the number of channels, s denotes the stride value, n denotes the Ghost bottleneck value.

2.2.3. Introduction of Coordinate Attention Mechanism

The inference process of Yolov5 network model can occasionally lose information related to dense and small target features, resulting in poor detection performance for dense small targets. Existing attention mechanisms for lightweight networks, such as SENet [31] and CBAM module [32], have limitations. The SENet module focuses on building interdependencies between channels but ignores spatial features. The CBAM module attempts to extract spatial features using large-scale convolution kernels but ignores long-range dependencies. The coordinate attention mechanism is introduced to address these limitations. This mechanism encodes channel relationships and long-range dependencies through precise location information, achieved by coordinate information embedding and coordinate attention generation.

The structure of the coordinate attention module is shown in Figure 5. The coordinate information embedding takes input X. Each channel is encoded along horizontal and vertical coordinates using pooling kernels of size $(H, 1)$ or $(1, W)$, respectively. Thus, the output of the cth channel with height h can be expressed as Equation (3).

$$z_c^h(h) = \frac{1}{W} \sum_{0 \le i \le W} x_c(h, i) \tag{3}$$

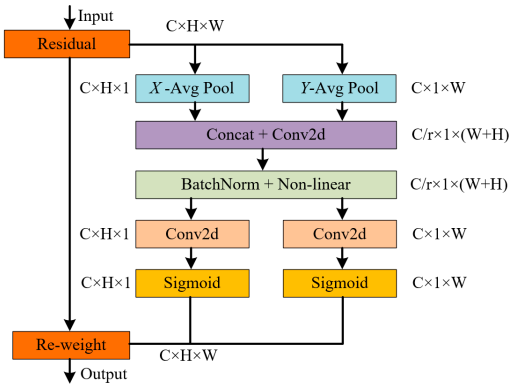


Figure 5. Structure of the coordinate attention module.

Similarly, the output of the cth channel of width w can be expressed as in Equation (4).

$$z_c^h(w) = \frac{1}{H} \sum_{0 \le i \le H} x_c(j, w) \tag{4}$$

These transformations aggregate features along each spatial direction, generating a pair of direction-aware feature maps. By doing so, the attention module can capture long-term dependencies along one spatial direction whilst preserving precise location information along the other, assisting the network in accurately locating the target of interest.

Coordinate attention generation aims to combine the width and height directional feature maps of the acquired global perceptual field. These maps are then fed into a convolution module with a shared 1×1 convolution kernel to reduce their dimensionality to the original C/r. The resulting batch-normalised feature map F_1 is then fed into a nonlinear activation function to obtain a feature map f with a shape of $1 \times (W + H) \times C/r$, as shown in Equation (5).

$$f = \delta\left(F_1\left(\left[z^h, z^w\right]\right)\right) \tag{5}$$

where δ is the nonlinear activation function and $[z^h, z^w]$ denotes the cascade operation along the spatial dimension.

Next, the feature map f is convolved with a 1×1 convolution kernel based on the original height and width to obtain two feature maps, F_h and F_w, with the same number of channels as the original. The feature maps are then passed through a sigmoid activation

function to obtain attention weights g^h for the height direction and g^w for the width direction, as shown in Equation (6).

$$\begin{cases} g^h = \delta\left(F_h\left(f^h\right)\right) \\ g^w = \delta(F_w(f^w)) \end{cases} \tag{6}$$

where σ represents the sigmoid activation function.

The original feature map is multiplied and weighted by the attention weights in the width and height directions to obtain the final feature maps with attention weights, as shown in Equation (7).

$$y_c(i,j) = x_c(i,j) \times g_c^h(i) \times g_c^w(j) \tag{7}$$

In response to the need to detect multiple dense targets in papaya orchards, this study proposes the introduction of a coordinate attention mechanism in YOLOv5s to improve the localisation accuracy of detecting dense papayas. This mechanism aims to solve the problem of dense target location information loss caused by 2D global pooling in existing attention mechanisms, such as SENet and CBAM.

2.2.4. Constructing a Two-Way Weighted Feature Pyramid

The neck component of YOLOv5s performs multiscale feature fusion using PANet, which currently sums all features during the fusion process. The bidirectional feature pyramid network (BiFPN) [33] proposed by the Google Brain team introduces learnable weights to determine the importance of different input features. It iteratively applies top-down and bottom-up multiscale feature fusion to enhance the transfer of feature information between different network layers. In this study, the PANet of YOLOv5s is improved by incorporating a bidirectional weighted feature pyramid structure as the feature fusion layer. The PANet and BiFPN structures are shown in Figure 6.

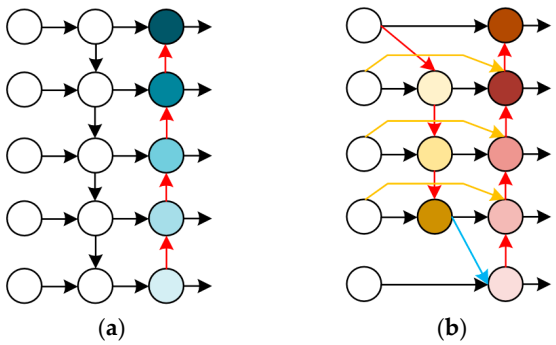


Figure 6. PANet and BiFPN structures. (**a**) PANet. (**b**) BiFPN. Notes: "○" denotes the network layer and "→" denotes the connection relationship between the network layers.

2.2.5. Loss Function

The YOLOv5s network model is trained using the scaled intersection over union (SIoU) bounding box loss function rather than the CIoU bounding box loss function. The CIoU does not consider the directional relationship between the ground truth box and the predicted box, leading to slow and inefficient convergence during training. The SIoU loss function introduces the vector angle between the ground truth box and the predicted box, consisting of four components: angle cost, dstance cost, shape cost, and IoU cost.

The angle loss (angle cost) is defined by Equation (8). As shown in Figure 7, when α is $\pi/2$ or 0, the angle loss is 0. The network model is trained to minimise α if α is less than $\pi/4$ and minimise β otherwise.

$$\Lambda = \cos\left[2 \times sin^2\left(arcsin\frac{c_h}{\sigma} - \frac{\pi}{4}\right)\right] \tag{8}$$

$$\sigma = \sqrt{\left(b_{c_x}^{gt} - b_{c_x}\right)^2 + \left(b_{c_y}^{gt} - b_{c_y}\right)^2} \tag{9}$$

$$c_h = max\left(b_{c_y}^{gt}, b_{c_y}\right) - min\left(b_{c_y}^{gt}, b_{c_y}\right) \tag{10}$$

where c_h is the height difference between the centres of the ground truth box and the predicted box, σ is the distance between the centres of the ground truth box and the predicted box, $\left(b_{c_x}^{gt}, b_{c_y}^{gt}\right)$ is the centre of the ground truth box, and B$\left(b_{c_x}, b_{c_y}\right)$ is the centre of the predicted box.

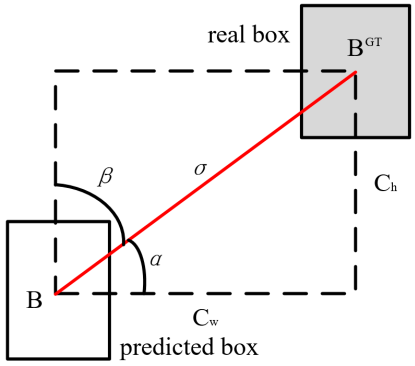


Figure 7. SIoU bounding box losses.

The distance cost Δ is given by Equation (11):

$$\Delta = 2 - e^{-(2-\Lambda)\left(\frac{b_{c_x}^{gt} - b_{c_x}}{c_{w1}}\right)^2} - e^{-(2-\Lambda)\left(\frac{b_{c_x}^{gt} - b_{c_x}}{c_{h1}}\right)^2} \tag{11}$$

where (c_{w1}, c_{h1}) is the width and height of the smallest enclosing rectangle of the ground truth and predicted boxes.

The shape cost Ω is defined by Equation (12):

$$\Omega = \left(1 - e^{-\frac{|w - w^{gt}|}{max(w, w^{gt})}}\right)^{\theta} + \left(1 - e^{-\frac{|h - h^{gt}|}{max(h, h^{gt})}}\right)^{\theta} \tag{12}$$

where (w, h) is the width and height of the predicted box, (w^{gt}, h^{gt}) is the width and height of the ground truth box, and θ is the parameter controlling the weight of the shape loss.

In summary, the SIoU loss function is defined as Equation (13):

$$\text{Loss}_{SIoU} = 1 - \text{IoU} + \frac{\Delta + \Omega}{2} \tag{13}$$

2.2.6. Improved YOLOv5s Network Model

The improved YOLOv5s network structure in this study is referred to asYOLOv5s-Papaya, as shown in Figure 8. Firstly, the Ghost module is introduced in the backbone to achieve lightweight improvement of the model. Secondly, the coordinate attention module (CA) is added between the backbone and neck and between the neck and head to enhance the model's performance in detecting dense small targets. Finally, the PANet structure is replaced by BiFPN in the neck section to enhance the transfer of feature information between different network layers and further improve the multiscale feature fusion capability.

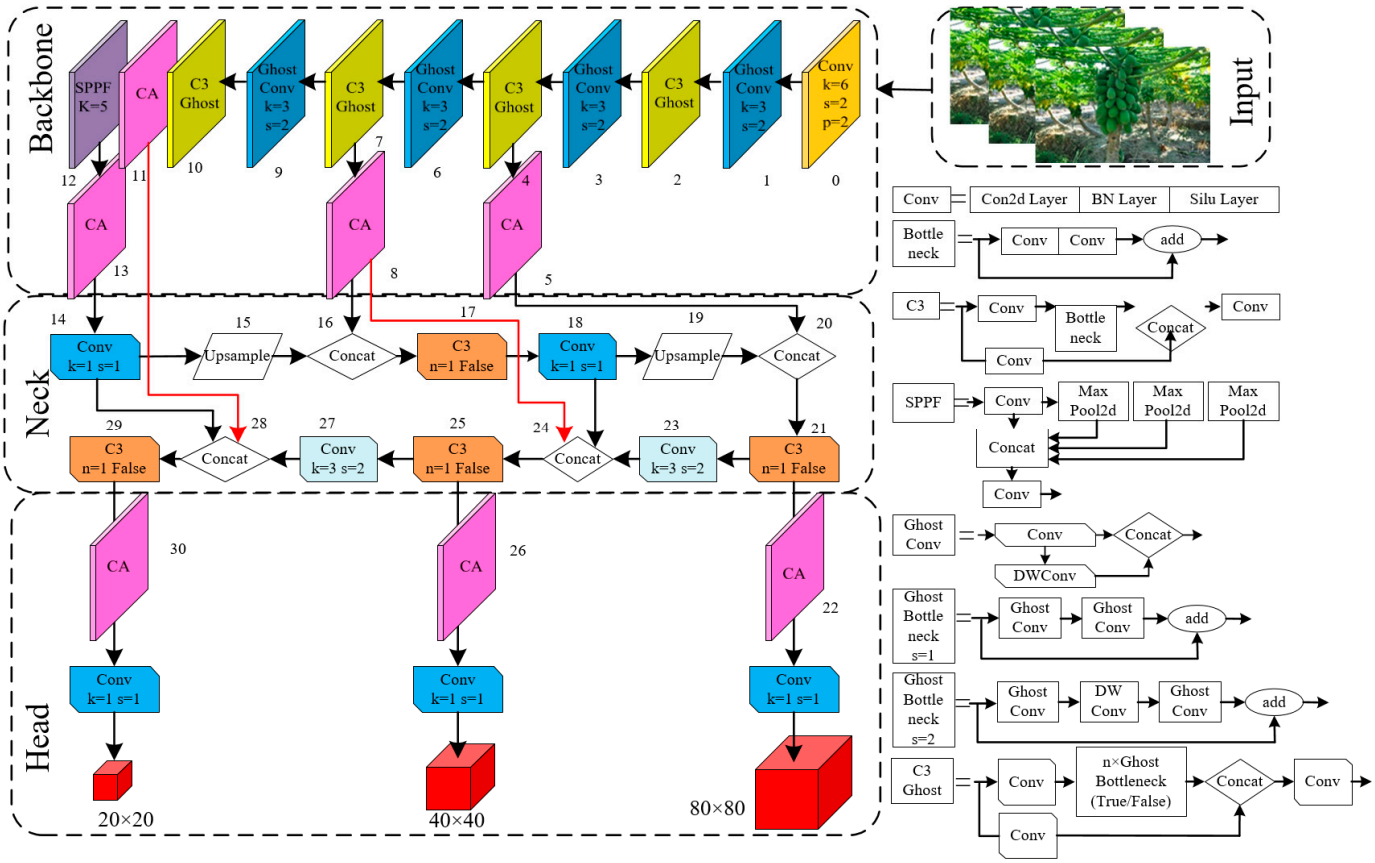


Figure 8. YOLOV5s-Papaya network structure.

3. Model Training and Testing

3.1. Training Processing Platform

In this study, a deep learning framework based on PyTorch 1.10.1 is used to train and test a dense papaya target detection model in a natural environment, with the following configurations: Intel(R) Xeon(R) E5-2683v3 CPU and NVIDIA GeForce RTX 3060 GPU; Windows 10 operating system; and an input image size of 640 × 640 pixels for the model. The training parameters are set as follows: batch size of 8300 training iterations, momentum of 0.937, initial learning rate of 0.001, and decay coefficient of 0.9.

3.2. Evaluation Indicators

The performance of dense papaya detection in a natural environment is evaluated using the following metrics: precision (P, %), recall (R, %), average precision (AP, %), number of parameters (Params, M), floating-point operations (FLOPs, G), and model size (MB). The metrics are calculated, as shown in Equations (14)–(16).

$$P = \frac{TP}{TP + FP} \times 100\% \tag{14}$$

$$R = \frac{TP}{TP + FN} \times 100\% \tag{15}$$

$$AP = \int_0^1 P(R)dR \times 100\% \tag{16}$$

where P denotes the proportion of all prediction frames detected correctly; R represents the proportion of correctly detected label frames amongst all label frames; TP is the number of correctly matched prediction frames; FP is the number of incorrectly predicted prediction

frames; FN is the number of missed label frames; AP denotes the average precision value of papaya; F_1 is the summed mean value of P and R.

4. Results and Discussion

4.1. Analysis of Model Training Results before and after Improvement

The same dataset and the same parameter settings were used during the training of both the improved YOLOv5s and YOLOv5s-Papaya models. The comparison curves of the bounding box loss functions for the two models were plotted by analysing the log files saved during the training process, as shown in Figure 9.

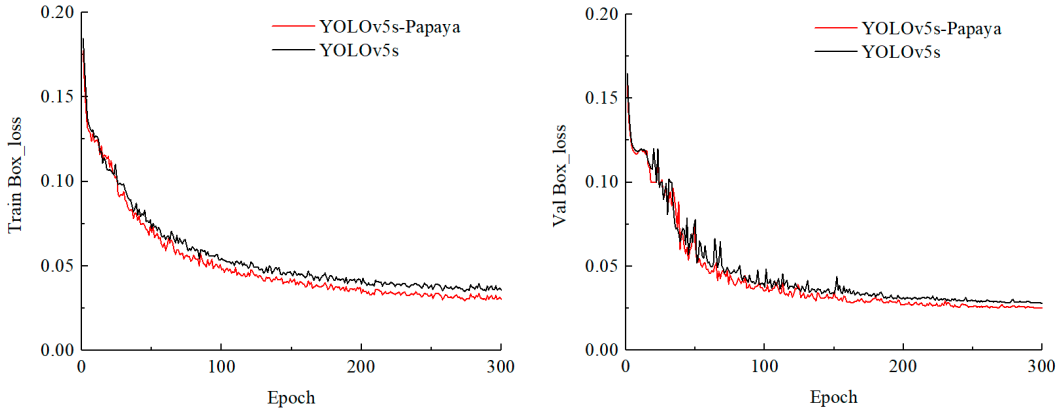


Figure 9. Comparison curve of the loss function of the bounding box of the model before and after the improvement. (**a**) Training set. (**b**) Validation set.

Figure 9a shows the bounding box loss curves of the model before and after the improvement on the training set, and Figure 9b shows the curves on the validation set. The trend of the bounding box loss curves before and after the improvement is similar. The decrease in the bounding box loss value becomes slower after 100 iterations, and it gradually stabilises after 250 iterations. The improved YOLOv5s-Papaya network, with the SIoU loss function, achieved a reduction of 0.005 in the bounding box loss value on the training set and of 0.002 on the validation set compared with the YOLOv5s network before the improvement. This indicates the improved performance of the model after incorporating the SIoU loss function. As shown in Figure 9, the improved YOLOv5s-Papaya network exhibits better performance than the YOLOv5s network before the improvement. The improved YOLOv5s-Papaya network demonstrates faster convergence and lower loss values, indicating that the improvement of the original loss function has improved the convergence ability of the network.

4.2. Results of Ablation Experiments

The purpose of the ablation experiments is to verify the effectiveness and feasibility of the improved modules. The results of the ablation experiments are shown in Table 1. Each experiment is identified by an ID, and ID1 represents the YOLOv5s network model without any improvements, ID2 introduces the Ghost module for model lightweight improvement, and ID3 replaces the PANet structure with BiFPN for feature fusion. ID4 incorporates a coordinate attention mechanism, ID5 improves the loss function during training, and ID6 combines all the aforementioned improvements in the YOLOv5s network model.

Table 1. Results of ablation experiments.

ID	Ghost Module	BiFPN	CA	SIoU	AP	Params/M	FLOPs/G
1	×	×	×	×	88.7	7.0	15.8
2	√	×	×	×	89.2	3.6	8.0
3	×	√	×	×	90.2	7.1	16.2
4	×	×	√	×	90.4	9.6	20.9
5	×	×	×	√	89.1	7.0	15.8
6	√	√	√	√	92.3	6.2	12.8

Notes: "√" Indicates that the current improvement method is used in the model, while "×" indicates that the current improvement method is not used in the model.

From the ablation experimental results in Table 1, the following observations can be made: comparing ID1 and ID2, the use of the Ghost Module to lighten the YOLOv5s network structure did not decrease the detection AP of the model but rather improved by 0.5% compared with the original network. Additionally, params decreased by 48.5%, and the computational effort was reduced by 49.3%. The introduction of the Ghost module replaces ordinary convolutions in the original network with linear operation operations, generating more feature mappings and ensuring a comprehensive understanding of dense multitarget papaya features. Therefore, the YOLOv5s model can be lightened whilst maintaining detection performance by incorporating the Ghost module. Comparing ID1 and ID3, after improving the PANet structure in the YOLOv5s network with BiFPN and introducing learnable weights to learn feature importance at different scales, the improved model achieved a 1.5 percentage point improvement in detection AP. The FLOPs and params increased by 0.1 M and 0.4 G, respectively. This indicates that the adoption of the BiFPN structure in the YOLOv5s model enhances the transfer of feature information between different network layers, leading to improved detection performance. Comparing ID1 and ID4, the model's detection AP increased by 1.7 percentage points after fusing the coordinate attention mechanism into the YOLOv5s network structure. This indicates that the fusion of the coordinate attention mechanism in the YOLOv5s model can improve the model's detection performance. However, the FLOPs and params of the model increased by 2.6 M and 4.8 G, respectively. Comparing ID1 and ID5, the AP of the model improved by 0.4% after improving the bounding box loss function during the training of the YOLOv5s model. By adding all the improved strategies from ID2 to 5 to the YOLOv5s model, the model's detection AP improved by 3.6 percentage points, params decreased by 11.4% and the computational effort decreased by 18.9% compared with the YOLOv5s network model before the improvement. The ablation experimental results show that the improved YOLOv5s-Papaya has better detection performance for dense multitarget papaya, and the use of the Ghost module reduces the complexity of the model.

4.3. Comparison of Detection Results of Different Algorithms

In order to verify the performance of various types of target detection models for dense multitarget papaya detection, nine highly representative network models—YOLOv3-Tiny, YOLOv4-Tiny, YOLOv5n, YOLOv5s, YOLOv7-Tiny, YOLOv8n, YOLOv8s, YOLOv8m, and YOLOv8l—are selected for the comparison test, with this paper's improved YOLOv5s-Papaya used for comparison tests. All models were trained and tested using the same papaya dataset, and AP, P, R, and model size, params, and FLOPs were selected as model evaluation metrics, and Table 2 shows the recognition results of each model for dense multitarget papaya.

Table 2. Detection results of different models.

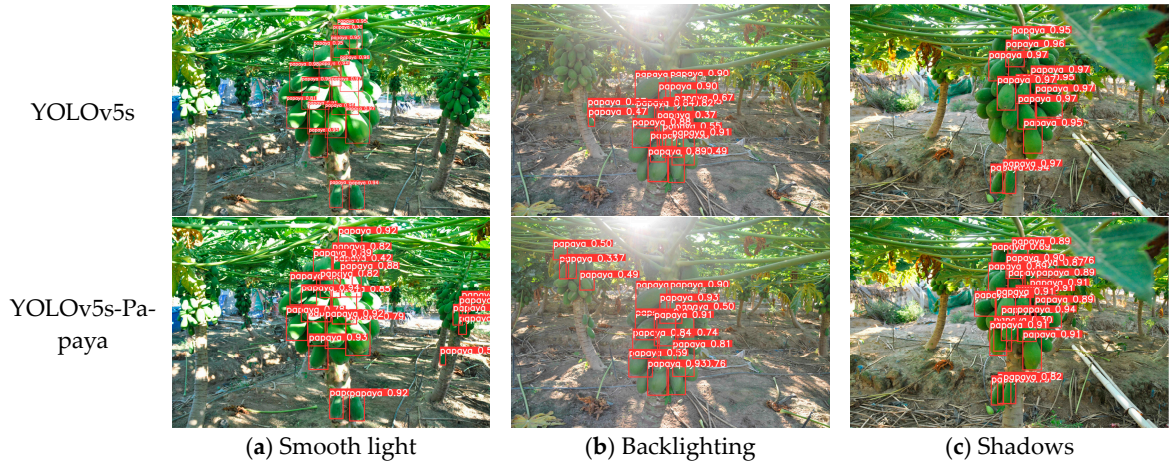
Models	AP (%)	P (%)	R (%)	Model Size (MB)	Params/M	FLOPs/G
YOLOv3-Tiny	89.9	91.2	83.3	16.6	8.6	12.9
YOLOv4-Tiny	85.5	87.1	77.8	6.3	3.0	6.4
YOLOv5n	90.5	79.5	90.7	3.67	1.8	4.1
YOLOv5s	88.7	86.1	85.3	14.5	7.0	15.8
YOLOv7-Tiny	91.3	84.7	87.0	12.3	6.0	13.2
YOLOv8n	84.6	78.6	76.9	5.94	3.0	8.1
YOLOv8s	87.3	77.9	81.3	21.4	11.1	28.4
YOLOv8m	88.9	81.1	82.7	49.6	25.8	78.7
YOLOv8l	90.6	83.7	84.5	83.5	43.6	164.8
YOLOv5s-Papaya	92.3	90.4	83.4	11.5	6.2	12.8

Table 2 shows that the AP value of the YOLOv5s-Papaya model is 92.3%. Compared to YOLOv3-Tiny, YOLOv4-Tiny, YOLOv5n, YOLOv5s, YOLOv7-Tiny, YOLOv8n, YOLOv8s, YOLOv8m, and YOLOv8l, AP increased by 2.4%, 5.1%, 1.8%, 3.6%, 1.0%, 7.7%, 5.0%, 3.4%, and 1.7%, respectively, and the detection AP of the YOLOv5s-Papaya model was the highest. The model size, params, and FLOPs of YOLOv5s-Papaya are 11.5 MB, 6.2 M, and 12.8 G, respectively. The model size, params, and FLOPs of YOLOv5s-Papaya are larger than those of the three lightweight network models, YOLOv4-Tiny, YOLOv5n, and YOLOv8n, but all are smaller than those of other network models. The detection accuracy of the YOLOv5s-Papaya network model is 90.4%. The P of the YOLOv5s-Papaya network model is 90.4%, which is 3.3, 10.9, 5.7, and 11.8 percentage points higher than that of the four lightweight networks, YOLOv4-Tiny, YOLOv5n, YOLOv7-Tiny, and YOLOv8n, respectively. Although YOLOv5s-Papaya reduces the P by 0.8 percentage points over YOLOv3-Tiny Lightweight, the amount of model parameters is reduced by 28.5%, the model size is reduced by 5.1 MB, and the detection accuracy is improved by 2.6%. Therefore, a comprehensive comparison shows that the YOLOv5s-Papaya model has better detection performance whilst achieving a lightweight improvement in the network model. This model can be embedded into a papaya-picking robot vision system to achieve automatic papaya-picking operations.

4.4. Analysis of Detection Results for Different Scenarios

4.4.1. Papaya Detection Results for Different Light Scenes

To investigate the detection performance of the improved YOLOv5s model for dense multitarget papaya under different lighting scenes, we selected papaya images taken in down-light, back-light, and shadow scenes for the experiments. The detection results are shown in Figure 10.



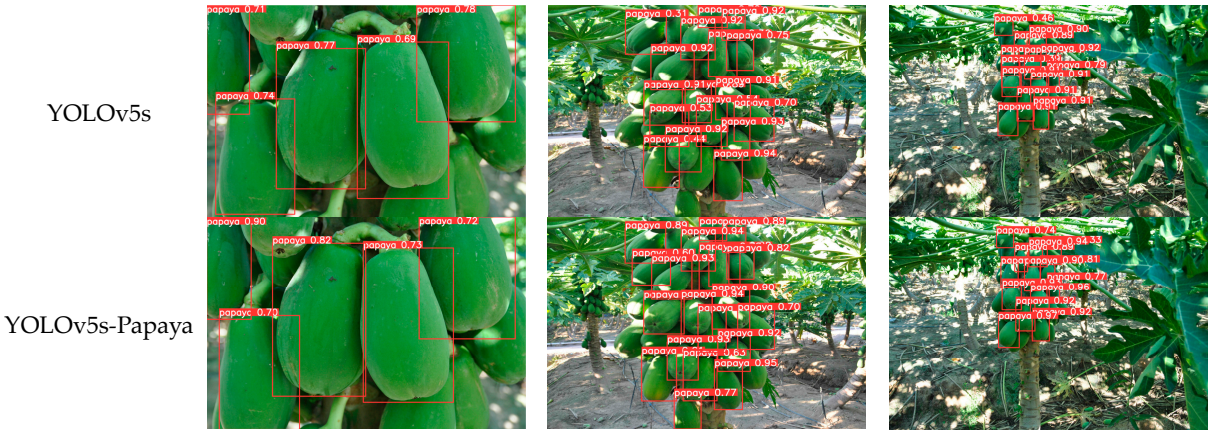
(**a**) Smooth light (**b**) Backlighting (**c**) Shadows

Figure 10. Dense multitarget papaya detection results for different lighting scenes.

As shown in Figure 10a, the YOLOv5s-Papaya model detects 21 dense multitarget papayas in the smooth light condition, while the original YOLOv5s model detects 16; as shown in Figure 10b, the YOLOv5s-Papaya model detects 17 dense multitarget papayas in the backlight condition, while the original YOLOv5s model detects 15; as shown in Figure 1c, the YOLOv5s-Papaya model detects 18 dense multitarget papayas in the shaded condition, while the original YOLOv5s model detects 11. The YOLOv5s-Papaya model identified significantly more overlapping shaded targets and small target papayas than the network before improvement under different lighting conditions. This indicates that the YOLOv5s-Papaya model outperformed the original YOLOv5s model in dense multitarget papaya detection. For severely overlapping obscured papayas, both the models before and after the improvement had partial missed detections. This is because the severely overlapping obscured papayas have the same colour as the leaves due to the green features of the epidermis. Additionally, they are affected by low-light conditions caused by the occlusion, leading to missed detections by the model. Moreover, heavily overlapped and obscured papayas lack texture feature information, which do not provide sufficient information to the model for making accurate judgements, resulting in missed detections.

4.4.2. Papaya Test Results for Different Fields of View

To investigate the detection performance of the improved YOLOv5s model for dense multitarget papaya under different fields of view, we conducted experiments using papaya images taken in three field-of-view scenarios: close range (0.2 m to 0.5 m), medium range (0.5 m to 1.0 m), and long range (1.0 m to 2.0 m). The detection results are shown in Figure 11.



(**a**) Proximity (**b**) Medium distance (**c**) Long distance

Figure 11. Dense multitarget papaya detection results for different field-of-view scenes.

As observed in Figure 10, the improved YOLOv5s-Papaya model recognises the same number of papayas as the original YOLOv5s model at close field-of-view angles. However, the YOLOv5s-Papaya model identifies significantly more papayas, especially overlapping and occluded targets and small targets, compared to the improved network at medium and long field-of-view angles. This indicates that the improved model has better detection performance under changing field-of-view angles.

In summary, the YOLOv5s-Papaya model outperforms the original YOLOv5s model in handling missed detections in different scenes. The model achieves an average detection time of 25 ms for images with a resolution of 640 × 640 in various scenes, with a detection accuracy of 92.3%. This demonstrates that the model exhibits stronger robustness and real-time performance, making it suitable for detecting dense multitarget papayas in complex orchard environments.

5. Conclusions

In this study, a YOLOv5s-Papaya deep CNN was proposed for the detection of dense multitarget papaya in the natural environment of an orchard. The experimental results showed that the YOLOv5s-Papaya achieved detection AP, P, and R values of 92.3%, 90.4% and 83.4%, respectively. The model size, params, and FLOPs were 11.5 MB, 6.2 M, and 12.8 G, respectively. Compared to the YOLOv5s network model, the YOLOv5s network model achieved a 3.6 percentage point improvement in detection accuracy, a 11.4% reduction in model parameters, and an 18.9% reduction in computational effort. Compared to YOLOv3-Tiny, YOLOv4-Tiny, YOLOv5n, YOLOv5s, YOLOv7-Tiny, YOLOv8n, YOLOv8s, YOLOv8m and YOLOv8l, the YOLOv5s-Papaya model exhibited the highest detection AP, with sequential increases in AP of 2.4%, 5.1%, 1.8%, 3.6%, 1.0%, 7.7%, 5.0%, 3.4%, and 1.7%. The model successfully identified a greater number of dense papayas compared to the original network before improvement in different lighting scenes and field-of-view angles. Additionally, the average time taken to detect images with a resolution of 640 × 640 was 25 ms. The YOLOv5s-Papaya model boasts a lighter structure and superior detection performance. This study provides a theoretical basis and technical support for intelligent recognition and picking of dense multitarget papayas in natural orchard environments.

Author Contributions: Conceptualization, L.W. and H.Z.; Data curation, L.W. and H.Z.; Formal analysis, H.Z. and Z.B.; Funding acquisition, W.F. and L.W.; Investigation, L.W., H.Z. and C.Y.; Methodology, C.Y. and H.Z.; Project administration, L.W. and W.F.; Resources, C.Y. and Y.W.; Software, L.W. and H.Z.; Supervision, W.F.; Validation, L.W., W.F., H.Z., C.Y. and Y.W.; Visualization, L.W. and H.Z.; Writing—original draft, L.W. and Z.B.; Writing—review and editing, L.W. and Z.B. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Key R&D projects in Hainan Province (Grant No. ZDYF2022XDNY231), the National Natural Science Foundation of China (Grant No. 32160424).

Data Availability Statement: The data presented in this study are available on demand from the first author at (wl_mac@shzu.edu.cn).

Acknowledgments: The authors would like to thank their schools and colleges, as well as the funding of the project. All support and assistance is sincerely appreciated. Additionally, we sincerely appreciate the work of the editor and the reviewers of the present paper.

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

References

1. Daagema, A.; Orafa, P.; Igbua, F. Nutritional Potentials and Uses of Pawpaw (*Carica papaya*): A Review. *Eur. J. Nutr. Food Saf.* **2020**, *12*, 52–66. [CrossRef]
2. Hua, X.; Li, H.; Zeng, J.; Han, C.; Chen, T.; Tang, L.; Luo, Y. A Review of Target Recognition Technology for Fruit Picking Robots: From Digital Image Processing to Deep Learning. *Appl. Sci.* **2023**, *13*, 4160. [CrossRef]
3. Lan, Y.; Yan, Y.; Wang, B.; Song, C.; Wang, G. Current status and future development of the key technologies for intelligent pesticide spraying robot. *Trans. Chin. Soc. Agric. Eng.* **2022**, *38*, 30–40.
4. Li, B.; Wang, M.; Wang, N. Development of a Real-Time Fruit Recognition System for Pineapple Harvesting Robots. In Proceedings of the 2010, Pittsburgh, Pennsylvania, 20–23 June 2010.
5. Wei, X.; Jia, K.; Lan, J.; Li, Y.; Zeng, Y.; Wang, C. Automatic Method of Fruit Object Extraction under Complex Agricultural Background for Vision System of Fruit Picking Robot. *Optik* **2014**, *125*, 5684–5689. [CrossRef]
6. Lv, J.; Xu, L. Method to Acquire Regions of Fruit, Branch and Leaf from Image of Red Apple in Orchard. *Mod. Phys. Lett. B* **2017**, *31*, 1740039. [CrossRef]
7. Lin, G.; Zou, X. Citrus Segmentation for Automatic Harvester Combined with AdaBoost Classifier and Leung-Malik Filter Bank. *IFAC-PapersOnLine* **2018**, *51*, 379–383. [CrossRef]
8. Wu, G.; Li, B.; Zhu, Q.; Huang, M.; Guo, Y. Using Color and 3D Geometry Features to Segment Fruit Point Cloud and Improve Fruit Recognition Accuracy. *Comput. Electron. Agric.* **2020**, *174*, 105475. [CrossRef]
9. Janiesch, C.; Zschech, P.; Heinrich, K. Machine learning and deep learning. *Electron. Mark.* **2021**, *31*, 685–695. [CrossRef]
10. Gill, H.S.; Murugesan, G.; Khehra, B.S.; Sajja, G.S.; Gupta, G.; Bhatt, A. Fruit Recognition from Images Using Deep Learning Applications. *Multimed. Tools Appl.* **2022**, *81*, 33269–33290. [CrossRef]

11. Li, D.; Sun, X.; Lv, S.; Elkhouchlaa, H.; Jia, Y.; Yao, Z.; Lin, P.; Zhou, H.; Zhou, Z.; Shen, J.; et al. A Novel Approach for the 3D Localization of Branch Picking Points Based on Deep Learning Applied to Longan Harvesting UAVs. *Comput. Electron. Agric.* **2022**, *199*, 107191. [CrossRef]
12. Zhang, S.; Xie, M. Real-Time Recognition and Localization Based on Improved YOLOv5s for Robot's Picking Clustered Fruits of Chilies. *Sensors* **2023**, *23*, 3408. [CrossRef]
13. Min, W.; Wang, Z.; Yang, J.; Liu, C.; Jiang, S. Vision-Based Fruit Recognition via Multi-Scale Attention CNN. *Comput. Electron. Agric.* **2023**, *210*, 107911. [CrossRef]
14. Pan, S.; Ahamed, T. Pear Recognition in an Orchard from 3D Stereo Camera Datasets to Develop a Fruit Picking Mechanism Using Mask R-CNN. *Sensors* **2022**, *22*, 4187. [CrossRef]
15. Gai, R.; Chen, N.; Yuan, H. A Detection Algorithm for Cherry Fruits Based on the Improved YOLO-v4 Model. *Neural Comput. Appl.* **2023**, *35*, 13895–13906. [CrossRef]
16. Wu, D.; Lv, S.; Jiang, M.; Song, H. Using Channel Pruning-Based YOLO v4 Deep Learning Algorithm for the Real-Time and Accurate Detection of Apple Flowers in Natural Environments. *Comput. Electron. Agric.* **2020**, *178*, 105742. [CrossRef]
17. Zhong-hua, Z.; Wei-kuan, J.; Wen-jing, S.; Su-juan, H.; Ze, J.; Yuan-jie, Z. Green Apple Detection Based on Optimized FCOS in Orchards. *Spectrosc. Spectr. Anal.* **2022**, *42*, 647–653.
18. Sun, J.; Qian, L.; Zhu, W.; Zhou, X.; Dai, C.; Wu, X. Apple detection in complex orchard environment based on improved RetinaNet. *Trans. Chin. Soc. Agric. Eng.* **2022**, *38*, 314–322.
19. Song, H.; Wang, Y.; Lü, S.; Jiang, M. Camellia Oleifera Fruit Detection in Natural Scene Based on YOLO V5s. *Trans. Chin. Soc. Agric. Mach.* **2022**, *53*, 234–242.
20. Wang, Y.; Xue, J. Lightweight object detection method for Lingwu long jujube images based on improved SSD. *Trans. Chin. Soc. Agric. Eng.* **2021**, *37*, 173–182.
21. Zhang, T.; Wu, F.; Wang, M.; Chen, Z.; Li, L.; Zou, X. Grape-Bunch Identification and Location of Picking Points on Occluded Fruit Axis Based on YOLOv5-GAP. *Horticulturae* **2023**, *9*, 498. [CrossRef]
22. Abeyrathna, R.M.R.D.; Nakaguchi, V.M.; Minn, A.; Ahamed, T. Recognition and Counting of Apples in a Dynamic State Using a 3D Camera and Deep Learning Algorithms for Robotic Harvesting Systems. *Sensors* **2023**, *23*, 3810. [CrossRef]
23. Zhou, J.; Zhang, Y.; Wang, J. A Dragon Fruit Picking Detection Method Based on YOLOv7 and PSP-Ellipse. *Sensors* **2023**, *23*, 3803. [CrossRef]
24. Guo, J.; Xiao, X.; Miao, J.; Tian, B.; Zhao, J.; Lan, Y. Design and Experiment of a Visual Detection System for Zanthoxylum-Harvesting Robot Based on Improved YOLOv5 Model. *Agriculture* **2023**, *13*, 821. [CrossRef]
25. Yan, B.; Fan, P.; Lei, X.; Liu, Z.; Yang, F. A Real-Time Apple Targets Detection Method for Picking Robot Based on Improved YOLOv5. *Remote Sens.* **2021**, *13*, 1619. [CrossRef]
26. Qi, X.; Dong, J.; Lan, Y.; Zhu, H. Method for Identifying Litchi Picking Position Based on YOLOv5 and PSPNet. *Remote Sens.* **2022**, *14*, 2004. [CrossRef]
27. Zhang, B.; Wang, R.; Zhang, H.; Yin, C.; Xia, Y.; Fu, M.; Fu, W. Dragon Fruit Detection in Natural Orchard Environment by Integrating Lightweight Network and Attention Mechanism. *Front. Plant Sci.* **2022**, *13*, 1040923. [CrossRef]
28. 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 (cvpr), Salt Lake City, UT, USA, 18–23 June 2018; IEEE: New York, NY, USA, 2018; pp. 8759–8768.
29. Zheng, Z.; Wang, P.; Ren, D.; Liu, W.; Ye, R.; Hu, Q.; Zuo, W. Enhancing Geometric Factors in Model Learning and Inference for Object Detection and Instance Segmentation. *IEEE Trans. Cybern.* **2022**, *52*, 8574–8586. [CrossRef]
30. Han, K.; Wang, Y.; Tian, Q.; Guo, J.; Xu, C.; 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, 13–19 June 2020; IEEE: New York, NY, USA, 2020; pp. 1577–1586.
31. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the 2018 IEEE/Cvf Conference on Computer Vision and Pattern Recognition (cvpr), Salt Lake City, UT, USA, 18–23 June 2018; IEEE: New York, NY, USA, 2018; pp. 7132–7141.
32. Woo, S.; Park, J.; Lee, J.-Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the Computer Vision—Eccv 2018, Munich, Germany, 8–14 September 2018; Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., Eds.; Springer International Publishing Ag: Cham, Switzerland, 2018; Volume 11211 Pt Vii, pp. 3–19.
33. 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. 10778–10787.