ORIGINAL RESEARCH
published: 09 June 2022
doi: 10.3389/fpls.2022.765523

YOLOF-Snake: An Efficient Segmentation Model for Green Object Fruit

Weikuan Jia[1,2]*, *Mengyuan Liu*[1], *Rong Luo*[3], *Chongjing Wang*[4], *Ningning Pan*[1], *Xinbo Yang*[1] and *Xintong Ge*[1,5]*

[1] School of Information Science and Engineering, Shandong Normal University, Jinan, China, [2] Key Laboratory of Facility Agriculture Measurement and Control Technology and Equipment of Machinery Industry, Zhenjiang, China, [3] School of Light Industry Science and Engineering, Qilu University of Technology, Shandong Academy of Sciences, Jinan, China, [4] China Academy of Information and Communications Technology, Beijing, China, [5] School of Medical Imaging, Xuzhou Medical University, Xuzhou, China

OPEN ACCESS

Edited by:
Roger Deal,
Emory University, United States

Reviewed by:
Marcin Wozniak,
Silesian University of Technology, Poland
Arpit Jain,
WhiteHat Jr. Education Technology Pvt., Ltd., India
Wei Ji,
Jiangsu University, China

****Correspondence:***
Weikuan Jia
jwk_1982@163.com
Xinting Ge
xintingge@163.com

Specialty section:
This article was submitted to Technical Advances in Plant Science, a section of the journal Frontiers in Plant Science

Received: *27 August 2021*
Accepted: *29 April 2022*
Published: *09 June 2022*

Citation:
Jia W, Liu M, Luo R, Wang C, Pan N, Yang X and Ge X (2022) YOLOF-Snake: An Efficient Segmentation Model for Green Object Fruit. Front. Plant Sci. 13:765523. doi: 10.3389/fpls.2022.765523

Accurate detection and segmentation of the object fruit is the key part of orchard production measurement and automated picking. Affected by light, weather, and operating angle, it brings new challenges to the efficient and accurate detection and segmentation of the green object fruit under complex orchard backgrounds. For the green fruit segmentation, an efficient YOLOF-snake segmentation model is proposed. First, the ResNet101 structure is adopted as the backbone network to achieve feature extraction of the green object fruit. Then, the C5 feature maps are expanded with receptive fields and the decoder is used for classification and regression. Besides, the center point in the regression box is employed to get a diamond-shaped structure and fed into an additional Deep-snake network, which is adjusted to the contours of the target fruit to achieve fast and accurate segmentation of green fruit. The experimental results show that YOLOF-snake is sensitive to the green fruit, and the segmentation accuracy and efficiency are significantly improved. The proposed model can effectively extend the application of agricultural equipment and provide theoretical references for other fruits and vegetable segmentation.

Keywords: automatic harvesting, green fruits, YOLOF-snake, deep-snake, fruits segmentation

INTRODUCTION

With the increasing maturity of artificial intelligence technology, new enlightenment is brought in fruit and vegetable production management and automated picking. The level of production automation (Bac et al., 2014; Jia et al., 2020c) and management intelligence (Bochtis et al., 2014; Caicedo Solano et al., 2020) is increased in the fruit and vegetable industry. The operating performance of vision systems directly affects the operational efficiency of agricultural equipment. It plays a vital role in the efficient production of fruits and vegetables (Patrício and Rieder, 2018; Tang et al., 2020; Tian et al., 2020), so achieving accurate segmentation and recognition of the fruit of an object becomes the key to precise yield measurement and automatic harvesting operations in orchards (Koirala et al., 2019; Van Klompenburg et al., 2020). However, due to the interference of branches and leaves, overlapping fruits, light and weather changes, operating angles, and the impact of the same color background of green fruit in the complex orchard environment, the efficient and accurate segmentation of green fruit is subject to many challenges, which attracts the attention of many scholars.

In previous research on object fruit segmentation and recognition, traditional machine learning has played a pivotal role in driving the development of intelligent agriculture (Jha et al., 2019; Sharma et al., 2020; Benos et al., 2021; Tahaseen and Moparthi, 2021). For example, Zhang et al. (2020) proposed a segmentation method combining color and texture features, using grayscale co-occurrence matrix (GLCM) to extract texture features. The recognition accuracy of the new method reached 0.94, but the process is affected by illumination and is not effective in segmenting the fruit of the object under uneven illumination. Besides, the darker area is not clearly segmented. Henila et al. (2020) constructed a fuzzy clustering-based threshold segmentation method for segmenting regions of interest in apple images with the largest cluster of pixels to calculate the threshold value, which has substantially improved the segmentation accuracy and efficiency compared with the grayscale thresholding method. Jia et al. (2020a) proposed an apple image recognition method based on PCNN and GA–Elman fusion. First, PCNN is utilized to implement segmentation of the target image, extract color, and shape features, GA–Elman classifier is designed, and finally, object fruit recognition is implemented. Aiming at the problem of apple fruit recognition with uneven coloring, Liu et al. (2019) proposed a target fruit segmentation algorithm based on superpixel features. The target image is segmented into pixel blocks by SLIC, and the target fruit and background are divided into two categories by SVM. And then, the classification results are further revised according to the adjacency relationship between superpixels. For the shadow problem of strongly illuminated images, Xu et al. (2019) proposed the method of fusing superpixels and edge probability maps to generate superpixel blocks with precise boundaries and using the relighting method to eliminate undetected shadows, which has strong robustness. However, the model training is time-consuming and the process is tedious. These methods have achieved relatively good recognition accuracy and efficiency and provide necessary theoretical and technical support for orchard yield measurement and machine picking (Pallathadka et al., 2021). However, most of these methods are studied for some specific scenarios and overly rely on the fruit color, shape, texture, and other features of the target fruit, which are greatly affected by light, resulting in poor robustness of the algorithms. In complex orchard environments, the extraction of target fruit features poses further difficulties for this identification method and does not meet the needs of agricultural equipment working under challenging situations.

In recent years, with the rapid development of deep learning and computing technology, deep learning algorithms have the advantages of end-to-end automatic detection and deep extraction of image features. Their robustness and accuracy have greatly improved, which are widely used in the fields of object detection and image segmentation (Zhao et al., 2019; Oksuz et al., 2020; Minaee et al., 2021). Inspired by this, deep learning has been increasingly used in agricultural production, such as pest and disease identification (Ushadevi, 2020; Liu and Wang, 2021), fruit and vegetable yield measurement (Yin et al., 2020; Maheswari et al., 2021), and automatic harvesting (Saleem et al., 2021; Yin et al., 2021), which substantially promote the development of agricultural technology. For complex environments, the research of green object fruit recognition based on deep learning theory has attracted the attention of many scholars, such as Farkhani et al. (2021), used a transformer-based multilayer attention procedure combined with fusion rules to accurately classify weeds by high-resolution attention maps, obtaining high accuracy results. Dhaka et al. (2021) studied and reviewed the existing models on the application of deep convolutional neural networks to predict plant diseases and insect pests from leaf images and compared the advantages and disadvantages of different technologies and models. Kundu et al. (2021) proposed an automatic and intelligent data collector and classifier framework that inherited the Internet of Things and deep learning technology. It will send the collected data to the cloud server and use Raspberry Pi to accurately predict the blast and rust diseases in pearl millet. Its classification accuracy can be comparable to the most advanced model, but it reduces the time by 86.67% (Kundu et al., 2021). Woźniak and Połap (2018) proposed developed neural network architecture, AANN, which processes relatively more miner information and makes numerical calculations more accurate. The proposed fusion is effective in the system structure and the training process on classification results (Woźniak and Połap, 2018). Li et al. (2021) optimized the U-Net model by combining the spatial pyramid pooling (ASPP) structure and merging U-Net's edge features and advanced functions. In addition, this model obtained the semantic boundary information of object fruit images by integrating the residual module and closed convolution, which effectively improved the segmentation accuracy of the object fruit (Li et al., 2021). Xiong et al. (2020) used the YOLOv2 model to detect green mango images based on mango images collected by UAVs, and the detection accuracy is only 1.1% compared with the manual measurement error. Mu et al. (2020) combined R-CNN and ResNet101 to design a green tomato detection model and compile the location map of tomatoes to achieve the detection, counting, localization, and size estimation for tomato ripeness detection and yield prediction. Gan et al. (2020) designed a thermal imaging rig to capture images of immature citrus and build a deep learning model for fruit counting by the temperature difference feature between the object fruit and the background. Jia et al. (2020b) improved the instance segmentation model Mask R-CNN to adapt the detection of apple targets by combining ResNet and DenseNet as the feature extraction network of the original model, which significantly improved the detection accuracy of apple targets in overlapping and branch-obscured environments. However, Mask R-CNN segmentation takes a long time and cannot meet the real-time requirements of segmentation. The above recognition models have substantially improved in terms of accuracy and robustness compared with traditional vision methods. However, in the complex orchard environment, the real-time operation capability of agricultural equipment needs to be further improved. Fruit detection and segmentation are essential for future agronomic management. The research method YOLOF-snake focuses on the rapid identification and segmentation of green fruit. It uses a detection method that only one layer of feature maps is used and a segmentation module that iteratively adjusts the contour of the target fruit. Fusion can reduce the running time of the current mainstream fruit

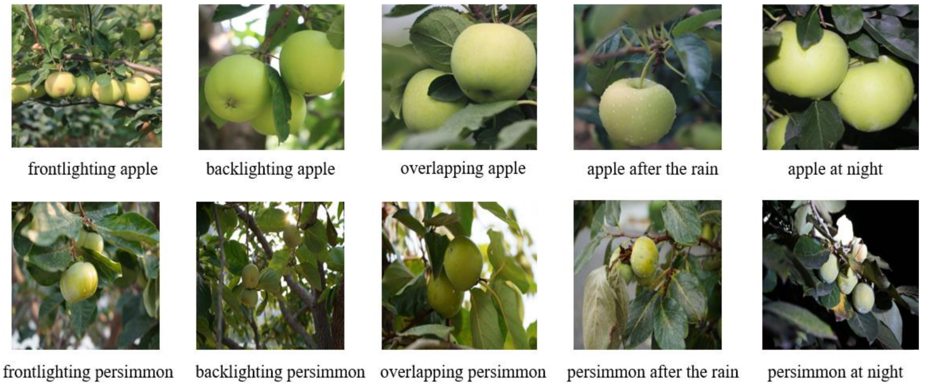

FIGURE 1 | Images of apple and persimmon in front light, backlight, overlap, after rain, and night.

segmentation method by about half. Different from other models, only the fifth-level feature map extracted from the backbone network is used. The fifth-level feature map not only contains enough context information to detect targets of various scales but also shortens the operation of the model detection time. In addition, the model is compared with other mainstream models, and the comparison revealed that the running time of the model is greatly reduced, while accuracy is guaranteed. It is being applied to the vision system of fruit and vegetable picking robots and can also greatly improve the efficiency of fruit and vegetable picking robots. At the same time, since most crop fruits are green during growth, this method can also be applied to the identification and segmentation of other green fruit, such as immature persimmons, immature tomatoes, cucumbers, green peppers, and other crops. Therefore, the method can accurately count the growth cycles of these fruits and simultaneously perform proper variable rate irrigation and fertilization on the monitored growth state or density of the fruits at each stage. At the same time, it improves the efficiency of resource utilization and the quality of the final ripe fruit. In addition, this method can also provide an essential reference for the production estimation of farm operations, fruit trade, retailers, and storage facilities. By detecting and quantifying the distribution of canopy fruits, farmers can obtain valuable information and provide references for optimizing these processes, significantly promoting the temporal and spatial management of agricultural production. As the most basic and vital part of agricultural robots, the vision system is used to analyze the specified targets from complex and diverse scenes, directly affecting the quality and efficiency of fruit picking. The design of the vision system aimed at rapid positioning and precise segmentation will significantly affect the real-time and reliability of the actual application of harvesting robots.

DATA SAMPLE COLLECTION AND PRE-PROCESSING

The images of green apples and persimmons are collected at the Shouguang Agricultural Demonstration Base using a Sony Alpha 7 camera with a resolution of 6000 × 4000. To facilitate the contrast experiments, the original size of 4000 × 4000 pixels is cropped and further reduced to 512 × 512 pixels. Close-up and long-range images are collected to show the adaptability of the model to large and small object fruit, and images of apples and persimmons are also collected under different shading conditions. To obtain accurate results, the images are also acquired in both front-lighting and back-lighting conditions. LabelMe data labeling software is used to label these images. Most miniature outer rectangular boxes of green apples and persimmons are used as ground-truth bounding boxes to reduce the interference of background factors during detection. **Figure 1** shows the green apple and persimmon fruits images that are collected under different lighting conditions.

As can be seen from **Figure 1**, blurred fruit boundaries in overlapping fruits, weak backlight light intensity on the fruit surface, and water droplets on the fruit surface after rain all become factors that affect the accuracy of green fruit recognition and segmentation. For the collection of green fruit images, various factors that may affect fruit recognition and segmentation in the complex orchard background are fully considered. The image samples are collected with maximum consideration of the complex actual environment of the orchard, and 568 persimmon images and 515 apple images are collected and classified into a variety of situations, including overlapping, direct light, after rain, backlight, and night. The details of the sample distribution are shown in **Table 1**.

YOLOF OBJECT DETECTION MODEL

Use of Single-Layer Features

The YOLOF-snake is presented in this article, a fruit segmentation model based on the YOLOF object detection method. The YOLOF algorithm is a rethinking of FPN for single-stage object detection and shows that the success of FPN lies in its separate processing ideas rather than feature fusion ideas. Its outstanding features are fast and accurate. Unlike the object detection models using FPN networks, YOLOF is a simple object detection framework that uses a layer of feature maps.

TABLE 1 | Sample distribution details of green fruit images.

Condition	Overlapping	Direct sunlight	After the rain	Backlight
Number of persimmon images	523	109	113	207
Number of apple images	459	89	103	265

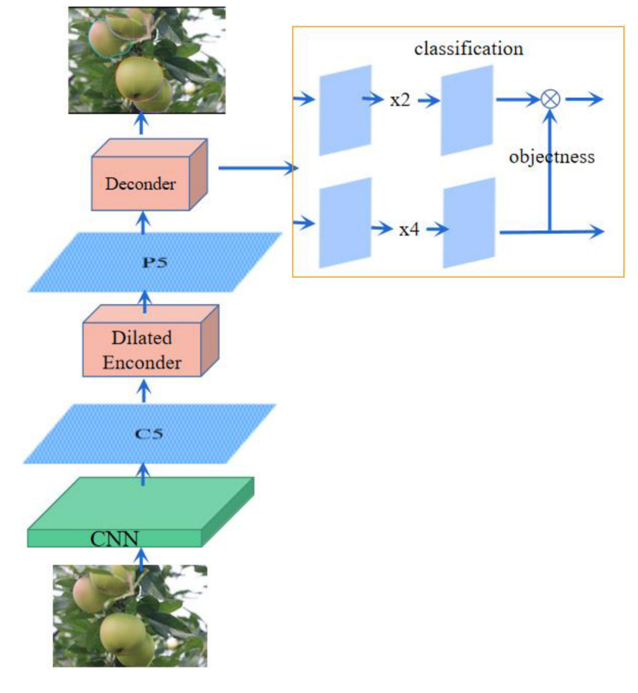

FIGURE 2 | YOLOF structure schematic diagram.

The original ResNet101 network is used, and after the green fruit features are extracted, only one layer of the feature map is used, with an inflated encoder and an equilibrium matching strategy added for object fruit detection. The object fruit class and ground truth are predicted. Three components are included: the backbone network, the encoder, and the decoder. The structure is shown in **Figure 2**.

Large-Scale Variation

For the detection of green fruit, it is a difficult task to detect the object fruit with a large-scale variation. Since only one layer of feature map is used in this method, a restricted sensory field can only be covered by one layer of features, and the mismatch between the sensory field and the object scale will make the detection effect poor. To address this problem, YOLOF first increases the receptive field by stacking the standard convolution and expanding the convolution, but it still cannot cover all the object scales. However, if the original feature map and the feature map after extending the receptive field are integrated using the residual linkage and construction extension module, then in this case, the feature maps of all target scales can be obtained. Finally, feature maps covering all scales of the target fruit can be achieved.

YOLOF-SNAKE FOR GREEN FRUIT SEGMENTATION

The YOLOF-snake algorithm completes target fruit segmentation based on the original YOLOF target fruit detection method, enabling the fruit picking robot to pick green fruits accurately and quickly. YOLOF is an object detection model in which only a 32-fold downsampled C5 feature map is used. The residual module is used to extract multi-scale contextual features for objects at different scales, and the deficiency of multi-scale features is compensated. A balanced matching mechanism is used to solve the positive sample imbalance problem caused by sparse anchors in the single feature map. YOLOF can demonstrate the good results of green fruit recognition, and the real-time performance and accuracy of the green fruit picking work of the picking robot are improved. Therefore, YOLOF is used to implement green fruit recognition, green fruit regression boxes are obtained and fed into an additional snake network, and green fruit segmentation is implemented.

The Overall Structure of YOLOF-Snake

Figure 3 illustrates the overall structure of YOLOF-snake, which consists of four parts.

Backbone Network

The acquired dataset images are fed into the backbone ResNet101 to extract features, and several layers of feature maps are output. Since the fifth layer feature map contains sufficient contextual information for detecting targets of various scales, the fifth layer feature map with a channel count of 2048 and a downsampling rate of 32 is selected.

Encoder Structure

The output C5 layer feature map is used by the encoder to expand the receptive field, matching the sensory field to the target scale. Two modules are included, the projection layer module and the residual module. The projection layer module consists of two parts, a 1 × 1 convolutional layer is used to reduce the size of the channels and a 3 × 3 convolutional layer, is used to refine the semantic information of the context to obtain a feature map of 512 channels. The residual module generates output features with multiple receptive fields that can cover all scales of the target fruit. The residual module superimposes four consecutive residual blocks with different convolution kernel expansion rates, each consisting of three convolutions. The first layer is a 1 × 1 convolution, after the first convolution channel is reduced to 128, then a 3 × 3 convolution expansion convolution is used to increase the receptive field, and finally, the channel size is recovered by a 1 × 1 convolution. The specific structure is shown in **Figure 4**.

Decoder Structure

The decoder classifies and regresses the output of the encoder and then inputs the regression box into the Deep-snake network for target fruit segmentation. The decoder contains two branches, similar to the classification and regression branches of RetinaNet. To achieve better recognition, it has two improvements. First,

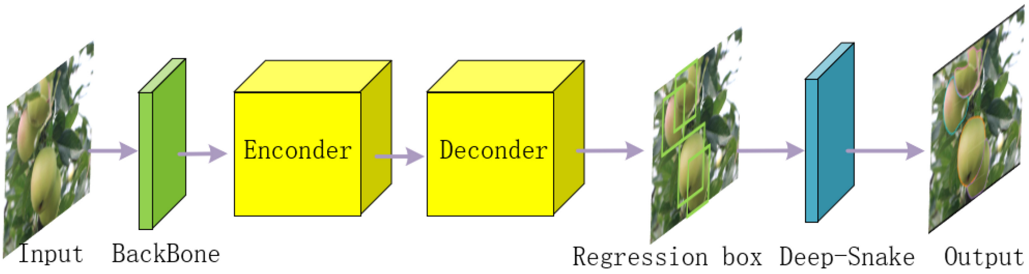

FIGURE 3 | Overall structure of YOLOF-snake. 1. Output the feature map based on the underlying backbone. 2. Enter the encoder. 3. Decoder to obtain the classification results and regression box. 4. Deep-snake network to segment the regression boxes for the object fruit.

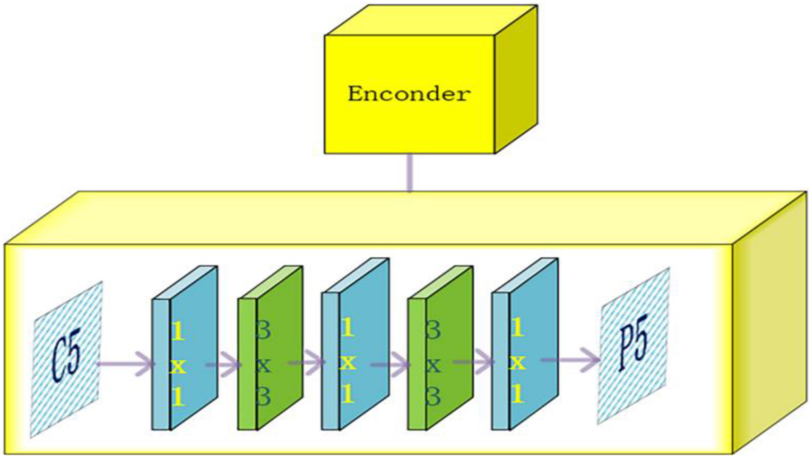

FIGURE 4 | Encoder structure diagram.

the different numbers of convolutional layers in the regression branch and classification branch are set, in the regression branch, four convolutional layers plus batch normalization and activation function layers are used, while in the classification branch two convolutional layers are set for regression. Second, an implicit object indicator is added to each anchor of the regression branch, and the final classification result is the result of multiplying the output of the classification branch and the score of the object indicator.

Segmentation Model

The regression box is taken to the midpoints, and a diamond shape is obtained after connecting the midpoints. Deep-snake network performs offset prediction for four points, the object is the extreme value point around the object, and the center of the extreme value point is extended uniformly to the two directions of the edge where the extreme value point is located. The contour is adjusted iteratively by Deep-snake until it overlaps with the boundary of the object fruit.

Deep-Snake Instance Segmentation Method

As the YOLOF method only detects and classifies the object fruit, it cannot accurately determine the location of the object fruit, which makes the green fruit picking robot cannot accurately and quickly carry out the picking process. Therefore, the contour-based instance segmentation method Deep-snake algorithm module is embedded after the YOLOF regression branch to achieve the segmentation results of the object fruit. It allows the green fruit picking robot to locate the object fruit quickly and accurately and has more effective results of occluded and overlapping fruits.

First, the regression box of the target fruit is obtained by the YOLOF method before inputting the Deep-snake module, and the regression box is gradually optimized as the boundary of the target fruit to carry out the segmentation of the target fruit. Second, for the learning of vertex features, the circular convolution is used to implement the topology, which helps to optimize the contours of green fruits, and to prevent the Deep-snake algorithm from falling into a local optimum solution, how to fine-tune the contours being learned directly from the data, as shown detail in **Figure 5**.

In traditional instance segmentation methods, the vertex coordinates are treated as variables to optimize the manually designed energy function and the target boundary is fitted by minimizing the energy function. The energy function is usually concave and requires manual design based on low-dimensional image features, which usually leads to locally optimal solutions.

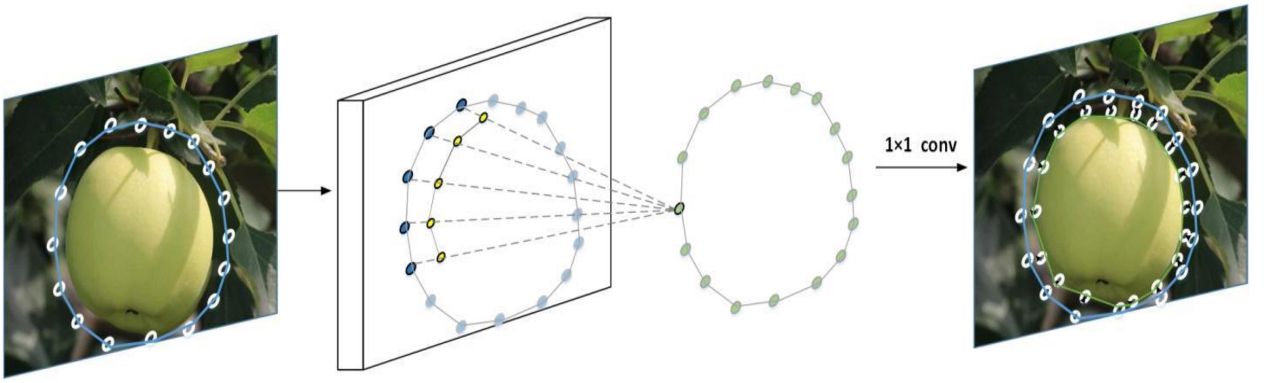

FIGURE 5 | Fine-tuning of the profile of the apple fruit.

Instead, learning how to fine-tune the contours directly from the data is chosen, for N vertices {Xi i = 1,...., N}, the feature vector for each vertex is first constructed, and the feature F of vertex Xi is the function [F(Xi); Xi'] corresponding to the network features and vertex coordinates, where F is the feature map of the backbone output, F(xi) is the bilinear difference output at vertex Xi, the additional Xi is used to describe the location relationship between the vertices, Xi is translation invariant, and the relative coordinates are obtained by the coordinates of each vertex subtracting the minimum x and y of all vertices in the contour.

After the feature vectors of the vertices are obtained, further learning of the features of the contour is required. The features of the vertices can be treated as a one-dimensional discrete signal f: Z → R^D, and then, the vertices are processed one by one. In order not to destroy the topology of the contour, a periodic signal is extended and is defined as follows:

$$\left(f_N\right)_i \triangleq \sum_{j=-\infty}^{\infty} f_{i-jN} = f_{i(\mathrm{mod}N)} \tag{1}$$

The circular convolution of Eq. 2 is applied for feature learning.

$$\left(\mathrm{f_N} * k\right)_\mathrm{i} = \sum\nolimits_{\mathrm{j}=-\mathrm{r}}^{\mathrm{r}} \left(\mathrm{f_N}\right)_{\mathrm{i+j}} \mathrm{k_j} \tag{2}$$

The periodic signal of Eq. 1 is the definition of the vertex feature, k: [−r, r] → R^D is the learnable convolution kernel, and ∗ is the standard convolution operation in Eq. 2. The graph representation is similar to the standard convolution operation and can be integrated into the current network quite simply. After feature learning, three 1 × 1 convolution layers are used by each vertex to offset the output, and the size of the cyclic convolutional kernel is 9 in the experiment.

Details of the Structure of the Example Segmentation Model YOLOF-Snake Network

The Deep-snake is added to the YOLOF object detection model for instance segmentation. First, the target box is generated by YOLOF, the midpoints of the target regression box are connected to form a rhombus box, and then, the Deep-snake algorithm is used to adjust the vertex target poles of the rhombus to form an octagonal contour. Finally, the Deep-snake algorithm is used to iteratively adjust to obtain the target shape. For the extraction of the initial contour, the idea of poles is adopted, after the rectangular regression box is obtained, and the centroids of the four edges are obtained $\left\{x_i^{ex} i = 1, 2, 3, 4\right\}$. For each polar point, the 1/4 of the length of the regression box in two directions along the box is expanded, and if it exceeds the range of the original box, it is stopped. Finally, several extended edges are added, and the octagon is formed. After that the octagon and the object edges of N points are sampled equally, where the sampling of the object edges is the ground-truth contour sampling the object edges equally for N points, N is 128, and the point of the upper pole x_1^{ex} will be the starting point. The 40 points are sampled equally before the diamond contour is input into the Deep-snake. If the vertices are far away from the ground-truth bounding box, the adjustment becomes more difficult. Therefore, an iterative method is used to perform Deep-snake adjustment with three iterations. The backbone contains eight circular convolutional layers, each using residual connections. Fusion block is used to fuse the multi-scale contour features in the backbone network, containing one 1 × 1 convolutional layer and one maximum pooling layer. The prediction branch uses three 1 × 1 convolutions to output each vertex's offset. The specific network structure is shown in **Figure 6**.

Selection of Positive and Negative Samples

The YOLOF method for green fruit recognition is an anchor-based method. For the selection of anchor box, each pixel point has three anchor boxes with different aspect ratios {1:2, 1:1, and 2:1}. To cover the fruit object more effectively, three sizes of anchor boxes are set for each aspect ratio as follows {2^0, $2^{1/3}$, and $2^{2/3}$}; thus for each pixel point, there will be nine anchor boxes. Each anchor box is associated with a K-dimensional vector, where K is the number of categories of the target fruit and a four-dimensional vector for the border regression. In general, the definition of the positive sample is based on the Intersection over

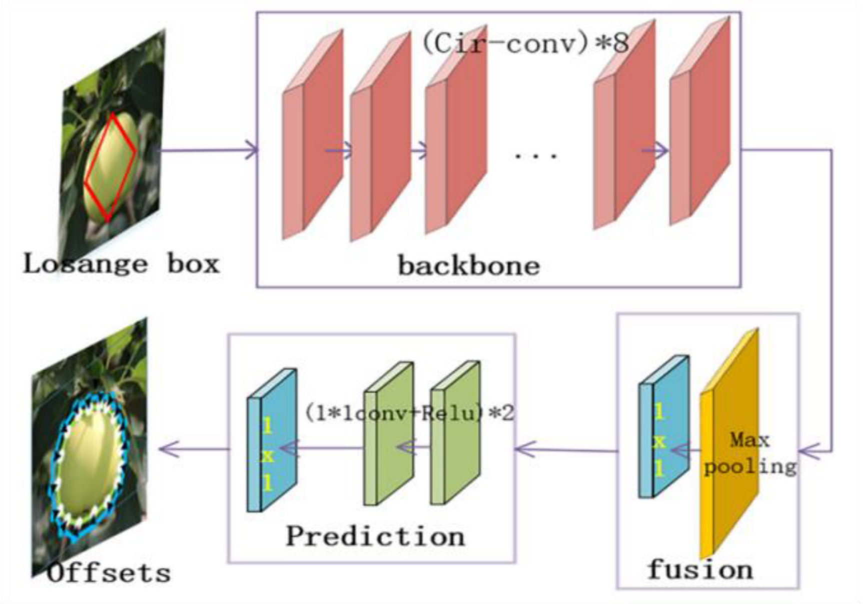

FIGURE 6 | Network structure of the add-on module.

Union (IoU) between the anchor box and the ground truth. In RetinaNet, anchor boxes are associated with ground truth when the IoU is greater than 0.5; IoU in (0, 0.4) is used as background; each anchor box is associated with a maximum of one ground truth. In the K-dimensional vector, the associated category value is 1, while others are 0. Anchor boxes with IoU between (0.4 and 0.5) are discarded. The border regression is to calculate the offset between the anchor box to the associated ground-truth bounding boxes. However, for our proposed structure, the number of anchor boxes is much reduced and very sparse because only one layer of features is used. Using the above matching strategy for sparse anchor boxes causes the problem that large ground truth will generate more positive anchor boxes than small ground truth, thus causing the imbalance problem of positive anchor boxes, which will cause the YOLOF target detector to focus only on large targets and ignore small targets.

A balanced matching strategy is used to solve the above problem. For each ground-truth bounding box, only the closest k anchor boxes are used as positive anchor boxes, so that the ground-truth bounding boxes of different sizes will have the same number of positive anchor boxes, ensuring that all ground truth can participate in training with the same probability. Meanwhile, the negative samples with IoU greater than 0.7 and the positive samples with IoU less than 0.15 are ignored, so that the negative samples with large IoU and the positive samples with small IoU are filtered out.

Loss Function

The loss function is used to estimate the degree of inconsistency between the predicted and true values of our model, which determines the effectiveness of our model for the fruit detection and segmentation. The loss function contains four components, the loss function is L_{YOLOF} in the detection module, and the loss function L_{snake} in the segmentation module consists of the loss function L_{ex} for the poles and the iterative contour adjustment loss function L_{iter} . In addition, the loss function of the detection module includes the loss function of the classification and the loss function of the regression boxes. The focal loss function is applied for the loss function of the classification L_{cls}. The experiments show that $\gamma\ = 2$ works best and the robust interval is $\gamma \in [0.5, 5]$. Besides, the Smooth-L1 loss function is employed for the loss function of the border regression $Lres$. So the overall loss function is as follows:

$$L_{y,s} = L_{YOLOF} + L_{snake}\ = L_{cls} + L_{res}\ + L_{ex} + L_{iter} \tag{3}$$

Where,

$$L_{cls} = f(x) = \begin{cases} -\alpha(1-y)^{\gamma} logy' \\ -(1-\alpha)y' logy \end{cases} \tag{4}$$

$$L_{re} = f(x) = \begin{cases} 0.5(\sigma x)^2, & if\ |x| < \frac{1}{\sigma^2} \\ |x| - \frac{0.5}{\sigma^2}, & otherwise \end{cases} \tag{5}$$

$$L_{ex} = \frac{1}{4}\sum_{i=1}^{4} \ell_1 \left(x^{\sim ex}_{i} - x_i^{ex}\right) \tag{6}$$

$$L_{iter} = \frac{1}{N}\sum_{i=1}^{N} \ell_1 \left(x^{\sim}{}_{i} - x_i^{gt}\right) \tag{7}$$

where $x^{\sim ex}_{i}$ is the predicted extreme value points, and $x^{\sim}{}_{i}$ is the deformed contour point, which is the boundary point of the true contour of the fruits. $\alpha = 0.25$, $\gamma = 2$, and $\sigma = 3$ are default values.

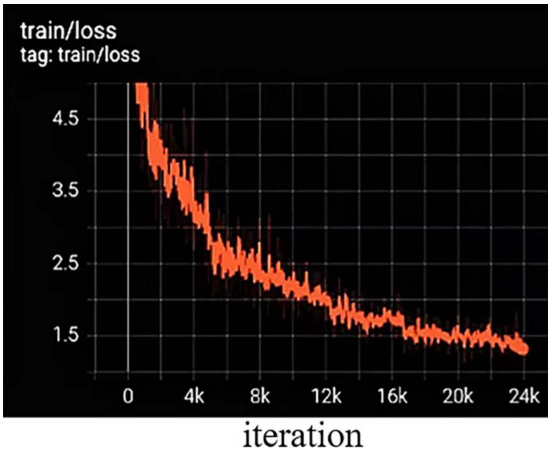

FIGURE 7 | Training error variation curve.

EXPERIMENTAL RESULTS

To obtain fast and accurate segmentation results for green fruit in complex environments, the loss function is optimized to balance the ground truth, confidence, and error of segmentation results. The trend of the loss function can be represented by the training error curve in **Figure 7**, and the training and validation sets are trained for about 24 iterations. The network is fitted quickly in the first six iterations, and the loss function stabilized after 16 iterations.

In addition, the model is pre-trained and the optimal training model is selected to test the segmentation of green fruits on the validation set to ensure the real-time performance and accuracy of the model. Finally, the detection and segmentation effects of the method and more advanced instance segmentation methods for green fruits are compared. The average precision (AP) and detection time (T) are used as the main performance evaluation metrics.

Experimental Platform and Details

The experiments are run on a server with Ubuntu 18.04 operating system, 64 GB of running memory, an NVIDIA 2060Ti graphics card with 32 GB of video memory, and a CUDA 10.0 environment. The experiments are based on the PyTorch deep learning framework, a virtual environment of python 3.7 on Anaconda, with an input image of 640 × 480 pixels and an initial learning rate of 0.0025.

The following section describes the details of the implementation.

The detection part: First, ResNet101 is employed as the backbone network to extract the features of the image, and after the extraction of the green fruit features, a fifth layer feature map with 32 times downsampling and 2048 number of channels is selected. After that, four identical residual blocks are employed to process the feature maps, and there are two branches to perform classification and boundary regression for output. Then, the negative samples with IoU less than 0.7 and positive samples with IoU greater than 0.15 are selected and a loss function based on stochastic gradient descent is used for training.

The segmentation part: First, the regression box of the target fruit obtained from the YOLOF model is optimized to the contour of the target fruit. The segmentation module used a recurrent convolutional network to iteratively adjust the contour until it is coincident with the boundary of the target fruit. The network of the segmentation module is iterated three times, and the results of each iteration are saved. Finally, the results of the three iterations are used to evaluate the YOLOF-snake segmentation model.

Evaluation Metrics

To evaluate the effectiveness of our model for green fruit segmentation, the average precision AP and time T are applied, where the calculation of AP entailed first calculating true cases (TP), false-positive cases (FP) based on the IoU, followed by ranking the confidence of each ground truth from high to low, obtaining the precision P and recall R. The value of AP is calculated after plotting the PR curve. The average of the ten is taken as mean average precision (mAP) (mAP $= 1/10 \sum_{i \in I} \mathrm{AP}_{IOU=i}$). mAP can comprehensively measure P, R, and threshold; thus, it can be strongly persuasive when evaluating the model. The equations for P, R, and AP are as follows:

$$P = \frac{TP}{TP + FP} \tag{8}$$

$$R = \frac{TP}{TP + FN} \tag{9}$$

$$\mathrm{AP}_{IoU=i} = 1/101 \sum\nolimits_{r \in R} p\ (r) = 1/101 \sum\nolimits_{r \in R} \max\nolimits_{\tilde{r}:\tilde{r} \geq r} p(\tilde{r}) \tag{10}$$

Where, TP—the number of samples that are actually positive and are detected as such.

FP—the number of samples that are actually negative and detected as positive.

FN—the number of samples that are actually negative and are detected as such.

i—threshold in IoU: [0.5, 0.55, 0.6,..., 0.95].

r—recall, the precision associated with recall.

R—[0, 0.01, 0.02,... 1.0] with an interval of 0.01 and a total of 101 values.

The ground truth obtained after the non-maximum suppression (NMS) method may not be the correct category, so a ground truth with a confidence level greater than a threshold of 0.5 is defined as a positive sample and vice versa; a positive sample with an IoU greater than a threshold of 0.6 with the ground-truth bounding box is considered a TP and vice versa as an FP. FN is the presence of an actual positive sample in a negative sample.

Results and Analysis

Model Training

To prevent experimental overfitting from occurring, pre-training is used to provide a larger number of training samples when

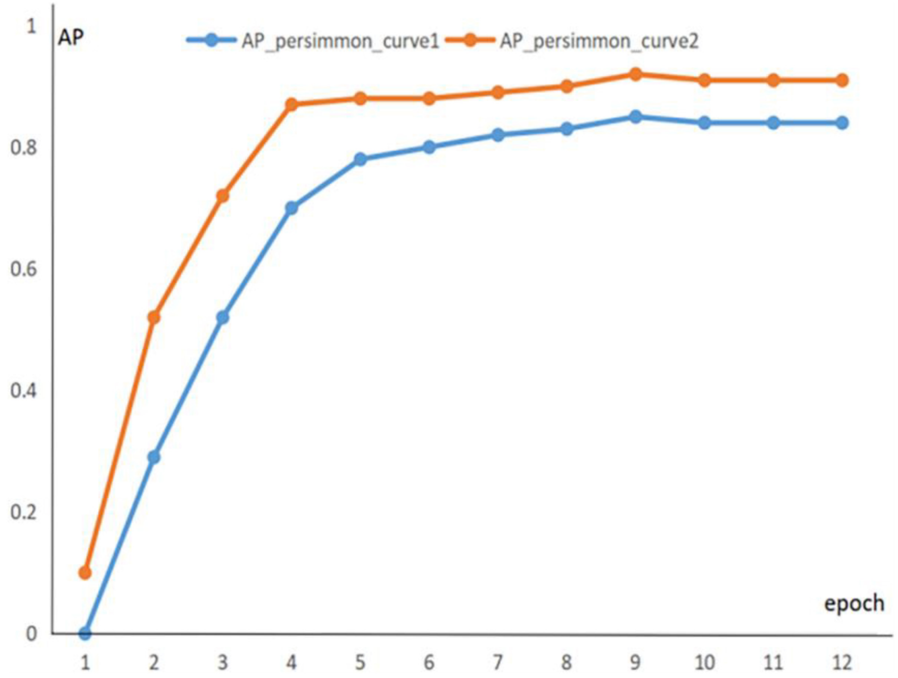

FIGURE 8 | AP evolution curve of persimmon dataset.

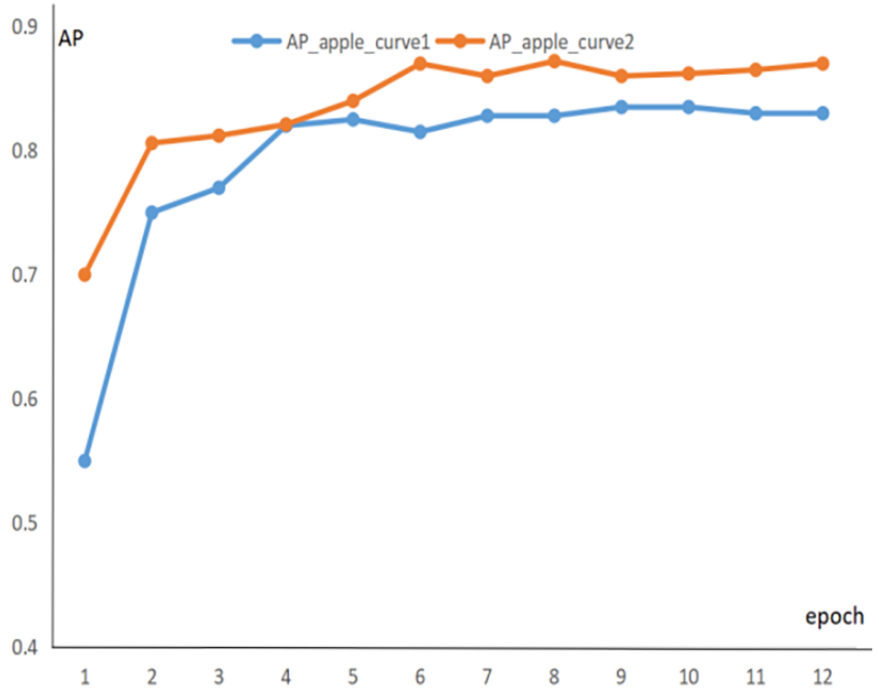

FIGURE 9 | AP evolution curve of apple dataset.

training the green fruit dataset, allowing for a faster fit. The 1600 images containing apples are extracted from the common COCO dataset, these images are applied for pre-training experiments, and the information of the pre-trained parameters is recorded. Then, the dataset is trained and the pre-trained parameters are transferred to the formal training as initial weights. It is found that pre-training significantly improved the segmentation of the model. **Figures 8**, **9** show the segmentation AP for the

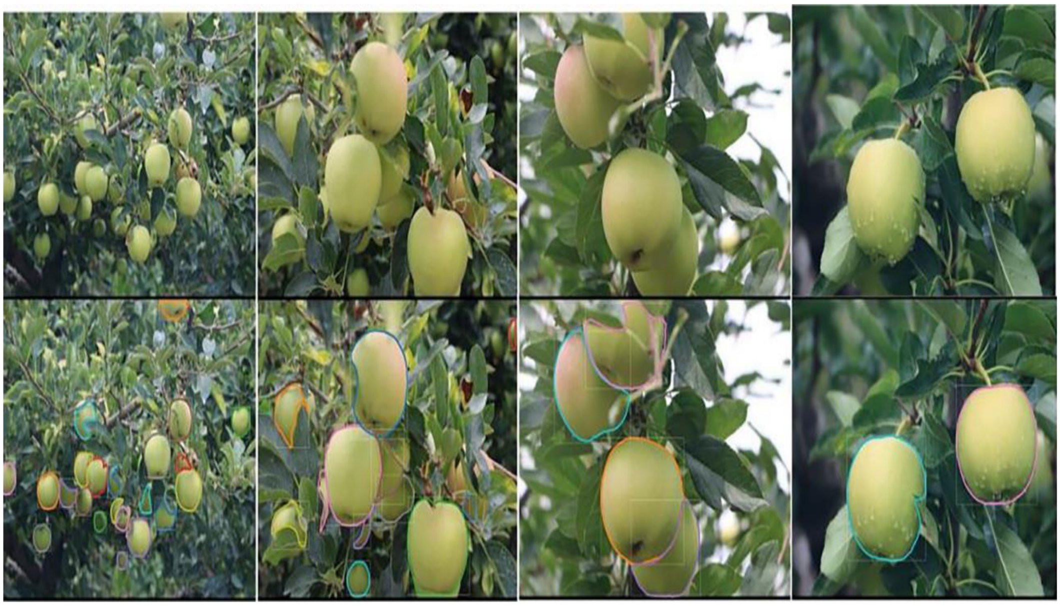

FIGURE 10 | Green apple result images.

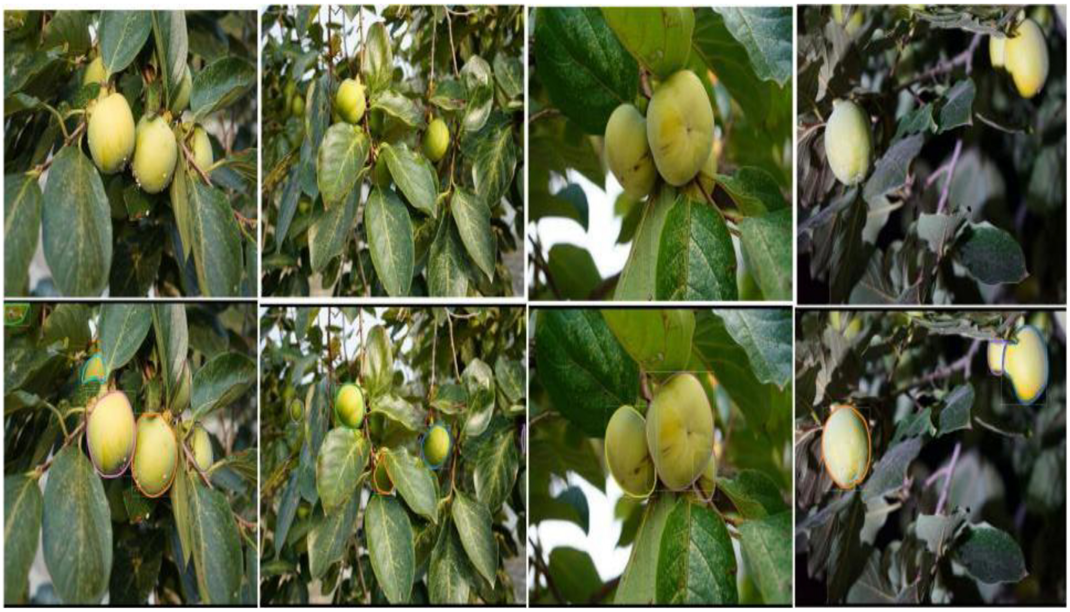

FIGURE 11 | Green persimmon result images.

apple and persimmon datasets with and without the pre-training method, respectively.

The blue curve in the figure indicates the AP curve for segmentation of green fruit with random initial weights, and the orange curve indicates the AP curve for segmentation of green fruit with pre-training parameters as initial weights. The above line graph shows that the orange curve is significantly higher than the blue curve for the segmentation AP of green fruit, which indicates that the segmentation accuracy and generalization ability of the model is improved by pre-training experiments, benefiting the model to be better applied to the segmentation of other green fruits.

Segmentation Effect

In the actual complex orchard background, the segmentation results of the fruits are affected by overlapping, blocking, and lighting factors, so the segmentation difficulty for each case is different. Overlapping obscured green fruits are more difficult

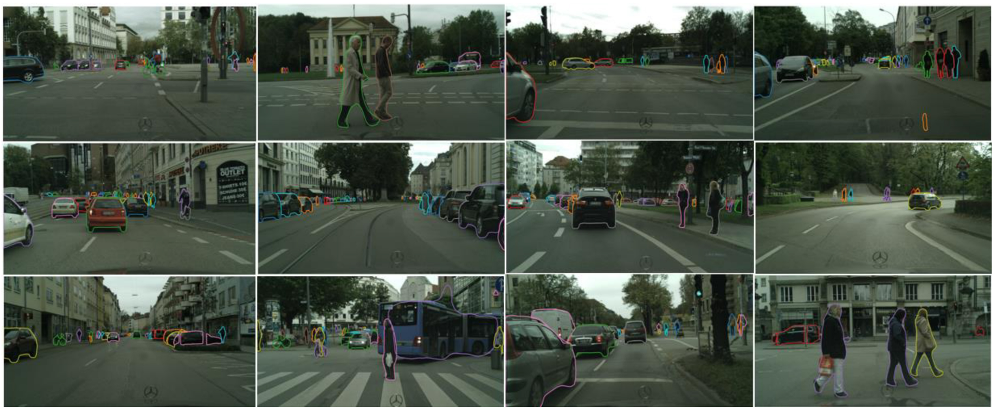

FIGURE 12 | Result images of the Cityscapes data set.

to segment, while images with a smaller number of fruits and a simple background are easier to segment and produce better results. Segmentation of green fruit is also more difficult in backlight and at night. The results of YOLOF-snake model for green fruit segmentation are shown in **Figures 10**, **11**.

In each pair of images of the segmentation effects, the top and bottom are the original images input to the model and the segmented result images, respectively. As can be seen, the green fruit against the same colored background can still be segmented without segmentation errors or missed detection errors. The target fruits whose edges are obscured by branches or overlapping fruits can also be accurately segmented. This indicates that the model has a good generalization capability and good resistance to interference.

At the same time, to verify the validity of the experimental results, the public dataset Cityscapes is chosen to conduct experiments. Experiments show that YOLOF-Snake can achieve high accuracy and good results on the public dataset. The following **Figure 12** shows some result images.

Ablation Experiments

Ablation experiments are carried out to analyze the residual block count problem in the YOLOF structure, and comparisons of the experiments are shown in **Table 2**.

As described in **Table 2**, the segmentation efficiency for green fruit is improved by changing the number of residual blocks, with the accuracy increasing with the number of residual blocks. However, the four residuals in the encoder are added to keep the model simple. Through ablation experiments, the residual blocks are applied to the model to obtain better segmentation results for green fruit and to deal with problems, such as fruit overlap and branch occlusion.

TABLE 2 | Number of residual blocks.

Number	mAP/%	mAP$_s$/%	mAP$_m$/%	mAP$_l$/%
0	62.3	47.6	79.5	72.5
2	63.5	47.6	70.1	75.2
√4	64.9	47.8	70.2	77.3
6	65.0	47.6	70.2	78.2
8	66.1	48.5	71.0	79.1

Comparisons

As the model is a two-stage approach to green fruit recognition and segmentation, the accuracy of recognition and segmentation is compared with other better-performing object detection and segmentation methods, respectively. Since the main feature of the model is simplicity and efficiency, it is also compared with other methods in terms of time. The current advanced and effective object detection and segmentation methods are collected. The same experimental platform configuration is utilized, and the evaluation results are shown in **Table 3**.

Table 3 shows comparisons of the recognition and segmentation accuracy of several advanced segmentation methods for apples and persimmons, respectively. The table shows the records of recognition and segmentation accuracy for green apples and green persimmons, respectively. "——" indicates that the method has no detection or segmentation ability.

TABLE 3 | Performance comparison of detection and segmentation methods.

Methods	Apple dataset		Persimmon dataset		Average time/s
	mAP	mAP^S	mAP	mAP^S	
SOLO	—	57.4	—	76.5	0.49
PolarMask	56.3	53.8	68.3	66.1	0.52
YOLACT	57.6	60.5	66.9	71.2	0.45
TensorMask	—	65.3	—	72.4	0.71
FCOS	61.2	—	78.6	—	0.42
RetinaMask	69.2	69.4	77.5	73.1	0.92
MaskR-CNN	70.3	70.9	79.9	79.3	0.48
MSR-CNN	71.2	71.6	80.3	80.9	0.56
OURS	71.0	71.4	81.8	82.6	0.23

As can be seen from **Table 3**, the method has the highest accuracy with better detection of apples and persimmons. Although the segmentation accuracy for apples is 0.2% lower than that of MS R-CNN (Huang et al., 2019), YOLOF-snake has a cleaner structure, faster detection speed, and smaller computation than MS R-CNN, enabling real-time and accurate detection of green fruits in a complex orchard background. A comparison of the detection times of the various models and the method is shown in **Table 3**. Through the above comparison and analysis, the method performs more outstandingly in terms of accuracy and real-time, which improves the efficiency of the work of the orchard picking robot and gives some practical significance to the solution of the problem of branch and leaf shading and fruits overlapping.

CONCLUSION

The YOLOF algorithm is applied by YOLOF-snake for the segmentation of green fruit in complex orchard backgrounds. The Deep-snake method is added to the YOLOF object detection model for the segmentation of green object fruit for branch shading and overlapping fruit, and poor light intensity at night and back-lighting in the complex orchards. The four residual blocks of YOLOF are used to solve the problem of restricted perceptual field. After the regression box of the object fruit is obtained, the Deep-snake network performs offset prediction on the four midpoints of the regression box, taking the extreme points around the object as the object and the center, extending uniformly in both directions to the boundary where the extreme points are located. The segmentation module then iteratively adjusts the contour lines until they coincide with the boundaries of the object fruit. Simultaneous circular convolution is used to implement vertex feature learning for the green object fruit to achieve a more desirable segmentation result for the green object fruit. The segmentation accuracy for apple fruits in the complex orchard backgrounds is 71.4%, and the segmentation accuracy for persimmon is 82.6%. Compared with MS R-CNN, although YOLOF-snake achieves similar results to MS R-CNN in terms of segmentation accuracy, the computational volume and complexity of the model are smaller and the network structure is simpler. While comparing with other network models, the model improves the recognition accuracy and speed. Overall, the model has the best detection performance.

The efficiency of orchard harvesting robots has been further improved by targeting the occlusion and overlap of green fruits in complex orchards. In pursuit of better fruit picking and fruit and vegetable production, other factors affecting fruit detection and segmentation should be considered in future research. Moreover, the network structure should be optimized by combining the ideas of lightweight networks and feature fusion to obtain better recognition and segmentation of green orchards.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding authors.

AUTHOR CONTRIBUTIONS

WJ: conceptualization, data curation, and writing – original draft preparation. ML: methodology, visualization, and writing – original draft preparation. RL: investigation and writing – reviewing and editing. CW: software and validation. NP: visualization and validation. XY: investigation, software, and validation. XG: conceptualization and writing – reviewing and editing. All authors contributed to the article and approved the submitted version.

FUNDING

This work was supported by the National Natural Science Foundation of China (Nos. 81801776, 21978139 and 81871508), Natural Science Foundation of Jiangsu Province (No. BK20170256), Natural Science Foundation of Shandong Province in China (ZR2020MF076 and ZR2019MB030), Focus on Research and Development Plan in Shandong Province (No. 2019GNC106115), and Taishan Scholar Program of Shandong Province of China (No. TSHW201502038).

REFERENCES

Bac, C. W., Van Henten, E. J., Hemming, J., and Edan, Y. (2014). Harvesting robots for high-value crops: state-of-the-art review and challenges ahead. *J. Field Robot.* 31, 888–911.

Benos, L., Tagarakis, A. C., Dolias, G., Berruto, R., Kateris, D., and Bochtis, D. (2021). Machine learning in agriculture: a comprehensive updated review. *Sensors* 21:3758. doi: 10.3390/s21113758

Bochtis, D. D., Sørensen, C. G., and Busato, P. (2014). Advances in agricultural machinery management: a review. *Biosyst. Eng.* 126, 69–81.

Caicedo Solano, N. E., GarcíaLlinás, G. A., and Montoya-Torres, J. R. (2020). Towards the integration of lean principles and optimization for agricultural production systems: a conceptual review proposition. *J. Sci. Food Agric.* 100, 453–464. doi: 10.1002/jsfa.10018

Dhaka, V. S., Meena, S. V., Rani, G., Sinwar, D., Ijaz, M. F., and Woźniak, M. (2021). A survey of deep convolutional neural networks applied for prediction of plant leaf diseases. *Sensors* 21:4749. doi: 10.3390/s21144749

Farkhani, S., Skovsen, S. K., Dyrmann, M., Jørgensen, R. N., and Karstoft, H. (2021). Weed Classification Using Explainable Multi-Resolution Slot Attention. *Sensors* 21:6705. doi: 10.3390/s21206705

Gan, H., Lee, W. S., Alchanatis, V., and Abd-Elrahman, A. (2020). Active thermal imaging for immature citrus fruit detection. *Biosyst. Eng.* 198, 291–303.

Henila, M., Chithra, P., Henila, M., and Chithra, P. (2020). Segmentation using fuzzy cluster-based thresholding method for apple fruit sorting. *IET Image Proc.* 14, 4178–4187.

Huang, Z., Huang, L., Gong, Y., Huang, C., and Wang, X. (2019). "Mask scoring r-cnn," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (Long Beach, CA: IEEE), 6409–6418.

Jha, K., Doshi, A., Patel, P., and Shah, M. (2019). A comprehensive review on automation in agriculture using artificial intelligence. *Artif. Intell. Agric.* 2, 1–12.

Jia, W., Zhang, Y., Lian, J., Zheng, Y., Zhao, D., and Li, C. (2020c). Apple harvesting robot under information technology: a review. *Int. J. Adv. Robot. Syst.* 17:1729881420925310.

Jia, W., Mou, S., Wang, J., Liu, X., Zheng, Y., Lian, J., et al. (2020a). Fruit recognition based on pulse coupled neural network and genetic Elman algorithm application in apple harvesting robot. *Int. J. Adv. Robot. Syst.* 17:1729881419897473.

Jia, W., Tian, Y., Luo, R., Zhang, Z., Lian, J., and Zheng, Y. (2020b). Detection and segmentation of overlapped fruits based on optimized mask R-CNN application in apple harvesting robot. *Comput. Electron. Agric.* 172:105380.

Koirala, A., Walsh, K. B., Wang, Z., and McCarthy, C. (2019). Deep learning–Method overview and review of use for fruit detection and yield estimation. *Comput. Electron. Agric.* 162, 219–234.

Kundu, N., Rani, G., Dhaka, V. S., Gupta, K., Nayak, S. C., Verma, S., et al. (2021). IoT and interpretable machine learning based framework for disease prediction in pearl millet. *Sensors* 21:5386. doi: 10.3390/s21165386

Li, Q., Jia, W., Sun, M., Hou, S., and Zheng, Y. (2021). A novel green apple segmentation algorithm based on ensemble U-Net under complex orchard environment. *Comput. Electron. Agric.* 180:105900.

Liu, J., and Wang, X. (2021). Plant diseases and pests detection based on deep learning: a review. *Plant Met.* 17, 1–18. doi: 10.1186/s13007-021-00722-9

Liu, X., Zhao, D., Jia W., Ruan, C., and Ji, W. (2019). Fruits segmentation method based on superpixel features for apple harvesting robot. *Trans. Chin. Soc. Agric. Mach.* 50, 22–30.

Maheswari, P., Raja, P., Apolo-Apolo, O. E., and Pérez-Ruiz, M. (2021). Intelligent fruit yield estimation for orchards using deep learning based semantic segmentation techniques—a review. *Front. Plant Sci.* 12:684328. doi: 10.3389/fpls.2021.684328

Minaee, S., Boykov, Y. Y., Porikli, F., Plaza, A. J., Kehtarnavaz, N., and Terzopoulos, D. (2021). "Image segmentation using deep learning: A survey," in *IEEE Transactions on Pattern Analysis and Machine Intelligence* (Piscataway: IEEE). doi: 10.48550/arXiv.2001.05566

Mu, Y., Chen, T. S., Ninomiya, S., and Guo, W. (2020). Intact detection of highly occluded immature tomatoes on plants using deep learning techniques. *Sensors* 20:2984. doi: 10.3390/s20102984

Oksuz, K., Cam, B. C., Kalkan, S., and Akbas, E. (2020). Imbalance problems in object detection: a review. *IEEE Trans. Pattern Anal. Mach. Intell.* 43, 3388–3415. doi: 10.1109/TPAMI.2020.2981890

Pallathadka, H., Mustafa, M., Sanchez, D. T., Sajja, G. S., Gour, S., and Naved, M. (2021). Impact of machine learning on management, healthcare and agriculture. *Mater. Today Proc.* doi: 10.1155/2021/8106467

Patrício, D. I., and Rieder, R. (2018). Computer vision and artificial intelligence in precision agriculture for grain crops: a systematic review. *Comput. Electron. Agric.* 153, 69–81.

Saleem, M. H., Potgieter, J., and Arif, K. M. (2021). Automation in agriculture by machine and deep learning techniques: a review of recent developments. *Precis. Agric.* 22, 2053–2091. doi: 10.3390/mi12060665

Sharma, A., Jain, A., Gupta, P., and Chowdary, V. (2020). Machine learning applications for precision agriculture: a comprehensive review. *IEEE Access* 9, 4843–4873.

Tahaseen, M., and Moparthi, N. R. (2021). "An Assessment of the Machine Learning Algorithms Used in Agriculture," in *2021 5th International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, (Piscataway: IEEE), 1579–1584.

Tang, Y., Chen, M., Wang, C., Luo, L., Li, J., Lian, G., et al. (2020). Recognition and localization methods for vision-based fruit picking robots: a review. *Front. Plant Sci.* 11:510. doi: 10.3389/fpls.2020.00510

Tian, H., Wang, T., Liu, Y., Qiao, X., and Li, Y. (2020). Computer vision technology in agricultural automation—A review. *Inform. Proc. Agric.* 7, 1–19.

Ushadevi, G. (2020). A survey on plant disease prediction using machine learning and deep learning techniques. *Intel. Artif.* 23, 136–154. doi: 10.1016/j.xinn.2021.100179

Van Klompenburg, T., Kassahun, A., and Catal, C. (2020). Crop yield prediction using machine learning: a systematic literature review. *Comput. Electron. Agric.* 177:105709.

Woźniak, M., and Połap, D. (2018). Adaptive neuro-heuristic hybrid model for fruit peel defects detection. *Neural Netw.* 98, 16–33. doi: 10.1016/j.neunet.2017.10.009

Xiong, J., Liu, Z., Chen, S., Liu, B., Zheng, Z., Zhong, Z., et al. (2020). Visual detection of green mangoes by an unmanned aerial vehicle in orchards based on a deep learning method. *Biosyst. Eng.* 194, 261–272.

Xu, W., Chen, H., Su, Q., Ji, C., Xu, W., Memon, M. S., et al. (2019). Shadow detection and removal in apple image segmentation under natural light conditions using an ultrametric contour map. *Biosyst. Eng.* 184, 142–154.

Yin, W., Wen Häni, N., Roy, P., and Isler, V. (2020). A comparative study of fruit detection and counting methods for yield mapping in apple orchards. *J. Field Robot.* 37, 263–282.

Yin, W., Wen, H., Ning, Z., Ye, J., Dong, Z., and Luo, L. (2021). Fruit Detection and Pose Estimation for Grape Cluster–Harvesting Robot Using Binocular Imagery Based on Deep Neural Networks. *Front. Robot. AI* 8:626989. doi: 10.3389/frobt.2021.626989

Zhang, C., Zou, K., and Pan, Y. (2020). A method of apple image segmentation based on color-texture fusion feature and machine learning. *Agronomy* 10:972.

Zhao, Z. Q., Zheng, P., Xu, S. T., and Wu, X. (2019). Object detection with deep learning: a review. *IEEE Trans. Neural Netw. Learn. Syst.* 30, 3212–3232. doi: 10.1109/TNNLS.2018.2876865

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.