Crop Organ Segmentation and Disease Identification Based on Weakly Supervised Deep Neural Network

Yang Wu and Lihong Xu *

College of Electronics and Information Engineering, Tongji University, Cao’an Road, NO.4800, Shanghai 201804, China; wu_tim@tongji.edu.cn
* Correspondence: xulhk@163.com; Tel.: +86-136-363-62684

Received: 4 August 2019; Accepted: 8 November 2019; Published: 10 November 2019

Abstract: Object segmentation and classification using the deep convolutional neural network (DCNN) has been widely researched in recent years. On the one hand, DCNN requires large data training sets and precise labeling, which bring about great difficulties in practical application. On the other hand, it consumes a large amount of computing resources, so it is difficult to apply it to low-cost terminal equipment. This paper proposes a method of crop organ segmentation and disease recognition that is based on weakly supervised DCNN and lightweight model. While considering the actual situation in the greenhouse, we adopt a two-step strategy to reduce the interference of complex background. Firstly, we use generic instance segmentation architecture—Mask R-CNN to realize the instance segmentation of tomato organs based on weakly supervised learning, and then the disease recognition of tomato leaves is realized by depth separable multi-scale convolution. Instance segmentation algorithms usually require accurate pixel-level supervised labels, which are difficult to collect, so we propose a weakly supervised instance segmentation assignment to solve this problem. The lightweight model uses multi-scale convolution to expand the network width, which makes the extracted features richer, and depth separable convolution is adopted to reduce model parameters. The experimental results showed that our method reached higher recognition accuracy when compared with other methods, at the same time occupied less memory space, which can realize the real-time recognition of tomato diseases on low-performance terminals, and can be applied to the recognition of crop diseases in other similar application scenarios.

Keywords: weakly supervised; leaf; instance segmentation; disease recognition

1. Introduction

Biotic stresses are the main factors that limit crop cultivation. Biotic stresses can lead to a significant reduction in output, which can bring huge losses to the agricultural economy. Therefore, the early identification of disease is critical to the selection of the right treatment [1], and it is also an important prerequisite for reducing crop losses and using less pesticide. All crops are susceptible to disease. On one hand, diseases affect the yield and quality. On the other hand, excessive chemical controls leave drug residues, which results in environmental pollution. With the improvement of people’s living standards, the need for quality of crops is more urgent. Therefore, early diagnosis and early treatment are problems that must be solved.

In recent years, there have been relatively few studies using neural network technology to identify plant diseases. Most of the researches focused on the problem of segmentation and extraction of plant leaf image information. At present, domestic and foreign researchers have focused on the leaf image segmentation of plants on arabidopsis, rice, barley, and other plants, and the purpose is to accurately segment each leaf of the plant to display the image information of the whole plant [2,3]. In addition, by segmenting and extracting the image of the lesion area on the leaf, it can help to prevent pests.
and diseases in crop production based on image information [4,5], which is also a hotspot of blade segmentation. Kawasaki et al. [6] developed a plant disease diagnosis system to identify two leaf diseases of cucumber by designing a convolutional neural network (CNN) network. Amara et al. [7] and Brahimi et al. [8] used the fine-tuning method to classify and identify leaf diseases while using deep convolutional neural network (DCNN), which can achieve satisfactory recognition effect. Sun et al. [9] improved the traditional AlexNet model, using CNNs model combining batch normalization and global pooling to identify a variety of leaf diseases. These studies demonstrate the feasibility and effectiveness of applying DCNN to the field of leaf disease identification. However, these models have a large number of parameters, high memory requirements, along with long training time, and they are difficult to use, let alone to adapt on low-cost terminals. Additionally, the identification of the above crop diseases is when the obvious symptoms appear in the late stage of crop disease, and the significance is weaker than early warning in practical application. Disease diagnosis is an important part of precision agriculture. Precision agriculture refers to the combination of remote sensing (RS), geographic information system (GIS), global positioning system (GPS), and other technologies with modern agricultural technologies to improve crop yield and quality, reduce production costs, reduce pollution that is caused by agricultural activities and improve environmental quality. Precision agriculture is an effective way to achieve sustainable development of agriculture with high quality, high yield, low consumption, and environmental protection. Low-altitude remote sensing has the characteristics of low operating cost, high flexibility, and real-time and rapid data acquisition. It has unique advantages in the application field of crop disease detection, so it has become the key research direction of modern precision agriculture.

Tomato is one of the important cash crops, as the world’s main vegetable crops, its cultivation area and yield are increasing. Additionally, due to our actual planting environment, only the tomato image can be collected. Hence, we selected tomato as the study object to establish the model of crop organ segmentation and leaf diseases diagnosis. Aiming at some problems in the above-mentioned disease identification, a tomato organ segmentation and leaf diseases identification method based on weakly supervised deep neural network is proposed. Moreover, it can be extended to other similar application scenarios of crop leaf diseases identification, while the method has potential application for all crops. When compared with other methods, the method can reach higher recognition accuracy. At the same time, the model takes up less memory space and it can realize real-time recognition of tomato diseases on low-performance terminals. The use of the weakly supervised method can reduce the dependence on accurate labeling of samples, the demand of disease samples, and prepare for the follow-up mid-term disease detection.

2. Materials

2.1. Dataset

Two datasets were used in the experiment: the actual greenhouse live dataset TJ-Tomato and the open dataset PlantVillage. The TJ-Tomato dataset photos were actually taken in the glass greenhouse, which uses a PTZ camera to pre-plan the shooting path to achieve the fixed-point image acquisition. The PlantVillage website (www.plantvillage.org) provides an open database for free access. The two-dimensional image of the crop contains abundant plant phenotypic information, and the image is acquired in the most direct way by analyzing a simple image acquisition device in the greenhouse and real-time correlation information analysis. The greenhouse is located in the Chongming base of the China National Center for Facilities and Agricultural Engineering and Technology Research, while using two types of PTZ camera, Hikvision DS-2DC7520IW-A (five-million pixels) and DS-2DE7220IW- A (two-million pixels), shooting to monitor the growth of the tomato, and obtain 2592*1944 and 1920*1080 resolution images, respectively.

The commonly used data acquisition method for plants phenotyping is the use of common image acquisition devices, such as cameras. In the process of image acquisition, most platforms use
single-independent shooting to obtain image information. This image acquisition method requires manual intervention, which has certain limitations on the automation level and efficiency of image acquisition. Therefore, the TJ-Tomato dataset uses a PTZ camera to pre-plan the shooting path to achieve fixed-point image acquisition. Image acquisition is achieved through existing surveillance cameras, which does not change the greenhouse status as far as possible, without interfering with the actual production and crop growth in the greenhouse, and making full use of existing resources. The two cameras come with a pan/tilt head, which can realize 360° rotation in the horizontal direction and 90° rotation in the vertical direction. It can set the fixed point or scan track according to the collection requirements, and time-lapse cruise shooting to achieve multi-angle plant shooting. According to the shooting distance and position, we initially selected 200 points with different focal lengths to capture the tomato leaves and fruits as the main target, and cruised from south to north according to the route of Figure 1.

![Image of installation site and camera cruise route.](Figure 1.)

PlantVillage is an Internet plant public image library that was launched by the University of Pennsylvania epidemiologist David in 2012 while using machine learning technology. More than 50,000 images of visible light leaves from 14 species of 38 species of labels were collected. Among them, there were 10 types of 18,160 tomato leaves, which were healthy leaves and nine leaf diseases. We used them as the basic datasets of crop diseases. Figure 2 shows an example of various class pictures for this dataset.

![Examples of tomato disease leaf: healthy, Tomato bacterial spot (TBS), Tomato early blight (TEB), Tomato late blight (TLB), Tomato leaf mold (TLM), Tomato mosaic virus (TMV), Tomato septoria leaf spot (TSLS), Tomato target spot (TTS), Tomato two-spotted spider mite (TTSSM), and Tomato yellow leaf curl virus (TYLCV) respectively.](Figure 2.)

**Figure 2.** Examples of tomato disease leaf: healthy, Tomato bacterial spot (TBS), Tomato early blight (TEB), Tomato late blight (TLB), Tomato leaf mold (TLM), Tomato mosaic virus (TMV), Tomato septoria leaf spot (TSLS), Tomato target spot (TTS), Tomato two-spotted spider mite (TTSSM), and Tomato yellow leaf curl virus (TYLCV) respectively.

### 2.2. Preprocessing

For the TJ-Tomato dataset, first select some images for label of the bounding box. It was difficult to select the complete blade due to the overlap of crop leaves and leaves in the image. It did not make
sense to label all leaves or fruits. Hence, we only marked more obvious leaves, which reduced the amount of mark work and increased the accuracy at the same time. Figure 3 shows the preprocessing result. In the figure, green box represents the target bounding box after preprocessing.

![Preprocessing result](image)

**Figure 3.** Preprocessing result. Green box represents the target bounding box.

The number of samples in each category of PlantVillage dataset was uneven and greatly varies. Additionally, there were too many or too few samples of a certain category. The data expansion strategy was used to expand the original dataset and balance the numbers among the samples of each category in order to prevent over-fitting, enhance the robustness and reliability of the model, and improve the versatility of the classification model. Seven common types of data expansion methods were used, including image horizontal mirroring, vertical mirroring, diagonal mirroring, horizontal-vertical mirroring, diagonal-horizontal mirroring, diagonal-vertical mirroring, and diagonal-horizontal-vertical mirroring. Choose different expansion methods based on the number of samples in the original category. For example, horizontal mirroring has expanded the original 1591 pictures of “healthy” to obtain 3182 pictures; the original 373 pictures of TMS have obtained 2984 pictures by the above seven expansion methods.

A total of 37,509 expanded images were used as the final dataset of tomato disease samples. Randomly selected 80% images of each category as training set, 10% of the images were used as the validation set, and 10% were used as test set. Table 1 shows the number of original and expanded images of the dataset, and Figure 4 shows an example of expanded TMV disease.

| Classes     | Images Number | Expand Number |
|-------------|---------------|---------------|
| healthy     | 1591          | 3182          |
| TBS         | 2127          | 4254          |
| TEB         | 1000          | 3000          |
| TLB         | 1909          | 3818          |
| TLM         | 952           | 3808          |
| TMV         | 373           | 2984          |
| TSLS        | 1771          | 3542          |
| TTS         | 1404          | 4212          |
| TTSSM       | 1676          | 3352          |
| TYLCV       | 5357          | 5357          |
| Total       | 18,160        | 37,509        |
3. Existed Difficulties

From above illustration, the actual greenhouse environment is much more complex than the laboratory study for the specific object of tomato, as shown in Figure 5. It mainly includes the following aspects:

(1) There were many kinds of backgrounds around the plants. When compared with tomato plants in the laboratory environment or open-air field environment, there were many artificial introductions besides soil, ground, and other factors, such as culture tanks (Figure 5a), ground water pipes (Figure 5b), etc.

(2) The light environment was more complicated. The glass structure of the greenhouse, the film on the ground, the sun visor, and the fill plate all exacerbated the refraction and reflection of light. The presence of a fill light also caused the color characteristics of the plant to be altered. At the same time, the planting density of the plants in the greenhouse was large, and the various organs, such as the leaf fruits of the plants, overlap and interlace, which caused many shaded areas on the image. As shown in Figure 5c, the leaves in the dark area and the bright area have obvious differences in characteristics.

(3) For the images that were collected by tomato in different growth cycles, the plant morphology was significantly different. When reaching the blossom and fruit period, the height of the plant significantly changed, the number of leaves increased, and the coverage of each other grew. As shown in Figure 5d, as the plant grows and ages, the leaves will gradually distort and wither, and some surfaces will also produce a large number of lesions and insect stings.

(4) While considering the current situation, such as cost and network transmission speed, the surveillance camera’s imaging cannot match the high-definition camera, especially in terms of resolution and focus; therefore, the overall image is in sub-optimal pixels. As a result, the characteristics of the fruit and leaves became more illegible. On the other hand, as the camera moved, the captured picture would produce a series of problems such as out-of-focus, image edge distortion and so on, as shown in Figure 5a.
Mask R-CNN [16] is a CNN based on the Faster R-CNN [17] architecture, which represents the pixel in an image is assigned a category ID according to the object to which it belongs, and it is more easily implemented and trained, and it can be easily applied to other fields, such as target detection and segmentation. Most instance segmentation algorithms require a split mask label that is to be assigned to all training samples. Labeling new categories is a time-consuming task. By contrast, box labels are very numerous and well collected. This raises the question: Can we train high-quality instance segmentation models for categories that do not have full instance segmentation? To this end, the Grabcut [13] algorithm is a graph-based image segmentation method. Firstly, a Gibbs energy function is defined, and then solve the min-cut of this function. This min-cut is the set of segmented pixels of the foreground and background. After selected the area in the box, the area outside the box is regarded as background, and the area within is regarded as the possible foreground area. By calculating the Gaussian mixture model (GMM) of the foreground and background, substituting the r, g, b value of each pixel into a single Gaussian model, the model with the largest value is selected as the attribution of the pixel, a graph is created, and the min-cut is solved for the graph, the loop is looped until convergence, thereby judging the foreground region and the background region in the marquee.

4. Related Work

Semantic segmentation: Semantic segmentation is a typical computer vision problem. Each pixel in an image is assigned a category ID according to the object to which it belongs, and it is more commonly used in autonomous driving [10,11], human-computer interaction [12], and so on. The Grabcut [13] algorithm is a graph-based image segmentation method. Firstly, a Gibbs energy function is defined, and then solve the min-cut of this function. This min-cut is the set of segmented pixels of the foreground and background. After selected the area in the box, the area outside the box is regarded as background, and the area within is regarded as the possible foreground area. By calculating the Gaussian mixture model (GMM) of the foreground and background, substituting the r, g, b value of each pixel into a single Gaussian model, the model with the largest value is selected as the attribution of the pixel, a graph is created, and the min-cut is solved for the graph, the loop is looped until convergence, thereby judging the foreground region and the background region in the marquee.

Figure 5. Existed difficulties in greenhouse image. (a) Image fuzzy and wrapping paper and words. (b) Ground water pipes. (c) Uneven brightness and shading. (d) Different plant morphology with different growth cycles.
segmentation models for categories that do not have full instance segmentation? To this end, we propose a weakly supervised instance segmentation task, which implements instance segmentation on the TJ-Tomato dataset without split mask tags, extending the broad concept of the visual world.

Image classification: An image processing method that separates the different categories of objects according to different characteristics reflected by the target in the image information. CNN have been widely used in image classification and image detection since 2012. Common CNNs include AlexNet [18], VGGNet [19], ResNet [20], Inception [21], etc. In machine learning algorithms, CNN have become the preferred solution for image classification, and their image recognition accuracy is very high, which can be widely used in a variety of applications across platforms. As a deep neural network, the power of CNN is its multi-layer structure that can automatically learn multiple levels of features: the shallower layers have smaller receptive fields and can learn some local area features; The deeper layers have larger receptive fields and they can learn more abstract features that are less sensitive to the size, position, and orientation of the object, thus contributing to improved recognition performance. The CNN can be used for image classification on the one hand, and, as a feature extractor on the other hand, and sent to the next, that is, the the pre-trained CNN model processes the input picture to obtain a convolution feature map.

5. OrganInstance Segmentation and Disease Identification

As mentioned above, it was difficult to directly segment leaves and identify diseases due to the variety of backgrounds around the plant and the complex light environment. A two-step strategy was adopted. The first step was to segment the tomato organs, and detect and segment the leaves and fruit organs of the tomato. The second step was to identify the tomato leaf diseases in order to reduce interference in the background and light environment.

5.1. Organ Instance Segmentation

5.1.1. Far and Near View Picture Classification

We studied the color characteristics of the picture and fitted the distance judgment formula, divided the picture into far view picture and near view picture through analyzing the complexity of the greenhouse environment and the objective quality defects of the camera image. The diversity of the data is largely due to the camera results of different focal lengths. The position of the region of interest in each image varies greatly, and the leaves of different colors and sizes of fruits and morphological features are not the same under the picture. When the focal length is relatively long, the fruit and leaves are usually small, the background occupies most of the image relative to the plant, and the overall characteristics of the plant are prominent. When the focal length is gradually shortened, the proportion of plants is larger and larger, and the texture characteristics of the plant are more obvious. According to this, the obtained pictures can be divided into two categories: far view and near view. Firstly, according to the human sensory recognition standard, a small sample of people is selected to define the distant picture and the close-up picture, and the picture structure features are mined, and the two types of pictures are generally presented. The law is further applied to the determination of general pictures.

It can be seen from the above analysis that the key to judging the distance is the proportion of the picture occupied by the fruit, the leaves, and the background. We used the color features in RGB space to roughly estimate the image since we did not need to accurately segment the three parts in this step. For the blade part, the super green feature is recognized as a relatively efficient means of discrimination. Through the color operator, the original three-dimensional problem was transformed into a one-dimensional problem, and the image was initially rough-classified by simple color feature analysis.

The average ratio of fruit, leaf, and background in the distant image was 8.54%, 41.69%, and 44.17%, respectively, while the average proportion of fruit, leaf, and background in the close-up image...
was 23.04%, 66.38%, and 13.75%, respectively. The ratio of fruit to background significantly varies. That is to say, if the proportion of the fruit and the leaves in the image occupied most of the image and the background ratio is small, we prefer to define it as a close-up. To quantify the decision criterion, define a distance determination formula (DD) [22]:

\[
DD = \alpha \cdot 3\sqrt{P_l} + \beta \cdot \log_2 P_f + \gamma \cdot e^{-0.4 \times P_b}
\]  

(1)

\(P_l, P_f, P_b\) represent the ratio of leaves, fruits, and background. \(\alpha, \beta, \gamma\) are the weight parameters for each category. Select 0.5 as the critical threshold for far and near view determination.

To achieve a better segmentation effect for both the larger and smaller targets, for far view images, because the target is smaller and it occupies fewer pixels, cut a large image into several small ones, and combined them into a image after processing. For the close-up image, since the target occupies more pixels, the resolution is reduced by compression, which thereby reduces the amount of calculation.

5.1.2. Weakly Supervised Instance Segmentation

The instance segmentation algorithm usually requires all of the training samples to be assigned an accurate pixel-level segmentation mask supervised tag. The collection of these tags is very difficult. Labeling new categories is a time-consuming and laborious task. However, the annotations are numerous and well collected. Therefore, a weakly supervised instance segmentation task is proposed to solve this problem. The instance segmentation is implemented by applying Mask R-CNN [16] on the TJ-Tomato dataset without the segmentation mask label.

For the TJ-Tomato dataset, first select some images for the bounding box labeling—labeling the more obvious tomato leaves. The initial segmentation is then implemented while using the algorithm described below. GrabCut [13] uses an iterative optimization method to solve the Gaussian Mixture Model (GMM) step by step. The Gaussian mixture model is a plurality of single Gaussian models, and a Gaussian model can be constructed to reflect the characteristics of the set of pixels. GrabCut uses RGB color space and K Gaussian components (K = 5) to model the target and background. For each pixel, it is either from a certain Gaussian component of the target GMM or from a certain Gaussian component of the background GMM. Table 2 shows the implementation process of the algorithm, details are as follows.

| Table 2. Algorithm implementation process. |
|-------------------------------------------|
| **Step 1** | The rectangle’s external pixels are marked as background and the internal pixels are marked as unknown. |
| **Step 2** | Create an initial split, the unknown pixels are classified as foreground, and the background pixels are classified as background. |
| **Step 3** | Create a GMM for the initial foreground and background |
| **Step 4** | Each pixel in the foreground class is assigned to the most probable Gaussian component in the foreground GMM. The background class does the same thing. |
| **Step 5** | Update the GMM according to the assigned pixel set in the previous step. |
| **Step 6** | Create a graph and execute the Graph cut [23] algorithm to generate a new pixel classification (possible foreground and background) |
| **Step 7** | Repeat steps 4-6 until convergence |

Image segmentation can be seen as a pixel mark problem. The target’s mark is set to 1 and the background mark is set to 0. This process can be minimized by minimizing the cut. Graph cut [23] uses the max flow algorithm to calculate the minimum energy cut edge globally based on the energy formula. If the image is segmented into \(L\), the energy of the image can be expressed as:

\[
E(L) = aR(L) + B(L)
\]  

(2)
R(L) is the regional term and B(L) is the boundary term, and a is the important factor between the region term and the boundary term, which determines their influence on energy. If a is 0, then only the boundary factor is considered, regardless of the regional factor. E(L) represents the weight, which is the loss function, also being called the energy function. The goal of the graph cut is to optimize the energy function to minimize its value. The region term reflects the overall characteristics of the pixel sample set, and the boundary term reflects the difference between the two pixels. Figure 6 shows the preliminary segmentation results. Figure 6a shows original images and Figure 6b shows the bounding box annotation, Figure 6c shows the preliminary segmentation.

![Preliminary segmentation results](image)

**Figure 6.** Preliminary segmentation results. (a) Original image, (b) Bounding box annotation, and (c) Preliminary segmentation.

The Facebook AI research team has made a number of contributions on the path of deep learning, such as R-CNN [24] and Fast R-CNN [25]. In 2016, Microsoft Research proposed Faster R-CNN [17], which reduced the amount of computation on the border search and further improved the speed of the algorithm. In 2017, the Facebook AI research team proposed Mask R-CNN [16] again to enhance the performance of Faster R-CNN on the border recognition by adding an object mask branch parallel to the existing branch. Mask R-CNN is used for target instance segmentation. In simple terms, target instance segmentation is basically object detection, but instead of using a bounding box, its task is to provide an accurate segmentation of the object. Figure 7 shows the weakly supervised organ segmentation results. The loss function of Mask R-CNN is:

\[ L = L_{cls} + L_{box} + L_{mask} \]  

(3)
Figure 7. Weakly supervised organ segmentation results. Left: Original; Right: Segmentation results.

5.2. Disease Identification Model Structure

CNNs can extract features of different levels of images. As the number of network layers increases, the extracted features are more abundant, and the ability to express semantic information is stronger. If you simply increase the number of network layers, it will make the gradient disappear or the gradient explode. He [20] et al. of Microsoft Research Institute proposed a residual neural network, which
allows the network depth to be greatly improved while achieving higher accuracy, in order to solve the problem of degradation caused by the excessively deep network. The main structure of the residual neural network is a stack of multiple residual learning modules. The residual neural network can solve the degradation problem that is caused by the excessively deep network, so the network can get better performance by constructing a deeper network structure. At present, the residual neural network performs well in various recognition tasks and it can obtain high accuracy. Therefore, the residual neural network is used as the infrastructure to identify crop diseases. However, when considering the particularity of crop disease identification, the residual neural network still has some shortcomings:

1. The residual neural network can increase the network depth to hundreds or even deeper through the bottleneck residual module, and achieve better recognition results when the dataset is large. However, the deeper the network, the larger the amount of parameters, which leads to a sharp increase in computing resources, without considering the training and storage issues of the model in practical application.

2. The convolutional layer of the residual neural network adopts the 3*3 convolution kernel for feature extraction, and the 1*1 convolution kernel only plays the role of dimensionality reduction or dimension-up, and the extracted features are relatively single, making the image expression of information not accurate enough.

For the identification of crop diseases, the support of high-performance workstations might be lacking in practical application. The deep network will increase the difficulty of model training, and the model after training has a large memory demand, which is difficult to adapt to the needs of low-cost terminals.

5.2.1. Multi-Scale Residual Learning Module

In the PlantVillage dataset, there were significant differences in the area of the image occupied by the leaves. The single-dimensional convolution kernel was not accurate enough to characterize the diseased leaves. The convolution kernels of different sizes have different receptive fields. Large convolution kernels focused on the extraction of global features, and smaller convolution kernels could extract more local features. Therefore, an improved residual learning module was proposed in order to make the extracted features more abundant, which used a multi-scale convolution kernel instead of a single-scale convolution kernel to construct a residual learning module, so that tomato disease recognition can achieve higher accuracy. At the same time, it can reduce the memory requirements of the model parameters. Studies have shown that the convolution of sparse connections can be approximated by merging multiple sparse matrices into denser sub-matrices. The convolutional layer in the original residual learning module was designed according to the Inception [21,26] structure in order to utilize the multi-scale convolution kernel, and the computational amount required for the 5*5 convolution kernel is relatively large, in order to reduce the amount of parameters and increase the calculation speed. In the actual use process, the 5*5 convolution kernel was replaced by two 3*3 convolution kernels, which allowed for the convolution layer to be extracted to different levels with different receptive fields. The feature made the network more adaptable to the scale of the target in the image, and expanded the width of the network, which can effectively avoid the over-fitting phenomenon caused by the excessively deep network, as shown in Figure 8 Module.

5.2.2. Lightweight Residual Learning Module

The deep neural network model has great challenges in running at low cost terminals due to the size and power consumption of the storage space. Methods, such as model compression and lightweight model design, can be used in order to solve such problems. At present, the commonly used method of terminals is to design a lightweight network architecture, of which MobileNet [27] is one of the mainstream lightweight networks proposed for mobile and embedded devices. It uses depth separable convolutions to build a lightweight deep neural network that decomposes standard volume integration into depthwise convolution and pointwise convolution. Depthwise convolution is
to separately convolve each channel, and the pointwise convolution is used to combine the information of each channel to greatly reduce the parameter and calculation amount. At the lower layer of the network, the standard convolution in the multi-scale residual learning module in Figure 8 is replaced with a depth separable convolution to obtain a lightweight residual learning module, as shown in Figure 9. Where conv represents standard convolution, and conv/dw and conv/pw represent depthwise convolution and pointwise convolution, respectively.

As the number of network layers increases, the receptive field becomes larger, characteristics become more abstract, channels number increases, and the convolution kernel number increases. Therefore, in the deeper layer of the network, the large convolution kernel is removed to reduce the parameters. In addition, the computational complexity can be reduced by the Factorizing Convolutions [28] operation, that is, the n*n volume integral is solved into two one-dimensional convolutions 1*n and n*1, as shown in Figure 10.
As the number of network layers increases, the receptive field becomes larger, characteristics become more abstract, channels number increases, and the convolution kernel number increases.

Therefore, in the deeper layer of the network, the large convolution kernel is removed to reduce the number of parameters. In addition, the computational complexity can be reduced by the Factorizing Convolutions [28] operation, that is, the n*n volume integral is solved into two one-dimensional convolutions 1*n and n*1, as shown in Figure 10.

The use of depthwise convolution has the problem of "unsatisfactory information flow", and the output feature map only contains a part of the input feature map. MobileNet uses pointwise convolution to solve this problem. ShuffleNet [29] uses the same idea to improve the network, which replaces the pointwise convolution with channel shuffle, and the channels of the feature maps of each part are disorderly disordered to form a new feature map. The problem of "unclear information flow" caused by depthwise convolution. MobileNet uses more convolutions, calculations, and parameters are inferior, but the number of nonlinear layers is increased, theoretically more abstract; ShuffleNet eliminates pointwise convolution, while using channel shuffle, which reduces the amount of parameters. Therefore, in the reduction module, the reduction reduction module that is shown in Figure 11 is used instead of the pooling operation commonly used in the CNN to achieve picture size reduction and channel expansion. Where conv/g represents group convolution, divided into four groups, and the convolution kernel size is 1*1.

5.2.3. Reduction Module

Figure 10. Lightweight residual learning module-module2.

Figure 11. Reduction module.

5.2.4. Leaf Disease Identification Model

The improved residual neural network in this paper is mainly composed of four Stages and three Reductions, which consists of the three modules described above. First, the input image passes through three 3*3 standard convolution kernels and one Max Pooling, and then alternates through Reduction and Stage modules, and the size of the feature map through the Reduction module is reduced to half.
The downsampling here is not taken. The pooling operation is replaced by the Reduction module, in which the deep separable convolution and channel shuffle are used instead of the standard convolution. Finally, through the average pooling, Dropout [30] layer, and finally output to the Softmax classifier for classification. Figure 12 shows the overall framework of the improved model.

![Framework of leaf disease recognition model](image)

Among them, Stage1 is made up of three 3x3 convolutions in series, the first convolution stride is 2, Stage2 is composed of two modules a1 and a2 in series, and Stage3 is composed of two modules b1 and b2 in series. The four modules a1, a2, b1, and b2 are the lightweight residual learning modules shown in Figure 9. Stage 4 is composed of two modules c1 and c2 in series, each of which is a multi-scale residual learning module that is shown in Figure 10. Reduction module is the module shown in Figure 11 to replace the commonly used pooling operation to achieve picture size reduction and channel expansion. The Dropout operation randomly removes some of the neurons with a certain probability during the training process, so that the corresponding parameters are not updated when the backpropagation is performed. The addition of the Dropout layer is to suppress the occurrence of over-fitting to a certain extent and improve the generalization ability of the model. Table 3 shows the output dimensions of each module in the model.

| Layer | Input | Stage1 | MaxPool | Reduction1 | Stage2 | Reduction2 |
|-------|-------|--------|---------|------------|--------|------------|
| Output Size | 256 x 256 x 3 | 128 x 128 | 64 x 64 | 32 x 32 x 128 | 32 x 32 x 128 | 16 x 16 x 256 |

5.3. Two-Step Strategy for Crop Organ Segmentation and Disease Identification

Figure 13 shows the two-step strategy for crop organ segmentation and disease identification above-mentioned. It consists of two steps. The first step corresponds to the organ instance segmentation in Section 5.1 above, in Figure 13, marked ① is far and near view picture classification, in order to achieve a better segmentation effect for both the larger and smaller targets, use different processing methods for far view and near images. Subsequently, the image after processing was segmented by the weakly supervised semantic segmentation algorithm that is introduced in Section 5.1.2, marked ② in figure. According to the segmentation results, every single leaf without background was extracted.
from the original image, and the lightweight disease identification method introduced in Section 5.2 was then used to determine the leaf diseases label, which is marked ③ in Figure 13. In segmentation results, different colors show different segments of leaves. Lf is the abbreviation of leaf in the figure, and the number represents the confidence of segmentation results.

Figure 13. Overall framework of two-step strategy.

6. Results and Discussion

6.1. Experimental Environment

The experimental configuration environment of this paper is as follows: Ubuntu16.04 LST 64-bit system, processor Intel Core i5-8400 (2.80GHz), memory is 8 GB, graphics card is GeForce GTX1060 (6G), using Tensorflow-GPU1.4 deep learning framework, use Python programming language.

A total of 37,509 images of the expanded tomato disease sample dataset were randomly divided into training set, verification set, and test set in order to verify the effectiveness of the leaf disease identification model proposed in this paper, of which the training set accounted for about 80% or 29,993 pictures. The verification set accounts for about 10% or 3,750 images, and the test set accounts for about 10% or 3,766 images; they are used to train the model, select the model, and evaluate the performance of the improved model. The training set and the test set are divided into batches by batch training. The batch training method is used to divide the training set and the test set into multiple batches. Each batch trains 32 pictures, that is, the minibatch is set to 32. After training 4096 images, the verification set is used to determine the retained model. After training all the training set images, the test set is tested. Each test batch is set to 32. Iterate through all the pictures in a training set as an iteration (epoch) for a total of 10 iterations. The model was optimized while using the momentum optimization algorithm and the learning rate is set at 0.001.

6.2. Analysis of Segmentation Results

It is difficult to determine the attribution because the fruit organs of crop leaves overlap more. The above example segmentation method cannot segment all of the prospects and it has no meaning. Therefore, through formula (1), the scores were calculated and five fruit distant pictures and five fruit
close-up pictures were selected, and the pixel results of the labeling results and ground truth were compared. The evaluation criteria take the image pixel accuracy (IA) of fruits and leaves into account.

\[
IA = \frac{P_c}{P_a}
\]

where \(P_c\) represents the number of pixels in the ground truth target corresponding to the segmentation result, the segmentation map and the ground truth classification, and \(P_a\) represents the number of all the pixels of the ground truth target corresponding to the segmentation result. As shown in Table 4, the segmentation accuracy of the four segmentation methods on 1–10 pictures is listed in the table.

**Table 4. Image pixel segmentation accuracy.**

|      | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   | 10  |
|------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| FCM  | 0.483 | 0.689 | 0.554 | 0.852 | 0.785 | 0.841 | 0.723 | 0.862 | 0.723 | 0.897 |
| Coseg | 0.858 | 0.798 | 0.891 | 0.673 | 0.735 | 0.564 | 0.745 | 0.884 | 0.885 | 0.913 |
| LDA  | 0.612 | 0.710 | 0.563 | 0.465 | 0.687 | 0.715 | 0.543 | 0.737 | 0.674 | 0.687 |
| Proposed | 0.963 | 0.961 | 0.975 | 0.925 | 0.892 | 0.963 | 0.917 | 0.905 | 0.937 | 0.954 |

6.3. Analysis of Disease Identification Results

In the experiment, the classification accuracy rate is used as one of the evaluation criteria of the experimental results. The classification accuracy rate is defined as the ratio of the number of diseased leaves correctly classified in the verification set to the total number of diseased leaves. The higher the classification accuracy, the higher the degree of classification. Consequently, the better the performance of the model. In addition, the algorithm focuses on the storage and calculation of models on low-cost terminals, so the number of model parameters and the detection speed are also used as model evaluation criteria.

6.3.1. Comparison of Different Depth Model Identification Indicators

Compare the improved lightweight model with several more advanced CNNs, including VGG16/19, ResNet-18/34/50/152, Inception V4, Inception-ResNet-V1/V2, MobileNet-V1/V2, and Res2Net50, which use these networks to diagnose and identify tomato diseased leaves. Table 5 lists the classification accuracy of tomato disease leaves and the size of the model after training under different neural network models.

**Table 5. Comparison of different depth model recognition indexes.**

| Models               | Accuracy(%) | GFlops | Fps(Images/sec) | Model Loading Time (s) |
|----------------------|-------------|--------|-----------------|------------------------|
| VGG16 [19]           | 94.65       | 35.82  | 73              | 1.81                   |
| VGG19 [19]           | 95.19       | 46.69  | 64              | 2.15                   |
| ResNet-18 [20]       | 97.2        | 6.98   | 368             | 0.27                   |
| ResNet-50 [20]       | 96.95       | 14.06  | 119             | 0.85                   |
| Inception V4 [34]    | 97.35       | 33.91  | 90              | 2.11                   |
| Inception-ResNet V2 [34] | 98.24  | 25.29  | 115             | 1.64                   |
| MobileNet-V1 [27]    | 96.52       | 1.49   | 291             | 0.59                   |
| MobileNet-V2 [35]    | 95.14       | 0.96   | 229             | 0.74                   |
| Res2net-50 [36]      | 97.26       | 9.53   | 112             | 1.88                   |
| Proposed             | 98.61       | 2.80   | 276             | 1.07                   |

It can be seen from Table 5 that, on the tomato disease leaf dataset, the improved residual neural network model can achieve 98.61% accuracy. Compared with the traditional convolutional network model, the proposed network model has higher accuracy than the comparison model, which shows the effectiveness of using multi-scale convolution in the residual module to improve the network performance. Moreover, the number of parameters in the model is significantly reduced and the
FLOPS of the model after training only accounts for 2.80G, which greatly reduces the amount of computation in the memory footprint. When compared with the lightweight networks MobileNet-V1 and MobileNet-V2, the improved residual neural network model has a slightly higher memory requirement, but the accuracy is improved. In the process of crop disease identification, in addition to the accuracy rate, speed is also an important indicator of the evaluation model. The fps listed in the table is the number of pictures detected per second based on the recognition of 3766 test pictures. The test results show that the detection speed is also in the forefront. Under the comprehensive consideration, the improved model still has certain advantages in performance. Figure 14 shows some practical examples of disease detection with failures.

Figure 14. Practical examples of diseases diagnosis with failures. Real tomato disease: (a) TBS, (b) TBS, (c) TLB, (d) TLM, (e) TLS, (f) TTS, (g) TTS, (h) TTS, (i) TTSSM, (j) TTSSM, (k) TTSSM, (l) TYLCV, respectively. Wrong diagnosis: (a) TTS, (b) TYLCV, (c) TEB, (d) TLS, (e) TEB, (f) TBS, (g) TMV, (h) TTSSM, (i) healthy, (j) TTS, (k) TTS, (l) TTSSM, respectively.

6.3.2. Influence of the Number of Layers on the Model

Reduction4 and Stage5 are added between Stage4 and Average pooling, so that the network structure becomes five stages, as shown in Figure 15, Reduction4 is the module shown in Figure 11, Stage 5 is composed of two modules d1 and d2 in series, each of which is a multi-scale residual learning module shown in Figure 10 in order to test the influence of increasing the model stage on the recognition performance, based on the overall framework of the leaf disease recognition model shown in Figure 12. The output of each layer of the added network is shown in Table 6. Experiments were carried out on the tomato disease leaf dataset. The results are shown in Table 7. The recognition result of the leaf disease model that is shown in Figure 12 is represented by Proposed, and the recognition result of the model shown in Figure 15 is represented by Proposed-S5. The results show that increasing the stage of the network can improve the accuracy of recognition, but it only increases by 0.11%, while the detection speed decreases more, the corresponding fps index drops by 32.2%; the calculation amount increases greatly, corresponding to the increase of 33.6% for the Flops indicator. Therefore, in the low-cost terminal, the use of a four stages network is more advantageous. The accuracy does not decrease significantly, while it greatly reduces the computation and improves the detection speed.

| Layer     | Output Size | Stage1   | MaxPool | Reduction1 | Stage2   | Reduction2 | Stage3   |
|-----------|-------------|----------|---------|------------|----------|------------|----------|
| Output Size | 256 × 256 × 3 | 128 × 128 × 64 | 64 × 64 × 64 | 32 × 32 × 128 | 32 × 32 × 128 | 16 × 16 × 256 | 16 × 16 × 256 |
| Layer     | Reduction3  | Stage4   | Reduction4 | Stage5     | AvgPool  | FC         | Softmax  |
| Output Size | 8 × 8 × 512 | 8 × 8 × 512 | 4 × 4 × 1024 | 4 × 4 × 1024 | 1024     | 10         | 10       |
Table 7. Comparison of different depth model identification indicators.

| Models        | Accuracy(%) | GFlops | Fps(Images/sec) | Model Loading Time |
|---------------|-------------|--------|-----------------|-------------------|
| Proposed-S5   | 98.72       | 3.74   | 187             | 1.12              |
| Proposed     | 98.61       | 2.80   | 276             | 1.07              |

Figure 15. Framework of five-stage leaf diseases recognition model.

Figure 16 shows some practical examples of leaf segmentation and diseases detection with successes. The first step is crop organ semantic segmentation, like Section 5.1, the middle column with the color mask is the output. The second step is the lightweight disease identification method that is mentioned in Section 5.2, where we adopted four-stage network model shown in Figure 12.

Figure 16. Practical examples of leaf segmentation and diseases detection with successes.

7. Conclusions and Future Work

Aiming at the shortcomings of DCNN in crop disease identification, this paper proposed a two-step strategy, including crop organ segmentation, based on weakly supervised deep neural network and
disease identification method using the lightweight model. In the segmentation of crop organs, the weakly supervised method is applied in a wider range, and precise mask labeling is not required. Only the bounding box is required to reduce the dependency on the pixel-level labeling of the sample. Moreover, this paper designed a lightweight disease identification network to reduce the memory and storage space requirements. It used multi-scale convolution to expand the network width, which makes the extracted features more abundant, and thus adapting to the needs of low-cost terminals with use deep separable convolution to reduce the model parameters; it can be extended to other similar application scenarios for crop disease identification.

The identification of crop diseases can be divided into three stages: in the late, middle, and early stage. The late stage refers to the diagnosis after the occurrence of the determined symptoms, at this time, the symptoms of the disease are obvious. The middle stage refers to the disease that is more likely to occur when the crop has a certain symptom, at this time, the warning effect is greater than other stages. The symptoms of diagnosis in early stage are not obvious, it is difficult to determine whether to use visual observation or computer interpretation, but the research significance and demand of the pre-diagnosis is greater, which is more conducive to the prevention of crops, preventing the spread of diseases. With the continuous improvement of UAV and sensor technology, the continuous development of image analysis and processing technologies and algorithms, the monitoring methods of crop diseases that are based on image processing will continue to be applied to practical applications. In the future, it will mainly be carried out from the following two aspects:

(1) Carry out research on crop disease identification based on UAV, and combine existing image processing techniques to apply crop disease identification algorithms to UAV to collect images. When combined with UAV positioning technology to determine the location of the diseased crops, manual or robotic methods can be used to directly remove the diseased plants and reduce the impact on other crops. In the future, it is necessary to transition from a simple test environment to a practical application of comprehensive consideration of crop growth patterns and environmental factors.

(2) Conduct a mid stage disease detection study. The use of weakly supervised method can reduce the dependence on accurate labeling of samples and reduce the demand for disease samples. As some symptoms of crops may be associated with symptoms in the mid-term of the disease, they may not necessarily occur in the future. At this time, accurate disease labeling cannot be achieved, and the training samples are relatively small. Therefore, weakly supervision and light weight ideas can be applied to mid-term crop disease identification of crops. The imaging characteristics of different growth cycles, the study of medium-term forecasting and diagnostic models, the establishment of early warning mechanisms, early diagnosis, and early prevention will probably promote the research progress of early crop diagnosis. In practical applications, the significance will be greater than the identification of diseases in the late stage, which is the focus of the next step.

Author Contributions: All authors provided ideas of the proposed method and amended the manuscript; Y.W. designed the experiments and organized the experimental data. L.X. established the guidance for the research idea, authored or reviewed drafts of the paper, approved the final draft.

Funding: This research was funded by Shanghai Science and Technology Innovation Action Plan of Shanghai Municipal Science and Technology Commission (No. 17391900900), National Natural Science Foundation of China (Grant No. 61573258).

Acknowledgments: We wish to thank Fanhuai Shi for his technical support.

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

References

1. Hiary, H.A.; Ahmad, S.B.; Reyalat, M.; ALRahamneh, Z. Fast and Accurate Detection and Classification of Plant Diseases. *Int. J. Comput. Appl.* 2011, 17, 31–38.
2. Scharr, H.; Minervini, M.; French, A.P.; Klukas, C.; Kramer, D.M.; Liu, X.; Luengo, I.; Pape, J.; Polder, G.; Vukadinovic, D.; et al. Leaf segmentation in plant phenotyping: A collation study. *Mach. Vis. Appl.* 2016, 27, 585–606. [CrossRef]
3. Zhou, J.; Fu, X.; Zhou, S.; Zhou, J.; Ye, H.; Nguyen, H.T. Automated segmentation of soybean plants from 3D point cloud using machine learning. *Comput. Electron. Agric.* 2019, 162, 143–153. [CrossRef]

4. Ma, J.; Du, K.; Zhang, L.; Zheng, F.; Chu, J.; Sun, Z. A segmentation method for greenhouse vegetable foliar disease spots images using color information and region growing. *Comput. Electron. Agric.* 2017, 142, 110–117. [CrossRef]

5. Dong, P.; Wang, X. Recognition of Greenhouse Cucumber Disease Based on Image Processing Technology. *Open J. Appl. Sci.* 2013, 3, 299–315. [CrossRef]

6. Kawasaki, Y.; Uga, H.; Kagiwada, S.; Iyatomi, H. Basic Study of Automated Diagnosis of Viral Plant Diseases Using Convolutional Neural Networks; Springer International Publishing: Cham, Switzerland, 2015; pp. 638–645.

7. Amara, J.; Bouaziz, B.; Algergawy, A. A Deep Learning-based Approach for Banana Leaf Diseases Classification. *BTW (Workshops)* 2017, 266, 79–88.

8. Brahimi, M.; Boukhalfa, K.; Moussaoui, A. Deep learning for tomato diseases: Classification and symptoms visualization. *Appl. Artif. Intell.* 2017, 31, 299–315. [CrossRef]

9. Sun, J.; Tan, W.; Mao, H.; Wu, X.; Chen, Y.; Wang, L. Recognition of multiple plant leaf diseases based on improved convolutional neural network. *Trans. Chin. Soc. Agric. Eng.* 2017, 33, 209–215.

10. Geiger, A.; Lenz, P.; Urtasun, R. Are we ready for autonomous driving? the kitti vision benchmark suite. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 3354–3361.

11. Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; Schiele, B. The cityscapes dataset for semantic urban scene understanding. *arXiv* 2016, arXiv:1604.01685.

12. Oberweger, M.; Wohlhart, P.; Lepetit, V. Hands deep in deep learning for hand pose estimation. *arXiv* 2015, arXiv:1502.06807.

13. Rother, C.; Kolmogorov, V.; Blake, A. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph. (TOG)* 2004, 23, 309–314. [CrossRef]

14. Hariharan, B.; Arbeláez, P.; Girshick, R.; Malik, J. *Simultaneous Detection and Segmentation*; Springer International Publishing: Cham, Switzerland, 2014; pp. 297–312.

15. Chen, X.; Girshick, R.; He, K.; Dollár, P. TensorMask: A Foundation for Dense Object Segmentation. *arXiv* 2019, arXiv:1903.12174.

16. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. *arXiv* 2017, arXiv:1706.03762.

17. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *arXiv* 2015, arXiv:1506.01497. [CrossRef] [PubMed]

18. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* 2012, 60, 1097–1105. [CrossRef]

19. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* 2014, arXiv:1409.1556.

20. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. *arXiv* 2016, arXiv:1512.03385.

21. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. *arXiv* 2015, arXiv:1409.4842.

22. Cao, Q.; Xu, L. Unsupervised Greenhouse Tomato Plant Segmentation Based on Self-Adaptive Iterative Latent Dirichlet Allocation from Surveillance Camera. *Agronomy* 2019, 9, 91. [CrossRef]

23. Boykov, Y.Y.; Jolly, M.P. Interactive graph cuts for optimal boundary & region segmentation of objects in ND images. In Proceedings of the eighth IEEE international conference on computer vision. ICCV 2001, Vancouver, BC, Canada, 7–14 July 2001; pp. 105–112.

24. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. *arXiv* 2014, arXiv:1311.2524.

25. Girshick, R. Fast r-cnn. *arXiv* 2015, arXiv:1504.08083.

26. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv* 2015, arXiv:1502.03167.

27. Howard, A.G.; Zhu, M.; Chen, B. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv* 2017, arXiv:1704.04861.

28. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the Inception Architecture for Computer Vision. *arXiv* 2016, arXiv:1512.00567.
29. Zhang, X.; Zhou, X.; Lin, M.; Sun, J. Shufflenet: An extremely efficient convolutional neural network for mobile devices. *arXiv* 2018, arXiv:1707.01083.

30. Hinton, G.E.; Srivastava, N.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv* 2012, arXiv:1207.0580.

31. Zhu, A.; Yang, L. An improved FCM algorithm for ripe fruit image segmentation. In Proceedings of the 2013 IEEE International Conference on Information and Automation (ICIA), Yinchuan, China, 26–28 August 2013; pp. 436–441.

32. Li, H.; Meng, F.; Wu, Q.; Luo, B. Unsupervised multiclass region cosegmentation via ensemble clustering and energy minimization. *IEEE Trans. Circuits Syst. Video Technol.* 2013, 24, 789–801.

33. Blei, D.M.; Ng, A.Y.; Jordan, M.I. Latent dirichlet allocation. *J. Mach. Learn. Res.* 2003, 3, 993–1022.

34. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A. Inception-v4, inception-resnet and the impact of residual connections on learning. *arXiv* 2017, arXiv:1602.07261.

35. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, C. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520.

36. Gao, S.H.; Cheng, M.M.; Zhao, K.; Zhang, X.Y.; Yang, M.H.; Torr, P. Res2Net: A New Multi-scale Backbone Architecture. *arXiv* 2019, arXiv:1904.01169. [CrossRef] [PubMed]