IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 7 June 2022, accepted 25 August 2022, date of publication 31 August 2022, date of current version 12 September 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3203106

RESEARCH ARTICLE

Identification and Depth Localization of Clustered Pod Pepper Based on Improved Faster R-CNN

SHIHAO ZHONG, WEIPING XU, TAIHUA ZHANG, AND HUAWEI CHEN
School of Mechanical and Electrical Engineering, Guizhou Normal University, Guiyang 550025, China

Corresponding author: Huawei Chen (huaweichen2004@yahoo.com)

This work was supported in part by the National Science Foundation of China under Award 72061006 and Award 71761007.

ABSTRACT Traditionally height of end effector of pod pepper harvester is fixed, which induces it hardly adapt to growth height of clustered peppers. Firstly, aiming at the problems of small size and clustered growth of pepper fruits during identification task, an improved Faster R-CNN algorithm is proposed. On the one hand, strategies such as increasing the types and number of high-resolution anchors and using RoI Align instead of RoI Pooling are employed to improve the detection accuracy for tiny targets. On the other hand, ResNet+FPN instead of VGG16 and ResNet backbone structure is adopted as the low-level feature extractor, so extracting capability for small features can be enhanced effectively. Furthermore, to precisely locate the position of clustered peppers, a height calculation model combining the 2D image recognition results with its depth information is advanced. Comparative experiments show that the overall accuracy AP and AP_{50} of our method reach 75.79% and 87.30%, respectively. Compared with VGG16 feature extraction model, the two indicators are improved by 8.7% and 1.3%, respectively. The small target detection accuracy AP^{small} is increased about 11.4%, with recall rate AR^{small} increased up to 10.2%. The overall loss rate Loss is reduced by 4.7%, which manifests greatly improvement compared to YOLOv3 model. The detection time of a single frame reaches 42ms, which is slightly longer than that of YOLOv3 network, but it can still meet the real-time detection requirements of pepper harvester. In 3D location experiment, the average absolute height error of clustered peppers from the ground is 4.4mm, that accounts to the relative average error of 1.1%, thus suffices the adjustment error requirement of the end effector.

INDEX TERMS Clustered pop pepper, depth information location, improved faster R-CNN network, object identification.

I. INTRODUCTION

Traditional clustered pod pepper harvesters equipped with drum-tooth-type picking heads can only carry out "one size fits all" operations, which always introduces more manual operation errors and thus induces substantial mechanical damage to fruits as well as high loss rate during pepper harvesting [1].

In recent years, with the continuous development of machine vision, image recognition technology is being widely integrated with agricultural machinery, marking a new application direction in agricultural automation and intelligence. At present, many scholars have applied image recognition technology to the agricultural field and achieved bountiful of excellent results. Wang [4] employed YOLOv4 based network and channel paper-cutting algorithm to greatly reduce the amount of calculation during impurity removing application for potatoes. Its detection accuracy rate was up to 91.43%. He *et al.* [5] combined multi-convolutional neural network with DXNet model to categorize apple fruits according to their external quality, and reached 97.84% classification accuracy rate. To identify kiwi fruit in a complex growth environment, Mu *et al.* [6] employed an improved AlexNet network and gained accuracy rate of 96.00%. Wang *et al.* [7] marked four varieties of kiwifruit, and adopted transfer learning method on DensNet121 network. their final recognization rate went up to 97.79%. In terms of pepper feature detection, Yang [8] introduced CNN network model to identify and

The associate editor coordinating the review of this manuscript and approving it for publication was Sabah Mohammed.

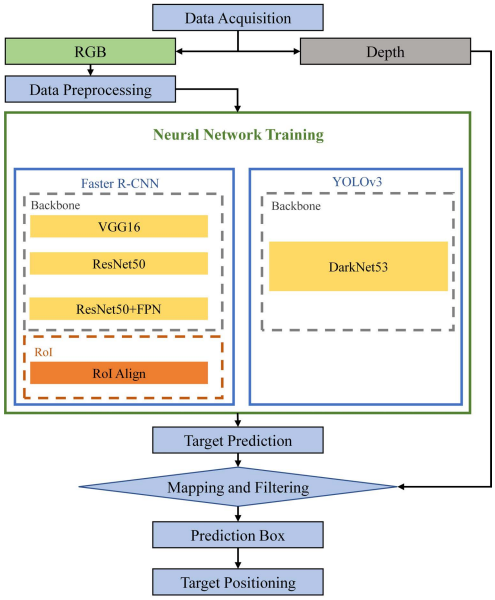


FIGURE 1. Flowchart of our method.

classify the defects of millet peppers, which achieved 93.13% and 98.76% recognition accuracy for both millet peppers with and without defects, respectively. To detect whether the fruit was damaged during the process of mechanized pepper-cap removal, Huynh *et al.* [9] obtained 95.2% recognition accuracy. Loti and Noor [10] comprehensively adopted machine learning and deep learning methods to analyze diseases and insect pests of red pepper and bird's-eye pepper. Their recognition accuracy reached 92.10%. It can be seen that the application of machine vision in the agricultural field is gradually becoming mature.

The clustered pod pepper is a variety of chili peppers. With uneven plant height and maturation period, and fruits mainly topping on branches, together with characteristics of small size and dense cluster, fruits of pod pepper are often difficult to be labelled and identified. In this paper, an improved Faster R-CNN target detection algorithm is proposed, in which different low-level feature extraction backbone networks, including VGG16, ResNet50 and ResNet50+FPN, will be compared horizontally, and optimization schemes are employed so that the detection accuracy of small targets can be obtained. Longitudinal comparison with YOLOv3 network, backboned DarkNet53, is carried out so effectiveness of the proposed method can be verified. Finally, combined with depth image information, spatial positioning calculation model and experiment are exerted, which will provide a location reference for automatic adjustment of the end effector of pepper harvester. The overall process is shown in Fig.1.

The main contributions of this paper are as follows:

1) Dataset of clustered pod pepper is constructed and manually annotated. To-tally 328 RGB-D (Red Green Blue-Depth) images of clustered pepper are acquired, and 3062 images are generated after expansion.

FIGURE 2. Sample data (RGB image on the left, Depth image on the right).

2) Improvement of recognition accuracy for small-sized cluster or individual fruit via hyperparameters and structure optimizing of Faster R-CNN network. ResNet50+FPN feature extraction layer, Anchor scale and quantity adjustment, and RoI Align sampling are comprehensively used to improve sampling accuracy and extraction ability of the network for small features.
3) A spatial height localization model is constructed, with combination of RGB-D depth image information. The calculated height is the crucial input parameter for automatic height adjustment of the end effector of pepper harvester.

II. MATERIALS AND METHODS

Our method will follow the pipeline of Fig.1.

A. DATA ACQUISITION AND PREPROCESSING

So far as we know, there hasn't one clustered pod pepper dataset publicly available on the Internet, so we have to construct the dataset from the beginning. The experimental images used in this paper were acquired from the pod pepper planting base in Baiyi Town, Wudang District, Guiyang, China. The collection time is 9:00 to 12:00 am on a sunny day on August 24, 2021 and a cloudy day on October 18, 2021. A total of 656 images of the clustered pepper were effectively captured by Intel RealSense D435i depth camera, including 328 color images and 328 depth images. The collected data samples are shown in Fig.2.

The acquired RGB images are manually labeled using LabelImg labeling software. During labelling, smallest circumscribed rectangle method for clustered pepper fruits is adopted. Annotations are saved in a file of XML format. All images are made in VOC format. In order to improve sample diversity for model training, it is necessary to enhance the collected data. In our experiment, the original data are processed by rotating, flipping, embossing, adding noise, color enhancement, and changing the grayscale and contrast of the picture. Hence the data volume of RGB images is 2406 pieces. Then rotate the original data by 15° and 30°, and finally we get 3062 RGB images. Depth images are synchronously processed. The experimental data preprocessing is shown in Fig.3. For model training, the training and validation data will be randomly allocated in a ratio of 9:1, that is, 2756 and 306 images in training set and validation set separately.

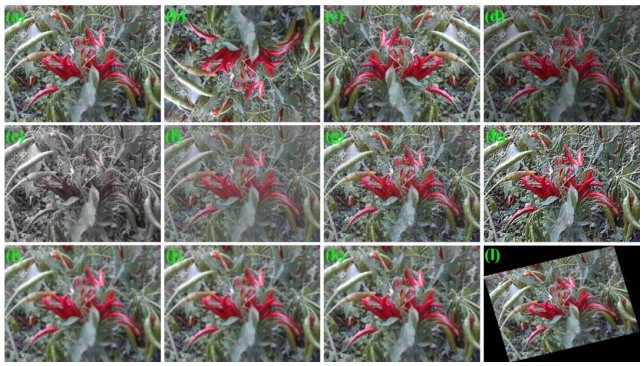


FIGURE 3. Data enhancing. ((a) Original image, (b) Flip up and down, (c) Flip left and right, (d) Brightness, (e) Grayscale, (f) Contrast, (g) Gaussian noising, (h) Emboss, (i) Gaussian blur, (j) Mean blur, (k) Median blur, (l) Rotation).

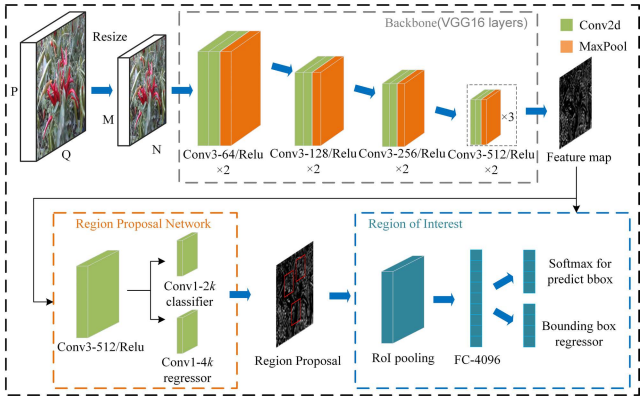


FIGURE 4. Schematic diagram of Faster R-CNN network.

B. FASTER R-CNN NETWORK OVERVIEW

The Faster R-CNN network was first proposed by Ren *et al.* [11] at the NIPS conference in 2015. The framework consists of three parts: feature extraction network (Backbone), Region Proposal Network (RPN) and Region of Interest pooling layer (RoI pooling). Synthesizing advantages of both R-CNN and Fast R-CNN, Faster R-CNN framework inserts a region proposal network (RPN) into the original Fast R-CNN network. Discarding traditional SS (selective search) and other candidate selection methods, RPN can greatly reduce the amount of computation and effectively reduce model training time. Faster R-CNN framework is shown in Fig.4.

Original images will turn into Feature Maps after passing through the feature extraction network, which share convolutional features with RPN and RoI Pooling layers. The RPN layer is the main highlights in Faster R-CNN network (Fig.5). In order to predict objects of different shapes and sizes during convolution operation, each time the convolution kernel slides, an anchor point will be generated in the center of the convolution kernel. Finally, the network model comprehensively corrects information from features, 2k classification and 4k bounding box regression to achieve the target of accurate object detectation.

Faster R-CNN network includes two loss functions, RPN loss and Fast R-CNN loss. In the RPN network, its loss

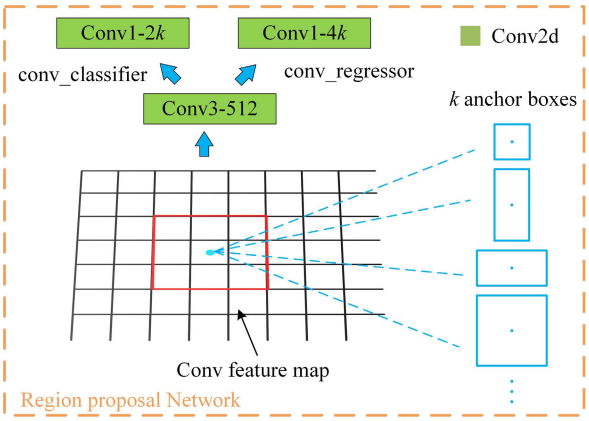


FIGURE 5. RPN sketch.

function [12] is:

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

where N_{cls} represents the number of all samples in a mini-batch, N_{reg} represents the number of anchor positions (the total number of anchors generated on the feature map), λ is the weight factor of the two losses.

L_{cls} denotes multi-class cross entropy loss (SoftMax Cross Entropy) for classification task, which is defined as:

$$L_{cls}\left(p_i,p_i^*\right)=-log\left(p_i\right) \tag{2}$$

where p_i represents the probability that the i-th anchor is predicted to be a true label; p_i^* represents 1 when the sample is a positive sample, and 0 otherwise.

For bounding box regression task, its loss function is same as Fast R-CNN:

$$L_{\text{reg}}\left(t_i,t_i^*\right)=R\left(t_i-t_i^*\right) \tag{3}$$

where t_i represents the predicted bounding box parameter corresponding to the i-th anchor; t_i^* represents its related GT value. R represents the $Smooth_{\text{L1}}$ loss function, which is defined as:

$$\text{Smooth}_{\text{L1}}\left(x\right)=\begin{cases}0.5x^2 & |x|<1\\ |x|-0.5 & \text{other}\end{cases} \tag{4}$$

C. MODEL OPTIMIZATION

1) BACKBONE NETWORK SUBSTITUTION

The initial feature extraction network of Faster R-CNN model is VGG16 convolutional neural network. A large number of experiments have shown that with the deepening of the number of network layers, convolutional neural network will not only cause training results to decline, but also lead to the problem of gradient explosion or gradient vanishing. In this regard, we introduce ResNet50+FPN network model as the backbone feature extraction network (Fig.6). ResNet50 is a residual network that provides an effective solution for gradient vanishing. Traditional Faster R-CNN network contains

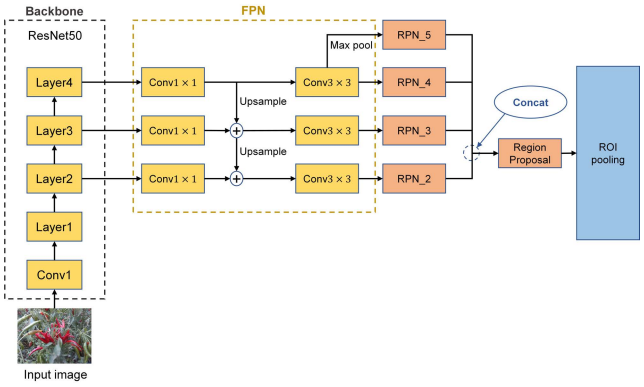


FIGURE 6. ResNet50+FPN network structure (Conv1-Conv5 are five layers in the Resnet50 network, five Conv1 × 1 modules were used to adjust the numbers of channels in different feature layers, "+" means feature information fusion by add method, four Conv3 × 3 modules are used for further feature information extraction).

only one feature layer after 16 times of downsampling, while FPN combines high-level semantic features of the network with underlying detail features. FPN has the ability of predicting candidate frames in multiple feature scope, so it is easier for the entire network to gather feature information of the target object and thus enhance the utilization of image features. In experiments, we choose three layers {layer2, layer3, layer4} in ResNet50 to participate in network weight training, and selects VGG16 and ResNet50 network models for horizontal comparison.

2) ANCHOR RESOLUTION INCREASING

Pod-pepper fruits are small-sized and densely growing. Implementation of Faster R-CNN without any alteration to our database can only result in poor detection accuracy, especially for individual and small clustered pepper fruits [13]. Our solution is to enlarge anchor scales to improve feature resolutions while keeping its original ratios (1:1, 1:2, 2:1) unchanged. In our experiment, anchor size is adjusted from [128^2, 256^2, 512^2] (Marked as Anchor-3) to [32^2, 64^2, 128^2, 256^2, 512^2] (Marked as Anchor-5).

3) ROI POOLING SCHEME ALTERATION

After RPN layers, the obtained regression hyper-parameters are float data. Then RoI Pooling layer is responsible for mapping candidate proposals to fix-sized Feature Map, after two quantization operations:

1) When one candidate proposal is mapped to the shared feature layer, its float coordinates are rounded up.
2) When one boundary area is divided into $k \times k$ units (bins) on average, float coordinates of unit corners are also rounded into integers.

The two quantization operations have changed the initial prediction range of the candidate frame, and will bring large deviations when abstracting small features, thus resulting in decrease of accuracy on tiny objects.

RoI Align method (Fig.7) proposed in Mask R-CNN [14] is an effective scheme to avoid data rounding up. RoI Align

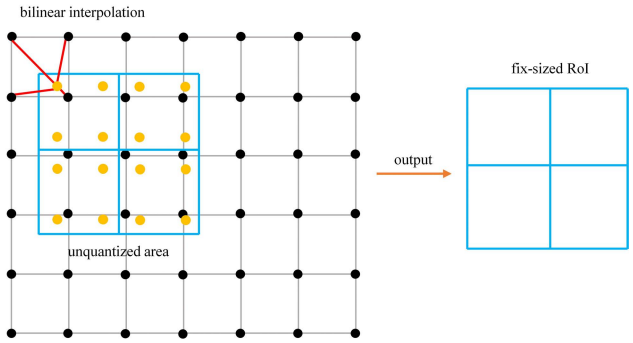


FIGURE 7. RoI Align sampling (Note: the black dots and yellow dots denotes *x* and *x* respectively, the black dot indicates each pixel, the yellow dot indicates the sampling points evenly divided in the non-quantized area, and the blue box indicates the non-quantized area).

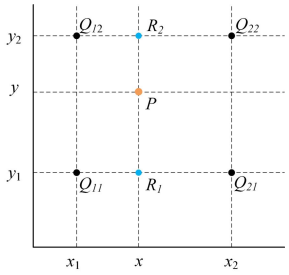


FIGURE 8. Bilinear interpolation in one subregion.

utilizes bilinear interpolation in each sub-cell to calculate output value of each sampling point, and output the maximum value in sub-region for fixed-sized RoI via Max pooling method, so it will not round up floating-point coordinates of candidate proposals and divided units.

Fig.8 details the bilinear interpolation method in one subregion. P is the coordinate point obtained by the model through the regression parameters. Q_{11}, Q_{12}, Q_{21}, and Q_{22} are the four points of the cell where P is located, and their coordinate values and pixel values are all parameters. First, the coordinates and pixel values of R_1 and R_2 are obtained by the first interpolation method, and then the coordinates and pixel values of point P are obtained by the second interpolation method. The calculation method is shown in formula (5)(6)(7).

$$R_1: f(x,y_1) \approx \frac{x_2-x}{x_2-x_1} f(Q_{11}) + \frac{x-x_1}{x_2-x_1} f(Q_{21}) \quad (5)$$

$$R_2: f(x,y_2) \approx \frac{x_2-x}{x_2-x_1} f(Q_{12}) + \frac{x-x_1}{x_2-x_1} f(Q_{22}) \quad (6)$$

$$P: f(x,y) \approx \frac{y_2-y}{y_2-y_1} f(x,y_1) + \frac{y-y_1}{y_2-y_1} f(x,y_2) \quad (7)$$

In the formula above, (x_1, y_1) and (x_2, y_2) are coordinate values of Q_{11} and Q_{22} respectively, and Q_{11}, Q_{12}, Q_{21}, Q_{22} are pixel values of four points.

D. OBJECT DETECTION - EXPERIMENTAL METHODS

1) COMPUTATIONAL DEVICE

In our experiment, Intel RealSense D435i depth camera was used for image shooting, which is capable of capturing both

TABLE 1. Computation server configurations.

Items	Info.
Operating system	Ubuntu20.04.4 LTS
CPU	Intel (R) Core (TM) i9-9900K * 8
Graphics card	NVIDIA GeForce RTX 2080Ti * 4 (GPU: 11G * 4 = 44G)
Programming language	Python3.8
Deep learning framework	Pytorch1.11.0

TABLE 2. Hyperparameter settings.

Parameter name	Value
Batch size	24
Epochs	200
Input shape	[640, 480]
Optimizer	SGD
Momentum	0.9
Weight decay	0.0001
NMS	0.3
Confidence threshold	0.5
IoU	0.7
Learning rate	0.001
Gamma	0.1/50epochs

RGB flows and Depth flows, and aligning these two flows for further image processing. Camera parameters include: resolution sizes of 640 × 480 for both color and depth map, frame rate 60 fps/s, and depth range in 0.3m-3m. The experiment was carried out on a computation server, as configured in Table 1. In order to further exert the computing power of this server, CUDA11.0 and cuDNN8.0.5 are installed for GPU computation. And also, a parallel computing strategy is implemented to balance memory load of multiple GPU processors for model-training acceleration.

2) COMPARATIVE MODEL

YOLOv3 network is a regression based one-stage object detection network. Compared with the Faster R-CNN, it generates candidate frames during prediction stage, and directly performs regression decoding on parameters of predicted frames, so tasks of object detection and classification could be satisfied at the same time that overall training cost is saved enormously. YOLOv3 is known for fast detection speed, as well as ability on feature extraction, so we choose YOLOv3 as the longitudinal comparison network.

3) HYPERPARAMETERS

The two deep learning networks Faster R-CNN and YOLOv3 used in the experiment run on the same server. In order to avoid interference of other factors on experimental results, training hyperparameters of both models are samely set (Table 2), including batch size, epochs, image size (input shape), optimizer, momentum, the regularization weight decay rate (weight decay), non-maximum suppression (NMS), confidential threshold, IoU, learning rate, and learning rate adjustment multiplier (gamma).

4) EVALUATION INDICATORS

COCO evaluation standard is used to performance evaluation, which includes indicators of *Precision*, Recall (recall rate), AP (average precision) and mAP (mean Average Precision):

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{8}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{9}$$

$$\text{AP} = \int_0^1 \text{P}(r)\text{d}r \tag{10}$$

$$\text{mAP} = \frac{\sum_{q=1}^{M} \text{AP}(q)}{\text{M}} \tag{11}$$

where TP (true positives), FP (false positives) and FN (false negatives) represent correctly classified positive samples, falsely classified positive samples and, falsely classified negative samples; AP refers to curve area covered by all Precision and Recall points in two-dimensional coordinate system; mAP is the average of APs of all classes; M denotes the total number of all categories.

We select AP, AP_{50}, AP^{small}, and AR^{small} as the exact indicators, of which the superscript small denotes indicator for tiny targets. Among them, AP is the mAP average value calculated from 10 IoU thresholds from 0.50 to 0.95 with a proportional interval of 0.05, and AP_{50} is the mAP value when the IoU threshold is 0.5, AP^{small} and AR^{small} are average precision and recall rate on tiny targets.

5) TRANSFER LEARNING

The augmented 3062 image data cannot meet the needs of weight training from the very beginning, we adopt the idea of transfer learning. Pre-trained weights exposed by PASCAL VOC2012 dataset are utilize, to enable faster convergence during model training.

E. SPATIAL POSITIONING

After one image passes through depth network, a prediction box of target object will be generated. The midpoint of the prediction box is marked as its image position, which denotes the exact locale of the targeted object in the 2D RGB image. However, in the world coordinate system, 3D coordinate of the target still needs to be determined through physical quantitative methods such as distance or depth measures, which the stored Depth image captured by depth camera can provide directly. RGB and Depth images have pixel correspondence. We can easily convert between pixel coordinates $\{u, v\}$ and world coordinates $\{X, Y, Z\}$ based on mapped RGB and Depth images. The conversion is expressed by formula (12).

$$Z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K\,[R,t] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{12}$$

where K is internal parameter matrix and $[R, t]$ external parameter matrix of the selected depth camera.

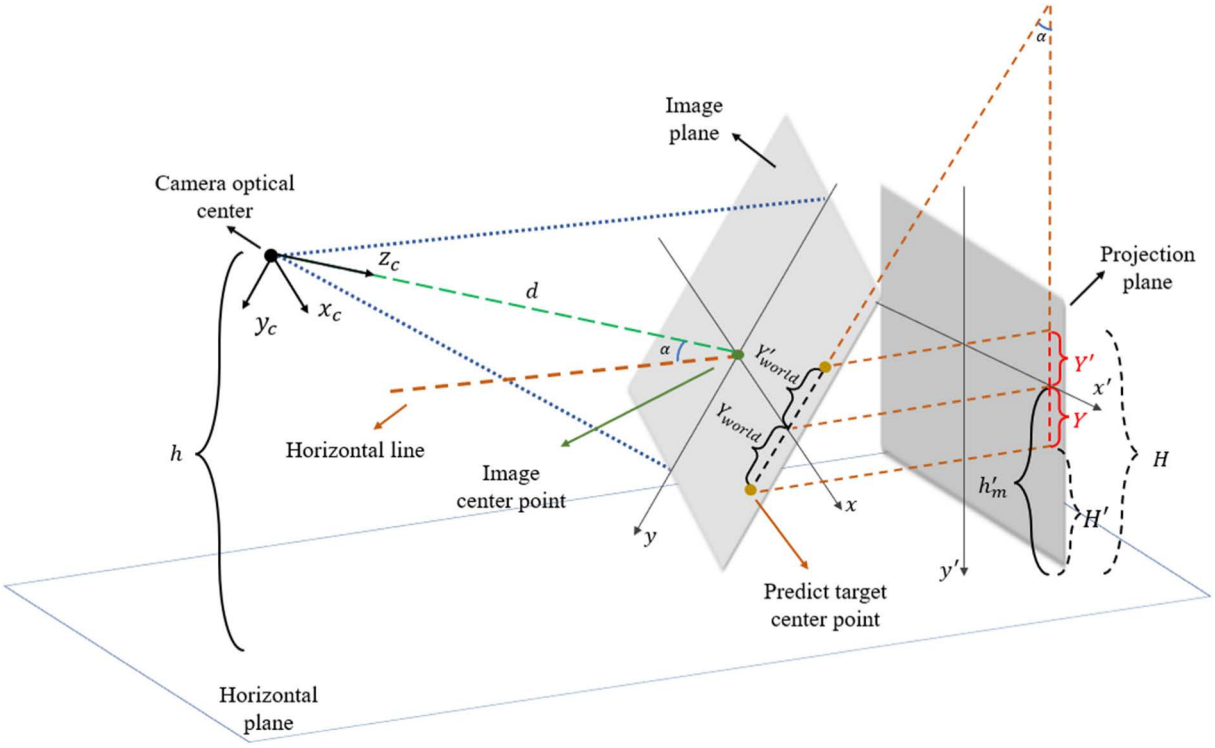


FIGURE 9. Schematic diagram of depth camera geometry.

In height positioning experiment, depth camera is fixed on a plane with known height. Let calibrated camera coordinate system $\{x_c, y_c, z_c\}$ coincide with world coordinate system $\{X_{world}, Y_{world}, Z_{world}\}$ (Fig.9). When the camera is placed horizontally ($\alpha = 0$), image plane will coincide with projection plane, and height value h of the camera optical center above the ground is equal to h'_m, central height of the plane where the predicted target located. In this case, the calculated Y_{world} value is a relative coordinate between the predicted target point and the image center point along Y axis of the image plane ($Y_{world} = Y$). When the camera is placed at an angle to the horizontal plane ($\alpha \neq 0$), image plane will incline an angle α with projection plane of the target in 3D space. By solving the triangle, we can get formula (13):

$$H = \begin{cases} (h - d*sin\alpha) \pm Y_{world} * \cos\alpha & \alpha \neq 0 \\ h \pm Y_{world} & \alpha = 0 \end{cases} \quad (13)$$

In formula (13), α is angle between central axis of camera and horizontal plane, d is depth value of the predicted target point, h is ground clearance height of camera optical center, and H is the actual ground clearance height of the target object.

III. MATERIALS AND METHODS

A. PEPPER FRUIT DETECTION

We conducted three groups of comparative experiments: experiments 1 to 3. In experiment 1, we fixed the value of *Anchor* and compared the overall impact of different ROI pooling operations on Faster R-CNN network. In experiment 2, we compared the influence of different *Anchor* values on the whole network. In experiment of 3, we selected Faster R-CNN and YOLOv3 network to implement horizontal and vertical experiments, so the network more suitable for recognizing clustered pod-peppers can be identified.

TABLE 3. ROI pooling contrast.

ROI	Backbone	AP	AP_{50}	AP^{small}	Upgrade
RoI Pool	VGG16	0.4574	0.7991	0.4291	-
	ResNet50	0.4733	0.7972	0.4063	-
RoI Align	VGG16	0.6552	0.8124	0.5408	+11.17
	ResNet50	0.6276	0.7887	0.5061	+9.98

1) EXPERIMENT 1: ROI POOLING CONTRAST

In [12], [15], and [16], optimization strategy of RoI Align instead of RoI Pool is adopted to increase detection accuracy of small target objects. In [17], this method is claimed to improve detection ability of for small targets of industrial aluminum profiles by 17%. In our experiment, anchor size is fixed to Anchor-3, and Faster R-CNN networks with backbone VGG16 and ResNet50 are compared. The experimental results are shown in Table 3.

It can be seen from Table 3 that RoI Align gains better performance than RoI Pooling. For small objects, the average accuracy AP^{small} has improved by 11.17% and 9.98% over

TABLE 4. The overall results of the network after anchor take different values.

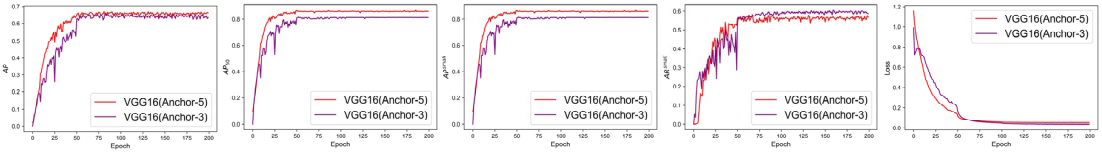
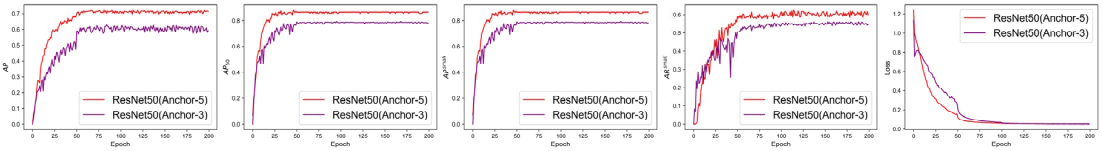
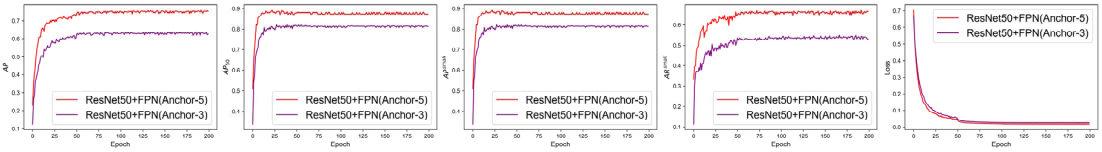
Faster R-CNN	Results				
Backbone	AP	AP_{50}	AP^{small}	AR^{small}	Loss
VGG16					
ResNet50					
ResNet50+FPN					

TABLE 5. Evaluation metrics of different models.

Model	Anchor	mAP			mAR	Loss	Speed	Param.
		AP	AP_{50}	AP^{small}	AR^{small}		(s/frame)	(M)
Faster R-CNN(VGG16)	Anchor-3	0.6552	0.8124	0.5408	0.5857	0.0554	0.051	43.95
Faster R-CNN(ResNet50)		0.6276	0.7887	0.5061	0.5475	0.0785	0.029	70.57
Faster R-CNN(ResNet50+FPN)		**0.6369**	**0.8156**	**0.4601**	**0.5413**	**0.0256**	**0.044**	**41.33**
Faster R-CNN(VGG16)	Anchor-5	0.6705	0.8601	0.5213	0.5691	0.0649	0.056	44.02
Faster R-CNN(ResNet50)		0.7204	0.8653	0.5783	0.6144	0.0753	0.029	70.64
Faster R-CNN(ResNet50+FPN)		**0.7579**	**0.8730**	**0.6351**	**0.6711**	**0.0179**	**0.042**	**41.40**
YOLOv3	-	0.4874	0.8033	0.2953	0.4955	3.8381	0.013	62.57

VGG16 and ResNet50 backbone respectively. Thus, in our application of clustered pod-peppers detection, RoI Align is chose for ROI pooling.

2) EXPERIMENT 2: ANCHOR COMPARISON

In order to verify the influence of different Anchor values on the evaluation parameters of the network. In this experiment, we fixed RoI Align as the pooling layer, and compared Faster R-CNN network with backbones of VGG16, ResNet50 and ResNet50+ FPN. The experimental results are shown in Table 4.

In Table 4, the performance of each network on Anchor-5 is better than that of Anchor-3. When backbone network of VGG16 and ResNet50 is trained to the 50th epoch, parameter curves fluctuate greatly due to adjustment of learning rate. Comparatively, our model which is backboned of ResNet50+FPN has lower oscillation amplitude and stronger robustness.

3) EXPERIMENT 3: HORIZONTAL AND VERTICAL COMPARISON

In this experiment, we listed the detailed results after training of each network under different conditions, and verified the effectiveness of the method in this paper through experimental comparison. Based on the maximum AP value, the evaluation index values corresponding to each network are shown in Table 5.

In horizontal comparison (Table 5), some evaluation indexes when taking Anchor-3 are lower than the results of other networks, but with Anchor-5, overall indexes display significant improvement. Among the results with Anchor-5, AP threshold and AP_{50} are increased by 8.7% and 1.3% respectively, compared with the original VGG16 backbone network. The indexes for small target AP^{small} and AR^{small} are increased about 11.4% and 10.2% separately. The overall loss rate Loss is reduced by 4.7%.

Fig.10 shows the longitudinal comparison results of Faster R-CNN and YOLOv3 when Faster R-CNN networks takes Anchor-5. Fig.10 (a-d) compares evaluation indicators of AP, AP_{50}, AP^{small}, and AR^{small}. In the showed epoch scopes (200 epochs), YOLOv3 is still climbing toward stabilization. By increasing the number of training epochs in the YOLOv3 network, YOLOv3 network parameters still have room for improvement. It obviously suggests that all Faster R-CNN based network models achieve best fitting effect after 75 epochs, and the ResNet50+FPN backboned Faster R-CNN converges faster and is more robust than other networks. Fig.10 (e)

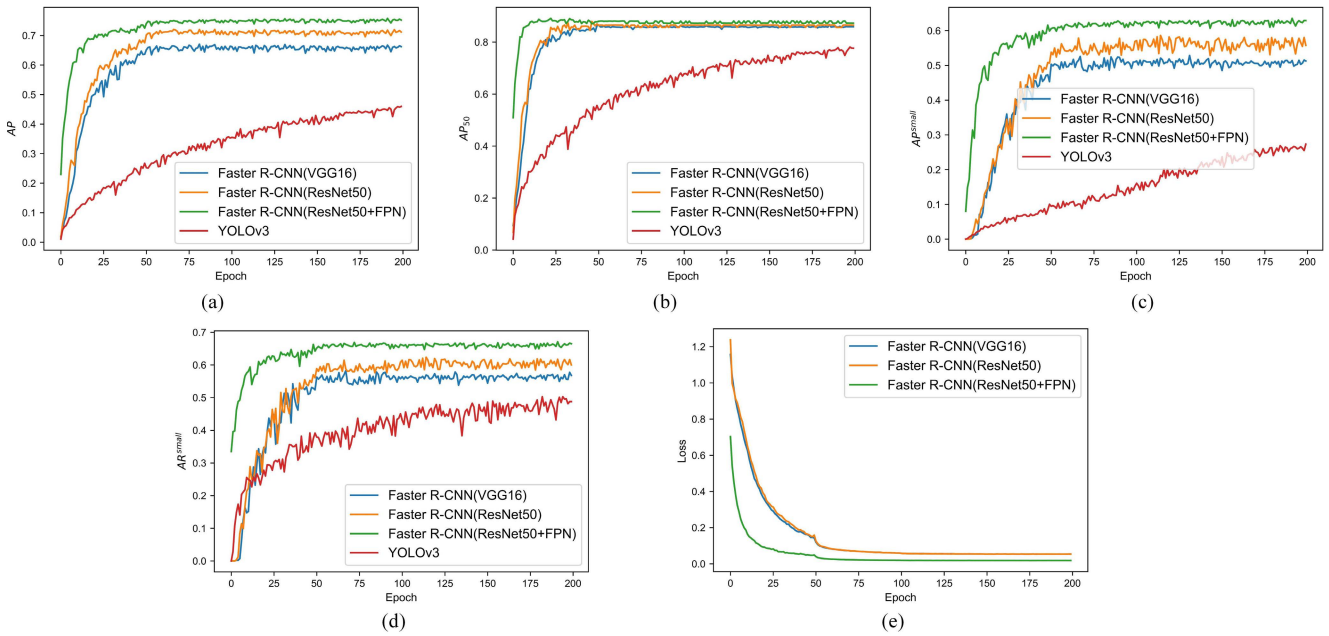


FIGURE 10. Schematic diagram of depth camera geometry. (a) AP longitudinal comparison chart. (b) AP_{50} longitudinal comparison chart. (c) AP^{small} longitudinal comparison chart. (d) AR^{small} longitudinal comparison chart. (e) Loss widthwise comparison chart.

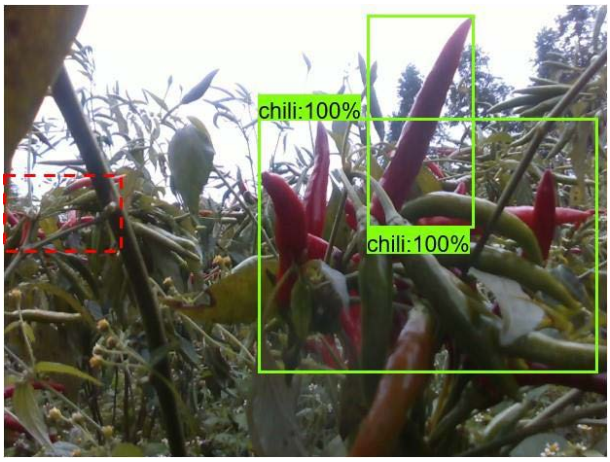


FIGURE 11. Model recognition failure example.

shows change of Loss rate of different feature extraction networks.

In real-time detection of farmland obstacles, real-time detection during tractors operating at 2-14 km/h can be satisfied when average detection speed of a single image amounts 530ms [18]. In the longitudinal comparison, our method reaches frame processing speed of 42ms. Compared with the YOLOv3 network, although the detection time is slightly longer, it still adapts real-time detection requirements of the existing pepper harvesters at driving speed of 1.8-8 km/h [1], which verifies the real-time performance and effectiveness of our proposed method.

And, we also encounter some failure cases. Fig.11, presents two typical cases: a missed target on the left (the red dotted box), and multiple frames for a single target on the right (the two green solid boxes). The reason for these two failure cases is different. For the first one, the targeted pepper-fruit cluster is blocked into background by a pepper stem. But for the second case, it is mainly caused by differential exposure to different branches of one fruit cluster, due to its scattered growth characteristic.

B. HEIGHT CALCULATION

The shooting range of the D435i depth camera is about 0.3m-3m. The following randomly selects 5 groups of RGB-D images at different depth detection points for comparative experiments, as shown in Table 6. According to our measurement, when the camera is positioned within 0.5m of the object, depth value in some areas cannot be recorded, resulting in invalid depth values. When shooting in the range of 2.5m-3.0m, height estimation error of predicted objects will be over 2%. Therefore, relatively accurately shot pixels by D435i are between depth range of 0.5m and 2.5m.

Thus, we set depth range of 0.5m-2.5m as the filtering condition for predicted proposals. Fig.12 shows the flowchart of our target filtering process. At the end of prediction network, multiple prediction boxes will be generated in RGB domain, then central-point coordinates of prediction boxes are mapped into their corresponding Depth image, and lastly qualified prediction boxes are selected according to the depth value filtering condition.

Among the four experimental groups that falls in the depth filtering range (Table 6), average absolute error is about 4.4mm, with relative error of 1.1%. [19] states that it is

TABLE 6. Experiments for spatial height estimation of detected targets.

Group	Depth range (/m)	Depth distance (/mm)	Actual height (/mm)	Estimated height (/mm)	Absolute error (/mm)	Relative error (/%)	Average error (/%)
1	0.5-1.0	944	255	259	+4	1.57	1.48
		945	291	295	+4	1.37	
		974	397	391	-6	1.51	
2	1.0-1.5	1094	379	376	-3	0.79	0.85
		1102	460	455	-5	1.08	
		1270	578	574	-4	0.69	
3	1.5-2.0	1500	417	422	+5	1.20	1.11
		1767	418	414	-4	0.96	
		1876	510	494	-6	1.18	
4	2.0-2.5	2156	530	526	-4	0.75	0.80
		2358	411	408	-3	0.73	
		2414	439	435	-4	0.91	
5	2.5-3.0	2646	419	407	-12	2.86	2.48
		2754	225	230	+5	2.22	
		3000	340	332	-8	2.35	

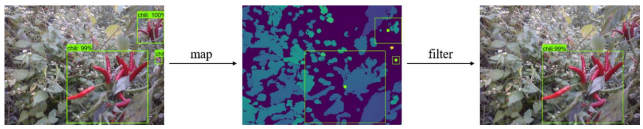


FIGURE 12. Prediction box filtering.

acceptable if height adjusting error for end effector of a harvester is less than 43mm. So, we are surely to conclude that our computation model satisfies height error requirement during adjustment of harvester end effector.

IV. DISCUSSION

There is still room for improvement in our experiments.

In object detection experiments, firstly, volume of images acquired in our dataset are insufficient due to limitation of experiment conditions. Data acquisition time and times are limited for that most clustered pod-peppers have short plucking period, with one growth season every year. And also, our training samples are too small, which will make convolutional neural network unable to completely capture characteristics and changes of objects [20]. Therefore, appropriate expanded data set will have better effect on improving the recognition accuracy of neural network. Secondly, the Intel RealSense D435i depth camera adopted has low accuracy and low resolution for image shooting on RGB and depth pairs. Although some blurred, distorted and incomplete original images are eliminated, the original images with target occlusion are not removed. The recognition and classification of low-resolution images has always been a challenging problem [21], but the detection model trained by high-resolution images usually cannot recognize or locate objects on low-resolution images [22]. Therefore, during network training, optimizing the target occlusion problem [23] and properly integrating higher-quality images is conducive to further improving the recognition accuracy. Finally, our method is three times slower than YOLOv3 network (Table 6), which is one of the main differences between one-stage network and two-stage network. Although our method meets the needs of real-time detection for pepper harvester, the lightweight network [24] has characteristics of less training parameters, fast detection speed, high precision and low demand for portable GPU, which can reduce the cost of algorithm landing and the complexity of equipment.

For height positioning model, there are systematic and computational model errors when estimating the height of clustered pod-peppers combined with depth information In case of our depth camera D435i, they include calibration error, 0-2% recording error for depth image, image distortion due to camera jitter, and estimation error incurred from formula (9). Scholars have attempted multiple high-precision-camera assisted shooting [26] and high-precision calibration algorithm [27] to minimize three-dimensional positioning errors.

Additionally, planting standardization in the data acquisition site is incomplete. Standardized planting method can not only reduce damage rate of harvested fruits [28], but also effectively improve the accuracy of target detection network and height positioning model. It is also possible to integrate detection and positioning of clustered pod-pepper fruits together. [29] have introduced an end-to-end RGB-D fusion deep learning network, where both tasks of target recognition and localization can realize at the same time.

V. CONCLUSION

In order to solve the problems of small target recognition and spatial localization for automatic height adjustment of end effector during pod pepper harvesting, which always characterized small-size, cluster growing and uneven growth, an improved Faster R-CNN deep learning network and deep information fusion model based on RGB-D image are proposed, so both tasks of 2D identification and 3D localization of pepper clusters are realized sequentially. Our main work includes:

1) Establishment of one experimental dataset on clustered pod-pepper. A total of 328 RGB and Depth images are collected, and then augmented into the amount of 3062. For model training, the dataset is separated into train and validation subsets according to the ratio of 9:1.
2) Improvement on Faster R-CNN network. By optimizing anchor resolution of the RPN layer and using ROI Align sampling, our model is more capable of extracting tiny features and targets. Resnet50+FPN, VGG16 and ResNet50 are selected as Faster R-CNN backbones for horizontal comparison, and the one-stage target detection network YOLOv3 for vertical comparison. The results show that AP and AP50 reach 75.79% and 87.30% respectively in our chosen model, the ResNet50+FPN based network, which takes 42*ms* for detectation of one single image. Our model shows higher recognition accuracy and better comprehensive performance than other models.
3) Construction of a height positioning model for proposed pepper cluster frame center. The average height estimation error is 1.1%, which meets error requirement for end effector height adjusting during pepper harvesting.

Above experimental data indicates that our proposed method meets requirements of real-time identification and height positioning for clustered pod-pepper harvesting. It could provide data input for lifting and lowering operations of harvester end effector, which is a crucial reference for its intelligent alteration design.

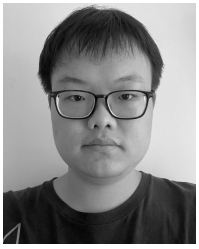
ACKNOWLEDGMENT

The authors would like to thank the laboratory team members, for their assistance during data collection. They would also like to thank Prof. Weiping Xu and Lin Shuyun of Guizhou Mountain Agricultural Machinery Research Institute for providing the experimental equipment. They would also like to thank Guizhou Normal University for supporting this project.

REFERENCES

[1] M. Zhang and D. Xu, "Design and test for the line pepper picking machine," *Mech. Res. App.*, vol. 32, no. 6, pp. 103–105, Dec. 2019.
[2] J. Su, C. Du, and J. Du, "Development and experiment of self-propelled capsicum harvester," *Agric. Eng.*, vol. 11, no. 6, pp. 17–19, Jun. 2021.
[3] S. Zhang, "Design of Moshine 4JZ-3600A self-propelled pepper harvester," *Xinjiang Agric. Mech.*, vol. 3, pp. 5–7, Jun. 2020.
[4] X. Y. Wang, "Real-time detection method of traffic information based on lightweight YOLOv4," *Trans. Chin. Soc. Agric. Mach.*, vol. 52, no. 8, pp. 241–247 and 262, Jun. 2021.
[5] J. R. He, S. Y. Xin, B. Liu, and D. J. He, "External quality grading method of Fuji apple based on deep learning," *Trans. Chin. Soc. Agric. Mach.*, vol. 52, no. 7, pp. 379–385, Apr. 2021.
[6] L. Mu, C. Gao, Y. Cui, K. Li, H. Liu, and L. Fu, "Kiwifruit detection of far-view and occluded fruit based on improved AlexNet," *Trans. Chin. Soc. Agric. Mach.*, vol. 50, no. 10, pp. 24–34, Aug. 2019.
[7] Q. Wang, L. Zhao, and Z. Niu, "Portable kiwi variety classification equipment based on transfer learning," *J. Phys., Conf. Ser.*, vol. 1865, no. 4, pp. 442–469, Apr. 2021.
[8] C. Yang, "Design of automatic detection system for millet pepper," M.S. thesis, Dept. Electron. Eng., Wuhan Polytech Univ., Wuhan, China, 2019.
[9] Q.-K. Huynh, C.-N. Nguyen, H.-P. Vo-Nguyen, P. L. Tran-Nguyen, P.-H. Le, D.-K.-L. Le, and V.-C. Nguyen, "Crack identification on the fresh chilli (Capsicum) fruit destemmed system," *J. Sensors*, vol. 2021, pp. 1–10, Feb. 2021.
[10] N. N. A. Loti, M. R. M. Noor, and S. Chang, "Integrated analysis of machine learning and deep learning in chili pest and disease identification," *J. Sci. Food Agricult.*, vol. 101, no. 9, pp. 3582–3594, Jul. 2021.
[11] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.
[12] G. Li, J. Su, and Y. Li, "An aircraft detection algorithm in SAR image based on improved faster R-CNN," *J. Beijing. Univ. Aeronaut. Astronaut.*, vol. 47, no. 1, pp. 159–168, May 2021, doi: 10.13700/j.bh.1001-5965.2020.0004.
[13] L. Dong and J. Xu, "Flower bud recognition of pear tree based on improved faster R-CNN," *J. Hebei Agric. Univ.*, vol. 44, no. 6, pp. 116–121, Nov. 2021, doi: 10.13320/j.cnki.jauh.2021.0110.
[14] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 386–397, Feb. 2020.
[15] Q. Wen, Z. Luo, R. Chen, Y. Yang, and G. Li, "Deep learning approaches on defect detection in high resolution aerial images of insulators," *Sensors*, vol. 21, no. 4, p. 1033, Feb. 2021.
[16] J. Yan, "Recognition of *Rosa roxbunghii* in natural environment based on improved faster RCNN," *Trans. Chin. Soc. Agric. Eng.*, vol. 35, no. 18, pp. 143–150, Jun. 2019.
[17] K. Xiang, S. Li, and M. Luan, "Aluminum product surface defect detection method based on improved faster RCNN," *Chin. J. Sci. Instrum.*, vol. 42, no. 1, pp. 191–198, Jan. 2021, doi: 10.19650/j.cnki.cjsi.J2007109.
[18] J. Xue, Y. Li, and X. Cao, "Obstacle detection based on deep learning for blurred farmland images," *Trans. Chin. Soc. Agric. Mach.*, vol. 53, no. 3, pp. 225–233, Mar. 2022.
[19] X. Wei and M. Zhang, "Extraction of crop height and cut-edge information based on binocular vision," *Trans. Chin. Soc. Agric. Mach.*, vol. 53, no. 3, pp. 225–233, Jan. 2022.
[20] J. G. A. Barbedo, "Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification," *Comput. Electron. Agricult.*, vol. 153, pp. 46–53, Oct. 2018.
[21] J. Wen, Y. Shi, X. Zhou, and Y. Xue, "Crop disease classification on inadequate low-resolution target images," *Sensors*, vol. 20, no. 16, p. 4601, Aug. 2020.
[22] X. Zhao, W. Li, Y. Zhang, and Z. Feng, "Residual super-resolution single shot network for low-resolution object detection," *IEEE Access*, vol. 6, pp. 47780–47793, 2018.
[23] X. Wang, J. Liu, and G. Liu, "Diseases detection of occlusion and overlapping tomato leaves based on deep learning," *Frontiers Plant Sci.*, vol. 12, p. 2812, Dec. 2021.
[24] J. Chen, D. Zhang, and Y. A. Nanehkaran, "Identifying plant diseases using deep transfer learning and enhanced lightweight network," *Multimedia Tools Appl.*, vol. 79, nos. 41–42, pp. 31497–31515, Aug. 2020.
[25] J. Wu, Y. Men, and D. Chen, "Lightweight network and parallel computing for fast pedestrian detection," *Int. J. Circuit Theory Appl.*, vol. 49, no. 4, pp. 1040–1049, Nov. 2020.
[26] B. Fu, F. Han, Y. Wang, Y. Jiao, X. Ding, Q. Tan, L. Chen, M. Wang, and R. Xiong, "High-precision multicamera-assisted camera-IMU calibration: Theory and method," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–17, 2021.
[27] J. Ren, F. Guan, T. Wang, B. Qian, C. Luo, G. Cai, C. Kan, and X. Li, "High precision calibration algorithm for binocular stereo vision camera using deep reinforcement learning," *Comput. Intell. Neurosci.*, vol. 2022, pp. 1–10, Mar. 2022.

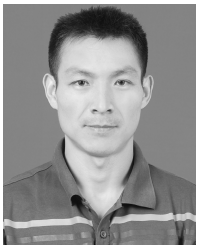
[28] S. Kang, Y. Kim, and H. Park, "Effect of planting distance on the mechanical harvesting of hot pepper," *Agriculture*, vol. 11, no. 10, pp. 1–12, Sep. 2021.
[29] J. Jiang, Y. Jiang, and L. Zhang, "Workpiece detection and localization system based on neural network and depth camera," *Transducer Microsyst. Technol.*, vol. 39, no. 8, pp. 82–85, Jul. 2020, doi: 10.13873/J.1000-9787(2020)08-0082-04.
[30] H. Peng, B. Huang, Y. Shao, Z. Li, C. Zhang, Y. Chen, and J. Xiong, "General improved SSD model for picking object recognition of multiple fruits in natural environment," *Trans. Chin. Soc. Agricult. Eng.*, vol. 34, no. 16, pp. 155–162, Aug. 2018.



SHIHAO ZHONG is currently pursuing the master's degree with the School of Mechanical and Electrical Engineering, Guizhou Normal University. His research interests include deep learning for classification, object recognition, tracking, and detection for the vision system of agricultural robot.

WEIPING XU received the master's degree in mechanical engineering from Guizhou University. He is currently a Professor with the School of Mechanical and Electrical Engineering, Guizhou Normal University, Guiyang, China. His research interests include digital machining and agricultural machinery design.

TAIHUA ZHANG received the Ph.D. degree in mechanical engineering from Zhejiang University. He is currently a Doctor with the School of Mechanical and Electrical Engineering, Guizhou Normal University, Guiyang, China. His research interests include digital design and manufacturing, and production engineering.



HUAWEI CHEN received the Ph.D. degree in mechanical engineering from the Beijing Institute of Technology. He is currently a Professor with the School of Mechanical and Electrical Engineering, Guizhou Normal University, Guiyang, China. His research interests include digital design and manufacturing, image processing, and 3D printing.