Novel green-fruit detection algorithm based on D2D framework

Jinmeng Wei[1], Yanhui Ding[1], Jie Liu[1], Muhammad Zakir Ullah[1], Xiang Yin[2], Weikuan Jia[1,3*]

(1. *School of Information Science and Engineering, Shandong Normal University, Jinan 250358, China*;
2. *School of Agricultural Engineering and Food Science, Shandong University of Technology, Zibo 255000, Shandong, China*;
3. *Key Laboratory of Facility Agriculture Measurement and Control Technology and Equipment of Machinery Industry, Zhenjiang 212013, Jiangsu, China*)

Abstract: In the complex orchard environment, the efficient and accurate detection of object fruit is the basic requirement to realize the orchard yield measurement and automatic harvesting. Sometimes it is hard to differentiate between the object fruits and the background because of the similar color, and it is challenging due to the ambient light and camera angle by which the photos have been taken. These problems make it hard to detect green fruits in orchard environments. In this study, a two-stage dense to detection framework (D2D) was proposed to detect green fruits in orchard environments. The proposed model was based on multi-scale feature extraction of target fruit by using feature pyramid networks MobileNetV2 +FPN structure and generated region proposal of target fruit by using Region Proposal Network (RPN) structure. In the regression branch, the offset of each local feature was calculated, and the positive and negative samples of the region proposals were predicted by a binary mask prediction to reduce the interference of the background to the prediction box. In the classification branch, features were extracted from each sub-region of the region proposal, and features with distinguishing information were obtained through adaptive weighted pooling to achieve accurate classification. The new proposed model adopted an anchor-free frame design, which improves the generalization ability, makes the model more robust, and reduces the storage requirements. The experimental results of persimmon and green apple datasets show that the new model has the best detection performance, which can provide theoretical reference for other green object detection.

Keywords: green-fruit detection, D2D framework, automatic harvesting, MobileNetV2+FPN, binary mask prediction, anchor-free
DOI: 10.25165/j.ijabe.20221501.6943

Citation: Wei J M, Ding Y H, Liu J, Ullah M Z, Yin X, Jia W K. Novel green-fruit detection algorithm based on D2D framework. Int J Agric & Biol Eng, 2022; 15(1): 251–259.

1 Introduction

At present, in the production of fruit and vegetable industry, picking operation as an important link of its production management is still based on manual picking, which makes it become the most time-consuming and laborious link in the whole fruit and vegetable production cycle[1,2]. Fruit and vegetable picking robots can effectively alleviate the high labor costs, low efficiency in the process of manual picking common problems. As an important part of the fruit picking robot, the accuracy, efficiency, and robustness of the visual recognition system in fruit detection will greatly affect the picking quality of robots[3,4]. However, in a complex orchard environment, the visual recognition system can be influenced by many factors such as light intensity, angle of the image taken, leaf occlusion, fruit color, background, and so on. These factors can highly influence the visual recognition system and bring more challenges in detecting targeted fruits. Therefore, fruit-harvesting robots equipped with stable visual recognition systems will become the key to realizing the efficient detection of the target fruit and the intelligent management of the orchard.

Traditional machine learning has laid the foundation for the research of computer vision, and the current machine learning method has been quite mature. Its simple workflow is favored by researchers. Traditional machine learning plays an important role in the field of object detection[5-7] and has achieved gratifying results in green fruit detection. Arefi[8] first removed the background in Red-Green-Blue (RGB) space, then extracted the ripe tomato area by combining RGB, Hue-Saturation-Intensity (HSI) and YIQ space. Finally, the shape features were used to locate the fruit area, and the overall accuracy of the algorithm reached 96.36%. Linker[9] proposed a "four-step" strategy to realize the prediction aimed at estimating orchard yield. The method was mainly based on fruit color, texture, edge shape and other feature information, and the recognition rate of green apples could reach 95% under natural light conditions. Based on RGB color space, Liao[10] used the Otsu threshold segmentation algorithm to remove the influence of branches in green apple images, extract the gray scale and texture features of leaves and apples, and establish a random forest recognition model. The recognition accuracy of green apples reached 88%. Tian[11] proposed a target fruit localization method based on depth image, and located the center and radius of the apple circle respectively through depth image and its corresponding RGB spatial information, so as to fit the target area. Li[12] combined applying saliency detection and Gaussian curve fitting algorithm, a novel

Received date: 2021-07-27 **Accepted date:** 2021-12-12
Biographies: Jinmeng Wei, MS candidate, research interest: image processing, artificial intelligence, agricultural information, Email: wjmeng0514@163.com; **Yanhui Ding**, PhD, Professor, research interest: machine learning, image processing, Email: yanhuiding@126.com; **Jie Liu**, MS candidate, research interest: target recognition, artificial intelligence, Email: liujiefeier@163.com; **Muhammad Zakir Ullah**, MS candidate, research interest: artificial intelligence, image processing, Email: zakirshah899@gmail.com; **Xiang Yin**, PhD, Associate Professor, research interest: agricultural information, navigation control, Email: yinxiang2013@yahoo.co.jp.
***Corresponding author: Weikuan Jia**, PhD, Associate Professor, research interest: smart agriculture, artificial intelligence. School of Information Science and Engineering, Shandong Normal University, No. 1# Daxue Rd., Changqing District, Jinan 250358, China. Tel: +86-531-86181750, Email: jwk_1982@163.com.

algorithm is used to detect green apples in natural scenes, the experimental results indicated that it was effective and feasible. Machine learning methods are mostly based on the texture features and color of the target fruit. When the fruit is affected by light intensity, occlusion and the color similarity between the target fruit and the leaf, the texture features of the target fruit are not obvious and the shape is missing. The above methods are difficult to meet the requirements of accuracy and speed when intelligent technologies are deployed to practical application.

In recent years, with the development of deep learning and CNN, the accuracy of image recognition has been greatly improved. Its advantages of end-to-end automatic detection process and deep extraction of image features eliminate many complex operations of traditional visual algorithms, which attract many researchers to apply it to target fruit location recognition[13,14]. Bargoti[15] et al. first used multi-scale multi-layer perceptron and CNN to segment apple images and extract apple targets in the images. Then, watershed algorithm and circular Hough transform were used to identify and count apple targets. Kang[16] obtained DSSNet-V2 by improving DASNet on the basis of achieving the class-level segmentation of target fruits, further realized the instant-level segmentation of target fruits and the class-level segmentation of branches and leaves, and solved the problem of identifying a cluster of fruits as the same region caused by overlapping factors in DSSNet. Jia[17] improved the instance segmentation model Mask RCNN to adapt to the detection of apple targets. By combining the residual network (ResNet) and densely connected convolutional networks (DenseNet) as the feature extraction network of the original model, the detection accuracy of apple targets under overlapping and foliage occlusion environments was greatly improved. Wang[18] proposed an apple recognition model based on R-FCN, by means of ResNet, RPN, and ROI sub-net module, the target fruits were detected in two stages. The recognition accuracy of this method is 95.1% on the test set containing occlusion, blur, and overlapping apple targets. The accuracy and applicability of the above vision model based on deep learning are better than the traditional machine learning methods. But these methods need a lot of computing and storage resources, picking efficiency cannot meet the needs of picking robots in real-time. In addition, the power consumption and stability of the picking robot should be considered when it is deployed in the real environment.

In order to improve the accuracy of target fruit recognition and enable the robot to meet the real-time operation requirements in the complex orchard environment, an object detection model optimized by dense to detection framework (D2D) was proposed. The new model uses the lightweight network MobileNetV2[19] as the backbone network for feature extraction, which solves the problem of high demand for computing and storage resources. The backbone network connects feature pyramid networks (FPN)[20] to realize multi-scale feature fusion, and RPN[21] is added to realize ROI region extraction. It can enhance the feature information and improve the anti-interference ability of the model to the complex environment of orchard. Moreover, multiple local regressions and discriminative ROI pooling of the new model make target regression and classification more accurate, and improve the real-time efficiency of intelligent picking. The new model solves the problem of low accuracy of fruit recognition by robots in the past, and further promotes the development of agricultural intelligent picking technology. Through the model comparison experiment, it can be proved that the accuracy of the new model has great advantages, which further promotes the development of intelligent picking technology and cross domain target detection.

2 Methods

2.1 Datasets collection and labeling

In this study, immature persimmons and green apples were selected as research objects, and the fruits presented green spherical, which met the research requirements.

Image collection object: Green persimmons and apples in growing period, the persimmon varieties include “niuxin” persimmon, “jixinhuang” persimmon, “jingmian” persimmon, etc. Apple variety is “gala” apple.

Image collection location: The mountain behind Shandong Normal University (Changqing campus) and the southern mountainous area of Jinan City, China.

Image collection equipment: Canon EOS 80D SLR camera. The camera used CMOS (complex metal oxide semiconductor) image sensor. The image resolution was 6000 pixels×4000 pixels, saved in JPG format, 24-bit color images.

Image collection conditions and setup: The images were taken from multiple angles in the real scene and the natural orchard environment, including perspective, close view, side view, front view, overlapping, occlusion, and other different angles. The image collection time was divided into the early morning, noon and night. In the early morning, fruit images were collected under the condition of soft light, so the frost and dew might appear on the fruit. At noon, fruit images were collected under the strong light environment (including the situation of forwarding light and backward light), and at the night, fruit images were collected under the LED artificial auxiliary light environment. The collected fruit images fully considered the complexity of the orchard environment and had randomness and representation, which can maximize the real-time operation requirements of agricultural equipment.

A total of 568 images of persimmons and 1361 images of apples were collected in the experiment, which were used as datasets after post-processing. As shown in Figures 1a and 1b, there are many complex situations, such as overlap, backlight, occlusion, direct light, distant view, night view, and so on. More representative and convincing results can be obtained by using data from many complex environments. In order to meet the requirements of orchard real-time object detection and reduce the subsequent experiment time, the image resolution of 6000×4000 pixels was compressed into 600×400 pixels. Before production of the datasets, it is necessary to preprocess these images, specifically including the normalization, clipping, flipping, smoothing, and other operations of data.

As shown in Figures 1a and 1b, when the fruit is shaded by the leaves and branches or the fruit overlaps, the outline of the fruit is easy to be not clear and complete. At night or in rainy weather, the fruit detection accuracy will always decrease because of the change in light and raindrops on the fruit surface.

First, LabelMe software was used to mark the green fruit images, the contour of fruit was marked as a connected area, and the category information of fruit was marked. The coordinates of the marking points and the labeled category would generate the corresponding JSON file. Then, the JSON file was converted into datasets in COCO format by Label Me software. The persimmon dataset in this study contains a total of 568 images of persimmons, including 412 images in the training set and 156 images in the test set. The apple dataset contains a total of 1361 images of apples, including 953 images in the training set and 408 images in the test set.

a. Images of green persimmons

b. Images of green apples

Figure 1 Images of green-fruit under different environmental conditions

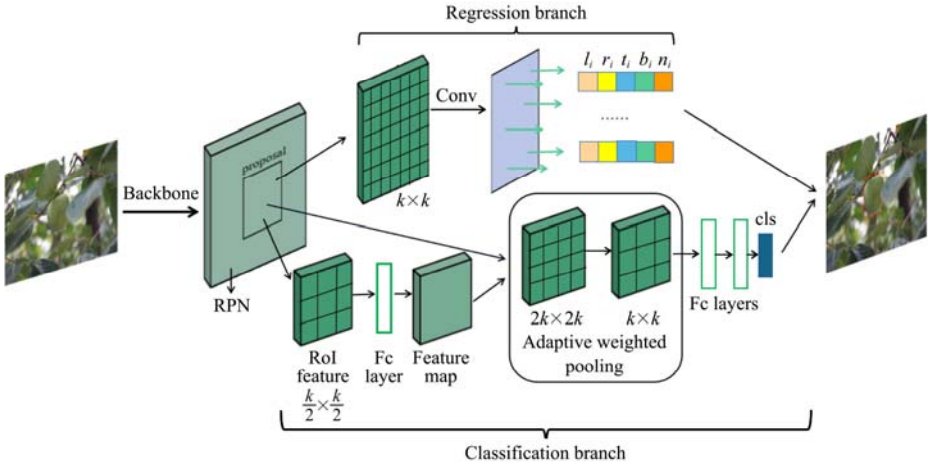
2.2 D2D object detection network for fruit detection

Inspired by D2D[22] model and considering the complex orchard environment, a two-stage anchor-free D2D detection model was proposed to realize the efficient and accurate detection of green targets. The accuracy of object detection can be improved by using the two-stage method. The D2D model generates the target bounding box in the first stage and identifies the target category of the bounding box in the second stage.

Figure 2 shows the overall structure of the new model. The lightweight MobileNetV2 structure was selected as the backbone network in our proposed model. The use of MobileNetV3[23] cannot greatly improve the accuracy and will reduce the robustness of the model. Therefore, the MobileNetV2 network with the best comprehensive effect is selected. The image features were extracted by the method of the first dimension increasing and then dimension decreasing. Then, the RPN structure was used to generate region proposals of green target fruits. In the process of object detection, classification and regression were divided into two independent parallel branches in the D2D network structure due to their different sensitive regions to feature space. The regression branch was used to accurately locate the target of the input ROI feature and generated a detection box containing green fruit; the classification branch classified the input proposal accurately, then generated the classification label and the classification confidence. The new model finally integrated the output results of the classification and regression branch, and then output detection boxes with classification labels and confidence.

The regression branch calculated the offsets of $k\times k$ local features of ROI features to the Ground Truth box, and selected whether each local feature belonged to the positive sample (the local features of ROI>0.5 belonged to the positive sample). Finally, the average value of all positive sample offsets was obtained as the global offset. The regression calculation of dense local features made the target location not limited to the coordinate of a central point (for example, Faster RCNN), which made the target location more accurate, reduced the dependence on a certain point, and improved the detection accuracy of green target fruit of the robot in the complex orchard environment.

The ROI Align[24] of $\frac{k}{2}\times\frac{k}{2}$ size was selected for the classification branch, followed by the fully connected layer to realize the lightweight weight offsets prediction and obtain the ROI feature of $2k\times 2k$ size. The four sampling points of each sub-region of the ROI feature were allocated different weights adaptively through convolution operation. The sampling points with discriminative features were allocated higher weights to obtain more effective feature information and improve the accuracy of classification.

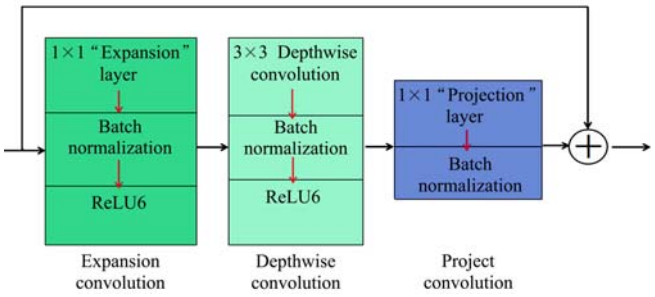


Note: RPN: region proposal network; Conv: convolution; ROI: regions of interest; Fc layers: fully connected layers; l_i, r_i, t_i and b_i respectively represent the offset from the i-th local feature to the left, right, up, and down of GT; n_i has only 1 and 0 values, which respectively represent that the i-th local feature belongs to the ground truth bounding box or background.

Figure 2 Overall structure of the flow chart of D2D

2.2.1 Backbone network

In the process of automatic orchard harvesting and yield measurement, the traditional network model chose ResNet-101 as the backbone network. In order to improve the operation efficiency of equipment, the backbone network of the new model is suitable for choosing a lightweight convolutional neural network with fewer parameters, less computation, and fast operation speed: MobileNetV2. The module is suitable for mobile devices (such as robot harvesting). The implementation process reduces the need for embedded hardware devices to access main memory, as shown in Figure 3. In contrast to the traditional Residual Network[25] structure, MobileNetV2 is a backward residual structure with a linear bottleneck, which extracts image features by first increasing dimension and then decreasing dimension to obtain more channel information.



Note: ReLU: Rectified Linear Units. In the first part, the expansion layer uses the size of 1×1 convolutional layer to map low dimensional space to high dimensional space. The second part is the structure of the depthwise separable convolution, which is used to extract features. In the third part, the projection layer also uses a convolutional layer to map high dimensional space to low dimensional space.

Figure 3 Backbone network diagram of the MobileNetV2

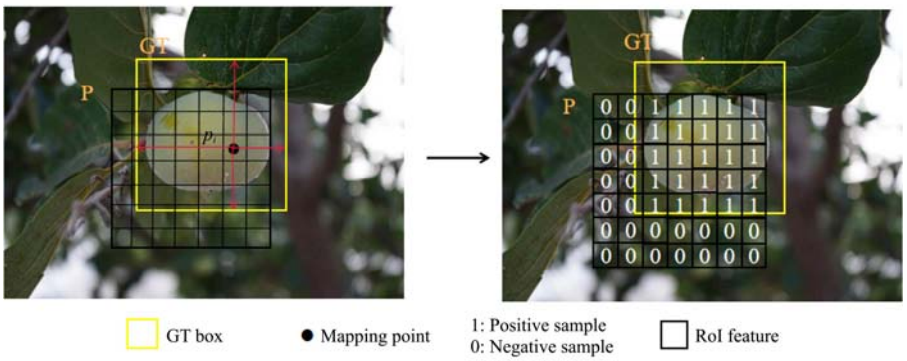
MobileNetV2 belongs to the lightweight convolutional neural network, which has low dimension, a small amount of computation and high speed in the convolutional layer. However, the accuracy of lightweight network is relatively low. In order to balance the relationship between efficiency and accuracy, MobileNetV2 also implements high-dimensional feature extraction on the basis of optimized speed. MobileNetV2 can improve the recognition accuracy by inserting a linear bottleneck after the convolution module to capture the features of ROI. MobileNetV2 implements lightweight high-dimensional feature extraction, which reduces the memory capacity requirements of the model. MobileNetV2 is followed by an FPN to achieve multi-scale feature fusion. The D2D model uses the backbone network of MobileNetV2+FPN to realize the high dimension extraction of image features, which improves the accuracy of the model.

2.2.2 Regression branch

Aiming at the multiple green target fruit region proposals generate by RPN structure, the ROI features with $k\times k$ adjacent local feature spaces are obtained by constructing ROI Align mapping, and the offsets of local features are regressed by using the fully convolutional network. The calculation method of local feature offsets in the regression branch is shown in Figure 4a. GT (Ground Truth) box represents the real target box, and the ROI feature is divided into $k\times k$ local features, let $p_i(x_i, y_i)$, $i\in[1, k^2]$ represent the coordinates of the local feature. Moreover, let (x_1, y_1) represent the coordinate of the upper left corner of GT and (x_2, y_2) represent the coordinate of the lower right corner of GT. Then, the model calculated the distance from each local feature $p_i(x_i, y_i)$ to the upper left and lower right corner of GT. The offset vectors from p_i to the upper, lower, left and right directions of GT are respectively represented by $(\hat{t}_i, \hat{b}_i, \hat{l}_i, \hat{r}_i)$. The offset values in the upper, lower, left and right directions of GT are presented by (t_i, b_i, l_i, r_i) and k^2 position offsets are obtained in the four directions. The specific equation is as follows:

$$\begin{cases} l_i = (x_i - x_1) / w_P \\ t_i = (y_i - y_1) / h_P \\ r_i = (x_2 - x_i) / w_P \\ b_i = (y_2 - y_i) / h_P \end{cases} \quad (1)$$

where, w_P and h_P are the width and height of P.



a. Calculation of local feature offsets in regression branch b. Distinction between positive and negative samples

Note: GT: Ground truth; P is the candidate proposal; p_i is the local feature of ROI feature.

Figure 4 Calculation of local feature offsets in regression branch and distinction between positive and negative samples

In order to reduce the interference of background to the overall offset, the regression branch distinguishes positive and negative samples of $k\times k$ local features, and introduces a vector $\hat{n}$ for binary mask prediction. As shown in Figure 4b, the local features of the overlapping area of GT and ROI features are positive samples belonging to the fruit target, and the others are negative samples. Positive and negative samples are marked by introducing vector $\hat{n}$, where $\hat{n} = \{\hat{n}_i : i \in [1, k^2]\}$, and in the training process, $\hat{n}_i$ adopts a normalized function, when $\sigma(\hat{n}_i) > 0.5$, $n_i = 1$, otherwise, $n_i = 0$. The local features of $n_i = 1$ are regarded as a positive sample, and the offsets of the local features of $n_i = 0$ are removed. Finally, the regression branch will predict five outputs $(\hat{l}_i, \hat{t}_i, \hat{r}_i, \hat{b}_i, \hat{n}_i)$, where,

$$n_i = \begin{cases} 1, & \text{if } p_i \in GT;\ \forall p_i \in P \\ 0, & p_i \notin GT \end{cases} \quad (2)$$

For all positive samples p_i with $\sigma(\hat{n}_i) > 0.5$, the regression branch calculates their offsets to the upper left corner and the lower right corner. Finally, the average offset of all positive samples will be calculated to represent the regression box.

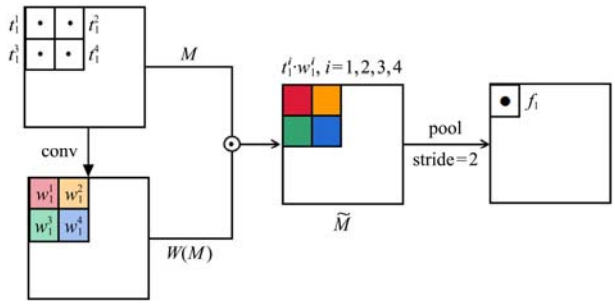
From the perspective of the regression branch, the detection accuracy and efficiency of the new method were improved. The regression branch calculates the offsets based on each local feature of the region proposal of the target fruit. Compared with the Faster RCNN[26] which takes the center point of the region proposal as the calculation basis, the accuracy of the target region obtained by the new model was higher. In the process of regression modeling, the new method is based on the offsets of local features. Compared with the Fully Convolutional One-Stage Object Detection (FCOS)[27] network represented by the offsets of pixels, the computation amount of the new method was greatly reduced and the training efficiency was effectively improved. The new method balances the detection accuracy and efficiency well and is beneficial to the real-time operation of the orchard field.

2.2.3 Classification branch

Inspired by Deformable ROI Pooling[28], the classification branch of D2D model uses fully connected layer (fc layer) to predict offsets of four sampling points in each sub-region of ROI, and then uses adaptive weights to assign higher weights to discriminative sampling points on ROI to obtain more accurate features. The overall structure of classification branches is shown in Figure 2, the overall structure can be divided into two parts: ROI Align and adaptive weight allocation.

Compared with the Deformable ROI Pooling structure, the overall framework of classification branch ROI Align is similar. However, the D2D model uses lighter weight to predict the offsets of sampling points. The target fruit region proposals use the size of $\frac{k}{2}\times\frac{k}{2}$ ROI Align operation to segment the ROI feature into $\frac{k}{2}\times\frac{k}{2}$ size, and parameters are only $\frac{1}{4}$ of the standard offset prediction. Different from the integer operation of ROI Pooling, the ROI Align feature mapping is retained to the decimal number, and the bi-linear interpolation method is adopted to calculate the final result. Moreover, the two-time quantization process is canceled to avoid the regression error caused by quantization.

The fully connected layers are adopted to learn the offsets, and the ROI feature is migrated horizontally and vertically according to the ratio of length to width. By increasing offsets of the sampling points on the convolution kernel of the feature map, the size of the receptive field can be changed to make the convolution kernel into a polygon, and more effective feature information can be obtained. Finally, ROI Align unifies the feature maps corresponding to ROIs of different sizes to a fixed size of $2k\times2k$.



Note: "●" represents the sampling points; f_1 represents the sampling points after pooling.

Figure 5 A process diagram of pooling sampling points in a sub-region of the ROI feature

The target fruit region proposal and the sampling points with offsets are used as the input of the convolutional layer, and the ROI features with discriminative features are obtained by adaptive weight pooling, as shown in Figure 5. In the ROI feature with the size of $2k\times2k$, sampling points are set as 4 in each sub-region. At the center point of the sampling point, the pixel at each center point is calculated by bi-linear interpolation. M represents the feature in ROI, that is $M\in\mathrm{ROI}^{2k\times2k}$, the weight of each sampling point is predicted by a convolutional network, denoted as $W(M)\in ROI^{2k\times2k}$. $W(M)$ represents the discrimination ability of sampling points in the sub-region of $2k\times2k$, spaces, and sampling points with discriminative features will be given higher weight. $\tilde{M}$ represents the weighted ROI feature as follows:

$$\tilde{M}=W(M)\odot M \tag{3}$$

where, $\odot$ is called Hadamard product, the weight $W(M)$ corresponding to each sampling point was obtained from M, and $\tilde{M}$ was operated by average pooling with a step size of 2, and the size of the ROI feature is re-mapped back to $k\times k$. Finally, two fully connected layers are connected as classifiers to obtain the classification score of the region proposal.

In this study, the method can adaptively assign higher weights to the more discriminative sampling points, and get the highly discriminative features. It has high accuracy and improves the detection efficiency in the target fruit binary classification.

2.2.4 Loss function

The quality of the loss function design can directly affect the performance of the model and plays an important role in the iterative optimization process of the model. When training the model, it is necessary to define the loss function first. Then a prediction test is obtained according to the forward propagation, and the test value is obtained by comparing it with the real sample. Finally, the back propagation is used to update the weight, and the loss function is iterated to the minimum to obtain the ideal detection model. The overall loss function of the D2D model is composed of the loss function of regression and classification modules, which are respectively regression loss and classification loss. When the predicted value is close to the true value, the loss function is low, and when the difference between predicted value and true value is close to 1, the loss function value is high. The D2D model uses cross entropy loss function in the classification process and binary cross entropy loss function in the regression process, the total loss function of the new model is defined as

$$\begin{aligned}L_{D2D}&=L_{cls}+L_{reg}\\&=-\sum_{k=1}^{N}(p_k*\log q_k)+\frac{1}{N}\sum_i-[y_i\cdot\log(x_i)+(1-y_i)\cdot\log(1-x_i)]\end{aligned} \tag{4}$$

where, L_{D2D} represents the overall loss function of D2D model; L_{cls} represents the loss function of classification branch; L_{reg} represents the loss function of regression branch; x_i represents the probability that the prediction is a positive sample; y_i represents the label of sample i, positive sample is 1, negative sample is 0.

In the binary classification model, the output of the binary classification prediction problem is often not the standard 1 and 0, and the original output of the neural network is not a probability value. Instead, the sigmoid function is used for activation processing, and then the probability of mapping the samples to positive and negative samples is mapped. Sigmoid function normalizes output value, and the range of value is between [0,1]. The advantage of using the cross entropy loss function is that the sigmoid function can avoid the reduction of the learning rate of the Mean Square Erroe loss function when the gradient descents. Because the learning rate can be controlled by the output error. The equation of the sigmoid function is as follows:

$$\sigma(z)=\frac{1}{1+e^{-z}} \tag{5}$$

where, z represents the input of sigmoid function, $z\in(-\infty, +\infty)$.

3 Analyses and results

3.1 Model training

The whole process of the experiment includes the processing of the data set, model training, model testing, and other main parts. The whole flow of the experiment is shown in Figure 6. In the experimental process, the setting of model parameters is also an important process, which directly determines the optimization of the model.

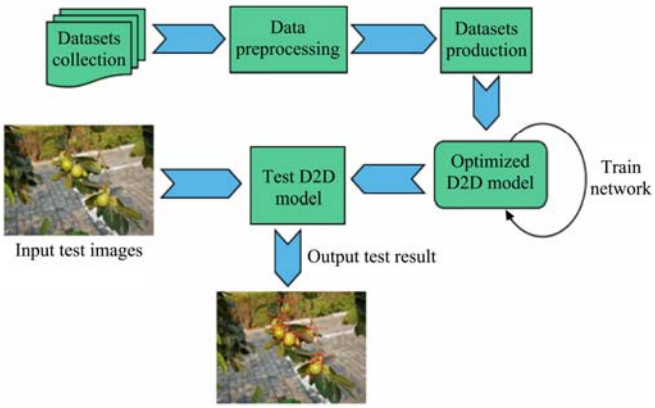


Figure 6 Overall flow chart of D2D model experiment

The experiment was performed on a personal computer, with a processor of Inter (R) Core (TM) i5-7200U, and Radeon™ R7 M445 graphics card with an 8GB of memory. The software environment was Ubuntu 16.04, Python 3.5, and PyTorch 3.7. The hardware environment is Intel i7-8700K, RAM 16G, and Nvidia GeForce GTX 1080 Ti GPU. In the process of model training, transfer learning can reduce the computation amount of the model and avoid the problem of overfitting.

The D2D model adopted the initial weight of preliminary training based on COCO datasets, which helps to stabilize the loss function and improve the training accuracy. The initial learning rate was set to 0.0025 during model training, with 24 of batch size, 150 times maximum iteration, 0.0001 weight attenuation and 0.9 momenta. In the training of the model, an iteration refers to the process of all data being sent into the network to complete a forward calculation and reverse propagation. A model usually needs multiple iterations to fit convergence, but it is not that the more iterations the better, too much training may lead to overfitting, so it is necessary to test and evaluate the trained model to find an optimal fit state.

3.2 Evaluation metrics

In order to quickly discover the possible problems of the model in the training process and iterative optimize the model, the new model needs to be evaluated. In this paper, average precision (AP) and average recall (AR) were used to evaluate the object detection performance of the model. The accuracy rate is the ratio of the correct detected target to the actual detected target, and the recall rate is the ratio of the correct detected target to the expected detected target. The equations for precision and recall are as follows:

$$\text{Precision}=\frac{\text{TP}}{\text{TP}+\text{FP}} \tag{6}$$

$$\text{Recall}=\frac{\text{TP}}{\text{TP}+\text{FN}} \tag{7}$$

where, TP represents the number of detection boxes for Intersection of Union (IOU)>0.5; FP represents the detection box with IOU≤0.5; FN represents the number of GT that was not detected. It is worth noting that the IOU represents the intersection threshold of the real box and the prediction box.

Under different IOU thresholds, the values of Precision and Recall will also change accordingly. Taking recall as the abscissa and precision as the ordinate, a Precision-Recall (P-R) curve can be drawn. The area under the P-R curve is the representation of Average Precision, and the equation of Average Precision (AP) can be expressed as follows:

$$\text{AP}=\int_0^1 P(r)\text{d}r \tag{8}$$

In this integral, $P(r)$ is a function with r as a parameter.

In fact, this integral is very close to the change of precision value multiplied by recall value for each threshold value, and then the product obtained under all thresholds is accumulated. The publicity is as follows:

$$\text{AP}\approx\sum_{k=1}^{S}P(k)\Delta r(k) \tag{9}$$

where, S represents the number of all images in the test set; $P(k)$ represents the Precision value when k images can be recognized; $\Delta r(k)$ represents the change of Recall value when the number of identified images changes from k–1 to k.

3.3 Green fruit detection results

In the experimental training process, the effect of real orchard scenes on fruit object detection will be fully considered. After the training, the optimal D2D model was selected based on mAP and mAR. The optimal D2D model was used to test the two datasets of green persimmons and green apples. And the visual analysis was made of the test results of the images in complex environments such as occlusion, overlap, rainy day, backlight, and night. As shown in Figure 7, the detection effects of persimmons and apples are shown in Figures 7a and 7b respectively. The average precision and average recall rate of the model detection of persimmon and apple are listed in Table 2 for comprehensive evaluation.

The complex orchard environment in which the fruit is located and the confusion of leaf color on the fruit bring serious challenges to object detection. As shown in Figure 7, the proposed model in this study gives the best detection accuracy, and there are almost no detection errors and omissions. Even the fruit that was badly shaded and overlapped, which resulted in unclear contour, or the small target fruit in the nighttime environment could be detected. And the color of the background leaves had little influence on the green fruit.

Table 2 shows the different average precision and average recall of persimmons and apples which are generated under different IOU thresholds, sizes, and quantities. Figure 8 shows the P-R curve of target fruits of different sizes when IOU=0.5. Take recall as abscissa, take 101 points evenly on the coordinate axis, and then calculate the precision corresponding to 101 different recall values respectively. In combination with Table 2, compared with distant fruits, severely occluded or overlapped fruits, the detection effect of close-range unshaded fruits is better. Among them, the effect of distant fruit is poor because of the small target, unclear outline, and overlap of leaves, which brings a great challenge to fruit recognition. At the same time, the density of fruit also has a certain influence on the effect of object detection. The sparse fruit outline is clear and complete, while the dense fruit is prone to occlusion and overlap, which brings difficulties to object detection. On the whole, the detection effect of the model

for the two kinds of fruits is good. Although the complex environment of the orchard may have a certain impact on the recognition effect of the target fruit, the new proposed model still has high accuracy and strong robustness.

a. Detection effect of green persimmons b. Detection effect of green apples

Note: Red boxes represent the fruit detected by the D2D model.

Figure 7 Detection effect of green-fruit under different environmental conditions

Table 2 Average precision and average recall of persimmon and apple datasets in different IOU, sizes and quantities

Metric	Persimmon value	Apple value
mAR^b	80.4	69.4
mAR^b_S	46.1	54.8
mAR^b_M	82.2	76.7
mAR^b_L	92.7	93.2
mAP^b	73.4	62.8
AP^b_{50}	89.6	85.9
mAP^b_S	35.7	45.6
mAP^b_M	74.6	69.7
mAP^b_L	87.7	89.6

Note: mAP^b: Mean Average Precision, the average value of AP for each class in object detection; AP^b_{50} : The AP value over a single threshold of IOU=0.5; mAP^b_S : The detection effect on small-scale fruits; mAP^b_M : The detection effect on medium-scale fruits; mAP^b_L : The detection effect on large-scale fruits; mAR^b: Mean Average Recall, the average value of AR for each class in object detection; mAR^b_S : Object detection of the average recall rate of small-scale fruits; mAR^b_M : Object detection of the average recall rate of medium-scale fruits; mAR^b_L : Object detection of the average recall rate of large-scale fruits.

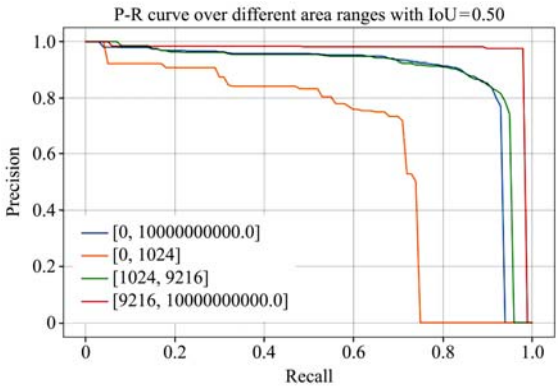


Figure 8 Under the threshold of IOU = 0.5, the P-R curve of the model for different scales of fruits with the maximum detection number of 100

Where the small size of fruit is less than 32^2, medium-sized fruits are between 32^2 and 64^2, the large size of fruit is large than 64^2.

3.4 Comparisons

In order to analyze the performance of this model objectively, our proposed model was compared with representative computer vision models such as Faster RCNN, Mask RCNN[29], FCOS, and SSD[30] algorithms. The experimental process adopted the same parameters and processing steps, and all persimmon datasets were used to evaluate and compare the average precision and average recall. The evaluation results were listed in Table 3. The comparison models were divided into two categories: anchor and anchor-free. Among them, Fast RCNN, Mask RCNN, and SSD are classic and stable object detection algorithms with anchor frame (Mask RCNN can be regarded as the combination of object detection and semantic segmentation, here only the part of object detection was compared). FCOS is a new algorithm with anchor-free. It can reduce the dependence on anchor frame in the detection process and improve the speed and accuracy of the model.

Table 3 Object detection performance comparison of five different networks

Method	Average recall	Average precision
Faster RCNN	76.9	71.2
Mask RCNN	77.9	73.2
FCOS	72.7	66.0
SSD	74.0	68.3
Our method	80.4	73.4

As can be seen from Table 3, the average recall rate and average precision of the D2D model are 80.4% and 73.4% respectively, which is higher than those of other models. It can be seen that the detection effect of the D2D model is better than that of other models. Compared with the FCOS model without anchor frame, the average recall and average precision of this model are 7.7% and 7.4% higher respectively. The average precision of this model is similar to the result of Mask RCNN, but the average recall

is 2.5% higher than that of Mask RCNN. This model still has advantages. A large number of experimental data show that the D2D model has a good detection effect. Moreover, this model uses the lightweight backbone network MobileNetV2, which is bound to be better than other models' efficiency. The environment of persimmon and apple datasets used in this study is complex, which contains a large number of overlapped and small target fruit. However, the detection effect of the D2D model on target fruits is pretty good, which met the real-time operation requirements of orchard target fruit detection equipment.

4 Conclusions

This study aimed to develop an object detection model in a complex orchard environment and put forward an efficient and accurate object detection model of green target fruit optimized by D2D. The new proposed model adopted a structure with an anchor-free, which avoided the dependence on anchor in the detection process, greatly improved the detection accuracy of the model, and made the model widely used in various agricultural fields. And the lightweight MobileNetV2 backbone network was introduced to reduce the number of parameters and calculations, which greatly improved the operation efficiency of the model; In the process of regression, regression was carried out based on ROI local features.

After the regression, positive and negative samples of local features were judged, which greatly improved the operating efficiency and detection accuracy of the model. In the process of classification, a lighter weight was used to obtain high discriminative features, to achieve the binary classification of green target fruit, and to improve the detection accuracy and robustness of the model. The proposed model was trained and validated on the immature persimmon and apple datasets, and further analysis and comparison were made by ablation experiments. The experimental results showed that the proposed model performed better in average recall rate and average precision compared with classical object detection models. The model has achieved a good verification on persimmon datasets, but still has a certain space for development. It can be summarized as follows:

1) The experimental dataset of this model is relatively small and larger dataset can be used for training and verification in the future;

2) Although this model is an object detection model, it may be used as an Instance Segmentation model by introducing a mask when judging positive and negative samples of local features.

Acknowledgements

This work was financially supported by the Natural Science Foundation of Shandong Province in China (Grant No. ZR2020MF076); the Focus on Research and Development Plan in Shandong Province (Grant No. 2019GNC106115); the National Nature Science Foundation of China (Grant No. 62072289); the Shandong Province Higher Educational Science and Technology Program (Grant No. J18KA308); the Taishan Scholar Program of Shandong Province of China.

[References]

[1] Jia W, Zhang Y, Lian J, Zheng Y J, Zhao D A, Li C J. Apple harvesting robot under information technology: A review. International Journal of Advanced Robotic Systems, 2020; 17(3): 1729881420925310. doi: 10.1177/179881420925310.

[2] Xiong Y, Ge Y, Grimstad L, From P J. An autonomous strawberry-harvesting robot: Design, development, integration, and field evaluation. Journal of Field Robotics, 2020; 37(2): 202–224.

[3] Tang Y C, Chen M Y, Wang C L, Luo L F, Li J H, Lian G P, et al. Recognition and localization methods for vision–based fruit picking robots: A review. Frontiers in Plant Science, 2020; 11: 510. doi: 10.3389/fpls.2020.00510.

[4] Fu L S, Gao F F, Wu J Z, Li R, Karkee M, Zhang Q. Application of consumer RGB-D cameras for fruit detection and localization in field: A critical review. Computers and Electronics in Agriculture, 2020; 177: 105687. doi: 10.1016/j.compag.2020.105687.

[5] Ilea D E, Whelan P F. Image segmentation based on the integration of colour–texture descriptors-A review. Pattern Recognition, 2011; 44(10-11): 2479–2501.

[6] Sharma P, Suji J. A review on image segmentation with its clustering techniques. International Journal of Signal Processing, Image Processing and Pattern Recognition, 2016; 9(5): 209–218.

[7] Jia W K, Zheng Y J, Zhao D A, Yin Xiang, Liu X Y, Du R C. Preprocessing method of night vision image application in apple harvesting robot. Int J Agric & Biol Eng, 2018; 11(2): 158–163.

[8] Arefi A, Motlagh A M, Mollazade K, Teimourlou R F. Recognition and localization of ripen tomato based on machine vision. Australian Journal of Crop Science, 2011; 5(10): 1144–1149.

[9] Linker R, Cohen O, Naor A. Determination of the number of green apples in RGB images recorded in orchards. Computers and Electronics in Agriculture, 2012; 81: 45–57.

[10] Liao W, Zheng L H, Li M Z, Sun H, Yang W. Green apple recognition in natural illuminations based on random forest algorithm. Transactions of the Chinese Society for Agricultural Machinery, 2017; 48(S1): 86–91. (in Chinese)

[11] Tian Y Y, Duan H C, Luo R, Zhang Y, Jia W K, Lian J, et al. Fast recognition and location of target fruit based on depth information. IEEE Access, 2019; 7: 170553–170563.

[12] Li B R, Long Y, Song H B. Detection of green apples in natural scenes based on saliency theory and Gaussian curve fitting. Int J Agric & Biol Eng, 2018; 11(1): 192–198.

[13] Koirala A, Walsh K B, Wang Z L, McCarthy C. Deep learning-method overview and review of use for fruit detection and yield estimation. Computers and Electronics in Agriculture, 2019; 162: 219–234.

[14] Boogaard F P, Rongen K H, Kootstra G W. Robust node detection and tracking in fruit-vegetable crops using deep learning and multi-view imaging. Biosystems Engineering, 2020; 192: 117–132.

[15] Bargoti S, Underwood J P. Image segmentation for fruit detection and yield estimation in apple orchards. Journal of Field Robotics, 2017; 34(6): 1039–1060.

[16] Kang H, Chen C. Fruit detection, segmentation and 3D visualisation of environments in apple orchards. Computers and Electronics in Agriculture, 2020; 171: 105302. doi: 10.1016/j.compag.2020.105302.

[17] Jia W K, Tian Y Y, Luo R, Zhang Z H, Lian J, Zheng Y J. Detection and segmentation of overlapped fruits based on optimized mask R-CNN application in apple harvesting robot. Computers and Electronics in Agriculture, 2020; 172: 105380. doi: 10.1016/j.compag.2020.105380.

[18] Wang D, He D. Recognition of apple targets before fruits thinning by robot based on R-FCN deep convolution neural network. Transactions of the CSAE, 2019; 35(3): 156–163. (in Chinese)

[19] Sandler M, Howard A, Zhu M L, Zhmoginov A, Chen L C. MobilenetV2: Inverted residuals and linear bottlenecks. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Lake City, USA: IEEE, 2018; pp.4510–4520. doi: 10.1109/CVPR.2018.00474.

[20] Lin T Y, Dollár P, Girshick R, He B. Feature pyramid networks for object detection. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, USA: IEEE, 2017; pp.935–944. doi: 10.1109/CVPR.207.106.

[21] Girshick R. Fast R-CNN. In: 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile: IEEE, 2015; pp.1440–1448.

[22] Cao J, Cholakkal H, Anwer R M, Khan F S, Pang Y, Shao L. D2Det: Towards high quality object detection and instance segmentation. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020; pp.11485–11494. doi: 10.1109/CVPR42600.2020.01150.

[23] Howard A, Sandler M, Chu G, Chen L C, Chen B, Tan M X, et al. Searching for MobilenetV3. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, IEEE, 2019; pp.1314–1324. arXiv: 1905.02244v1.

[24] Zhang Y Q, Chu J, Leng L, Miao J. Mask-refined R-CNN: A network for

refining object details in instance segmentation. Sensors, 2020; 20(4): 1010. doi: 10.3390/s20041010.

[25] Zhang K, Sun M, Han T X, Yuan X F, Guo L R, Liu T. Residual networks of residual networks: Multilevel residual networks. IEEE Transactions on Circuits and Systems for Video Technology, 2017; 28(6): 1303–1314.

[26] Ren S Q, He K M, Girshick R, Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016; 39(6): 1137–1149.

[27] Tian Z, Shen C H, Chen H, He T. FCOS: Fully convolutional one-stage object detection. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 2019; pp.9627–9636. doi: 10.1109/ICCV.2019.00972.

[28] Dai J F, Qi H Z, Xiong Y W, Li Y, Zhang G D, Hu H, et al. Deformable convolutional networks. In: 2017 IEEE International Conference on Computer Vision (ICCV), 2017; pp.764–773. doi: 10.1109/ICCV.2017.89.

[29] He K M, Gkioxari G, Dollár P, Girshick R. Mask R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision. 2017; pp.2961–2969.

[30] Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C Y. SSD: Single shot multibox detector. In: Computer Vision-ECCV 2016. Lecture Notes in Computer Science, Springer, Cham, 2016; 9905: 21–37. doi: 10.1007/978-3-319-46448-0_2.