Instance Segmentation and Number Counting of Grape Berry Images Based on Deep Learning

Yanmin Chen 1,2,3,4, Xiu Li 1, Mei Jia 1, Jiuliang Li 1, Tianyang Hu 1 and Jun Luo 1,2,3,4,*

1 College of Informatics, Huazhong Agricultural University, Wuhan 430070, China; chenyanmin@webmail.hzau.edu.cn (Y.C.); 0430@webmail.hzau.edu.cn (X.L.); jia.mei@webmail.hzau.edu.cn (M.J.); ljlui@webmail.hzau.edu.cn (J.L.); hutianyang2015@outlook.com (T.H.)
2 Shenzhen Institute of Nutrition and Health, Huazhong Agricultural University, Shenzhen 518120, China
3 Shenzhen Branch, Guangdong Laboratory of Lingnan Modern Agriculture, Genome Analysis Laboratory of the Ministry of Agriculture, Shenzhen 518120, China
4 Agricultural Genomics Institute at Shenzhen, Chinese Academy of Agricultural Sciences, Shenzhen 518000, China
* Correspondence: luojun@mail.hzau.edu.cn

Abstract: In order to achieve accurate segmentation of each grape image per berry, we construct a dataset composed of red globe grape samples and use a two-stage "localization–segmentation" framework-based mask region convolutional neural network (Mask R-CNN) and one-stage “pixel classification without localization” framework-based You Only Look At CoefficientTs (YOLACT) and segmenting objects by locations (SOLO) models in the grape segmentation experiments. The optimal performance of the model Mask R-CNN was applied for further study. To address the problem of overlapping and occlusion causing inaccurate fruit detection in this model, the postprocessing algorithm of the Mask R-CNN model was improved by using the linear weighting method, and the experimental results were significantly improved. The model average precision (AP), mean average precision (mAP), and mean intersection of union (mIoU) improved by 1.98%, 2.72%, and 4.30%, respectively. The correlation coefficient was improved from 93.59% to 96.13% by using the improved Mask R-CNN to count the number of red globe grape berries, which also further illustrates that the fruit detection problem was well solved. Using the generalized method on untrained images of different grape varieties in different scenes also achieved good segmentation results. In this study, we provide a method for segmenting and counting grape berries that is useful for automating the grape industry.

Keywords: grape berry; deep neural network; instance segmentation; Mask R-CNN; non-maximum suppression; number counting

1. Introduction

The grape planting area is relatively large across the globe, and traditional grape planting, management, picking, grading techniques, and other work require many humans, material, and other resources [1]. Grape is a kind of spike-shaped fruit with a complex shape. The size of grape berries is an important fruit quality of grapes, and the yield and quality are determined by the size of grape berries. Grape berry size is an important factor affecting grape grading, quality grading and price evaluation. Considering the entire bunch of grapes as the research object does not meet production demand; therefore, it is necessary to consider the grape berry as the research object to meet the actual production needs. However, due to the complexity of grape morphology and the mutual occlusion of the berries, the large difference in the size and color of the berries makes the segmentation of grape berries a challenging problem.
In this paper, we propose an improved Mask R-CNN-based grape berry segmentation method to address the problem of missed detection caused by berry occlusion in grape images, and propose to improve the Mask R-CNN model by using linear weighting to solve this problem, while achieving grape berry counting by filtering out instances with low segmentation accuracy and the number of instances that are far from other nearest berries.

1.1. Related Work

Generally, traditional image segmentation methods have used typical image processing and machine learning algorithms [2,3], which for grapes includes the identification of grape clusters based on color features [4] and grape berries based on geometric features [5], texture features [6], and integrated features [7]. Based on the color and texture features, combined with the texture and color characteristics of LBP for feature extraction, the K-nearest neighbor (KNN) algorithm was used to achieve the segmentation of red and white grape varieties [8]. Based on texture morphology, a high-resolution vineyard image was obtained by controlling artificial lighting at night. The number of grapes was calculated by segmenting the grape clusters from the image background by applying a threshold intensity level to the transformed image hue color layer [9]. The segmentation of red globe grapes and white grapes was achieved by detecting and locating grape clusters in color images using color mapping and morphological dilation techniques in natural environments [10]. The abovementioned literature has all achieved the segmentation of grapes; however, they all have certain limitations, and none of the individual grapes were segmented. The limitations were mainly that the color features were not suitable for distinguishing green grapes, the morphological features were difficult to identify when the fruit was occluded, and the texture features could be difficult to distinguish under certain lighting conditions.

With the rapid development of deep learning, image recognition and detection are widely used in fruit images such as such as grapes [11], blueberries [12], kiwis [13], strawberries [14], waxberry [15], and apples [16]. The image segmentation model Mask R-CNN is applied to segment the grape clusters, grape leaves [17], and grape cluster [18]. In [14], a strawberry fruit detection and segmentation method based on the Mask R-CNN algorithm was proposed, and by analyzing the shape and edge features of the mask image generated by Mask R-CNN, they obtained the location of strawberry-picking points. In [15], based on a Mask R-CNN network, the segmentation of overlapping bayberry fruits was realized via data augmentation processing, which improved the segmentation accuracy of occluded and overlapping fruits. In [16], an identification method of an apple-picking robot system based on a Mask R-CNN network for the identification and segmentation of overlapping apples in an orchard environment was proposed. In [19], a semantic segmentation-based implementation of grape berry segmentation and counting was proposed. We focus on each individual grape berry; therefore, this paper develops the study with instance segmentation. The above research proves the feasibility of using deep learning to achieve grape berry grade segmentation.

1.2. Contribution

The grape image data of the laboratory automation equipment scene and the natural orchard scene were collected, and the single-stage instance segmentation models YO-LACT and SOLO and the two-stage instance segmentation model Mask R-CNN model were initially established. We discovered that the location–segmentation framework-based Mask R-CNN model achieved the best results; hence, the Mask R-CNN model was selected and investigated in further. There was a problem of missed detection when many overlapping candidate boxes existed. Therefore, the Mask R-CNN model was improved by a linear weighting method to achieve a higher precision segmentation of individual grape berries. Good results were obtained using the Mask R-CNN with the improved model for counting individual grapes in the red globe grape images. The main contributions of our work are as follows:
The detection segmentation-based instance segmentation method can better solve the segmentation of grape images in industrial and natural scenes. An improved linear weighting post-processing method solves the berry missed detection problem in whole grape. The improved model can segment grape images with only a few annotations.

2. Materials and Methods

2.1. Sample Collection

To analyze and segment the grape berry practically, we collected grape images in automation equipment and natural scenes in the orchard. The original images of the grape image dataset contain 48 images of laboratory scenes of red grapes and 25 images of natural scenes of red grapes, for a total of 73 images.

All indexes of single bunch grape images and their number of grape berries in our experiments are shown in Figure 1.

![Figure 1. Number of grape berries in grape images.](image)

2.1.1. Grape Image Data in the Scene of Automated Equipment

The automated equipment scene grape image capture is shown in Figure 2. The device included the object to be detected, grapes, the electronically controlled track, an electronically controlled rotating hook, an industrial camera, LED light sources, a brightness adjustment knob, a computer, a case, and a control panel. The grapes were fixed on the hanging device, an electronically controlled rotating hook, and the background of the image acquisition was black or white cardboard. The image was captured using an industrial camera, model JAI AD-080, fixed in the front of the window and placed in the center. The distance between the camera and the hanging grapes was 50 cm, and the adjusted field of view was 30 cm (width) \times 40 cm (height). LED lamps were located on the upper, left, and right sides of the camera to provide multiple light sources, and the brightness was adjusted using the brightness adjustment knob at the bottom of the case. The control panel controlled the left-to-right movement of the electronically controlled track and the rotation of the electronically controlled rotating hook.
The images of red globe grapes captured in the natural scene of the orchard were taken in natural daylight, as shown in Figure 4.

Figure 4. Example images of red globe grapes in an orchard setting.

2.2. Image Preprocessing

The original image size was $1024 \times 768$ pixels for the automated equipment scene and $3024 \times 4032$ pixels for the natural orchard scene. To reduce the training time of the network, while keeping the original image from deforming in the preprocessing stage, the preprocessing operation first filled the image with black as a square according to the maximum side length, and Figure 5 shows the results of the filled images in the laboratory.
scene and the natural scene, with the black part as the filled area. The filled images were uniformly scaled to 512 × 512 pixels after filling.

![Figure 5. Schematic diagram of image padding.](image)

This dataset is divided into training, validation and testing subsets according to 8:1:1. To enrich the dataset and improve the model generalization ability, data augmentation is performed on the training and validation sets, respectively. The original dataset was augmented with horizontal flip, random rotation, and fuzzy processing methods due to the small amount of original data. Only horizontal flip was used because the distribution of grape clusters in the upper part of the grape bunches was high and the distribution of grape clusters in the lower part was low; therefore, only horizontal flip was used to maintain this feature. A rotating within 5° left and right rotation method were used to slightly adjust the view sight of grape images, and then 5 Gaussian kernels were applied to filter the images for further.

After the above two steps, the 43 images from the automated equipment scene were expanded to 438 images, and the 22 images from the natural scene of the orchard were expanded to 210 images; the total dataset was expanded from 65 images to 648 images. The training set includes 380 laboratory scene images and 190 natural scene images, the validation set contains 50 laboratory scene images and 28 natural scene images, and the testing set includes 5 laboratory scene images and 3 natural scene images.

2.3. Instance Segmentation Model

Instance segmentation [20] currently consists mainly of two phases or a single phase that focuses on simultaneously detecting and segmenting all object instances in an image.

In this experiment, the two-stage instance segmentation model Mask R-CNN model and the one-stage instance segmentation models YOLACT and SOLO models were selected for the grape berry segmentation problem. mAP, AP0.50, AP0.75 and mIoU values of the three models were compared, and the Mask R-CNN model with the best results was selected for the Mask R-CNN model segmentation. The problem of missed detection of occluded fruit berries that occurs in Mask R-CNN model is further proposed to replace the non-maximum suppression (NMS) in Mask R-CNN model with soft non-maximum suppression (Soft-NMS) to solve this problem. Finally, the improved model Soft-MRBS (grape berry segmentation based on mask R-CNN with Soft-NMS) is used for fruit segmentation on unlabeled untrained grape images of other varieties. The experimental flow chart is shown in Figure 6.

2.3.1. Mask R-CNN Model

Mask R-CNN [21] is a superior performance object detection and segmentation algorithm that integrates target detection (classification and localization) and semantic segmentation (localization region pixel segmentation) into a single framework, which can effectively detect objects in images while generating high-quality segmentation masks for each instance. Mask R-CNN extends the faster regions with CNN features (R-CNN) [22,23] target detection framework by combining faster R-CNN and fully convolutional networks [24] (FCN) to achieve higher accuracy compared to using only FCN. The core of the Mask
R-CNN is to add an extra branch at the end of the model, as shown in the red box in Figure 7, to perform semantic segmentation on the candidate region of target detection to retain the spatial location information. The Mask R-CNN consists of two stages; the first stage uses the region proposal network (RPN) to obtain the candidate bounding box poses, and the region of interest (ROI) combines the feature map with the RPN output to conduct the mapping in the feature map, extract the ROI, and adjust it to a uniform size. The second stage is the model output stage, in which the Mask R-CNN model contains three branches, which are used to predict the object categories, bounding boxes, and masks in the ROI of region of interest. This experiment is based on the original Mask R-CNN model code implementation.

![Figure 6. Experimental flow chart.](image)

![Figure 7. Mask R-CNN structure.](image)

### 2.3.2. YOLACT Model

YOLACT [25] is a typical single-stage instance segmentation model with the network structure shown in Figure 8. The model design is similar to that of Mask R-CNN; by adding another segmentation branch to the single-stage target detection model you only look once (YOLO) [26], it achieves real-time detection segmentation. The main idea of YOLACT is to localize the masks via two parallel branches, i.e., generating masks and predicting mask coefficients. One of the branch tasks uses the FCN network to generate prototype masks with the same image size as the original image, and the other branch is to add a head network to the target detection branch to predict the mask coefficients, which are selected using the NMS algorithm to obtain the final unique bounding box. Finally, the results of the two branches are linearly combined to obtain the final prediction results.
2.3.3. SOLO Model

A SOLO model has been proposed to solve the instance segmentation problem [27]. The core idea comes from the reconstruction of the instance segmentation problem: how to distinguish individual instances. The article points out that the location and shape of different instances in the image are different, and then, the concept of an instance category is proposed. The structure of the SOLO model is shown in Figure 9, which divides the image into $S \times S$ regions. (1) The semantic category branch predicts the category to which each region in $S \times S$ regions belongs. The output size of this branch is $S \times S \times C$ ($C$ indicates the class probability for each object instance). (2) The instance segmentation prediction branch predicts the object segmentation result in each region, and the output size of this branch is $H \times W \times S^2$. The SOLO model needs to predict the feature maps of each layer using the FPN network, and these feature maps are predicted by the semantic category branch and the instance segmentation prediction. The branch predicts the final segmentation result.

![Figure 8. YOLACT structure.](image)

![Figure 9. SOLO structure.](image)

2.4. Improved Algorithm
2.4.1. Non-Maximum Suppression Algorithm

The essence of the non-maximum suppression (NMS) algorithm is to search for local maxima and suppress non-maxima factors, widely used in the field of computer vision, such as image segmentation, target detection, face recognition, etc. NMS is mainly used to eliminate the redundant detection frames, which is used in the Mask R-CNN model. The
implementation process is to first arrange the detection frames in the order of confidence from largest to smallest and select the highest confidence as the target candidate frame; the second is to calculate the IoU between the remaining candidate frames and the target border. If the value of the IoU is greater than the threshold value, the candidate frame is discarded directly; otherwise, it is retained. The NMS is as in Equation (1).

\[
s_i = \begin{cases} 
  s_i, & \text{iou}(M, b_i) < N_t \\
  0, & \text{iou}(M, b_i) \geq N_t 
\end{cases}
\]

(1)

where:
- \(M\): the candidate box with the highest detection score;
- \(b_i\): the current candidate box;
- \(s_i\): the current candidate box detection score;
- \(N_t\): IoU threshold.

2.4.2. Soft-NMS

The main problem of the NMS algorithm is that the adjacent candidate boxes with high overlap are directly removed. Therefore, when there are real targets in the highly overlapping adjacent candidate frames, they are difficult to retain, resulting in a decrease in the model detection accuracy, which in turn affects the subsequent segmentation accuracy, resulting in the missing detection problem. To solve this problem, an improved algorithm of NMS, Soft-NMS [28], was adopted, and the pseudocode of the algorithm is shown in Algorithm 1, where \(f(\text{iou}(M, b_i))\) is a weighting function, including a linear weighting function and a Gaussian weighting function, which attenuates adjacent candidate frames with a high degree of overlap with the detection frame \(M\); the higher the degree of overlap, the greater the attenuation amplitude. In this paper, the idea of linear weighting is used to improve the problem of the missed detection of overlapping grapes. The formula of linear weighting is as shown in Formula (2). When the degree of overlap \(\text{iou}(M, b_i)\) is greater than the threshold \(N_t\), it is different from NMS, which directly removes adjacent candidate boxes. Soft-NMS retains the candidate frame, assigns a smaller score value \(s_i\), and sets the score threshold to remove the candidate frames whose score is lower than the threshold.

\[
s_i = \begin{cases} 
  s_i, & \text{iou}(M, b_i) < N_t \\
  s_i(1 - \text{iou}(M, b_i)), & \text{iou}(M, b_i) \geq N_t 
\end{cases}
\]

(2)

Algorithm 1: Soft-NMS

Input: candidate box set: \(B = \{b_1, \ldots, b_N\}\); detection score set \(S = \{s_1, \ldots, s_N\}\); IoU threshold \(N_t\)
Output: The candidate boxes set \(D\) and detection score set \(S\) processed by the algorithm
1: \(D = \{\}\)
2: while \(B \neq \text{empty}\) do
3: \(m = \text{argmax } S\)
4: \(M = b_m\)
5: \(D = D \cup M, B = B - M\)
6: for \(b_i\) in \(B\) do
7: \(s_i = s_i f(\text{iou}(M, b_i))\)
8: end
9: end
10: return \(D, S\)

2.4.3. Soft-MRBS Model

Soft-MRBS used the Soft-NMS method after the proposals parts. The model framework is shown in Figure 10. The model consisted of three parts: the input data, the improved Mask R-CNN network, and the prediction results.
The improved Mask R-CNN network was mainly divided into three parts: the backbone, the region proposal network (RPN), and the candidate region detection and segmentation network [29]. First, the red globe grapes image was sent to the backbone network (Backbone) for feature extraction, and the feature map (Feature Map) was obtained. The Soft-MRBS implemented in this paper used ResNet101 [30] as the backbone network to extract image features and introduced the feature pyramid network FPN (Feature Pyramid Networks) to extract the feature information of different scales in the grape image; secondly, the feature map output by the backbone network was used as the input of the region proposal network RPN, and the RPN scanned the image through a sliding window to generate candidate regions (Proposals). Soft-NMS replaced the original NMS in the Mask R-CNN with Soft-NMS) for grape berry instance segmentation.

3. Results and Discussion
3.1. Experimental Configuration

The experimental running platform was configured as NVIDIA GTX2080Ti*2, using the tensorflow_gpu-2.1.0, keras 2.1.6 deep learning framework for model code writing, the platform was Python 3.5.2 CUDA 10.1, and the image processing libraries used were OpenCV 3.4.2 and PIL 8.1.2.

3.2. Evaluation Indicators

1. Mean Intersection over Union (mIoU):

   \[ mIoU = \frac{|A \cap B|}{|A \cup B|} = \frac{1}{k} \sum_{i=0}^{k} \frac{p_u}{p_i + \sum_{j=0}^{k} p_j - p_u} \]

   where:
   
   - \( k \): the total number of categories output by the model;
   - \( p_i \): the area of the category \( i \) of the segmentation;
   - \( p_u \): the total area of the segmentation mask;
   - \( p_i \) and \( p_u \) are not superimposed.

Figure 10. Model of Soft-MRBS (grape berry segmentation based on Mask R-CNN with Soft-NMS) for grape berry instance segmentation.
where:
\( k \): the total number of categories output by the model;
\( k + 1 \): includes the background;
\( p_{ii} \): number of correctly classified pixels;
\( p_{ij} \): represents the number of pixels that belong to class \( i \) but are misclassified as class \( j \);
\( p_{ji} \): represents the number of pixels that belong to class \( j \) but are misclassified as class \( i \).

2. Average Precision (AP):
Taking the recall rate as the abscissa and the precision as the ordinate, we draw the PR curve, and the result of integrating the PR curve is the average precision \( AP \), which is defined as shown in Formula (4).

\[
AP = \int_0^1 PdR
\]  

(4)

To accurately analyze the model detection performance, \( AP \) with \( IOU \) thresholds between 0.50 and 0.95 were calculated with an interval of 0.05, and the \( APs \) under all thresholds were averaged to obtain the \( mAP \).

3. Coefficient of determination (\( R^2 \)):
\( R^2 \)-squared is a convenient and intuitive statistical measurement that indicates the closeness of predicted values and actual values, known as the coefficient of determination.

\[
R^2 = \frac{SSR}{SSt} = \frac{\sum (\hat{y}_i - \bar{y})^2}{\sum (y_i - \bar{y})^2}
\]  

(5)

where:
\( \hat{y}_i \): predictive value;
\( \bar{y} \): Average of actual values;
\( y_i \): actual value.

3.3. Model Training
The model training adopted the method of transfer learning [31]. Model weights pre-trained on the common objects in context (COCO) dataset [32] were introduced to initialize the model before model training. The image size was 512 \( \times \) 512 pixels, the batch size was 2, and the model momentum was set to 0.9, which remained unchanged during model training. The learning rate was adjusted to the optimum on each model, and the learning rate decay was set. The purpose was to appropriately reduce the learning rate during the training process and make the model converge to the optimal solution. The training parameters of the grape kernel example segmentation model were configured as shown in Table 1.

<table>
<thead>
<tr>
<th>Model</th>
<th>Optimizer</th>
<th>Rate</th>
<th>Decay</th>
<th>Momentum</th>
<th>Batch Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>YOLACT</td>
<td>SGD</td>
<td>( 1 \times 10^{-2} )</td>
<td>( 5 \times 10^{-4} )</td>
<td>0.9</td>
<td>2</td>
</tr>
<tr>
<td>SOLO</td>
<td>SGD</td>
<td>( 1 \times 10^{-2} )</td>
<td>( 1 \times 10^{-4} )</td>
<td>0.9</td>
<td>2</td>
</tr>
<tr>
<td>Mask R-CNN</td>
<td>SGD</td>
<td>( 1 \times 10^{-3} )</td>
<td>( 1 \times 10^{-4} )</td>
<td>0.9</td>
<td>2</td>
</tr>
</tbody>
</table>

3.4. Experimental Results
3.4.1. Preliminary Experimental Results
The evaluation results of the example segmentation models are summarized in Table 2. The experiments showed that for the three model structures YOLACT, SOLO, and Mask R-CNN, the Mask R-CNN model achieved the highest segmentation accuracy. The Mask R-CNN model reached 88.08% and 82.04% for the \( AP_{0.50} \) and the \( AP_{0.75} \), respectively, 69.93%
for the mAP and 85.98% for the mIoU on the test set, indicating that the test set of the segmentation mask had 85.98% of the regions matched with the real labels. In comparison, the Mask R-CNN model was 2.51% higher than the SOLO model in the mIoU, 1.39% higher in the AP0.50, 1.77% higher in the AP0.75, and 2.68% higher in the mAP. The Mask R-CNN model was 3.07% higher than the YOLACT model in the mIoU, 2.94% higher in the AP0.50, 2.96% higher in the AP0.75, and 3.34% higher in the mAP.

Table 2. Instance segmentation model performance evaluation.

<table>
<thead>
<tr>
<th>Model</th>
<th>mIoU (%)</th>
<th>AP0.50 (%)</th>
<th>AP0.75 (%)</th>
<th>mAP (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>YOLACT</td>
<td>82.91</td>
<td>85.14</td>
<td>79.08</td>
<td>66.59</td>
</tr>
<tr>
<td>SOLO</td>
<td>83.47</td>
<td>86.69</td>
<td>80.27</td>
<td>67.25</td>
</tr>
<tr>
<td>Mask R-CNN</td>
<td>85.98</td>
<td>88.08</td>
<td>82.04</td>
<td>69.93</td>
</tr>
</tbody>
</table>

Figures 11 and 12 visualize the results of the fruit segmentation on the test set of the YOLACT, SOLO, and Mask R-CNN models, respectively. To better observe the segmentation results, each fruit region was randomly filled with different colors. From the segmentation results, the three network models could detect most grape berries; however, there were differences in the segmentation details. YOLACT could essentially detect the fruit region; however, the edge of the fruit segmentation was not smooth enough, and there were many redundant boxes. The SOLO model better located each berry area, and achieved better segmentation for complete grape berries; however, the quality of the edge segmentation of the overlapping berries was not precise and complete, and there was a problem of the missed detection of individual berries. Mask R-CNN was better than the other two models in the overall edge details of fruit segmentation, but there was also the problem of the missed detection of individual fruit. Therefore, this research chose the Mask R-CNN model as the preliminary model for grape berry segmentation, and further improved the Mask R-CNN model through the linear weighting idea in Soft-NMS.

![Figures 11 and 12](image)

**Figure 11.** Automated device scenario test results. The second line is an enlarged view of area 1. The third line is an enlarged view of area 2. The fourth line is an enlarged view of area 3.
Segmentation performance comparison of red globe grape based on Mask R-CNN and Soft-MRBS models.

Table 3. Segmentation performance comparison of red globe grape based on Mask R-CNN and Soft-MRBS models.

<table>
<thead>
<tr>
<th>Model</th>
<th>mIoU (%)</th>
<th>AP0.50 (%)</th>
<th>AP0.75 (%)</th>
<th>mAP (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Automated device scenario</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mask R-CNN</td>
<td>88.12</td>
<td>88.85</td>
<td>83.09</td>
<td>71.10</td>
</tr>
<tr>
<td>Soft-MRBS</td>
<td>90.20</td>
<td>90.91</td>
<td>86.53</td>
<td>79.62</td>
</tr>
<tr>
<td>Orchard natural scene</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mask R-CNN</td>
<td>85.25</td>
<td>84.89</td>
<td>78.12</td>
<td>68.29</td>
</tr>
<tr>
<td>Soft-MRBS</td>
<td>86.24</td>
<td>84.95</td>
<td>78.98</td>
<td>72.35</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mask R-CNN</td>
<td>85.98</td>
<td>88.08</td>
<td>82.04</td>
<td>69.93</td>
</tr>
<tr>
<td>Soft-MRBS</td>
<td>89.53</td>
<td>90.06</td>
<td>84.76</td>
<td>74.23</td>
</tr>
</tbody>
</table>

The mixed scenarios show the performance comparison between the Mask R-CNN model and the Soft-MRBS model. The AP0.50, AP0.75, and mAP of the Mask R-CNN model were 88.08%, 82.04%, and 69.93%, individually, and the segmentation accuracy mIoU was 85.98%. In addition, the AP0.50, AP0.75, and mAP of the soft-NMS-based soft-MRBS model were 90.06%, 84.76%, and 74.23%, individually, and the mIoU was 89.53%.
The model \( AP_{0.50} \) was improved by 1.98\%, the \( AP_{0.75} \) was improved by 2.72\%, the \( mAP \) was improved by 4.30\%, and the \( mIoU \) was improved by 3.55\%, achieving a relatively high accuracy of grape berry segmentation. The \( AP_{0.75} \) is an important metric in instance segmentation because it reflects the ability to segment closely overlapping objects. The significant improvement in the \( AP_{0.75} \) showed that the Soft-MRBS was good at detecting objects that were highly overlapping with the ground truth, which proved the effectiveness of the Soft-MRBS model for solving the problem of fruit occlusion.

The experimental results showed that the Mask R-CNN model had high segmentation accuracy and high segmentation quality for the grape berry edges, especially for the masked grape berry edges, which reflected the possibility and potential of the model to solve the grape berry segmentation problem, but there were still some problems. Most of the missed detection problems existed in areas where the grape berries were obscured by a large area, and the obscured grape berries were prone to missed detection. Grapes, as a spiky fruit, have many overlapping berries, i.e., single berries are obscured by one or more berries. By observing Figures 13 and 14, the segmentation results of the Mask R-CNN and Soft-MRBS models on the same test images are shown, and it is clear from the comparison of the results that the missed berries were identified.

**Figure 13.** Comparison of the Mask R-CNN and Soft-MRBS results for automated device scenarios.

The second line is an enlarged view of area 1. The third line is an enlarged view of area 2.
3.4.3. Red Globe Grape Berries Number Count

When the Mask R-CNN model segments grape images, each berry is given a prediction frame and the accuracy of the segmented grape, and we count the number of grape berries by counting the number of prediction frames.

The statistics of grape kernel counting for the red globe grapes on the Mask R-CNN model and the improved model are shown in Figure 15. The correlation coefficients of the kernel counting on the Mask R-CNN model and the improved model soft-MRBS model were 93.59% and 96.13%, respectively, and their correlation coefficients were improved by 2.54%. This result indicates that the soft-MRBS model counted better than the Mask R-CNN, which further indicates that the improved model greatly improved the performance of grape berry segmentation and counting. This fruit counting result also showed the feasibility of the Mask R-CNN model for grape fruit segmentation and the success of the soft-MRBS model improvement.
3.5. Generalization Experiments

To further verify the adaptability of the Soft-MRBS model in practical applications, we selected three different types of grapes, Kyoho, Xiahei, and Red Rose, which were not trained in the different scenarios, for fruit segmentation tests. Table 4 shows the generalization test results of the new grape varieties, with an mIoU of 87.24% and mAP of 69.57%.

Table 4. Generalization test results of the Soft-MRBS on Kyoho, Xiahei and Red Rose.

<table>
<thead>
<tr>
<th>Model</th>
<th>mIoU (%)</th>
<th>mAP (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Soft-MRBS</td>
<td>87.24</td>
<td>69.57</td>
</tr>
</tbody>
</table>

Figures 16 and 17 show the test results of the not-trained new grape varieties. To summarize, the Soft-MRBS grape berry segmentation model proposed in this paper also had strong generalization performance for the new varieties of grapes and had better performance on the grape berry color, fruit size, ear compaction, branch, and leaf occlusion, etc. This shows its good adaptability and potential application in complex environments of automation equipment and orchards.
4. Conclusions

In this paper, we considered red globe grapes as the research object, relied only on RGB images and a small number of refined annotated samples, established a small sample red globe grapes dataset, and improved the postprocessing algorithm of the Mask R-CNN model, a two-stage instance segmentation model, using a linear weighting method. The improved model enhanced the segmentation performance, improved the segmentation accuracy, and further solved the overlapping problem. The problem of the omission of fruit detection in the Mask R-CNN model was further solved. Then, fruit counting was performed on the red globe grape images using the improved Mask R-CNN model, and good results were obtained. Through the generalization test, the applicability of the model was extended to different grape varieties, and the fruit-level segmentation of different scenarios and varieties of grapes was realized; thus, it has wide applicability and application value. The above research on a deep learning segmentation method for overlapping berries not only solves the segmentation problem of overlapping berries of grapes but also has guiding significance for key nodes of the grape production process, such as yield estimation, orchard picking, and post-production grading processes; it also plays a key driving role in the establishment of grape phenotype automation systems, such as agricultural monitoring, robotic picking, and post-production grading systems.

Author Contributions: Conceptualization, funding acquisition, J.L. (Jun Luo); methodology, T.H.; software, J.L. (Jiuliang Li); validation, formal analysis, M.J.; investigation, resources, data curation, X.L. writing—original draft preparation, writing—review and editing, visualization, supervision, project administration, Y.C. All authors have read and agreed to the published version of the manuscript.

Funding: The Fundamental Research Funds for the Central Universities (grant nos. 2662022XXYJ006, 2662017PY059 and 2662015PY066), the National Natural Science Foundation of China (grant nos. 61176052), the Cooperation funding of Huazhong Agricultural University-Shenzhen Institute of Agricultural Genomics, Chinese Academy of Agricultural Sciences, and Yingzi Tech & Huazhong Agricultural University Intelligent Research Institute of Food Health (No. IRIFH202212).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References


Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.