



A detection algorithm based on improved YOLOv5 for coarse-fine variety fruits

Yanru Fan¹ · Yuliang Cai² · Huijun Yang^{1,3,4}

Received: 18 July 2023 / Accepted: 5 November 2023 / Published online: 7 December 2023
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2023

Abstract

Fruit detection and recognition is a key technology in precision agriculture such as automated picking, orchard yield measurement, and fruit sorting. Although current algorithms have good detection performance for single-class objects in living scenes, they have lower detection and recognition accuracy for different varieties of fruits with high similarity in complex environments and consume high computing resources and costs, which cannot be applied to edge devices for real-time detection and sorting of varieties fruits species. This paper proposed a lightweight model based on YOLOv5 for the detection and recognition of coarse-fine variety fruits in clean or complex scenes. First, the networks with different widths and depths of YOLOv5 were trained to find the best baseline detection net; second, GhostConv, a lightweight convolution layer, was introduced to reduce parameters and computations in the baseline network; finally, the input channels of the depth convolution in the backbone was expanded and the Wise-IoU bounding box loss function was introduced to improve the detection accuracy of the baseline network. The experimental results showed that, considering both detection performance and model complexity, YOLOv5s performs better as the baseline network. After optimization, the model parameters and weight volume were reduced by approximately 33%, the mean average precision at 0.5 IoU(mAP@.5) increased by 0.6%, and increased by 1.2% at IoU = 0.5:0.95(mAP@.5:.95). The improved model achieved the reasonable balance between detection accuracy and complexity. It is suitable for real-time detection, online grading, and rapid sorting of fruits in precision agriculture.

Keywords Deep learning · Fruit detection · Improved YOLOv5 · Coarse-fine variety fruits

Introduction

China is one of the largest fruit producing countries in the world. Following the grain and vegetable industries, the fruit industry has developed into the third largest agricultural

plantation industry [1]. However, compared with developed countries, most of the fruit picking and post-harvest processing task rely on high-cost and inefficient manual work, which restricts the automation of the fruit industry, and results in backward of post-production commercialization processing technology [2]. With the acceleration of population aging process and urbanization, labor shortage will also bring difficulties and challenges to the sustainable development of the fruit industry. Information technology-supported precision agriculture provide a better solution for these problems. Automated intelligent equipment and machinery can be used instead of manual labor to complete tasks such as automatic fruit picking, accurate yield measurement in orchards and fruit sorting, thus saving labor and material expenses. Fruit detection and recognition is a key technology to realize vision system in precision agriculture, which can provide category information and location information for intelligent agricultural equipment.

The traditional fruit detection algorithm is based on color, geometry, texture and other features to classify and detect

Yuliang Cai and Huijun Yang have contributed equally to this work.

✉ Huijun Yang
yhj740225@nwfufu.edu.cn

- ¹ College of Information Engineering, Northwest A & F University, Yangling 712100, Shaanxi, China
- ² College of Horticulture, Northwest A & F University, Yangling 712100, Shaanxi, China
- ³ Key Laboratory of Agricultural Internet of Things, Ministry of Agriculture and Rural Affairs, Yangling 712100, Shaanxi, China
- ⁴ Key Laboratory of Agricultural Information Perception and Intelligent Service, Ministry of Agriculture and Rural Affairs, Yangling 712100, Shaanxi, China

fruit targets, including three parts: region selection, feature extraction and classification, which is relatively mature but not robust to uneven illumination, fruit occlusion and similar color of fruit and background, and cannot meet the real-time requirements of application scenarios [3]. Since 2014, the advanced development in deep learning technology, particularly of convolutional neural networks (CNN), has significantly improved the state of the art in object detection [4]. Deep learning technology can self-learn and automatically extract feature information from images, with stability and efficiency in fruit detection and recognition [5, 6]. Deep learning-based fruit detection algorithms are divided into two types: region-based models and regression-based models. The region-based models consist of two stages: generating candidate regions and extracting features from each candidate box for bounding box and classification regression tasks, which leads to a slower recognition speed, such as Fast Region-based CNN (R-CNN) [7], Faster R-CNN [8], and Mask R-CNN [9]. The regression-based object detection method abandons the stage of extracting candidate regions, and directly obtains the category probability and position of the object, resulting in a simpler network structure, such as You Only Look Once (YOLO) series [10–12] and Single Shot MultiBox Detector (SSD) [13]. Although some accuracy is sacrificed, the detection and recognition speed of the algorithm has been improved.

Although deep learning-based object detection algorithms have superior detection performance, they require large numbers of parameters and high computational costs, making them difficult to deploy on edge devices with limited space and computing power. As a result, the application of algorithms to real-time detection and efficient sorting of fruits is challenging [14, 15]. Due to the lack of public detection datasets for coarse-fine variety fruits, the current model cannot detect both coarse and fine variety fruits with highly similar phenotypes.

To address the problems that existing fruit detection models are too complex and cannot accurately detect coarse-fine variety fruits, this paper proposes a detection and recognition model for multiple coarse-fine variety fruits based on improved YOLOv5, which has high robustness and accuracy, low complexity, and can be applied to real-time detection, online grading and fast sorting of coarse-fine variety fruits. The main contributions are as follows:

- (1) A fruit image dataset containing 20 different fruit varieties under clean and complex backgrounds was constructed to address the lack of public detection datasets for coarse-fine variety fruits, which can provide data support for the research, optimization, and application of detection models for coarse-fine variety fruits.
- (2) To address the difficulty of complex networks in deploying on edge devices with limited computing

power, a lightweight network based on GhostConv and C3Ghostv1 was proposed by introducing deep convolution on baseline detection network suitable coarse-fine variety fruits, which greatly reduced the number of parameters and computation of the model.

- (3) To address the poor accuracy of existing networks for fine variety fruit detection in complex scenes, a feature extraction network based on C3Ghostv2 was proposed to obtain rich overall features of fruits and the bounding box loss function was optimized to improve the localization of ordinary-quality anchor boxes in complex scenes.

Related works

Traditional fruit detection and recognition methods typically use manually designed methods to extract features. First, features such as size, shape, color, and texture of the fruit image are extracted, and then a classifier is built by fusing one or more of these features to achieve automatic fruit classification and recognition. Liu et al. [16] developed a machine vision algorithm based on elliptical boundary models to recognize immature and ripe grapefruit fruits on trees by converting images from RGB space to Y'CbCr space and then fitting an implicit second-order polynomial of elliptical boundary models in Cr-Cb color space using ordinary least squares (OLS). To achieve accurate detection of litchi fruits in natural environments, Yu et al. [17] used color and texture features to train a random forest binary classification model to identify litchi fruits and proposed a ripe litchi identification method based on multiscale detection and non-maximal suppression algorithm to further improve the fruit detection accuracy. Pothen and Nuske [18] proposed a high-precision key point detection algorithm using round fruits such as grapes and apples, which determines the marquee location of fruits by intensity variation and gradient direction on the fruit surface, and uses a random forest classifier to identify fruit species. Traditional fruit detection and recognition algorithms are relatively mature, but the detection accuracy is highly dependent on the extracted features and trained classifiers, resulting in low detection rate, slow detection speed, and poor applicability for targets in complex environments.

With the improvement of computer performance, many deep learning-based object detection algorithms have been applied to fruit detection and recognition tasks, resulting in significantly improved detection performance and speed. Prakash and Prakasam [19] proposed an intelligent fruit classification system based on convolutional neural network and bilinear pooling of heterogeneous streams. Gao et al. [20] proposed a multi-class apple detection method based on improved Faster R-CNN network for the problem

of reduced fruit picking efficiency due to several branches shading in fruit forests, with mean average precision (mAP) reaching 87.9% and average detection time of 0.241 s for a single image. The regression-based object detection algorithm has faster detection recognition speed and is suitable for practical application scenarios, and many improved algorithms based on it have been proposed. Mirhaji et al. [21] trained and tested different versions of the YOLO model using image datasets of orange trees under different lighting conditions, and adapted the models using a migration learning strategy, finally concluding that YOLO-V4 is the best model for orange detection. To address inaccurate cherry fruit detection due to leaf shading, Gai et al. [22] proposed an improved YOLOv4 deep learning algorithm, which increased the density between network layers to enhance feature extraction by combining DenseNet in the backbone of YOLOv4, and its average accuracy was improved by 0.15 compared to the original model. Wang et al. [23] proposed the DSE-YOLO algorithm to detect multi-stage strawberries by introducing DSE modules in the backbone to extract various details and semantic features in horizontal and vertical dimensions, which achieved mAP of 86.58% and F1 score of 81.59%. Yao et al. [24] developed a defect detection model based on YOLOv5 for kiwi flaw detection, which can detect defects accurately at a fast speed. In response to the lack of accuracy in detecting small tomato targets, Wang et al. [25] proposed an improved small mobile network YOLOv5 (SM-YOLOv5) detection algorithm based on YOLOv5 for target detection of tomato picking robots in plant factories. Ma et al. [26] proposed a lightweight fruit recognition network DGCC-Fruit based on YOLOv5 to detect fine-grained fruits in different environments.

Deep learning-based fruit detection algorithms can learn features automatically from training data and exhibit strong fruit recognition capabilities in complex environments. However, their parameters and computational complexity are too high to be deployed on edge devices for real-time fruit detection tasks. Additionally, current deep learning-based fruit detection algorithms primarily focus on coarse-variety fruit datasets in ideal environments, with weak

model generalization capabilities, which cannot accurately detect multi-objective and fine-variety fruits in complex environments.

Materials and methods

Dataset

Samples in the experiments

Considering the wide cultivation area, variety diversity, species and varieties classification of fruits in botany, five coarse variety of apples, cherries, watermelons, oranges and pomelos as well as their 20 fine variety fruits were selected as experimental samples in this study, the variety information is shown in Table 1 and Appendix. Specifically, seven varieties of cherry samples were procured in Yantai and Yangling, China for this paper, namely, black pearl, red light, huang mi, lapins, rainier, tieton, and pioneer. Samples of three varieties of apples, ruiyang, ruixue, and aifei, were obtained at the Baishui Apple Experiment Station of Northwest Agriculture and Forestry University, China. Watermelon varieties include futian watermelon, dafugui watermelon, xinfunong watermelon, and chengyu watermelon. Pomelo varieties include meizhou pomelo, liangping pomelo, and liangjiang pomelo. All samples were stored in a cooler at 0–5 °C and 85% relative humidity for up to 5 days to preserve freshness prior to the experiment.

Dataset creation

JPG images of 5 coarse variety fruits and their 20 fine variety fruits were collected using a motorized turntable and a normal camera in a simple background. The camera parameters are shown in Table 2. The acquisition process consisted of three steps:

Table 1 Variety of samples

Coarse-variety ^a	Fine-variety ^b			
Apple	Ruyang	Ruixue	Aifei	
Cherry	Black Pearl	Red Light	Huang Mi	Lapins
	Rainier	Tieton(dalian)	Tieton(yantai)	Pioneer
Watermelon	Futian	Dafugui	Xinfunong	Chengyu
Pomelo	Meizhou	Liangping	Liangjiang	
Orange	Gannan orange	Sugar orange		

^aSpecies in Botany

^bVariety in Botany

Table 2 Camera parameters

Parameters	Model or size
Branding	Redmi k60 rear camera
Sensor	Sony imx766 / 0.5 inch
Pixels	64 million
Aperture	f/1.79
Panoramic	120°
Storage methods	JPG

- (1) Fixed the camera position, placed the fruit samples in the center of the motorized turntable, and adjusted the initial pose of each sample to neck facing upward.
- (2) Controlled the motorized turntable to rotate 360° clockwise and took pictures of the samples every 30° to obtain pictures of the samples at multiple angles.
- (3) Adjusted the sample pose to neck facing forward, repeated the operation in (2), and a single sample could get 24 pictures in a single background.

To improve the generalization and applicability of the model, multi-angle, multi-variety, and different fruit densities images were captured in different complex scenes indoors and in the field. Finally, as shown in Fig. 1a, 13,198 fruit images with 3024*4032 pixels were get and stored in JPG format after compression.

To address insufficient data and unbalanced samples of fruit images, the fruit dataset was expanded using data enhancement methods such as flip, rotation, crop, and brightness transformation and a total of 23,198 images were obtained. The data-enhanced images are shown in Fig. 1b.

As shown in Fig. 1c, the images were manually labeled by LabelImg software. Firstly, the target fruit in the image was marked by the smallest outer rectangle and its variety was indicated, then a txt format annotation file was generated, which contained the variety of fruit, the coordinates x and y of the center point of the rectangle, width w and height h relative to the image. Finally, the dataset was divided into training set, validation set and test set in the ratio of 8:1:1, and the details of the dataset are shown in Table 3.

Algorithmic optimization

Baseline network selection

As the detection algorithm with superior detection speed and accuracy performance among the YOLO family of algorithms [27], YOLOv5 consists of four parts: input, backbone, neck, and head [28]. The training image is fed into the backbone after data enhancement, adaptive anchor frame computation and adaptive image scaling at the input. Backbone mainly consists of CBS, C3, and SPPF modules, which is used to

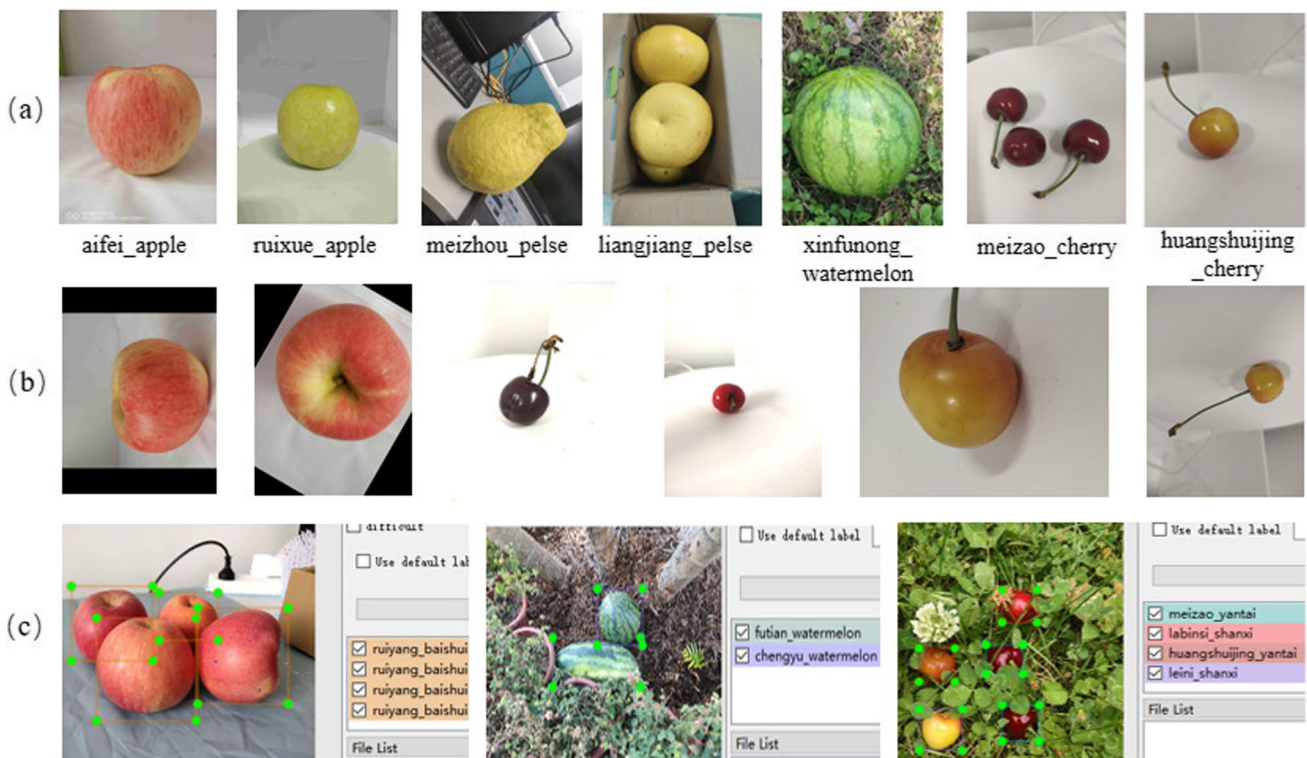


Fig. 1 Dataset creation. **a** The original captured images. **b** The data enhanced images. **c** The image annotation process

Table 3 Details of the self-made dataset

Fruit variety	Number of pictures		
	Training set	Test set	Validation set
Ruiyang_apple	609	75	75
Ruixue_apple	650	75	75
Aifei_apple	495	75	75
Black Pearl_cherry	600	75	75
Red Light_cherry	600	75	75
Huang Mi_cherry	600	75	75
Lapins_cherry	650	75	75
Rainier_cherry	660	75	75
Tieton_cherry(dalian)	660	75	75
Tieton_cherry(yantai)	660	75	75
Pioneer_cherry	660	75	75
Futian_watermelon	600	75	75
Dafugui_watermelon	600	75	75
Xinfunong_watermelon	600	75	75
Chengyu_watermelon	600	75	75
Meizhou_pomelo	600	75	75
Liangping_pomelo	600	75	75
Liangjiang_pomelo	600	75	75
Sugar orange	600	75	75
Gannan_orange	600	75	75
Mixed varieties	4198	500	500

extract feature maps at three scales. C3 module performs feature extraction on the feature maps, which reduces the repetition of the gradient information during the optimization process of convolutional neural network, thus reducing the amount of computation and guaranteeing the accuracy rate. Neck fuses the features of the backbone output. By using a structure that combines Feature Pyramid Networks (FPN) and Perceptual Adversarial Networks (PAN), it fully fuses the high-level semantic features and the low-level localization features. Head generates bounding boxes and predicts varieties using loss functions and Non-Maximum Suppression (NMS).

Depth_multiple is used as a scaling factor for the residual blocks in C3 and controls the depth of the network. Meanwhile, width_multiple is used as a scaling factor for the number of channels, controlling the width of the network. As shown in Table 4, YOLOv5 is divided into YOLOv5n, v5s, v5m, and v5l according to depth_multiple and width_multiple. It can be seen that the YOLOv5n network has the simplest structure, the smallest network depth, and the fewest parameters. The YOLOv5l network has the most complex structure, the deepest network, and the most parameters. While complex networks can achieve better detection accuracy, they require more parameters and computational costs, and take longer to train and detect. To detect coarse-fine variety fruits accurately and quickly, YOLOv5s was chosen as the baseline network, which can maintain a balance between detection performance and model complexity.

Table 4 YOLOv5 network with different scaling factor

Scaling factor	YOLOv5n	YOLOv5s	YOLOv5m	YOLOv5l
Depth_multiple	0.33	0.33	0.67	1.00
Width_multiple	0.25	0.50	0.75	1.00

Algorithm improvement

Deep learning-based object detection algorithms require high computational resources, which result in limit application to real-time detection and sorting of fruits in real operations due to hardware conditions. To facilitate the application of fruit detection networks to practical operations, this paper proposed a low-complexity and high-precision fruit detection network based on YOLOv5s. First, the lightweight module—C3Ghostv1 was constructed by introducing the lightweight convolution—GhostConv, and further the lightweight network structure based on GhostConv, C3Ghostv1 was proposed to reduce the model complexity; second, the C3Ghostv2 module was introduced into the backbone to enable the deep convolution to extract rich overall target features in higher dimensions and improve the performance for detecting phenotypically similar fruit varieties, which expands the input channels of residual structure in C3Ghostv1; Finally, Wise-Intersection over Union (IoU) loss function with dynamic non-monotonic focusing mechanism was introduced to improve the detection and generalization performance of the model for multi-target and fine-variety fruits in complex environments. The improved network structure is shown in Fig. 2.

(1) Network structure lightweighting

In the process of extracting fruit features from a neural network, many feature maps with high similarity are generated, which usually ensure a comprehensive understanding of the input data and have an important impact on the performance of the network [29]. Compared to depthwise convolution, standard convolution, which is utilized to generate numerous similar and redundant feature maps in YOLOv5, requires more parameters and computation, resulting in difficulty in deployment on edge devices. Therefore, a lightweight network based on GhostConv and C3Ghostv1 was proposed to generate fruit redundant feature maps, which reduced the model complexity by introducing depthwise convolution.

GhostConv Module As shown in Fig. 3, GhostConv first uses 1×1 standard convolution to generate part of the intrinsic feature maps, then uses 5×5 depthwise convolution to generate the “ghost” feature maps of the intrinsic feature maps, and finally superimposes the intrinsic feature maps and the “ghost” feature maps on the channels to obtain the output feature maps.

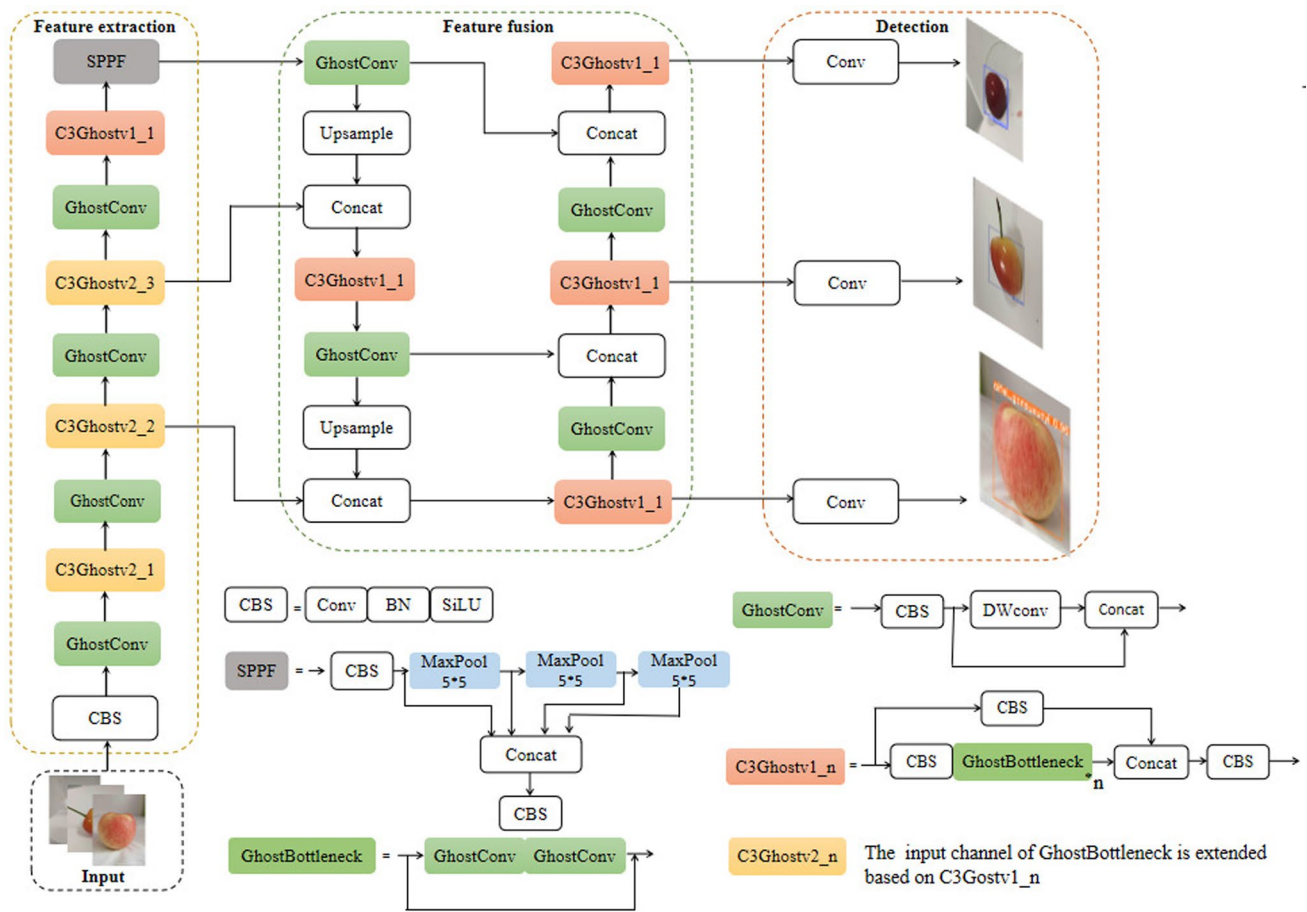


Fig. 2 Proposed network based on YOLOv5s

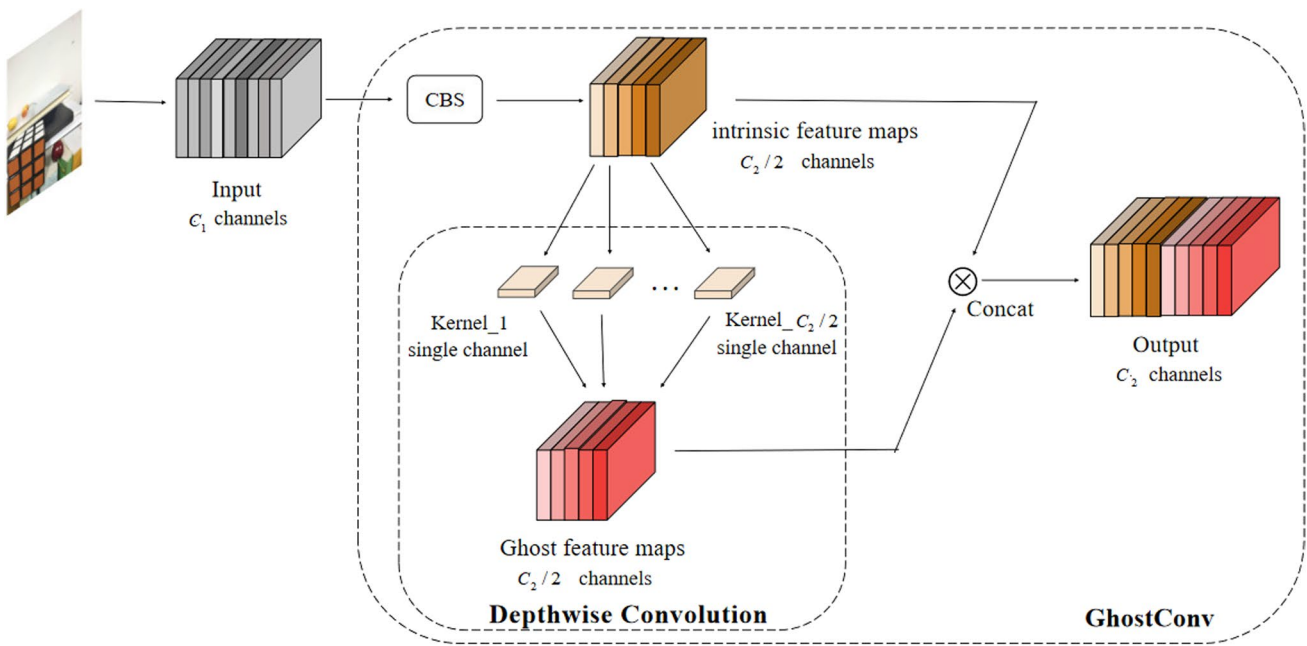


Fig. 3 Structure of GhostConv module

Assuming that the input feature map size is $h \cdot w \cdot c$, the convolution kernel size is $k \cdot k$, and the output is $h'' \cdot w'' \cdot c_2$, the Computational volume S_1 and params P_1 of the standard convolution are shown in Eqs. 1 and 2, respectively.

$$S_1 = c_2 \cdot h' \cdot w'' \cdot c_1 \cdot k \cdot k \tag{1}$$

$$P_1 = c_2 \cdot c_1 \cdot k \cdot k \tag{2}$$

In GhostConv, $1/2c_2$ intrinsic feature maps are obtained by standard convolution, and the same number of “ghost” feature maps are obtained by depthwise convolution. Assuming that the size of the convolution kernel for deep convolution is $d \cdot d$, the computational volume S_2 and parameters P_2 are shown in Eqs. 3 and 4, respectively.

$$S_2 = \frac{1}{2}c_2 \cdot h' \cdot w' \cdot (c_1 \cdot k \cdot k + d \cdot d) \tag{3}$$

$$P_2 = \frac{1}{2}c_2 \cdot (c_1 \cdot k \cdot k + d \cdot d) \tag{4}$$

If $k \approx d$, according to Eqs. 1–4, the theoretical acceleration ratio of standard convolution and GhostConv r_s is shown in Eq. 5, and the parameter compression ratio r_p is shown in Eq. 6.

$$r_s = \frac{S_1}{S_2} = \frac{c_1 \cdot k \cdot k}{\frac{1}{2} \cdot c_1 \cdot k \cdot k + \frac{1}{2}d \cdot d} \approx 2 \tag{5}$$

$$r_p = \frac{P_1}{P_2} \approx 2 \tag{6}$$

As shown in Eqs. 5 and 6, the introduction of GhostConv in the network can theoretically save 1/2 of training time and reduce 1/2 of parameters, which facilitates the deployment of the model to accomplish the task of real-time fruit detection and fast sorting in real operations.

C3Ghostv1 Module As shown in Fig. 4, C3Ghostv1 uses two branches to process the input fruit feature maps in

parallel. One branch uses standard convolution to halve the input feature channels and extract low-level fruit features. The other branch reduces the dimension of input feature channels with 1×1 standard convolution, and then extracts high-level abstract features of fruits through multiple GhostBottlenecks. As the basic residual unit, GhostBottleneck first processes the input fruit feature map with two stacked GhostConv modules, where the first GhostConv module does not use the SiLU activation function to avoid the loss of fruit information due to the nonlinear activation function [29]; then the result is associated with the input feature map through residual connections to obtain the output. By introducing GhostConv, GhostBottleneck can alleviate gradient vanishing caused by deepening the network at a lower cost, which is beneficial for the network to extract more complex fruit features to identify fruits with small phenotypic differences. Finally, the results of the two branches are stacked on the channel dimension.

C3Ghostv1 preserves the reuse of fruit features by adopting a hierarchical feature fusion strategy, while avoiding excessive repetitive gradient information by truncating the gradient flow, thus ensuring the model’s ability to learn different fruit features and reducing the network’s parameters and computation, accelerating the training and inference speed.

(2) Enhancement of network feature extraction capability

In depthwise convolution, the channel information of the input image is separated during the calculation process, making it impossible to obtain the associated information of different channels at the same spatial position and limiting the extracted features to the input feature dimensions [30]. Although the dimensionality reduction in C3Ghostv1 can alleviate the complexity of the network, a limited number of features extracted by the depthwise convolution at a lower dimension makes it difficult to extract rich overall fruits feature in backbone, which is not conducive to identifying fruits with high similarities.

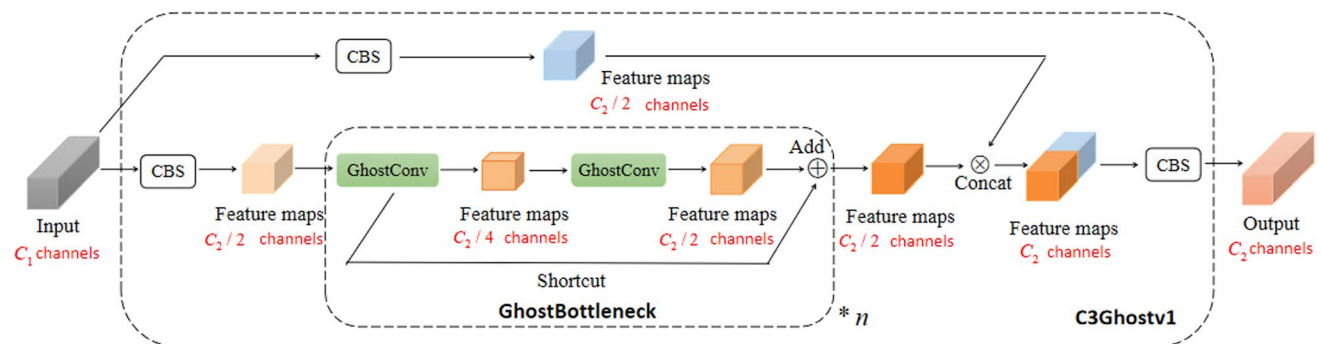


Fig. 4 Structure of C3Ghostv1 module

To improve the feature extraction capability of the network, C3Ghostv2 module was designed in the backbone inspired by the idea of MobileNetv2 [30], which enabled depthwise convolution to capture rich overall features of different fruits at higher dimensions. As shown in Fig. 5, this module uses two parallel branches to process the input features. One branch uses 1*1 convolution to increase the channel dimension before the GhostBottleneck, doubling the number of channels. The other branch uses standard convolution to extract shallow features. Finally, a 1*1 convolution is used to reduce the dimension of the combined feature map from both branches, ensuring that the input and output have the same number of channels. By expanding the input channels of GhostBottleneck, C3Ghostv2 enables depthwise convolution to extract more features and reduce feature loss, which facilitates the network to extract comprehensive information of different fruit varieties, thereby improving the detection performance of the model on different fruit variety and reducing false detection rates.

(3) Optimization of bounding box regression loss function

YOLOv5 constructs a loss function weighted by bounding box regression (BBR) loss, classification loss, and objectness loss, where the bounding box regression loss directly determines the localization performance of the model. YOLOv5 adopts the Complete-IoU [31] bounding box loss function, which includes two penalty terms added to IoU loss [32]: normalized distance and aspect ratio between anchor boxes and target boxes. However, its lack of a focusing mechanism for accurate localization of ordinary-quality anchor boxes results in high-quality anchor boxes, ordinary-quality anchor boxes, and low-quality anchor boxes contributing equally to the loss, which limits the improvement of detection performance for multi-object fruits in complex scenes. Therefore, Wise-IoU [33] bounding box loss function was introduced to

effectively improve the model’s detection performance and generalization ability for fruits in complex scenes, which utilized a dynamic non-monotonic focusing mechanism to reduce the competitiveness of high-quality anchor boxes and mitigate the harmful gradients generated by low-quality anchor boxes.

As shown in Fig. 6, for the anchor box $B = [x\ y\ w\ h]$, x and y correspond to the center coordinates of the bounding box, and w and h represent the width and height of the bounding box. Similarly, $B_{gt} = [x_{gt}\ y_{gt}\ w_{gt}\ h_{gt}]$ describes the properties of the target box.

First, Wise-IoU v1 constructs a two-layer attention-based bounding box loss as defined in Eq. 7.

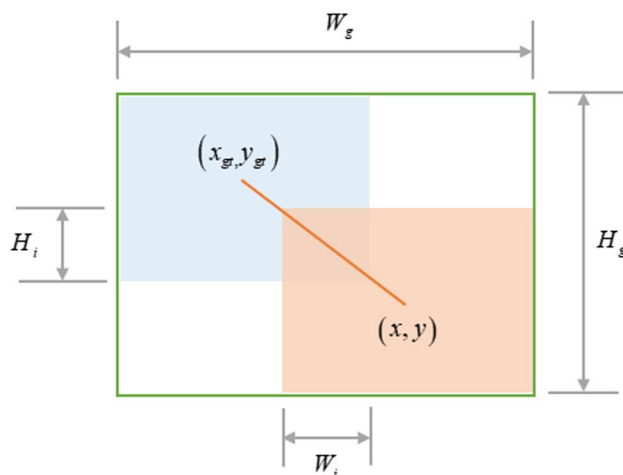


Fig. 6 Schematic diagram of the anchor and target boxes. (x, y) is the coordinate of the center point of the anchor box. (x_{gt}, y_{gt}) is the coordinates of the center point of the target box. W_i and H_i are the length and width of the overlapping rectangular area of the anchor box and the target box respectively. W_g and H_g are the length and width of the minimum enclosing box of the anchor box and the target box, respectively

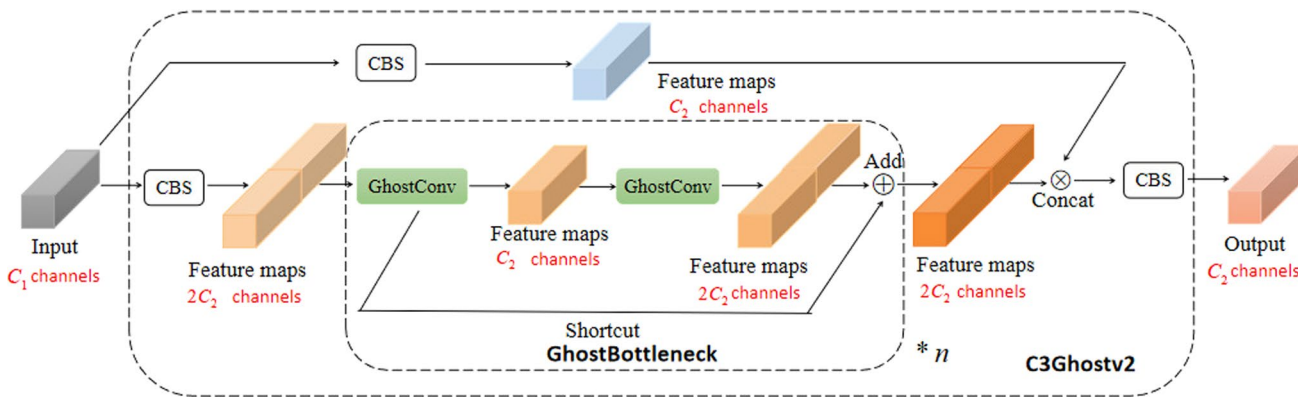


Fig. 5 Structure of C3Ghostv2 module

$$\mathcal{L}_{WIoUv1} = \mathcal{R}_{WIoU} \mathcal{L}_{IoU} \quad (7)$$

Where $\mathcal{R}_{WIoU} \in [1, e)$ is the penalty term for the distance between the center points of the anchor box and the target box, which significantly amplifies the IoU loss of the ordinary-quality anchor box, as shown in Eq. 8. $\mathcal{L}_{IoU} \in [0, 1]$ is used to measure the overlap between the anchor box and the target box, which reduces the contribution of the penalty term of the high-quality anchor box to the loss, as shown in Eq. 9. Thus, the loss optimization focuses on the ordinary-quality anchor box, which is beneficial for the localization of multi-target, hard-to-detect fruits under the interference of other objects in complex scenes.

$$\mathcal{R}_{WIoU} = \exp \left(\frac{(x - x_{gt})^2 + (y - y_{gt})^2}{(W_g^2 + H_g^2)^*} \right) \quad (8)$$

$$\mathcal{L}_{IoU} = 1 - IoU = 1 - \frac{W_i H_i}{wh + w_{gt} h_{gt} - W_i H_i} \quad (9)$$

Then, as shown in Eq. 10, Wise-IoU v3 utilizes outlier degree β to construct a non-monotonic focusing coefficient r , which is then applied to Wise-IoU v1.

$$\mathcal{L}_{WIoUv3} = r \mathcal{L}_{WIoUv1}, r = \frac{\beta}{\delta \alpha^{\beta - \delta}} \quad (10)$$

$$\beta = \frac{\mathcal{L}_{IoU}^*}{\mathcal{L}_{IoU}} \in [0, +\infty)$$

α and δ are hyperparameters, and outlier degree β describes the quality of anchor boxes. A small outlier degree means a high-quality anchor box, to which a small gradient gain r is assigned, so that the bounding box regression focuses on the ordinary-quality anchor box, further enhancing the localization of fruits in complex scenes. Additionally, a smaller gradient gain is assigned to anchor boxes with higher outlier degree, effectively preventing low-quality examples from generating significant harmful gradients, thus improving the model's generalization performance.

Experiments and discussion

Experimental setup

The operating system used for all experiments in this paper was Ubuntu 16.04 LTS. The CPU model was Intel Xeon Silver 4210 with a clock frequency of 2.20 GHz and 64 GB of RAM. The GPU model was GeForce RTX 2080 Ti with 11 GB of VRAM and 125 GB of memory. The model training environment was based on the PyTorch deep learning

framework and utilized Python 3.8 as the programming language. CUDA 10.2 and CUDNN 8.2.1 were used to accelerate the GPU. To enable a fair comparison between the results of all the experimental configurations, the hyper-parameters for the YOLO-based models were standardized. The input images in the network were set to 640 \times 640 pixels, and the batch size was set to 32. Moreover, the models were trained for 150 epochs with an initial learning rate of 0.001 and a weight decay of 0.0005.

Evaluation metric

In this paper, precision, recall, and average precision (AP) were used as evaluation metrics for model detection accuracy. Precision is defined as the ratio of correctly predicted positive samples by the model to all the samples predicted as positive. Recall indicates the proportion of positive samples correctly identified as positive by the model out of all positive samples. It measures the ability of the model to correctly recognize positive samples. Average precision (AP) is defined as the area under the precision and recall curve at various detection thresholds. A higher AP value indicates better detection performance of the model. Mean average precision (mAP) is defined as the average AP across multiple object categories, measuring the model's detection performance for all object categories. Precision, recall, average accuracy, mean average precision are computed using Eqs. 11–14:

$$Pr = \frac{TP}{TP + FP} \quad (11)$$

$$Rc = \frac{TP}{TP + FN} \quad (12)$$

$$AP = \int_0^1 Pr(Rc) dRc \quad (13)$$

$$mAP = \frac{1}{c} \sum_{i=1}^c AP_i \quad (14)$$

Where true positive (TP) is the true classification result where fruits are correctly detected with IoU area over 0.5 threshold. False positive (FP) refers to the falsely detected fruits, while missed detected fruits are denoted as false negative (FN)

In addition, the lightness of the network model was measured in terms of floating-point operations (FLOPs), the number of model parameters and model size. FLOPs refer to the amount of calculations during the forward propagation

of the network and are used to evaluate the computational complexity of the model.

Experimental results

Comparative analysis of different YOLOv5 models on self-made dataset

To select a suitable baseline model, YOLOv5 models of different depths and widths were trained on self-made fruit dataset, and the results are shown in Fig. 7. The training and validation loss charts show that complex network models have lower bounding box regression loss, objectness loss, and classification loss on the fruit training set. However, their objectness loss increases on the validation set in the later stages of training, such as YOLOv5m and YOLOv5l, implying that the models appear to be overfitted. Additionally, the mAP charts for each model demonstrate that the complex models do not show significant improvement in fruit detection performance.

The model complexity is characterized by the FLOPs, parameters and model size. The more complex the model, the larger the FLOPs, number of parameters and model size. The detection performance and complexity of various models are shown in Table 5. YOLOv5l exhibits the best detection performance for fruits, achieving an mAP of 93.3% at 0.5 IoU(mAP@.5), and an mAP of 84.8% at IoU = 0.5:0.95(mAP@.5:.95). However, it is worth that YOLOv5l is the most complex model among the options with model size of 93 MB, 46.21M parameters and 108 GFLOPs, which is not conducive to deployment on the edge devices for real-time fruit detection and online sorting. Moreover, Although YOLOv5n has the most simplified network architecture, its performance for fruit detection has a large gap compared with other models. The model size of YOLOv5s is 14.5 MB, the parameters are 7.06 M, and the floating-point operations are 16.1GFLOPs. Compared with YOLOv5l, the size of YOLOv5s is simplified by 84.41%, and the model computation is reduced by 85.10%. However, there is only a small decrease in detection performance, with

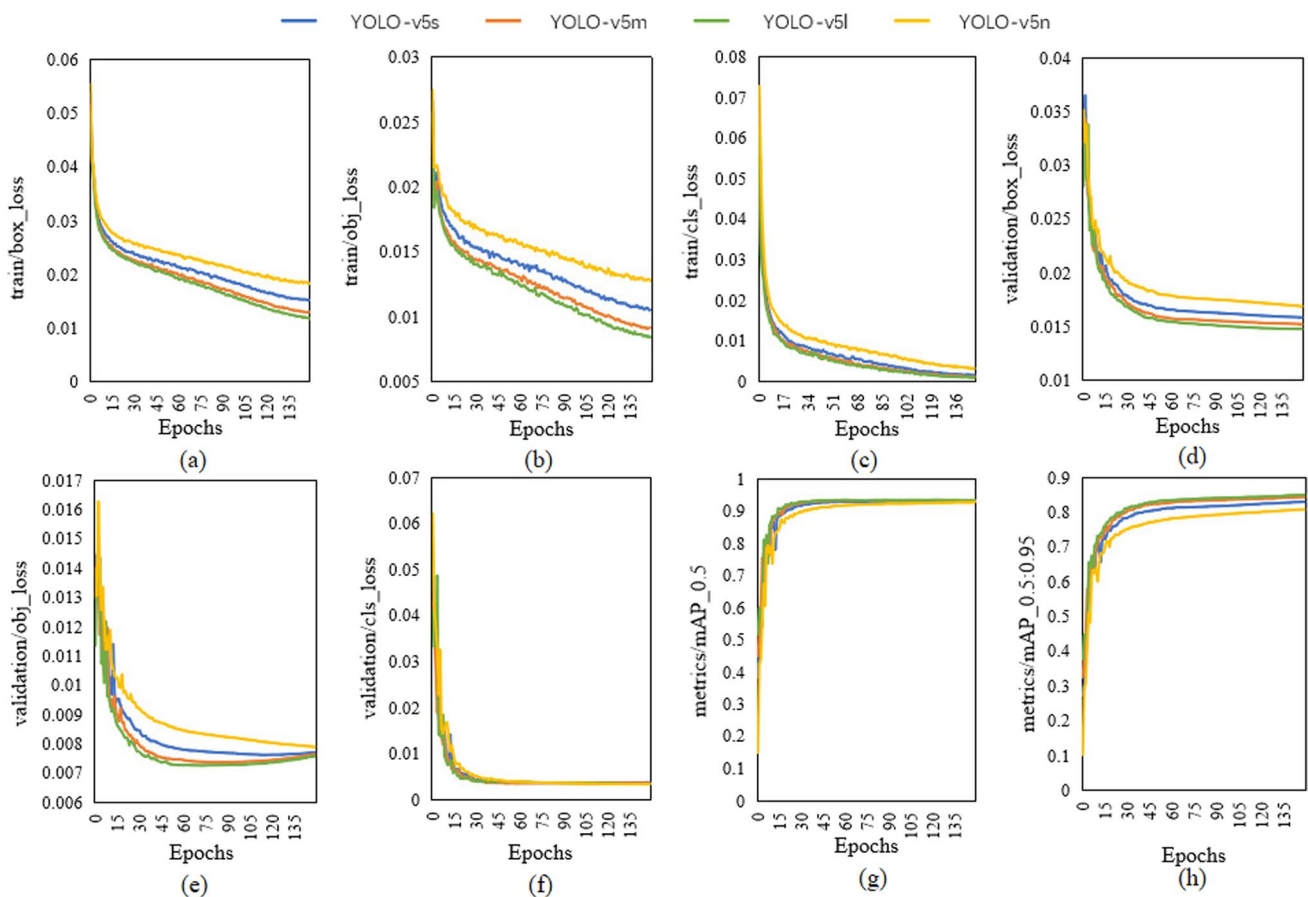


Fig. 7 Training results of different models of YOLOv5. **a–c** Are the bounding box regression loss, objectness loss and classification loss on the training set, respectively. **d–f** Are the bounding box regression loss, objectness loss and classification loss on the validation set,

respectively. **g** Denotes the mAP at 0.5 IoU, and **h** is the mean value of mAP for different IoU thresholds (from 0.5 to 0.95 with a step size of 0.05)

Table 5 Detection performance and complexity parameters of different models of YOLOv5

Model	Precision/%	Recall/%	mAP@.5/%	mAP@.5:.95/%	Size/MB	Params/M	GFLOPs
YOLOv5n	96.7	90.1	92.4	80.5	4.9	2.38	5.2
YOLOv5s	97.8	91	93	83	14.5	7.06	16.1
YOLOv5m	97.7	91.1	93.2	84.3	42.4	20.93	48.2
YOLOv5l	97.5	91.3	93.3	84.8	93	46.21	108

a decrease of only 0.3% in mAP@0.5 and a decrease of 1.8% in mAP@0.5:0.95. In summary, YOLOv5s is more suitable for the detection of multiple varieties fruits in clean and complex backgrounds, so YOLO-v5s was used as the baseline for improvement in this paper.

Results of ablation experiments

The variation of the loss of each improved model during the training process is shown in Fig. 8. Compared with the baseline, the loss of the model on the training set as well as the validation set converged to a larger value after the introduction of C3Ghostv1, while the introduction of C3Ghostv2 in the backbone alleviated the increased loss values caused by the C3Ghostv1 module. In addition, the introduction of the Wise-IoU loss function in the baseline not only accelerated the convergence of the model on the validation set, but also reduced the loss of the model, indicating the effectiveness of the Wise-IoU loss function on model performance improvement. The introduction of C3Ghostv1, C3Ghostv2, and the Wise-IoU loss function in YOLOv5s showed comparable losses to the baseline on the training set. However, it exhibited superior performance on the validation set.

The mean average precision change of each improved model during the training process is shown in Fig. 9. The mAP0.5 and mAP0.5:0.95 of the model decreased after

the introduction of C3Ghostv1, which were improved and slightly above the baseline after further introduction of C3Ghostv2 in backbone. Finally, the mAP was further enhanced by introducing the Wise-IoU bounding box loss function.

The results of ablation experiments on the test set are shown in Table 6. Compared with the baseline, the accuracy and recall of the model decreased by 0.3% and 0.4%, respectively, and both mAP@.5 and mAP@.5:.95 decrease by 0.4% after introducing C3Ghostv1. In addition, parameters decreased by 48%, the computation volume decreased by 49%, and the model size decreased by 46%. The above indicates that the C3Ghostv1 module reduces the detection performance although it can greatly reduce the model complexity. After further introducing the C3Ghostv2 module in the backbone of the network, the accuracy and mAP@.5:.95 both improved by 0.3% compared with the baseline model, which were 0.6% and 0.7% higher than the introduction of C3Ghostv1 alone. Furthermore, there was only a slight increase in parameters and model size. The above shows that expanding the input channels of deep convolution can enhance the network's ability to extract features and improve the detection performance of the model without explosive increase in the model size and computational effort. After introducing Wise-IoU in the baseline, the complexity of the model remained unchanged, while precision and recall both

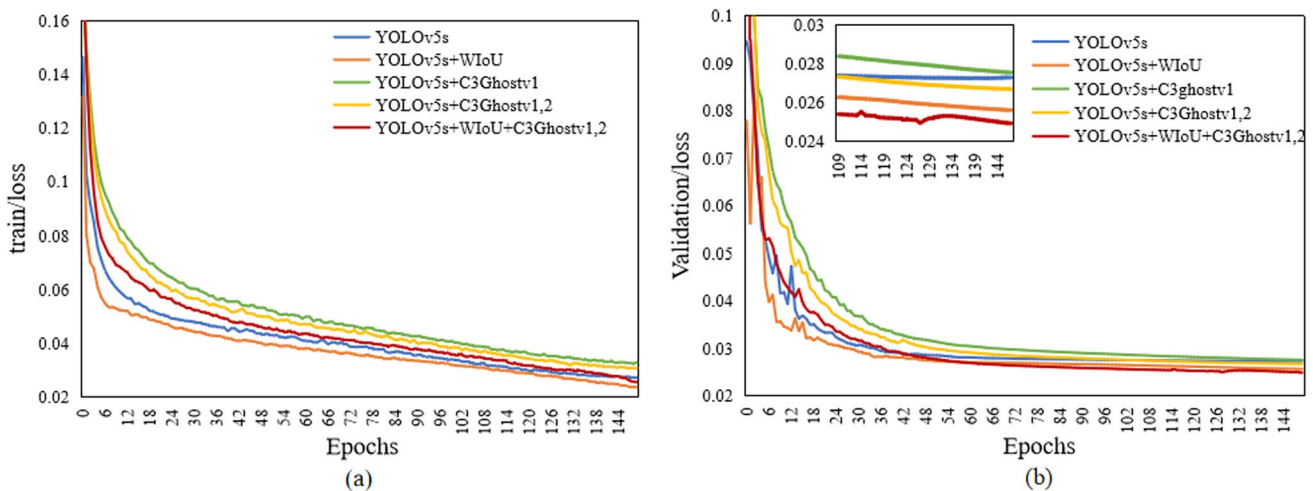


Fig. 8 Losses of different improved algorithms during training, loss is the sum of box_l_oss, obj_loss and cls_loss. **a** The loss variation of the training set during training. **b** The loss variation of the validation set during training

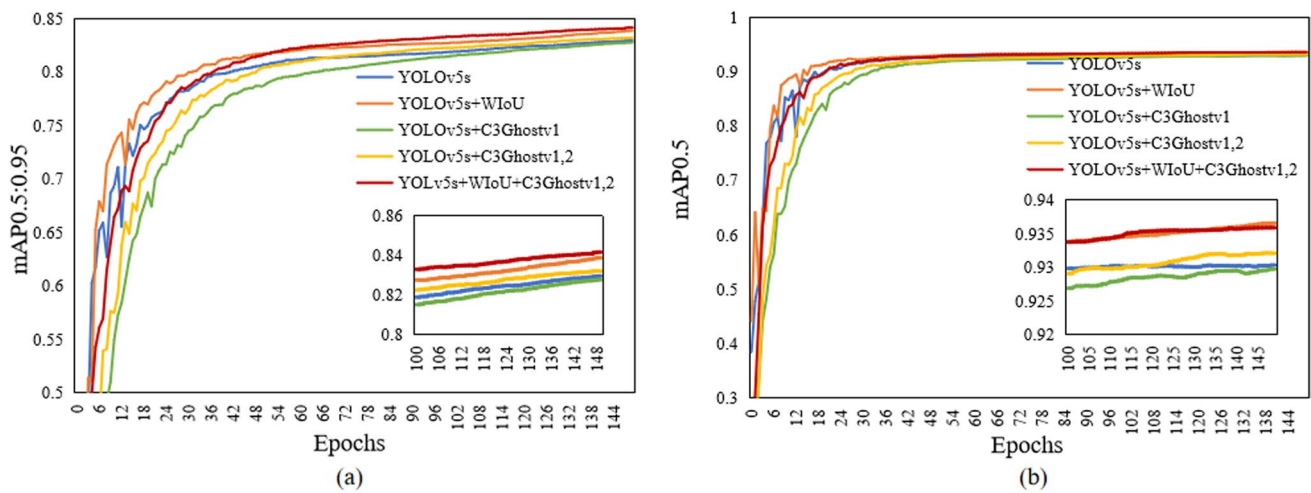


Fig. 9 Mean average accuracy variation of different improved algorithms during the training process. **a** The mean average precision at 0.5 IoU (mAP@.5). **b** The mean average precision at IoU = 0.5:0.95 (mAP@.5:.95)

Table 6 Results of ablation experiments

Model	Precision/%	Recall/%	mAP@.5/%	mAP@.5:.95/%	Size/MB	Params/M	GFLOPs
YOLOv5s(baseline)	97.8	91	93	83	14.5	7.06	16.1
YOLOv5s+WIoU	98.2(+ 0.4)	91.4(+ 0.4)	93.5(+ 0.5)	83.9(+ 0.9)	14.5	7.06	15.9
YOLOv5s+C3Ghostv1	97.5(- 0.3)	89.7(- 0.3)	92.6(- 0.4)	82.6(- 0.4)	7.9(- 6.6)	3.7(- 3.36)	8.2(- 7.9)
YOLOv5s+C3Ghostv1,2	98.1(+ 0.3)	90.5(- 0.5)	93.2(+ 0.2)	83.3(+ 0.3)	9.9(- 4.6)	4.7(- 2.36)	14.9(- 1.2)
YOLOv5s+WIoU+C3Ghostv1,2	98.3(+ 0.5)	90.7(- 0.3)	93.6(+ 0.6)	84.2(+ 1.2)	9.9(- 4.6)	4.7(- 2.36)	14.9(- 1.2)

improved by 0.4%. Moreover, the mAP0.5 and mAP@.5:.95 increased by 0.5% and 0.9% respectively. This indicates the effectiveness of the Wise-IoU loss function in multi-object fruit detection under complex backgrounds. The results show that introducing C3Ghostv1, C3Ghostv2, and the Wise-IoU loss function in the baseline achieves the best balance between detection performance as well as model complexity. Compared with the baseline, the accuracy, mAP@.5 and mAP@.5:.95 increased by 0.5%, 0.6% and 1.2%, respectively. In addition, the parameters decreased by 32%, the computational effort decreased by 1.2 GFLOPs, and the model size decreased by 33%.

Test image visualization

To investigate the reliability of the proposed model, the detection of photos in the test set was performed using the model. Figs. 10 and 11 show the detection results of several images in clean background and complex environment, respectively. The experimental results show that the proposed model can accurately identify the coarse-fine

variety while detecting the fruit location, thereby achieving the task of fruit detection and recognition in a variety of environments.

Comparison of object detection algorithms on the self-made dataset

To verify the effectiveness of the proposed model, several target detection algorithms, such as Faster R-CNN, Single Shot MultiBox Detector (SSD), YOLOv6, YOLOv7, YOLOv8, etc., were trained on the self-made dataset, and the results are compared in Table 7. It can be observed that the proposed network has the best detection performance with mAP0.5 of 93.6% and mAP0.5:0.95 of 84.2%. Additionally, it has a relatively low complexity with a model size of only 9.9 MB and 4.71 M parameters, which is lower than most single-stage detection algorithms and significantly lower than two-stage object detection algorithm, such as Faster R-CNN. The results show that the proposed network achieves the reasonable balance between detection performance and model complexity.

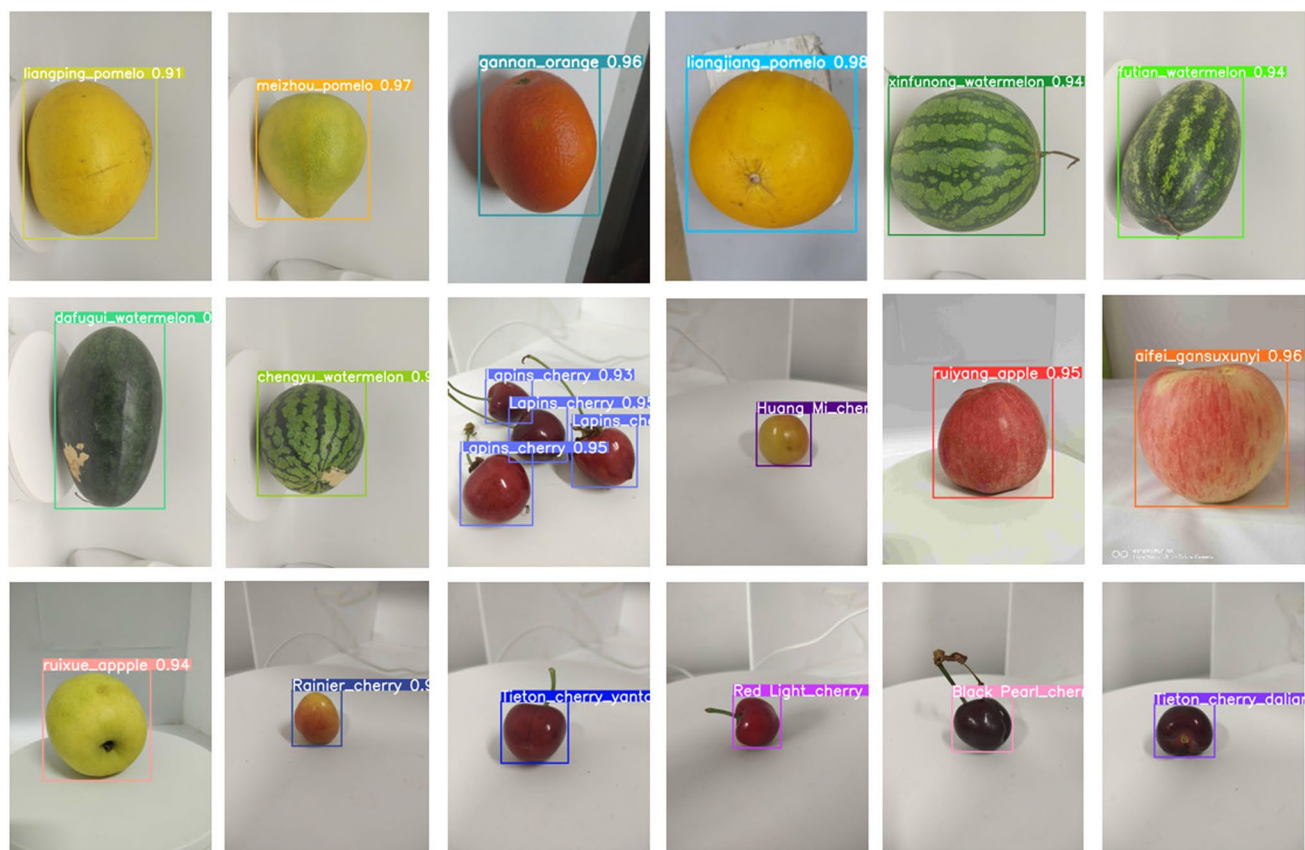


Fig. 10 Detection in clean background

Comparison of the results of different object detection algorithms on public dataset

As a standard dataset to measure the network's ability to detect and classify images, PASCAL VOC 2007 provides 20 kinds of images in different contexts and contains a training set, a validation set and a test set. The training and validation sets contain 5011 images and the test set contains 4952 images. To verify the generalization performance of the proposed model, as shown in Table 8, the model was compared with several other popular detection networks on the public dataset VOC2007. Each model was trained without pre-training weights to eliminate the influence of pre-training weights on the results. It can be found that the two-stage detector Faster R-CNN has the best detection performance with mAP_{0.5} of 66.5%, but its network is the most complex with a model size of 297.83 MB, which is 30 times larger than the proposed network. The proposed algorithm has the best detection performance in single-stage detectors with mAP_{0.5} of 63.8% and mAP_{0.5:0.95} of 38.2%, and it has an advantage in model complexity with a model size of only 9.9 MB and floating-point operations of 14.6 GFLOPs, which achieves an optimal balance between the detection performance and model complexity.

Conclusion

To address the problems of excessive complexity of current fruit detection models and the inability to accurately detect fine variety fruits in complex scenes, this paper proposed a lightweight and high-precision fruit detection model based on a single-stage target detection network YOLOv5 with a self-made fruit dataset. The main findings are as follows: (1) Through image acquisition, manual annotation, dataset division and enhancement, an object detection dataset containing 20 varieties of fruits in clean and complex backgrounds was constructed, solving the current problem of missing public datasets for fine variety fruits detection. (2) By introducing deep convolution, a lightweight network structure based on GhostConv and C3Ghostv1 was proposed with 4.4 M parameters, size of 9.9 MB, and computational volume of 14.9 GFLOPs, which solved the problem of excessive complexity of existing networks and provided support for model deployment in edge devices with limited space and computational resources. (3) By introducing C3Ghostv2 module and Wise-IoU loss function, the mAP@.5 of the model reached 93.6% and mAP@.5:.95 reached 84.2%, which solved the problem of the existing network in low accuracy of coarse-fine

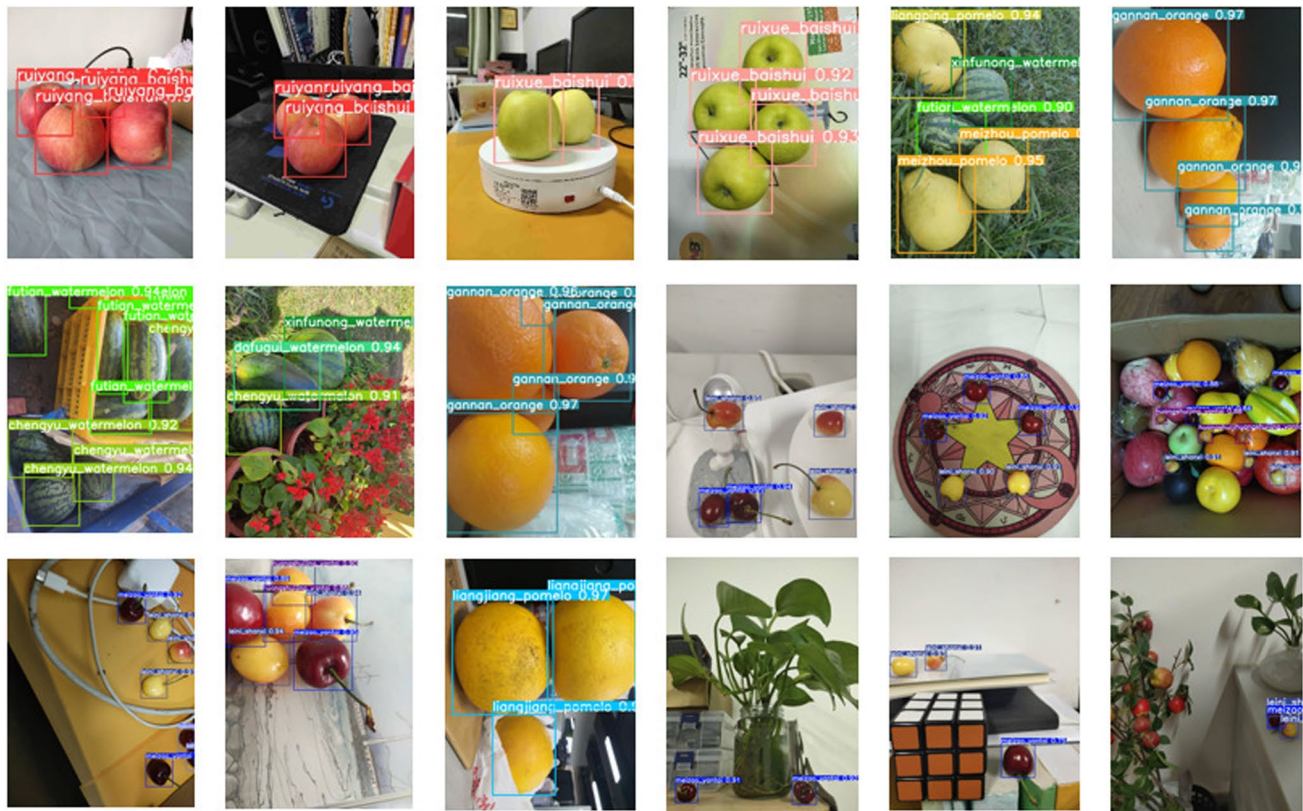


Fig. 11 Detection in complex scenes

Table 7 Comparison of object detection algorithms on the self-made dataset

Model (backbone)	Precision/%	Recall/%	mAP@.5/%	mAP@.5:.95/%	Size/MB	Params/M	GFLOPs
YOLOv5s	97.8	91	93	83	14.5	7.06	16.1
YOLOv5s (ShuffleNet V2)	96.4	88.9	92	80	8.1	3.84	8.1
YOLOv5s (MobileNet V3)	97.2	89.3	92.4	80.9	11.5	5.58	2.7
YOLOv6s	–	–	92.5	83	20.51	9.70	45.2
YOLOv7-tiny	96.1	89	91.8	78.3	12.4	6.06	13.3
YOLOv8n	97.2	89.8	92.6	84.1	6.5	3.16	8.9
SSD300 (VGG16)	–	–	89.1	77.5	200.59	26.28	–
Faster R-CNN (Resnet50)	–	–	90.5	78.1	297.83	39.02	–
Improved YOLOv5s	98.3	90.7	93.6	84.2	9.9	4.71	14.6

Bold values indicate the best results under the corresponding evaluation metrics

variety fruits detection in complex environment and satisfied real-time detection, online grading and fast sorting of many kinds of fruits in precision agriculture.

Although the generalization performance has been validated on the public dataset PASCAL VOC2007, the

proposed model has limitations in outdoor complex fields and detectable fruit varieties. In the future, the dataset can be expanded by adding multiple fruit varieties and complex field scenarios to improve the generalizability of the model.

Table 8 Comparison of object detection algorithms on public dataset

Model (backbone)	Precision/%	Recall/%	mAP@.5/%	mAP@.5:.95/%	Size/MB	Params/M	GFLOPs
YOLOv5s	65.7	60.7	61.9	34.9	14.5	7.06	15.9
YOLOv5s (ShuffleNet V2)	59.0	53.1	53.2	28.1	8.1	3.84	8.1
YOLOv5s (MobileNet V3)	58.4	55.4	55.6	30.9	11.5	5.58	2.7
YOLOv6s	–	–	59.3	35.1	18.51	9.02	45.2
YOLOv7-tiny	63.9	63.1	63.5	37.1	12.4	6.07	13.3
YOLOv8n	65.8	60.1	62.8	38.5	6.5	3.15	8.7
SSD300 (VGG16)	–	–	63.6	32.3	200.59	26.28	–
Faster R-CNN (Resnet50)	–	–	66.5	33.7	297.83	39.02	–
Improved YOLOv5s	67.7	59.6	63.8	38.2	9.9	4.71	14.9

Bold values indicate the best results under the corresponding evaluation metrics

Appendix: variety information of the samples

Coarse-variety	Fine-variety	Place of production	Variety information	Coarse-variety	Fine-variety	Place of production	Variety information
Apple	Ruiyang	Baishui Experimental Station in Shaanxi	Smooth appearance, round or slightly flattened shape, bright red color, crisp and juicy, balanced sweet and sour taste, good storage ability		Red Light	Yantai City, Shangdong Province	Medium to large-sized fruit, bright red color, sweet taste, firm texture, crisp and refreshing mouthfeel
	Ruixue	Baishui Experimental Station in Shaanxi	Round or slightly flattened shape, light yellow or pale green color, crisp and refreshing, mainly sweet in taste, good storage ability		Huang Mi	Yantai City, Shangdong Province	Medium to large-sized fruit, bright yellow color, semi-transparent, sweet and sour taste, soft texture, juicy flesh
	Aifei	Xunyi City, Gansu Province	Slightly flattened shape, rose red color crisp and refreshing, purely sweet without acidity, good storage ability		Lapins	Shaanxi Province	Medium to large-sized fruit, deep red color with purple spots, mostly sweet with minimal acidity, firm texture
Cherry	Black Pearl	Yantai City, Shangdong Province	Small to medium-sized fruit, color, sweet, juicy, flavorful taste, firm and crunchy texture		Rainier	Shaanxi Province	Medium-sized fruit, heart shaped, yellow with a reddish tinge, crunchy sweet flavor
					Tieton	Yantai City, Shangdong Province	Medium to large-sized fruit, deep red color, abundant and juicy flesh, sweet taste

Coarse-variety	Fine-variety	Place of production	Variety information	Coarse-variety	Fine-variety	Place of production	Variety information
	Tieton	Dalian City, LiaoNing Province	Large-sized fruit, deep red color, abundant and juicy flesh, sweet taste		Liangjiang	Chongqing City	Flat-round shape, golden color, thin and smooth skin, translucent and pale yellow flesh
	Pioneer	Yantai City, Shangdong Province	Medium to large-sized fruit, deep red or dark red-purple color, rich sweet taste, crisp and juicy, strong disease resistance	Orange	Gannan Orange	Ganzhou City, Jiangxi Province	Flat-round or elliptical in shape, orange-yellow or orange-red skin, light yellow flesh, sweet and sour taste
Watermelon	Futian	FutianTown, Fujian Province	Large-sized fruit, outer skin is a vibrant green color with stripes, flesh is a bright red color, juicy and refreshing in taste		Honey Mandarin	Hunan Province	Nearly round shape, orange-red color, smooth skin, sweet, crispy, juicy, seedless
	Dafugui	Guangxi Province	Oval-shaped, deep green in color, indistinct stripes, red-fleshed, high sweetness	<p>Acknowledgements Research supported by Foundation of Key Research and Development Program of Shaanxi province (2023-YBNY-229, 2021NY-179), Undergraduate Training Program for Innovation and entrepreneurship plan (202210712189, S202210712661, X202210712348).</p> <p>Data availability The data that support the findings of this study are available from the corresponding author upon reasonable request.</p> <p>Declarations</p> <p>Conflict of interest The authors declare that they have no conflict of interest.</p>			
	Xinfunong	Guangxi Province	Round-shaped, green skin with dark green stripes, red flesh, crispy texture, seedless	<p>References</p> <ol style="list-style-type: none"> 1. B. Yuan, C. Chen, Evolution of a development model for fruit industry against background of rising labor intensive or extensive adjustment? Sustainability (2019). https://doi.org/10.3390/su11143864 2. Q. Li, W. Jia, M. Sun, S. Hou, Y. Zheng, A novel green apple segmentation algorithm based on ensemble U-Net under complex orchard environment. Comput. Electron. Agric. 180, 105900 (2021). https://doi.org/10.1016/j.compag.2020.105900 3. Y. Tang, H. Zhou, H. Wang, Y. Zhang, Fruit detection and positioning technology for a <i>Camellia oleifera</i> C. Abel orchard based on improved YOLOV4-tiny model and binocular stereo vision. Expert Syst. Appl. 211, 118573 (2023). https://doi.org/10.1016/j.eswa.2022.118573 4. L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, M. Pietikäinen, Deep learning for generic object detection: a 			
	Chengyu	Hunan Province	Small-sized, high-spherical fruit, green skin with dark green stripes, orange-yellow flesh, with seeds				
Pomelo	Meizhou	Meizhou City, Guangdong Province	Gourd-shaped, smooth and yellow skin, sweet and refreshing flesh, with a honey-like taste				
	Liangping	Chongqing City	Flat-round shape, golden color, thin and smooth skin, translucent and pale yellow flesh				

- survey. *Int. J. Comput. Vis.* **128**, 261–318 (2020). <https://doi.org/10.1007/s11263-019-01247-4>
5. L. Fu, Y. Feng, J. Wu, Z. Liu, F. Gao, Y. Majeed, A. Al-Mallahi, Q. Zhang, R. Li, Y. Cui, Fast and accurate detection of kiwifruit in orchard using improved YOLOV3-tiny model. *Precis. Agric.* **22**, 754–776 (2021). <https://doi.org/10.1007/s11119-020-09754-y>
 6. C. Qi, J. Gao, S. Pearson, H. Harman, K. Chen, L. Shu, Tea chrysanthemum detection under unstructured environments using the TC-YOLO model. *Expert Syst. Appl.* **193**, 116473 (2022). <https://doi.org/10.1016/j.eswa.2021.116473>
 7. R. Girshick, Fast R-CNN, in *Proceedings of the IEEE International Conference on Computer Vision* (2015), pp. 1440–1448
 8. F. Networks, Fast and accurate detection of kiwifruit in orchard using improved YOLOV3-tiny model. *Adv. Neural Inf. Process. Syst.* **28** (2015). https://proceedings.neurips.cc/paper_files/paper/2015/file/14bfa6bb14875e45bba028a21ed38046-Paper.pdf
 9. K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask R-CNN, in *Proceedings of the IEEE International Conference on Computer Vision* (2017), pp. 2961–2969
 10. J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: unified, real-time object detection, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 779–788. <https://doi.org/10.1109/CVPR.2016.91>
 11. C. Li, L. Li, H. Jiang, K. Weng, Y. Geng, L. Li, Z. Ke, Q. Li, M. Cheng, W. Nie, et al., YOLOv6: a single-stage object detection framework for industrial applications. *arXiv Preprint* (2022). <http://arxiv.org/abs/2209.02976>, <https://doi.org/10.48550/arXiv.2209.02976>
 12. C.-Y. Wang, A. Bochkovskiy, H.-Y.M. Liao, YOLOv7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2023), pp. 7464–7475. <https://doi.org/10.48550/arXiv.2207.02696>
 13. W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A.C. Berg, SSD: single shot multibox detector, in *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*. (Springer, 2016), pp. 21–37. https://doi.org/10.1007/978-3-319-46448-0_2
 14. C. Wang, Z. Wang, K. Li, R. Gao, L. Yan, Lightweight object detection model fused with feature pyramid. *Multimed. Tools Appl.* **82**(1), 601–618 (2023). <https://doi.org/10.1007/s11042-022-12127-4>
 15. L. Zhu, Z. Xie, J. Luo, Y. Qi, L. Liu, W. Tao, Dynamic object detection algorithm based on lightweight shared feature pyramid. *Remote Sens.* **13**(22), 4610 (2021). <https://doi.org/10.3390/rs13224610>
 16. T.-H. Liu, R. Ehsani, A. Toudeshki, X.-J. Zou, H.-J. Wang, Identifying immature and mature pomelo fruits in trees by elliptical model fitting in the Cr–Cb color space. *Precis. Agric.* **20**, 138–156 (2019). <https://doi.org/10.1007/s11119-018-9586-1>
 17. L. Yu, J. Xiong, X. Fang, Z. Yang, Y. Chen, X. Lin, S. Chen, A litchi fruit recognition method in a natural environment using RGB-D images. *Biosyst. Eng.* **204**, 50–63 (2021). <https://doi.org/10.1016/j.biosystemseng.2021.01.015>
 18. Z.S. Pothan, S. Nuske, Texture-based fruit detection via images using the smooth patterns on the fruit, in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. (IEEE, 2016), pp. 5171–5176. <https://doi.org/10.1109/ICRA.2016.7487722>
 19. A.J. Prakash, P. Prakash, An intelligent fruits classification in precision agriculture using bilinear pooling convolutional neural networks. *Vis. Comput.* **39**(5), 1765–1781 (2023). <https://doi.org/10.1007/s00371-022-02443-z>
 20. F. Gao, L. Fu, X. Zhang, Y. Majeed, R. Li, M. Karkee, Q. Zhang, Multi-class fruit-on-plant detection for apple in snap system using faster R-CNN. *Comput. Electron. Agric.* **176**, 105634 (2020). <https://doi.org/10.1016/j.compag.2020.105634>
 21. H. Mirhaji, M. Soleymani, A. Asakereh, S.A. Mehdizadeh, Fruit detection and load estimation of an orange orchard using the yolo models through simple approaches in different imaging and illumination conditions. *Comput. Electron. Agric.* **191**, 106533 (2021). <https://doi.org/10.1016/j.compag.2021.106533>
 22. R. Gai, N. Chen, H. Yuan, A detection algorithm for cherry fruits based on the improved YOLO-v4 model. *Neural Comput. Appl.* (2021). <https://doi.org/10.1007/s00521-021-06029-z>
 23. Y. Wang, G. Yan, Q. Meng, T. Yao, J. Han, B. Zhang, DSE-YOLO: detail semantics enhancement YOLO for multi-stage strawberry detection. *Comput. Electron. Agric.* **198**, 107057 (2022). <https://doi.org/10.1016/j.compag.2022.107057>
 24. J. Yao, J. Qi, J. Zhang, H. Shao, J. Yang, X. Li, A real-time detection algorithm for kiwifruit defects based on YOLOv5. *Electronics* **10**(14), 1711 (2021). <https://doi.org/10.3390/electronics10141711>
 25. X. Wang, Z. Wu, M. Jia, T. Xu, C. Pan, X. Qi, M. Zhao, Lightweight SM-YOLOv5 tomato fruit detection algorithm for plant factory. *Sensors* **23**(6) (2023). <https://www.mdpi.com/1424-8220/23/6/3336>
 26. Y. Ma, D. Liu, H. Yang, DGCC-fruit: a lightweight fine-grained fruit recognition network. *J. Food Meas. Charact.* (2023). <https://doi.org/10.1007/s11694-023-02020-5>
 27. P. Yan, Q. Sun, N. Yin, L. Hua, S. Shang, C. Zhang, Detection of coal and gangue based on improved YOLOv5. 1 which embedded SCSE module. *Measurement* **188**, 110530 (2022). <https://doi.org/10.1016/j.measurement.2021.110530>
 28. Z. Ying, Z. Lin, Z. Wu, K. Liang, X. Hu, A modified-YOLOv5s model for detection of wire braided hose defects. *Measurement* **190**, 110683 (2022). <https://doi.org/10.1016/j.measurement.2021.110683>
 29. K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, C. Xu, GhostNet: more features from cheap operations, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020), pp. 1580–1589. <https://doi.org/10.48550/arXiv.1911.11907>
 30. M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.-C. Chen, MobileNetV2: inverted residuals and linear bottlenecks, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 4510–4520. <https://doi.org/10.1109/cvpr.2018.00474>
 31. Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, D. Ren, Distance-IoU loss: faster and better learning for bounding box regression, in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34 (2020), pp. 12993–13000. <https://doi.org/10.1609/aaai.v34i07.6999>
 32. J. Yu, Y. Jiang, Z. Wang, Z. Cao, T. Huang, UnitBox: an advanced object detection network, in *Proceedings of the 24th ACM International Conference on Multimedia* (2016), pp. 516–520. <https://doi.org/10.1145/2964284.2967274>
 33. Z. Tong, Y. Chen, Z. Xu, R. Yu, Wise-IoU: bounding box regression loss with dynamic focusing mechanism. *arXiv Preprint* (2023). <http://arxiv.org/abs/2301.10051>, <https://doi.org/10.48550/arXiv.2301.10051>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.