

# Automatic detection of pecan fruits based on Faster RCNN with FPN in orchard

Chunhua Hu<sup>1</sup>, Zefeng Shi<sup>1</sup>, Hailin Wei<sup>2\*</sup>, Xiangdong Hu<sup>1</sup>, Yuning Xie<sup>1</sup>, Pingping Li<sup>3</sup>

(1. College of Information Science and Technology, Nanjing Forestry University, Nanjing 210037, China;

2. Hunan Academy of Forestry, Changsha 410004, China;

3. College of Biology and Environment, Nanjing Forestry University, Nanjing 210037, China)

**Abstract:** Although the development of the robot picking vision system is widely applied, it is very challenging for fruit detection in orchards with complex light and environment, especially for fruit colors similar to the background. In recent, there are few studies on pecan fruit detection and location based on machine vision. In this study, an accurate and efficient pecan fruit detection method was proposed based on machine vision under natural pecan orchards. In order to solve the illumination problem, a light compensation algorithm was first utilized to process the collected samples, and then an improved Faster Region Convolutional Neural Network (Faster RCNN) with the Feature Pyramid Networks (FPN) was established to train the samples. Finally, the pecan number counting method was introduced to count the number of pecan. A total of 241 pecan images were tested, and comparison experiments were carried out. The mean average precision (mAP) of the proposed detection method was 95.932%, compared with the result without uneven illumination correction (UIC), which was increased by 0.849%, while the mAP of the Single Shot Detector (SSD)+FPN was 92.991%. In addition, the number of clusters was counted using the proposed method with an accuracy rate of 93.539% compared with the actual clusters. The results demonstrate that the proposed network has good robustness for pecan fruit detection in different illumination and various unstructured environments, and the experimental achievement has great potential for robot-picking visual systems.

**Keywords:** pecan fruit, fruit detection, Faster RCNN, FPN, uneven illumination correction

**DOI:** 10.25165/ijabe.20221506.7241

**Citation:** Hu C H, Shi Z F, Wei H L, Hu X D, Xie Y N, Li P P. Automatic detection of pecan fruits based on Faster RCNN with FPN in orchard. Int J Agric & Biol Eng, 2022; 15(6): 189–196.

## 1 Introduction

*Carya illinoensis*, also known as American pecan, has rich nutrition and a crisp taste. It is a world nut with high content of polyunsaturated fatty acids in its kernel and has healthcare functions such as reducing cholesterol. Therefore, various kinds of pecans are widely planted, and it is very necessary to study the fruit size, quantity statistics, and picking of automatic pecan in trees. Non-contact automatic detection of single pecan is the primary task. However, it is difficult to segment pecan fruit automatically due to the color similarities between fruit and leaves in the complex environment.

Fruits detection system sensors mainly include color cameras<sup>[1]</sup>, spectral cameras<sup>[2]</sup>, thermal cameras<sup>[3]</sup>, RGB-Depth (RGB-D) cameras<sup>[4]</sup>, and Lidar<sup>[5]</sup>. Each of these sensors has its strengths and weaknesses when used in real-field conditions, with the best choice depending on the specific application. As a fruit-picking

system, visual sensors such as color cameras and RGB-D cameras are more widely applied in picking robots. Previous research on the task of in-field fruit detection methods detection based on machine vision are mainly combining features with machine learning to detect fruits. The appearance features including color threshold, shape feature, and texture<sup>[1,6,7]</sup> were used as fruit detection features, and the fruit classifiers were established such as Bayesian classifier, *K*-means clustering<sup>[8]</sup>, *K*-Nearest Neighbor (KNN) clustering<sup>[9]</sup>, Artificial Neural Network (ANN)<sup>[10]</sup>, and Support Vector Machine (SVM)<sup>[11]</sup>. The fruits in the previous research are all single-fruit and have obvious color or shape features, and the features are easily extracted. However, the skin color of pecan fruits is similar to that of its leaves, and most of its fruits are clustered together, the traditional color features and shape features are difficult to be extracted.

With the development of deep learning, convolution neural network (CNN) methods have been widely used in object detection, since the Region Convolutional Neural Network (RCNN) has been proposed in 2014<sup>[12]</sup>, which the Region Propose Network (RPN) was introduced to the network, and the RPN proposes the Region of Interest (ROI) of foreground class. Objects detection approaches have been evolving, becoming more efficient than traditional computer vision methods. In order to improve the detection speed, the Fast RCNN<sup>[13]</sup> and Faster RCNN<sup>[14,15]</sup> have been successively proposed. Later, He et al.<sup>[14]</sup> proposed an instance segmentation algorithm known as Mask RCNN. Sun et al.<sup>[16]</sup> introduced the Faster RCNN using Resnet-50 backbone with residual blocks, and a *K*-means clustering method was used to adjust more anchor sizes to improve the detection accuracy. Kang

**Received date:** 2021-12-06 **Accepted date:** 2022-10-06

**Biographies:** **Chunhua Hu**, PhD, Professor, research interest: computer vision and artificial intelligence, Email: chunhh@163.com; **Zefeng Shi**, MS, research interest: intelligent agricultural equipment and technology, Email: zefengshi@njfu.edu.cn; **Xiangdong Hu**, MS, research interest: artificial intelligence and digital forestry, Email: hxd@njfu.edu.cn; **Yuning Xie**, MS, research interest: urban remote sensing identification, Email: 1253437154@qq.com; **Pingping Li**, PhD, Professor, research interest: agricultural information engineering and facility horticulture, Email: lipingping@ujf.edu.cn.

\***Corresponding author:** **Hailin Wei**, PhD, Senior Engineer, research interest: crop information science and digital forestry, Hunan Academy of Forestry, Changsha 410004, China. Tel: +86-13548583644, Email: hailinwei82@163.com.

et al.<sup>[17]</sup> developed a deep neural network DaSNet-v2 to perform detection and segmentation on fruits and branches in apple orchard environments and gave the performance evaluation parameters of different backbones for apples and branches. Redmon et al.<sup>[18,19]</sup> combined the RPN branch and classification branch into a single network, leading to more concise architecture and better computational efficiency, the network is the You Only Look Once (YOLO), which achieves state-of-the-art performance in object detection with high computation speed. Due to their strong feature extraction ability and autonomous learning ability<sup>[20]</sup>, the YOLO network is widely applied in fruit detection<sup>[21,22]</sup>. Tian et al.<sup>[23]</sup> applied a customized YOLO-V3 network to apple detection. Koirala et al.<sup>[24]</sup> adopted the YOLO architecture in the yield estimation of mango fruit, and accurate detection performance was reported from their work. The above research mainly focused on single-fruit detection, and there are few researchers on clustered fruit detection, such as pecan fruits. Yang et al.<sup>[21]</sup> developed an FPN-strengthened Mask-RCNN in strawberry detection under a non-structured environment, the good results in both detection and instance segmentation tasks are shown in their work. They gave the fruit detection results of 100 test images with an average detection precision rate of 95.78%, a recall rate of 95.41%, and a mean intersection over union (MIoU) rate of 89.85%.

Though all the above studies, which utilize color or shape features and machine learning, have made some progress toward automatic fruit detection and localization. However, there are few researchers on the detection and location of pecan fruit based on machine vision. Now, there are still some problems in fruit detection in a natural unstructured environment, especially in fruits with high overlap state 1) some color-threshold-based methods cannot recognize fruit with similar background color; 2) some shape-feature-based methods cannot detect the clustered fruit; 3) some methods cannot solve the problem of uneven illumination; 4) some methods cannot detect the fruits with different sizes.

Aiming at the current shortcomings of fruit detection, the main objectives of this study were as follows 1) different kinds of cameras were employed to collect sample data at different times and in different orchards; 2) all the pictures were preprocessed using the illumination equalization method; 3) different scale samples were labeled; 4) all the labeled samples were trained using Faster RCNN+FPN; 5) the area of the overlapping area and the number of overlapping fruits were calculated; 6) a large number of experiments were carried out to verify the performance of the algorithm.

## 2 Methods and materials

### 2.1 Framework of the proposed method

In this study, an automatic pecan detection method was proposed based on Faster RCNN with FPN. To improve the robustness of the method, two different cameras were used to acquire training samples with different image sizes and introduce a light equalization process to solve the illumination non-uniformity problem. Subsequently, labelImg was employed to label the training samples of different sizes and construct the training network. In order to provide better picking information for the picking robots, an overlapping region calculation method was proposed to calculate the number of overlapping pecans. Finally, the comparison experiments were carried and the performance of the proposed method was analyzed. The framework of the method is shown in Figure 1.

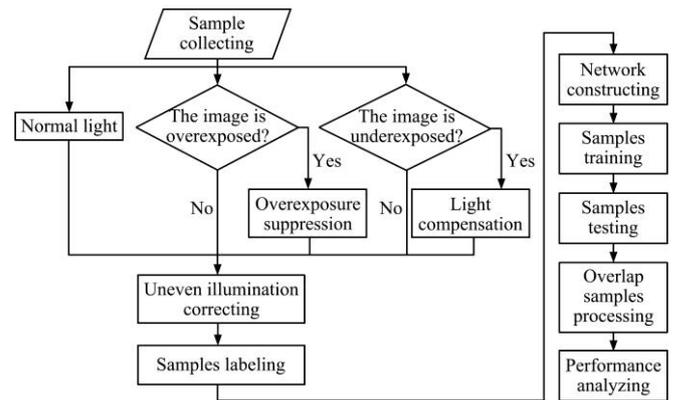


Figure 1 Framework of the proposed method

### 2.2 Data acquisition

To verify the effectiveness and feasibility of the proposed model algorithm, the photos of pecans taken in the experimental base of Nanjing Forestry University, Zhangmiao Village, Houbai Town, Jurong City, Zhenjiang, Jiangsu Province, China (119°9'6"E, 31°52'45"N). It belongs to the central monsoon climate zone of the subtropical region, the average annual temperature is 15.6 °C, and the average annual sunshine is 2157 h. The annual average precipitation is 1018.6 mm.

Pawnee was selected as this research subject. These pictures are taken from 9:00 a.m. to 5:00 p.m., August to September 2021. The device was a Nikon D7000 with an image size of 2464×1632 pixels and a resolution of 300 dpi, and the other images were captured by the Hunan Academy of Forestry using Realsense with an image size of 1280×720 pixels. The distance between the camera and the pecan is about 0.5-2.5 m. The difference in image sizes output from different shooting equipment can improve the robustness of the experiment.

In order to prevent the poor performance of the model caused by insufficient diversity of training samples, not only using different shooting devices but also the following measures were taken during the process of image acquisition. Considering the difference in imaging results caused by different light conditions, images were collected in sunny and cloudy, and the images were also captured along with light and backlight. Respectively, in the process of sampling, different forms and occlusions of pecan organs were taken into account, and fruits with different maturity were photographed from multiple angles to increase the diversity of samples. Figure 2 gives the partial samples taken in different situations.

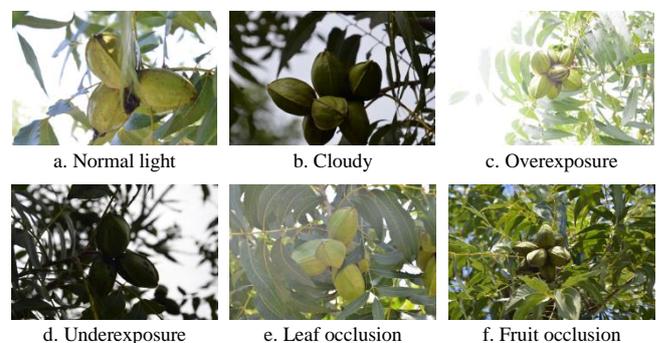


Figure 2 Examples of photos of pecans in various situations

### 2.3 Uneven illumination correcting

Due to rapid changes in sunlight and metering errors in the camera at the time of the shooting, it is possible to take photos that are overexposed, seeing the example in Figure 2c. In such photos, highlight details are lost due to over-exposed areas, resulting in

insufficient detail levels in the image. Therefore, it is easy to cause insufficient feature learning of pecan in the network. In this study, the dark channel image defogging algorithm (DCID)<sup>[25]</sup> was used to detect white dots in images using dynamic thresholds, including two steps, one is white spot detection and the other is white spot adjustment to suppress the exposure of backlight or overexposed photos in the data set. So that the details in the images were clearer and conducive to feature extraction. The picture after illumination compensation is shown in Figure 3a.

On account of the rapid uneven change of sunlight and the errors caused by the automatic metering of the camera during the shooting, the photos taken may be overexposed (Figure 2c) and underexposed (Figure 2d). In order to reduce the interference of the subsequent detection of pecans, the photos with problematic exposure should be corrected. Some photos may be underexposed due to a lack of light and changes in shooting angles. Such images have relatively low contrast and dull colors due to the lack of detail in the underexposed areas. Therefore, it is easy to interfere with the detection and recognition of pecan. In this study, Zero-Reference Deep Curve Estimation (Z-RDCE)<sup>[26]</sup> was employed to enhance the low-light image by formulating the low-light image and to realize exposure compensation for under-exposed pictures in the data set. Figure 3b shows the compensation result of the underexposed picture (Figure 2d). The number of photos including overexposure and underexposure in the data set and the methods of processing with uneven illumination correcting are listed in Table 1. All the underexposed and overexposed images are corrected by uneven illumination and then input into the network structure.

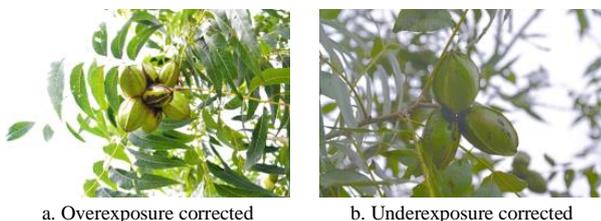


Figure 3 Comparison before and after uneven illumination correcting

Table 1 Number of photos and the methods of processing with uneven illumination correcting

State	Number	Method
Overexposure	46	DCID
Underexposure	53	Z-RDCE

Note: Z-RDCE: zero-reference deep curve estimation; DCID: dark channel image defogging algorithm.

#### 2.4 Data construction and annotation

In this study, labelImg software was used to label the pecan in the dataset. After each image was annotated, a corresponding XML file containing the category and location information of the target, similar to the dataset format of the PASCAL visual object classes challenge 2007<sup>[27]</sup>, was generated. A Python script was used to augment a small number of sample maps, including random flip (horizontal, vertical), transform angle ( $0^{\circ}$ - $180^{\circ}$ ), random scaling of the original image scaling factor, etc. And then the expanded set was randomly divided into 4:1 ratios between the training set and the test set. The expanded training set was used to train the model.

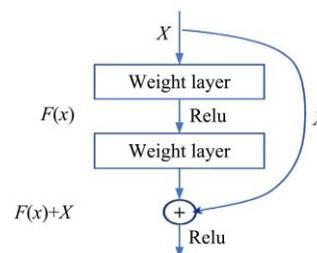
#### 2.5 Pecan fruit detection model structure

Faster RCNN is the most classic object detection algorithm in two-stage. It is divided into two steps, the first stage finds the

anchor rectangle of the object to be detected in the picture (two classifications of the background and the object to be detected), and the second stage classifies the object to be detected in the anchor frame. In Fast RCNN, VGG16 is chosen as the backbone. However, it requires a lot of parameters, and the training time is too long, many people choose ResNet 50 instead of VGG16 gradually. Compared with the VGG network, ResNet has lower complexity and fewer parameters and solves the problem of decreasing correlation between gradients by using residual structure. However, Fast RCNN uses the last layer extracted from Backbone for prediction, and the detection effect of small targets is not ideal. Therefore, the FPN module was added in the original Faster RCNN in this study to combine shallow details with high-level semantics to improve the success rate of small object detection.

##### 2.5.1 An improved feature extraction network

As the number of network layers deepens, the convolutional layer can learn deeper abstract features, which may bring higher accuracy. It is proved that the accuracy of recognition increases with the increase of network depth. However, the simple stacked convolutional layer cannot train the network smoothly due to the gradient explosion when propagating backward. He et al.<sup>[28,29]</sup> proposed a deep residual ResNet model based on the concept of identity mapping in order to break through the problems of reduced accuracy and limited depth of the deep network. This method solves the degradation problem by fitting the residual graphs with the multi-layer network. Since the residual structure does not increase model parameters, the difficulties of gradient disappearance and training degradation are effectively alleviated, and the convergence performance of the model is improved. The residual learning module is shown in Figure 4. ResNet 50 which achieves a good result in the feature extraction performance was used as the backbone network for feature extraction in this study. Therefore, the ResNet model using 101 layers and 152 layers did not be considered in this study.



Note:  $x$  is the input variable;  $F(x)$  is the residual function; ReLU: Rectified Linear Unit.

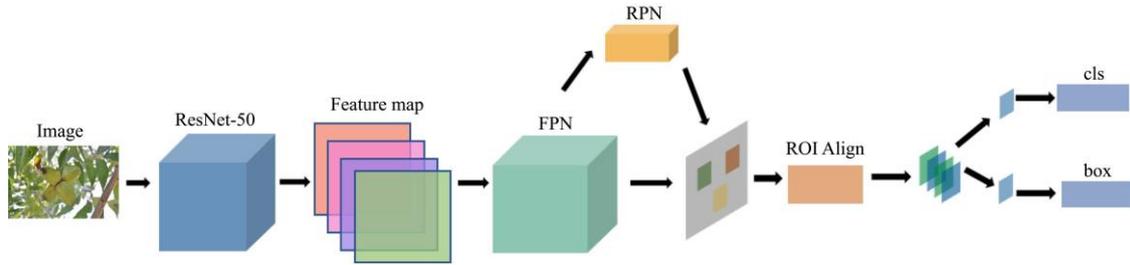
Figure 4 Residual learning module

Image feature extraction is based on shared convolution layers. Low-level features such as edges and angles were extracted by the underlying network. High-level features that describe target categories were extracted at the higher level. In order to better represent the target on multiple scales, the feature pyramid network (FPN)<sup>[30]</sup> was introduced to extend the backbone network as shown in Figure 4, which is especially effective for the detection of small targets. Since direct mapping is difficult to learn, the basic mapping relationship from input variable  $x$  to residual function  $F(x)$  is no longer being learned, but the difference between the two is learned, which is the residual, and in order to calculate  $F(x)$ , this residual is simply added to the input. The top-level features of FPN architecture were merged with the underlying features by up-sampling, with each layer independently predicting feature maps. Using the feature map of the four blocks whose output channels are {256, 512, 1024, 2048} respectively as the input of FPN. After FPN

processing, the channels are all the same to 256. Then put it into RPN for candidate box generation. In RPN, candidate boxes of different sizes are predicted for different feature layers.

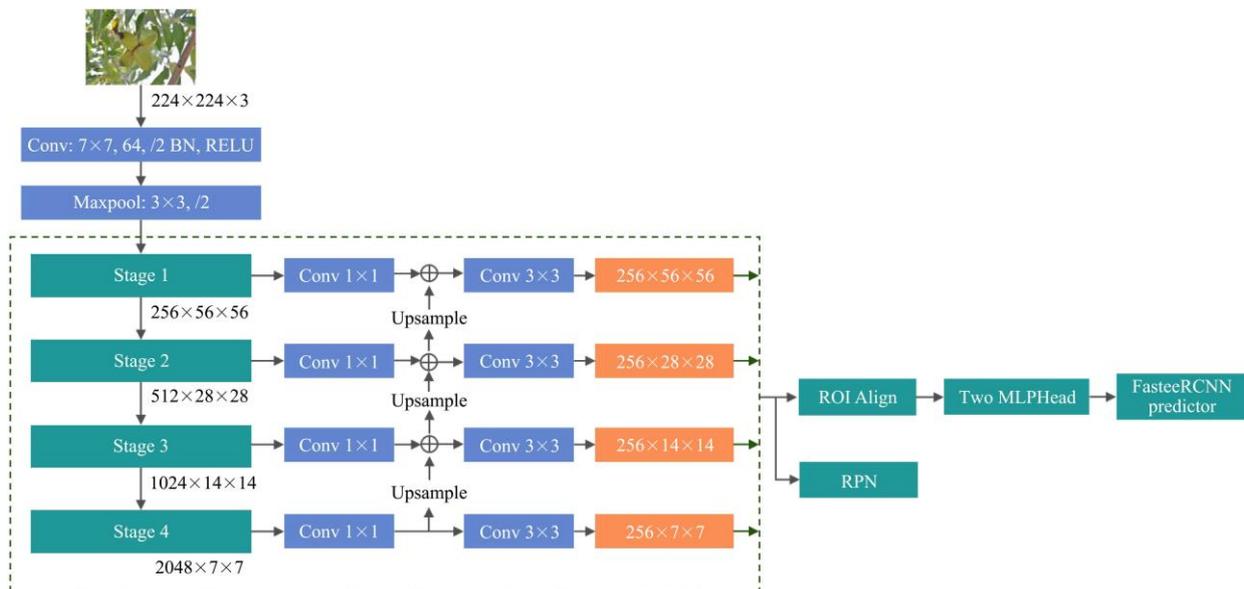
In addition, the ROI Pooling in traditional Faster RCNN will first quantify the coordinate of the candidate box generated by RPN into integer pixel coordinate value, and then divide the candidate area into  $7 \times 7$ . The pixel value is rounded during Pooling, which

causes certain errors. Therefore, the original ROI Pooling is replaced by ROI Align, the coordinates of the floating-point candidate box generated by RPN are maintained, the quantization process is canceled, and the bilinear interpolation algorithm is used to generate the final feature map. Figure 5 illustrates the improved Faster RCNN network. Figure 6 shows the junction of ResNet-50 and FPN.



Note: ResNet-50: Residual networks with 50 layers; RPN: Region Propose Network; FPN: Feature Pyramid Networks; ROI: Regions of Interest; cls: Classification; bbox: bounding box.

Figure 5 Improved framework for the Faster RCNN



Note: Conv: Convolution; MLP Head: Multi-Layer Perceptron Head; Faster RCNN: Faster Region Convolutional Neural Network.

Figure 6 Structure at the junction of ResNet-50 and FPN

### 2.5.2 Feature pyramid networks

In object detection, the low-level feature semantic information is less, but the target location is accurate. High-level feature semantic information is rich, but the target location is rough. General networks use bottom-up convolution and the feature map of the last layer is used for prediction. However, because high-level features are rough for target location, the detection effect of small target objects will be sharply reduced. In addition, due to the different focal lengths of cameras, the sizes of objects in the images are different. Therefore, this multi-scale feature fusion method is adopted to fuse the shallow features with high resolution and the deep features with rich semantic information.

### 2.5.3 ROI Align

Both traditional Faster RCNN and Fast RCNN use ROI Pooling, which maps candidate boxes predicted by RPN to the original map, and then divides candidate areas into  $7 \times 7$  sizes. But there may be a small error in these two operations, first, the floating point coordinates of the prediction candidate box are mapped to the original image; and the entire operation must be taken, which will produce a certain error, and second, in the candidate area was divided into  $7 \times 7$  areas, if the image size is not equal to an integer

because of the image size itself, after the two integer operations, the final result may produce a large error. For this reason, we choose ROI Align to replace the original ROI Pooling. ROI Align eliminates the above two round operations and uses bilinear interpolation to make the final position more accurate.

### 2.6 Overlap samples processing

In order to provide better picking information for the picking robot, we propose an overlapping region calculation method to calculate the area of overlapping pecan. Two pecan detection frames are shown in Figure 7. The coordinates of the left upper corner and the right lower corner of the rectangular detection frame of one pecan are  $(x_1, y_1)$  and  $(x_2, y_2)$ , and the coordinates of the upper left corner and the lower right corner of the rectangular detection frame of another pecan are  $(x_3, y_3)$  and  $(x_4, y_4)$ . Assuming that:

$$x_{\min} = \max(\min(x_1, x_2), \min(x_3, x_4)) \quad (1)$$

$$y_{\min} = \max(\min(y_1, y_2), \min(y_3, y_4)) \quad (2)$$

$$x_{\max} = \min(\min(x_1, x_2), \max(x_3, x_4)) \quad (3)$$

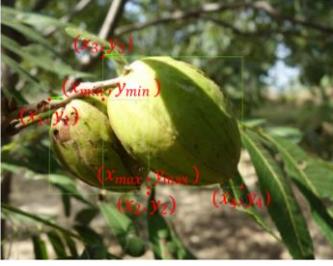
$$y_{\max} = \min(\min(y_1, y_2), \max(y_3, y_4)) \quad (4)$$

where,  $x_{\min}$ ,  $y_{\min}$ ,  $x_{\max}$ , and  $y_{\max}$  are the coordinates of the left upper corner and the coordinates of the right lower corner of the

overlapping area.

If two pecan test frames have overlapped, the overlap area  $S$  is:

$$S = (x_{\max} - x_{\min}) \times (y_{\max} - y_{\min}) \quad (5)$$



Note: The coordinates of the left upper corner and the right lower corner of the rectangular detection frame of one pecan are  $(x_1, y_1)$  and  $(x_2, y_2)$ , and the coordinates of the upper left corner and the lower right corner of the rectangular detection frame of another pecan are  $(x_3, y_3)$  and  $(x_4, y_4)$ .  $x_{\min}$ ,  $y_{\min}$ ,  $x_{\max}$ , and  $y_{\max}$  are the coordinates of the left upper corner and the coordinates of the right lower corner of the overlapping area.

Figure 7 Schematic of overlapping sample processing

Combining the above method of calculating the overlapping area  $S$ , we have realized the calculation method of calculating the clusters of the entire image. All coordinates of the boxes in the same image used established network are marked as  $P$ ,  $P_k(x_{1k}, y_{1k}, x_{2k}, y_{2k})$  ( $k=1, 2, 3, \dots, m$ ) the  $k^{\text{th}}$  coordinate of the box,  $m$  is the total number of boxes.  $D$  is the matrix of overlap areas between each detection box and the remaining boxes;  $Q$  is the index of the boxes;  $N$  is the index matrix of the same cluster boxes, which initializes  $N=0$ . The number of rows that are not all zero in  $N$  is recorded as  $n$ , and  $n$  is the number of clusters.

The detailed procedures for calculating clusters are given as follows:

Step 1 Input original image.

Step 2 Detect the coordinate Boxes of the box in the image, and mark the coordinates as  $P(\text{rect.min.x}, \text{rect.min.y}, \text{rect.max.x}, \text{rect.max.y})$ , record the id of each box as  $k$  ( $k=1, 2, 3, \dots, m$ ).

Step 3 Create a two-dimensional matrix  $D$  with a size of  $m \times m$ , and use coordinates  $P$  to calculate the overlap area between each box. Greater than 0 is True, otherwise False, write the result into matrix  $D$ .

Step 4 Create a dictionary, the key of the dictionary is  $e$  ( $e=1, 2, 3, \dots, m$ ), which means the number of clusters. The value corresponding to each key is the id of the box in this cluster.

Step 5 Traverse all keys in the dictionary. And initialize a queue  $Q$ . Check all the boxes. If the id of the box is out of the value of the dictionary, it means that it has formed a cluster. Otherwise, put this box into the value of the corresponding cluster, and put this value into the queue.

Step 6 When the queue is not empty, take the first element  $k$  of the queue and traverse all the values of the  $k^{\text{th}}$  row in the matrix  $D$ . If  $A[k][i]$  ( $i=1, 2, 3, \dots, m$ ) is not in the value in the dictionary, and  $i$  is not Equal to  $k$  and  $A[k][i]$  is True, it means that the boxes of  $k$  and  $i$  overlap, add  $i$  to the value, and put  $i$  into the queue;

Step 7 Repeat 6 until the queue is empty, repeat 5 until the key of the entire dictionary is traversed, and the search ends.

Step 8 Output the number of non-empty values in the dictionary.

## 2.7 Performance analyzing

The data set described in Section 2.2 was used for the evaluation of image detection results obtained by the proposed network model. The precision, recall, and accuracy, which these indexes can be used to evaluate the proposed algorithm's detection performance<sup>[31]</sup>. They were used as the evaluation metrics, and

they were respectively defined as follows, the precision is defined as,

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (6)$$

The recall rate is the ratio of relevant targets in the returned results to all relevant targets, and it is defined as,

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (7)$$

$$\text{Accuracy} = \frac{T}{R} \quad (8)$$

where TP is the number of cases that are positive and detected positive; FP is the number of cases that are negative but detected positive; FN is the number of cases that are positive but detected negative<sup>[32]</sup>.

The average precision (AP) is usually used as a metric, and the calculation formula is

$$\text{AP} = \int_0^1 p(r) d(r) \quad (9)$$

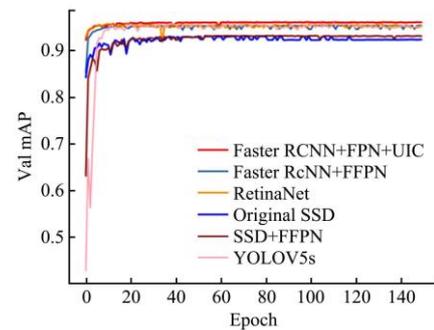
where,  $p$  represents Precision;  $r$  represents Recall;  $p$  is a function of  $r$ . So, the mAP equals the average of all categories of average precision.

## 3 Results and discussion

The performance of the proposed detection method was evaluated using the dataset mentioned in Section 2.2. The experiment was performed under the deep learning development framework of Pytorch, with NVIDIA 3060 for GPU acceleration, Inter (R) Core (TM) i7-8700k CPU, and 16G memory. Numerous comparison experiments were performed to evaluate the effectiveness and practicability of the proposed method.

### 3.1 Evaluation of the network

The data set described was used for the evaluation of image segmentation results obtained by the proposed network model. An experiment of the original Faster RCNN was made which without ROI Align structure and FPN, the accuracy of the experimental result is 87.9%, while the original Faster RCNN which with ROI Align structure, the accuracy of the experimental result is 89.9%, which increases by 2%. The precision, recall, and accuracy were used as the evaluation metrics. To verify the efficiency of the proposed network, the original SSD, SSD+FPN, RetinaNet, YOLOV5s (without UIC), Faster RCNN+FPN, and the proposed network (Faster RCNN+FPN+uneven illumination correcting (UIC)) were adopted by illumination to train and test the same samples. The evaluations of mAP were as illustrated in Figure 8. From these results, the proposed network has better than the other classical networks' inefficiency and obtains more accuracy.



Note: mAP equals the average of all categories of average precision; UIC: Uneven illumination correction; SSD: Single Shot Detector.

Figure 8 Comparison of classical network structures of mAP

It can be seen from Table 2 that the mAP of the different network structures. The Faster RCNN+FPN+UIC mAP improved by 3.723% compared with the original SSD. The mAP of Faster RCNN+FPN+UIC improved by 1.919% over YOLOv5s. What's more, the Faster RCNN+FPN+UIC mAP improved by 0.849% compared with Faster RCNN+FPN without illumination correcting. This indicates that the proposed detection method is better than the original model for the detection and recognition of pecans. Table 2 also lists the speed of different network structures. Among them, YOLOv5s (without UIC) runs at the fastest speed of 0.005 s and Faster RCNN+FPN at 0.070 s. The addition of uneven illumination correcting results in an increase in Faster RCNN+UIC runtime of 0.084 s, the detection speed is still relatively fast, which does not affect the picking speed.

**Table 2 mAP and speed of the different network structures**

Network structure	mAP/%	Speed/s
Original SSD	92.209	0.009
SSD+FPN	92.991	0.011
Retinanet	95.128	0.072
Faster RCNN+FPN	95.083	0.070
YOLOv5s (without UIC)	94.013	0.005
Faster RCNN+UIC	95.932	0.084

Note: SSD: Single Shot Detector; mAP equals the average of all categories of average precision; FPN: Feature Pyramid Networks; Faster RCNN: Faster Region Convolutional Neural Network; UIC: Uneven illumination correction.

**3.2 Pecan detection results using the proposed network**

To verify the robustness of the proposed network, the image under different light and different growth states was chosen to detect pecan using the proposed network, and the detection results are shown in Figure 9. From the test results, it can be seen that the network not only can detect the part pecans treated with uneven light compensation but also one or multiple clusters of pecans can be detected. Among them, due to the change of shooting angle, the pictures with occlusion caused by fruits or leaves can also be well identified, which reflects the robustness of the network in this study.

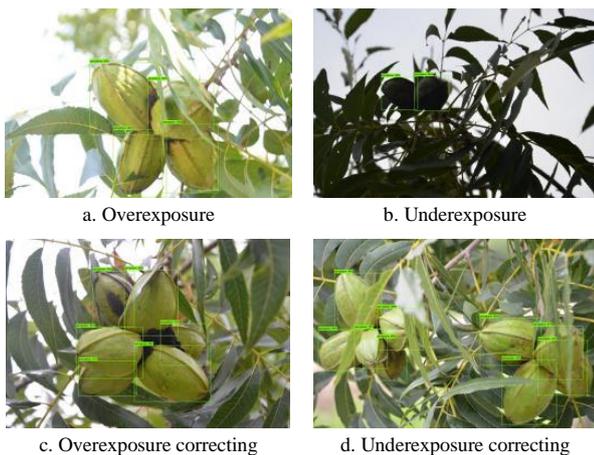


Figure 9 Pecan detection results using the proposed network

The experiments comparing different uneven illumination corrections were also carried out. From Figure 10, the RCNN+FPN+UIC has slight improvement over Faster RCNN+FPN+overexposure processed by DCID or underexposure processed by Z-RDCE.

It can be seen from Table 3 that the mAP of the Faster RCNN after uneven illumination correcting is higher than the others. The Faster RCNN+FPN+UIC mAP improved by 0.849% compared with Faster RCNN+FPN. The Faster RCNN+underexposure

processed by Z-RDCE has improved by 0.6% over Faster RCNN+FPN. In addition, the Faster RCNN+FPN+overexposure processed by DCID is 0.832% higher than Faster RCNN+FPN without any illumination correction.

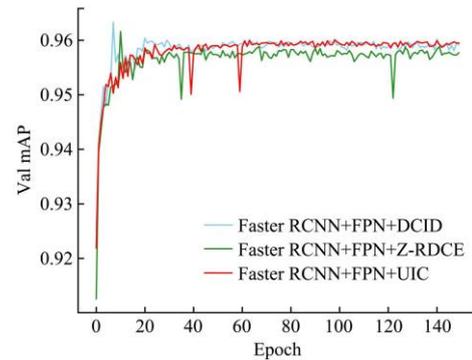


Figure 10 mAP of the Faster RCNN after uneven illumination correcting

**Table 3 mAP of the Faster RCNN after uneven illumination correction**

Network structure	mAP/%
Faster RCNN+FPN	95.083
Faster RCNN+underexposure	95.738
Faster RCNN+FPN+overexposure	95.906
Faster RCNN+FPN+UIC	95.932

**3.3 Evaluation of the sample numbers counting**

The overlap coefficient (OC) was used to evaluate the accuracy of detection results, and the ratio of pixels that belong to the target or background truth in target bounding boxes are correctly divided into the target or background. The higher the value is, the better the detection performance will be achieved. Because the pecan is clustered and overlapped, to count the accurate quantity and provide picking information for the picking robot, this experiment mainly makes statistical analysis on the pecan fruit area and single fruit area using the algorithm in Section 2.6.

Through cluster detection of box coordinates obtained by detection, the accuracy rate of the automatic counting results using the algorithm in Section 2.6 is 93.539%. Multiple detection boxes are grouped into a cluster, which has high accuracy recognition. However, a single box is grouped into a cluster, which may lead to false detection due to the lack of depth information in the images. The result is shown in Table 4.

**Table 4 Number and error rate of value and tested clusters**

Value item	Single box	Multiple boxes	Clusters	Accuracy rate/%
Number of actual clusters	50	283	333	93.539
Number of clusters tested	73	283	356	

**3.4 Discussion**

The goal of this study was to develop a new method for identifying pecans using different photographic equipment and under different lighting conditions. The contribution of the proposed method in this study involves pre-lighting the photos with uneven illumination correcting and then importing the corrected photos into the network, which is different from many existing methods, such as the color of mature fruit differs greatly from the background color<sup>[21,33-37]</sup>, which has obvious color characteristics; and the fruit shape is almost round, the extracted features are analyzed using shape, color, and texture to identify fruits and non-fruits<sup>[38]</sup>, the color and background of the fruit are similar, but the shape of the fruit has distinct features (curved, elongated,

etc.)<sup>[39]</sup>. Specifically, Fan et al.<sup>[33]</sup> combined local image features and color information, proposing a pixel patch segmentation method based on gray-centered red-green-blue (RGB) color space. Kang et al.<sup>[34]</sup> improved the deep neural network DaSNet-v2, which can perform detection and instance segmentation on fruits, and semantic segmentation on branches. Because of the bright color and large volume of apples, they can have a high recognition rate and detection rate. Other fruits with smaller sizes can also be easily identified due to their distinctive colors, such as citrus<sup>[35]</sup>, tomato<sup>[36]</sup>, strawberry<sup>[21]</sup>, and litchi<sup>[37]</sup>. However, these experimental results were either recognized and detected under normal light conditions, or the fruits were given a large amount of light at night for data collection, without taking into account the unstructured experimental situation under natural conditions. Due to the influence of uneven illumination, the difficulty of fruit recognition and detection is improved. Wan et al.<sup>[38]</sup> chose pineapple and Chen et al.<sup>[39]</sup> chose banana as the object of research, which has large fruits and obvious texture and shape characteristics. They did not consider the influence of natural light. Therefore, when the natural light changes rapidly or the angle at which the photo is taken changes, the photo that has not undergone uneven illumination correction may affect the accuracy of fruit recognition and detection. We perform uneven illumination correction on the collected photos and then input the corrected images into the proposed constructed network, which improves the accuracy of recognition and detection of pecans.

#### 4 Conclusions

The segmentation of pecan in the natural environment is of great significance to the picking robot. In this study, the images of pecans were preprocessed, exposure suppression and exposure compensation were performed for the photos with abnormal exposure, and the photos with uneven illumination compensation were input into the network. Then the ResNet-50 with residual structure was used to replace the VGG 16 network. Compared with the original Faster RCNN model, this model added the FPN structure to solve the problem of low accuracy of pecan detection under similar fruit and background colors and overlapping occlusion to a certain extent. At the same time, the dilated convolution with different expansion rates is used to extract the features and perform fusion to achieve high segmentation accuracy. According to the experimental results, the following conclusions can be drawn: the results show that the mAP of pecans increased from 92.209% to 95.932%. The Faster RCNN+FPN+UIC mAP improved by 0.849% compared with Faster RCNN with FPN.

In future studies, the network will be applied to pecans detection and picking under complex light conditions. It can also be used to predict the yield of pecan. In addition, it will be applied to the detection of other kinds of fruits to achieve more agricultural applications.

#### Acknowledgements

This work was funded by the Forestry Science and Technology Innovation Fund Project of Hunan Province (Grant No. XLK202108-4) and the Priority Academic Program Development of Jiangsu Higher Education Institutions.

#### [References]

- [1] Kurtulmus F, Lee W S, Vardar A. Immature peach detection in colour images acquired in natural illumination conditions using statistical classifiers and neural network. *Precision Agriculture*, 2014; 15: 57–79.
- [2] Yang C, Lee W S, Williamson J G. Classification of blueberry fruit and leaves based on spectral signatures. *Biosystems Engineering*, 2012; 113(4): 351–362.
- [3] Stajanko D, Lakota M, Hočevár M. Estimation of number and diameter of apple fruits in an orchard during the growing season by thermal imaging. *Computers and Electronics in Agriculture*, 2004; 42: 31–42.
- [4] Méndez Perez R, Cheein F A, Rosell-Polo J R. Flexible system of multiple RGB-D sensors for measuring and classifying fruits in agri-food industry. *Computers and Electronics in Agriculture*, 2017; 139: 231–242.
- [5] Tsoulas N, Paraforos D S, Xanthopoulos G, Zude-Sasse M. Apple shape detection based on geometric and radiometric features using a LiDAR laser scanner. *Remote Sensing*, 2020; 12(15): 2481. doi: 10.3390/rs12152481.
- [6] Zhou R, Damerow L, Sun Y, Blanke M M. Using colour features of cv. ‘Gala’ apple fruits in an orchard in image processing to predict yield. *Precision Agriculture*, 2012; 13: 568–580.
- [7] 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.
- [8] Liakos K G, Busato P, Moshou D, Pearson S, Bochtis D. Machine learning in agriculture: A review. *Sensors*, 2018; 18(8): 2674. doi: 10.3390/s18082674.
- [9] Wang C L, Tang Y C, Zou X J, SiTu W M. A robust fruit image segmentation algorithm against varying illumination for vision system of fruit harvesting robot. *Optik*, 2017; 131: 626–631.
- [10] Zhao C Y, Lee W S, He D J. Immature green citrus detection based on colour feature and sum of absolute transformed difference (SATD) using colour images in the citrus grove. *Computers and Electronics in Agriculture*, 2016; 124: 243–253. doi: 10.1016/j.jilleo.2016.11.177.
- [11] Ye Q, Huang P, Zhang Z, Zheng Y, Fu L, Yang W. Multiview learning with robust double-sided twin SVM. *IEEE Transactions on Cybernetics*, 2021; 52(12): 12745–12758.
- [12] Girshick R, Donahue J, Darrell T, Malik J. Rich feature hierarchies for accurate object detection and semantic segmentation. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus: IEEE, 2014; pp.580–587. doi: 10.1109/CVPR.2014.81.
- [13] Ren S, He K, Girshick, Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transaction Pattern Analysis and Machine Intelligence*, 2017; 39(6): 1137–1149.
- [14] He K, Gkioxari G, Dollár P, Girshick R. Mask R-CNN, 2017. doi: 10.1109/TPAMI.2018.2844175.
- [15] Wang J M, Chen X X, Cao L, An F, Chen B Q, Xue L F, et al. Individual rubber tree segmentation based on ground-based LiDAR data and faster R-CNN of Deep Learning. *Forests* 2019; 10: 793. doi: 10.3390/f10090793.
- [16] Sun J, He X, Ge X, Wu X, Shen J, Song Y. Detection of key organs in tomato based on deep migration learning in a complex background. *Agriculture* 2018; 8: 196. doi: 10.3390/agriculture8120196.
- [17] 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.
- [18] Redmon J, Divvala S, Girshick R, Farhadi A. You Only Look Once: Unified, real-time object detection. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016; pp.779–788. doi: 10.1109/CVPR.2016.91.
- [19] Fu C Y, Liu W, Ranga A, Tyagi A, Berg A C. DSSD: Deconvolutional Single Shot Detector, arXiv, 2017; arXiv: 1701.06659.
- [20] Kamilaris A, Prenafeta-Boldú F X. Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture*, 2018; 147: 70–90.
- [21] Yu Y, Zhang K L, Zhang D X, Yang L, Cui T. Fruit detection for strawberry harvesting robot in non-structural environment based on Mask-RCNN. *Computers and Electronics in Agriculture*, 2019; 163: 104846. doi: 10.1016/j.compag.2019.06.001.
- [22] Vasconez J P, Delpiano J, Vougioukas S, Auat Cheein F. Comparison of convolutional neural networks in fruit detection and counting: A comprehensive evaluation. *Computers and Electronics in Agriculture*, 2020; 173: 105348. doi: 10.1016/j.compag.2020.105348.
- [23] Tian Y N, Yang G D, Wang Z, Wang H, Li E, Liang Z Z. Apple detection during different growth stages in orchards using the improved YOLO-V3 model. *Computers and Electronics in Agriculture*, 2019; 157: 417–426.
- [24] Koirala A, Walsh K B, Wang Z, McCarthy C. Deep learning for real-time fruit detection and orchard fruit load estimation: Benchmarking of ‘MangoYOLO’. *Precision Agriculture*, 2019; 20: 1107–1135.
- [25] He K M, Sun J, Tang X O. Single image haze removal using dark

- channel prior. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. 2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Miami: IEEE, 2009; pp.1956–1963. doi: 10.1109/TPAMI.2010.168.
- [26] Guo C L, Li C Y, Guo J C, Loy C C, Hou J H, Sam K W. Zero-reference deep curve estimation for low-light image enhancement. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020; pp.1777-1786. doi: 10.1109/CVPR42600.2020.00185.
- [27] Everingham M, Zisserman A, Williams C K I, Van Gool L, Allan M, Bishop C M, Chapelle O, et al. The 2005 PASCAL visual object classes challenge. In: Machine Learning Recognising Tectual Entailment, MLCW 2005, Springer, 2005; 3944: pp.117-176. doi: 10.1007/11736790\_8.
- [28] He K M, Zhang X Y, Ren S Q, Sun J. Identity mappings in deep residual networks. In: Computer Vision – ECCV 2016, Springer, 2016; 9908: 630-645. doi: 10.1007/978-3-319-46493-0\_38.
- [29] Fu L Y, Zhang D, Ye Q L. Recurrent thrifty attention network for remote sensing scene recognition. IEEE Transactions on Geoscience and Remote Sensing, 2021; 59(10): 8257–8268. .
- [30] Lin T Y, Dollár P, Girshick R, He K, Hariharan B, Belongie S. Feature Pyramid Networks for Object Detection. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu: IEEE, 2017; pp.936–944. doi: 10.1109/CVPR.2017.106.
- [31] Yang Q M, Xiao D Q, Lin S C. Feeding behavior recognition for group-housed pigs with the Faster R-CNN. Computers and Electronics in Agriculture, 2018; 155: 453–460. doi: 10.1016/j.compag.2018.11.002.
- [32] Vahidi H, Klinkenberg B, Johnson B A, Moskal L M, Yan W. Mapping the individual trees in urban orchards by incorporating volunteered geographic information and very high resolution optical remotely sensed data: A template matching-based approach. Remote Sensing, 2018; 10(7): 1134. doi: 10.3390/rs10071134.
- [33] Fan P, Lang G D, Yan B, Lei X Y, Guo P J, Liu Z J, et al. A method of segmenting apples based on gray-centered RGB color space. Remote Sensing, 2021; 13(6): 1211. doi: 10.3390/rs13061211.
- [34] Kang H W, 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.
- [35] Zhuang J J, Luo S M, Hou C J, Tang Y, He Y, Xue X Y. Detection of orchard citrus fruits using a monocular machine vision-based method for automatic fruit picking applications. Computers and Electronics in Agriculture, 2018; 152: 64–73.
- [36] Sun J, He X F, Ge X, Wu X H, Shen J F, Song Y Y. Detection of key organs in tomato based on deep migration learning in a complex background. Agriculture, 2018; 8(12): 196. doi: 10.3390/agriculture8120196.
- [37] Liang C X, Xiong J T, Zheng Z H, Zhong Z, Li Z H, Chen S M, et al. A visual detection method for nighttime litchi fruits and fruiting stems. Computers and Electronics in Agriculture, 2020; 169(6): 105192. doi: 10.1016/j.compag.2019.105192.
- [38] Wan Nurazwin Syazwani R, Muhammad Asraf H, Megat Syahirul Amin M A, Nur Dalila K A. Automated image identification, detection and fruit counting of top-view pineapple crown using machine learning. Alexandria Engineering Journal, 2022; 61: 1265–1276.
- [39] Chen T C, Zhang R H, Zhu L X, Zhang S A, Li X M. A method of fast segmentation for banana stalk exploited lightweight multi-feature fusion deep neural network. Machines, 2021; 9(3): 66. doi: 10.3390/machines9030066.