IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 6, 2019, accepted September 14, 2019, date of publication September 18, 2019, date of current version October 7, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2942144

Cucumber Fruits Detection in Greenhouses Based on Instance Segmentation

XIAOYANG LIU[1], DEAN ZHAO[1], WEIKUAN JIA[2], WEI JI[1], CHENGZHI RUAN[3], AND YUEPING SUN[1]

[1]School of Electrical and Information Engineering, Jiangsu University, Zhenjiang 212013, China
[2]School of Information Science and Engineering, Shandong Normal University, Jinan 250358, China
[3]School of Mechanical and Electrical Engineering, Wuyi University, Wuyishan 354300, China

Corresponding authors: Dean Zhao (dazhao@ujs.edu.cn) and Xiaoyang Liu (leoliuxy@foxmail.com)

This work was supported in part by the National Nature Science Foundation of China under Grant 31571571, Grant 61973141, and Grant 61903288, in part by the Natural Science Foundation of Jiangsu Province in China under Grant BK20170536, in part by the Natural Science Foundation of Shandong Province in China under Grant ZR2017BC013, in part by the China Postdoctoral Science Foundation under Grant 2018M630797, in part by the Fujian Province Natural Science Fund Project under Grant 2018J01471, and in part by the Fujian Province Outstanding Young Scientists Training Project, Fujian T&R (2018), under Grant 47.

ABSTRACT The cucumber fruits have the same color with leaves and their shapes are all long and narrow, which is different from other common fruits, such as apples, tomatoes, and strawberries, etc. Therefore, cucumber fruits are more difficult to be detected by machine vision in greenhouses for special color and shape. A pixel-wise instance segmentation method, mask region-based convolutional neural network (Mask RCNN) of an improved version, is proposed to detect cucumber fruits. Resnet-101 is selected as the backbone of Mask RCNN with feature pyramid network (FPN). To improve the detection precision, region proposal network (RPN) in original Mask RCNN is improved. Logical green (*LG*) operator is designed to filter non-green background and limit the range of anchor boxes. Besides, the scales and aspect ratios of anchor boxes are also adjusted to fit the size and shape of fruits. Improved Mask RCNN has a better performance on test images. The test results are compared with that of original Mask RCNN, Faster RCNN, you only look once (YOLO) V2 and YOLO V3. The F_1 score of improved Mask RCNN in test results reaches 89.47%, which is higher than the other methods. The average elapsed time of improved Mask RCNN is 0.3461 s, which is only lower than the original Mask RCNN. Meanwhile, the mean value and standard deviation of location deviation in improved Mask RCNN are 2.10 pixels and 1.73 pixels respectively, which are lower than the other methods.

INDEX TERMS Machine vision, cucumber detection, Mask RCNN, instance segmentation.

I. INTRODUCTION

Fruit detection is an important research filed in precision agriculture, which is widely applied to yield estimation, and fruit picking robot [1]–[3]. The related researches include the detection of apples [4]–[6], kiwis [7], [8], oranges [9], tomatoes [10], litchis [11], and peppers [12] etc. Cucumber fruits have the same color with leaves and their shapes are long and narrow, which is different from other common fruits. Therefore, the cucumber fruits are more difficult to be detected by machine vision for special color and shape. Zhang *et al.* [13] adopted a three-layer back propagation (BP) neural network to segment cucumber fruits from the background. The blue and saturation color components extracted from different color spaces as the input of the BP network. Wang *et al.* [14] adopted a pulse coupled neural network (PCNN) to segment cucumber fruits. The researches listed above adopted different methods to detect cucumbers, but the general strategy is similar. Firstly, a segmentation method was proposed to segment cucumber fruits from the background based on color or intensity of pixels. However, the results of segmentation were rough, which also contained other connected regions except for cucumber fruits. Then, morphology operations, texture and shape features were also employed to filter other regions in the next steps. The experimental results of researches listed above indicated that these methods were difficult to reach high precision rate and easier to be effected by illuminations. To improve the precision rate of detection, Yuan et al. taken spectral images [15] and near infrared images (NIR) [16] as samples for the detection of

The associate editor coordinating the review of this manuscript and approving it for publication was Xiao-Yu Zhang.

cucumber fruits. The gray values of leaves and fruits in a NIR image have a greater difference when the wavelength is 850nm. In addition to increasing grayscale difference, the special shape of cucumber fruits was also taken into consideration by some researchers. Bao *et al.* [17] designed a multi-template matching library including 65 cucumber images and applied them to detect fruits in a natural environment.

With the rapid development of the convolutional neural network (CNN) in recent years, the detection speed and accuracy of CNNs are higher than traditional object detection algorithms generally. Different types of CNNs have been applied to detect a variety of fruits. Tao *et al.* [18] adopted Faster RCNN to detect peaches, apples and oranges. In this study, two kinds of CNNs, ZFnet and VGG16, were used as the backbone network of Faster RCNN to detect all kinds of fruits mentioned above respectively and the detection precisions were all more than 90%. Halstead *et al.* [19] used Faster RCNN framework based on VGG16 architecture to detect sweet peppers and estimate ripeness by learning a parallel layer. YOLO and single shot multi-box detector (SSD) have higher speed than Faster RCNN. Tian *et al.* [20] proposed an improved YOLO V3 model to detect apples during different growth stages in orchards. Lamb and Chuah [21] presented an optimized SSD and run it on a Raspberry Pi 3B to detect strawberries. Some works on machine learning and computer vision were also published, which can improve the performance of CNNs [22]–[24].

However, the CNNs listed above only can detect fruits by rectangle bounding boxes. The location accuracy of boxes is enough for suborbicular fruits, but the horizontal location accuracy is not enough for long and narrow cucumber fruits. An instance segmentation method, Mask RCNN of an improved version, is proposed to detect cucumber fruits in pixel level in our study. Pixel-wise object detection not only can detect objects, but also can locate objects with higher accuracy. Yu *et al.* [25] used Mask RCNN to detect ripe and unripe strawberries and proposed a visual location method to determine picking points. However, the original Mask RCNN is designed to detect a variety of objects rather than a specified object. In our study, Mask RCNN is only used to detect cucumber fruits. Therefore, a number of anchor boxes produced by the original Mask RCNN are redundant and the aspect ratios and sizes of anchor boxes do not fit the shape of cucumber fruits. Therefore, the detection efficiency and accuracy for cucumber fruits can be improved further. Some improvements are made for better performance in our study. In consideration of cucumber color, *LG* operator is designed and added into RPN to limit anchor boxes in green regions. Furthermore, the scales and aspects of anchor boxes are redesigned to fit the size and shape of cucumber fruits. Finally, the detection accuracy and location accuracy of improved Mask RCNN is evaluated by comparing with original Mask RCNN, Faster RCNN, YOLO V2 and YOLO V3. Compared with our previous works [26]–[28], this work focuses on the detection of cucumber fruits and is more challenging.

II. MATERIALS AND METHODS

A. IMAGE DATA ACQUISITION

The original images of cucumber fruits were taken in a greenhouse by Canon EOS 760D. The greenhouse is located in Jiangsu Agricultural Expo Garden in Zhenjiang city, Jiangsu province, China. The cucumbers in the greenhouse were cultivated by soilless culture technology and grew on vertical ropes, as shown in Fig. 1a. The collected images were saved as default jpg format with resolution 6000 × 4000. To improve computing speed, the original images were resized to 600 × 400.

Total of 522 images including cucumber fruits were taken. The sample images are shown in Fig. 1. Each cucumber fruit in these images is labeled in pixel level manually. The image annotation tool, Labelme, is applied to label fruits by polygons. The corresponding annotated files are saved as json format. To prevent the network from overfitting and memorizing the exact details of images, image augmentation is employed to expand image dataset further. The detailed methods of image augmentation include a combination of resizing, rotation, reflection, shear, translation transformations and adding noise. The number of images is expanded to 6132. 80% images in the dataset are used as the training set and other images are used as the testing set.

B. THE STRUCTURE OF IMPROVED MASK RCNN

Mask RCNN is an object instance segmentation method proposed by He *et al.* [29], which extends from Faster RCNN by adding a branch for predicting an object mask in parallel with the original branch for predicting a bounding box of an object. The main framework of original Mask RCNN is not changed and a minor improvement is proposed to adapt original framework to the detection of cucumber fruits in our study. The framework of improved Mask RCNN for pixel-wise segmentation of cucumber fruits is shown as Fig. 2. A cucumber image is first input to the convolutional backbone that can extract image features and output a feature map. The FPN is employed to improve backbone, which helps to extract features from different scales and detect objects with different sizes. Secondly, RPN will output region of interests (RoIs) based on the output feature map in the previous step. RoIs are shown as the green rectangles on the feature map in Fig. 2. Then, the RoIAlign method is used to reshape RoIs into a fixed size, which is used to replace RoI Pooling in Faster RCNN. RoIAlign improves the operation of spatial quantization and contributes to the pixel-to-pixel alignment between network inputs and outputs. Finally, the RoIs with fixed size will flow to two different branches. The original branches can predict cucumber fruits by regression and classification. The new branch is mask branch that is a small fully convolutional network (FCN) and usually applied to semantic segmentation. The predicted mask is shown as the

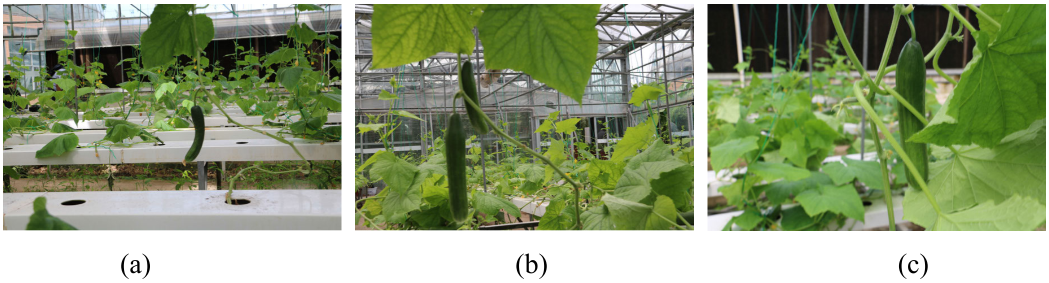


FIGURE 1. Cucumber images taken in a greenhouse.

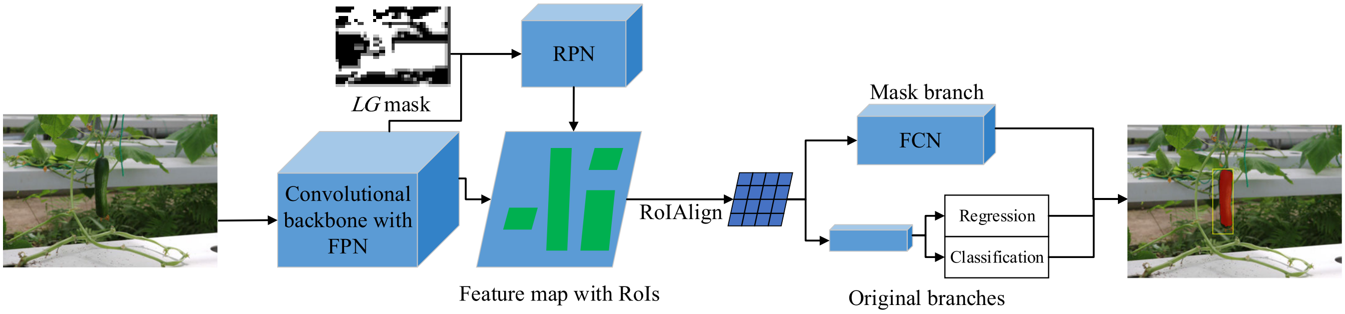


FIGURE 2. The improved Mask RCNN framework for pixel-wise segmentation of cucumber fruits.

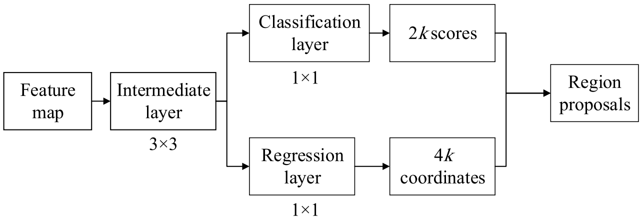


FIGURE 3. The structure of RPN.

red region in Fig. 2. The yellow rectangle box is predicted by original branches. The *LG* mask does not belong to original Mask RCNN, which is added into original framework. It is produced by *LG* operator and used to improve the RPN in original Mask RCNN.

Compared to Faster RCNN, the loss function of Mask RCNN also has to be changed because of the added branch. The new loss function consists of three components, which is shown as Eq. 1. The improved Mask RCNN adopts the same loss function with original Mask RCNN.

$$Loss = L_{cls} + L_{box} + L_{mask} \tag{1}$$

where $Loss$ is the loss function of Mask RCNN. L_{cls} is the loss of classification. L_{box} is the regression loss of bounding boxes. L_{mask} is a cross-entropy loss of predicted masks.

C. IMPROVEMENTS OF RPN

RPN is first proposed in Faster RCNN, which is used to propose object regions with higher possibility. The RPN structure in Mask RCNN is shown as Fig. 3, which is similar to original branches in Fig. 2. Firstly, each point on the feature map is used to generate k ($k = 15$) anchor boxes with 5 scales and 3 aspect ratios on the input image. Meanwhile, two 1×1 convolutional layers, a classification layer and a regression layer, are generated from the feature map and an intermediate layer. The classification layer outputs $2k$ scores that estimate the probability of object or not object for each box. The regression layer outputs $4k$ coordinates of boxes. Next, all anchor boxes are sorted and filtered based on these scores and coordinates. Finally, region proposals are predicted by non-maximum suppression (NMS) for remaining boxes.

In practice, RPN in Mask RCNN is not efficient and precision enough. For example, a feature map of a size 40×60 will generate more than 30 thousand anchor boxes. However, a great number of anchor boxes are located in the background and need to be filtered in the next steps. Therefore, it is necessary to adjust the structure of RPN to detect cucumber fruits effectively.

Cucumber fruits have the same color with leaves and branches. They are green and different from the background. An improvement of RPN is proposed based on the color feature of cucumbers. A simple color operator is employed to filter non-green background and generate a binary mask image. Next, the mask image is mapped to the feature map. If a point of feature map is on non-green background, the point will not generate anchor boxes.

The color operator *EXG*(Excess green) is generally used to extract green objects from the background [30]. In this study, a simpler operator *LG* (Logical green) is proposed to replace *EXG* operator, which makes up of logical operations and color components in RGB color space. The equations of *EXG* and *LG* are shown as Eq.2 and Eq. 3 respectively. The Eq. 2 shows that *EXG* operator involves arithmetic operations and the Eq. 3 shows that *LG* operator only involves simple

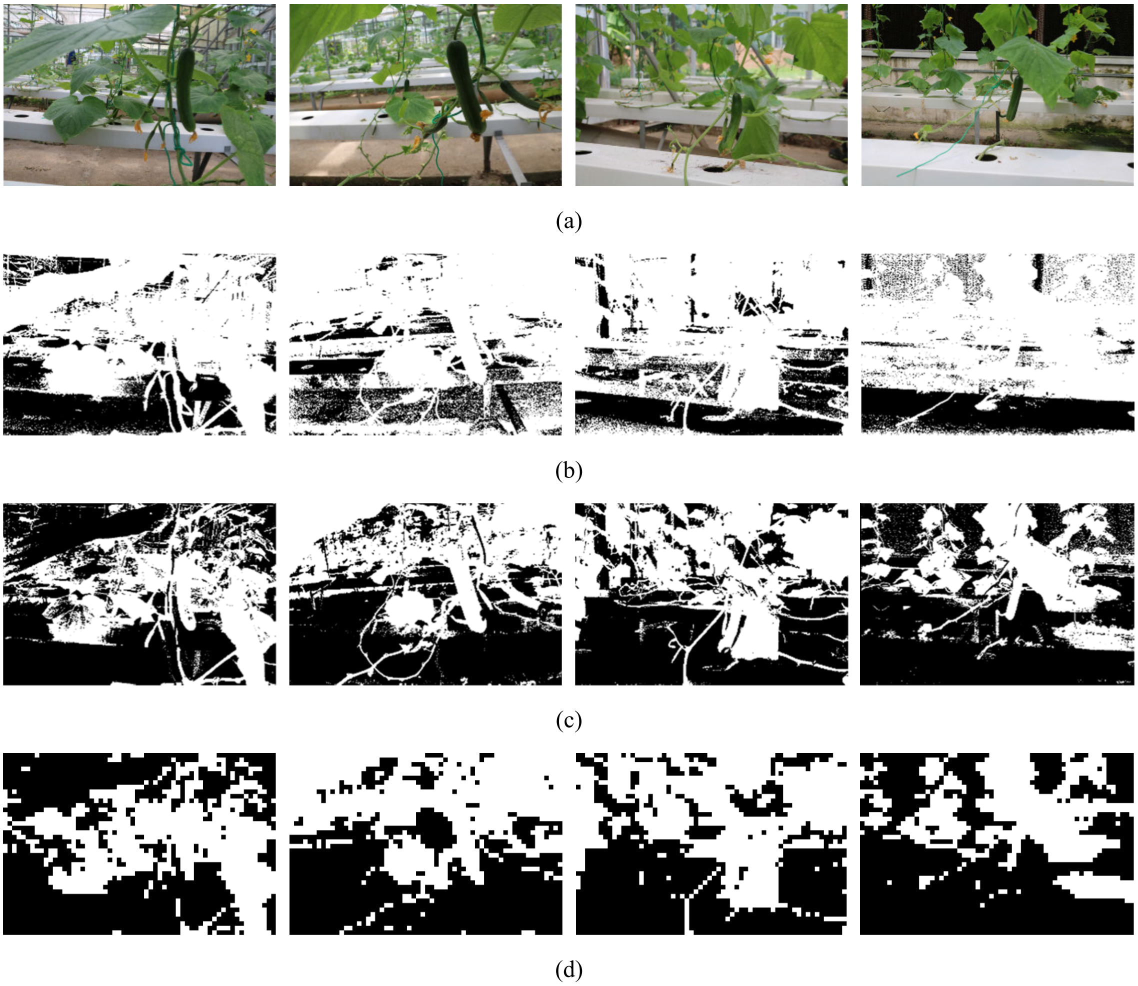


FIGURE 4. Examples of binary mask images. (a) Original images. (b) Binary mask images generated by *EXG*. (c) Binary mask images generated by *LG*. (d) Binary masks of feature maps.

logical operations.

$$EXG = 2G - R - B \tag{2}$$

$$LG = (G > R)\,\&\,(R > B) \tag{3}$$

where R, G, and B are the red, green, and blue components in RGB color space respectively.

The detailed steps of filtering non-green background with LG operator are stated as follow. Firstly, the operator can generate a grayscale image that has distinct intensity difference between objects and background. Then, OTSU method is employed to determine an optimal threshold. Finally, the optimal threshold is used to segment the grayscale image into a binary mask image.

The OTSU method is a threshold segmentation algorithm, which maximizes the between-class variance and is applied in statistical discriminant analysis widely. In a digital image, there are L distinct intensity levels. A selected threshold Th, $0 < Th < L - 1$, is used to separate all pixels of an input image into two classes, C_0 and C_1. The probability P_0 of class C_0 is shown as Eq. 4, which is the ratio of the number of the pixels in class C_0 to the number of all pixels in the input image. The probability P_1 of class C_1 is shown as Eq. 5, which is similar to the equation of P_0. The between-class variance σ_B^2 is defined as Eq. 6. If a value of Th can make σ_B^2 reach its maximum value, the value of Th is the optimal threshold.

$$P_0 = \sum_{i=0}^{Th-1} \frac{n_i}{N} \tag{4}$$

$$P_1 = \sum_{i=Th}^{L-1} \frac{n_i}{N} \tag{5}$$

$$\sigma_B^2 = P_0 P_1 (m_0 - m_1)^2 \tag{6}$$

where n_i denotes the number of pixels with intensity i. N is the number of all pixels in an input image. m_0 is the mean intensity value of the pixels assigned to class C_0 and m_1 is the mean intensity value of the pixels assigned to class C_1.

Four examples of binary mask images generated by EXG and LG are shown in Fig. 4b and Fig. 4c respectively. The white regions in images are green. In the background of greenhouses, EXG cannot segment green regions effectively and some non-green regions in the background are also classified as green objects. These examples indicate that LG has a better performance in a greenhouse. Therefore, the operator LG is adopted to filter the non-green background. Next, these

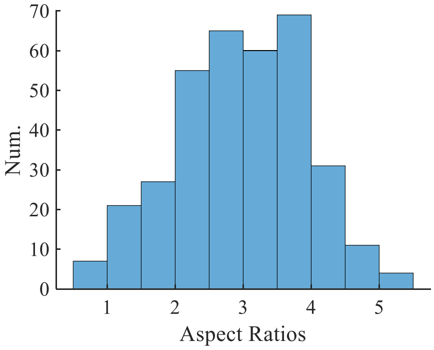


FIGURE 5. Statistical data of aspect ratios.

mask images generated by *LG* are mapped to the same size of feature maps, which are shown in Fig 4d.

The shape of cucumber fruits is different from other objects. Therefore, aspect ratios of anchor boxes are also need to be changed, so that they can fit the shape of cucumber fruits. In RPN of original Mask RCNN, there are 15 kinds of anchor boxes that include 3 different aspect ratios. The aspect ratios that are 1:2, 1:1 and 2:1 need to be changed as appropriate values. To design a set of aspect ratios for cucumber fruits, 350 labeled cucumber fruits are selected randomly. The statistical data of their aspect ratios are shown in Fig. 5. The x axis is used to show aspect ratios that are divided into 12 bins in [0, 6]. The y axis is used to show the number of cucumber fruits in each bin. Figure 5 indicates that more than 90% of aspect ratios range from 1:1 to 5:1. Finally, 2:1 and 4:1 are selected as the aspect ratios of anchor boxes in improved RPN.

The original scales of anchor boxes are set as 32, 64, 128, 256 and 512, which also do not fit the size of cucumber fruits. To determine appropriate scales of anchor boxes, it is necessary to analyze the size of cucumber fruits in resized images with the resolution 600 × 400. Firstly, the bounding rectangle of each fruit is determined based on corresponding labeled region, because anchor boxes also are rectangles. Then, areas of 350 bounding rectangles are calculated and the distribution of areas are shown in Fig. 6. The areas distribute in 26 bins from 0 to 26000 and the size of each bin is 1000. The areas of 90% of rectangle boxes are less than 12000 pixels. Therefore, the scales of anchor boxes in improved RPN are set as 32, 64 and 128. In general, there are 6 different kinds of anchor boxes including 2 different aspect ratios and 3 different scales. The Mask RCNN with improved RPN is called improved Mask RCNN.

III. RESULTS AND DISCUSSIONS

A. TRAINING AND TESTING MASK RCNN

In addition to improved RPN, the selection of backbone is an important factor to affect the precision of cucumber detection. In this experiment, Resnet-101 is selected as the backbone of improved Mask RCNN. It adopts residual network structure and has more convolution layers than other common backbones, such as Resnet-50, VGG-19 and GoogLeNet etc. Therefore, Resnet-101 can extract deeper semantic features. The improved Mask RCNN was trained and tested on TensorFlow platform and a personal computer (PC). The PC has an Intel Xeon E5-2678 CPU, 64 GB RAM and 4 GPUs (NVIDIA GeForce GTX 1080 Ti).

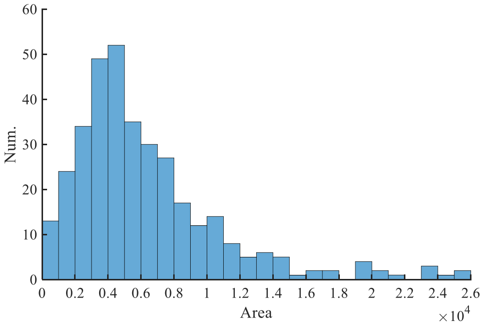


FIGURE 6. The distribution of fruit areas.

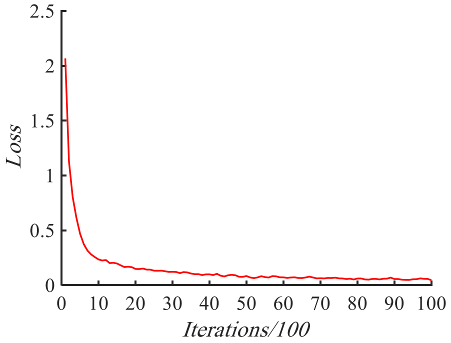


FIGURE 7. Training loss of improved Mask RCNN.

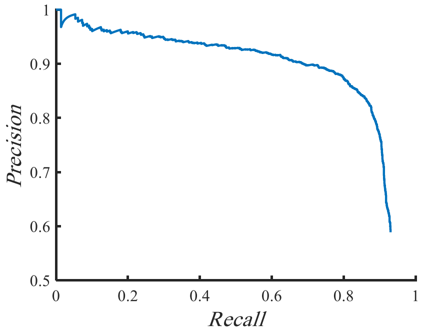


FIGURE 8. P-R curve of improved Mask RCNN.

The image-centric training method was adopted to train improved Mask RCNN model. The initial learning rate was set as 0.001 and learning momentum was set as 0.9. The size of mini-batch is set as 32. The weight decay was set as 0.0001. The threshold of *IoU* (Intersection over Union) was set as 0.7. The total number of iterations was set as 10000. The training loss of Mask RCNN is shown in Fig.7. It indicates that the value of loss decreases rapidly from 1 to 1000 iterations and then decreases slowly in the next iterations.

1226 images in expanded image dataset are used to test the performance of the trained improved Mask RCNN. The P-R curve is shown in Fig.8. It indicates that the trained model can reach enough detection accuracy. Detection results of 4 images taken in a greenhouse are shown in Fig.9. The

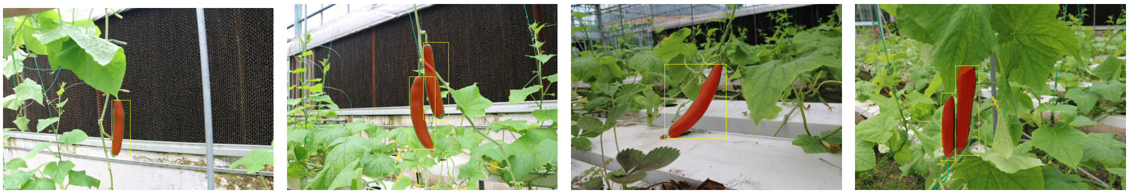


FIGURE 9. Examples of detection results by Mask RCNN.

red masks on these images are detected by the mask branch in improved Mask RCNN and the yellow rectangles are determined by classification and regression branches. Fig. 9 indicates that trained improved Mask RCNN model can segment cucumber fruits in pixel level. Most of the fruit pixels are segmented precisely, but a part of pixels on the edge of fruits are segmented falsely.

B. COMPARISONS OF FRUIT DETECTION

The Mask RCNN adopted in this study is improved based on the features of cucumber fruits. To validate the performance of improved Mask RCNN further, it is compared with original Mask RCNN, Faster RCNN, YOLO V2 and YOLO V3. Improved Mask RCNN, original Mask RCNN and Faster RCNN are all two-stage convolutional neural networks for object detection. YOLO is one of one-stage object detection methods, which predicts bounding boxes and class probabilities by framing object detection as a regression problem. It can detect images in real-time and is faster than Faster RCNN. YOLO V2 adopts Darknet-19 that is similar to the VGG network as its backbone. YOLO V3 adopts Darknet-53 that is similar to the Resnet as its backbone. Besides, YOLO V3 also adopts the structure of FPN to implement multi-scale object detection. Compared with YOLO V2, YOLO V3 can detect objects with higher accuracy.

In the comparative experiment, original Mask RCNN and Faster RCNN both adopted resnet-101 as their backbones, which is same as the backbone of improved Mask RCNN. Besides, they also adopted the same hyper-parameters to train their models. The structure of YOLO is different from them. Therefore, the hyper-parameters of YOLO V2 and V3 were slightly different. Firstly, all sample images in training and testing sets are resized to 416 × 416 so that they can input the network of YOLO. The initial learning rate was set as 0.001 and learning momentum was set as 0.9. The size of mini-batch is set as 128. The weight decay was set as 0.0005. The total number of iterations was also set as 10000.

The variables, *Precision* (Eq. 7), *Recall* (Eq. 8), F_1 (Eq. 9) and T, are used to describe the performances of the 5 algorithms. The variable T is average elapsed time of detecting an image.

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

$$F_1 = \frac{2^{*}Precision^{*}Recall}{Precision + Recall} \tag{9}$$

TABLE 1. Detection results of 5 methods.

Method	*Precision* (%)	*Recall* (%)	F_1 (%)	T(s)
Faster RCNN	85.80	83.66	84.72	0.2908
Original Mask RCNN	88.72	86.35	87.52	0.3872
Improved Mask RCNN	90.68	88.29	89.47	0.3461
YOLO V2	81.84	76.25	78.95	0.0288
YOLO V3	86.27	81.63	83.89	0.0327

where *TP* means true positive that is the number of fruits detected correctly. *FP* means false positive that is the number of other objects detected as fruits. *FN* means false negative that is the number of fruits detected falsely. '*TP*+*FP*' means all detected fruits and '*TP*+*FN*' means all fruits in images. The variable F_1 is used to measure the performance of these methods by balancing the weights of *Precision* and *Recall*.

1226 (20%) testing images are used to test these methods, the test results are listed in Table 1. Compared with the one-stage object detection methods, two-stage methods have a better performance. Although the precision rate of YOLO V3 is slightly higher than that of Faster RCNN, the F_1 score of Faster RCNN is yet higher than that of YOLO V3. However, the average elapsed times of one-stage methods are much less than two-stage methods.

In all two-stage methods, the performance of Faster RCNN is worst. In addition to the mask branch, the improvements of FPN and RoIAlign make the performance of original Mask RCNN better than Faster RCNN and also make its elapsed time more than Faster RCNN. Table 1 shows that the precision rate of improved Mask RCNN is 90.68% and the recall rate is 88.29%. Compared with the F_1 scores of other two-stage methods, the performance of improved Mask RCNN is best. Besides, average elapsed time of improved Mask RCNN is a little less than that of original Mask RCNN for the improvements in RPN, but it is also more than that of Faster RCNN.

C. LOCATION OF CENTRAL POINTS

In this study, the instance segmentation is used to replace object detection methods, because the central points of cucumber fruits are difficult to be located accurately in horizontal direction for their narrow shape. The object detection methods detect a fruit by a rectangle box and instance segmentation methods detect a fruit in pixel level. To evaluate

TABLE 2. Statistical values of location deviation.

Location deviation	YOLO V2	YOLO V3	Faster RCNN	Original Mask RCNN	Improved Mask RCNN
Mean value	6.12	5.74	5.88	3.37	2.10
Standard value	5.57	4.28	4.47	2.70	1.73

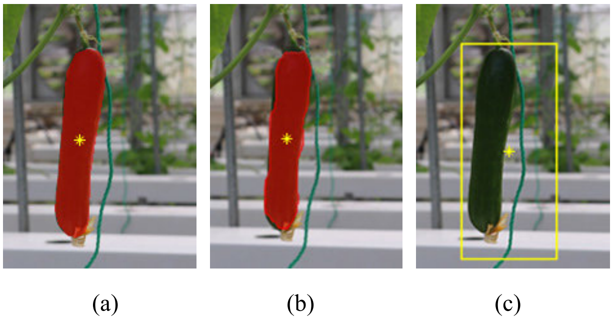


FIGURE 10. Location of central points. (a) Labeled fruit in pixel level and its central point. (b) Detected fruits in pixel level and its central point. (c) Detected fruits by a rectangle box and its central point.

location accuracy of different methods, the central points of fruits produced by different methods are compared with that produced by labeled masks. The asterisks in Fig. 10 are the central points of fruits. The red region on the fruit in Fig. 10a is labeled manually and the red region in Fig. 10b is detected by original Mask RCNN or improved Mask RCNN. Besides, the horizontal and vertical coordinates of central point are the mean values of the coordinates of all pixels in red region. The rectangle box in Fig. 10c is produced by Faster RCNN, YOLO V2 or YOLO V3 and the central point of the fruit is determined by the center of the rectangle box.

The central points produced by labeled masks are called reference points. Reference points and corresponding central points produced by different methods are put in the same image coordinate system and then the Euclidean distance from a reference point to each corresponding central point is calculated. The distance is called location deviation. Fruits in 1226 (20%) labeled testing images have been detected by the 5 methods in previous subsection. The location deviations of these fruits are calculated in the experiment of this subsection. The statistical values of location deviation are shown in Table 2.

Table 2 shows the mean values and standard deviations of location deviations. YOLO V2, YOLO V3 and Faster RCNN all adopt rectangle boxes to locate objects, but the mean value and standard value of YOLO V3 is lower than YOLO V2 and Faster RCNN. It indicates that the location accuracy of YOLO V3 is higher than YOLO V2 and Faster RCNN because YOLO V3 can detect multi-scale objects. The mean values and standard deviations of original Mask RCNN and improved Mask RCNN are both lower than that of other methods, which indicates fruit detection in pixel level can reach higher location accuracy than fruit detection by a rectangle box. Meanwhile, improved Mask RCNN can reach higher location accuracy than original Mask RCNN, because its improvements are efficient to improve location accuracy.

IV. CONCLUSION

In this study, an improved Mask RCNN is proposed to detect cucumber fruits in pixel level. Cucumber fruits are difficult to be detected effectively by the traditional object detection methods for their special color and shape. Improved Mask RCNN not only can detect fruits effectively but also can reach high location accuracy.

(1) In consideration of the features of cucumber fruits, the RPN in original Mask RCNN is improved. The *LG* operator is proposed to filter non-green background and limit anchor boxes in the green regions. Besides, the scales and aspect ratios of anchor boxes are adjusted to fit the shape and size of cucumber fruits.

(2) The trained model of improved Mask RCNN has a good performance on testing images. The test result shows that the precision and recall rates are 90.68% and 88.29% respectively. However, its average elapsed time is slightly slow and cannot realize real-time detection. The slow time is mainly caused by the two-stage Faster RCNN structure. The further study is to apply efficient one-stage structure and FCN to realize instance segmentation.

(3) The location of cucumbers is important for picking fruits by robots. Fruit detection by a rectangle box cannot locate fruits in high precision for long and narrow shape. The location accuracy of improved Mask RCNN is not only higher than Faster RCNN, YOLO V2 and YOLO V3 but also higher than original Mask RCNN.

REFERENCES

[1] W. Ji, X. Meng, Y. Tao, B. Xu, and D. Zhao, "Fast segmentation of colour apple image under all-weather natural conditions for vision recognition of picking robots," *Int. J. Adv. Robot. Syst.*, vol. 13, p. 24, Feb. 2016.

[2] W. Ji, G. Chen, B. Xu, X. Meng, and D. Zhao, "Recognition method of green pepper in greenhouse based on least-squares support vector machine optimized by the improved particle swarm optimization," *IEEE Access*, vol. 7, pp. 119742–119754, 2019. doi: 10.1109/ACCESS.2019.2937326.

[3] Z. De-An, L. Jidong, J. Wei, Z. Ying, and C. Yu, "Design and control of an apple harvesting robot," *Biosyst. Eng.*, vol. 110, no. 2, pp. 112–122, Oct. 2011.

[4] X. Liu, D. Zhao, W. Jia, W. Ji, and Y. Sun, "A detection method for apple fruits based on color and shape features," *IEEE Access*, vol. 7, pp. 67923–67933, 2019.

[5] X. Liu, D. Zhao, W. Jia, C. Ruan, S. Tang, and T. Shen, "A method of segmenting apples at night based on color and position information," *Comput. Electron. Agricult.*, vol. 122, pp. 118–123, Mar. 2016.

[6] X. Liu, W. Jia, C. Ruan, D. Zhao, Y. Gu, and W. Chen, "The recognition of apple fruits in plastic bags based on block classification," *Precis. Agricult.*, vol. 19, no. 4, pp. 735–749, Aug. 2018.

[7] L. Fu, B. Wang, Y. Cui, S. Su, Y. Gejima, and T. Kobayashi, "Kiwifruit recognition at nighttime using artificial lighting based on machine vision," *Int. J. Agricult. Biol. Eng.*, vol. 8, no. 4, pp. 52–59, Aug. 2015.

[8] L. Fu, E. Tola, A. Al-Mallahi, R. Li, and Y. Cui, "A novel image processing algorithm to separate linearly clustered kiwifruits," *Biosyst. Eng.*, vol. 183, pp. 184–195, Jul. 2019.

[9] J. Xiong, Z. Liu, L. Tang, R. Lin, R. Bu, and H. Peng, "Visual detection technology of green citrus under natural environment," *Trans. Chin. Soc. Agricult. Mach.*, vol. 49, no. 4, pp. 45–52, Apr. 2018.

[10] X. Ling, Y. Zhao, L. Gong, C. Liu, and T. Wang, "Dual-arm cooperation and implementing for robotic harvesting tomato using binocular vision," *Robot. Auton. Syst.*, vol. 114, pp. 134–143, Apr. 2019.

[11] J. Xiong, Z. He, R. Lin, Z. Liu, R. Bu, Z. Yang, H. Peng, and X. Zou, "Visual positioning technology of picking robots for dynamic litchi clusters with disturbance," *Comput. Electron. Agricult.*, vol. 151, pp. 226–237, Aug. 2018.

[12] H. Li, M. Huang, Q. Zhu, and Y. Guo, "Peduncle detection of sweet pepper based on color and 3D feature," in *Proc. ASABE Annu. Int. Meeting*, Detroit, MI, USA, 2018, p. 1.

[13] L. Zhang, Q. Yang, Y. Xun, X. Chen, Y. Ren, T. Yuan, Y. Tan, and W. Li, "Recognition of greenhouse cucumber fruit using computer vision," *New Zland J. Agricult. Res.*, vol. 50, no. 5, pp. 1293–1298, Feb. 2010.

[14] H. Wang, C. Ji, B. Gu, and Q. An, "In-greenhouse cucumber recognition based on machine vision and least squares support vector machine," *Trans. Chin. Soc. Agricult. Mach.*, vol. 43, no. 3, pp. 163–180, Mar. 2012.

[15] T. Yuan, C. Ji, Y. Chen, W. Li, and J. Zhang, "Greenhouse cucumber recognition based on spectral imaging technology," *Trans. Chin. Soc. Agricult. Mach.*, vol. 42, pp. 172–176, Nov. 2011.

[16] T. Yuan, C. Xu, Y. Ren, Q. Feng, Y. Tan, and W. Li, "Detecting the information of cucumber in greenhouse for picking based on NIR image," *Spectrosc. Spectral Anal.*, vol. 29, no. 8, pp. 2054–2058, Aug. 2009.

[17] G. Bao, S. Cai, L. Qi, Y. Xun, L. Zhang, and Q. Yang, "Multi-template matching algorithm for cucumber recognition in natural environment," *Comput. Electron. Agricult.*, vol. 127, pp. 754–762, Sep. 2016.

[18] Y. Tao, J. Zhou, K. Wang, and W. Shen, "Rapid detection of fruits in orchard scene based on deep neural network," in *Proc. ASABE Annu. Int. Meeting*, Detroit, MI, USA, 2018, p. 1.

[19] M. Halstead, C. McCool, S. Denman, T. Perez, and C. Fookes, "Fruit quantity and ripeness estimation using a robotic vision system," *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 2995–3002, Oct. 2018.

[20] Y. Tian, G. Yang, Z. Wang, H. Wang, E. Li, and Z. Liang, "Apple detection during different growth stages in orchards using the improved YOLO-V3 model," *Comput. Electron. Agricult.*, vol. 157, pp. 417–426, Feb. 2019.

[21] N. Lamb and M. Chuah, "A strawberry detection system using convolutional neural networks," in *Proc. IEEE Int. Conf. Big Data*, Seattle, WA, USA, Dec. 2018, pp. 2515–2520.

[22] X.-Y. Zhang, S. Wang, and X. Yun, "Bidirectional active learning: A two-way exploration into unlabeled and labeled data set," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 12, pp. 3034–3044, Dec. 2015.

[23] X. Zhang, S. Wang, X. Zhu, X. Yun, G. Wu, and Y. Wang, "Update vs. upgrade: Modeling with indeterminate multi-class active learning," *Neurocomputing*, vol. 162, pp. 163–170, Aug. 2015.

[24] X.-Y. Zhang, H. Shi, X. Zhu, and P. Li, "Active semi-supervised learning based on self-expressive correlation with generative adversarial networks," *Neurocomputing*, vol. 345, pp. 103–113, Jun. 2019.

[25] Y. Yu, K. Zhang, L. Yang, and D. Zhang, "Fruit detection for strawberry harvesting robot in non-structural environment based on mask-RCNN," *Comput. Electron. Agricult.*, vol. 163, Aug. 2019, Art. no. 104846.

[26] W. Ji, Z. Qian, B. Xu, G. Chen, and D. Zhao, "Apple viscoelastic complex model for bruise damage analysis in constant velocity grasping by gripper," *Comput. Electron. Agricult.*, vol. 162, pp. 907–920, Jul. 2019.

[27] W. Ji, X. Meng, Z. Qian, B. Xu, and D. Zhao, "Branch localization method based on the skeleton feature extraction and stereo matching for apple harvesting robot," *Int. J. Adv. Robot. Syst.*, vol. 14, no. 3, pp. 1–9, May 2017.

[28] W. Ji, Z. Qian, B. Xu, Y. Tao, D. Zhao, and S. Ding, "Apple tree branch segmentation from images with small gray-level difference for agricultural harvesting robot," *Optik*, vol. 127, pp. 11173–11182, Dec. 2016.

[29] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 2961–2969.

[30] H. O. Cruz, M. Eckert, J. M. Meneses, and J. F. Martínez, "Precise real-time detection of nonforested areas with UAVs," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 2, pp. 632–644, Feb. 2017.

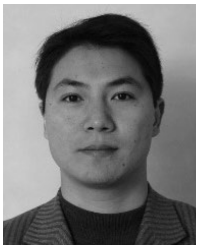
XIAOYANG LIU received the bachelor's degree from the School of Electrical and Information Engineering, Jiangsu University, in 2014, where he is currently pursuing the Ph.D. degree under the supervision of Prof. D. Zhao. His research interests include object detection in agriculture that involves computer vision, machine learning, and deep learning.

DEAN ZHAO is currently a Professor with Jiangsu University and the Executive Director of the Automation Institute, Jiangsu Province. His main research interests include robot control technology, agriculture bio-information control technology, and agricultural machinery control technology. He is also the Deputy Director of professional committee members of the agricultural electrification and automation of Chinese society of agricultural engineering.

WEIKUAN JIA was born in 1982. He received the Ph.D. degree. He is currently a Lecturer and a Supervisor of the M.A. degree with Shandong Normal University. His main research interests include artificial intelligence, smart agriculture, and robot control technology.

WEI JI was born in Henan, China, in 1974. He received the Ph.D. degree in electrical engineering from Southeast University, Nanjing, China, in 2007. Since 2007, he has been with the School of Electrical and Information Engineering, Jiangsu University, Zhenjiang, China, where he is currently an Associate Professor. His current research interests include robot motion control and intelligent control.

CHENGZHI RUAN received the B.S. and M.S. degrees from the Anhui University of Technology, Ma'anshan, China, in 2007 and 2010, respectively, and the Ph.D. degree in control science and engineering from Jiangsu University, Zhenjiang, China, in 2018. He is currently an Associate Professor with the School of Mechanical and Electrical Engineering, Wuyi University, Wuyishan, China. His current research interests include agricultural robot, image processing, and automatic control.



YUEPING SUN was born in Changzhou, China, in 1982. He received the Ph.D. degree from Jiangsu University, Zhenjiang, China, in 2016, where he has been a Lecturer with the School of Electrical and Information Engineering, since 2004. His current research interests include motor drives, motor movement control, hybrid electric vehicles, and intelligent control.