Journal of Computers Vol. 33 No. 4, August 2022, pp. 45-56
doi: 10.53106/199115992022083304004

Method for Detection of Ripe Navel Orange Fruit on Trees in Various Weather

Qian-Li Zhang[1], Qiu-Sheng Li[1*], Jun-Yong Hu[1], Xiang-Hui Xie[2]

[1] School of Physics and Electronic Information, Gannan Normal University, Ganzhou, Jiangxi, 341000, China
zhangqianli@gnnu.edu.cn, bjliqiusheng@163.com, gnu_hjy@163.com
[2] Research Center of Intelligent Control Engineering Technology, Gannan Normal University, Ganzhou, Jiangxi, 341000, China
1249925393@qq.com

Received 27 August 2021; Revised 14 January 2022; Accepted 14 February 2022

Abstract. The algorithm used is based on the Faster R-CNN network. It uses resnet101_vd as the backbone network to extract navel orange image features. It minimizes the error between the inferred bounding box and the actual labeled bounding box through the RPN network and the ROI Pooling layer and non-maximum suppression method. The AP during model training reached 92.34% on sunny days, 96.84% on cloudy days, and 90.05% on foggy days. When evaluating the model, it reached 92.34% on sunny days, 96.89% on cloudy days, and 89.2% on foggy days. The processing time of this model is 63.84fps on sunny days, 68.6fps on cloudy days, and 56.29fps on foggy days. It meets the requirements of rapid and accurate identification in actual picking. In addition, it compares this model with the Faster RCNN with vgg16 as the backbone network and YOLO-v4 models. It effectively improves detection accuracy and speed. Moreover, it reduces the number of false detection and missed detection of navel orange detection. It improves position accuracy. This paper realizes the efficient detection of ripe navel orange fruits on trees in various weather.

Keywords: machine vision, target detection, comparative test, navel orange

1 Introduction

Navel oranges are grown in more than 100 countries and regions worldwide [1]. China is also a significant producer of navel oranges, especially in the southern Jiangxi region. Its unique soil conditions and climatic environment make Gannan navel oranges crispy and tender and well-known at home and abroad. Ganzhou has become the country's largest main producing area [2]. However, the labor intensity is high due to the concentrated maturity period of navel oranges, and the short picking cycle. Moreover, the harvesting efficiency is low, thereby increasing labor 0costs and detrimental to the economic benefits of fruit farmers. Therefore, it is of great significance to realize intelligent picking of navel oranges, improve the production efficiency of navel oranges, and study the automation of navel orange picking. One of the core problems of automated fruit and vegetable picking relies on the accurate detection and recognition of picking targets by picking robots. The recognition of the target fruit in the natural environment is affected by many factors, including different lighting conditions, different proportions of occlusion, and complex background environments.

For the identification of fruits and vegetables, related scholars have researched different directions. Mainly identify fruits and vegetables such as apples, grapes, tomatoes. At present, there are relatively little researches on the identification of navel oranges. Gherlone [3] based on the detection algorithm of network grey-scale features to identify green citrus fruits. Lu [4] proposed to use the contour features of the fruit surface to carry out circular light distribution, combined with the Hough transform for circle fitting. The algorithm has a recall rate of 81.2% for 20 citrus orchard scene images. After using local binary pattern (LBP) texture features, 25 images were tested. The accuracy rate reaches 82.3%, but the conditions are limited by the manual configuration of the light source during the image acquisition process. Liu [5] converted the RGB and HIS color channels and separated them by the Otsu algorithm to obtain the threshold to identify ripe tomato fruits effectively. Chu [6] distinguished the navel orange fruit from the background area through a segmentation algorithm based on color channels. After the edges are extracted, the corner points are eliminated, and the round fruits are detected by the least-squares method. The recognition rate is 95%, and it takes an average of 93.548ms to process a picture. It also points out that it has a certain versatility for other round fruits. Based on dense connectivity, Sun [7] and others have designed an FSD

* Corresponding Author

model of a small black spot small target detector that is densely distributed on the surface of the navel orange. It realized that the map reached 87.479% at a speed of 131fps. Moreover, there are fewer calculation parameters and a compact structure. Ling [8] used RGB images and Ada Boost classifier to detect mature tomatoes, with a successful recognition rate of 95%. Li [9] performed three-dimensional clustering based on two-dimensional image extraction depth information and optimized SOM and K-means algorithms. Recognizing the occluded and stuck tomato images, the correct recognition rate was 87.2%. Li [10] and others detected green apples in natural environments based on the YOLOv3 model. The component information of HSV and YUV color space is extracted and compared, and the background interference is removed. The average valid positive rate of the best recognition effect is 93.93%. The theory of machine learning algorithms developed and matured in the 1990s. It has been successfully applied in many fields. Nevertheless, deep learning models have emerged with the emergence of big data and the increase in computer computing power. It considerably changed the application pattern of machine learning, especially in the field of intelligent agriculture. It is mainly due to the excellent applicability of the neural network to learn sample data and the level of representation for image tasks. The effect of deep learning models is significantly improved over traditional machine learning algorithms. However, due to the application in different complex backgrounds, there are still false detection, missed detection, and inaccurate positions. The detection accuracy and speed need to be improved [11].

To sum up, there are various challenges faced by navel orange fruit identification. In this paper, an improved algorithm is designed to detect and identify ripe navel oranges on trees in sunny, cloudy and foggy weather. The main technical achievements of this paper are summarized as follows.

1) For the first time, images in the actual orchard scene were collected to create a navel orange fruit detection dataset.

2) Select the resnet_101_vd network as the backbone network to extract image features. A deeper network can extract stronger feature information, and add deformable convolution, which is more conducive to distinguish overlapping fruit targets.

3) Use k-means to cluster anchors that match the actual fruit size, which is convenient for extracting candidate frames.

4) Fusion of different feature maps through multi-scale features to share features.

5) The experimental results show that the model can improve the detection accuracy and speed up the training speed to a certain extent.

The main arrangement of this paper is as follows. Section 2 describes the construction and preprocessing of the dataset. Section 3 details the construction of the fruit detection model. Section 4 presents the training of the model and the results. Section 5 analyzes the experimental results. Section 6 presents the limitations of the work and directions for future research.

2 Related Work

2.1 Data Collection

The realization of the automatic detection and recognition model of navel orange relies on the enormous data set resources [12] as the essential support. These large data sets help the model training stage to learn and adjust all embedding parameters, minimizing the risk of network overfitting. Currently, there are no relevant resources for the public data set of navel orange images. Based on this experiment, we have prepared a navel orange dataset that includes sunny, cloudy, and foggy environments. All images were collected at the Golden Shield Orchard in Ganzhou, Jiangxi, in November and early December 2020. The pixel of the camera is about 20 million. To ensure that the influence of non-human subjective factors is excluded, the shooting direction and angle, the lighting environment, and the distance between the navel orange fruit and the equipment are all randomly selected. The foggy image is generated by adding noise to the cloudy image, but it is mutually exclusive with the cloudy image in the experiment. Finally, 445 sunny images, 427 cloudy images, and 434 foggy images were selected. Some image samples under various weather conditions are shown in Fig. 1.

First, performing simple data cleaning on the collected images. Including image deblurring (to filter low-definition images to ensure data quality) and image deduplication (to filter many duplicate images to improve the efficiency of necessary image processing). After the data cleaning, the data quality can be improved to facilitate the next data labeling operation.

(a) Sunny day (b) Cloudy day (c) Foggy day

Fig. 1. Image samples under various weather conditions

The labeling of target detection task data uses the LabelImg labeling tool. In order to reduce the impact of many problems in the original data labeling on the accuracy of the model, the navel oranges that are not obvious are not labeled, the navel oranges that appear as a point are not labeled, and the overlap is greater than 95% and labeled in the same box. The two navel oranges are marked with two boxes, which reduces the data review task. Furthermore, convert the marked data into PascalVOC format. For training, the data is divided into the training set and test set.

2.2 Data Set Division

Using the retention method, we randomly split the data set into a training set and a test set proportionally. The training set is used for iterative training to determine the model's parameters, and the test set is used to judge the effect of the model finally. In this experiment, 80% of the data is used as the training set and 20% as the test set. The data set includes 1726 images of four weather environments, and the input image size is 400*400. The data distribution is shown in Table 1.

Table 1. Data distribution table

	Training set	Test set	Total
Sunny	356	89	445
Cloudy	342	85	427
Foggy	347	87	434

2.3 Data Preprocessing

Data preprocessing is a crucial step when training neural networks. Appropriate preprocessing methods can help the model converge better and prevent overfitting. In order to ensure the speed of network operation, data preprocessing is usually accelerated.

Making some random changes to the image before training and perform data enhancement on the training set. Similar but not identical samples are generated through scaling, random rotation, random cropping, contrast adjustment, hue adjustment, and saturation adjustment operations on the existing original image. Increase the amount of training sample data, suppress over-fitting, and improve the model's generalization ability. After the data is preprocessed, the model is loaded again.

We normalize each feature so that the value of each feature is scaled between 0 and 1. Not only makes the model training more efficient, but because the range of each feature value itself is the same, the weight before the feature can also represent the contribution of the variable to the prediction result. We normalize the input pictures to ensure that the input information types are consistent.

3 The Model Design

In the target detection model, a series of anchor frames are usually generated on the picture according to specific rules, and these anchor frames are regarded as possible candidate regions. The model predicts whether these can-

didate regions contain the target, and if it contains the target, it needs to predict the target category further. Since the position of the anchor frame is fixed, it is unlikely to coincide with the target bounding box. Therefore, it is necessary to fine-tune the anchor frame to form a prediction frame that accurately describes the target position. The model needs to predict the magnitude of the fine-tuning. In the training process, the model learns to determine whether the candidate area represented by the anchor frame contains the target, the category of the target, and the extent to which the target bounding box needs to be adjusted relative to the position of the anchor frame through continuous learning and adjustment of parameters. In this paper, the Faster R-CNN network [13] called the two-stage target detection algorithm is used to detect the navel orange fruit as the target. The detection process is shown in Fig. 2.

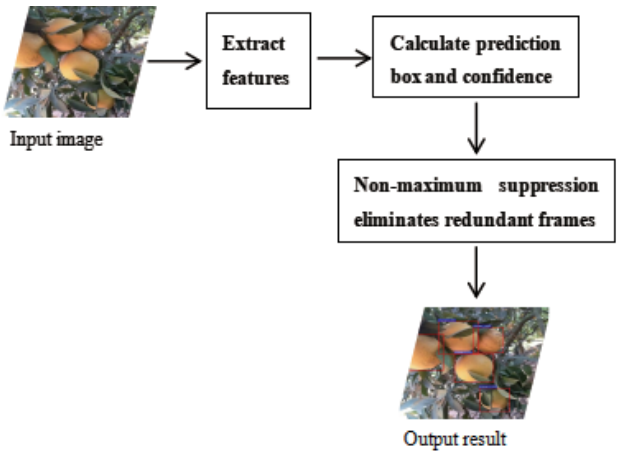


Fig. 2. Detection process

In the first stage, candidate regions are generated, using Anchor instead of selective search to select candidate regions, selecting the Anchor that contains the object, and entering the ROI Pooling to extract features. The second stage classifies the candidate area and predicts the location of the target object. The classification branch obtains the positive and negative samples of the Anchor classification, and the regression branch obtains the offset of the Anchor to the ground truth frame. The offset is learned from the actual frame to obtain the final prediction frame, as shown in Fig. 3.

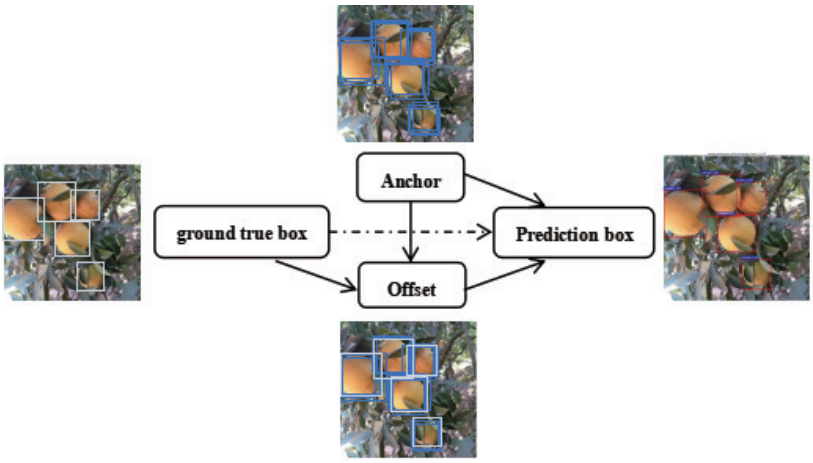


Fig. 3. Schematic diagram of a learning

3.1 Network Structure

The network is mainly composed of multiple convolutional layers (Conv), pooling layer (Pooling), and fully connected layer (FC). By using deep convolutional neural networks, feature maps with richer semantic meaning

can be obtained. We use ResNet101_vd as the backbone of the network to extract image features layer by layer. Due to the irregular occlusion of the navel orange growth shape [14], the position of the prediction frame is not accurate. The last convolution layer in the backbone network, ResNet101_vd, is selected to introduce deformable convolution. The final output feature map is used to characterize information such as object location and category. This not only improves accuracy but also significantly reduces trainable parameters. First, the candidate area needs to be gsignifican than aenerated through the RPN network, and then the target classification and coordinate position prediction are made for the RoI. Its network structure is shown in Fig. 4.

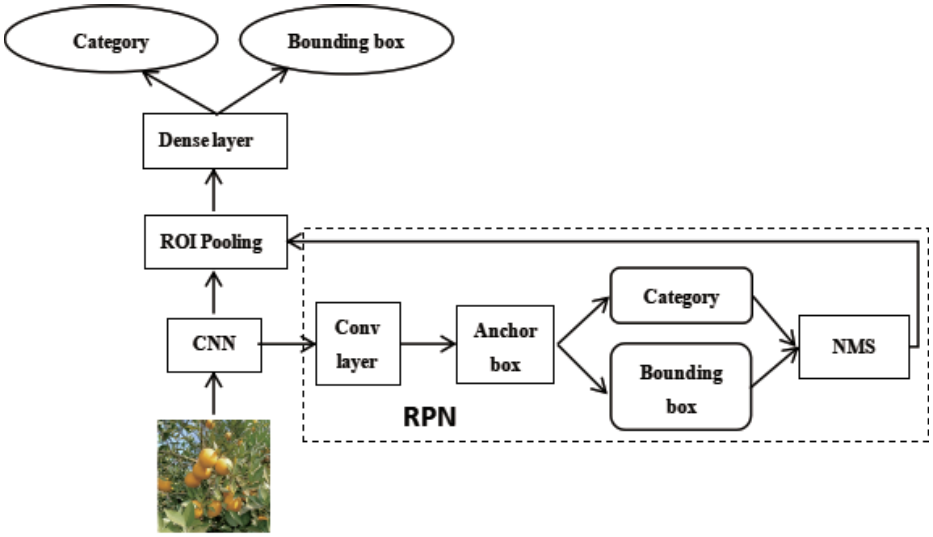


Fig. 4. Network structure

First, input the picture, and then perform deep feature extraction through the backbone network. Then, the candidate area is generated through the RPN network, and the classification of the candidate frame is completed (whether the navel orange is classified or the background environment), and the position of the navel orange target is roughly located. Secondly, the ROI Pooling layer performs accurate target classification and position correction of the candidate frame. CNN features are not calculated repeatedly for multiple feature maps. The core idea is that the candidate boxes share the feature map features and keep the output size consistent. Then, the fully connected layer is used to characterize the characteristics of the candidate area target corresponding to the feature map. Finally, the candidate navel orange target is judged through the classification and regression branch, and the bounding box position coordinates are corrected to obtain the target's actual category and precise position [15].

3.2 Loss Function

The loss function of the Faster R-CNN two-stage network is the sum of the classification loss and the position regression loss of the two networks [16]. The sum of the final loss value is the loss of the entire network. Then back-propagation is carried out, and various parameters of the network are continuously updated iteratively. The calculation formula is as follows.

$$L(\{p_i\},\{t_i\}) = \frac{1}{N_{cls}}\sum_i L_{cls}\left(p_i, p_i^*\right) + \ddot{e}\frac{1}{N_{reg}}\sum_i p_i^* L_{reg}\left(t_i, t_i^*\right) \quad . \tag{1}$$

Since the RPN network part performs a simple two-class classification for the target and the background, the target is one, and the background is 0. The number of anchors selected during training is N_{cls}. More anchors are added to the training to increase the proportion of complex samples in the loss function and reduce the false detection rate. The p_i represents the probability that it is predicted to be a navel orange target. $L_{cls}(p_i,p_i^*)$ represents the log loss of the navel orange target and the environmental background, and the calculation formula is as follows.

$$L_{cls}(P_i, P_i^*) = -\log[P_i^* P_i + (1-P_i^*)(1-P_i)] . \quad (2)$$

In the regression loss, λ refers to the weight of the regression loss, and N_{reg} refers to the number of anchors. The $p_i^* L_{reg}(t_i, t_i^*)$ means returning to the bounding box only when the sample is positive. The anchor with the largest overlap area with the IOU of the ground truth box or the overlap area is more significant than a predetermined threshold.

$$L_{\text{reg}}(t_i, t_i^*) = R(t_i - t_i^*) . \quad (3)$$

R refers to the piecewise function Smooth L1 function, making it smoother and more derivable near the 0 points, and Momentum optimisation not affecting the convergence. Just set the parameter rcnn_bbox_loss when defining the model Faster R-CNN class.

$$Smooth_{L1}(x) = \begin{cases} 0.5x^2 \times 1/\sigma^2, |x| < 1 \\ |x| - (0.5/\sigma^2), \text{otherwise} \end{cases} . \quad (4)$$

This t_i represents the coordinate vector of the predicted frame, and t_i^* represents the coordinate vector of the actual frame. The calculation formula is as follows.

$$\begin{cases} t_x = (x - x_a)/w_a, t_y = (y - y_a)/h_a \\ t_w = \log(w/w_a), t_h = \log(h/h_a) \\ t_x^* = (x^* - x_a)/w_a, t_y^* = (y^* - y_a)/h_a \\ t_w^* = \log(w^*/w_a), t_h^* = \log(h^*/h_a) \end{cases} . \quad (5)$$

These x, y, w, h represent the coordinates, width, and height of the center point of the predicted box of the RPN network. Furthermore, x_a, y_a, w_a, h_a correspond to the coordinates, width, and height of the center point of the anchor box candidate frame. Moreover, x^*, y^*, w^*, h^* are the coordinates, width, and height of the center point of the ground truth box target actual label box [17]. The final optimization goal of the loss function is to make the prediction box and the actual box overlap as much as possible.

3.3 Optimization Algorithms

In order to minimize the loss of the model, an optimization algorithm is required. The optimization algorithms of deep learning mainly include SGD, Momentum, RMSProp, Adam, and Nadam algorithms. This paper uses the Momentum optimization algorithm, and the calculation method is as follows [18].

$$\begin{cases} v_t = \tilde{a}v_{t-1} + \varsigma \nabla_{\grave{e}} J(\grave{e}) \\ \grave{e} = \grave{e} - v_t \end{cases} . \quad (6)$$

The parameter update consists of two parts. One is the gradient at the current moment, and a parameter change range needs to be added. The parameter update direction is the direction in which the two vectors are added. In updating the parameters, when the dimension of the direction at the gradient point is the same, the momentum term increases, and when the dimension of the direction at the gradient point is different, the momentum term decreases. Therefore, the convergence speed is accelerated.

4 Experiment

4.1 Network Configuration

The practice platform is Baidu's AI training platform AI Studio. The framework adopts the domestic mainstream deep learning framework PaddlePaddle produced by Baidu. The hardware environment includes CPU 4, RAM 32 GB, GPU v100, video memory 16 GB, and disk 100 GB. Environment configuration includes python 3.7, PaddlePaddle 1.8.4.

The total number of model training iterations num_epochs is set to a more considerable value. According to the index performance of the model iteration process on the verification set, judge whether the model converges and then terminate the training early. Batch Size is positively related to the video memory height of the machine. Since single card training is used, batch_size is set to 8, and learning_rate corresponds to 0.000125.

When training a model, using pre-trained model weights on a public data set can reduce unnecessary consumption. However, there is a big difference between the navel orange data set and the public data set during training. In order to avoid the initial gradient being too large, at the beginning of the training, with the training iteration, the learning rate is slowly increased to the set learning rate with a small value in the unit of step. When the model starts training, the learning rate will start from 0.0. After 1000 batch data iterations, it linearly increases to the set learning rate. This indicates that after the model starts training, within the first 1000 steps, the learning rate will linearly increase from 0.0 to the set 0.000125.

With the training iteration in the later model training stage, the learning rate gradually decays in the epoch. Each learning rate decays as the previous learning rate multiplied by lr_decay_gamma, and lr_decay_gamma is 0.1. num_epochs is 150, and learning_rate is 0.000125. lr_decay_epochs is [118, 133]. This indicates that after the model started training, in the first 118 epochs, the learning rate used during training was 0.000125. Between the 118th and 133rd epochs, the learning rate used for training is 0.000125x0.1=0.0000125. After 133 epochs, the learning rate used is $0.000125 \times 0.1 \times 0.1 = 0.00000125$.

4.2 Evaluation Index

The target detection model trained in this experiment uses AP, loss, P-R, curve, and detection speed to evaluate the model's performance. The AP calculation formula is as follows.

$$\begin{cases} P = \dfrac{TP}{TP + FN} \\ R = \dfrac{TP}{FP + FN} \\ R = \dfrac{TP}{FP + FN} \end{cases}. \tag{7}$$

P represents the detection accuracy rate, and R represents the recall rate. TP is the number of navel oranges that have been correctly identified as navel oranges. FP is the number of leaves, branches, sky, and other backgrounds incorrectly identified as navel oranges. FN is the number of navel oranges incorrectly identified as leaves, branches, sky, and other backgrounds. The area under the P-R curve is the AP value. The Loss function is used to evaluate the gap between the model's predicted value and the actual value. The smaller the loss value, the better the model performance.

4.3 Model Training

After the model structure is defined, the steps for training are as follows. First, the network forward propagation calculates the network output and loss function. Then carry out the backward error propagation according to the loss function, pass the network error forward from the output layer, and update the parameters in the network. Finally, it is repeated until the network training error reaches a prescribed level or the training round reaches the set value.

Network-based deep migration learning reuses pre-trained network structure and connection parameters and migrates to the task of navel orange detection. Since the neural network has iterative and continuous abstraction

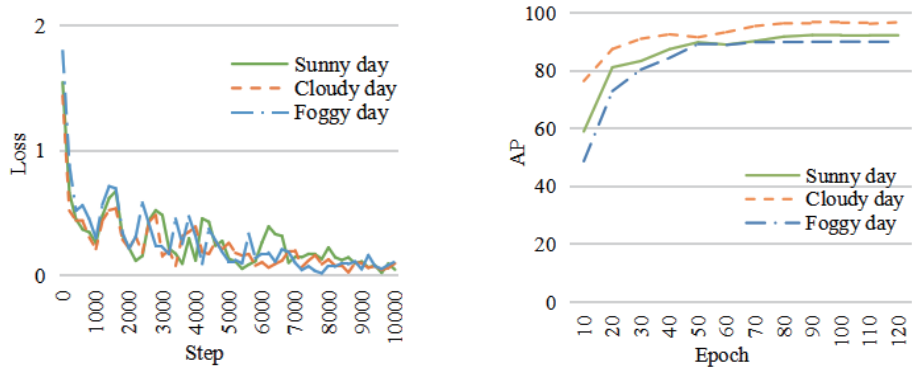
characteristics, the front layer of the network acts as a feature extractor, and the extracted features are general [19].

The number of training rounds and optimization function settings with the same backbone network, whether to use the pre-training model is very different. The performance comparison of transfer learning to the model is shown in Table 2.

Table 2. Comparison of migration learning performance

Data set	Transfer learning	AP
Sunny day	no	79.18%
Sunny day	yes	**92.34%**
Cloudy day	no	85.9%
Cloudy day	yes	**96.84%**
Foggy day	no	81.47%
Foggy day	yes	**90.05%**

We use the pre-trained model on Imagenet to train again. The fruit detection AP on the sunny day data set increased from 79.18% to 92.34%. The detection AP on the cloudy data set increased from 85.9% to 96.84%. The detection AP on foggy days increased from 81.47% to 90.05%. Moreover, the detection AP evaluated by the model on the test set was 92.34% on sunny days, 96.89% on cloudy days, and 89.2% on foggy days. Therefore, using transfer learning can optimize model performance. Fig. 5 shows the training loss value (a) and training accuracy (b) after using the pre-training model.



(a) Loss (b) AP

Fig. 5. Training

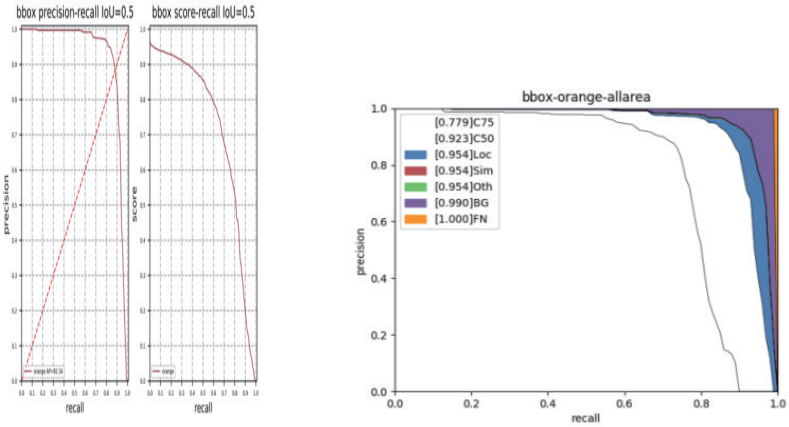
Combined with transfer learning, the trained model is stable. And it verified that the data set distribution is relatively uniform. The detection effect is better on cloudy days. On a sunny day, the light spots generated by sunlight will affect the recognition performance. The foggy day is limited by the dim and low environment and the heavy fog noise on the image, which is not good.

5 Discussion

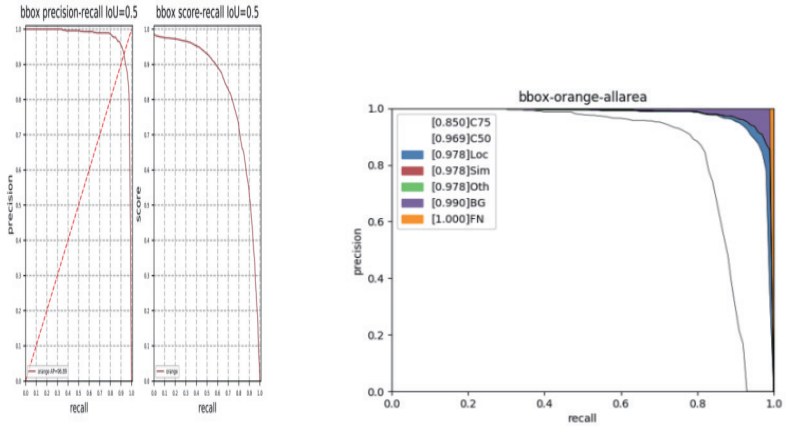
5.1 Model Effect Analysis

This paper chooses the two-stage detection model FasterRCNN as the baseline model. The backbone network chooses ResNet101_vd, introduces deformable convolution, and uses the pre-trained model obtained on

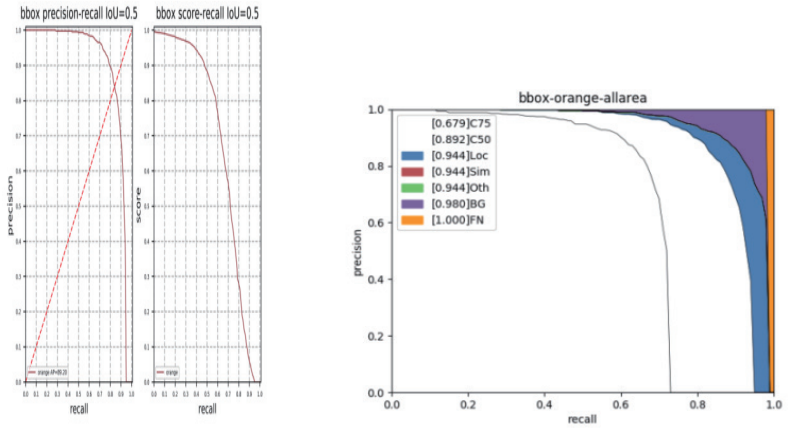
Imagenet. After the training model is completed, the accuracy AP of the model on the test set is 92.34% on sunny days, 96.89% on cloudy days, and 89.2% on foggy days. We evaluate the P-R curve on the test set and analyze the reasons for the model's prediction errors on the test set. The result is shown in Fig. 6.



(a) Sunny day



(b) Cloudy day



(c) Foggy day

Fig. 6. P-R curve

Fig. 6 shows the comparison of a single precision-recall curve with IoU set to 0.5 (left picture) and 7 PR curves with other values (right) under sunny (a), cloudy (b), and foggy (c) environments. Figure). In the figure on the right, due to the gradual relaxation of the assessment requirements, the AP indicated by the curve gradually increases.

C75 means that the IoU is set to 0.75, and the AP is 0.779 for sunny days, 0.850 for cloudy days, and 0.679 for foggy days. C50 means IoU is set to 0.5, and AP is 0.923 for sunny days, 0.969 for cloudy days, and 0.892 for foggy days. The white area between C50 and C75 represents the AP gain that relaxes IoU from 0.75 to 0.5. Loc means IoU is set to 0.1, and AP is sunny, cloudy, and foggy, respectively. In the same way, the area of the blue area represents the AP gain of IoU relaxation. The larger the area, the more the detection frame position is not accurate enough. Sim said that based on Loc, if the categories of the detection frame and the truth frame are not the same, but both belong to the same subcategory, then the detection frame is judged to be correct. The larger the red area between Sim and Loc, the higher the degree of confusion between sub-categories. This experiment only detects a single target of navel orange, and there is no confusion between sub-categories. Oth means that based on Sim if the subcategories of the detection frame and the truth frame are not the same, it is judged that the detection frame is correct. The larger the area of the green area between Oth and Sim, the higher the degree of confusion between subcategories. There is no such situation in the navel orange data set in this article. Therefore, AP is the same value under the evaluation conditions of Loc, Sim, and Oth. BG means that based on Oth, the detection frame of the background area is not considered wrong. The larger the purple area between BG and Oth, the greater the number of false detections in the background area. FN means that based on BG, the missing truth box is not considered wrong. The larger the area of the orange area between FN and BG, the greater the number of missing truth boxes.

Analyzing Fig. 7 shows that the detection effect of cloudy navel oranges is better, and there are a few cases where the detection frame does not reach IoU 0.5, and there are fewer false detections and missed detections location is more accurate. The most severe problems in foggy days are false detections, wrong locations, and missed detections. The most severe problem in sunny weather is false detection and wrong location.

We choose the current mainstream one-stage detection network YOLOv4 and the two-stage detection network Faster RCNN with vgg16 as the backbone network to train the navel orange fruit detection model. Moreover, compared with the performance of the algorithm model used in this paper. The results are shown in Table 3.

Table 3. Performance comparison

Network model	Data set	AP
The algorithm of this article	sunny day	**92.34%**
	cloudy day	**96.84%**
	foggy day	**90.05%**
Faster RCNN_vgg16	sunny day	90.65%
	cloudy day	92.18%
	foggy day	88.31%
YOLO-v4	sunny day	90.77%
	cloudy day	92.24%
	foggy day	86.42%

The detection performance of the algorithm used in this paper is better than Faster RCNN_vgg16 and YOLO-v4 on sunny, cloudy, and foggy days. And the performance of Faster RCNN_vgg16 and YOLO-v4 are comparable. The experimental results show that the detection algorithm used in this paper is better.

5.2 Forecast Visualization

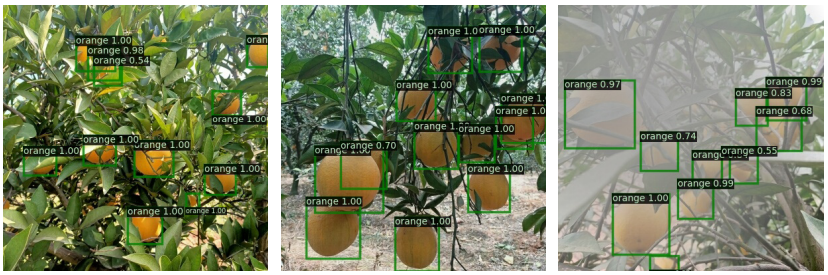
The test data does not have annotated information, so image augmentation is not required:

1) We read the specified picture, enter the network, calculate the prediction box and score, and then use non-maximum value suppression to eliminate redundant boxes.

2) The forecast results are visualized. We identify the navel orange fruit on the final output image.

3) Framing the bounding box of each detected fruit and mark the category and confidence.

The visualization of part of the test results is shown in Fig. 7.



(a) Sunny day (b) Cloudy day (c) Foggy day

Fig. 7. Test results

On sunny days, there are problems of misdetection and insufficient location accuracy. The detection effect is better on cloudy days, and the main problem on foggy days is missed detection and false detection. For the test data set, the model estimates that the AP reached 92.34% on sunny days, 96.89% on cloudy days, and 89.2% on foggy days. The results show that the model is well developed and has good generalization.

6 Conclusions

This reflects the potential of this method to detect and identify mature navel oranges on trees in various weather environments (such as sunny, cloudy, and foggy). The model training showed that the AP reached 92.34% on sunny days, 96.89% on cloudy days, and 89.2% on foggy days. When evaluating the model, the AP reached 92.34% on sunny days, 96.89% on cloudy days, and 89.2% on foggy days. This model is also compared with famous structures such as Faster RCNN, which uses vgg16 as the backbone network and YOLOv4, and the performance is better. The processing time of this model is about 63.84fps on sunny days, 68.6fps on cloudy days, and 56.29fps on foggy days, which meets the requirements for real-time applications. Therefore, this study proposes a robust method to meet the requirements of efficient real-time harvesting in unstructured navel orange orchards.

The method used in this paper has the potential to detect and identify mature navel oranges on trees in a variety of complex weather conditions. Therefore, this study proposes a robust method to meet the requirements of efficient real-time and accurate harvesting in unstructured navel orange orchards. Due to the limited training data in this article, a large number of image dataset training models with different backgrounds can be added in the future to further improve the detection accuracy and generalization ability. Future work will explore the detection of navel oranges at night to provide support for the realization of all-weather automated operations. In the follow-up, a deep learning model will also be trained to estimate the size of the navel orange fruit, and provide ideas for the classification of navel orange products.

Acknowledgements

This work was partially supported by the National Natural Science Foundation of China [grant agreement no. 42061067] and the Science and Technology Project of Jiangxi Provincial Department of Education of China [grant agreement no. GJJ201408].

References

[1] S.-Q. Wang, P.-H. Xie, S.Q. Wang, Analysis on the development status of Gannan navel orange characteristic industry based on SWOT analysis, Contemporary Rural Finance and Economics (2)(2021) 8-12.

[2] B.-X. Ma, Y.-T. Jia, W.-J. Mei, G.-G. Gao, C. Lv, Q. Zhou, Study on the Recognition Method of Grape in Different Natural Environment, Modern Food Science and Technology 31(9)(2015) 145-149+168.
[3] M. Gherlone, L. Iurlaro, M.-D. Sciuva, A novel algorithm for shape parameter selection in radial basis functions collocation method, Composite Structures 94(2)(2012) 453-461.
[4] J. Lu, W.-S. Lee, H. Gan, X.-W. Hu, Immature citrus fruit detection based on local binary pattern feature and hierarchical contour analysis, Biosystems Engineering 171(2018) 78-90.
[5] C. Liu, L. Gong, W. Zhang, Manipulating complex robot behavior for autonomous and continuous operations, in: V. Sezer, S. Öncü, P.B. Baykas (Eds.), Service Robotics [Internet], London: IntechOpen, 2020.
[6] G.-L. Chu, W. Zhang, Y.-J. Wang, N.-N. Ding, Y.-Y. Liu, A method of fruit picking robot target identification based on machine vision, Journal of Chinese Agriculture Mechanization 39(2)(2018) 83-88.
[7] X. Sun, G. Li, S. Xu, FSD: feature skyscraper detector for stem end and blossom end of navel orange, Machine Vision and Applications 32(1)(2021) 1-13.
[8] X. Ling, Y.-S. Zhao, L. Gong, C.-L. Liu, T. Wang, Dual-arm cooperation and implementing for robotic harvesting tomato using binocular vision, Robotics and Autonomous Systems 114(2019) 134-143.
[9] H. Li, H.-X. Tao, L.-H. Cui, D.-W. Liu, J.-T. Sun, M. Zhang, Recognition and Localization Method of Tomato Based on SOM-K-means Algorithm, Transactions of the Chinese Society for Agricultural Machinery 52(1)(2021) 23-29.
[10]D.-H. Li, X.-J. Bao, X. Yu, Q. Gao, Detection and recognition of green apple in natural environment based on YOLOv3 network, Laser Journal (42)(1)(2021) 71-77.
[11]F.-S. Wang, Q.-S. Wang, J.-G. Chen, F.-R. Liu, Improved Faster R-CNN target detection algorithm based on attention mechanism and Soft-NMS, Laser & Optoelectronics Progress 58(24)(2021) 2420001:1-12.
[12]P. Jing, M.-Z. Li, T. Cheng, N.-N. Sun, H.-C. Zhang, X.-B. Xia, J.-C. Yang, Application of Machine Learning Algorithms in Smart Production of Apple, Journal of Jilin Agricultural University (2)(2021) 138-145.
[13]S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: towards real-time object detection with region proposal networks, IEEE Transactions on Pattern Analysis & Machine Intelligence 39(6)(2017) 1137-1149.
[14]J.-Q. Fan, T.-J. Huo, X. Li, T. Qu, B.-Z. Gao, H. Chen, Covered vehicle detection in autonomous driving based on Faster RCNN, in: Proceedings of the 39th Chinese Control Conference (CCC), 2020.
[15]P.-C. Zhang, Research and system implementation of rapid target object recognition in an open environment [Master dissertation] Beijing: Beijing University of Posts and Telecommunications, 2019.
[16]K.-H. Zhang, H.-K. Shen, Solder Joint Defect Detection in the Connectors Using Improved Faster- RCNN Algorithm, Applied Sciences 11(2)(2021) 576.
[17]J.-T. Wang, W.-L. Song, K.-X. Li, J.-P. Huang, H. -M. Jia, A Stoma Detection Method for Living Plant Leaves with Faster R-CNN, Journal of Northeast Forestry University (2)(2020) 34-39.
[18]S. Ruder, An overview of gradient descent optimization algorithms. <https://arxiv.org/abs/1609.04747>, 2016 (accessed 20.11.10).
[19]C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, C. Liu, A survey on deep transfer learning. <https://arxiv.org/abs/1808.01974>, 2018 (accessed 20.11.21).