agronomy

MDPI

Article

Detection and Counting of Small Target Apples under Complicated Environments by Using Improved YOLOv7-tiny

Li Ma [1], Liya Zhao [1], Zixuan Wang [1], Jian Zhang [2,3,*] and Guifen Chen [4,*]

[1] College of Information Technology, Jilin Agricultural University, Changchun 130118, China; mali@jlau.edu.cn (L.M.); zhaoliya@mails.jlau.edu.cn (L.Z.); 2012100221@mails.jlau.edu.cn (Z.W.)
[2] Faculty of Agronomy, Jilin Agricultural University, Changchun 130118, China
[3] Department of Biology, University of Columbia Okanagan, Kelowna, BC V1V 1V7, Canada
[4] Institute of Technology, Changchun Humanities and Sciences College, Changchun 130118, China
* Correspondence: jian.zhang@ubc.ca (J.Z.); chenguifen@jlau.edu.cn (G.C.)

Abstract: Weather disturbances, difficult backgrounds, the shading of fruit and foliage, and other elements can significantly affect automated yield estimation and picking in small target apple orchards in natural settings. This study uses the MinneApple public dataset, which is processed to construct a dataset of 829 images with complex weather, including 232 images of fog scenarios and 236 images of rain scenarios, and proposes a lightweight detection algorithm based on the upgraded YOLOv7-tiny. In this study, a backbone network was constructed by adding skip connections to shallow features, using P2BiFPN for multi-scale feature fusion and feature reuse at the neck, and incorporating a lightweight ULSAM attention mechanism to reduce the loss of small target features, focusing on the correct target and discard redundant features, thereby improving detection accuracy. The experimental results demonstrate that the model has an mAP of 80.4% and a loss rate of 0.0316. The mAP is 5.5% higher than the original model, and the model size is reduced by 15.81%, reducing the requirement for equipment; In terms of counts, the MAE and RMSE are 2.737 and 4.220, respectively, which are 5.69% and 8.97% lower than the original model. Because of its improved performance and stronger robustness, this experimental model offers fresh perspectives on hardware deployment and orchard yield estimation.

Keywords: YOLOv7-tiny-Apple; small target; fruit detection and counting; digital agriculture

Citation: Ma, L.; Zhao, L.; Wang, Z.; Zhang, J.; Chen, G. Detection and Counting of Small Target Apples under Complicated Environments by Using Improved YOLOv7-tiny. *Agronomy* **2023**, *13*, 1419. https://doi.org/10.3390/agronomy13051419

Academic Editor: Roberto Marani

Received: 30 April 2023
Revised: 17 May 2023
Accepted: 19 May 2023
Published: 20 May 2023

1. Introduction

Apples are a vital agricultural commodity worldwide and a significant contributor to economic development in the agricultural sector. In the United States, Washington State alone accounts for 67% of the entire apple production [1]. Apples are not only delicious but are also packed with a variety of nutrients, including vitamins C and E. These nutritional benefits have made apples a popular choice among consumers. [2,3]. The majority of orchard management today is performed by human labor, which is not only physically taxing but also ineffective, expensive, and prone to error [4].

The integration of intelligence in orchards, driven by the advancement of smart agriculture [5], has emerged as a crucial factor in obtaining precise product information. Nonetheless, detecting fruits accurately in natural environments presents significant challenges. Issues such as fluctuating lighting conditions, overlapping shading, and the resemblance between distant small fruits and the background can lead to inaccuracies in fruit detection [4,6,7]. Accurate fruit detection holds great research value and practical significance for the development of automated harvesting and yield estimation in orchards. Therefore, it is an important area of focus for researchers in this field.

In recent years, there have been some notable advancements in the automated detection and management of orchards, which can be categorized into two main approaches: traditional methods and deep learning algorithms. Traditional image processing methods

have primarily focused on extracting information such as color, shape, and basic features from images [8]. These extracted features are then classified using techniques such as support vector machines or artificial neural networks [9], forming the basis for fruit detection and segmentation [10]. Tian et al. [11] proposed a localization technique based on image depth information to find the circle center, fit the contour, and improve recognition accuracy with a recognition rate of 96.61%. Lin et al. [12] developed a support vector machine-based segmentation algorithm for citrus detection using density clustering to reduce the effect of the environment with excellent performance. Wang et al. [13] used the Retinex image enhancement algorithm, and two-dimensional discrete wavelet transform to apply it to fruit detection based on color features and texture features, with a final accuracy of 85.6%. Wang et al. [14] proposed an algorithm for identifying and locating obscured apples based on K-mean clustering, with a higher localization rate and 89% recognition accuracy than Hough transform and contour curvature methods. Zhang et al. [15] proposed an insulator profile detection method based on an edge detection algorithm. The Canny algorithm was selected as the main algorithm for insulator profile detection, which provides an efficient, accurate, and reliable way for the automated detection of insulator profiles with some practical value.

This paper has selected a deep learning algorithm with strong autonomous learning and feature extraction abilities to address these challenges. These algorithms demonstrate strong robustness and generalization capabilities in fruit detection, making them highly effective for automating orchard processes. Li et al. [16] proposed a target detection algorithm based on YOLOv4-tiny detection of green peppers, combining attention mechanism and multi-scale prediction ideas, with an average accuracy of 95.11%, model size of 30.9 MB, and FPS of 89. Tian et al. [4] presented an enhanced YOLOv3 model for detecting apples at various growth stages. They incorporated DenseNet feature enhancement propagation and enhanced feature reuse techniques to enhance the feature layers at low and medium resolutions. Their model achieved an average detection time that can handle a resolution of 3000 $\times$ 3000 per frame, outperforming the original model and Faster-R-CNN. Zhang et al. [17] proposed a YOLOv4-tiny-based apple detection model with the backbone introducing GhostNet feature extraction network with a CA attention mechanism, introducing depth-separable convolution in the neck and YOLO head for reconstruction, and FPN adding CA attention module to enhance the feature extraction of small targets with an average accuracy of 95.72%. The model has a 37.9 MB model size and a 45.2 FPS. Tu et al. [18] proposed an improved method based on Multi-Scale Fast Region Convolutional Neural Networks (MS-FRCNN) to detect lower-level features by merging feature maps from shallower convolutional feature maps, used in the region of interest pooling, effectively improving the detection of small passionfruit. Uddhav Bhattarai and Manoj Karkee [19] proposed CountNet, a weakly supervised flower/fruit counting network based on deep learning, to learn the number of objects from image-level annotations as input, yielding good MAE and EMSE in an orchard setting. Qian et al. [20] proposed HOG+SVM and an improved YOLOv5-based method for fast recognition of multiple apples in complex environments with parameter reconstruction, the inclusion of an attention mechanism module and fine-tuning of the loss function to better extract the features of different apples and improve the recognition ability of the model. Jan Weyler et al. [21] proposed a method to predict the bounding boxes of crops and weeds automatically, as well as the key points of leaves, with a good ability to estimate leaf numbers, and this method achieved excellent performance in complex scenarios with overlapping plants at different growth stages compared to Mask R-CNN. Chen et al. [22] proposed a Citrus-YOLOv7 model to detect citrus, introducing a small target detection layer, lightweight convolution, and CBAM attention mechanism to achieve multi-scale feature extraction and fusion with an average accuracy of 97.29%, an average detection time of 69.38 ms, and a model size of 24.26 MB. Sun et al. [23] proposed an optimized Retinanet-PVTv2 that introduced a gradient coordination mechanism to detect small green fruits/begonia in a nocturnal environment and showed APs of 85.2% and 61.0% on the NightFruit and Gala datasets, respectively. Yonis Gulzar [24] proposed a

TL-MobileNetV2 model used to classify 40 fruits, removing the classification layers present in the MobileNetV2 architecture, adding five different layers, and also using different pre-processing and model adjustments, effectively improving the efficiency and accuracy of the model with an accuracy of 99%. Normaisharah Mamat et al. [25] proposed an automatic image annotation advancement method using repetitive annotation tasks for the automatic annotation of objects, which was evaluated on different YOLO networks, and the results showed that the proposed method is fast and highly accurate in fruit classification, providing great value in image annotation. Yasir Hamid [26] proposed a deep convolutional neural network model using MobileNetV2 architecture and data augmentation techniques to implement an intelligent seed classification system, and the results show that it has high accuracy and is important for the sustainability of seed analysis. Sonam Aggarwal [27] proposed an artificial intelligence-based stacked integration approach to predict protein subcellular localization, which combines the three powerful pre-trained models, and the results showed that combining different weak convolutional neural networks gave better predictions than individual models with better network performance.

This study proposes an improved YOLOv7-tiny detection and counting network for small target apples to better cope with the inevitable realities of natural environments, complex backgrounds, and complex weather. The dataset was enriched with data enhancement techniques, adding three innovations to the backbone and neck of the model, improving detection accuracy for small target apples, and reducing the number of parameters in the model. The results show that the model has high robustness and generalization performance in detection and counting and can be deployed to mobile applications to facilitate automated orchard management.

The following are the contributions of the authors' work in this study:

(1) Construction of an Apple dataset with complex weather using the MinneApple public dataset, cropping and marking this, and simulating rain and fog scenarios using data augmentation techniques;

(2) Borrowing from DenseNet idea to construct a backbone network by adding to shallow features through skip connections to reduce the loss of small target features;

(3) Multi-scale fusion using P2BiFPN to detect small fruits with low resolution at a distance;

(4) A lightweight ULSAM attention mechanism is used to discard redundant features without increasing the model volume.

This study aims to enhance the model's ability to detect and count small target apples with complex backgrounds in natural environments and to adapt to complex weather conditions.

2. Materials and Methods

2.1. Construction of the Apple Dataset

2.1.1. Image Acquisition and Pre-Processing

The MinneApple dataset was used for this experiment. It was gathered at the University of Minnesota Horticultural Research Center (HRC) between June 2015 and September 2016 [28]. Video clips from various parts of the orchard were collected using a standard Samsung Galaxy S4 phone, and images were then captured to create the desired images "https://conservancy.umn.edu/handle/11299/206575 (accessed on 10 September 2022)". The dataset used in this study was collected from both sunny and shaded areas of tree rows, spanning multiple days to capture diverse lighting conditions. The dataset was challenging, as it included heavily shadowed and crowded scenes with apples that were difficult to locate. Additionally, the dataset comprised various regions with different apple types. The dataset selected in this paper is the training set of Image images in the detection in MinneApple. In this study, we cropped and labeled the images and classified them into three categories: GreenApple, Red1Apple, and Red2Apple. The GreenApple tree has a thick trunk with smooth, green-colored outer skin. The Red1Apple tree is loosely structured, with a yellow-green base color on the fruit and a slightly powdery texture. The Red2Apple

tree is conical in shape, with smooth and evenly colored outer skin, appearing bright red. GreenApple is mostly under bright lighting conditions, and they have an extremely similar background. Red1Apple is in weak lighting conditions with insufficient brightness, while Red2Apple is in backlit and dim lighting conditions. For these three categories of fruit, we pre-processed the data.

The dataset is mainly for the detection of small target apples, as shown in Figure 1. We selected some images to crop them into 841 images of different sizes, reducing the input image size from 1280 × 720 to images of different resolution sizes, including 302 GreenApple of 320 × 320 resolution size, 252 Red1Apple of 320 × 320 resolution size and 410 × 410 resolution size, and 287 Red2Apple of 320×320 resolution size. The cropped images were manually annotated using the online web-based tagging tool makesense.ai "https://www.makesense.ai/ (accessed on 20 September 2022)" and finally exported in the generated .voc format, which was converted to YOLO format to complete the construction of the initial dataset.

Figure 1. Cropping of the data set.

2.1.2. Data Augmentation

A dataset with complex weather was constructed using image enhancement methods to simulate different levels of rain and fog scenarios. This is shown in Figure 2. This is used to test the generalization and robustness of our model. From the overall dataset of three types of apples, 80 images were randomly selected from each fruit class for single-fold data augmentation. The iaa.Fog() function was used to simulate foggy weather conditions, and then the iaa.Rain(speed = (0.1, 0.3)) function was used to simulate rainy weather conditions. This completed the enhancement processing under different weather conditions. Data cleaning was performed in this process, and the final enhanced dataset obtained contained 232 foggy weather images and 236 rainy weather images. The original images and the enhanced images form a dataset with complex weather, containing a total of 829 images, as shown in Table 1. The constructed dataset is randomly partitioned into training, testing, and validation sets according to the ratio of 7:2:1. Finally, 586 images of the training set, 164 images of the test set, and 66 images of the validation set were obtained.

Table 1. The volume of data before and after enhancement and for different weather.

Category	Raw Data Volume	Weather Data Enhancement			Experimental Data Volume
		Normal Scenario	Fog Scenario	Rain Scenario	
GreenApple	302	142	79	76	297
Red1Apple	252	92	73	80	245
Red2Apple	287	127	80	80	287

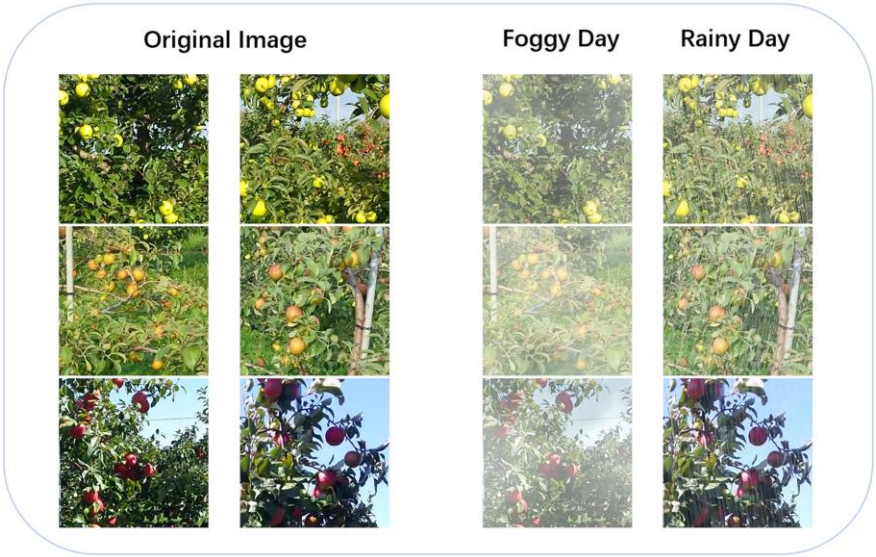


Figure 2. Data enhancement in different weather.

2.2. Selection of Models

CNN-based networks are crucial for fruit detection in terms of target detection. Some of the more established and well-liked detection networks include the Faster Rcnn [29], SSD [30], CenterNet [31], and YOLO series. Object detection is transformed into a regression problem by the single-stage object detection network known as YOLO. By processing the image with a single CNN, YOLO can directly obtain the class and location coordinates of the object. This end-to-end detection network greatly increases the speed of detection. The YOLOv7 network model [32] of the YOLO family has good detection performance. One of them, YOLOv7-tiny, is a lightweight model of the YOLOv7 series. Ease of deployment and detection with accuracy and real-time performance is key to small-target apple detection. In this regard, we compared the lightweight YOLOv7-tiny model with Faster Rcnn-VGG, SSD-MobileNetV2, and CenterNet-ResNet50 for the study. As shown in Table 2, it can be seen that the YOLOv7-tiny detection effect is more pronounced, while the detection speed is fast and model volume is small, and can provide some subsequent mobile deployment technical support.

Table 2. Results of the inter-model selection comparison.

Model	F1 Score/%	Precision/%	Recall/%	mAP0.5/%	Params/MB	FPS
YOLOv7-tiny	72.3	77.2	68.1	74.9	6.01	126.56
Faster Rcnn-VGG	50.33	41.16	66.51	55.95	547.0	24.90
SSD-MobileNetV2	47.3	95.07	31.93	64.16	16.6	81.98
CenterNet-ResNet50	48.3	95.48	40.56	71.22	131.0	69.74

2.3. YOLOv7-tiny-Apple Construction

The YOLOv7-tiny model consists of the backbone, neck, and head. The backbone network is mainly composed of CBL, ELAN-T, and SPPCSPC to perform feature extraction. To enhance the perceptual field of the model and strengthen the features, the neck network PANet merges the characteristics learned at various scales. The head is convolved by 1×1 to transform the features into the final prediction information to obtain the final prediction results. This model has excellent speed and outstanding network performance. For the detection in the complex background of the natural environment, to solve the problem of distant small fruits, missing contours, and weather interference caused by the decline in detection accuracy, this paper has improved YOLOv7-tiny, as shown in Figure 3. The shallow feature fusion helps to reduce the loss of small targets by borrowing the DenseNet

idea in the backbone part and adding a skip connection(a) in the P3 part. The neck is fused at multiple scales by upgrading PANet to P2BiFPN(b), adding a small target detection layer while removing the large target detection layer, and mixing shallow features with deep features for feature fusion reuse and improved detection accuracy. A lightweight ULSAM attention mechanism(c) is added here to focus on the right target discarding redundant features.

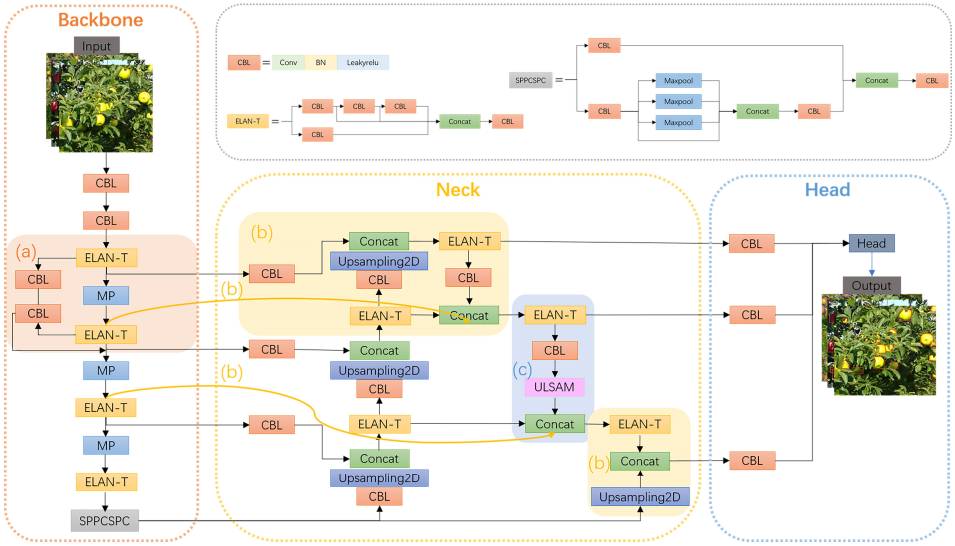


Figure 3. Apple detection and counting model based on YOLOv7-tiny-Apple, (a) skip connection, (b) P2BiFPN, and (c) ULSAM attention mechanism.

2.3.1. Shallow Feature Fusion

Drawing on the DenseNet [33] idea, we make better use of feature information and improve the efficiency of information propagation between layers. DenseNet connects all layers for channel merging and feature reuse, as shown in Figure 4. x_0, x_1,x_2,x_3,x_4 indicate the feature map of each output layer, and h1,h2,h3,h4 indicate the nonlinear transformation; each layer can accept all the previous feature map (l-1) layers [34], and the feature map of each layer is expressed in Equation (1).

$$x_l = H_l[x_0, x_1, \ldots, x_{l-1}] \tag{1}$$

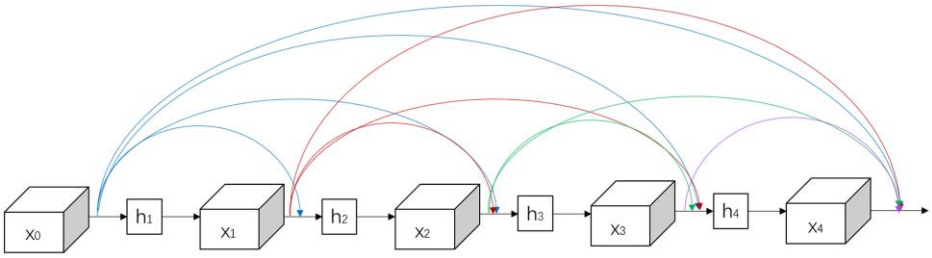


Figure 4. DenseNet feature extraction process.

This experiment proposes to add a skip connection to the p3 position of the backbone network, which can more effectively utilize the feature information to achieve feature reuse and reduce the loss of small target feature information in the transmission process. For this reason, this experiment is set up with two layers; too many connections can cause slow detection speed problems. The input is a 416 × 416 size image that passes through a downsampling convolution layer with a channel number of 32 × 32 and a 3 × 3 convolution kernel and passes to a downsampling convolution layer with a 64 × 64 feature map size and a 3 × 3 size convolution kernel. The convolutional layer of the backbone network

consists of Conv (convolution), BN (batch normalization), and the LeakyRelu activation function. P2 is followed by a convolutional downsampling, and a convolution is connected after p3 to connect it to p2, for which the skip connection is integrated to propagate the connected features forward again, greatly preserving the small target apple features in the process.

2.3.2. Fusion of Extremely Small Target Features in P2BiFPN

The conventional FPN [35], as in Figure 5a, possesses only a single top-down information flow, which causes the loss of feature information. PANet [36] is used in the YOLOv7-tiny network, as shown in Figure 5b, and is a bottom-up information enhancement based on the FPN, and the ability to acquire features still needs to be improved. In this study, the small target detection layer P2 is added to YOLOv7-tiny for stronger feature fusion, preserving the semantic information of small, shallow targets. It is achieved by removing the P5 detection head, fusing the original P5 feature layer with the P4, and finally, outputting the P4 detection head. Drawing on BiFPN [37], which is improved based on PANet, using the idea of bi-directional fusion is a simple and efficient feature fusion mechanism where each input feature has a different resolution, for which efficient feature fusion is performed. As shown in Figure 5, it can be seen that the concat connection of P3P4 in the backbone and P3P4 in the neck, respectively, top-down and bottom-up operations will achieve higher feature fusion. The detection accuracy was improved while the number of parameters was reduced, and this was fused to the whole neck, which we named P2BiFPN, as in Figure 5c. For the detection of small targets, especially small fruits with low resolution at a distance, they possess more reliable accuracy under weather interference.

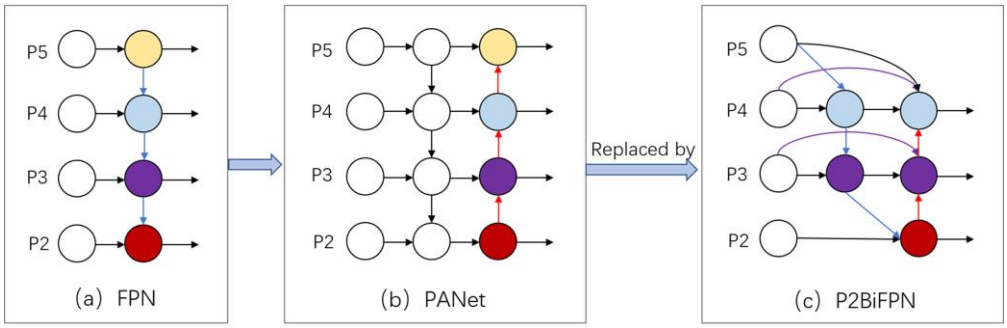


Figure 5. The improvement process of the neck network, with different colored circles representing elements of different sizes, (**a**) FPN, (**b**) PANet, and (**c**) P2BiFPN.

2.3.3. ULSAM Discards Superfluous Features

The detection and counting of small target apples are inaccurate in the case of complex backgrounds and weather changes, especially since GreenApple has great similarity to the background. To solve these problems and discard useless features, we propose adding a ULSAM (Ultra-Lightweight Subspace Attention Mechanism) to the neck [38], which can learn each attention mapping in each feature subspace. As shown in Figure 6, m is the number of input channels, G indicates that each group has G feature maps, h, w is the spatial dimension of the feature maps, forming a set of intermediate feature maps, using the linear relationship between different feature subspaces to integrate the channel information and have a more efficient effect on the learning of features, and do not burden the network. In this experiment, the ULSAM module is added to PANet with the parameters set to a 128 × 128 feature map size, a 26 × 26 picture size, and a default g = 4 with connection downsampling. It is well embedded into the network, increasing the detection accuracy while the model size does not change, satisfying the detection model of the lightweight network.

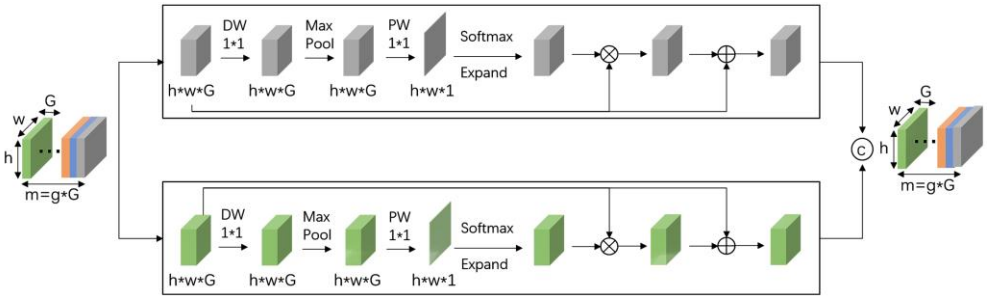


Figure 6. ULSAM structure diagram.

2.4. Experiment Environments

In this experiment, a desktop computer is used, and the running environment is the Pytorch deep learning framework built with the Ubuntu 18.04.6LTS system, configured with 11th Gen Intel(R) Core (TM) i7-11700k@3.60GHz*16, GPU is NVIDIA GeForce GTX 1080Ti/PCIe/SSE2, using CUDA10.2, OpenCV, Cudnn, and other related libraries to implement the detection model of small target applies in the context of complex environments.

2.5. Evaluation Indicators

In this experiment, the average metrics of three types of apples were selected and validated on a randomly divided test set. The mean F1 Score (mF1), Precision, Recall, and mean Average Precision(mAP@0.5) evaluation metrics were used to test the performance of the network, and N represents the number of categories. The evaluation metric is used to measure the performance of the whole model [39], calculated as:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \times 100\% \tag{2}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \times 100\% \tag{3}$$

$$\text{F1} = \left(\frac{1}{\text{N}}\sum 2\frac{\text{R} \times \text{P}}{\text{R} + \text{P}}\right)^2 \tag{4}$$

$$\text{AP} = \int_0^1 \text{P}(\text{R})\text{dR} \times 100\% \tag{5}$$

$$\text{mAP} = \frac{\sum_1^{\text{N}} \int_0^1 \text{P}(\text{R})\text{dR}}{\text{N}} \times 100\% \tag{6}$$

where TP represents the number of true positive correctly identified samples, FP represents the number of false positive misidentified negative samples, and FN represents the number of false negative missed positive samples.

In evaluating apple counting, RMSE (Root Mean Square Error) [40] and MAE (Mean Absolute Error) [40] are utilized to measure the effectiveness of counting. K is the number of images, pi is the true number of apple labels, and qi is the number of detected apples, calculated as:

$$\text{RMSE} = \sqrt{\frac{1}{\text{K}}\sum_{\text{i}=1}^{\text{K}}(\text{p}_\text{i} - \text{q}_\text{i})^2} \tag{7}$$

$$\text{MAE} = \frac{1}{\text{K}}\sum_{\text{i}=1}^{\text{K}}|\text{p}_\text{i} - \text{q}_\text{i}| \tag{8}$$

3. Results

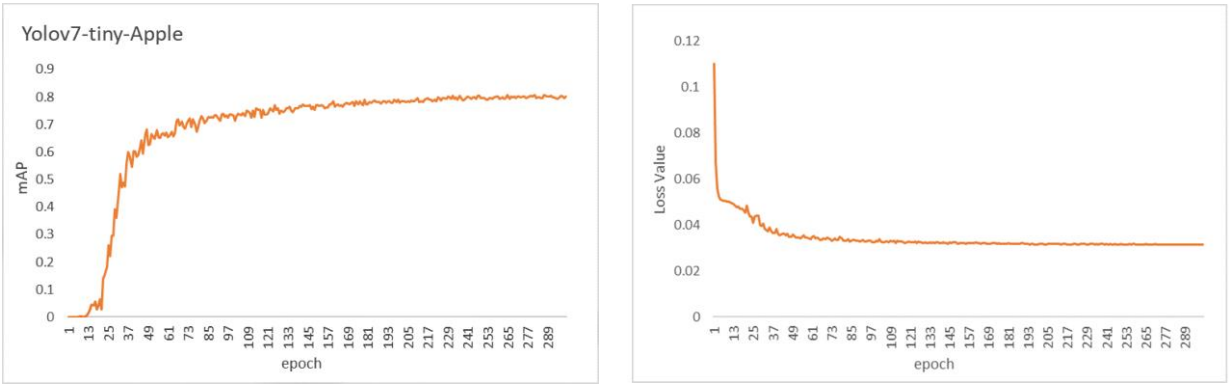
3.1. Training Process

In this study, the same initial training parameters are set for each group of experiments. We input an image size of 416 × 416, an epoch value of 300, a learning rate of 0.01, a momentum of 0.937, and a weight_decay of 0.0005, and the optimizer chosen is Adam, as shown in Table 3. The data is recorded using Tensorboard during the training process, and the training set loss is written for each iteration; the validation set loss is written for each training round, and the model weights are saved.

Table 3. All hyperparameters and values.

Parameter	Value
Input size pixels	416 × 416
Training epochs	300
Learning rate	0.01
Momentum	0.937
Weight decay	0.0005
Optimizer	Adam

As shown in Figure 7, the model's detection accuracy values and training loss values vary with the number of iterations during the training process. Gradually converging from 50 rounds onwards, the detection accuracy gradually increases, and the loss of the model gradually decreases, and finally, the 250 rounds level off, and the accuracy and loss values no longer change. It can be seen that the YOLOv7-tiny-Apple model does not suffer from over- and under-fitting and gradient disappearance [41] and can be applied to the detection and counting of small target fruits.



(**a**) (**b**)

Figure 7. Loss and mAP with iteration. (**a**) mAP curve chart; (**b**) Loss Value curve chart.

3.2. Ablation Experiments

In this experiment, ablation experiments verify each improvement point's effectiveness. We propose a total of three innovation points, and the results of the ablation experiment are shown in Table 4.

After improving the first skip connection j, it is clear that the network adding the skip connection improves mAP by 3.6%, mF1, Precision, and Recall by 3.1%, 2.6%, and 3.5%, respectively, on the original base network, and the number of parameters of the model increases by 0.16 M. The second improvement point, P2BiFPN, improves mAP by 3.8%, mF1, Precision, and Recall by 3.5%, 1.8%, and 4.8%, respectively, compared with the base network, and the amount of parameters of the model is reduced by 1.11 MB, which greatly reduces the complexity of the model. The third improvement point is that the ULSAM attention mechanism increases the detection accuracy while the number of parameters

of the model does not change, satisfying the detection model of the lightweight network. Compared with the base network, mAP improved by 2.9%, mF1, Precision, and Recall improved by 2.3%, 1.4%, and 3.3%, respectively; the number of parameters of the model did not change, while FPS was reduced but could meet the fast detection of the model.

Table 4. Comparison between the improvement points of the ablation experiment.

j	ULSAM	P2BiFPN	F1 Score/%	Precision/%	Recall/%	mAP0.5/%	Params/MB	FPS
			72.3	77.2	68.1	74.9	6.01	158.73
√			75.4	79.8	71.6	78.5	6.17	156.25
	√		74.6	78.6	71.4	77.8	6.01	133.33
		√	75.8	79.0	72.9	78.7	4.90	123.45
√	√		76.0	82.7	70.5	79.1	6.17	125
	√	√	76.6	79.5	74.0	79.8	4.90	104.16
√		√	75.8	81.2	71.4	79.7	5.06	111.11
√	√	√	76.8	80.1	74.1	80.4	5.06	101.01

All three improvement points significantly enhanced the accuracy of the base model in this experiment. We systematically combined them in pairs, as shown in Figure 8, demonstrating the feasibility of the model's detection accuracy after each combination. Finally, by integrating all three improvements, we achieved an mAP (mean Average Precision) value of 80.4%, Precision, Recall, and mF1 values of 80.1%, 74.1%, and 76.8%, respectively. Moreover, the resulting model has a compact size of 5.06 MB. This shows that the improvement of this experiment is more effective in feature extraction in the complex background of the natural environment, largely discarding the redundant interference and having superior detection performance.

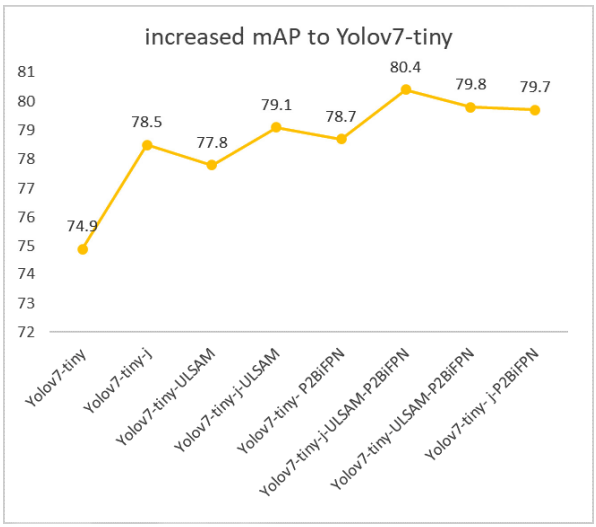


Figure 8. mAP line graph of ablation experiment.

For counting, this study uses the images in the test set for testing, and the counting results are shown in Table 5. From the results, it can be seen that for the first innovation point j skip connection, RMSE is reduced by 0.281, and MAE is slightly higher by 0.006; for the second innovation point, P2BiFPN, RMSE is reduced by 0.285, and MAE is slightly higher by 0.091; for the third innovation point, the ULSAM attention mechanism, RMSE is reduced by 0.32, and MAE is reduced by 0.043. The counting metrics of this experiment, RMSE reduced by 0.416 and MAE reduced by 0.165, are also effective in terms of counting and have performance improvement.

Table 5. Count results of ablation experiments.

Model	MAE	RMSE
YOLOv7-tiny	2.902	4.636
YOLOv7-tiny-j	2.908	4.355
YOLOv7-tiny-ULSAM	2.859	4.316
YOLOv7-tiny-j-ULSAM	2.902	4.534
YOLOv7-tiny-P2BiFPN	2.993	4.351
YOLOv7-tiny-j-ULSAM- P2BiFPN	2.737	4.220
YOLOv7-tiny-ULSAM-P2BiFPN	2.889	4.312
YOLOv7-tiny- j-P2BiFPN	2.902	4.507

3.3. Comparison Experiments

The more classical and popular Faster Rcnn-VGG, SSD-MobileNetV2, CenterNet-ResNet50, the newer and prominent lightweight networks YOLOv5s [42], YOLOx-tiny [43], YOLOv6t [44], and YOLOv7 were selected for comparison with the seven models of YOLOv7-tiny-Apple in this study, and the detection results are shown in Table 6.

Table 6. Accuracy comparison between different models.

Model	F1 Score/%	Precision/%	Recall/%	mAP0.5/%	Params/MB	FPS
YOLOv7-tiny	72.3	77.2	68.1	74.9	6.01	126.56
Faster Rcnn-VGG	50.33	41.16	66.51	55.95	547.0	24.90
SSD-MobileNetV2	47.3	95.07	31.93	64.16	16.6	81.98
CenterNet-ResNet50	48.3	95.48	40.56	71.22	131.0	69.74
YOLOv7	74.2	80.5	69.0	77.8	36.49	55.55
YOLOv5s	75.42	81.3	70.5	78.4	7.02	178.57
YOLOx-tiny	70.6	77.2	65.2	72.14	5.06	84.83
YOLOv6t	75.1	79.7	71.0	76.4	9.67	105.15
YOLOv7-tiny-Apple	76.8	80.1	74.1	80.4	5.06	101.01

By comparing the experimental tests, it was found that the mAP of YOLOv7-tiny-Apple was the highest among the models, reaching 80.4%, which was 2% higher than that of YOLOv5, the best-performing model, ensuring the detection accuracy, while the number of parameters was also the smallest. Compared with Recall, the model sensitivity of YOLOv7-tiny-Apple is 6%, 5.1%, 3.6%, 8.9%, and 3.1% higher than the other models, respectively. When comparing Precision, SSD-MobileNetV2, and CenterNet-ResNet50 stand out more but have no advantage in the remaining areas. The value of mF1 was compared with several models and found that the value of this model was the highest and had better detection accuracy for detecting different classes of fruits. Relative to the FPS, the FPS of YOLOv7-tiny-Apple was 101.01Hz, which was higher than Faster Rcnn-VGG, SSD-MobileNetV2, CenterNet-ResNet50, YOLOv7, and YOLOx-tiny and lower than the other three models, but it is sufficient for fast detection of fruits. This shows the superiority of the present model compared to other models, and compared with other current YOLO small model series, we can see that the present model has higher detection accuracy while the model is also the smallest, which is sufficient to achieve lightweight, small target detection in the complex background of a natural environment, illustrating the superiority of the present experimental model.

The test set photos were also used to detect the counting images from the comparison trials. To examine the efficacy of this experimental approach, Figure 9 displays the MAE and RMSE counting impacts of the evaluation model. It is evident that the YOLOv7-tiny-Apple model's MAE and RMSE, which are 2.737 and 4.220, respectively, better than the counting effects of other models and are the best counting effects. As shown in Table 7.

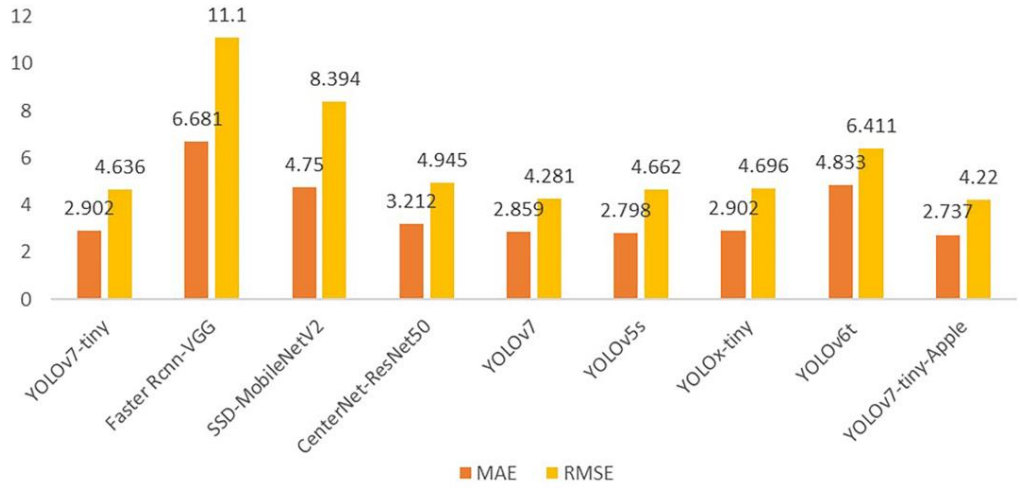


Figure 9. Histogram of counting effect between different models.

Table 7. Comparison of experimental count results.

Model	MAE	RMSE
YOLOv7-tiny	2.902	4.636
Faster Rcnn-VGG	6.681	11.10
SSD-MobileNetV2	4.750	8.394
CenterNet-ResNet50	3.212	4.945
YOLOv7	2.859	4.281
YOLOv5s	2.798	4.662
YOLOx-tiny	2.902	4.696
YOLOv6t	4.833	6.411
YOLOv7-tiny-Apple	2.737	4.220

4. Discussion

Based on the aforementioned discussion, it is evident that the detection model employed in this experiment exhibits high accuracy. To assess the model's robustness in detecting apples within complex environments, we conducted a comparative visual analysis. The analysis distinguished between correctly detected boxes and missed detection cases, represented by oval white boxes. Common challenges observed in both cases include heavy leaf shading and loss of fruit outline. By examining these factors, we were able to evaluate the effectiveness of fruit detection in three distinct scenarios.

4.1. Fruits in Dim or Backlit Conditions

As shown in Figure 10, the comparison of fruit detection results in dim or backlight conditions shows that YOLOv7-tiny-Apple can meet the extremely difficult detection of small fruits and fruit overlap in dim conditions. For the four fruits in the lower right corner in the extreme dimness, their fruits were missed by (h) Faster Rcnn-VGG, (i) SSD-MobileNetV2 and two fruits were missed by (b) YOLOv7-tiny (c), YOLOv7 (d), YOLOv5s, (g) CenterNet-ResNet50, and one fruit was missed by (e) YOLOv6t, (f) YOLOx-tiny. There are two overlapping heavily occluded fruits in the upper left corner, (b) YOLOv7-tiny, (c) YOLOv7, (g) CenterNet-ResNet50, (h) Faster Rcnn-VGG did not detect them, (d) YOLOv5s, (e) YOLOv6t, (f) YOLOx-tiny, and (i) SSD-MobileNetV2 detected one fruit and the other heavily occluded fruit was not detected. At the top, there are two dim fruits, and (h) Faster Rcnn-VGG misses them all, while (b) YOLOv7-tiny, (c) YOLOv7, and (g) CenterNet-ResNet50 all omit one. This model, (a) YOLOv7-tiny-Apple, can detect all fruits, which has strong feature extraction and high localization accuracy.

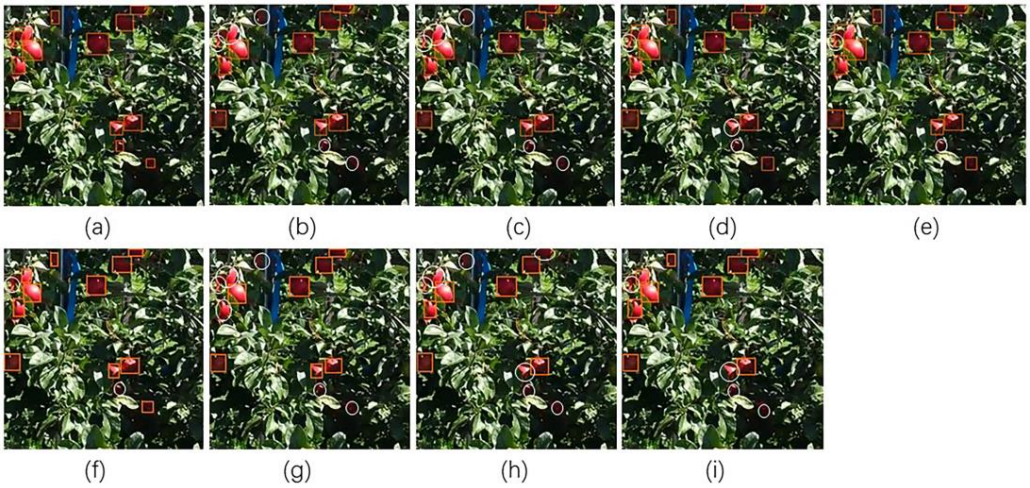


Figure 10. Detection effect of different models in dim or backlit scenes. (**a**) YOLOv7-tiny-Apple; (**b**) YOLOv7-tiny; (**c**) YOLOv7; (**d**) YOLOv5s; (**e**)YOLOv6t; (**f**) YOLOx-tiny; (**g**) CenterNet-ResNet50; (**h**) Faster Rcnn-VGG; (**i**) SSD-MobileNetV2.

4.2. Fruits under Low Light and Blurred Outline of Small Fruits in the Distance

As shown in Figure 11, YOLOv7-tiny-Apple targets fruits that are more difficult to capture with low natural light and insufficient brightness for detection. It is found that the blurring of the distant small fruit outline deepens the detection difficulty even more in the case of weak light. One to two fruit misses occur at the top of the picture, as shown in (b) YOLOv7-tiny, (c) YOLOv7, (d) YOLOv5s, (e) YOLOv6t, (f) YOLOx-tiny, (g) CenterNet-ResNet50, (h) Faster Rcnn-VGG, and (i) SSD-MobileNetV2. The loss and low resolution of the rightmost fruit outline were not detected by any of the other models, except for (a) YOLOv7-tiny-Apple, which detected it accurately. In particular, (h) Faster Rcnn-VGG, (i) SSD-MobileNetV2 miss significantly. It is clear that YOLOv7-tiny-Apple possesses higher detection performance and significantly overcomes the issue of inaccurate detection of far-off small fruits.

Figure 11. Detection effects of different models under low natural light and low brightness. (**a**) YOLOv7-tiny-Apple; (**b**) YOLOv7-tiny; (**c**) YOLOv7; (**d**) YOLOv5s; (**e**) YOLOv6t; (**f**) YOLOx-tiny; (**g**) CenterNet-ResNet50; (**h**) Faster Rcnn-VGG; (**i**) SSD-MobileNetV2.

4.3. Fruits in the Case of Bright or Overlapping Shadows and High Background Similarity

As shown in Figure 12, under the bright and overlapping shadows, there is an extreme similarity between GreenApple and the background. The fruit in the lower left corner is shaded with higher similarity to the leaves as shown in Figure (b) YOLOv7-tiny,

(e) YOLOv6t, (f) YOLOx-tiny, (g) CenterNet-ResNet50, and (h) Faster Rcnn-VGG is difficult to detect. At the same time, the contours are missing more severely, and the whole leaf covering the fruit is difficult to be detected accurately. As shown in the upper right corner of (b) YOLOv7-tiny, (c) YOLOv7, (d) YOLOv5s, (e) YOLOv6t, (f) YOLOx-tiny, (h) Faster Rcnn-VGG, (i) SSD-MobileNetV2 can be seen; all missed this fruit. For the rightmost fruit, there is also its omission. However, (a) YOLOv7-tiny-Apple can accurately detect and locate it and better extract the features of the green fruit, which greatly solves the background similarity. The problem of background similarity is considered solved, which reflects the superiority of this model.

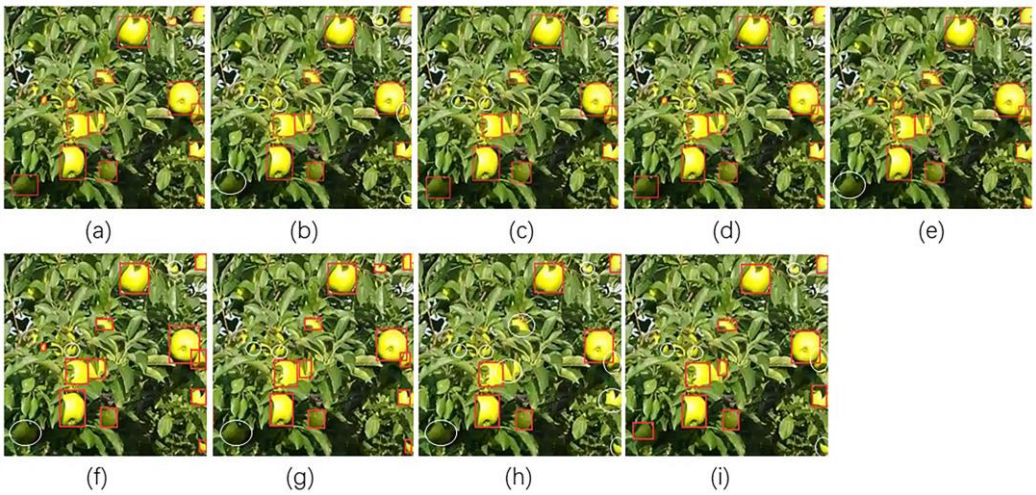


Figure 12. Detection effects of different models in bright and shadow overlapping scenes with extreme similarity to the background. (**a**) YOLOv7-tiny-Apple; (**b**) YOLOv7-tiny; (**c**) YOLOv7; (**d**) YOLOv5s; (**e**) YOLOv6t; (**f**) YOLOx-tiny; (**g**) CenterNet-ResNet50; (**h**) Faster Rcnn-VGG; (**i**) SSD-MobileNetV2.

4.4. Extreme Detection in Variable Weather

It can be seen from the above that the experiment possesses good accuracy and adaptability for the detection of apples in complex environments. To further verify YOLOv7-tiny-Apple's robustness to complex weather, we selected representative distant green fruits in rainy and foggy weather scenarios for testing. The small green apples in the distance are more difficult to detect in complex weather due to the large background similarity, high degree of leaf occlusion, and low pixels, so we verify this. As can be seen from Figure 13, YOLOv7-tiny-Apple can effectively detect most of the distant small green apples on rainy and foggy days, and only some small green apples and the part of the target feature that is not oriented are not detected. In summary, under the interference of complex weather, our detection model is still effective, with strong positioning accuracy and superior robustness.

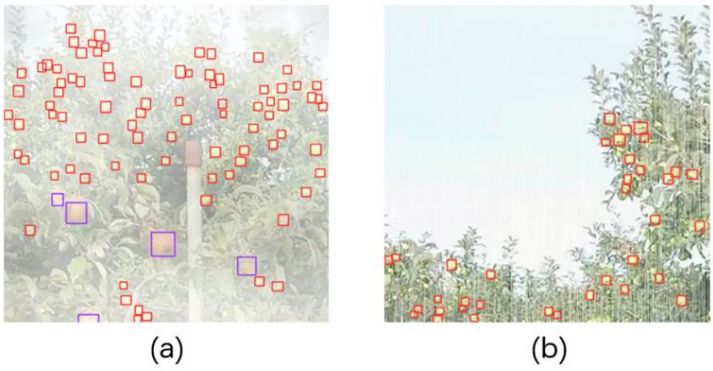


Figure 13. YOLOv7-tiny-Apple detection for rain and fog scenarios. (**a**) Foggy weather; (**b**) rainy weather.

5. Conclusions

A lightweight small-target apple detection and counting model YOLOv7-tiny-Apple, is proposed based on YOLOv7-tiny, which provides theoretical support for designing apple detection and counting models. This model can cope with the detection of small target apples under the complex weather of the natural environment, further promoting the picking and estimation of production in automated orchards, and can also provide some technical support in the deployment of equipment. Excellent potential for real-time application in orchard management. This study has a certain encouraging influence on increasing labor cost effectiveness, increasing production quality, and enhancing agricultural production efficiency. It aids in real-time apple identification and more effective orchard management by automatically identifying and counting apples. Additionally, by efficiently decreasing the use of computational resources and time and increasing the efficiency of practical application, this work plays a constructive role in promoting and using object detection technology, which has some practical implications.

(1) This model uses a skip connection integrated into the backbone network of YOLOv7-tiny to fuse the shallow features, strengthen the small target features, solve the problem of missing part of small target features, and greatly preserve the apple features;
(2) Due to the complex environment, the distant small fruit contours in the image are seriously missing, which affects the detection results. For this reason, we propose P2BiFPN added to the neck, which carries out multi-scale fusion and feature reuse, which not only improves the accuracy but also reduces the volume of the model to a certain extent;
(3) To further optimize the model, we add the ULSAM completely lightweight attention mechanism, which resolves the issue of identifying extreme similarity between the target and the background while ensuring the accuracy of the model;
(4) The majority of current research relies on the identification of complex background fruits in normal weather, which is inflexible to the effects of changing weather. In this experiment, the used dataset was processed using image enhancement techniques, and the rain and fog scenarios were simulated to verify the detection ability of the model in different weather with good detection performance.

Compared with other models, the improved model has better generalization and robustness and can be adapted to the task of small target apple detection in natural environments with complex backgrounds of variable weather. Possible techniques for using the model could also include aspects such as efficient inference techniques and algorithm optimization, which are used to improve the detection accuracy and efficiency of the model for different applications. Future work will concentrate on lightweight optimization, higher detection accuracy, and mobile device deployment to further enhance technical monitoring and control of smart orchards.

Author Contributions: The contributors are G.C. and L.Z. for conceptualization; Z.W. for methodology; J.Z. for formal analysis; Z.W. for data curation; L.M. and L.Z. for investigation/writing—original draft/supervision; L.M. and G.C. for visualization; J.Z. for writing—review/editing. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China–Joint Fund (u19a2061), Strategic Research and Consulting Project of Chinese Academy of Engineering (No. JL2023-03), Jilin Provincial Department of Education Project (No. JJKH20210336KJ), Jilin province science and technology development plan project (No. 20210204050YY), Jilin University Student Innovation Training Program (S202210193103).

Data Availability Statement: Data supporting the findings of this study are available from the corresponding author.

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

References

1. Lu, S.; Chen, W.; Zhang, X.; Karkee, M. Canopy-Attention-YOLOv4-Based Immature/Mature Apple Fruit Detection on Dense-Foliage Tree Architectures for Early Crop Load Estimation. *Comput. Electron. Agric.* **2022**, *193*, 106696. [CrossRef]
2. Koutsos, A.; Tuohy, K.M.; Lovegrove, J.A. Apples and Cardiovascular Health—Is the Gut Microbiota a Core Consideration? *Nutrients* **2015**, *7*, 3959–3998. [CrossRef] [PubMed]
3. Blažek, J.; Paprštein, F.; Zelený, L.; Křelinová, J. The Results of Consumer Preference Testing of Popular Apple Cultivars at the End of the Storage Season. *Hortic. Sci.* **2019**, *46*, 115–122. [CrossRef]
4. Tian, Y.; Yang, G.; Wang, Z.; Wang, H.; Li, E.; Liang, Z. Apple Detection during Different Growth Stages in Orchards Using the Improved YOLO-V3 Model. *Comput. Electron. Agric.* **2019**, *157*, 417–426. [CrossRef]
5. Zhao, C. Current situations and prospects of smart agriculture. *J. South China Agric. Univ.* **2021**, *42*, 1–7.
6. Bargoti, S.; Underwood, J. Deep Fruit Detection in Orchards. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June May 2017; pp. 3626–3633.
7. Kang, H.; Chen, C. Fruit Detection and Segmentation for Apple Harvesting Using Visual Sensor in Orchards. *Sensors* **2019**, *19*, 4599. [CrossRef] [PubMed]
8. Sun, J.; He, X.; Ge, X.; Wu, X.; Shen, J.; Song, Y. Detection of Key Organs in Tomato Based on Deep Migration Learning in a Complex Background. *Agriculture* **2018**, *8*, 196. [CrossRef]
9. Jia, W.; Wang, Z.; Zhang, Z.; Yang, X.; Hou, S.; Zheng, Y. A Fast and Efficient Green Apple Object Detection Model Based on Foveabox. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, 5156–5169. [CrossRef]
10. Jia, W.; Zhang, Y.; Lian, J.; Zheng, Y.; Zhao, D.; Li, C. Apple Harvesting Robot under Information Technology: A Review. *Int. J. Adv. Robot. Syst.* **2020**, *17*, 1–16. [CrossRef]
11. Tian, Y.; Duan, H.; Luo, R.; Zhang, Y.; Jia, W.; Lian, J.; Zheng, Y.; Ruan, C.; Li, C. Fast Recognition and Location of Target Fruit Based on Depth Information. *IEEE Access* **2019**, *7*, 170553–170563. [CrossRef]
12. Lin, G.; Tang, Y.; Zou, X.; Li, J.; Xiong, J. In-Field Citrus Detection and Localisation Based on RGB-D Image Analysis. *Biosyst. Eng.* **2019**, *186*, 34–44. [CrossRef]
13. Wang, C.; Lee, W.S.; Zou, X.; Choi, D.; Gan, H.; Diamond, J. Detection and Counting of Immature Green Citrus Fruit Based on the Local Binary Patterns (LBP) Feature Using Illumination-Normalized Images. *Precis. Agric.* **2018**, *19*, 1062–1083. [CrossRef]
14. Wang, D.; Song, H.; Tie, Z.; Zhang, W.; He, D. Recognition and Localization of Occluded Apples Using K-Means Clustering Algorithm and Convex Hull Theory: A Comparison. *Multimed. Tools Appl.* **2016**, *75*, 3177–3198. [CrossRef]
15. Zhang, C.; Liu, X.; Chen, B.; Yin, P.; Li, J.; Li, Y.; Meng, X. Insulator Profile Detection of Transmission Line Based on Traditional Edge Detection Algorithm. In Proceedings of the 2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), Dalian, China, 27–29 June 2020; pp. 267–269.
16. Li, X.; Pan, J.; Xie, F.; Zeng, J.; Li, Q.; Huang, X.; Liu, D.; Wang, X. Fast and Accurate Green Pepper Detection in Complex Backgrounds via an Improved Yolov4-Tiny Model. *Comput. Electron. Agric.* **2021**, *191*, 106503. [CrossRef]
17. Zhang, C.; Kang, F.; Wang, Y. An Improved Apple Object Detection Method Based on Lightweight YOLOv4 in Complex Backgrounds. *Remote Sens.* **2022**, *14*, 4150. [CrossRef]
18. Tu, S.; Pang, J.; Liu, H.; Zhuang, N.; Chen, Y.; Zheng, C.; Wan, H.; Xue, Y. Passion Fruit Detection and Counting Based on Multiple Scale Faster R-CNN Using RGB-D Images. *Precis. Agric.* **2020**, *21*, 1072–1091. [CrossRef]
19. Bhattarai, U.; Karkee, M. A Weakly-Supervised Approach for Flower/Fruit Counting in Apple Orchards. *Comput. Ind.* **2022**, *138*, 103635. [CrossRef]
20. Hao, Q.; Guo, X.; Yang, F. Fast Recognition Method for Multiple Apple Targets in Complex Occlusion Environment Based on Improved YOLOv5. *J. Sens.* **2023**, *2023*, e3609541. [CrossRef]
21. Weyler, J.; Milioto, A.; Falck, T.; Behley, J.; Stachniss, C. Joint Plant Instance Detection and Leaf Count Estimation for In-Field Plant Phenotyping. *IEEE Robot. Autom. Lett.* **2021**, *6*, 3599–3606. [CrossRef]
22. Chen, J.; Liu, H.; Zhang, Y.; Zhang, D.; Ouyang, H.; Chen, X. A Multiscale Lightweight and Efficient Model Based on YOLOv7: Applied to Citrus Orchard. *Plants* **2022**, *11*, 3260. [CrossRef]
23. Sun, M.; Xu, L.; Luo, R.; Lu, Y.; Jia, W. GHFormer-Net: Towards More Accurate Small Green Apple/Begonia Fruit Detection in the Nighttime. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, 4421–4432. [CrossRef]
24. Gulzar, Y. Fruit Image Classification Model Based on MobileNetV2 with Deep Transfer Learning Technique. *Sustainability* **2023**, *15*, 1906. [CrossRef]
25. Mamat, N.; Othman, M.F.; Abdulghafor, R.; Alwan, A.A.; Gulzar, Y. Enhancing Image Annotation Technique of Fruit Classification Using a Deep Learning Approach. *Sustainability* **2023**, *15*, 901. [CrossRef]
26. Hamid, Y.; Wani, S.; Soomro, A.B.; Alwan, A.A.; Gulzar, Y. Smart Seed Classification System Based on MobileNetV2 Architecture. In Proceedings of the 2022 2nd International Conference on Computing and Information Technology (ICCIT), Tabuk, Saudi Arabia, 25–27 January 2022; pp. 217–222.
27. Aggarwal, S.; Gupta, S.; Gupta, D.; Gulzar, Y.; Juneja, S.; Alwan, A.A.; Nauman, A. An Artificial Intelligence-Based Stacked Ensemble Approach for Prediction of Protein Subcellular Localization in Confocal Microscopy Images. *Sustainability* **2023**, *15*, 1695. [CrossRef]
28. Häni, N.; Roy, P.; Isler, V. MinneApple: A Benchmark Dataset for Apple Detection and Segmentation. *IEEE Robot. Autom. Lett.* **2020**, *5*, 852–858. [CrossRef]

29. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
30. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In *Computer Vision—ECCV 2016*; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Springer International Publishing: Cham, Switzerland, 2016; pp. 21–37.
31. Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; Tian, Q. CenterNet: Keypoint Triplets for Object Detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6569–6578.
32. Wang, C.-Y.; Bochkovskiy, A.; Liao, H.-Y.M. YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors 2022. *arXiv* **2022**, arXiv:2207.02696.
33. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely Connected Convolutional Networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2261–2269.
34. Xu, D.; Wu, Y. Improved YOLO-V3 with DenseNet for Multi-Scale Remote Sensing Target Detection. *Sensors* **2020**, *20*, 4276. [CrossRef] [PubMed]
35. Ghiasi, G.; Lin, T.-Y.; Le, Q.V. NAS-FPN: Learning Scalable Feature Pyramid Architecture for Object Detection. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 7029–7038.
36. Wang, K.; Liew, J.H.; Zou, Y.; Zhou, D.; Feng, J. PANet: Few-Shot Image Semantic Segmentation with Prototype Alignment. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 9196–9205.
37. Chen, J.; Mai, H.; Luo, L.; Chen, X.; Wu, K. Effective Feature Fusion Network in BIFPN for Small Object Detection. In Proceedings of the 2021 IEEE International Conference on Image Processing (ICIP), Anchorage, AK, USA, 19–22 September 2021; pp. 699–703.
38. Saini, R.; Jha, N.K.; Das, B.; Mittal, S.; Mohan, C.K. ULSAM: Ultra-Lightweight Subspace Attention Module for Compact Convolutional Neural Networks. In Proceedings of the 2020 IEEE Winter Conference on Applications of Computer Vision (WACV), Snowmass Village, CO, USA, 1–5 March 2020; pp. 1616–1625.
39. Guo, S.; Li, L.; Guo, T.; Cao, Y.; Li, Y. Research on Mask-Wearing Detection Algorithm Based on Improved YOLOv5. *Sensors* **2022**, *22*, 4933. [CrossRef]
40. Hodson, T.O. Root-Mean-Square Error (RMSE) or Mean Absolute Error (MAE): When to Use Them or Not. *Geosci. Model Dev.* **2022**, *15*, 5481–5487. [CrossRef]
41. Yang, B.; Gao, Z.; Gao, Y.; Zhu, Y. Rapid Detection and Counting of Wheat Ears in the Field Using YOLOv4 with Attention Module. *Agronomy* **2021**, *11*, 1202. [CrossRef]
42. Zhang, P.; Li, D. EPSA-YOLO-V5s: A Novel Method for Detecting the Survival Rate of Rapeseed in a Plant Factory Based on Multiple Guarantee Mechanisms. *Comput. Electron. Agric.* **2022**, *193*, 106714. [CrossRef]
43. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. YOLOX: Exceeding YOLO Series in 2021. *arXiv* **2021**, arXiv:2107.08430.
44. Li, C.; Li, L.; Jiang, H.; Weng, K.; Geng, Y.; Li, L.; Ke, Z.; Li, Q.; Cheng, M.; Nie, W.; et al. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications 2022. *arXiv* **2022**, arXiv:2209.02976.