agronomy

Article

Method for Segmentation of Banana Crown Based on Improved DeepLabv3+

Junyu He [1,2], Jieli Duan [1,2,*], Zhou Yang [1,2,3], Junchen Ou [1,2], Xiangying Ou [1,2], Shiwei Yu [1,2], Mingkun Xie [1,2], Yukang Luo [1,2], Haojie Wang [1,2] and Qiming Jiang [1,2]

1 College of Engineering, South China Agricultural University, Guangzhou 510642, China
2 Guangdong Laboratory for Lingnan Modern Agriculture, Guangzhou 510642, China
3 School of Mechanical Engineering, Guangdong Ocean University, Zhanjiang 524088, China
* Correspondence: duanjieli@scau.edu.cn

Abstract: As the banana industry develops, the demand for intelligent banana crown cutting is increasing. To achieve efficient crown cutting of bananas, accurate segmentation of the banana crown is crucial for the operation of a banana crown cutting device. In order to address the existing challenges, this paper proposed a method for segmentation of banana crown based on improved DeepLabv3+. This method replaces the backbone network of the classical DeepLabv3+ model with MobilenetV2, reducing the number of parameters and training time, thereby achieving model lightweightness and enhancing model speed. Additionally, the Atrous Spatial Pyramid Pooling (ASPP) module is enhanced by incorporating the Shuffle Attention Mechanism and replacing the activation function with Meta-ACONC. This enhancement results in the creation of a new feature extraction module, called Banana-ASPP, which effectively handles high-level features. Furthermore, Multi-scale Channel Attention Module (MS-CAM) is introduced to the Decoder to improve the integration of features from multiple semantics and scales. According to experimental data, the proposed method has a Mean Intersection over Union (MIoU) of 85.75%, a Mean Pixel Accuracy (MPA) of 91.41%, parameters of 5.881 M and model speed of 61.05 f/s. Compared to the classical DeepLabv3+ network, the proposed model exhibits an improvement of 1.94% in MIoU and 1.21% in MPA, while reducing the number of parameters by 89.25% and increasing the model speed by 47.07 f/s. The proposed method enhanced banana crown segmentation accuracy while maintaining model lightweightness and speed. It also provided robust technical support for relevant parameters calculation of banana crown and control of banana crown cutting equipment.

Keywords: banana crown; banana; improved DeepLabv3+; semantic segmentation; attention mechanism; activation function; deep learning

Citation: He, J.; Duan, J.; Yang, Z.; Ou, J.; Ou, X.; Yu, S.; Xie, M.; Luo, Y.; Wang, H.; Jiang, Q. Method for Segmentation of Banana Crown Based on Improved DeepLabv3+. *Agronomy* **2023**, *13*, 1838. https://doi.org/10.3390/agronomy13071838

Academic Editor: Juncheng Ma

Received: 5 June 2023
Revised: 4 July 2023
Accepted: 7 July 2023
Published: 11 July 2023

1. Introduction

The Food and Agriculture Organization of the United Nations lists bananas as the fourth-largest food crop globally, ranking behind rice, wheat, and maize [1]. Bananas are grown in over 130 countries worldwide, and China stands as the third largest producer with a growing area of 350,000 to 400,000 hectares. In 2022, China produced 1.235 million tons of bananas [2]. While various aspects of banana production, such as harvesting, cleaning, packaging, and transportation, have undergone mechanization, banana crown cutting still heavily relies on manual labor [3]. However, with the continuous expansion of China's banana industry, it is crucial to explore intelligent automatic crown cutting technology to overcome the limitations associated with manual crown cutting. Manual crown cutting is labor-intensive, inefficient, and costly. Introducing advanced automatic crown cutting technology for bananas can enhance post-harvest handling, reduce labor expenses, and increase income for fruit growers.

In recent years, numerous academics have conducted extensive research on the identification and segmentation of fruits and vegetables by fruit and vegetable picking robots.

The primary objectives of these studies are to enhance the effectiveness of fruit and vegetable production while achieving greater intelligence and automation in the production process [4,5]. Among these research endeavors, the robot's ability to perform the necessary tasks relies heavily on precise segmentation of fruit and vegetable targets. When processing bananas after picking, preserving the integrity of the banana crown becomes crucial. Since both the banana finger and the banana crown exhibit a green color, achieving accurate segmentation poses additional challenges. Therefore, it is of utmost importance to achieve precise segmentation of the banana crown to enable automatic crown cutting of the banana.

There are not many studies on the banana crown segmentation method currently, but it can be explored by examining the green fruit and vegetable division method. The two main techniques used for segmenting fruits and vegetables nowadays are deep learning methods and conventional image segmentation methods. Among the conventional image segmentation methods, the most widely used ones are the Otsu algorithm, K-means clustering algorithm, and the fuzzy C-means (FCM) algorithm. Cui et al. [6] compared multiple color spaces and chose the R-G color components to segment kiwifruits using the Otsu algorithm, successfully separating the fruit from the background area. Wuzor et al. [7] utilized the K-means clustering algorithm to separate the guava region from the background, followed by watershed segmentation and morphological manipulation to accomplish single-guava segmentation. Marlinda et al. [8] used the fuzzy C-means (FCM) algorithm to separate mangoes from the background and measured their maturity.

Traditional image segmentation techniques are easily influenced by environmental elements in real-world application scenarios. Therefore, deep learning techniques with high accuracy and robustness are preferred to ensure the stability of selecting robot operations. Deep learning methods are trained on a large number of samples to extract deeper features, making them suitable for scenarios where both the target and background are green. Consequently, there is a lot of research on green fruit and vegetable segmentation using deep learning methods. For example, Li et al. [9] combined the edge features and advanced features of UNet with the Atrous Spatial Pyramid Pooling (ASPP) structure to segment green apples in intricate orchard landscapes. Hussain et al. [10] employed transfer learning on the Mask R-CNN technique to segregate samples of green fruits and stems. Wang et al. [11] proposed a unique deep learning-based fruit segmentation method SE-COTR, which achieved accurate real-time segmentation of green apples, with an average segmentation accuracy of 61.6%. Liu et al. [12] suggested a DLNet model with an average accuracy of 80.9% for accurately segmenting green fruits in a fuzzy environment. Ma et al. [13] proposed using a deep convolutional neural network to detect cucumber illness symptoms and separate them from leaves with an accuracy of 93.4%.

The following are some examples of how DeepLabv3+ has been used to segment green targets. Yan et al. [14] improved DeepLabv3+ and proposed a method for tea segmentation and picking point localization based on lightweight convolutional neural networks to address the issue of tea bud picking points in real environments, achieving a Mean Intersection over Union (MIoU) of 91.85%. Zhang et al. [15] enhanced DeepLabv3+ to perform high-precision and rapid lettuce segmentation in complex background and lighting conditions. Yu et al. [16] utilized the Swin transformer as a feature extraction network and incorporated a convolution block attention module into DeepLabv3+ to obtain the Swin-DeepLabv3+ model for weed segmentation in soybean fields, achieving an MIoU of 91.53%. Deng et al. [17] employed DeepLabv3+ to semantically segregate seedlings and weeds to get weed location information, with the DeepLabv3+ model achieving a pixel accuracy of up to 92.2%. Li et al. [18] utilized the mixed attention method in DeepLabv3+ to segment cucumber leaves and lesions, achieving an MIoU of 81.23%.

Currently, significant progress has been made in segmenting green targets, considering both the target and background are green. The DeepLabv3+ semantic segmentation algorithm has been widely applied and proven to deliver high-precision and swift segmentation of green targets even in complex backgrounds and challenging lighting conditions.

As a result, DeepLabv3+ was selected for an upgrade to achieve accurate segmentation of banana crowns.

To enhance the efficiency of banana crown cutting and enable intelligent cutting, a lightweight semantic segmentation model capable of accurately and swiftly segmenting banana crowns is necessary. Consequently, an upgraded DeepLabv3+ model is proposed in this study, incorporating the following enhancements:

(1) Substituting the backbone network of the traditional DeepLabv3+ model with MobilenetV2, reducing computational requirements and training time.
(2) Adding the Shuffle Attention mechanism to the Atrous Spatial Pyramid Pooling (ASPP) module and replacing the activation function with Meta-ACONC. This results in Banana-ASPP, a novel feature extraction module that facilitates the processing of high-level features.
(3) Introducing the Multi-scale Channel Attention Module (MS-CAM) to the Decoder to improve the integration of features from multiple semantics and scales.

As a result, a highly accurate and robust banana crown segmentation model is generated, poised to improve the efficiency and intelligence of banana crown cutting.

2. Materials and Methods

2.1. Construction of the Banana Crown Image Dataset

2.1.1. Image Acquisition and Data Enhancement

Since there isn't a publicly accessible banana crown image dataset, this article will create one in order to conduct its research. Because banana crown cutting is typically done in in a simple shack, the banana crown images used in this paper are acquired in the same environment, as seen in Figure 1. A total of 508 images of banana crown were taken using a 16-megapixel high-resolution camera at various distances and angles. Since banana crown cutting operations are mainly carried out during the day, the images utilized in the experiment were collected from 10 a.m. to 6 p.m.

Figure 1. Banana crown sample collection environment.

To enhance the model's generalization capacity and mitigate the risk of overfitting during training, data augmentation techniques are employed on the dataset. Through offline augmentation methods, including 90-degree rotation, brightness adjustment, and noise additon, a set of 2318 banana crown samples is generated for the purpose of this study. And the example of data augmentation is shown in the Figure 2.

Figure 2. Selected samples of data augmentation (**a**) Original image (**b**) enhanced brightness (**c**) reduced brightness (**d**) noise addition.

2.1.2. Image Annotation

As illustrated in Figure 3, the banana bunch comprises three distinct components: A represents the banana crown, B denotes the banana fingers and C signifies the banana shaft.

For augmented dataset, LabelMe was used to mark parts of the banana crown in the image, while other parts are the background. The resulting data after tagging is recorded in PASCAL VOC format. Then, use the 7:3 ratio to split the dataset into training and validation sets.

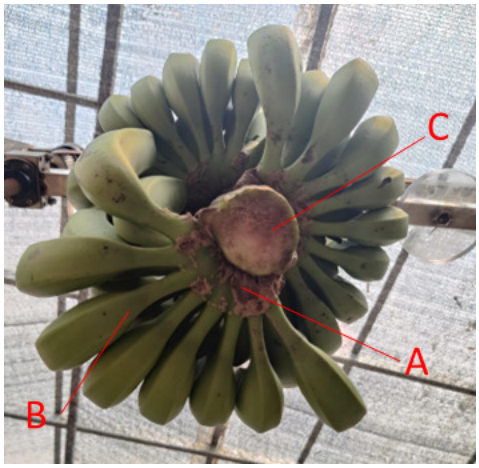

Figure 3. An explanation of each banana component.

2.2. An Overview of the DeepLabv3+ Model

The DeepLabv3+ model is the most recent generation of network models in the DeepLab series. Compared to DeepLabv1 [19], DeepLabv2 [20] and DeepLabv3 [21], it has a significant improvement in segmentation effect and segmentation accuracy. Because of its strong performance and computational efficiency, it is frequently used in the segmentation tasks of green fruits and vegetables.

The DeepLabv3+ model [22] is comprised of two principal components: the Encoder and the Decoder. The Encoder encompasses two key elements: the backbone network and the Atrous Spatial Pyramid Pooling (ASPP) module. DeepLabv3+ adopts a deeper Xception structure as its backbone network, enabling expedited computations and efficient memory utilization. Following the processing of the input image by the backbone network, two outputs are generated: high-level features and low-level features. While the low-level features are immediately handled by the Decoder, the high-level features undergo processing in the ASPP module. The ASPP module incorporates five branches, involving a 1×1 convolution and three 3×3 dilated convolutions with dilation rates of 6, 12 and 18, respectively, alongside a global average pooling operation. Building upon the input of high-level features, the ASPP module produces five outputs, which are combined to generate multi-scale information and subsequently transmitted to the Decoder.

The high-level features, after being processed by the ASPP module, undergo upsampling by a factor of four. Meanwhile, the low-level features in the Decoder segment are adjusted dimensionally via a 1 × 1 convolution operation. The fusion of these two sets of features serves as an amalgamation of detailed and semantic information, thereby leading to a substantial enhancement in segmentation performance. Ultimately, the prediction results are obtained through an additional upsampling by a factor of four, following feature optimization employing a 3 × 3 convolution.

2.3. Improved DeepLabv3+ Banana Crown Segmentation Model

The structure of the improved DeepLabv3+ banana crown segmentation model is shown in the Figure 4, with the improved parts mainly including the backbone network, Atrous Spatial Pyramid Pooling (ASPP) module, and the Decode section.

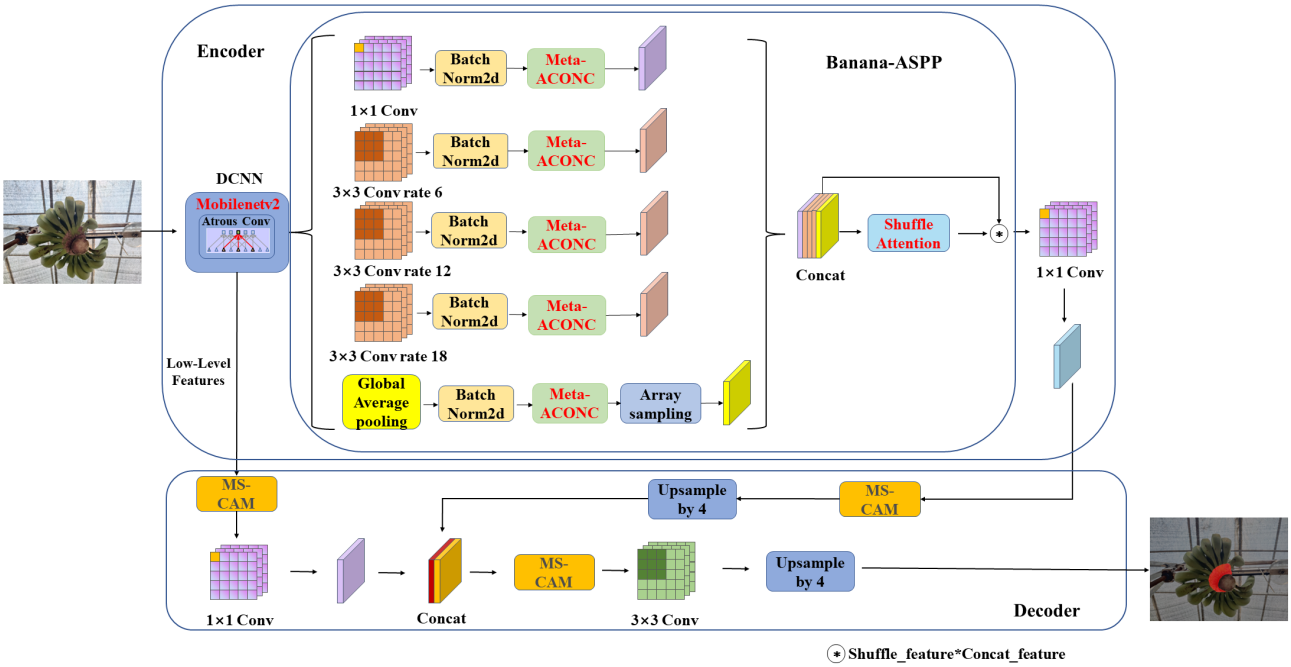

Figure 4. The structure of improved DeepLabv3+ banana crown segmentation model.

2.3.1. Backbone Network

MobilenetV2 [23] is a lightweight network specifically designed for deployment on mobile devices. It inherits the advantages of the Mobilenet family by replacing standard convolution with depthwise separable convolution [24]. The process of depthwise separable convolution consists of two main steps: depthwise convolution and pointwise convolution. In MobilenetV2, the input features are initially divided into multiple single channel features, and each of these single channel features undergoes convolution using a single 3 × 3 convolution kernel, resulting in an equal number of output features as there are channels. As depthwise convolution operates independently on each channel, it fails to effectively exploit feature information from different channels at the same spatial position. To address this limitation, pointwise convolution is employed to weight and combine the results of depthwise convolution, producing new output features. Through this approach, depthwise separable convolution reduces computational complexity by approximately two-thirds compared to traditional convolution methods.

In addition to leveraging the foundation of deep separable convolution, MobilenetV2 introduces two novel architectural components: Inverted Residuals and Linear Bottlenecks. These components are illustrated in Figure 5. The network begins by expanding the input channels through a 1 × 1 convolution in the expansion layer, facilitating the transformation of low-dimensional space to high-dimensional space. Subsequently, a 3 × 3 depthwise

convolution is applied to extract relevant features. To map the high-dimensional portion back into the low-dimensional space, the number of channels is reduced via a 1×1 convolution in the projection layer. Notably, instead of employing the ReLU activation function for data transformation, a linear bottleneck approach is adopted. The output of the linear transformation is then merged with the initial input to produce the final output, resulting in an inverse residual module nested within a linear bottleneck layer.

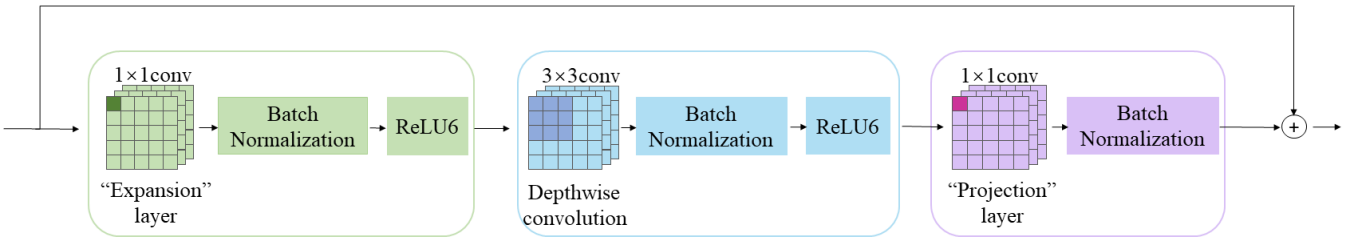

Figure 5. The structure of Inverted Residual module and Linear Bottleneck.

2.3.2. Design of the Banana-ASPP Module

To enhance the ability of the Atrous Spatial Pyramid Pooling (ASPP) module in extracting pertinent information of the banana crown, we have devised the Banana-ASPP module, as illustrated in Figure 6. The improvements incorporated into this module can be categorized as follows:

(1) The utilization of the Shuffle Attention mechanism enables processing of the features got from the five branches to obtain corresponding weights. These weights are then multiplied with their respective features to suppress irrelevant features such as the background, thereby enhancing the module's focus on the characteristic features of the banana crown. The structure of the Shuffle Attention mechanism as depicted in the figure.

(2) Replaced the conventional ReLU activation function in the ASPP module with the Meta-ACONC activation function. This substitution enables the module to achieve adaptive switching between linear and nonlinear activations during the feature extraction process. As a result, the model's effectiveness in extracting relevant features from banana crowns is improved.

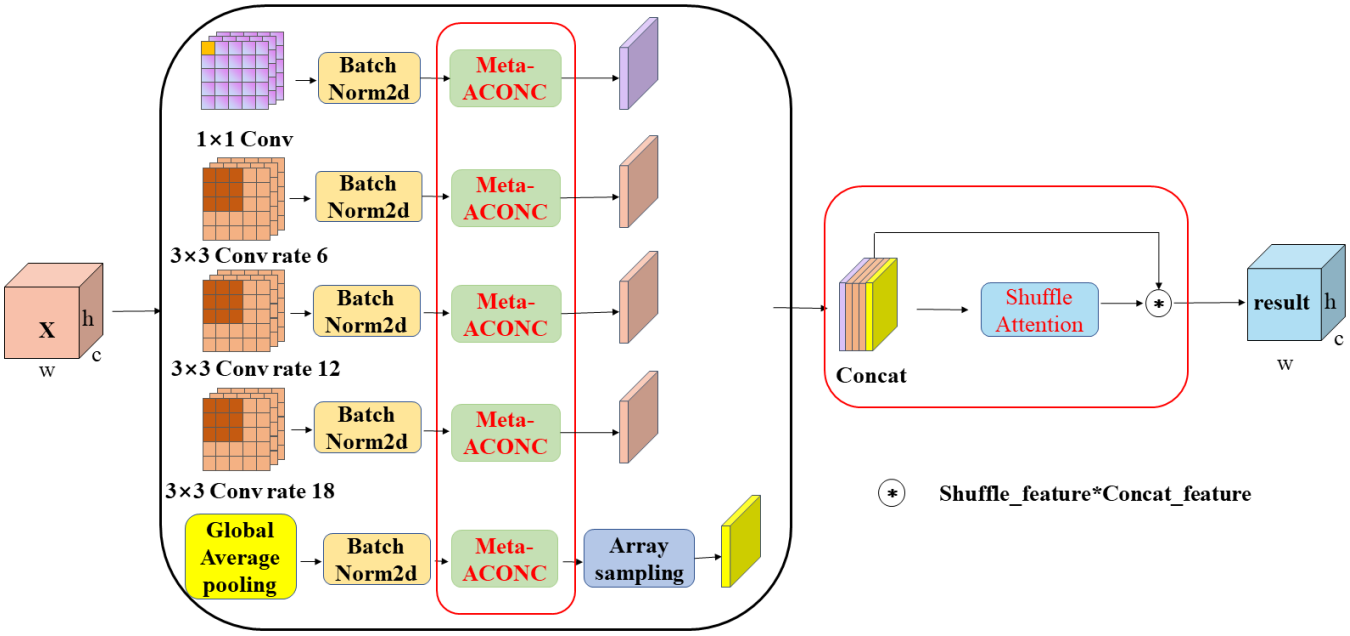

Figure 6. The structure of the Banana-ASPP module.

(1) Shuffle Attention Mechanism

The structure of Shuffle Attention Mechanism [25] is depicted in Figure 7. In this structure, the input data is initially sorted for computation, which significantly reduces the computational workload. Subsequently, the data for each group is partitioned into two sections: one section is processed by the GroupNorm (GN) which is a spatial attention method, while the other section undergoes channel attention mechanism processing. Following these individual processes, the information from both sections is concatenated to consolidate the extracted features. Finally, Channel Shuffle is employed to facilitate information flow between distinct groups. By effectively integrating the spatial attention mechanism and the channel attention mechanism, Shuffle Attention enables the ASPP module to extract salient characteristics from critical regions and channels.

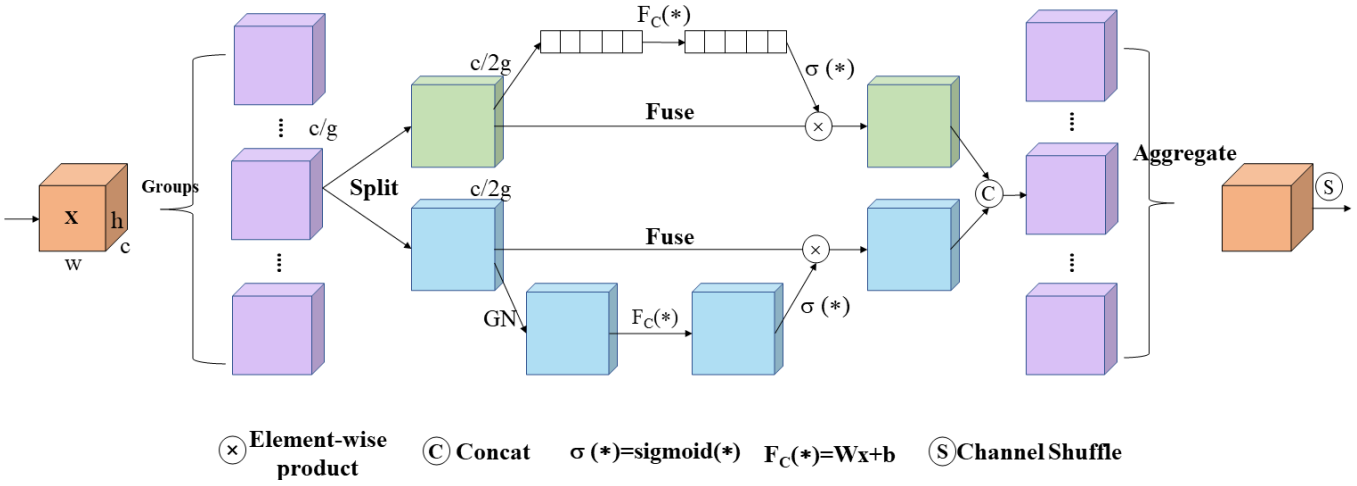

Figure 7. The structure of Shuffle Attention Mechanism.

(2) Meta-ACONC

Meta-ACONC [26] is an activation function that can implement adaptive switching between linear and nonlinear, and its two most significant components are ACONC expressions and hyperparameters β. The ACON family is a set of activation function variants generated by smoothing the Maxout activation function family, ACONC works best in the ACON family, and its function expression is indicated in the Equation (1).

$$f_{ACONC}(X) = (p_1 - p_2)x \cdot [\beta(p_1 - p_2)x] + p_2 x \quad (1)$$

In the formula, x is the input value of the function, the parameters p_1 and p_2 in the formula are two learnable parameters that control the upper and lower bounds of the function, respectively, the initial values of p_1 and p_2 are a random tensor obtained by the randn function. And the parameter β is in charge of dynamically adjusting whether the activation function is linear or nonlinear, allowing neurons to activate or not activate adaptively. The function is nonlinear as β approaches positive infinity, linear as β approaches zero. Adaptive learning is used to acquire the value of parameter β, and general adaptive learning is made up of three schemes: layer learning, channel learning, and pixel learning, Meta-ACONC employs adaptive learning of the entire channel to minimize parameters.

2.3.3. Multi-Scale Channel Attention Module

The multi-scale channel attention module, also known as MS-CAM [27], is suggested to more effectively combine the characteristics of several semantics or scales. To improve the model's segmentation accuracy and robustness, MS-CAM is added to the Decoder, which is placed as shown in Figure 8. First, we added MS-CAM after two input features of the Decoder, so that the MS-CAM can reprocess the features of these two outputs, fuse the

local feature information with the global feature information, and output the banana crown information to the decoder. The MS-CAM is then added after the high-level and low-level features have been spliced, to process them such that the low-level and high-level semantic features can be better merged and interacted.

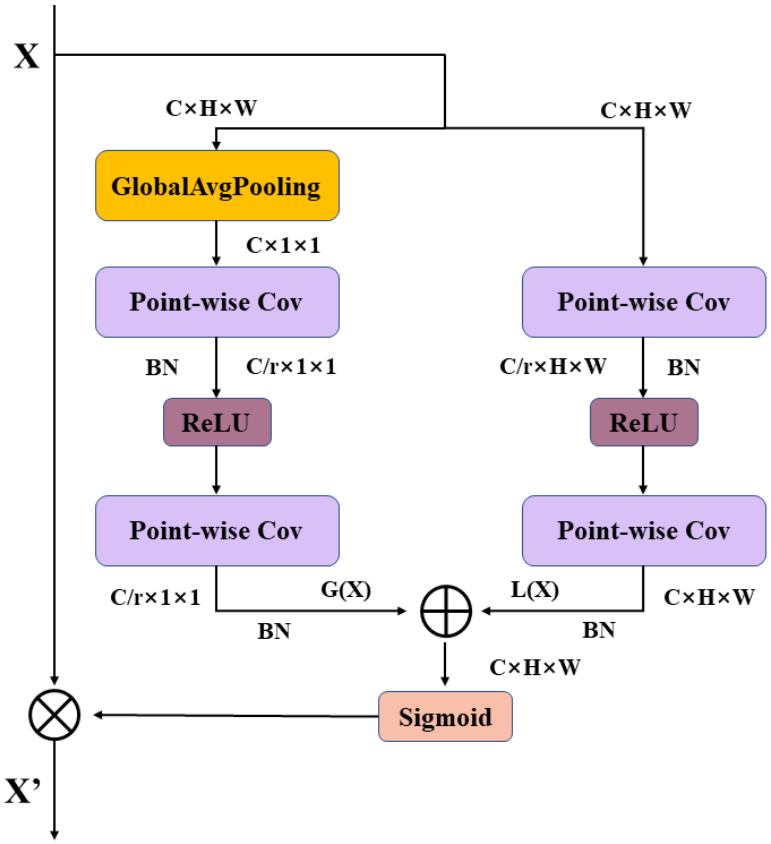

Figure 8. The structure of the Multi-Scale Channel Attention Module.

The MS-CAM structure is given in the Figure 8, it consists of two branches, the left of which is the global channel branch, and the global average pooling is used to make the acquired feature $G(x)$ contain global information. The right branch is the local channel branch, and it is possible to calculate the local channel information $L(x)$ from it. Finally, the MS-CAM module fuses the global channel information $G(x)$ and the local channel information $L(x)$, and the output X' is generated after the attention operation on the input feature X is output by the weight value, as stated in Equation (2).

$$X' = X \oplus M(X) = X \otimes \sigma(L(X) \oplus G(X)) \tag{2}$$

In the above formula, $\oplus$ denotes the broadcast addition operation. Because global channel information employs the global average pooling operation, the size of global channel information $G(x)$ differs from local channel information $L(x)$, so it must be added using the broadcast addition operation. $\otimes$ means that the corresponding elements of the two features are multiplied. $\sigma(L(X) \oplus G(X))$ represents MS-CAM module operation, in other words, the global and local information are combined, and the output value range is specified between (0, 1).

2.4. Evaluation Metrics

In this study, we evaluate the model's performance using certain evaluation metrics, such as Mean Pixel Accuracy (MPA), Mean Intersection over Union (MIoU), the number of parameters and Frames Per Second (FPS). MPA and MIoU were two evaluation metrics

used to assess the model's performance on the test set. The FPS represents the inverse of the time required for a model to process an image. A higher FPS value indicates that the model can reason about an image in less time, thus indicating faster processing speed. The number of parameters is used to evaluate the lightness of the model, and FPS is used to evaluate the speed of the model.

The mathematical expression of MPA and MIoU are shown in Equations (3) and (4). In the formula, n denotes the number of categories to be divided; In this article, n is 1; p_{ii} is the number of pixels anticipated to belong to category I; That is, the number of pixels successfully predicted. The number of pixels that belong to category I and are predicted to belong to category J is represented by p_{ij}, while the number of pixels that belong to category J but are predicted to belong to category I is represented by p_{ji}.

$$MPA = \frac{1}{n+1}\sum_{i=0}^{n}\frac{p_{ii}}{\sum_{j=0}^{n}p_{ij}} \times 100 \tag{3}$$

$$MIoU = \frac{1}{n+1}\sum_{i=0}^{n}\frac{p_{ii}}{\sum_{j=0}^{n}p_{ij} + \sum_{j=0}^{n}p_{ji} - p_{ii}} \times 100 \tag{4}$$

3. Results and Analysis

3.1. Experimental Environment

To provide a quantitative demonstration of the efficacy of the proposed methods in this study, all experiments will be conducted within a standardized experimental environment. The model training will be performed on a computer system equipped with a Windows 10 operating system and an Intel Core i5-12400F@2.50 GHz CPU. The graphics card employed is the NVIDIA GTX3060 with 12 GB of video memory. For programming the model, Python 3.7 will be utilized, and the PyTorch deep learning framework, version 1.7.1, will serve as the foundation for training. The CUDA11.1 architecture will be adopted as the unified computing device framework throughout the experiments.

3.2. Parameter Settings

In this paper, the MobilenetV2 backbone feature extraction network is pre-trained on the ImageNet dataset using the transfer learning method, and the MobilenetV2 pre-trained weights based on DeepLabv3+ are obtained. Based on the pre-trained weights, the parameters of the MobilenetV2 network model are adjusted for the banana crown image dataset. The dataset contains 2318 images. The training set consisted of 1619 images chosen at random, whereas the test set consisted of 695 images. The input image size is uniformly adjusted to 512 × 512 during the training phase. The training batch size is set to 8 for freeze training and 4 for non-frozen training, and the total number of training epochs is 200, where the freeze training period is 50 and the non-frozen training period is 150. The starting rate of learning is 0.007. We use the stochastic gradient descent optimizer and the Cos learning rate update technique.

3.3. Analysis of Experimental Results

3.3.1. Backbone Network Validation

The improved DeepLabv3+ model will be applied to banana crown cutting operations, necessitating the deployment of the model onto banana crown cutting robots. This entails strict requirements on both the prediction speed and the number of model parameters. Consequently, the lightweight network, MobilenetV2, was selected as the backbone network for the model instead of Xception. In this section, a comparison is conducted between DeepLabv3+-MobilenetV2 and the original DeepLabv3+ as well as DeepLabv3+-Resnet50. The training and testing phases are performed within the same experimental environment, aiming to validate the effectiveness of the backbone network selection. The results of this comparison are presented in Table 1.

As shown in the Table 1, DeepLabv3+ with various backbone networks has diverse effects on the banana crown verification set, with DeepLabv3+-MobilenetV2 having a MIoU of 84.98%, MPA of 90.19%, parameters of 5.814 M, and FPS of 41.41 f/s. Among the models utilizing different backbone networks, DeepLabv3+-MobileNetV2 achieves the highest MIoU, indicating superior performance in terms of segmentation accuracy. However, its MPA is slightly lower than the model employing ResNet50 as the backbone network. Nonetheless, DeepLabv3+-MobileNetV2 offers the advantage of faster detection speed and a smaller number of parameters. When compared to the original DeepLabv3+, the MPA is essentially the same, the MIoU is increased by 1.17%, the number of parameters is reduced by 89.37% and the FPS is increased by 27.43 f/s. Replacing the network backbone with MobilenetV2 not only achieve the model's lightweight and improve model speed, but also secures the model's accuracy for object segmentation. The reduction in parameter count and the improvement in model speed achieved by replacing the backbone network can be attributed to the implementation of enhanced deep separable convolution within the MobilenetV2 architecture. This integration facilitates the optimization of both spatial and temporal complexity within the network, leading to a reduction in parameter quantity. As a result, the overall efficiency of the model is enhanced, resulting in improved computational performance and faster inference speeds. This facilitates the later deployment of the model on the banana crown cutting devide and validated the decision to use MobilenetV2 as the backbone network.

Table 1. Comparison of the influence of different backbone networks.

Model	Backbone Network	MIoU/%	MPA/%	Parameters/M	FPS/(f/s)
DeepLabv3+	MobilenetV2	84.98	90.19	5.814	41.41
	Xception	83.81	90.20	54.709	13.98
	Resnt50	71.92	95.56	40.510	30.20

3.3.2. Comparison of Attention Mechanism Effects

To test if the use of the MS-CAM enhances the model's segmentation effect and whether the MS-CAM utilized has advantages, seven newer attention modules are added to the the location of MS-CAM, and the same dataset is used for training and validation. Table 2 lists the evaluation indicators for each of the attention modules utilized for comparison, including Convolutional Block Attention Module (CBAM) [28], Polarized Self-Attention (PSA) [29], SimAM [30], Coordinate Attention (CA) [31] , S2Attention [32], Double Attention (A2) [33] and Criss Cross Attention (CCA) [34].

Table 2. Comparison of attention mechanism effects.

Model	Backbone Network	Attention Mechanism	MIoU/%	MPA/%	FPS/(f/s)
DeepLabv3+	MobilenetV2	/	84.98	90.19	41.41
		CBAM	85.21	90.36	63.38
		PSA	85.29	90.29	61.52
		SimAM	85.25	90.17	62.98
		CA	85.20	90.85	62.51
		S2Attention	85.39	90.82	63.98
		A2	85.22	90.71	62.71
		CCA	85.36	90.51	64.15
		MS-CAM	85.40	90.75	63.42

According to the Table 2, the adoption of different attention modules yields varying degrees of improvement in the segmentation performance of the DeepLabv3+ model. Specifically, with MS-CAM as the attention mechanism, the MIoU reached 85.4%, while the MPA achieved a value of 90.75%. Furthermore, the model has a FPS of 63.42 f/s. When compared to the DeepLabv3+-MobilenetV2 model, MIoU increased by 0.42%, MPA

increased by 0.56% and FPS increased by 22.01 f/s, demonstrating the effectiveness and rationality of selecting MS-CAM. In comparison to alternative attention mechanisms, the model incorporating MS-CAM demonstrated the highest MIoU and FPS values, thereby indicating its superiority in terms of both segmentation effectiveness and model speed. While the employment of MS-CAM does not yield the highest MPA value, with a slight 0.1% margin from the best performer, the consideration of other evaluation metrics establishes MS-CAM as the optimal choice. It presents certain advantages over alternative attention mechanisms, contributing to the enhancement of model segmentation performance.

3.3.3. Comparison of Different Activation Functions

To validate the rationality of selecting the Meta-ACONC activation function, we replaced the commonly used ReLU activation function with a more recent activation function in the Atrous Spatial Pyramid Pooling (ASPP) module. We then compared the evaluation metrics of the model utilizing the Meta-ACONC activation function with those of models employing alternative activation functions. The activation functions employed in this experiment included ReLU [35], FReLU [36], DyReLU [37], Hardswish [38] and Meta-ACONC [26].

As illustrated in Table 3, the utilization of Meta-ACONC as the activation function in ASPP yielded an MIoU of 85.49%, MPA of 90.41% and FPS of 62.29 f/s. Regarding MIoU, the model employing Meta-ACONC outperformed other models by 0.24%, implying its capability to effectively delineate object boundaries and generate more precise segmentation masks, consequently enhancing the model's segmentation performance. Conversely, a slight reduction in MPA was observed when Meta-ACONC replaced the original activation function, suggesting a minor decrease in pixel-level segmentation accuracy due to Meta-ACONC's nonlinearity, leading to some instances of pixel misclassification or inaccuracy. In terms of FPS, the usage of Meta-ACONC results in a modest loss in speed since Meta-ACONC must implement the selection of the activation function from linear to nonlinear, which necessitates an increase in computation, resulting in a decrease in model speed. Overall, employing Meta-ACONC as an activation function in ASPP sacrifices some pixel precision and model speed, but the overall segmentation effect of the model on banana crowns improves, hence using Meta-ACONC is reasonable.

Table 3. Comparison of different activation functions.

Model	Backbone Network	ASPP	Activation Function	MIoU/%	MPA/%	FPS/(f/s)
DeepLabv3+	MobilenetV2	Shuffle-ASPP	ReLU	85.25	90.51	64.54
			FReLU	84.84	90.92	62.16
			DyReLU	85.15	91.10	60.60
			Hardswish	84.89	90.44	64.16
			Meta-ACONC	85.49	90.41	62.29

3.3.4. Ablation Experiment

To test the effectiveness of adding the Banana-ASPP module as well as the feasibility of added MS-CAM to the Decoder part, three different groups of improvement schemes were set up for ablation experiments using semantic segmentation evaluation indicators. Table 4 displays the experimental outcomes, where "✓" denotes that the designated module was utilized in the experiment and "×" denotes that it was not introduced.

(1) Group one: On the basis of replacing the backbone network of DeepLabv3+ with MobilenentV2, add the Shuffle Attention module to the Atrous Spatial Pyramid Pooling Module to process the features.
(2) Group two: On the basis of Group one, replace the activation function in the Atrous Spatial Pyramid Pooling Module with Meta-ACONC.
(3) Group three: On the basis of Group two, MS-CAM is added to the Decoder part to fuse features of different semantics or scales.

Table 4. Ablation experiment of Improved DeepLabv3+ model.

Model	Backbone Network	Shuffle Attention	Meta-ACONC	MS-CAM	MIoU/%	MPA/%	FPS/(f/s)
DeepLabv3+	MobilenetV2	×	×	×	84.98	90.19	41.41
		✓	×	×	85.25	90.51	64.54
		✓	✓	×	85.49	90.41	62.29
		✓	✓	✓	85.75	91.41	61.05

According to the Table 4, when the shuffle attention module is added to the Atrous Spatial Pyramid Pooling module and the activation function is replaced with Meta-ACONC, MIoU reached 85.49%, MPA reached 90.41%, and FPS reached 62.29 f/s. Regarding MIoU, the model incorporating the Banana-ASPP module demonstrates a superior performance to DeepLabv3+-MobilenetV2, with an improvement of 0.51%. The inclusion of Meta-ACONC in the model leads to an increase in computational complexity, resulting in a slight reduction in both MPA and FPS values when replacing ReLU with Meta-ACONC. Nevertheless, compared to DeepLabv3+-MobilenetV2, there is a noticeable enhancement of 0.22% in MPA and a substantial improvement of 50.42% in model speed, thereby highlighting the effectiveness of the Banana-ASPP module in enhancing both segmentation performance and model speed. By incorporating the Banana-ASPP module and adding MS-CAM to the decoder, the model achieves an MIoU of 85.75%, an MPA of 91.41% and a FPS of 61.05 f/s. Compared to the model utilizing only the Banana-ASPP module, there is a 0.26% improvement in MIoU and a 1% improvement in MPA, while FPS experiences a slight reduction of 1.24 f/s. Overall, the integration of MS-CAM into the decoder section further enhances the accuracy of the banana crown segmentation model while maintaining its speed.

In order to show the effect of model improvement more intuitively, original model, MobilenetV2, MobilenentV2+Shuffle Attention, MobilenentV2+Shuffle Attention+Meta-ACONC and MobilenentV2+Shuffle Attention+ Meta-ACONC+MS-CAM were selected for testing. The images are labeled by LabelMe, and the results are used for comparison. Selecting three differrent images as the input images, the result predicted by the model is shown in Figure 9.

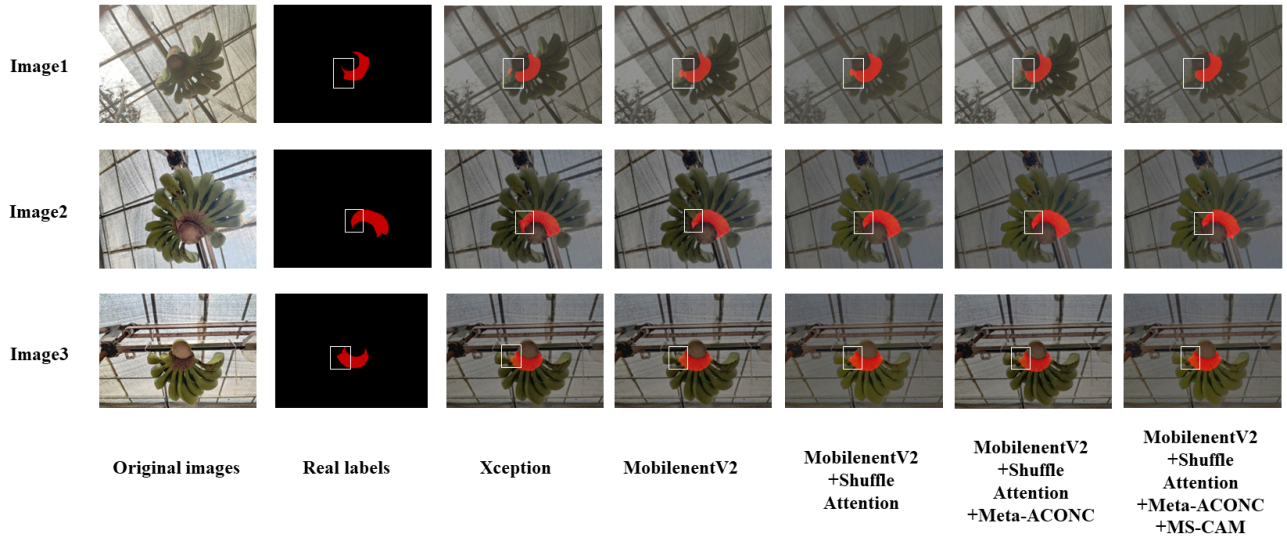

Figure 9. A comparison of the model prediction effects.The white box in the figure is to better represent the improved effect of the model.

As can be seen from Figure 9, as indicated in the white box, the results predicted by model are contrasted to the results obtained by the LabelMe marker. It shows that the

original model cannot achieve the segmentation of banana crowns well. When observing the predicted results from left to right, it becomes evident that they progressively approach the results obtained through LabelMe annotations. Specifically, the regions corresponding to the edges of the banana crown and smaller areas exhibit better segmentation when utilizing the MobilenetV2+Shuffle Attention+Meta-ACONC+MS-CAM configuration. This finding suggests that the improved DeepLabv3+ model performs superior segmentation of the banana crown compared to the original model.

3.3.5. Compare with Other Models

To evaluate and compare the segmentation performance of the proposed method with other established models, a range of widely-used semantic segmentation models including PSPNet, UNet, HRNetV2, and the original DeepLabv3+ model are selected for the experiment in this study. And these models are combined with commonly utilized backbone networks for the purpose of conducting comprehensive comparisons. The evaluation metrics used in the experiments are presented in Tables 5 and 6.

Table 5. Comparison with other lightweight networks

Model	Backbone Network	MIoU/%	MPA/%	Parameters/M	FPS/(f/s)
Improved DeepLabv3+	/	85.75	91.41	5.881	61.05
PSPNet	MobilenetV2	77.87	84.42	2.376	78.43
HrnetV2	HrnetV2-w18	81.40	90.11	9.637	17.69

According to the results presented in Table 5, when comparing the proposed model in this study with commonly used lightweight networks, the improved DeepLabv3+ model demonstrates superior performance in terms of both MIoU and MPA. Specifically, PSPNet and HRNetV2 achieve MIoU values of 77.87% and 81.40%, and MPA values of 84.42% and 90.11%, respectively. When compared to the enhanced DeepLabv3+, both the MIoU and MPA values are lower, indicating that the upgraded DeepLabv3+ model outperforms these lightweight networks in terms of segmentation accuracy. Despite PSPNet exhibiting a slightly better performance in terms of model parameters and speed, considering all the evaluation metrics, the improved DeepLabv3+ remains the superior choice. This is primarily attributed to the fact that the improved DeepLabv3+ model strikes a better balance between lightweight design and segmentation performance, thus enabling more accurate segmentation of banana crowns.

Table 6. Contrast with the commonly used semantic segmentation models.

Model	Backbone Network	MIoU/%	MPA/%	Parameters/M	FPS/(f/s)
Improved DeepLabv3+	/	85.75	91.41	5.881	61.05
DeepLabv3+	Xception	83.81	90.20	54.709	13.98
	Resnt50	71.92	95.56	40.510	30.20
PSPNet	Resnet50	77.56	85.96	46.707	16.24
HrnetV2	HrnetV2-w32	82.75	90.95	65.848	10.59
UNet	Resnet50	81.89	89.84	43.934	11.46
	Vgg	85.13	92.08	24.892	7.48

From the Table 6, when the suggested network is compared to the commonly used ordinary semantic segmentation network, the improved DeepLabv3+ MIoU is shown to be superior to other networks, showing that the improved DeepLabv3+ segmentation is better. In terms of the number of parameters and FPS, the enhanced DeepLabv3+ outperforms

previous networks, showing that it is lighter and has faster model speed, making it more suited for mobile deployment. In terms of MPA, Unet-Vgg's 92.08% and DeepLabv3+-Resnet50's 95.56% are better than the improved DeepLabv3+, first of all, DeepLabv3+-Resnet50 performs poorly in other aspects and cannot meet the requirements. Then, while UNet-Vgg is higher in MIoU values than the improved DeepLabv3+, its slower model speed does not match the need for lightweight and fast banana crown cutting operations, so the upgraded DeepLabv3+ is still a model that can finish the banana crown segmentation better than the commonly used ordinary semantic segmentation model.

According to the data and analysis in Tables 5 and 6, the improved DeepLabv3+ achieves segmentation performance while maintaining model lightweight, and the improvement is relatively successful.

4. Conclusions

This paper proposes a method for the segmentation of banana crowns based on an improved DeepLabv3+ model, aiming to achieve accurate and rapid segmentation while enabling deployment on mobile devices. Firstly, the traditional backbone network of the DeepLabv3+ model is replaced with MobilenetV2, reducing the model's weight, training time, and the number of parameters, while improving model speed. Then, the Atrous Spatial Pyramid Pooling (ASPP) module is enhanced by adding the Shuffle Attention mechanism and switching out the activation function for Meta-ACONC, creating a new feature extraction module called Banana-ASPP that excels at extracting high-level features. Furthermore, the Multi-scale Channel Attention Module (MS-CAM) is incorporated into the Decoder to effectively combine attributes from various meanings and scales, resulting in more comprehensive information on the banana crown.

According to experimental findings, the proposed method for banana crown segmentation based on the improved DeepLabv3+ model achieves a Mean Intersection over Union (MIoU) of 85.75%, a Mean Pixel Accuracy (MPA) of 91.41%, with model parameters totaling 5.881 M and a processing speed of 61.05 f/s. The experiments demonstrate that this research's suggested method for banana crown segmentation based on the improved DeepLabv3+ model can effectively segment the banana crown, providing substantial technological support for adaptive diameter adjustment of the banana crown cutting device.

In future studies, we plan to collect banana crown images from various locations and cultivars to create more diverse datasets, enabling the model to learn generic feature representations for banana crowns and enhancing its applicability. Additionally, we aim to explore the utilization of this model in combination with an RGB-D camera to determine the cutting radius of the banana crown. Further modifications to the model are necessary to increase its speed and improve its compatibility with automation devices for banana crown cutting. This includes reducing the number of model parameters and enhancing the performance of real-time image segmentation.

Author Contributions: Conceptualization, J.H.; methodology, J.D.; software, J.H.; validation, J.H. and S.Y.; formal analysis, J.H.; investigation, J.D.; resources, J.D.; data curation, J.O., X.O., Y.L., M.X., Q.J. and H.W.; writing—original draft preparation, J.H.; writing—review and editing, J.D.;visualization, J.H.; supervision, J.D.; project administration, J.D. and Z.Y.; funding acquisition, J.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Natural Science Foundation of China (Grant No. 32271996), Guangdong Laboratory for Lingnan Modern Agriculture Project (Grant No. NT2021009), the open competition program of top ten critical priorities of Agricultural Science and Technology Innovation for the 14th Five-Year Plan of Guangdong Province (Grant No. 2022SDZG03), China Agriculture Research System of MOF and MARA (Grant No. CARS-31-11), Guangdong Provincial Special Fund For Modern Agriculture Industry Technology Innovation Teams (Grant No. 2023KJ109).

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. The data are not publicly available due to the privacy policy of the organization.

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

References

1. Xie, J. Fruit scientific research in New China in the past 70 years: Banana. *J. Fruit Sci.* **2019**, *36*, 1429–1440.
2. Fu, L.; Yang, Z.; Wu, F.; Zou, X.; Lin, J.; Cao, Y.; Duan, J. YOLO-Banana: A lightweight neural network for rapid detection of banana bunches and stalks in the natural environment. *Agronomy* **2022**, *12*, 391. [CrossRef]
3. Zhenzhen, X.; Liang, Z.; Xiaojun, L.; Jie, S.; Jin, S.; Ming, Z. Classification, integration of preliminary processing technology in banana producing areas of China. *Trans. Chin. Soc. Agric. Eng.* **2015**, *31*, 332–336.
4. Tang, Y.; Chen, M.; Wang, C.; Luo, L.; Li, J.; Lian, G.; Zou, X. Recognition and localization methods for vision-based fruit picking robots: A review. *Front. Plant Sci.* **2020**, *11*, 510. [CrossRef]
5. Zheng, T.; Jiang, M.; Feng, M. Vision based target recognition and location for picking robot. *Chin. J. Sci. Instrum.* **2021**, *42*, 28–51.
6. Cui, Y.; Su, S.; Wang, X.; Tian, Y.; Li, P.; Zhang, F.; et al. Recognition and feature extraction of kiwifruit in natural environment based on machine vision. *Nongye Jixie Xuebao = Trans. Chin. Soc. Agric. Mach.* **2013**, *44*, 247–252.
7. Wuzor, G.K.; Woods, N.C. On tree guava fruit detection and yield estimation. *Int. J. Sci. Engg. Res.* **2020**, *11*, 723–731.
8. Marlinda, L.; Fatchan, M.; Widiyawati, W.; Aziz, F.; Indrarti, W. Segmentation of Mango Fruit Image Using Fuzzy C-Means. *Sinkron* **2021**, *5*, 275–281. [CrossRef]
9. Li, Q.; Jia, W.; Sun, M.; Hou, S.; Zheng, Y. A novel green apple segmentation algorithm based on ensemble U-Net under complex orchard environment. *Comput. Electron. Agric.* **2021**, *180*, 105900. [CrossRef]
10. Hussain, M.; He, L.; Schupp, J.; Lyons, D.; Heinemann, P. Green fruit segmentation and orientation estimation for robotic green fruit thinning of apples. *Comput. Electron. Agric.* **2023**, *207*, 107734. [CrossRef]
11. Wang, Z.; Zhang, Z.; Lu, Y.; Luo, R.; Niu, Y.; Yang, X.; Jing, S.; Ruan, C.; Zheng, Y.; Jia, W. SE-COTR: A Novel Fruit Segmentation Model for Green Apples Application in Complex Orchard. *Plant Phenomics* **2022**, *2022*, 0005. [CrossRef] [PubMed]
12. Liu, J.; Zhao, Y.; Jia, W.; Ji, Z. DLNet: Accurate segmentation of green fruit in obscured environments. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, 7259–7270. [CrossRef]
13. Ma, J.; Du, K.; Zheng, F.; Zhang, L.; Gong, Z.; Sun, Z. A recognition method for cucumber diseases using leaf symptom images based on deep convolutional neural network. *Comput. Electron. Agric.* **2018**, *154*, 18–24. [CrossRef]
14. Yan, C.; Chen, Z.; Li, Z.; Liu, R.; Li, Y.; Xiao, H.; Lu, P.; Xie, B. Tea Sprout Picking Point Identification Based on Improved DeepLabV3+. *Agriculture* **2022**, *12*, 1594. [CrossRef]
15. Zhang, Y.; Wu, M.; Li, J.; Yang, S.; Zheng, L.; Liu, X.; Wang, M. Automatic non-destructive multiple lettuce traits prediction based on DeepLabV3+. *J. Food Meas. Charact.* **2023**, *17*, 636–652. [CrossRef]
16. Yu, H.; Che, M.; Yu, H.; Zhang, J. Development of Weed Detection Method in Soybean Fields Utilizing Improved DeepLabv3+ Platform. *Agronomy* **2022**, *12*, 2889. [CrossRef]
17. Xiangwu, D.; Song, L.; Long, Q.; Shuting, Y. Method study on semantic segmentation of weeds at seedling stage in paddy fields based on DeepLabV3+ model. *J. Chin. Agric. Mech.* **2023**, *44*, 174.
18. Li, K.; Zhang, L.; Li, B.; Li, S.; Ma, J. Attention-optimized DeepLab V3+ for automatic estimation of cucumber disease severity. *Plant Methods* **2022**, *18*, 109. [CrossRef]
19. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv* **2014**, arXiv:1412.7062.
20. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 834–848. [CrossRef]
21. Chen, L.C.; Papandreou, G.; Schroff, F.; Adam, H. Rethinking atrous convolution for semantic image segmentation. *arXiv* **2017**, arXiv:1706.05587.
22. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 801–818.
23. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR), Munich, Germany, 8–14 September 2018; pp. 4510–4520.
24. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 1251–1258.
25. Zhang, Q.L.; Yang, Y.B. Sa-net: Shuffle attention for deep convolutional neural networks. In Proceedings of the ICASSP 2021–2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 6–11 June 2021; pp. 2235–2239.
26. Ma, N.; Zhang, X.; Liu, M.; Sun, J. Activate or not: Learning customized activation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 19–25 June 2021; pp. 8032–8042.

27. Dai, Y.; Gieseke, F.; Oehmcke, S.; Wu, Y.; Barnard, K. Attentional feature fusion. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision((WACV)), Virtual Conference, 5–9 January 2021; pp. 3560–3569.
28. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
29. Liu, H.; Liu, F.; Fan, X.; Huang, D. Polarized self-attention: Towards high-quality pixel-wise regression. *arXiv* **2021**, arXiv:2107.00782.
30. Yang, L.; Zhang, R.Y.; Li, L.; Xie, X. Simam: A simple, parameter-free attention module for convolutional neural networks. In Proceedings of the International Conference on Machine Learning, PMLR, Virtual Event, 18–24 July 2021; pp. 11863–11874.
31. Hou, Q.; Zhou, D.; Feng, J. Coordinate attention for efficient mobile network design. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Virtual Conference, 19–25 June 2021; pp. 13713–13722.
32. Chen, L.; Wu, Z.; Ling, J.; Li, R.; Tan, X.; Zhao, S. Transformer-s2a: Robust and efficient speech-to-animation. In Proceedings of the ICASSP 2022—2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, 23–27 May 2022; pp. 7247–7251.
33. Chen, Y.; Kalantidis, Y.; Li, J.; Yan, S.; Feng, J. A^2-Nets: Double Attention Networks. In Proceedings of the Neural Information Processing Systems (NIPS), Montreal, ON, Canada, 3–8 December 2018; pp. 350–359;
34. Huang, Z.; Wang, X.; Huang, L.; Huang, C.; Wei, Y.; Liu, W. Ccnet: Criss-cross attention for semantic segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision(ICCV), Seoul, South Korea, 27 October–2 November 2019; pp. 603–612.
35. Glorot, X.; Bordes, A.; Bengio, Y. Deep sparse rectifier neural networks. In Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, Fort Lauderdale, FL, USA, 11–13 April 2011; pp. 315–323; JMLR Workshop and Conference Proceedings.
36. Ma, N.; Zhang, X.; Sun, J. Funnel activation for visual recognition. In Proceedings of the Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 351–368;
37. Chen, Y.; Dai, X.; Liu, M.; Chen, D.; Yuan, L.; Liu, Z. Dynamic relu. In Proceedings of the Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 351–367;
38. Howard, A.; Sandler, M.; Chu, G.; Chen, L.C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for mobilenetv3. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1314–1324.