IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON GEOMETRIC ALGEBRA IN SIGNAL PROCESSING

Received August 13, 2020, accepted August 27, 2020, date of publication September 4, 2020, date of current version September 21, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3021739

Semantic Segmentation of Litchi Branches Using DeepLabV3+ Model

HONGXING PENG[1], CHAO XUE[1], YUANYUAN SHAO[2], KEYIN CHEN[3], JUNTAO XIONG[1], ZHIHUA XIE[1], AND LIUHONG ZHANG[1]

[1]College of Mathematics and Informatics, South China Agricultural University, Guangzhou 510642, China
[2]College of Mechanical and Electronic Engineering, Shandong Agricultural University, Tai'an 271018, China
[3]School of Electronic and Information Engineering, Jiaying University, Meizhou 514015, China

Corresponding authors: Yuanyuan Shao (syy007@sdau.edu.cn) and Juntao Xiong (xiongjt2340@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61863011, Grant 31701325, Grant 31571568, and Grant 31570180; in part by the National Key Research and Development Program of China under Grant 2018YFD020030106; in part by the Meizhou Applied Science and Technology Special Fund Project under Grant 190920104640234; in part by the Agricultural Science and Technology Commissioner Project of Guangdong Province under Grant 40 SCAU; in part by the Natural Science Foundation of Guangdong Province under Grant 2018A030313330; and in part by the Science and Technology Program of Chengdu under Grant 2018YF0501197SN.

ABSTRACT Litchi is often harvested by clamping and cutting the branches, which are small and can easily be damaged by the picking robot. Therefore, the detection of litchi branches is particularly significant. In this article, an fully convolutional neural network-based semantic segmentation algorithm is proposed to semantically segment the litchi branches. First, the DeepLabV3+ semantic segmentation model is combined with the Xception depth separable convolution feature. Second, transfer learning and data enhancement are used to accelerate the convergence and improve the robustness of the model. Third, a coding and a decoding structure are adopted to reduce the number of network parameters. The decoding structure uses upsampling and the shallow features to fuse, and the same weight is assigned to ensure that the shallow feature semantics and the deep feature semantics are evenly distributed. Fourth, using atrous spatial pyramid pooling, we can better extract the semantic pixel position information without increasing the number of weight parameters. Finally, different sizes of hole convolution are used to ensure the prediction accuracy of small targets. Experiment results demonstrated that the DeepLabV3+ model using the Xception_65 feature extraction network obtained the best results, achieving a mean intersection over union (MIoU) of 0.765, which is 0.144 higher than the MIoU of 0.621 of the original DeepLabV3+ model. Meanwhile, the DeepLabV3+ model using the Xception_65 network has greater robustness, far exceeding the PSPNet_101 and ICNet in detection accuracy. The aforementioned results indicated that the proposed model produced better detection results. It can provide powerful technical support for the gripper picking robot to find fruit branches and provide a new solution for the problem of aim detection and recognition in agricultural automation.

INDEX TERMS Semantic segmentation, DeepLabV3+, litchi branches, picking robot.

I. INTRODUCTION

With the increasing industrialization of social industrial structures, the number of people engaged in agricultural production has been decreasing and the automation and mechanization of agriculture will become its main production methods in the future. As a subtropical fruit, litchi has a very short maturity period, and the weather is hot and rainy in southern China. If this fruit cannot be harvested on time, production will suffer, causing serious economic losses. A litchi-picking robot can effectively solve the problems of labor shortage and large-scale planting, which can significantly reduce the production cost of litchi and alleviate the decrease in productivity caused by the loss of agricultural population.

The automatic picking methods used for apples and guava cannot be used for litchi owing to the complexity of its shape, color, and growing environment. Because litchi grows in clusters with a large number of fruits, the branch is not obvious. The ideal picking method needs to detect the litchi branches

The associate editor coordinating the review of this manuscript and approving it for publication was Zhihai He.

and make the robot hold and cut them to pick the fruits. Therefore, the detection of litchi branches is an important part of realizing the automatic picking of litchi fruits and cannot use the automatic picking of apples and guava to apply to litchi picking. This study used a deep learning algorithm to semantically segment the litchi branches for nondestructive picking.

II. RELATED WORKS

At this stage, there are already many research studies in the field of fruit recognition [1]. The traditional target detection algorithm is more suitable for the situation with obvious characteristics and simple background. For litchi detection in natural environment, the background is complex and changeable, it's difficult to extract features with traditional detection algorithm. However, deep learning can use a huge data set to complete model training, extract the rich features of the same target to complete the detection, make the algorithm more robust and generalized, and easier to apply to actual scene. Tao and Zhou [2] proposed a method for apple recognition using point cloud data to improve the recognition ability and perception ability of robots in a three-dimensional (3D) space. Using point cloud information to extract color features and 3D geometric features, their proposed method uses the support vector machine classifier of the genetic algorithm to classify apples, branches, and leaves. The experiment result of Tao and Zhou showed that the recognition accuracy for apples and fruit branches was 92.3% and 88.03%, respectively, and the leaf segmentation accuracy was 80.34%, indicating that their proposed method has high recognition accuracy and performance. Wei *et al.* proposed an improved Otsu threshold algorithm using new features in the Ohta color space to cope with the problem of targeting fruits in complex agricultural settings. Zhuang *et al.* [4] proposed a mature citrus detection method based on a monocular vision system. The block-based local homomorphic filtering algorithm used by their method ensures that only the local blocks identified as having a nonuniform illumination distribution are filtered and that the RG components are adaptively enhanced. Chromaticity mapping is used for better threshold segmentation.

The introduction of deep learning provides a new way for segmentation algorithms to perform their task. Tian *et al.* [5] used the improved YOLO-V3 model [6] to identify the different growth cycles of apples to assess the fruit growth. Sa *et al.* [7] proposed a novel multimodal information fusion faster R-convolutional neural network (-CNN) model [8] using color (RGB) images and near-infrared image information, which improved the F_1 value of sweet pepper detection from 0.807 to 0.838. Bargoti and Underwood [9] proposed an image processing framework for fruit detection and counting that uses feature learning algorithms including multiscale multilayer perceptron and CNN to detect and count apples. The effect of the F_1 value reached 0.861.

In recent years, image semantic segmentation has become a hotspot in the field of deep learning. Zheng *et al.* [28] proposed that CRF be fully modeled into CNN, so that the network can be trained end-to-end with the usual back propagation algorithm, avoiding the post-processing methods used for target rendering. However, the existing target recognition methods have the following problems in processing pixel-level classification. First, the large interesting field of CNN causes the pixel classification output to be coarse and Max Pooling layers reduce the possibility of fine segmentation, resulting in non-acute angle boundaries and blob-like shapes. Second, for similar pixels and pixels with consistent space and appearance, CNN lacks the smooth constraint that motivates them to output the same category, resulting in inaccurate segmentation. Liu *et al.* [29] proposed Markov Random Fields(MRFs) and Conditional Random Fields(CRFs) which could solve above problems. MRF and CRF can be used as a post-processing method to refine the results of other models. Szegedy *et al.* [11] proposed the Inception CNN architecture and launched a 22-layer deep neural network named GoogLeNet in ILSVRC 2014. An error rate of 6.67% for the top 5 score was obtained in the classification challenge, and 43.9% of the mean average precision (mAP) was obtained in the detection challenge. Subsequently, the Google team launched Batch Normalization to launch BN-GoogLeNet, which solved the problem of gradient disappearance and slow convergence, and improved the training speed and classification effect. In the same year, the Google team launched InceptionV3 [13], which was proposed to solve the large volume integral in a small convolution, which significantly reduced the convolution kernel parameters and calculations.

Deep learning in the field of semantic segmentation originated from fully convolutional networks (FCNs) (Long *et al.* [30]), which promoted the original CNN structure, using up-convolution for upsampling, and which could be used without a fully connected layer. Intensive predictions are made to achieve pixel-level classification. SegNet [17] moved the maximum pooling index to the decoder, which improves the segmentation resolution. The DeepLab architecture [18], which mainly uses hole convolution, proposes a cavity pooling of the pyramid model in the spatial dimension, using a fully connected Conditional Random Field (CRF). DeepLabV2 [19], [20] has an encoder with a well-designed decoder module that uses a fully connected CRF, and it proposes that the hole convolution pool maintains the same receptive field without increasing the parameters. DeepLabV3 [19], [20] improved DeepLabV2 in terms of reducing the feature resolution, multiscale objects, and translation invariance in deep convolution, using residual network models for feature extraction. In the PASCAL Visual Object Classes (VOC) Challenge 2012, it achieved an MIoU of 86.9.

Although deep learning and agricultural picking continue to develop, robots that use pick-type end effectors for picking are not widely used at present. Traditional segmentation algorithms are difficult to use and are inaccurate for segmentation of branches in a wild environment. With the development of modern GPU parallel computing and deep learning, it is possible to obtain semantic information in

images through complex algorithms. Therefore, this article proposes an improved DeepLabV3+ semantic segmentation model to segment growing branches because of their small sizes and fragility, takes litchi branches as the research object, uses images semantic segmentation technology to segment the litchi branches images, accomplish the expected goal and achieve segmentation results. Pixel-level semantic segmentation can extract semantic prediction semantics from irregular targets, and then make semantic predictions on the targets. The litchi branches after semantic segmentation can be detected easier, which can provide the basis of early operation for location of litchi picking points.

III. IMAGE DATA PREPROCESSING

A. IMAGE DATA ACQUISITION

The collection dates of the litchi images used in the experiment were June 29, 2018 (sunny), July 8, 2018 (cloudy to rainy), July 10, 2018 (sunny), and May 30, 2019 (sunny). The collection locations were from the orchards in Guangzhou and Zengcheng, China. We used a Canon EOS 60D camera to capture 5184 × 3450-pixel images, a FinePix F500EXR camera to capture 4608 × 3456-pixel images, and several Huawei phones to capture 3968 × 2976-pixel images. Litchi varieties included Guiwei, Feizixiao, Huaizhi, and Luomichi. Weather conditions included rainy, cloudy, and sunny days, and the picking time was from 0800 to 1700. The sampling data had a large difference, which was convenient for strengthening the robustness and test difficulty of the detection network.

The experimental data were sampled from the field, and 703 samples were randomly selected from the obtained samples for data labeling. We randomly selected 1609 samples with data enhancement as the training set and 500 samples as the test set. The number of samples satisfies the data requirements of pixel-based semantic segmentation.

B. IMAGE DATA AUGMENTATION

Data augmentation, as a method of data preprocessing, plays an important role in deep learning. In general, effective data enhancement can better improve the robustness of the model and obtain stronger generalization ability. The general methods of data enhancement are flipping, rotating, panning, etc. Because the labor cost of the full supervision training is huge, the experiment performed used artificial data enhancement, and each sample was up-and-down and symmetrically mirrored, which obtained three times more data volume and provided data resources for deep learning (see Fig. 1).

C. DATA ANNOTATION

In the experiment, the open source tool LabelMe was used for data supervised training. LabelMe (LabelMe software, Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, USA) is an image labeling software with a graphical interface, which can label polygons, rectangles, circles, polylines, line segments and points. Seven hundred three sheets were marked, and the label format was the JSON format. After the code processing, the JSON format

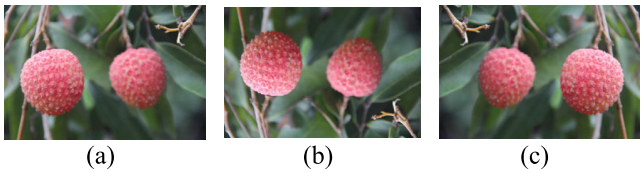


FIGURE 1. (a) Original image, (b) vertical flip, (c) horizontal flip.

was converted into a single-channel image and stored in the PASCAL VOC data format for convenience of usage.

IV. METHODOLOGIES

Image semantic segmentation has been studied for decades and is divided into strong supervision training and weak supervision training under supervised learning (Fig. 2). This article mainly explains the fully supervised training and shows how to achieve the goal of image semantic segmentation through pixel-level classification.

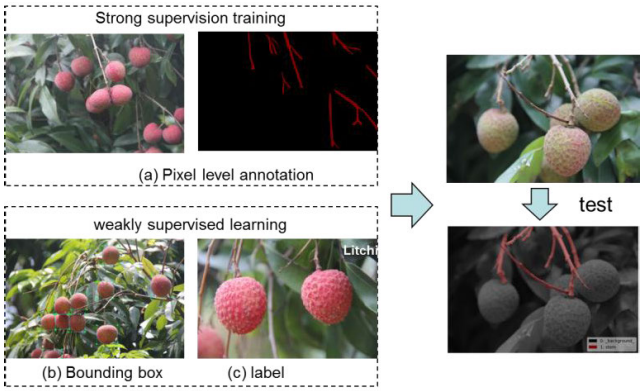


FIGURE 2. Examples of supervised learning.

A. FULLY CONVOLUTIONAL NETWORK

In the field of fully supervised training, Hariharan *et al.* [21] first proposed a deep CNN [22] for semantic segmentation. In the same year, Long *et al.* [30] proposed a FCN for semantic segmentation. As shown in Fig. 3, the network weights are adjusted using feedforward inference and feedback learning, and the fully connected layers used for classification are discarded. The entire network uses convolution operations, obtains depth information by downsampling, and restores the original size by upsampling, to realize the prediction for each pixel.

With the advent of FCNs, a large number of semantic segmentation algorithms based on them have emerged. The experiment in this study used the DeepLab framework to achieve great results in the field of semantic segmentation through the concept of multipath fusion.

B. ATROUS SPATIAL PYRAMID POOLING

To solve the information loss caused by pooling, the DeepLabV3+ model adopts atrous spatial pyramid pooling (ASPP), which can better extract features at different

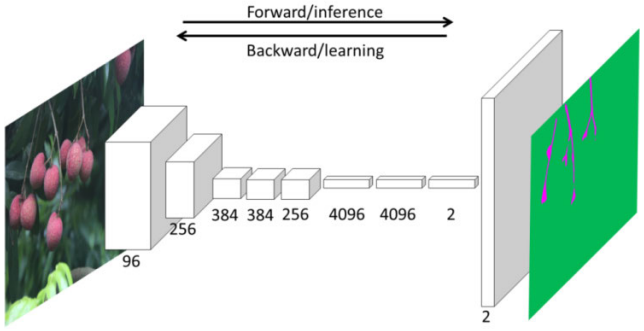


FIGURE 3. Illustration of a fully convolutional network.

resolutions and different feature layers for semantic segmentation. In the case where the receptive field should be unchanged, the number of weight parameters is reduced and the location information loss caused by the mean pooling is solved, as shown in Fig. 4 [18].

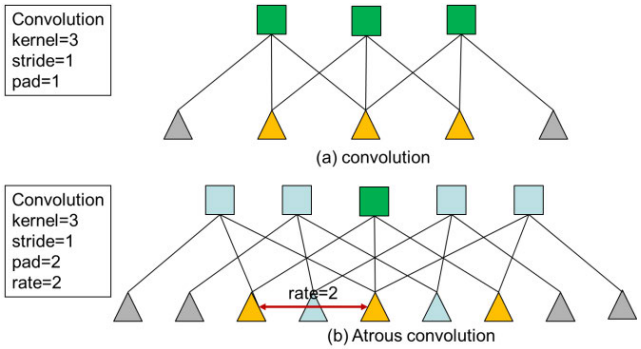


FIGURE 4. One-dimensional view of atrous convolution.

Atrous convolution can expand the receptive field without increasing the volume and the parameters. Atrous convolution, rather than mean pooling, can better obtain the details after the convolution. Refer to (1), w is a filter with a length of k and an input signal of x.

$$y[i] = \sum_{k=1}^{K} x[i + r \bullet k]w[k] \tag{1}$$

The above equation is a downsampling of step 2; the resolution of the image is reduced, and then a convolution operation with a convolution kernel size of 7×7 is performed to obtain a feature map, which is restored to the original resolution by double upsampling. The convolution kernel is used as a 7×7-size hole convolution. The feature map is obtained after a direct convolution. The comparison results showed that the map obtained by the hole convolution is more detailed. Although the hole convolution increases, and the nonzero filter value is considered in the calculation, the actual parameters are not increased and the operation cost is lower, as shown in Fig. 5 [19], [20].

C. CODEC MODEL STRUCTURE

DeepLabV3+ uses a codec structure with shallow features and deep upsampling features. As shown in Fig. 6 [19], [20],

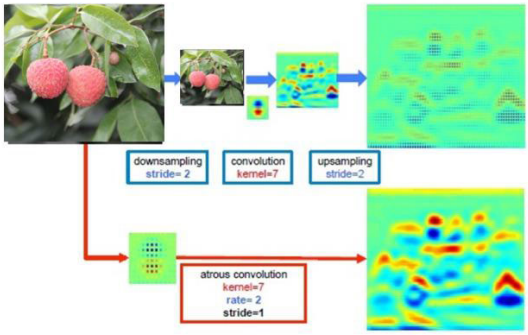


FIGURE 5. Atrous convolution vs. upsampling.

the input image is fed into a deep CNN to obtain a high-resolution abstract feature map with a lower resolution, and different volume convolutions are used to perform the convolution. In deep feature sampling, the obtained high-level feature map is fused with upsampling four times and the shallow features to realize the decoded output.

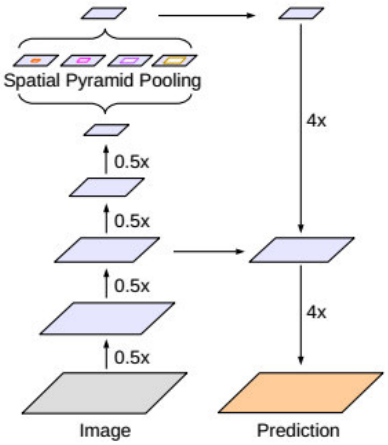


FIGURE 6. Illustration of the atrous convolutional codec structure.

The DeepLabV3+ model is divided into two structures: an encoding and a decoding part. The coding part removes the deep pool of the feature extraction network to keep the high-level abstract information large enough to facilitate the prediction of the pixel location information. Replacing the deep pooling layer with ASPP preserves more details under the same conditions of the receptive field, and the training parameters are not increased, which improves the model prediction performance. Through multiscale information sampling, the target samples are obtained with different amounts of information, which enhances the robustness of the model. The use of a 1×1-size convolution after a multi-scale hole convolution increases the nonlinearity of the coding structure. The decoding part first receives the shallow features and uses the 1×1-size convolution to reduce the number of channels of the feature map, so that the feature map obtained by upsampling four times after the encoding is substantially the same as the number of channels of the feature map, which is beneficial to the learning of the model. The convolutional

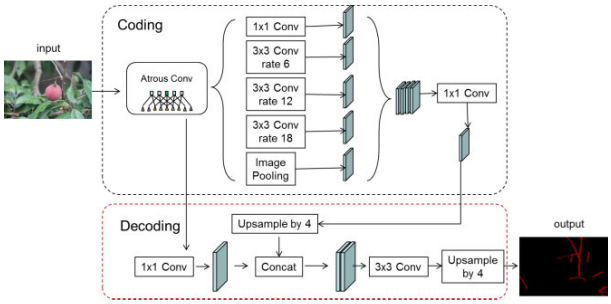


FIGURE 7. Structure of the DeepLabV3+ model.

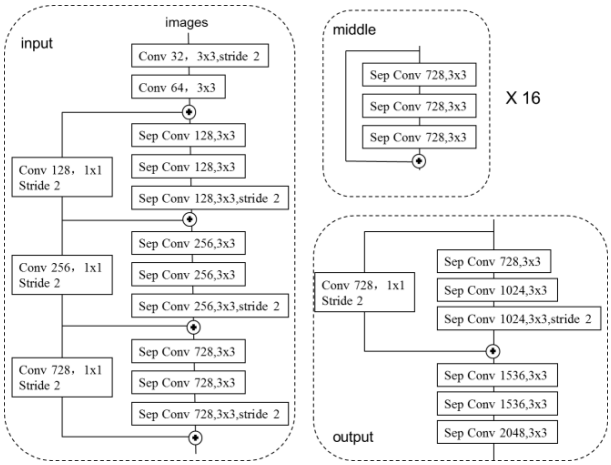


FIGURE 8. Illustration of the Xception network structure. Sep Conv, separable convolution.

shallow features are merged with the upsampled deep features, and the convolution is used to refine the feature details. After upsampling four times, the same resolution as that of the original image is restored to obtain the final prediction result. The structure of the DeepLabV3+ model is shown in Fig. 7 [19], [20].

D. FEATURE EXTRACTION NETWORK

The improved Xception (Chollet, [23]) feature extraction model used by DeepLabV3+ has been improved at different depths. The largest pooling layer in the model is replaced by multiscale hole convolution, the local normalization layer is added, and the nonlinear transformation is performed using the ReLU activation function.

To increase the depth of the middle layer to enhance the feature extraction ability, we used depthwise separable convolution to reduce the model parameters, which makes the model learning more efficient. Its structure is shown in Fig. 8(Chollet, [23]).

E. LOSS FUNCTION

DeepLabV3+ uses a negative class cross-entropy cost function, which is defined as follows:

$$p_k(x) = \frac{e^{a}k^{(x)}}{\sum_{k=1}^{K} e^{a}k^{(x)}} \tag{2}$$

$$E = \sum_{x} w(x) \log(p_{l(x)}(x)) \tag{3}$$

Refer to (2), the network output is a softmax classification function for the pixel level, x is the position of the two-dimensional pixel point, and $a_k(x)$ represents the position of the pixel point x in the channel k of the network output layer.The output is the confidence of each individual pixel x in the k class. Equation (3) indicates that the total loss of DeepLab uses cross-entropy loss, and $p_l(x)$ represents the output probability of the real tag.

To prevent overfitting and improve model robustness, we usually add regularization terms after the loss function. The L2 regularization term is added here to penalize the loss function. L2 regularization is also called ridge regularization and is defined in Equation (4):

$$L_2 = \frac{1}{2}\eta \sum_{i=1}^{n} \theta^2 \tag{4}$$

where η is the regularization coefficient and θ is the weight. In the backpropagation optimization, as the loss of the loss function is reduced, the loss of the regular term is also reduced.

F. EVALUATION STANDARD

To measure the performance and learning cost of each model, and to evaluate the model more effectively, the experiment used multiple levels of control parameter variables for the evaluation. The main evaluation indicators included the training time of the model, the accuracy of the model prediction, the memory occupancy, and the size of the model parameters. Under the conditions of a controlled hardware configuration and fixed parameters, a comparison experiment was carried out

There are many criteria for measuring the accuracy of image segmentation. In general, MIoU is the most representative evaluation index. It refers to the intersection of the set of predicted values of the model and the set of true values of the sample labels. The ratio of the unions is determined by calculating the intersection of each class and adding the average. Its mathematical expression is

$$MIoU = \frac{1}{k+1} \sum_{i=0}^{k} \frac{p_{ii}}{\sum_{j=0}^{k} p_{ij} + \sum_{j=0}^{k} p_{ji} - p_{ii}} \tag{5}$$

where k is the number of categories, for a total of k+1 classes (including a background class); p_{ii} is the number of pixels predicted to be correct; p_{ij} is the number of pixels predicted to be the background but is actually a positive label; and p_{ji} is the number of pixels predicted to be the foreground but is actually a negative label.

G. TRANSFER LEARNING

Transfer learning is based on the network weights saved by previous researchers in the big data set and migrated to the experimental network with a similar structure when the hardware configuration ability is insufficient, and the learning time is too long. At this point, the weights of the training in the

big data set will be used in their own experiments and only a fine tuning is needed to obtain a better result model [25]. Through transfer learning, the gradient disappearance and gradient explosion problem can be effectively prevented. The neural network gets faster and provides more effective convergence, saves on learning time cost, improves the learning efficiency, and enhances the robustness of the model.

Transfer learning makes a trained convolutional neural network model suit for a new task through simple adjustment. The convolution layer of trained convolution neural network can extract image features, and the extracted feature vector can be input into the fully connected layer with simple structure to achieve better recognition and classification. So the feature vector extracted by convolution layer can be used as a more concise and more expressive vector of the image. Therefore, the trained convolution layer and the full connection layer suitable for the new task will form a new network model, and a little training on the new network model can handle the new classification and recognition task.

At present, transfer learning is very common in neural networks. The experiment can make the network convergence faster and more efficient by learning the migration on big data sets. Only the last layer of the network needs to be modified, as the front layer of the feature extraction network is pretrained. Training based on the parameters reduces the problem of insufficient generalization ability and insufficient precision owing to the small amount of data.

V. EXPERIMENTAL

A. TYPES OF GRAPHICS

The experiment used the TensorFlow deep learning framework. The hardware equipment used had the following configurations and installed software: Intel Core i7-6700 CPU @ 3.40 GHz × 8 threads, 16 GB of RAM, GeForce GTX TITAN X GPU with 12 GB of RAM, 500-GB mechanical hard disk, NVIDIA driver version 390.87, CUDA version 9.0.176, CUDNN 7.0.5 neural network acceleration library, Linux Ubuntu 18.04 LTS operating system, Python version 3.6, and TensorFlow version 1.8.0.

According to the hardware configuration of the experimental machine, the TensorFlow learning framework was used to convert the data into TensorFlow's unique binary TFRecord format for data reading. The organized training set was 96 MB in size, and the test set size was 30 MB. Using the DeepLabV3+ semantic segmentation network and a random gradient descent method for parameter learning, we set the number of samples per batch of incoming network to 8 and selected the following feature extraction models: Xception_65, Xception_41, and Xception_71. The coding structure used ASPP with 6, 12, and 18 holes; the sample clipping size was 321 × 321; the weight decay coefficient was 0.00004; the training iteration number was 50,000 times; and the control variables were basically the same. A comparison experiment was then performed, as shown in Table 1 and Table 2. The experiment results showed that DeepLabV3+ used the Xception_65 feature extraction network to obtain the best results, achieving an MIoU of 0.765.

The segmentation effect diagram was showed in Fig.9. It can be seen that the position of branches can be segmented,

TABLE 1. Training parameters and results.

Model_variant	Atrous_rates	Output_stride	Decoder_output_stride	Train_crop_size	Train_batch_size	Training_number_of_step	Weight_decay
Mobilenet_v2	-	-	-	321x321	8	50000	0.00004
Resnet_v1_50	6, 12, 18	16	4	321x321	8	50000	0.0001
Resnet_v1_50_beta	6, 12, 18	16	4	321x321	8	50000	0.0001
Resnet_v1_101	6, 12, 18	16	4	321x321	8	50000	0.0001
Resnet_v1_101_beta	6, 12, 18	16	4	321x321	8	50000	0.0001
Xception_41	6, 12, 18	16	4	321x321	8	50000	0.00004
Xception_71	6, 12, 18	16	4	321x321	8	50000	0.00004
Xception_65	6, 12, 18	16	4	321x321	8	50000	0.00004

TABLE 2. Evaluation parameters and results.

Model_variant	Eval_crop_size	MIoU	Time_consumed(ms)	Size (MB)	Time_total	Pretraining_model
DeepLabV3+Mobilenet_v2	1505x1505	0.671	130	8.7	3 h 29 m 57 s	COCO+VOC
DeepLabV3+*Resnet_v1_50*	1505x1505	0.621	277	107.3	6 h 59 m 56 s	ILSVRC-2012
DeepLabV3+*Resnet_v1_50_beta*	1505x1505	0.715	296	107.8	7 h 29 m 57 s	ImageNet
DeepLabV3+*Resnet_v1_101*	1505x1505	0.715	337	183.6	9 h 9 m 55s	ILSVRC-2012
DeepLabV3+*Resnet_v1_101_beta*	1505x1505	0.719	357	184.1	9 h 39 m 54s	ImageNet
DeepLabV3+Xception_41	1505x1505	0.493	371	113.3	9 h 29 m 55s	ImageNet
DeepLabV3+Xception_71	1505x1505	0.632	535	175.4	8 h 59 m 52s	ImageNet
DeepLabV3+Xception_65	1505x1505	**0.765**	457	165.6	13 h 59 m 52s	ImageNet

FIGURE 9. Segmentation effect diagram of DeepLabV3+_Xception_65 (a) Image 1. (b) Image 2. (c) Image 3.

which provided the basis of early operation for the location of picking points.

B. EXPERIMENTAL RESULT

The experiment adopted a "ploy strategy" combined with the stochastic gradient descent method for optimization; the initial learning rate was set to 0.0001, each step was optimized, and the learning rate was reduced by 10 times. The momentum factor was set to 0.9, and each batch was fed into the network with eight cropping samples, and the sampling size was 321 × 321. A regularization term was added to the loss function to optimize the algorithm.

As shown by the graph in Fig. 10(a), the learning rate decreases as the number of training steps increases. The purpose is to greatly find the bottom of the convex optimization and obtain better model performance. The graph shown in Fig. 10(b) shows the loss reduction curve of the regularization term in the loss function. By adding the regularization term, we made the model more robust. Experiment results showed that the prediction accuracy of the model can be significantly increased after adding the regularization term.

Because the transfer learning algorithm uses a pre-training model, it can quickly converge to a small loss, as shown in Fig. 11. Through observation, it was found that, when the loss value drops to approximately 0.15, no large fluctuations are generated, which proves that the model has converged to the optimal state and the training can be suspended.

The experimental evaluation parameters were selected in accordance with the training parameters, as shown in Table 3. Because the experimental sample specifications were different, the largest specification sample was selected as the cutting standard. The final evaluation of the crop size was 1505 × 1505, which is in accordance with the integer multiple decoding standard of the encoder output, and the maximum evaluation score was obtained. The experiment with DeepLabV3+_Xception_65 obtained an MIoU of 0.765.

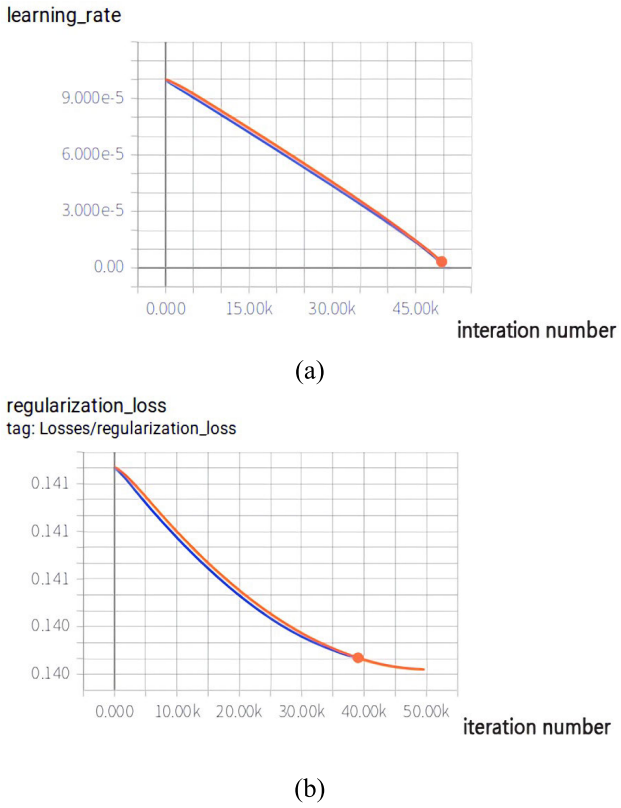


FIGURE 10. Learning rate decline curve. (b) Regularized loss reduction curve.

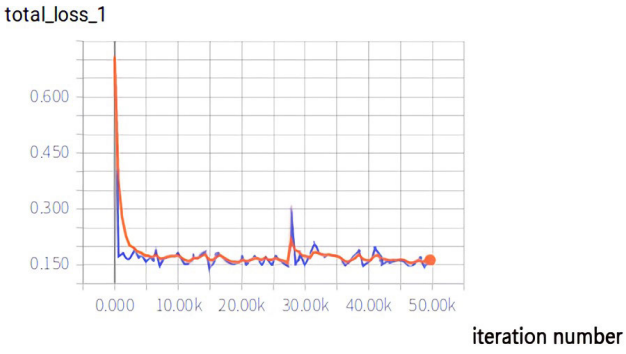


FIGURE 11. Training loss curve.

C. COMPARATIVE EXPERIMENT

Contrast experiment is the key factor to evaluate the quality of the model. This study used multiple sets of comparative experiments, and multiple evaluation indicators showed the evaluation results more comprehensively and concretely.

As shown in Table 3, the comparison results of similar models showed that the image semantic segmentation network using the Xception_65 feature extraction network has a higher MIoU; at the same time, however, there are also large models and deep convolution layers, which can lead to a longer detection time. It seems that Xception_65 and MobileNetV2 have advantages in detection accuracy and detection efficiency, respectively.

TABLE 3. IoU of figure 13.

Index	Resnet101_beta	MobilenetV 2	Xception_65
Image1	0.721	0.666	0.767
Image2	0.719	0.672	0.760
Image3	0.720	0.670	0.769
Iamge4	0.716	0.668	0.766

TABLE 4. Comparison results of models with different architectures.

Model	MIoU	Size(MB)
DeepLabV3+_Xception_65	0.765	165.6
PSPNet_101	0.557	265
ICNet	0.472	85

Table 2 shows a comparison of the pretrained models of the different data sets used by the different models with the time required to train the 50,000 steps. The results showed that the training time of the small network of MobileNet is shorter, whereas large networks require a longer training time. ResNet_50_beta and ResNet_101_beta are based on ResNet_50 and ResNet_101, respectively, and use large convolution kernels instead of multiple small convolution kernels in the starting layer. The experimental data showed that the parameters of the model after the large convolution replace the small convolution increase and that the model size increases. However, owing to the loss of shallow information, the accuracy of the segmentation network is slightly improved.

As shown in Figure 12, from the result of the large number of model comparisons, three strong representative models are selected for visual comparison. The value of the IoU is as shown in Table 3. From the comparison effect, it can be seen that the Xception_65 model has outstanding effects in refining the edges and in detecting accuracy. It can be seen that ASPP and depthwise separable convolution have contributed greatly to the improvement of model capabilities.

This study also conducted comparative experiments between different architectures. Experiment results showed that the DeepLabV3+ architecture model is far more robust than PSPNet_101 (Zhao *et al.* [26]) and ICNet [26], [27], as shown in Table 4.

VI. CONCLUSION

In this study, we used an image semantic segmentation technology to classify litchi branches by using pixel-level classification and achieved the desired goal and separation effect. During the experiment, the DeepLabV3+ semantic segmentation framework was selected, and its segmentation principle and segmentation advantages are systematically explained in this article. The DeepLabV3+ semantic segmentation model combined with the Xception_65 feature extraction network realized the semantic segmentation of litchi branches. Its MIoU reached 0.765, achieving the maximum separation effect within the allowable range of the hardware environment and performing numerous contrast experiments.

Select the feature extraction network based on depthwise separable convolution. Experiment results showed that the features acquired by its feature extraction network are more detailed, the information abstraction extraction ability is stronger, and its generalization and sampling abilities are highlighted in the horizontal development of convolutional neural networks.

Use a vertically developed residual network for comparative experiments. Experiment results showed that the Xception model of the residual network is mainly insufficient in the local aspect, and, thus, corresponding experiments were also conducted on the models with different architectures.

Adopt a coding and a decoding structure to reduce the number of network parameters; with atrous spatial pyramid pooling, the semantic pixel position information can be extracted more efficiently without increasing the number of weight parameters. The decoding uses upsampling and the shallow features to fuse, and the same weight is assigned to ensure that the shallow feature semantics and the deep feature semantics are evenly distributed. The use of different sizes of hole convolution ensures the prediction accuracy of small targets.Image semantic segmentation plays an important role in the field of computer vision. Pixel-level semantic segmentation can extract semantic prediction semantics from irregular targets and then postprocess the targets. The location information of the branches is obtained through semantic segmentation, which provides powerful technical support for the gripper picking robot to find the fruit branches and which provides a new solution for the problem of aim detection and recognition in agricultural automation.

REFERENCES

[1] Y. Tang, M. Chen, C. Wang, L. Luo, J. Li, G. Lian, and X. Zou, "Recognition and localization methods for vision-based fruit picking robots: A review," *Frontiers Plant Sci.*, vol. 11, pp. 1–17, May 2020.

[2] Y. Tao and J. Zhou, "Automatic apple recognition based on the fusion of color and 3D feature for robotic fruit picking," *Comput. Electron. Agricult.*, vol. 142, pp. 388–396, Nov. 2017, doi: 10.1016/j.compag.2017.09.019.

[3] X. Wei, K. Jia, J. Lan, Y. Li, Y. Zeng, and C. Wang, "Automatic method of fruit object extraction under complex agricultural background for vision system of fruit picking robot," *Optik*, vol. 125, no. 19, pp. 5684–5689, Oct. 2014.

[4] J. J. Zhuang, S. M. Luo, C. J. Hou, Y. Tang, Y. He, and X. Y. Xue, "Detection of orchard citrus fruits using a monocular machine vision-based method for automatic fruit picking applications," *Comput. Electron. Agricult.*, vol. 152, pp. 64–73, Sep. 2018, doi: 10.1016/j.compag.2018.07.004.

[5] Y. Tian, G. Yang, Z. Wang, H. Wang, E. Li, and Z. Liang, "Apple detection during different growth stages in orchards using the improved YOLO-V3 model," *Comput. Electron. Agricult.*, vol. 157, pp. 417–426, Feb. 2019, doi: 10.1016/j.compag.2019.01.012.

[6] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," NASA Astrophys. Data Syst., Tech. Rep., 2018. [Online]. Available: http://arxiv.org/abs/1804.02767

[7] I. Sa, Z. Ge, F. Dayoub, B. Upcroft, T. Perez, and C. McCool, "DeepFruits: A fruit detection system using deep neural networks," *Sensors*, vol. 16, no. 8, p. 1222, Aug. 2016, doi: 10.3390/s16081222.

[8] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017, doi: 10.1109/TPAMI.2016.2577031.

[9] S. Bargoti and J. P. Underwood, "Image segmentation for fruit detection and yield estimation in apple orchards," *J. Field Robot.*, vol. 34, no. 6, pp. 1039–1060, Sep. 2017, doi: 10.1002/rob.21699.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf., Process. Syst.*, 2012, pp. 1097–1105.

[11] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 1–9, doi: 10.1109/CVPR.2015.7298594.

[12] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*. [Online]. Available: http://arxiv.org/abs/1502.03167

[13] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826, doi: 10.1109/CVPR.2016.308.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[15] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn.*, 2015, pp. 1–14.

[16] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017.

[17] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017, doi: 10.1109/TPAMI.2016.2644615.

[18] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected CRFs," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–14.

[19] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018, doi: 10.1109/TPAMI.2017.2699184.

[20] L. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 801–818.

[21] B. Hariharan, P. Arbelaez, R. Girshick, and J. Malik, "Simultaneous detection and segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 297–312.

[22] L. Xu, J. Ren, C. Liu, and J. Jia, "Deep convolutional neural network for image deconvolution," in *Proc. Adv. Neural Inf., Process. Syst.*, 2014, pp. 1790–1798.

[23] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 1800–1807, doi: 10.1109/CVPR.2017.195.

[24] W. Yang, J. Zhang, Z. Xu, and C. Zhao, "An improved focal loss function for semantic segmentation," *Semicond. Optoelectron.*, vol. 40, no. 4, pp. 555–559, 2019.

[25] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010, doi: 10.1109/TKDE.2009.191.

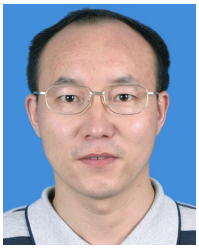
[26] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 6230–6239, doi: 10.1109/CVPR.2017.660.

[27] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia, "ICNet for real-time semantic segmentation on high-resolution images," in *Proc. Comput. Vis. Pattern Recognit.*, Sep. 2017, pp. 405–420.

[28] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. S. Torr, "Conditional random fields as recurrent neural networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1529–1537, doi: 10.1109/ICCV.2015.179.

[29] Z. Liu, X. Li, P. Luo, C. C. Loy, and X. Tang, "Deep learning Markov random field for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 8, pp. 1814–1828, Aug. 2018, doi: 10.1109/TPAMI.2017.2737535.

[30] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, 2015, pp. 3431–3440, doi: 10.1109/CVPR.2015.7298965.



HONGXING PENG received the B.S. degree from Jiangxi Normal University, Nanchang, China, in 1998, the M.S. degree from Tianjin Normal University, Tianjin, China, in 2004, and the Ph.D. degree from South China Agricultural University, Guangzhou, China, in 2014. From 2004 to 2014, he was a Lecturer with South China Agricultural University, where he has been an Associate Professor with the Computer Science and Engineering Department, since 2015. His research interests include computer vision, image processing, artificial intelligence, robot, big data, and virtual reality.

CHAO XUE received the B.A. degree from Henan Normal University, in July 2017, the M.A. degree from Shandong Agricultural University, in June 2019, and the degree from South China Agricultural University. His current research interests include computer vision and image processing.

YUANYUAN SHAO was born in Jining, Shandong, China, in 1980. She received the Ph.D. degree in mechanical and electronic engineering from the Shandong University of Science and Technology, Qingdao, Shandong, in 2012.

Since 2016, she has been an Assistant Professor with the College of Mechanical and Electronic Engineering, Shandong Agricultural University, Tai'an, Shandong. She is the author of two books, more than 20 articles, and more than four inventions. Her research interests include smart agriculture, nondestructive testing of agricultural products quality, imaging process, deep learning, hyperspectral technology, and mechanical design.

KEYIN CHEN received the Ph.D. degree from South China Agricultural University. He is currently a Postdoctoral Researcher of agricultural electrification and automation engineering with the Nanjing Institute of Agricultural Mechanization and a Lecturer of electrical engineering and its automation with the Key Laboratory for Protection and Precise Utilization of Characteristic Agricultural Resources in Mountainous Areas of Guangdong Province and the School of Physics and Electronic Engineering, Jiaying University. His research interests include agricultural robot, machine vision, and bionic intelligence technology.

JUNTAO XIONG was born in Jingzhou, Hubei, China, in 1981. He received the Ph.D. degree in agricultural mechanization engineering from South China Agricultural University, PA, in 2012. From 2007 to 2020, he has worked with the College of Mathematics and Information, South China Agricultural University. He is the author of 30 articles and more than ten inventions. His research interest includes the application of machine vision and artificial intelligence.

ZHIHUA XIE is currently pursuing the master's degree with South China Agricultural University.

LIUHONG ZHANG is currently pursuing the master's degree with South China Agricultural University.