



Article A Size-Grading Method of Antler Mushrooms Using YOLOv5 and PSPNet

Yanqiang Wu¹, Yongbo Sun¹, Shuoqin Zhang², Xia Liu³, Kai Zhou^{1,*} and Jialin Hou^{1,*}

- ¹ College of Mechanical and Electronic Engineering, Shandong Agricultural University, Taian 271018, China
- ² Physical Education College, Shandong Agricultural University, Taian 271018, China
- ³ Shandong Nongfa Mushroom Industry Group Co., Ltd., Dongying 257029, China

* zhoukai2017@sdau.edu.cn (K.Z.); jlhou@sdau.edu.cn (J.H.)

Abstract: Quality grading in antler mushroom industrial production is a labor-intensive operation. For a long time, manual grading has been used for grading, which produces various problems such as insufficient reliability, low production efficiency, and high mushroom body damage. Automatic grading is a problem to be solved urgently for antler mushroom industrial development with increasing labor costs. To solve the problem, this paper deeply integrates the single-stage object detection of YOLOv5 and the semantic segmentation of PSPNet, and proposes a Y-PNet model for real-time object detection and an image segmentation network. This article also proposes an evaluation model for antler mushroom's size, which eliminates subjective judgment and achieves quality grading. Moreover, to meet the needs of efficient and accurate hierarchical detection in the factory, this study uses the lightweight network model to construct a lightweight YOLOv5 single-stage object detection model. The MobileNetV3 network model embedded with a CBAM module is used as the backbone extractor in PSPNet to reduce the model's size and improve the model's efficiency and accuracy for segmentation. Experiments show that the proposed system can perform real-time grading successfully, which can provide instructive and practical references in industry.

Keywords: antler mushroom; deep learning; real time; size grading

1. Introduction

The edible fungus industry is important in China's agriculture and ranks sixth after grain, cotton, oil, fruit, and vegetables in agriculture proportion, which has a strong potential in promoting local economic development. As an edible fungus species, antler mushrooms have a variety of nutrients and pharmacological effects that attract many consumers. With various functions and good taste, antler mushroom is an agricultural product; thus, quality grading is an important part after harvesting. Traditional antler mushroom quality-grading operation uses manual contact grading, which produces a low production efficiency, large grading errors, and high labor training costs. Concurrently, fresh antler mushroom tissue is fragile and can be easily damaged during traditional manual grading, resulting in defects and browning on the antler mushroom surface [1,2]. Therefore, the traditional manual grading method cannot meet the needs of factory production. Size grading by machine vision technology in the factory production is significant in solving the above problems in traditional size grading.

With the rapid development of information technology, machine learning has extracted information from images through a computer simulation of human vision and then analyzed it to guide real production [3,4], which has become one of the hot topics of artificial intelligence research. Applying machine learning to modern agricultural production can effectively promote the development of modern agriculture [5,6] and the efficiency of agricultural production. Arthur et al. [7] proposed a deep neural network classifier for tomato external defect detection, which overcame the challenge of a high variance in low sample defect classes. Jiang et al. [8] used the deep learning method to classify apple



Citation: Wu, Y.; Sun, Y.; Zhang, S.; Liu, X.; Zhou, K.; Hou, J. A Size-Grading Method of Antler Mushrooms Using YOLOv5 and PSPNet. *Agronomy* **2022**, *12*, 2601. https://doi.org/10.3390/ agronomy12112601

Academic Editors: Hao Yang, Xiuliang Jin, Zhenhai Li, Changping Huang and Dameng Yin

Received: 21 September 2022 Accepted: 19 October 2022 Published: 23 October 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). images, studied and evaluated the deep neural networks with different convolution layers and different number of neurons, and proposed a method to detect apple fruit infection and timely prevent further infection caused by environmental factors. Xu et al. [9] used a CNN model to automatically classify five kinds of corn seeds, which was significantly better than the traditional manual selection, and proposed a fast method for corn seed classification combining machine vision and deep learning. The above research shows that machine learning has a relatively wide range of applications in the agricultural field and has achieved good results.

Since 1991, in the field of edible mushrooms, scholars have been trying to apply deep learning methods in edible mushroom maturity detection and quality grading to change the hindered development of the edible mushroom industry [10,11]. Vooren et al. [10] used image analysis technology to classify mushrooms according to their length, width, and other features and tested 460 sets of experimental images. Experiments showed that the recognition rate reached 80%. Heinemann et al. [12] used the watershed method, the smart operator method, and the closed operator method to classify twin mushrooms and developed an automatic size-sorting system for fresh twin mushrooms. V1Zhányó et al. [13] used imaging methods to achieve the nondestructive detection of mushrooms according to features such as color, freshness, disease, size, and shape characteristics. Chen et al. [14] studied an evaluation method to classify dried shiitake mushrooms according to their appearance features, such as size, color, and damage degree, and developed an automatic grading system to automatically classify dried shiitake mushroom by their character. A total of 250 dried shiitake mushrooms were detected during the experiments, and results showed that the accuracy rate was 97.6%. Yu et al. [15] obtained mushroom location information on a cultivation bed by applying a localization algorithm and outline description of mushrooms, which provided a theoretical basis for image processing. Li et al. [16] used computer vision technology to detect different degrees of damage in shiitake mushrooms, and experiments showed that the overall recognition rate of the recognition technology was 96.5%. Concha Meyer et al. [17] developed a system for estimating mushroom volume using optical imaging technology. The measured mushroom volume obtained by this system was similar to the real mushroom volume, which had certain guiding importance in the evaluation and calculation of mushroom volume. In addition, Lu et al. [18] proposed a system for image measurement based on YOLOv3 [19], which improved the local accuracy through a position correction algorithm and used a bounding box to measure the size of mushrooms to obtain the growth data of mushrooms cultivated in greenhouses. Jiang et al. [20] developed a system for online quality detection and sorting for fresh button mushrooms using machine vision. The system used the watershed algorithm, the speededup robust features (SURF) algorithm, and the conversion of RGB images to Lab images to obtain the size, defects, and browning degree of mushrooms. In an experiment with 100 mushrooms, the average sorting accuracy was 96.45%, achieving quality detection and classification results. Existing research has thus improved and upgraded neural networks in many ways. The classification ability of edible fungi has been improved, and a foundation has been set to apply computer vision technology to edible fungi classification.

To solve the problems produced by the traditional size grading of antler mushrooms, based on real-time segmentation technology [21], this paper proposes a real-time object detection and image segmentation network (Y-PNet) combining YOLOv5 [22] and PSP-Net [23], which realizes the detection, marking, and segmentation of antler mushroom. Y-PNet first reads and detects video through YOLOv5. When the image is detected, PSPNet is immediately called for a pixel-level contour segmentation, which realizes the purpose of real-time segmentation. In this paper, the Y-PNet real-time object detection and image segmentation algorithm is used to build the size-grading system, which realizes the noncontact size-grade detection of antler mushrooms.

The primary results of this study include the following:

(1) A real-time object detection and image segmentation network (Y-PNet) is proposed combining YOLOv5 and PSPNet and a detection system for real-time grading is established to create a noncontact operation method during the production process.

(2) The CSPDarknet backbone extraction network of YOLOv5 is replaced by MobileNetV3 [24] and establishes a lightweight real-time detection network. The network can reduce the number of parameters in reasoning process, improve the performance of the algorithm and the efficiency of real-time detection, and solve the problem of missed detection.

(3) In order to make the segmentation ability more efficient and improve the accuracy of the feature extraction of PSPNet, CBAM [25] is added after stages 1, 2, 3, and 4 of MobileNetV3. Then, the original Resnet50 network of PSPNet is replaced with the modified MobileNetV3 to build a lightweight semantic segmentation network.

2. Size Classification Algorithm

2.1. YOLOv5 Network Model

YOLOv5 is a single-stage object detection algorithm that has the advantages of a high detection accuracy and fast running speed. Its network includes four models: YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. The parameters and volumes of the four models increase accordingly. The primary difference between these models is the number of feature extraction modules and convolution kernels included at specific locations in the network. The YOLOv5s model uses the smallest network depth and feature map dimension as possible to ensure a small weight file and fast detection speed. While ensuring the accuracy of object detection, the model is conducive to operation on mobile devices to achieve real-time object detection. The basic YOLOv5 architecture is mainly based on YOLOv4 [26], given in the PyTorch framework. Figure 1 shows the network structure of YOLOv5.



Figure 1. YOLOv5 network structure.

The network structure of YOLOv5 primarily consists of four parts: input, backbone, neck, and head. The input primarily includes three parts: mosaic data augmentation, image size processing, and adaptive anchor box calculation. During mosaic data enhancement, four pictures are combined to enrich the background of the picture. During picture size processing, the smallest black boxes are adaptively added to the original images of different lengths and widths and uniformly scale them to a standard size, leaving the aspect ratio of the original images unchanged. During adaptive anchor box calculation, the output prediction box is compared with the real box based on the initial anchor box; the difference is calculated and then updated in reverse and the parameters are continuously iterated to obtain the most suitable one.

The backbone mainly contains CSP1_X with a residual structure, Focus, SPP, and CBL modules. Because the backbone network is deep, when backpropagating between layers, using CSP1_X with a residual structure can enhance the gradient value and effectively prevent the gradient disappearance caused by a deepening network, and the obtained feature granularity is finer. The focus module slices the image, expands the input channel by 4 times, and obtains the undersampling feature map after one convolution. Concurrently, the number of calculations is reduced, and the speed is improved. SPP achieves the fusion of local features and global feather map level. CBL is a convolutional module used to obtain features.

The neck uses the structure of combining FPN and PAN; combines the conventional FPN layer with the bottom-up feature pyramid; fuses the extracted semantic features and position features; and fuses the features of the backbone layer and the detection layer. The CSP2_X structure strengthens the network feature fusion ability, enabling the model to obtain richer feature information.

The head outputs a vector with the class probability of the target object, the object score and the object bounding box position. The detection network consists of three detection layers. Feature maps of different sizes are used to detect target objects of different sizes. Each detection layer outputs the corresponding vector and finally generates the predicted bounding box and category of the object in the original image and labels it.

2.2. PSPNet Network Model

The core idea of PSPNet (pyramid scene parsing network) is to use the global pyramid pooling module to divide the acquired feature layers into grids of different sizes, and average pooling is performed inside each grid. The scene analysis network built in this way can aggregate the context information of different regions to solve the problem of category confusion and discontinuity in the prediction results caused by the lack of utilization of the global scene in ordinary FCN, thus markedly improving the ability of the network to analyze the scene.

The pyramid pooling scale fusion structure is shown in Figure 2. After the network model obtains a feature map, it first goes through a network for feature extraction. The original author of this part used the ResNet network model of atrous convolutions [27], and the role of atrous convolution was primarily to increase the receptive field and obtain sufficient global information on the target. Then, the extracted features are input to the spatial pyramid pooling module, and four pooling kernels of different scales, such as 1×1 , 2×2 , 3×3 , and 6×6 , are used for average pooling. Then, to maintain the weights of the global features, the number of channels is reduced to 1/4 of the original through a 1×1 convolution layer, and all pooled feature maps are upsampled to the preinput size to connect the input feature maps. Then, the pyramid pooling global feature map with twice the number of channels can be obtained. The process of combining features is actually the process of merging the shallow features of the target with the deep features processed and output by the feature extraction network, the process of fully considering the context information. Finally, the 1×1 Conv performs the fusion and dimensionality reduction of multiscale information, and then, the final feature layer is constructed. The structure of the pyramid pooling scale fusion is pooled at multiple scales, which improves the feature information of each scale in the final feature map, making it easier for the network to capture the global information in the image. The algorithm proposed fully solves the problem that the FCN-based semantic segmentation model lacks a suitable strategy to use the category cues in the global scene, and the segmentation results have contextual relationship mismatch, category confusion, and an inability to pay attention to insignificant targets [23].



Figure 2. PSPNet network structure.

2.3. Arithmetic Optimization

2.3.1. Improvement of Backbone Network for Feature Extraction

Considering that the industrial cultivation of antler mushrooms must ensure a highefficiency fruiting and that the images collected by industrial cameras are of high resolution, the primary feature extraction networks of the original YOLOv5 algorithm and PSPNet algorithm, are CSPDarknet and ResNet, respectively. The network model has many parameters and calculations, and exhibits a high complexity, and is thus not suitable for real-time detection and model deployment [28]. When the transmission speed of the production line cannot match the detection speed, missed detections may also occur, which will cause the detection results to deviate from the real results, resulting in inaccurate detection.

Taking ResNet as an example, its primary goal is to extract features by first reducing, then convoluting, and finally increasing the dimension. ResNet typically solves the problem of information loss and gradient disappearance of traditional convolutional neural networks or fully connected networks during information transmission to ensure that the deep network can be trained smoothly. ResNet introduces diversity into the receptive field through skip connection. However, the high-dimensional features become increasingly abstract after several feature extractions. Although this process is beneficial for object classification, too many model parameters are required, which leads to information loss and a loss of accuracy in the segmentation results.

Aiming at the above problems, the MobileNet network model was introduced into the backbone feature extraction network of the traditional YOLOv5 algorithm and PSP-Net algorithm. This model is a lightweight deep neural network proposed by Google for embedded devices such as mobile terminals. It is the result of a comprehensive consideration of training accuracy and network model performance, and its core idea is the separable convolution operation [29]. MobileNetV3 in this series of models was built upon the first two generations of V1 and V2, and includes new improvements, achieving excellent performance and speed among existing lightweight models. MobileNetV3 is mainly improved from MnasNet [30], and the main improvements include the following aspects:

(1) A redesigned time-consuming layer structure, including the reduction of the number of convolution kernels in the first layer from 32 to 16 and the last-stage streamlining, which ensures the accuracy and improves the speed of the operation, saving 14% of the total inference time;

(2) The NetAdapt algorithm, which obtains the optimal number of filters used by the expansion layer and the optimal number of channels output by the bottleneck layer, and automatically simplifies the pretraining model while maintaining a high accuracy;

(3) An inherited depthwise separable convolution of V1 and the residual structure with the linear bottleneck of V2;

(4) In the structure containing SE, the number of channels in the expansion layer is reduced by a quarter, which not only improves the accuracy, but also saves time;

(5) Relu6 is replaced by *h-swish* (*x*). Formula (1) is given below;

(6) For the SE module, Relu6(x + 3)/6 is used to approximate a sigmoid.

The inverse residual structure with a linear bottleneck inherited from V2 by MobileNetV3 is exactly opposite to that of ResNet. During feature extraction, this method first increases the dimension, then perform convolution, and finally perform a dimensionality reduction operation, which guarantees that the detailed features of the target are not lost. Concurrently, a deep convolution kernel is only performed on one channel of the feature layer; thus, the number of calculations are not too high:

$$h\text{-swish}(x) = x \frac{ReLU6(x+3)}{6} \tag{1}$$

The network structure of MobileNetV3 is shown in Figure 3.





2.3.2. Introduction of the CBAM Module

Bottled antler mushrooms grow randomly, even in the same matrix. Therefore, more details are required when using antler mushrooms as detection objects. The traditional PSP-Net semantic segmentation simply uses atrous convolution to increase the receptive field to downsample the object, which neglects some features of the detected object, resulting in incomplete image partial details and making the segmentation edges rough. Therefore, a lightweight attention mechanism (CBAM [25]) module that combines spatial and channel dimensions was embedded in the improved PSPNet semantic segmentation algorithm to improve the computation speed of the algorithm and the contour segmentation accuracy of antler mushrooms.

A CBAM (convolutional block attention module) is a lightweight convolutional attention module that sequentially derives attention maps along two independent dimensions of channel and space from a given intermediate feature map. Then, the attention map is multiplied by the input feature map for adaptive feature refinement. The structure of the CBAM is shown in Figure 4. Because the CBAM is a widely used lightweight attention module, it can be easily embedded into the basic CNN for end-to-end training, and only a small amount of computation is added. To calculate channel attention effectively, it is necessary to compress the spatial dimension of the input feature map. To aggregate spatial information, average pooling is typically used, but maximum pooling can collect unique object feature information and can infer more detailed channel attention. Therefore, the features of both average pooling and max pooling are used in the spatial information aggregation. The structure of the channel attention mechanism is shown in Figure 4, and the final channel attention mapping expression M_c is shown in Formula (2):

$$M_{C}(F) = \sigma(MLP(AvgPool(F)) + NLP(MaxPool(F)))$$

= $\sigma(W_{1}(W_{0}(F_{avg}^{c})) + W_{1}(W_{0}(F_{max}^{c})))$ (2)

where F_{avg}^c and F_{max}^c represent the average pooling feature and max pooling feature, respectively. These two descriptors are then transformed to a shared network to produce the proposed channel attention map M_c . The shared network consists of a multilayer perceptron (MLP) with one hidden layer. To reduce the number of parameters, the hidden layer size is set to $R/C = r \times 1 \times 1$, where R is the drop rate. After applying the shared network to each descriptor, the output feature vectors are merged using elementwise summation. \otimes represents the sigmoid function. F' is obtained by element multiplication of F and Mc(F), where F' is the weighted result of F through the channel attention mechanism. The formula is shown in Formula (3):

$$F' = M_C(F) \otimes F \tag{3}$$

Spatial attention focuses on the question of where the most informative parts are located. To compute spatial attention, average pooling and max pooling are performed along the channel axis. Then, they are integrated into an efficient feature descriptor and applied to a convolutional layer to generate a spatial attention map of size $R \times H \times W$, where R, H, and W are the number of channels, height, and width of the input feature map, respectively. This

process encodes locations that require attention or suppression, which is a supplement to the channel attention. The structure of the spatial attention mechanism is shown in Figure 4. The final output spatial attention mapping expression M_S is shown in Formula (4):

$$M_{S}(F) = \sigma(f^{7 \times 7}([AvgPool(F); MaxPool(F)])) = \sigma(f^{7 \times 7}([F^{s}_{avg}; F^{s}_{max}]))$$
(4)

where the channel information of one feature map is aggregated using two pooling operations to generate two 2D maps: the size of F_{avg}^s is $1 \times H \times W$, and the size of F_{max}^s is $1 \times H \times W$. σ is the sigmoid function, and $f^{7 \times 7}$ is a convolution operation with a filter size of 7×7 .

In this paper, we introduced MobileNetV3-large into PSPNet in order to retain a good detection accuracy after the introduction of the lightweight network [24]. In addition, the improved PSPNet tends to ignore the feature maps of different levels in the feature extraction stage, resulting in the loss of some image details and poor segmentation effect. We added a CBAM attention mechanism to a stage of MobileNetV3-large, and the improved MobileNetV3-large network structure is shown in Table 1.

Table 1. Improved architecture of MobileNetV3- large.

Input	Operator	Exp Size	#out	SE	NL	S
$512^{2} \times 3$	conv2d	-	16	-	HS	2
$256^2 \times 16$	bneck, 3×3	16	16	-	RE	1
$256^2 \times 16$	CBAM					
$256^{2} \times 16$	bneck, 3×3	64	24	-	RE	2
$128^2 \times 24$	bneck, 3×3	72	24	-	RE	1
$128^2 \times 24$	CBAM					
$128^2 \times 24$	bneck, 3×3	72	40	\checkmark	RE	2
$64^2 \times 40$	bneck, 3×3	120	40	\checkmark	RE	1
$64^2 \times 40$	bneck, 3×3	120	40	\checkmark	RE	1
$64^2 \times 40$	CBAM					
$64^2 \times 40$	bneck, 3×3	240	80	-	HS	2
$32^2 \times 80$	bneck, 3×3	200	80	-	HS	1
$32^2 \times 80$	bneck, 3×3	184	80	-	HS	1
$32^{2} \times 80$	$bneck, 3 \times 3$	184	80	-	HS	1
$32^2 \times 80$	bneck, 3×3	480	112	\checkmark	HS	1
$32^{2} \times 112$	$bneck, 3 \times 3$	672	112	\checkmark	HS	1
$32^{2} \times 112$	bneck, 3×3	672	160	\checkmark	HS	2
$32^2 \times 160$	CBAM					
$32^{2} \times 160$	bneck, 3×3	672	160	\checkmark	HS	1
$16^2 \times 160$	$bneck, 3 \times 3$	960	160	\checkmark	HS	1

In this paper, in order to replace the Resnet network with MobileNetV3 in PSP-Net and ensure a high detection accuracy, MobileNetV3-large with higher accuracy than MobileNetV3-small was added [24]. Then, to solve the problem that the improved PSPNet tends to ignore the feature maps of different levels in the feature extraction stage, which leads to the loss of some image details and the poor segmentation effect, we improved MobileNetV3-large by adding a CBAM attention mechanism, and the improved network structure of MobileNetV3-large is shown in Table 1. MobileNetV3 was divided into five stages by downsampling multiple times. As shown in Table 1, we added a CBAM attention mechanism behind a stage on the 1st, 2nd, 3rd and 4th layers of MobileNetV3-large to improve the feature extraction ability of PSPNet. Moreover, the first column, Input, in Table 1 represents shape changes for each feature layer of MobileNetV3. The second column, Operator, represents the block structure that each feature layer experiences. In MobileNetV3, feature extraction goes through many bneck structures. The exp size represents the number of channels in bneck after the inverse residual structure rises. *#out* indicates the number of channels in a feature layer when input into bneck. SE indicates whether to add the SE

attention mechanism. NL represents the type of the activation function, HS represents h-swish, and RE represents Relu. *s* represents the step size used for each block structure.



Figure 4. CBAM network structure and channel and spatial attention structure.

2.4. Y-PNet Algorithm for Real-Time Object Detection and Image Segmentation

The original PSPNet model exhibits a high computational complexity, which is not conducive to mobile platform deployment and cannot perform real-time segmentation. This paper comprehensively considered the characteristics of YOLOv5 real-time detection and PSPNet image segmentation. The two algorithms, YOLOv5 and PSPNet, were improved and fused deeply, and a real-time object detection and image segmentation network Y-PNet was constructed whose network structure is shown in Figure 5. The improvements over the original YOLOv5 and PSPNet are as follows:

(1) The CSPDarknet backbone extraction network of YOLOv5 was replaced by MobileNetV3 [24], and we established a lightweight real-time detection network. The network could reduce the number of parameters in the reasoning process, improve the performance of the algorithm and the efficiency of real-time detection, and solve the problem of missed detection.

(2) In order to make the segmentation ability more efficient and improve the accuracy of the feature extraction of PSPNet, a CBAM [25] was added after stages 1, 2, 3, and 4 of MobileNetV3. Then, the original Resnet50 network of PSPNet was replaced with the modified MobileNetV3 to build a lightweight semantic segmentation network.

By improving the traditional model to ensure that the Y-PNet real-time object detection and image segmentation algorithm had a high detection rate, the accuracy of the detection results met the high-efficiency and high-quality requirements of assembly-line antler mushroom size classification. The network detection process was as follows:

The Y-PNet real-time object detection and image segmentation network obtained the captured real-time video via a camera's IP address. Once the camera had obtained the real-time video of the antler mushroom transported by the factory assembly line, the video image was transmitted to the network. First, the real-time video is processed by the Y-PNet first-stage network, YOLOv5. When the network detected an antler mushroom, it automatically identified the antler mushroom and obtained the location and coordinates of

the antler mushroom in the video. Then, when the mushroom reached a fixed position, the obtained information was passed to the Y-PNet second-stage network, the PSPNet semantic segmentation algorithm. At this stage, the contour of the antler mushroom image was segmented, and the overall pixel area of the antler mushroom was calculated to represent the real area of the antler mushroom, which was basic for grading.

The antler mushrooms of different sizes provided by the antler mushroom production factory were divided into three grades: I, II, and III. The algorithm in this paper was used to detect the pixel area corresponding to the three different grades of antler mushrooms at a fixed distance. The size evaluation criterion for antler mushrooms corresponding to the grading network (Y-PNet) was obtained: when the pixel area of the entire mushroom body was greater than or equal to 50,000, the antler mushroom maturity was defined as level I; when the pixel area was between 30,000 and 50,000, the maturity was defined as level II; and the pixel area below 30,000 was defined as class III. With this evaluation criterion, the mushroom was segmented, the average area of the bottled mushroom was output, and different mushroom size grades were defined according to the detected pixel average area size.



Figure 5. Y-PNet network structure.

3. Dataset Construction and Evaluation Criterion of Algorithm

3.1. Dataset Processing and Training

The image data used in this paper were collected from the antler mushroom production base of Shandong Nongfa Fungi Industry Group Ltd, Dongying City, Shandong Province, China. Photos were taken on 11 March 2022 using a Hikvision MV-CE200-10GC industrial camera. Bottled antler mushrooms were shot at a fixed distance of 0.3–0.4 m from the camera, and the images were saved in JPG format with a resolution of 5472 pixels \times 3672 pixels. A total of 2359 pictures of bottled antler mushrooms of different sizes were taken. To reduce the sensitivity of the network model to the image and the problem of unbalanced samples, we used flipping, rotating, cropping, mirroring, and other methods to enhance the data and improve the model generalizability, and we finally obtained 8351 sample images of bottled antler mushrooms.

The LabelMe tool was used to label each image manually. To train the YOLOv5 object detection algorithm in the first stage, the rectangular annotation tool was used to mark the dataset of the bottle body and antler mushroom body. To train the second stage, the PSPNet semantic segmentation network model, the polygon annotation tool was used to label the dataset, and the labels were divided into the antler mushroom body and background. Finally, we divided the two sets into a training set, validation set, and test set at a ratio of 7:2:1. The training set was used for model training, the validation set was used for real-time feedback on the model training during the training process, and the test set was

used to evaluate the model performance. The number of samples of each dataset is shown in Table 2, and the annotation example diagram of the dataset is shown in Figure 6.

Natural Mada	Label Catagoria	Number of Samples				
Network Widde	Label Category	Training Set	Validation Set	Test Set	Total	
YOLOv5 stage	Antler mushroom	5846	1670	835	8351	
	bottle	5846	1670	835	8351	
PSPNet stage	Antler mushroom	5846	1670	835	8351	
	background	5846	1670	835	8351	

Table 2. Dataset related information.



Figure 6. Example diagram of the dataset.

The computer configuration of the experimental platform was as follows: the CPU was an Intel Core i7-8700, the operating system was Windows 10, the Python version was 3.8.8, and the deep learning framework and version were PyTorch 1.10.1. We set the training epochs to 300, the learning rate to 0.01, and the batch size to 16. During training with the mosaic data enhancement method, images were randomly cropped and scaled, and then randomly arranged and spliced to form a single image. While enriching the dataset, a small sample target was added to improve the training speed of the network. Concurrently, the training set and the validation set were used for cross-validation after each iteration so that problems in the training process could be identified and solved quickly to reduce computation time. Finally, the model performance was tested using the test set.

3.2. Different Evaluation Criteria

To verify the effectiveness of the proposed method, different evaluation criteria were used for the two deep learning algorithm models. First, for the YOLOv5 single-stage object detection algorithm, the precision, recall, and mean average precision (mAP) were used as evaluation criteria to evaluate the model, which included the following three aspects [31]:

(1) Precision is the proportion of correctly classified samples to all samples:

$$Precision = \frac{IP}{TP + FP}$$
(5)

(2) Recall is the proportion of correctly classified positive classes in all positive classes:

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

(3) The mean average precision (mAP) is the mean of the average precision of all categories in the dataset:

$$mAP = \frac{\sum_{i=1}^{m} AP_i}{m}$$
(7)

The number of true positives (TP) is the number of positive samples that are detected correctly; the number of true negatives (TN) is the number of negative samples that are detected correctly; the number of false-positives (FP) is the number of positive samples that are detected incorrectly; the number of false negatives (FN) is the number of false negative samples that are detected; and m is the number of samples in the test set.

For the PSPNet semantic segmentation algorithm, the model performance was evaluated from four perspectives: mean pixel accuracy (MPA), mean intersection over union (MIoU), pixel accuracy (PA), and training time. The relevant formulae are as follows:

(1) MPA is the average pixel accuracy of all classes in the image, and Formula (8) is as follows:

$$MPA = \frac{\sum_{i=1}^{k} \frac{p_{ii}}{\sum_{i=0}^{k} p_{ij}}}{k}$$
(8)

(2) The mean intersection over union (MIoU) is the average of the accumulated IoU values of each class of image pixels, as shown in Formula (9):

$$MIoU = \frac{\sum_{i=0}^{k} \frac{p_{ii}}{\sum_{j=0}^{k} p_{ij} + \sum_{j=0}^{k} p_{ji} - p_{ii}}}{k+1}$$
(9)

where *k* is the number of classes of pixels; p_{ii} is the number of pixels that are real and predicted to be class *i*; p_{ij} is the number of pixels that are real class *i* but predicted to be class *j*; and p_{ji} is the number of pixels that are real class *j* but predicted to be class *i*.

(3) The pixel accuracy (PA) is the ratio of correctly marked pixels to the total pixels and Formula (10) is shown as follows:

$$PA = \frac{\sum_{i=0}^{k} p_{ii}}{\sum_{i=0}^{k} \sum_{j=0}^{k} p_{ij}}$$
(10)

(4) The training time is the time from the start to the end of the training, and the running speed of the model can be evaluated and calculated in minutes (min).

4. Results and Analysis

4.1. Results of YOLOv5 Algorithm

To verify the effectiveness of embedding the MobileNetV3 lightweight network model in the YOLOv5 under the same environment configuration, the improved YOLOv5 algorithm was trained and predicted in the same dataset with the current mainstream neural networks ResNet101, ResNet50, and traditional YOLOv5, while ensuring the same basic training parameters, including the number of epochs, batch size, and learning rate as described above. To describe the training results of each model, the precision (P), recall (R), mean average precision (mAP), detection rate (s/image), and model size were recorded and counted separately. The experimental results are shown in Table 3:

Model	P (%)	R (%)	mAP@0.5 (%)	Speed (s/Image)	Params	
Traditional YOLOv5s	76.8	84.0	84.3	0.17	7.2	
YOLOv5s +MobileNetV3-large	92.1	87.5	92.4	0.09	5.4	
YOLOv5s +MobileNetV3-small	89.3	83.5	88.1	0.05	3.2	
YOLOv5s +ResNet101	88.6	93.7	87.4	0.14	42	
YOLOv5s +ResNet50	85.4	91.3	88.6	0.11	23	

Table 3. Experimental results of Y-PNet's YOLOv5 algorithm.

Table 3 shows that in the antler mushroom size classification task, the YOLOv5 model in the first stage of Y-PNet achieved excellent results with the MobileNetV3 lightweight network as the backbone feature extraction network. The precision of the YOLOv5s + MobileNetV3-large network was 92.1%, which was higher than that of ResNet101, the size of the model parameters was much lower than that of ResNet series models, and the detection speed was also higher than this network model, which demonstrated that MobileNetV3 achieved good network performance. The proposed algorithm was based on an improvement made by the original YOLOv5 algorithm embedded in the MobileNetV3large lightweight network model. The precision was nearly 16% higher than that of the original YOLOv5 algorithm detection result with a higher detection accuracy, and the network model size was markedly lower compared with traditional models such as ResNet50 and ResNet101. The achieved precision thus met the high-efficiency requirements of the assembly line for antler mushroom size grading while ensuring accuracy.

In addition, to feel the detection difference more intuitively between the improved algorithm and YOLOv5, dense antler mushroom images were selected for detection and comparison, and the results are shown in Figure 7a,b. The detection results showed that the network in the first stage had better detection accuracy for occluded targets. Therefore, the improved Y-PNet algorithm proposed in this paper markedly increased image processing speed while maintaining a high accuracy for the detection of occluded targets.





(a) Y-PNet's detection results

(**b**) YOLOv5's detection results

Figure 7. Comparison of Y-PNet and YOLOv5's detection results.

4.2. Results of PSPNet Algorithm

4.2.1. Results of Model Comparison

To verify the effectiveness of adding the improved MobileLabv3 lightweight network model to the original PSPNet semantic segmentation algorithm, mainstream semantic segmentation models such as the improved PSPNet algorithm in the second stage of the proposed Y-PNet, PSPNet, Deeplabv3+, and a PSPNet that only adds the nonimproved

MobileLabv3 lightweight network model were trained and predicted on the same dataset. The same basic parameters were set during the experiment to ensure the accuracy of the experimental results. In this experiment, the overall performance of different models was evaluated using four evaluation indicators: mean intersection over union (MIoU), mean pixel accuracy (MPA), pixel accuracy (PA), and training time. Results are shown in Table 4.

Table 4. Experimental results of Y-PNet's PSPnet algorithm.

Model	MIoU (%)	PA (%)	MPA (%)	Time (min)
PSPNet-ResNet50	74.82	85.61	84.91	3251
Deeplabv3+-Xception	81.65	93.14	92.39	2781
PSPNet-MobileNetV3	80.33	92.24	91.72	2379
Ours	93.73	96.35	95.47	2142

Table 4 shows that after adding the lightweight MobileNetV3 network model to the traditional PSPNet semantic segmentation algorithm, the computational efficiency markedly increased, even above that of the traditional Deeplabv3+ semantic segmentation. Therefore, it was feasible to reduce the number of parameters by adding a lightweight network. Furthermore, because an SE channel attention was introduced into the MobileNetV3 lightweight network, the results showed that the segmentation accuracy of the antler mushroom was marginally improved. However, certain deficiencies were present compared with the segmentation results of the Deeplabv3+ network. From the final improved model, it can be seen that after embedding the CBAM module into a stage in the MobileNetV3 network, the accuracy of the segmentation was improved to a certain extent, and the segmentation accuracy was significantly better than that of the traditional Deeplabv3+ network segmentation model, so that the PA and MPA, respectively, reached 96.35% and 95.47%. These results showed that the CBAM lightweight attention mechanism exhibited a good versatility and low memory consumption while enhancing the feature extraction ability of the model. These results also showed that the improvement of the traditional PSPNet semantic segmentation algorithm in this paper was effective.

In addition, to describe the convergence speed of different models more intuitively and fairly, each model in this experiment used the cross-entropy loss function during training and performed a complete training process. The trend of the loss of each model in the dataset of this study is shown in Figure 8. Figure 8 shows that the improved PSPNet semantic segmentation model reached convergence in the 34th round of training, which was faster and more stable than for the other three models. As a result, the loss of the proposed algorithm decreased most quickly and was the most stable of the tested algorithms, demonstrating that the proposed algorithm was more efficient. A comparison of the experimental results showed that the proposed algorithm performed better than the other three network segmentation models in terms of training time, detection accuracy, and detection efficiency, which showed that the proposed algorithm was more suitable for antler mushroom size classification production operations with a high resolution and more detailed features.



Figure 8. Comparison of the change in trend of the loss value of different models.

4.2.2. Results of Image Segmentation

The proposed and other algorithms were used to segment the same set of antler mushroom test sets, and a real-time pixel-level image annotation was performed to compare the visualization results, as shown in Figure 9. Segmentation results clearly showed that the proposed algorithm model had a higher segmentation accuracy and efficiency. Therefore, the Y-PNet real-time object detection and image segmentation network proposed in this study is a high-precision and high-efficiency network model.



Figure 9. Comparison of segmentation effect of different models.

4.3. Experimental Platform Construction and Experimental Site

To facilitate the experimental operation and verify the stability and feasibility of the system, it was necessary to build an experimental platform and an intelligent management and control system for the size classification detection of antler mushrooms. The platform and system are shown in Figures 10 and 11, respectively. The experimental platform controlled the rotation speed and forward and reverse rotation of the AC stepper motor by sending different pulse frequencies to the driver through the Altai motion control card and changing the level of the direction signal through the upper computer control system, thereby changing the transport speed and the transportation direction. The platform provided a basis for the subsequent scene simulation experiment of grading and damage

detection of antler mushrooms combined with the size-grading detection system. The grading detection portion of this device integrated machine vision technology, deep learning technology, and other detection methods, and could detect multiple indicators concurrently, conducting a more comprehensive size classification detection of antler mushrooms.



Figure 10. Experimental platform for the size classification detection of antler mushrooms.

To improve the management and control of the system, we designed an intelligent management and control system for size classification detection of antler mushrooms, which integrated data management control and production equipment management control for antler mushrooms to integrate the deep learning detection model, machine vision algorithm, and related hardware, including a stepper motor driver, stepper motor, camera, and transmission platform. This system achieved the function of antler mushroom size classification. Based on PyQt5, we designed and developed an intelligent management and control system for the size classification detection of antler mushrooms, which provided camera control, adjustable conveyor speed, and automatic dual-grading detection of antler mushrooms and communicated with other systems and higher-level systems via the network database.

The intelligent management and control system for size classification detection of antler mushrooms primarily included three modules: real-time detection video display, size classification of antler mushroom maturity, and database interaction. After the initial real-time recognition of bottle-grown antler mushrooms by constructing the new first stage of the Y-PNet network, the image was transferred to the second stage of the Y-PNet network. The average pixel area of the antler mushroom in the bottle was calculated by image segmentation to achieve the size classification of the antler mushroom. Concurrently, the data information obtained in real time could be imported into an SQL server database, which could provide multiuser acquisition of data information.



Figure 11. Intelligent control system for the size classification and detection of antler mushrooms

To verify the accuracy and efficiency of the Y-PNet antler mushroom size real-time object detection and image segmentation network, this paper used the antler mushroom size classification intelligent management and control system to conduct a size classification experiment for 150 selected bottles of antler mushrooms. Because the manual detection process primarily relies on subjective judgment to classify the size of antler mushrooms and the antler mushroom size classification assembly line is generally operated by one person and one station, professionally trained staff were selected to participate in this comparative experiment to ensure the reliability and accuracy of the experiment, and three experiments were performed. The average detection statistics obtained from the experiments are shown in Table 5, which shows that the average detection accuracy of the intelligent management and control system was 94%, which was approximately 18% higher than that of manual detection. Compared with manual labor, the proposed system reduced the grading and screening time by one-sixth. Therefore, the comparison experiment indicated that the intelligent management and control system for the size classification detection of antler mushrooms achieved high-efficiency and high-quality detection results while satisfying nondestructive testing. Therefore, this system can be applied to the classification production process of antler mushrooms to increase efficiency, reduce errors, and mitigate damage that occurs during traditional manual size classification detection, which can effectively improve the efficiency of the classification production process and reduce production cost.

Table 5. Experimental comparison results.

Detection Type	Detection Accuracy (%)	Detection Completion Time (s)	Detection Damage Rate (%)
Intelligent management and control system	94	61	0
Labor (1 person)	76	352	19

5. Conclusions

To solve size grading, this paper used MobileNetV3 as the backbone extraction network to improve YOLOv5's results. The detection accuracy and speed were increased by 15% and 47%, respectively. The MobileNetV3-embedded CBAM was used to improve PSPNet, which improved the pixel accuracy to 96.35% and reduced the training time to 34%. In addition, by combining an improved YOLOv5 network with an improved PSPNet, Y-PNet, a real-time object detection and image segmentation network, was proposed based on machine vision to realize the real-time detection and segmentation of antler mushroom images. Finally, this study built an experimental platform for the size classification and detection of antler mushrooms. We studied online diagnosis and evaluation methods of antler mushroom size, proposed an evaluation model for the size of the antler mushroom, eliminated subjective judgment, and achieved the noncontact size grading of antler mushrooms. Experiments showed that the proposed algorithm could provide real-time classification and meet the real needs of industrial size classification of antler mushrooms. It is important to reduce production costs, improve classification efficiency and quality, and promote the development of antler mushroom industrialization.

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

Funding: Key Research and Development Project of Shandong Province (2022CXGC010610), Natural Science Foundation of Shandong Province (ZR2022QE103), China Agriculture Research System of MOF and MARA: CARS-24-D-01, China Postdoctoral Science Foundation (2019M662410).

Data Availability Statement: The code in this article cannot be published due to privacy, and can be obtained from the corresponding author upon reasonable request.

Conflicts of Interest: All authors declare there is no conflict of interest.

References

- 1. Lin, X.; Xu, J.-L.; Sun, D.-W. Investigation of moisture content uniformity of microwave-vacuum dried mushroom (*Agaricus bisporus*) by NIR hyperspectral imaging. *LWT-Food Sci. Technol.* **2019**, 109, 108–117. [CrossRef]
- Gowen, A.A.; Tsenkova, R.; Esquerre, C.; Downey, G.; O'Donnell, C.P. Use of near infrared hyperspectral imaging to identify water matrix co-ordinates in mushrooms (*Agaricus bisporus*) subjected to mechanical vibration. *J. Near Infrared Spectrosc.* 2009, 17, 363–371. [CrossRef]
- 3. Yang, S.; Huang, L.; Zhang, X. Research and application of machine vision in monitoring the growth of facility seedling crops. *Jiangsu Agric. Sci.* **2019**, *47*, 179–187.
- He, Z. Design and Implementation of Control System in Plant Factory Based on Visual Patrol Robot. Ph.D. Thesis, University of Science and Technology of China, Hefei, China, 2019.
- Pereira Valente, J.R.; Guiffrida, M.V. Leaf counting from uncontrolled acquired images from greenhouse workers. In Proceedings of the Computer Vision Problems in Plant Phenotyping (CVPPP 2019), Long Beach, CA, USA, 17 June 2019; pp. 1–2.
- 6. Dewi, T.; Risma, P.; Oktarina, Y. Fruit sorting robot based on color and size for an agricultural product packaging system. *Bull. Electr. Eng. Inform.* **2020**, *9*, 1438–1445. [CrossRef]
- Da Costa, A.Z.; Figueroa, H.E.H.; Fracarolli, J.A. Computer vision based detection of external defects on tomatoes using deep learning. *Biosyst. Eng.* 2020, 190, 131–144. [CrossRef]
- Jiang, H.; Li, X.; Safara, F. IoT-based agriculture: Deep learning in detecting apple fruit diseases. *Microprocess. Microsyst.* 2021, 104321, *in press.* [CrossRef]
- 9. Xu, P.; Tan, Q.; Zhang, Y.; Zha, X.; Yang, S.; Yang, R. Research on Maize Seed Classification and Recognition Based on Machine Vision and Deep Learning. *Agriculture* **2022**, *12*, 232. [CrossRef]
- Vooren, J.G.V.D.; Polder, G.; van der Heijden, G.W.A.M. Application of image analysis for variety testing of mushroom. *Euphytica* 1991, 57, 245–250. [CrossRef]
- Zheng, J.; Feng, W.; Liu, B.; Wang, F. Design of Agaricus Bisporus Automatic Grading System Based on Machine Vision. In Proceedings of the International Conference on Computer and Computing Technologies in Agriculture, Dongying, China, 19–21 October 2016; Springer: Cham, Switzerland, 2016; pp. 388–395.
- 12. Heinemann, P.H.; Hughes, R.; Morrow, C.T.; Sommer, H.J., III; Beelman, R.B.; Wuest, P.J. Grading of mushrooms using a machine vision system. *Trans. ASAE* **1994**, *37*, 1671–1677. [CrossRef]
- ViZhányó, T.; Felföldi, J. Enhancing colour differences in images of diseased mushrooms. Comput. Electron. Agric. 2000, 26, 187–198. [CrossRef]
- 14. Chen, H.H.; Ting, C.H. The development of a machine vision system for shiitake grading. *J. Food Qual.* **2004**, 27, 352–365. [CrossRef]
- 15. Yu, G.H.; Zhao, Y.; Li, G.; Shi, H. Algorithm for locating individual mushroom and description of its contour using machine vision. *Trans. CSAE* **2005**, *21*, 101–104.
- Li, J.B.; Wang, J.Y. Defects detection of Lentinus edodes surface based on computer vision technology. *Packag. Food Mach.* 2010, 28, 1–5.

- 17. Concha-Meyer, A.; Joseph, E.; Hengjian, W. Volume estimation of strawberries, mushrooms, and tomatoes with a machine vision system. *Int. J. Food Prop.* **2018**, *21*, 1867–1874. [CrossRef]
- 18. Lu, C.-P.; Liaw, J.-J.; Wu, T.-C.; Hung, T.-F. Development of a mushroom growth measurement system applying deep learning for image recognition. *Agronomy* **2019**, *9*, 32. [CrossRef]
- 19. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. arXiv, 2018, arXiv:1804.02767.
- Jiang, F.; Yang, X.; Wang, Y.; Yang, L.; Sun, B. Design of an Online Quality Inspection and Sorting System for Fresh Button Mushrooms (*Agaricus bisporus*) Using Machine Vision. *Eng. Lett.* 2022, 30, 1–6.
- Cheng, T.; Wang, X.; Chen, S.; Zhang, W.; Zhang, Q.; Huang, C.; Zhang, Z.; Liu, W. Sparse Instance Activation for Real-Time Instance Segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 19–24 June 2022; pp. 4433–4442.
- 22. Thuan, D. Evolution of Yolo Algorithm and Yolov5: The State-of-the-Art Object Detention Algorithm. Bachelor's Thesis, Oulu University of Applied Sciences, Oulu, Finland, 2021.
- Zhao, H.; Shi, J.; Qi, X. Pyramid scene parsing network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2881–2890.
- Howard, A.; Sandler, M.; Chu, G.; Chen, L.C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for mobilenetv3. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 1314–1324.
- Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
- Bochkovskiy, A.; Wang, C.Y.; Liao, H. YOLOv4: Optimal Speed and Accuracy of Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 1–17.
- He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
- 28. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* 2017, arXiv:1704.04861.
- 29. Gao, S.P.; Zhao, Q.Y.; Qi, X.G.; Cheng, M.F. Research on the improved image classification method of MobileNet. *CAAI Trans. Intell. Syst.* **2021**, *16*, 11–20.
- Tan, M.; Chen, B.; Pang, R.; Vasudevan, V.; Sandler, M.; Howard, A.; Le, Q.V. Mnasnet: Platform-aware neural architecture search for mobile. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 2820–2828.
- Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 779–788.