Segmentation of field grape bunches via an improved pyramid scene parsing network

Shan Chen[1,2,3], Yuyang Song[4], Jinya Su[5], Yulin Fang[4], Lei Shen[1,2,3], Zhiwen Mi[1,2,3], Baofeng Su[1,2,3*]

(1. *College of Mechanical and Electronic Engineering, Northwest A&F University, Yangling 712100, Shaanxi, China*;
2. *Key Laboratory of Agricultural Internet of Things, Ministry of Agriculture and Rural Affairs, Yangling 712100, Shaanxi, China*;
3. *Shaanxi Key Laboratory of Agricultural Information Perception and Intelligent Services, Yangling 712100, Shaanxi, China*;
4. *College of Enology, Northwest A&F University, Yangling 712100, Shaanxi, China*;
5. *School of Computer Science and Electronic Engineering, University of Essex, Colchester, CO4 3SQ, UK*)

Abstract: With the continuous expansion of wine grape planting areas, the mechanization and intelligence of grape harvesting have gradually become the future development trend. In order to guide the picking robot to pick grapes more efficiently in the vineyard, this study proposed a grape bunches segmentation method based on Pyramid Scene Parsing Network (PSPNet) deep semantic segmentation network for different varieties of grapes in the natural field environments. To this end, the Convolutional Block Attention Module (CBAM) attention mechanism and the atrous convolution were first embedded in the backbone feature extraction network of the PSPNet model to improve the feature extraction capability. Meanwhile, the proposed model also improved the PSPNet semantic segmentation model by fusing multiple feature layers (with more contextual information) extracted by the backbone network. The improved PSPNet was compared against the original PSPNet on a newly collected grape image dataset, and it was shown that the improved PSPNet model had an Intersection-over-Union (IoU) and Pixel Accuracy (PA) of 87.42% and 95.73%, respectively, implying an improvement of 4.36% and 9.95% over the original PSPNet model. The improved PSPNet was also compared against the state-of-the-art DeepLab-V3+ and U-Net in terms of IoU, PA, computation efficiency and robustness, and showed promising performance. It is concluded that the improved PSPNet can quickly and accurately segment grape bunches of different varieties in the natural field environments, which provides a certain technical basis for intelligent harvesting by grape picking robots.

Keywords: grape bunches, semantic segmentation, deep learning, improved PSPNet
DOI: 10.25165/j.ijabe.20211406.6903

Citation: Chen S, Song Y Y, Su J Y, Fang Y L, Shen L, Mi Z W, et al. Segmentation of field grape bunches via an improved pyramid scene parsing network. Int J Agric & Biol Eng, 2021; 14(6): 185–194.

1 Introduction

The harvesting of wine grapes is undoubtedly one of the most time-consuming and labor-intensive parts of the wine production chain. In addition, with the aging population, the agricultural labor force is gradually decreasing and the cost of manual harvesting is increasing, which significantly reduces the market competitiveness of the product. Manual harvesting also poses a great risk to the health of the grape harvesters. Therefore, the use of robots for mechanized and intelligent picking of grapes has become an inevitable trend for future development. Real-time detection and segmentation of grapes remain a challenging task due to the complexity of the natural environments in the field and the diversity of the background[1,2].

Many scholars have conducted numerous studies on the division of grape bunches. Murillo-Bracamontes et al.[3] proposed a method based on Hough transform to find grape cluster boundaries for segmenting grape bunches, but the method is only for grape images in a laboratory setting, without considering the adverse effects of other complex backgrounds such as lighting, trunks, branches and leaves. Reis et al.[4] proposed a system based on image processing techniques that can detect and localize grape bunches in color images in a natural environment. The system achieved an accuracy of 97% and 91% for colored and white grapes, respectively. The disadvantage therein is that the system works at night and is not adapted to the complex lighting changes in the field during the day. Liu and Whitty[5] segmented field grape bunches for grape yield estimation based on Support Vector Machine (SVM) with an accuracy of 88.0%, but the study was conducted on colored grape images collected at night with strict requirements for lighting conditions and color differences between grapes and background. Pérez-Zavala et al.[6] used the Histogram of Oriented Gradients (HOG) as a shape description conforming to the Local Binary Pattern (LBP) to obtain texture information based on the comparative analysis of image feature vectors and SVM to select the best strategy to separate grape bunches. However, the method is performed on images of high quality without the intrusion of leaves and other backgrounds. In addition, convolutional neural networks (CNNs), as a deep learning method, have been successfully applied for the detection and recognition of grape bunches in natural field environments with better

Received date: 2021-07-24 **Accepted date:** 2021-10-02

Biographies: Shan Chen, Master candidate, research interest: image processing, Email: chenshan@nwafu.edu.cn; **Yuyang Song**, PhD, Associate Professor, research interest: viticulture, Email: yuyangsong@nwsuaf.edu.cn; **Jinya Su**, PhD, Associate Professor, research interest: machine learning, Email: j.su@essex.ac.uk; **Yulin Fang**, PhD, Professor, research interest: horticulture, Email: fangyulin@ nwsuaf.edu.cn; **Lei Shen**, Master candidate, research interest: smart agriculture, Email: shenlei@nwafu.edu.cn; **Zhiwen Mi**, PhD candidate, research interest: UAV remote sensing, Email: 18392106659@nwafu.edu.cn.

***Corresponding author: Baofeng Su**, PhD, Associate Professor, research interest: plant phenotyping. College of Mechanical and Electronic Engineering, Northwest A&F University, Yangling 712100, Shaanxi, China. Tel: +86-29-87091527, Email: bfs@nwsuaf.edu.cn.

performance in a large number of classification tasks. Cecotti et al.[7] implemented the detection of different varieties of grapes based on ResNet networks with an accuracy of 99%, but the method was also only tested on high-quality images. Milella et al.[8] successfully achieved grape bunch recognition based on the VGG19 deep neural network with an accuracy of 91.52%. Marani et al.[9] proposed a new cluster beam pixel segmentation strategy based on the VGG19 network for segmenting white grape images in the field, and the results showed that the segmentation accuracy and the Intersection-over-Union (IoU) of grape bunches was 80.58% and 45.64%. It can be seen that although the method can segment white grapes in the field environments, the segmentation results are not good enough and needs further improvement.

In recent years, with the development of deep learning technology, semantic segmentation methods based on deep learning have also been used in a large number of natural image segmentation fields[10-15]. For plant fruit segmentation, Kang and Chen[16] used a deep convolutional neural network (entitled DaSNet) for real-time detection and semantic segmentation of apples in apple orchards, and finally obtained a segmentation accuracy of 86.5% for apples. Royet al.[17] implemented semantic segmentation of the decayed portion of RGB apple images based on the improved U-Net framework and obtained an IoU of 86.6%. Li et al.[18] used the DeepLabV3 semantic segmentation method to segment the acquired RGB images of field litchi, and the results showed that the litchi detection accuracy was 83.33% and the average detection time was 0.464 s. Kestur et al.[19] proposed a new MangoNet semantic segmentation network with better robustness in terms of scale, illumination, contrast and occlusion to accurately segment mangoes in an orchard environment. Although the above studies have achieved relatively good segmentation results, they are generally limited to the segmentation target with a significant color difference from the background, and the segmentation performance will be greatly degraded when the color and background share a certain similarity.

The idea of the attention mechanism is to teach the system to pay more attention to the target information and ignore irrelevant information, which can improve the network performance without adding additional computation load. In recent years, attention mechanisms have also been applied in the field of image segmentation. For example, Shu et al.[20] embedded the Convolutional Block Attention Module (CBAM) attention mechanism into DeepLab-V3+ to obtain an end-to-end semantic slicing network AMNet, and obtained 77.66% mean IoU on the PASCAL VOC2012 dataset. Lin et al.[21] proposed a GLNet model incorporating the SE (Squeeze-and-Excitation) attention module, which achieved 80.8% accuracy on the Cityscapes test dataset. Atrous convolution was proposed in the field of image segmentation, which is useful for target segmentation as it extracts more semantic information of the target object by increasing the perceptual field of the feature map. Wang et al.[22] applied atrous convolution to the model backbone feature extraction network in order to achieve an accurate segmentation of poplar plums in the natural environment, with an average detection accuracy and recall of 97% and 91%, effectively achieving accurate identification and segmentation of poplar plums. Li et al.[23] artificially obtain the boundary semantic information of the target image, apply atrous convolution to resolve the contradiction between the resolution of the feature map and the received field, retain more multi-scale contextual information, and eventually achieve an effective segmentation of green apples in real orchards.

Under natural field conditions, the captured grape images are affected by various environmental conditions such as uneven illumination and complex backgrounds, which have a great impact on the real-time segmentation of grape images. In addition, the near-field problem of colorless grapes also poses a challenge to grape segmentation. In this study, the semantic segmentation model PSPNet was improved by adding attention mechanism and atrous convolution to the backbone feature extraction network in order to improve the feature extraction ability of the network. Meanwhile, multiple feature layers in the enhanced feature extraction stage were also fused to obtain more image details for achieving an accurate and efficient segmentation of different varieties of grape bunches. The improved PSPNet was validated on the newly collected grape bunches dataset and also compared against the baseline PSPNet and the state-of-the-art DeepLab-V3+ and U-Net with promising performance. As a result, the improved PSPNet based grape bunches segmentation model was able to provide some technical support for intelligent harvesting of grapes and could also provide a basis for subsequent research on grape bunches-related phenotypes.

2 Data acquisition and pre-processing

2.1 Image data acquisition

In this study, grape images with complex backgrounds in the field conditions were used as the object of study for bunches segmentation experiments. The data collection site for grape images was located at the wine grape production demonstration base in Yangling District, Shaanxi Province, China, and the experimental area is shown in Figure 1.

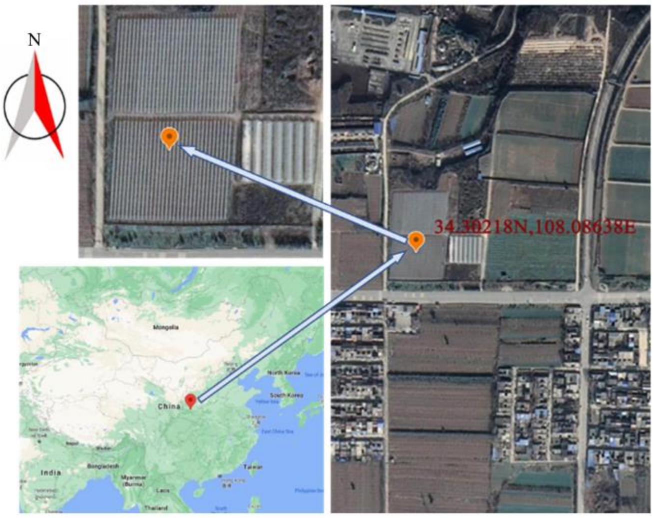
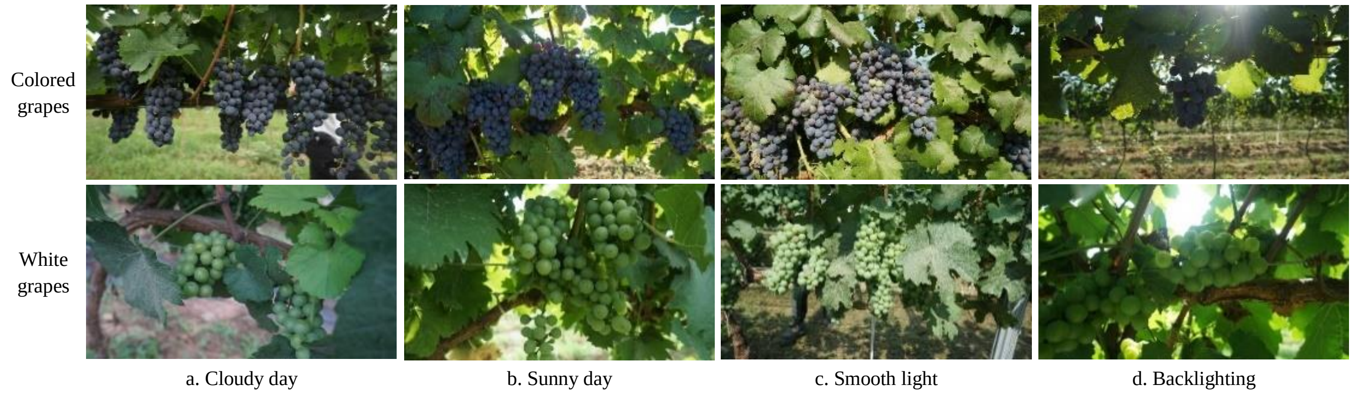


Figure 1 Geographical location of the experimental orchards in this study

Six wine grape varieties were selected for the trial, including three white grape varieties: Chardonnay, Guinness, and Riesling; and three colored grape varieties: Cabernet Sauvignon, Matheran, and Syrah. A SONY ILCE-5100L digital camera manufactured by Sony Thailand was used for grape image acquisition, with an image spatial resolution of 3008×1668 pixels, an aperture value of *f*/3.2, and an exposure time of 1/60 s. The image acquisition was done in July-August 2020, and the acquisition time was 9:00 a.m.-12:00 p.m. every day, in order to increase the diversity of image samples and enhance the algorithm robustness against environmental variations. The camera lens was randomly distanced from the grape at 50-100 cm during the acquisition, and the acquisition environment included different weather conditions such as sunny and cloudy days, as well as different lighting conditions such as downlight and backlight. Some of the collected grape images are shown in Figure 2. A total of 1856 grape images were collected, where the number of images for each grape variety is listed in Table 1.



a. Cloudy day b. Sunny day c. Smooth light d. Backlighting

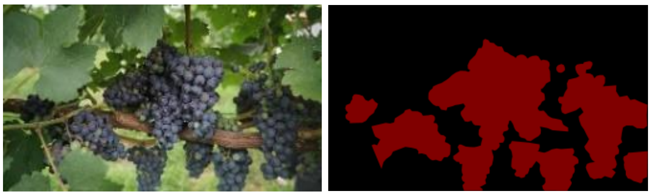
Figure 2 Images of different grape varieties in a complex field environment

Table 1 Number of grape image samples for different varieties

Parameters		Sunny		Cloudy day	Total /frames
		Downlight	Backlight		
Colored grape varieties	Cabernet Sauvignon	81	83	154	318
	Matheran	122	107	78	307
	Syrah	86	97	150	333
White grape varieties	Chardonnay	56	85	169	310
	Noble Fragrance	74	72	156	302
	Riesling	95	103	88	286
Total/frames		514	547	795	1856

2.2 Image pre-processing

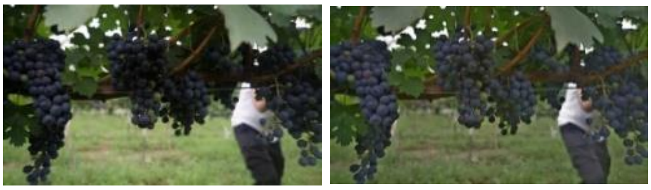
In order to train the grape bunches segmentation network model, the captured images should be labelled first. The image labeling tool “Labelme” is used to label the grape image samples with segmentation masks, and after labeling, a JSON file is generated, which is converted into a 24-bit grayscale map as the label of the image samples. The image of the grape mask after labeling is shown in Figure 3.



a. Original image b. Masked images

Figure 3 Image annotation of grape bunches

In the process of capturing grape images under natural field conditions, there were complex background environments and variable lighting conditions, which make the scene unevenly lit. As a result, some important target details in the image could not be highlighted or masked out, which greatly reduced the image quality. Therefore, the grape images need to be processed for light equalization. It was shown that the adaptive correction algorithm based on two-dimensional gamma function for illuminated inhomogeneous images had better performance over the Retinex theory, histogram equalization algorithm and morphological filtering method[24]. Therefore, this study adopted a two-dimensional gamma function-based adaptive correction algorithm for illumination inhomogeneous images (i.e., reduce the luminance value at too strong regions and increase the luminance value at too dark regions), so as to realize the adaptive correction process. The effect of image illumination correction is shown in Figure 4.



a. Original image b. Corrected image

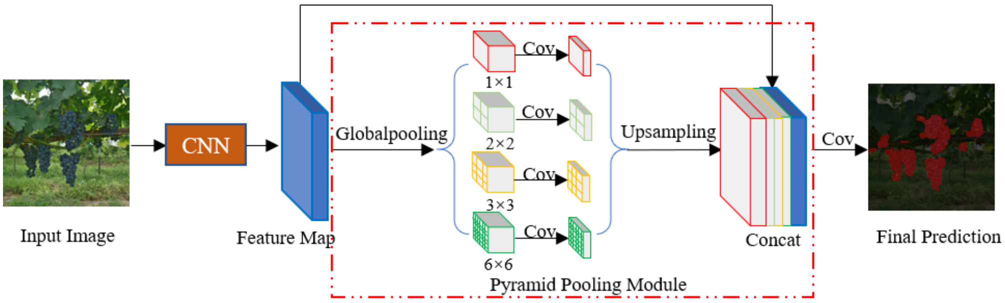
Figure 4 Image illumination unevenness correction of grape bunches

Since the model in this study required an image input resolution of 473×473 pixels, and the acquired initial image resolution was 3008×1688 pixels, in order to improve the training efficiency of the model, these 1856 acquired images and their corresponding mask images were uniformly cropped to 473×473 pixels. The original data were also augmented by rotating plus or minus 10 °and flipping 180 °horizontally to improve the overfitting problem that may be caused by too few samples in the process of model training. By the above method, the image samples are finally expanded to 7424, and the training set, validation set, and test set are divided according to the ratio of 8:1:1.

3 Construction and improvement of PSPNet segmentation model

3.1 Baseline PSPNet model

The original PSPNet model consists of a backbone feature network (CNN) and an enhanced feature extraction structure (PSP module), and its network framework is shown in Figure 5. Since the residual structure can solve the problem of gradient disappearance and network degradation that occurs with the deepening of the network. ResNet-50 was selected as the base network of the backbone feature extraction network. There is a bottleneck structure in ResNet-50, which first used 1×1 convolution to reduce the dimension, then 3×3 convolution, and finally 1×1 convolution to raise the dimension. Compared to the direct convolution with a 3×3 network, the ResNet-50 structure is better, with fewer parameters and so more efficient network training. In this study, the ResNet-50 was to initialize the network parameters with pre-trained weights on the VOC12+SBD dataset and perform migration learning and fine-tuning of the model on the dataset of this study.



Note: CNN: Convolutional Neural Network; Cov: Convolution

Figure 5 Structure of the original PSPNet Framework

The core of the PSPNet model is mainly the Pyramid pooling module. This module can divide the incoming feature layer into multiple regions of different sizes, and each region is pooled individually and equally within each region, thus enabling the aggregation of contextual information from different regions and improving the ability to obtain global information. In the classical structure of PSP, the input incoming feature layers are generally divided into 1×1, 2×2, 3×3, and 6×6 grids. The feature map extracted from the backbone feature network was passed through a pyramid pooling module to obtain fused features with overall information, and the features were cascaded with the feature layers extracted from the backbone network to obtain the final features. Finally, a convolutional layer was used to obtain the final output.

Considering the characteristics of the image dataset in this study, a semantic segmentation network for grape bunches based on the improved PSPNet model was constructed by adapting and optimizing the existing PSPNet network. Three main improvements were made in the backbone feature extraction network and enhanced feature extraction stages: 1) embedding CBAM attention mechanism in the backbone feature extraction network; 2) replacing parts of the standard convolution in the Cov5 stage of the backbone feature extraction network with an atrous convolution; 3) fusing the multilayer features extracted by the backbone network in the pyramid pooling module. The motivations and details regarding the above three modifications were summarized as below.

3.2 Backbone feature extraction network embedding CBAM attention module

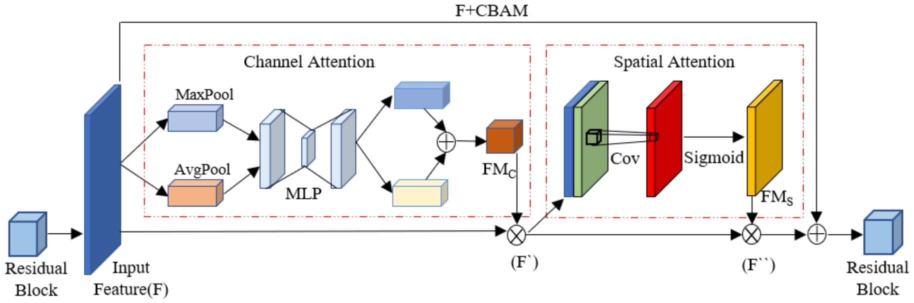
Attention not only tells the direction of attention but also enhances the representation of regions of interest. Improving the representation of target features by using attentional mechanisms means focusing on important features and suppressing unnecessary ones[25]. The objects of this study were divided into colored grapes and white grapes, where the white grape bunches have strong close-up characteristics with the surrounding background. Therefore, the CBAM convolutional attention mechanism was added to the backbone feature extraction network to improve the feature extraction capability of the model for the target grape bunches.

In this study, ResNet-50 was used as the backbone feature extraction network of the model, which contains five layers of convolutional residual blocks with different structures, and each residual block was formed by stacking convolutional layers with the same operation, and the layers were connected with a jump structure, which overcomes the problems of degradation of learning performance and gradient explosion of the deep network and can better learn high-level semantic features. In order not to change the network structure of ResNet-50, the embedding of the CBAM attention mechanism was performed after Cov1 and Cov5 phases, as shown in Figure 9.

The CBAM attention mechanism contains the channel attention mechanism and the spatial attention mechanism, as shown in Figure 6. For the incoming feature maps, the CBAM module will infer the attention maps sequentially along two independent dimensions, channel and space, and then multiply the attention maps with the incoming feature maps for adaptive feature optimization. In the CBAM, the channel attention mechanism puts the incoming feature maps through global-based maximum pooling and average pooling, respectively, and then inputs the pooling results into the multilayer perceptron (MLP), respectively, and sums the output results, and then obtains a weight coefficient FM_C through the sigmoid activation function, and finally multiplies the weight coefficient with the original feature map F to obtain the scaled new feature F`. The channel attention mechanism can be expressed as follows:

$$\mathrm{FM}_C=\sigma(\mathrm{MP}(\mathrm{MaxPool}(F)+\mathrm{MLP}(\mathrm{AvgPool}(F)))) \qquad (1)$$

where, σ is the activation function; MLP denotes the multilayer perceptron.



Note: CBAM means Convolutional Block Attention Module; MLP means multilayer perceptron; FM_C is the weight coefficient obtained by the sigmoid activation function, and finally multiplies the weight coefficient with the original feature map F to obtain the scaled new feature F`; $\mathrm{FM}_S$ is the weight coefficient obtained by the sigmoid activation function, and finally multiplies the weight coefficients with the incoming features $F$` to obtain the final features F``.

Figure 6 CBAM attention mechanism

The spatial attention takes the incoming features *F*` from the channel attention module as input, performs the average pooling and the maximum pooling of one channel dimension respectively and performs the serial cascade, then uses a 7×7 convolutional layer to downscale the feature channels, obtains the weight coefficients FM_S through the sigmoid activation function, and finally multiplies the weight coefficients with the incoming features *F*` to obtain the final features *F*``. The spatial attention mechanism can be expressed as follows:

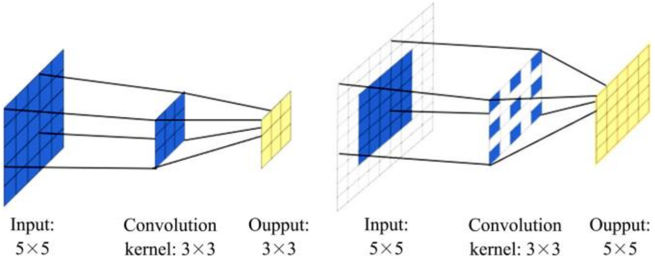
$$FM_S=\sigma\{f^{7\times7}[\mathrm{AvgPool}(F`);\ \mathrm{MaxPool}(F`)]\} \qquad (2)$$

where, σ is the activation function; f denotes the convolution kernel as a 7×7 convolutional layer; (;) denotes the serial cascade.

Finally, the final generated features *F*`` by the CBAM module are added to the original input features F to obtain the input of the next convolutional residual block.

3.3 Atrous Convolutional injection backbone network

Atrous Convolution is now widely used in tasks such as semantic segmentation and target detection[26-31]. By injecting holes on top of the convolution map of standard convolution, increases the feature map perceptual field without losing resolution and avoids the loss of spatial location information, as shown in Figure 7. Assuming that the input incoming feature map is 5×5 pixels in size, the first standard convolutional kernel is 3×3 pixels in size with a step size of 1. For each feature point on the output feature map, the perceptual field is 3×3 pixels, as shown in Figure 7a. For the atrous convolution, the convolution kernel size is 3 × 3 pixels, the Dilation rate is 2, the step size is 1, and the padding is 2. The perceptual field of each feature point on its output feature map is 5 × 5 pixels, as shown in Figure 7b. It can be seen that the atrous convolution increases the feature perceptual field while ensuring the feature map size remains unchanged, avoiding the loss of spatial resolution, which is important for semantic segmentation.



a. Standard convolution b. Atrous convolution

Figure 7 Illustration of the standard and atrous

In this study, the backbone feature extraction network ResNet-50 was modified to replace part of the convolution in the Cov5 stage with anatrous convolution, as shown in Table 2. Replace the 3×3 standard convolution in both Conv Block and Identity Block with 3×3 atrous convolution. Adding the atrous convolution to the backbone network increases the subsequent computational cost of the model, and adding the atrous convolution changes the structure of the network, which result in the inability to load more weight parameters pre-trained by the ResNet-50 model on the VOC12+SBD dataset, and thus make the training accuracy and training speed of the PSPNet model much lower. Therefore, in this study, only the case of adding the atrous convolution at the Cov5 stage was studied, and the feature map size of each stage of the final backbone network was 1/2, 1/4, 1/8, 1/16, and 1/16 of the original feature map size, respectively.

Table 2 Comparison of ResNet-50 structure before and after Cov5 improvement

Network layer name	Original ResNet-50	Improvements to ResNet-50
Cov5	1×1, standard convolution, number 512	1×1, standard convolution, number 512
	3×3, standard convolution, number 512	3×3, atrous convolution, number 512
	1×1, standard convolution, number 2048	1×1, standard convolution, number 2048
	1×1, standard convolution, number 2048, step size 1	1×1, standard convolution, number 2048, step size 1
	1×1, standard convolution, number 512	1×1, standard convolution, number 512
	3×3, standard convolution, number 512	3×3, atrous convolution, number 512
	1×1, standard convolution, number 2048	1×1, standard convolution, number 2048

3.4 Pyramid pooling module incorporating multi-layer features

The feature layers obtained through the backbone feature extraction network were passed into the pyramid pooling module. The module incorporates the features of four different pyramid scales, (e.g., 1×1, 2×2, 3×3, and 6×6), where 1×1 is a global average pooling of the entire input incoming feature map to generate a single bin output, the coarsest layer of features. The other three scales were divided into 2×2, 3×3, and 6×6 sub-regions for the incoming feature maps, and each sub-region was pooled on average to obtain three different scales of pooled features. To ensure the weight of the global features, the feature channels were downsampled using 1×1 convolution after different pyramid scales, and then the low-dimensional feature maps were upsampled by bilinear interpolation to the same scale as the original feature maps.

In order to get more detailed information about the grape bunches region, this study designed a multi-feature layer fusion structure for grape bunches region information extraction in the enhanced feature extraction stage of the model, whose structure is shown in Figure 8. When the input image was passed through the ResNet-50 backbone feature extraction network for feature extraction, four downsamplings were completed in sequence, and the fifth one was not compressed in terms of length and width, and only the number of channels was changed. The feature maps of each output layer were denoted by f_1, f_2, f_3, f_4, and f_5, respectively. For each downsampling, the feature map size was reduced to 1/2 of the original input image size and the number of feature channels becomes twice the original. The output sizes of f_5, f_4 and f_3 were 30×30×2048, 30×30×1024 and 60×60×512, respectively. In this study, the feature layer f_5 was first input into the pyramid pooling module to get the feature fused with the overall information, and then cascade f_5 with the feature to get the final feature F_5, at which time the output size of F_5 was 30×30×4096. The number of feature channels is adjusted to 1024 by 1×1 convolution and then fused with f_4 and 1×1 convolution to reduce the dimension to 1024 to obtain feature F_5`. Input feature $F_5$` into the pyramid pooling module to get the feature fused with overall information, and then cascade F_5` with this feature to get the final feature F_4, the output size of F_4 was 30×30×2048 at this time. The number of feature channels was adjusted to 1024 by 1×1 convolution and the length and width of the feature map were expanded to twice the original by bilinear interpolation, then fused with f_3 and 1×1 convolution was performed to reduce the dimension to 1024 to obtain feature F_4`. The feature $F_4$` was fed into the pyramid pooling module to get the feature fused with the overall information, and then F_4` was cascaded with this feature to get the final feature F_3. F_3 was the final feature output of the improved PSPNet model with an output size of 60×60×2048. In this way, the model incorporates the multilayer features extracted by the backbone feature network, utilizing not only the semantic information of the high-level features but also the spatial location information of the low-level features, so that more detailed information of the grape bunches can be extracted, which is conducive to the improvement of the model segmentation accuracy.

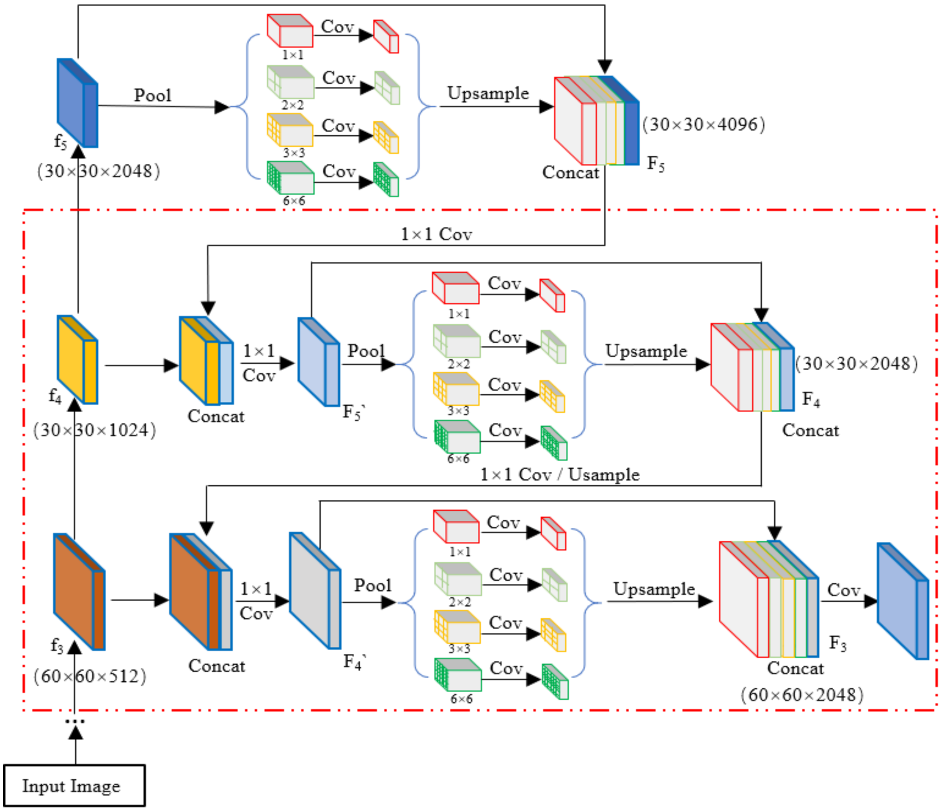


Figure 8 Multi-featured layer fusion structure

3.5 Improved PSPNet model

The proposed architecture of the ResNet-50 backbone feature network-based semantic segmentation model for grape bunches in natural environments is shown in Figure 9. The architecture consisted of two main modules: a backbone feature extraction network module embedded with CBAM attention mechanism and atrous convolution, and a pyramid pooling module fusing multi-layer features of the backbone network. The former improved the attention to the region of interest in the image by embedding CBAM attention mechanism between some convolutional residual blocks of the backbone network and injecting atrous convolution in the Cov5 stage, while suppressing useless features, and cancels the downsampling operation in the Cov5 stage to increase the perceptual field of the feature map to avoid the loss of image spatial location information and enhance the extraction of target features by the network. The latter aggregates contextual information at different scales through the pyramid pooling module and fuses multiple feature layers extracted by the backbone network to obtain more detailed information. This enhances the performance of the model to segment the grape bunches region in the complex field environment.

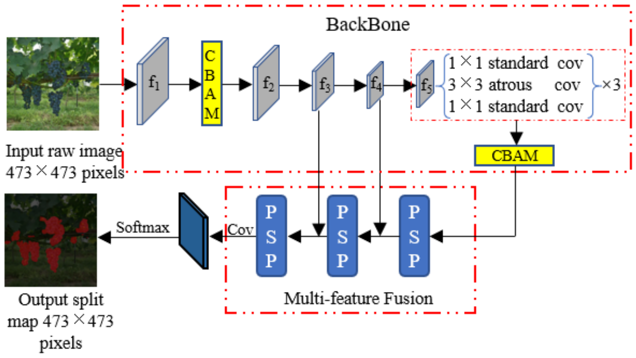


Figure 9 Improved PSPNet model framework for grape bunches segmentation

4 Results and discussion

4.1 Experimental platform and model training

The software environment used for the experiments was Windows 10 (64-bit) operating system, Python version 3.6.7, PyCharm 2020.1 Professional, with Tensorflow-GPU (version 1.13.1) as the back-end Keras (version 2.1.5) deep learning open source framework. The experimental hardware environment is AMD Ryzen 7 3700X 8-Core Processor with 3.6 GHz and 16GB RAM, and NVIDIA GeForce RTX 2070 SUPER with 8 GB video memory, equipped with CUDA 10.0 and CUDNN 7.4 as the acceleration toolkit for network model training. Model training was performed using the Adam optimizer for fine-tuning the parameters of the grape bunches region segmentation network model. The momentum was set to be 0.9, the initial learning rate was 0.0001, and the batch size was set to be 2. The ReduceLROnPlateau callback function was used to continuously shrink the learning rate during the training process to continuously adapt to the training of the model; it was also paired with the EarlyStopping callback function to interrupt the training when the model validation loss is no longer decreasing, so that the optimal model can be obtained quickly and accurately.

Iteratively train 100 epochs, save one model weight for every 2 epochs, and take the model with the highest accuracy as the final model.

4.2 Metrics for model performance evaluation

In this study, all the grapes in the image were grouped into one category and the background as the other categories. Therefore, the Pixel Accuracy (PA) and the IoU ratio are used as the evaluation indices of model performance. The higher the value of the indicator, the more effective the model is.

The pixel accuracy is the percentage of the number of correctly predicted category pixels in the image to the total number of predicted category pixels, and the expression is calculated as:

$$\mathrm{PA}=\sum_{i=0}^{k}\frac{p_{ii}}{\sum_{j=0}^{k}p_{ji}}\times 100\% \tag{3}$$

The intersection-to-merge ratio is the ratio of the model's intersection of the set of predicted and true value pixels of a category in the image to the set of merged pixels, and the expression is calculated as:

$$\mathrm{IoU}=\sum_{i=0}^{k}\frac{p_{ii}}{\sum_{j=0}^{k}p_{ij}+\sum_{j=o}^{k}p_{ji}-p_{ii}}\times 100\% \tag{4}$$

where, p_{ii} indicates the number of pixels that are actually in category I and are predicted to be in category I; p_{ij} indicates the number of pixels actually in category I but predicted to be in category J; p_{ji} indicates the number of pixels actually in category J but are predicted to be in category I; k indicates the number of different categories in the dataset, which is 2 in this study.

4.3 Experimental results and analysis

The experimental results of the PSPNet model with different structural compositions on the grape image test dataset in a natural field environment are shown in Table 3. Model 1 represents the original PSPNet model with ResNet-50 as the backbone network, with 83.06% IoU and 85.78% PA for grapes on the test set. Model 2 embeds the CBAM attention mechanism in the first convolutional residual layer and the last convolutional residual layer of the ResNet-50 backbone network, with an IoU of 84.36% and a PA of 87.18% for grapes, indicating an improvement of 1.3% and 1.4%, respectively, compared to Model 1. Model 3 adds the atrous convolution to the ResNet-50 backbone network based on Model 2, and the IoU of grapes is 85.73% and the PA is 89.49%, which is 2.67% and 3.71% higher than Model 1, respectively. Model 4 fuses multiple feature layers in the pyramid pooling stage based on Model 3, the IoU and PA of grapes are 87.42% and 95.73%, which is 4.36% and 9.95% higher than Model 1, respectively. Model 4 obtains the best performance among all models.

Table 3 Experimental results of different structural models on the test set

Model	Backbone Network	Whether the backbone network is embedded with CBAM modules	Whether the backbone network joins the atrous convolution	PSP module fuses multiple feature layers	IoU/%	PA/%
1	ResNet-50	No	No	No	83.06	85.78
2	ResNet-50	Yes	No	No	84.36	87.18
3	ResNet-50	Yes	Yes	No	85.73	89.49
4	ResNet-50	Yes	Yes	Yes	87.42	95.73

To further verify the effectiveness of the model, two other commonly used semantic segmentation models, DeepLab-V3+ and U-Net, were also trained by using the same dataset and parameters, where the experimental results are shown in Table 4. As can be seen from Table 4, the intersection ratio of the model in this study is slightly lower than that of the DeepLab-V3+ model, with a difference of 0.46%, but the pixel accuracy rate is 2.28% higher, and the average processing time of a single image is shortened by 72 ms, which is a significant improvement for the latter two. While the U-Net model is slightly higher than the improved PSPNet model by 17 ms in terms of average processing time for a single image, it is 0.74% and 4.84% lower in terms of cross-merge ratio and pixel accuracy, respectively, which are significantly inferior to the model. This again shows the better performance of the improved PSPNet.

Table 4 Test results of different semantic segmentation models

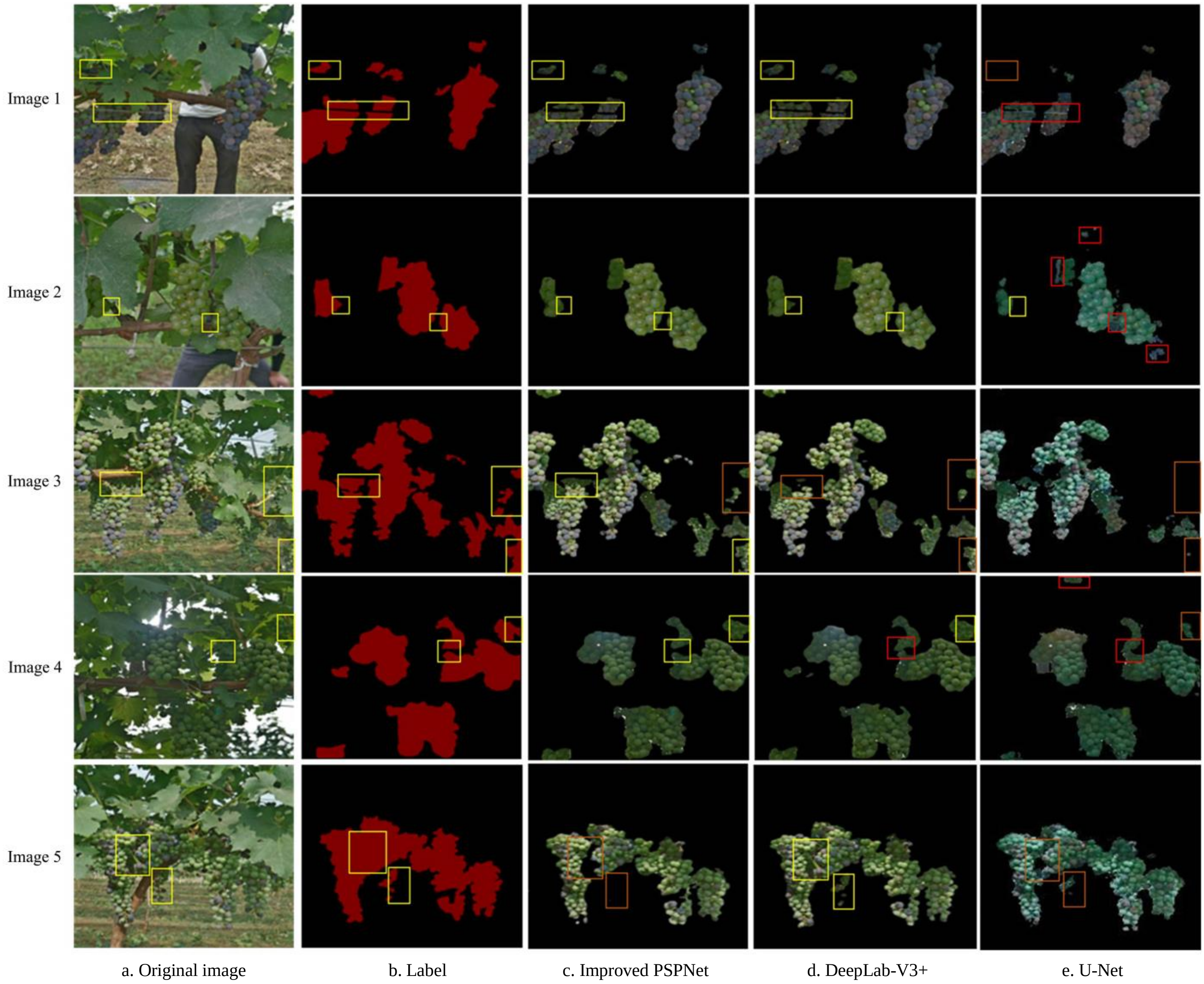
Model methodology	IOU/%	PA/%	Average processing time for a single image/ms
DeepLab-V3+	87.88	93.45	197
U-Net	86.68	90.89	108
Improved PSPNet	87.42	95.73	125

4.4 Different model segmentation results and analysis

In this study, experiments were conducted on several models using the grape test dataset, and some of the segmentation results are shown in Figure 10. It can be seen that the improved PSPNet model and other models successfully performed the task of segmenting grape bunches both for colored and colorless grape varieties.

The improved PSPNet model had good segmentation results for different varieties of grape images, but there were still exist some problems of missed segmentation, as shown by the orange boxed content in Images 3 and 5, where the model did not segment the grape region very completely (Figure 10c). DeepLab-V3+, as the current more advanced semantic segmentation network, also has good segmentation results, but there are also certain problems of missed segmentation and wrong segmentation, as shown in the orange box in Image 3, there were more grape areas that were not segmented, compared with the improved PSPNet model the problem of missed segmentation was slightly more prominent, the red box in image 4 shows that there are parts of the region with a grape leaf background is incorrectly divided into bunches (Figure 10d). The U-Net model has the worst segmentation effect and the problems of missed segmentation and wrong segmentation are relatively prominent because it does not consider the utilization of the multi-scale information in the image. As shown by the orange boxed content in Image 1, Image 3, Image 4 and Image 5, U-Net has a large number of grape regions that are not well segmented or not segmented at all compared to the other two models. In addition, the model in the red box of image 1 mistakenly segmented the wire, and the model in the red boxed area of Images 2 and 4 mistakenly segmented the grape leaves and grape branches (Figure 10e).

Overall, the improved PSPNet network structure was better than other network structures for segmenting grape bunches regions in natural scenes, especially in terms of detail processing.



Note: Image1 represents the image of colored grapes on a cloudy day; Image2 represents the image of white grapes on a cloudy day; Image3 represents the image of grapes under smooth light conditions; Image4 represents the image of grapes under backlight conditions; Image5 represents the image of grapes under normal light.

Figure 10 Segmentation results of different models of grape bunches

4.5 Comparative trials of different grape varieties

Since the collected images of wine grapes contained colored grapes and white grapes, there was a very significant difference in the color of the bunches of these two types of grapes. Colored wine grape bunches usually appeared red or purple, with a distinct difference from the background. White grape bunches usually appeared lime green and pale yellow, and resemble the background more. And the color of the grape bunches surface is the main factor for feature extraction and segmentation. Therefore, in order to further verify the segmentation effect of the improved PSPNet model on different color varieties of grapes, colored and white grapes test datasets were constructed respectively. There were 382 images of colored grapes and 361 images of white grapes. The segmentation results of this model against DeepLab-V3+ and U-Net models for different color varieties of grape bunches are shown in Table 5.

From Table 5, it can be seen that the improved PSPNet model has a better segmentation effect than the U-Net model for both colored and white grapes. In addition, compared with the DeepLab-V3+ model, the model proposed in this study is slightly lower in the segmentation effect of colored grapes but has a significant advantage in the segmentation effect of white grapes. Figure 11 represents the partial segmentation effect of the model in this study for different color varieties of grapes under down-light, back-light and normal conditions. It can be seen that the improved PSPNet model can reduce the complex background interference and has good robustness against environmental variations.

Table 5 Test results of different models for different grape varieties

Different colors varieties of grapes	Number of samples/frame	Model Methodology	IoU/%	PA/%
Colored Grapes	382	Improved PSPNet	87.36	95.44
		DeepLab-V3+	90.13	94.46
		U-Net	86.86	90.22
White Grape	361	Improved PSPNet	87.45	96.08
		DeepLab-V3+	85.72	93.26
		U-Net	86.25	89.54

4.6 Discussion

The accurate segmentation of the grape bunches from their surroundings is a prerequisite for mechanized grape harvesting. In this study, the structure of PSPNet was improved, a pyramidal scene parsing network, to segment grape bunches in natural environments. The experiment results show the improved performance. However, there are also limitations to the proposed method. As shown in the orange-marked boxes in Figure 11, regardless of grapes being colored or white, when the bunches on the grape images are relatively discrete, the model in this study cannot accurately and completely segment the berry regions.

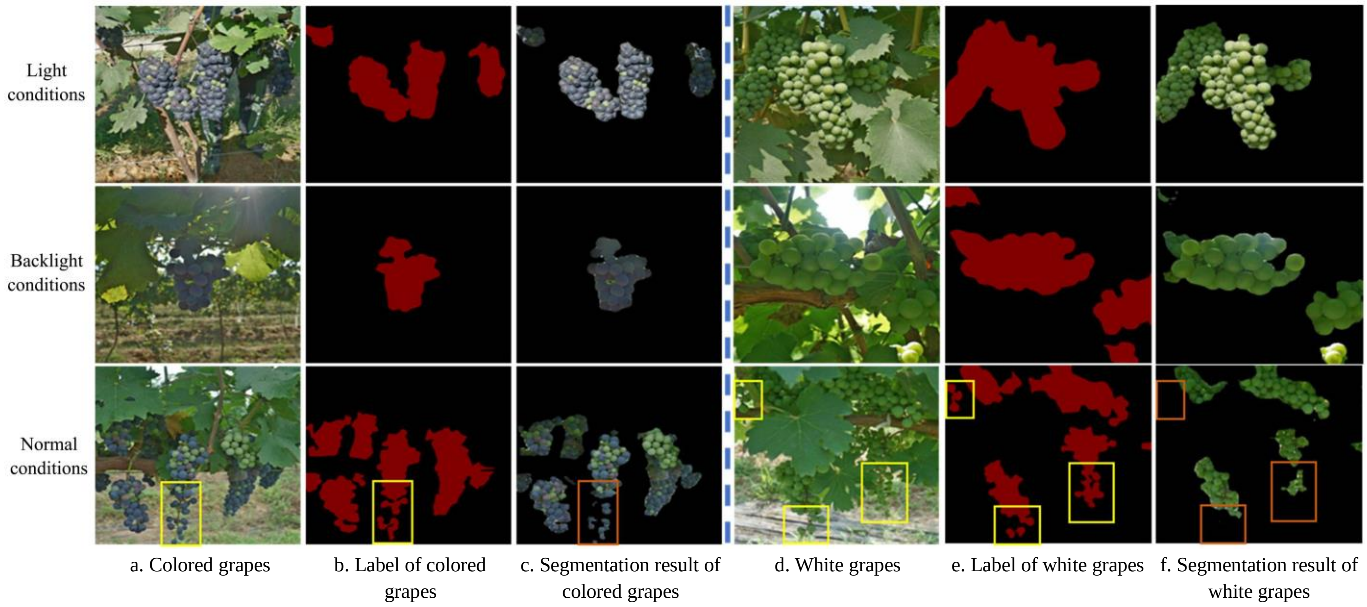


Figure 11 Segmentation results of the improved PSPNet model under different environmental conditions

The reason for this problem is mainly that the sample size of discrete grape bunches is relatively small when constructing the grape bunches segmentation dataset, most grape bunches are continuous throughout, and the model is basically trained with whole bunches segmentation. The network model can be further improved in the future to enhance the feature extraction ability of the model for grape bunches without losing model accuracy. In addition, the grape bunches dataset can also be further improved (e.g. increasing the number of discrete grape bunches samples) to enhance the model’s ability to extract discrete grape bunches features.

5 Conclusions

The extraction of grape bunches areas in the natural field environment is affected by many factors such as light and complex backgrounds, as well as the close-up nature of white grape varieties. In this study, a semantic segmentation method was proposed for grape images based on an improved PSPNet model. This method can effectively partition the grape bunches regions of different varieties, which can provide some technical support for intelligent harvesting in vineyards and also provide some basis for the study of grape bunches-related phenotypes.

The method was based on PSPNet semantic segmentation model and was improved by adding CBAM attention mechanism in Cov1 and Cov5 stages of the backbone feature extraction network ResNet-50 and replacing part of the standard convolution in Cov5 stage with atrous convolution, and fusing multiple feature layers in the enhanced feature extraction stage PSP module to extract more comprehensive contextual information. The experimental results on the collected grape image dataset show that the grape intersection ratio IoU of the improved PSPNet model is 87.42%, the pixel accuracy PA is 95.73%, and the average processing time of a single image is 125 ms, which is generally better than the DeepLab-V3+ and U-Net model, and obtains the best segmentation of grape bunches region in the natural field environments. In addition, by experimenting on a test set of grape images with different color varieties, better segmentation results were obtained for the improved PSPNet under both down-light and backlight conditions, indicating that the improved PSPNet model has good robustness against environmental variations.

Acknowledgements

This study was financially supported by the Key R&D Project of Ningxia Hui Autonomous Region (Grant No. 2019BBF02013) and Guangxi Key R&D Program Project (Grant No. Gui Ke AB21076001). The authors would like to acknowledge Yixue Liu and Yulin Liu for their help in field data collection and the support of the staff at the experimental site.

[References]

[1] Yu Y, Zhang K, Yang L, Zhang D. Fruit detection for strawberry harvesting robot in non-structural environment based on Mask-RCNN. Computers and Electronics in Agriculture, 2019; 163: 104846. doi: 10.1016/ j.compag.2019.06.001.

[2] Xiong J, Liu Z, Lin R, Bu R, He Z L, Yang Z G, Liang C. Green grape detection and picking-point calculation in a night-time natural environment using a charge-coupled device (CCD) vision sensor with artificial illumination. Sensors (Basel, Switzerland), 2018; 18(4): 969. doi: 10.3390/s18040969.

[3] Murillo-Bracamontes E A, Martinez-Rosas M E, Miranda-Velasco M M, Martinez-Reyes H L, Martinez-Sandoval J R, Cervantes-De-Avila H. Implementation of Hough transform for fruit image segmentation. Procedia Engineering, 2012; 35: 230–239. doi: 10.1016/j.proeng.2012.04.185.

[4] Reis M J, Morais R, Peres E, Pereira C, Contente O, Soares S, Valente A, Baptista J, Ferreira P J S, Cruz J B. Automatic detection of bunches of grapes in natural environment from color images. Journal of Applied Logic, 2012; 10(4): 285–290. doi: 10.1016/j.jal.2012.07.004.

[5] Liu S, Whitty M. Automatic grape bunch detection in vineyards with an SVM classifier. Journal of Applied Logic, 2015; 13(4): 643–653. doi: 10.1016/j.jal.2015.06.001.

[6] Pérez-Zavala R, Torres-Torriti M, Cheein F A, Troni G. A pattern recognition strategy for visual grape bunch detection in vineyards. Computers and Electronics in Agriculture, 2018; 151: 136–149. doi: 10.1016/j.compag.2018.05.019.

[7] Cecotti H, Rivera A, Farhadloo M, Pedroza M A. Grape detection with

convolutional neural networks. Expert Systems with Applications, 2020; 159: 113588. doi: 10.1016/j.eswa.2020.113588.

[8] Milella A, Marani R, Petitti A, Reina G. In-field high throughput grapevine phenotyping with a consumer-grade depth camera. Computers and Electronics in Agriculture, 2019; 156: 293–306. doi: 10.1016/j.compag.2018.11.026.

[9] Marani R, Milella A, Petitti A, Reina G. Deep neural networks for grape bunch segmentation in natural images from a consumer-grade camera. Precision Agriculture, 2021; 22(2): 387–413. doi: 10.1007/s11119-020-09736-0.

[10] Dias P A, Tabb A, Medeiros H. Multispecies fruit flower detection using a refined semantic segmentation network. IEEE robotics and automation letters, 2018; 3(4): 3003–3010.

[11] Lin K, Gong L, Huang Y, Liu C, Pan J. Deep learning-based segmentation and quantification of cucumber powdery mildew using convolutional neural network. Frontiers in plant science, 2019; 10: 155. doi: 10.3389/fpls.2019.00155.

[12] Ganchenko V, Starovoitov V, Zheng X. Image semantic segmentation based on high-resolution networks for monitoring agricultural vegetation. In: 2020 22nd International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), IEEE, 2020; pp.264–269. doi: 10.1109/SYNASC51798.2020.0050.

[13] Tassis L M, De Souza J E T, Krohling R A. A deep learning approach combining instance and semantic segmentation to identify diseases and pests of coffee leaves from in-field images. Computers and Electronics in Agriculture, 2021; 186: 106191. doi: 10.1016/j.compag.2021.106191.

[14] Chen S, Zhang K, Zhao Y, Sun Y, Ban W, Chen Y, et al. An approach for rice bacterial leaf streak disease segmentation and disease severity estimation. Agriculture, 2021; 11(5): 420. doi: 10.3390/agriculture11050420.

[15] Esgario J, Castro P, Tassis L M, Krohling R A. An app to assist farmers in the identification of diseases and pests of coffee leaves using deep learning. Information Processing in Agriculture, 2021; In press. doi: 10.1016/j.inpa.2021.01.004.

[16] Kang H, Chen C J S. Fruit detection and segmentation for apple harvesting using visual sensor in orchards. Sensors, 2019; 19(20): 4599. doi: 10.3390/s19204599.

[17] Roy K, Chaudhuri S S, Pramanik S. Deep learning based real-time Industrial framework for rotten and fresh fruit detection using semantic segmentation. Microsystem Technologies, 2021; 27: 3365–3375.

[18] Li J H, Tang Y C, Zou X J, Lin G C, Wang H J. Detection of fruit-bearing branches and localization of litchi clusters for vision-based harvesting robots. IEEE Access, 2020; 8: 117746–117758.

[19] Kestur R, Meduri A, Narasipura O. MangoNet: A deep semantic segmentation architecture for a method to detect and count mangoes in an open orchard. Engineering Applications of Artificial Intelligence, 2019; 77: 59–69. doi: 10.1016/j.engappai.2018.09.011.

[20] Shu B, Mu J, Zhu Y. AMNet: Convolutional neural network embedded with attention mechanism for semantic segmentation. In: Proceedings of the 2019 3rd High Performance Computing and Cluster Technologies Conference, 2019; pp.261–266. doi: 10.1145/3341069.3342988.

[21] Lin C-Y, Chiu Y-C, Ng H F, Shih T K, Lin K-H. Global-and-local context network for semantic segmentation of street view images. Sensors, 2020; 20(10): 2907. doi: 10.3390/s20102907.

[22] Wang Y, Lyu J, Xu L, Gu Y, Zou L, Ma Z. A segmentation method for waxberry image under orchard environment. Scientia Horticulturae, 2020; 266: 109309. doi: 10.1016/j.scienta.2020.109309.

[23] Li Q, Jia W, Sun M, Hou S, Zheng Y. A novel green apple segmentation algorithm based on ensemble U-Net under complex orchard environment. Computers and Electronics in Agriculture, 2021; 180: 105900. doi: 10.1016/j.compag.2020.105900.

[24] Amiri S A, Hassanpour H. A preprocessing approach for image analysis using gamma correction. International Journal of Computer Applications, 2012; 38(12): 38–46.

[25] Woo S, Park J, Lee J-Y, Kweon I S. CBAM: Convolutional block attention module. In: Proceedings of the European conference on computer vision - ECCV 2018, Springer, Cham, 2018; pp.3–19. doi: 10.1007/978-3-030-01234-2_1.

[26] Hesamian M H, Jia W, He X, Kennedy P J. Atrous convolution for binary semantic segmentation of lung nodule. In: ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing, IEEE, 2019; pp.1015–1019. doi: 10.1109/ICASSP.2019.8682220.

[27] Fu H, Meng D, Li W, Wang Y. Bridge Crack Semantic Segmentation Based on Improved Deeplabv3+. Journal of Marine Science and Engineering, 2021; 9(6): 671. doi: 10.3390/jmse9060671.

[28] Chen Y, Li Y, Wang J, Chen W, Zhang X. Remote sensing image ship detection under complex sea conditions based on deep semantic segmentation. Remote Sensing, 2020; 12(4): 625. doi: 10.3390/rs12040625.

[29] Wang W, Fu Y, Dong F, Li F. Semantic segmentation of remote sensing ship image via a convolutional neural networks model. IET Image Processing, 2019; 13(6): 1016–1022.

[30] Huang L, He M, Tan C, Jiang D, Li G, Yu H. Jointly network image processing: multi-task image semantic segmentation of indoor scene based on CNN. IET Image Processing, 2020; 14(15): 3689–3697. doi: 10.1049/iet-ipr.2020.0088.

[31] Pi Y, Nath N D, Behzadan A H. Detection and semantic segmentation of disaster damage in UAV footage. Journal of Computing in Civil Engineering, 2021; 35(2): 04020063. doi: 10.1061/(asce)cp.1943-5487.0000947.