# Sundry Bacteria Contamination Identification of Lentinula Edodes Logs Based on Deep Learning Model

Dawei Zu [1,2,†], Feng Zhang [1,†], Qiulan Wu [1,\*], Cuihong Lu [1], Weiqiang Wang [1] and Xuefei Chen [1]

1. School of Information Science & Engineering, Shandong Agricultural University, Tai'an 271018, China
2. School of Journalism and Communication, Shandong University, Jinan 250100, China
* Correspondence: wqlsdau@163.com
† These authors contributed equally to this work.

**Abstract:** Lentinula edodes logs are susceptible to sundry bacteria contamination during the culture process, and the manual identification of contaminated logs is difficult, untimely, and inaccurate. Aiming to solve this problem, this paper proposes a method for the identification of contaminated Lentinula edodes logs based on the deep learning model Ghost–YOLOv4. Firstly, a data set of Lentinula edodes log sundry bacteria contamination was constructed. Secondly, in view of the problems that the YOLOv4 network parameters are too large and that the detection speeds of Lentinula edodes log videos are slow, the backbone feature extraction network was replaced with a lightweight network, GhostNet, and the YOLOv4 enhancement feature extraction network PANet and the Yolo Head modules were completed. The modification of the network reduced the number of parameters of the network and improved the detection speed of the network. Finally, the feature extraction network introduced the migration learning pre-training model, which reduced the computational pressure and overfitting problems of the model and further improved the performance of the Ghost–YOLOv4 network. Not only did the constructed Ghost–YOLOv4 ensure the accuracy of the identification and detection of Lentinula edodes log sundry bacteria contamination, but it also had better results in detection speed and real-time performance, and it provides an effective solution for the lightweight deployment of a target detection model on embedded equipment in culture sheds.

**Keywords:** crop disease monitoring; precision agriculture; artificial intelligence; Ghost–YOLOv4

## 1. Introduction

Lentinula edodes logs are often contaminated by sundry bacteria during the culture process [1–4]. When the production environment of Lentinula edodes logs is not thoroughly cleaned, the disinfection of the culture material is not adequate, and the operation is not standardized; thus, sundry bacteria are provided with a good opportunity to cause Lentinula edodes log diseases [5,6]. At the same time, in the process of pre-cultivation and cultivation, the Lentinula edodes logs need puncture ventilation to increase the oxygen content in the culture material, eliminate the waste gas released by the growth of the mycelium, and shorten the time for the physiological maturation of the mycelium. If the contaminated Lentinula edodes logs are not selected in time when they are punctured, the sundry bacteria on the contaminated Lentinula edodes logs remain on the puncturing needle, which leads to the contamination of the Lentinula edodes logs that need to be punctured next, causing great economic losses to enterprises [7].

At present, the identification of Lentinula edodes log sundry bacteria contamination in China is mainly based on the experience of experts in the agricultural field and technicians in plant protection [8]. Technicians need to have good observation abilities and rich experience to accurately identify the type of Lentinula edodes log contamination. This traditional identification method, which depends on personal experience, has great limitations. When there are too many Lentinula edodes logs and too many varieties to be

detected, the probability of error in the identification method based on human experience increases. For example, there are various problems, such as careless observations by rod inspectors; a heavy workload; and the untimely selection of contaminated Lentinula edodes logs, resulting in the proliferation of sundry bacteria [9].

With the development of deep learning [10–12], the era of automatic detection of crop diseases [13–15] has come. Deep learning image recognition models have good graphic feature extraction performance, and they can quickly and non-destructively monitor and recognize crop diseases within the visible light range and with higher accuracy, faster detection speed, and better stability [16]. Sravan et al. [17] used the ResNet50 model for plant disease identification, and after fine-tuning the training method of super parameters, the model achieved the highest classification accuracy of 99.26%. Zi caifei et al. [18] used rice blast images as a research object, and they proposed a rice blast recognition method based on deep learning. By learning the common health status of rice and using the rice blast picture set, the model features could be obtained, and the final model could be used for detection and judgment. Huang Linsheng et al. [19] improved the residual network resnet18, introduced a concept module, and used its multi-scale convolution kernel structure to extract disease features at different scales. The average recognition accuracy of the improved multi-scale attention residual network model multi-scale-se-resnet18 on the eight crop disease data sets collected in a complex field environment reached 95.62%.

The identification and classification of diseases related to the contamination of Lentinula edodes logs should refer to the development results of computer vision technology [20–23] and should improve the efficiency of monitoring the contamination of Lentinula edodes logs by means of efficient treatment methods. On the basis of this, a recognition method of Lentinula edodes log sundry bacteria contamination based on the deep learning model Ghost–YOLOv4 is proposed.

The main contributions of this paper are as follows:

(1) In terms of identification of the sundry bacteria contamination of Lentinula edodes logs, domestic edible fungus companies basically rely on manual inspection by inspectors. At the same time, literature related to the sundry bacteria contamination identification of Lentinula edodes logs was not found. Therefore, this paper may report the first research in China to apply a deep learning image recognition model to the identification of Lentinula edodes log sundry bacteria contamination.

(2) This study constructed a data set of Lentinula edodes log sundry bacteria contamination, including 4126 images, 3 types of sundry bacteria contamination, and 1 type of normal Lentinula edodes log, collected and annotated by ourselves.

(3) In order to deploy the lightweight target detection model on the Lentinula edodes log puncturing machine and realize the real-time monitoring of the Lentinula edodes log sundry bacteria contamination, a Ghost–YOLOv4 Lentinula edodes log sundry bacteria contamination target detection algorithm is proposed, which provides an effective solution for the selection of contaminated Lentinula edodes logs. The research in this paper is based on the demand of the edible fungi industry; it is of great significance to reduce the spread of sundry bacteria contamination, improve the product quality of Lentinula edodes logs, and increase the economic benefits of the company.

## 2. Materials and Methods

*2.1. Lentinula Edodes Log Sundry Bacteria Contamination Data Set*

2.1.1. Data Acquisition

From May 2021 to September 2021, images and videos of Lentinula edodes log sundry bacteria contamination were collected manually in the culture shed of a smart factory of a company in Shandong. An image of part of the Lentinula edodes log sundry bacteria contamination was taken on the spot with a Canon camera. The model of the camera was Canon EOS 600D, the image resolution was 5184 × 3456, the focal length was 38 mm, the ISO speed was 400, and the shutter speed was 1/60 s. There was an LED strip light panel

every 2 m in the cultivation shed, 20 W white light, and good lighting conditions. The fixed camera was placed 50 cm from the Lentinula edodes logs for shooting. The shooting standard was to shoot 4 images of each Lentinula edodes log in a 90-degree rotation manner. The second part was to add a camera to the piercing machine in the cultivation shed. The model of the camera was Hikvision E14a, the resolution was 2560 × 1440, and the field of view was 80 °H 90 °D. When the Lentinula edodes log piercing machine pulled the Lentinula edodes logs from the shelf and rotated the piercing hole, the camera could shoot the video in 360 degrees. In the method of adding an image acquisition device to the Lentinula edodes log piercing machine, images were captured while the piercing hole was ventilated, thereby improving the efficiency of image acquisition. This method collected 35 videos, which were divided into frames and sampled. In order to make a training set, a validation set, and a test set, the 35 videos were divided into frames. Because each video contained a large number of frames with very similar content, if all these pictures were used as data sets, it would greatly increase the annotation workload. Therefore, this paper used footnotes to sample the frames generated by each video. Specifically, 1 frame was collected every 60 or 100 frames, and it was added to the final Lentinula edodes log sundry bacteria contamination data set. Finally, 4126 pictures were obtained, which were divided into a training set, a validation set, and a test set according to the ratio of 7:2:1. The two acquisition methods collected 942 images of Aspergillus flavus-contaminated Lentinula edodes logs (as shown in Figure 1a,b), 893 images of Trichoderma viride-contaminated Lentinula edodes logs (as shown in Figure 1c,d), 664 images of Neurospora-contaminated Lentinula edodes logs (as shown in Figure 1e,f), and 1627 images of normal Lentinula edodes logs (as shown in Figure 1g,h), resulting in a total of 4126 images of Lentinula edodes logs. Figure 1 shows the data set of the sundry bacteria contamination of Lentinula edodes logs.
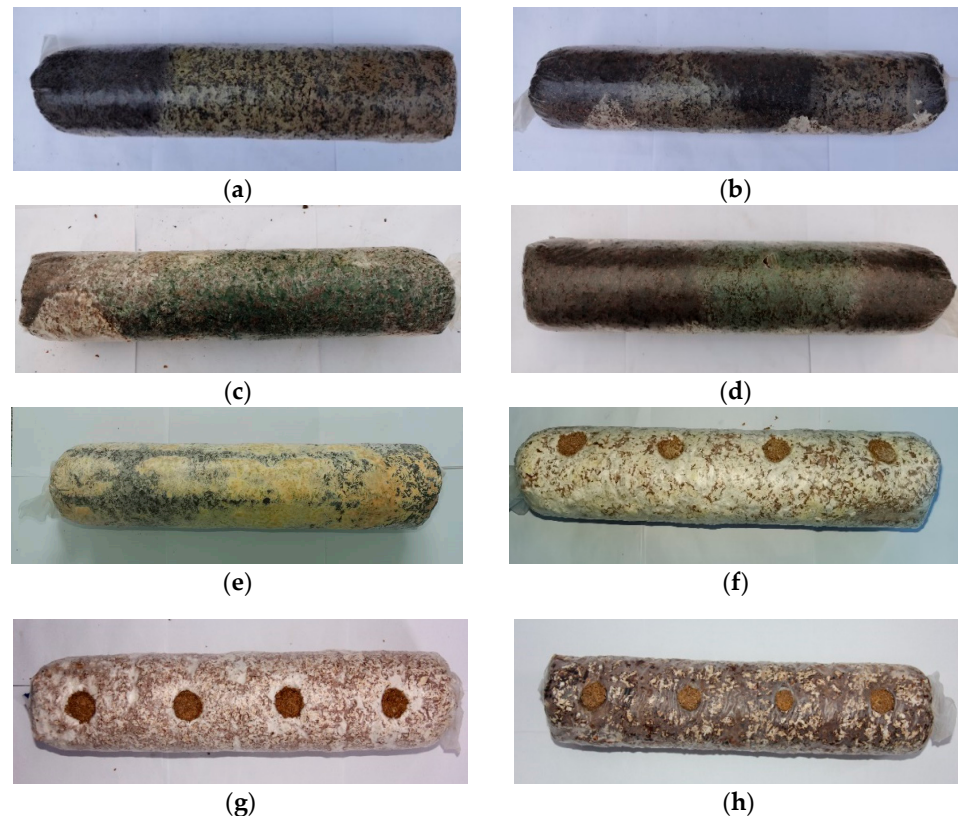


**Figure 1.** Example of Lentinula edodes log sundry bacteria contamination sample. (**a**,**b**) are Aspergillus flavus-contaminated Lentinula edodes logs; (**c**,**d**) are Trichoderma viride-contaminated Lentinula edodes logs; (**e**,**f**) are Neurospora-contaminated Lentinula edodes logs; (**g**,**h**) are normal Lentinula edodes logs.

### 2.1.2. Data Set Preprocessing

In this experiment, the data set of Lentinula edodes log sundry bacteria contamination was used, with a total of 4126 original collected images. The images were noised, inverted, rotated, and randomly adjusted by the python script data enhancement operation; the data volume was expanded to 16,504 images, which were saved in the JPEGImages folder in .jpg format.

In the experiment, labelImg labeling software was used to label the types of bacterial contamination in the dataset. For example, Figure 2 shows the labeling of Trichoderma viride-contaminated Lentinula edodes logs. The a priori box was used to select the Trichoderma viride-contaminated Lentinula edodes logs, and the names of the contaminated bacteria were entered. The File List in the lower right corner displays the image names that needed to be labeled.



**Figure 2.** Annotation of Lentinula edodes log sundry bacteria contamination image.

After labeling, the labelImg software automatically generated .xml label files in the pascal VOC format [24]. These label files recorded the location, size, and category information of all target a priori boxes, and the corresponding pictures of Lentinus edodes were named and stored in the Annotations label folder.

The experiment completed the labeling of 16,504 images; the Lentinula edodes log sundry bacteria contamination data set was collected and cleaned, and the missing and incorrect labels in the .xml file were repaired. The repair method was as follows: for missing labels, the labelImg software was used to repeat the above method to label the images of Lentinula edodes log sundry bacteria contamination; for incorrect labels, the prior frame on the image was deleted and redrawn, and the contaminated bacteria name was entered. Finally, using the voc_annotation.py file, the images and annotation files of Lentinula edodes log sundry bacteria contamination were divided into a training set, a validation set, and a test set according to the ratio of 7:2:1.

### 2.2. Construction of the Recognition Model

#### 2.2.1. The Classic YOLOv4 Algorithm

YOLOv4 [25] is one of the most commonly used target detection algorithms, and it can accurately detect target objects in many scenes. Its network structure is shown in

Figure 3, which is mainly divided into three parts: a backbone feature extraction network, an enhanced feature extraction network, and a head prediction network.
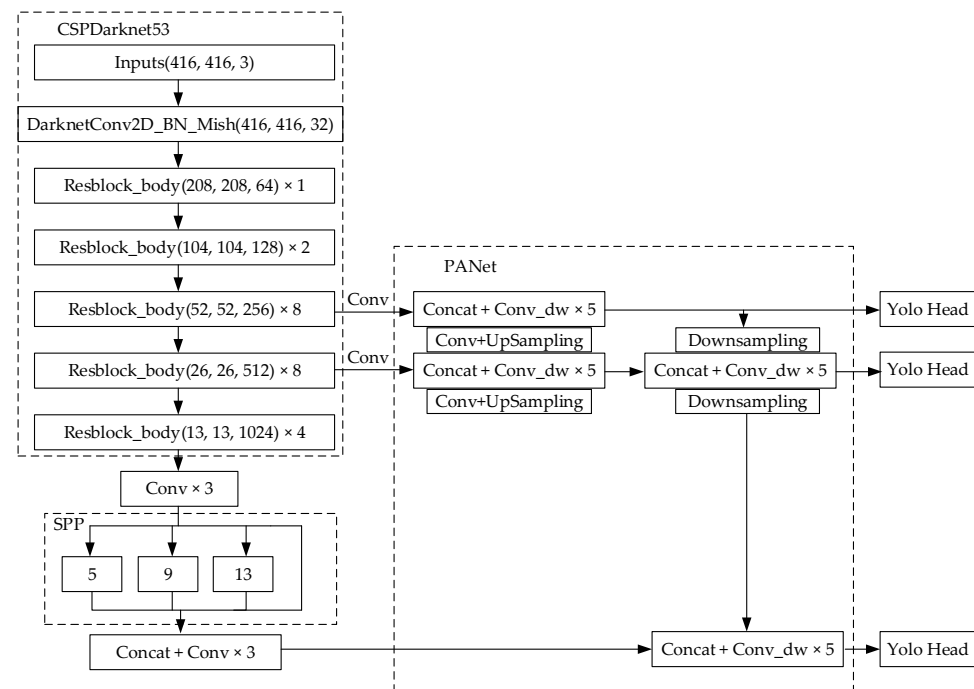


**Figure 3.** YOLOv4 model structure.

Among them, the backbone feature extraction network is CSPDarknet53, which consists of multiple residual blocks. When the feature image is input into the YOLOv4 network, the YOLOv4 network first adjusts the size of the feature image to $416 \times 416$, and then it segments the adjusted feature image into $13 \times 13$, $26 \times 26$, and $52 \times 52$; that is, the function of the CSPDarknet53 feature extraction network is to obtain $52 \times 52$, $26 \times 26$, and $13 \times 13$, three effective characteristic layers. The enhanced feature extraction network mainly includes the Spatial Pyramid Pooling (SPP) structure and the Path Aggregation Network (PANet) [26]. The SPP module performs $5 \times 5$, $9 \times 9$, and $13 \times 13$ maximum pooling of the three scales; then, the tensor concat is fused into a feature map, and the dimension reduction through convolution separates important context feature information on the premise of ensuring speed. The PANet includes up-sampling and down-sampling operations. By fusing top-down semantic features and bottom-up strong positioning features, the loss of target shallow feature information is alleviated, and the representation ability of the model is increased. The head prediction network, namely, the Yolo Head module, is the output layer of the YOLOv4 network [27]. The three heads correspond to three sizes, which are $19 \times 19$, $38 \times 38$, and $76 \times 76$. The position and confidence score of the boundary prediction frame are calculated on three feature grids of different sizes, and the non-maximum suppression algorithm is used to consider the position information of the center point of the boundary box and retain the best prediction frame.

### 2.2.2. GhostNet Network

GhostNet [28–30] is a lightweight CNN proposed by Huawei in 2020. The significance of its research is that it can make full use of computer computing power and storage resources in mobile terminals and small embedded devices in order to achieve the best performance of the model so as to meet various needs in computer vision.

The core module of the GhostNet network is the Ghost Module. As shown in Figure 4, Ghost Module generates m original feature maps T through one convolution and completes the traditional convolution in two steps. The first step uses $1 \times 1$ ordinary convolution for mapping to generate m necessary feature enrichment (Identity operation). The second step

is to use the depthwise separable convolution block to perform layer-by-layer convolution, perform linear transformation and stacking on T, and generate s feature maps. Finally, the feature maps of the two steps are spliced together through the concat layer to obtain the final *n* output feature maps [31,32], where *n* = m × s.
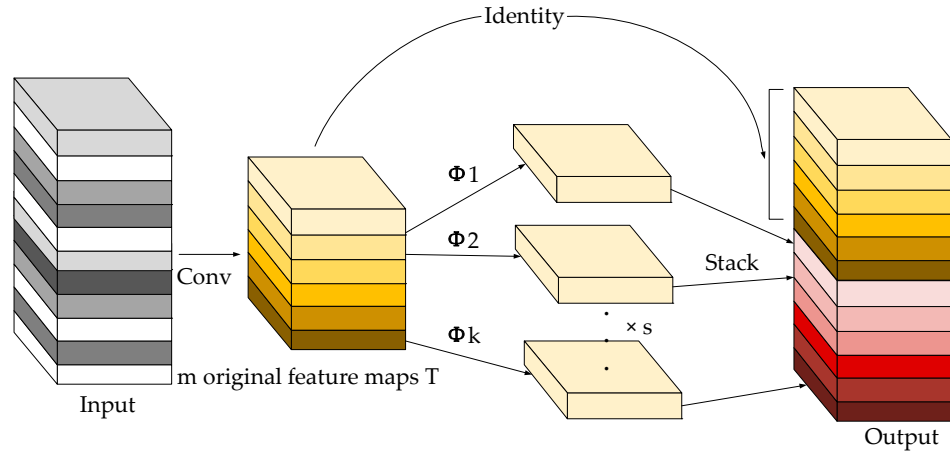


**Figure 4.** Ghost Module.

The core idea of the GhostNet network is to generate more features with fewer parameters. Compared with an ordinary convolutional neural network, the Ghost Module can ensure that the total number of parameters and computational complexity are effectively reduced without changing the scale of the output characteristic map, and the Ghost Module has the advantages of plug and play and convenient transplantation.

The Ghost bottleneck consists of two Ghost Modules [33], and its structure is shown in Figure 5. When the step size of the Ghost bottleneck is set to 1, two Ghost Modules are used for feature extraction in the trunk part, while no processing is carried out in the part of the residual side, and the input and output are added directly. When the step size of the Ghost bottleneck is set to 2, a Ghost Module is first used to extract the features of the input feature layer, channel-by-channel convolution is used to compress the height and width of the previous feature layer, a Ghost Module is used to extract the features again, and, finally, a residual edge is added.
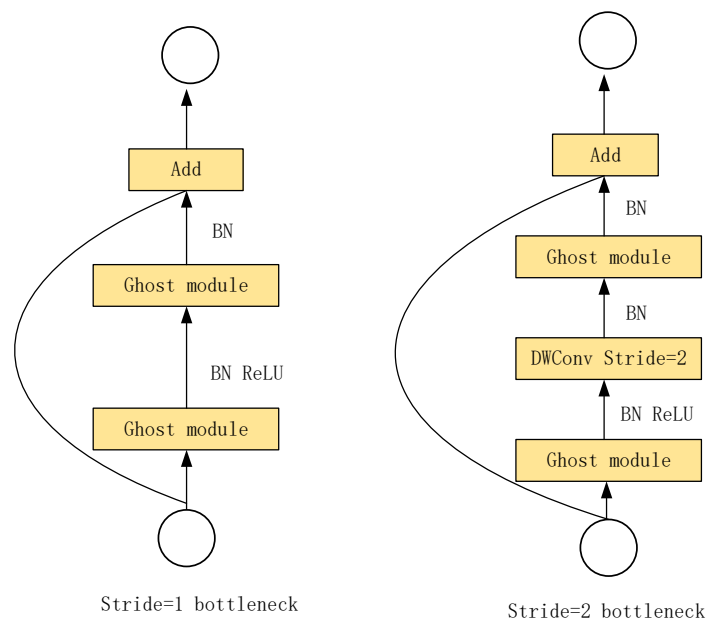


**Figure 5.** Ghost bottleneck.

### 2.2.3. Ghost–YOLOv4 Detection Algorithm of Lentinula Edodes Log Sundry Bacteria Contamination

The ultimate goal of this paper was to design a lightweight Lentinula edodes log contamination detection algorithm that can be applied to embedded devices. CSPDarknet53, the backbone feature extraction network in YOLOv4, improves operation accuracy by increasing the input feature information and the calculation amount of the network; this undoubtedly increases the GPU memory of the network and the storage capacity of the model, which also leads to the shortcomings of the YOLOv4 algorithm, such as high delay and slow speed in mobile terminals and embedded devices [34]. Therefore, a lightweight network model is needed to optimize the YOLOv4 network so that the YOLOv4 network can run smoothly in embedded equipment and achieve the goal of the real-time monitoring of the contamination of Lentinula edodes logs. On the basis of this, this paper designed a recognition model of Lentinula edodes log sundry bacteria contamination: Ghost–YOLOv4. The network structure is shown in Figure 6.
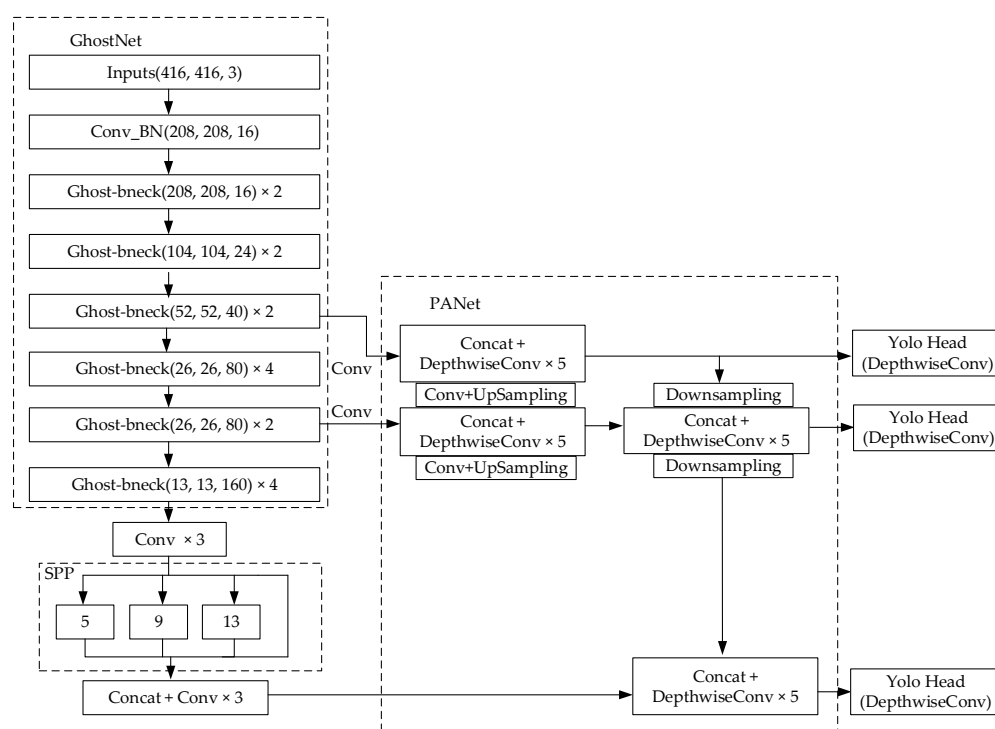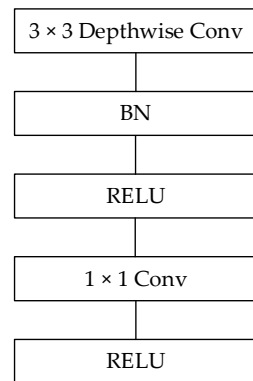


**Figure 6.** Ghost–YOLOv4 network structure.

The improved lightweight Ghost–YOLOv4 network uses GhostNet as the backbone extraction network to replace the original CSPDarknet53 network. The specific approach was to replace the backbone feature extraction network resblock module of YOLOv4 with the Ghost bottleneck module and to build it according to the network structure of GhostNet. It was found that, in the third, fifth, and sixth stages of the feature extraction network GhostNet, the three effective feature layers, $52 \times 52 \times 40$, $26 \times 26 \times 80$, and $13 \times 13 \times 160$, needed to be input into the enhanced feature extraction network. As shown in Figure 6, the output of the Ghost bottleneck in stages 3 and 5 of the GhostNet module entered the later output network through the full connection layer, and the output of the Ghost bottleneck in stage 6 entered the later output network from the SPP network. This completed the splicing of the three feature maps of different sizes, $13 \times 13$, $28 \times 28$, and $56 \times 56$, in the GhostNet network, and the YOLOv4 multi-scale was featured to form the final feature extraction network.

The Ghost–YOLOv4 network adopted PANet for parameter aggregation. PANet is a bidirectional fusion network that can be from top to bottom and from bottom to top, and

an adaptive feature channel is added between the lowest and top features, which can fully integrate different feature layers in the whole Ghost–YOLOv4 network. In the process of feature fusion, in order to further reduce the parameters of the model, the size of all the convolution kernels in PANet is $3 \times 3$, and they are replaced by a deeply separable convolution block [35–37]. A depthwise separable convolution block is composed of a $3 \times 3$ depthwise separable convolution and a $1 \times 1$ normal convolution, which can greatly reduce the parameter computation cost compared with the traditional convolution. The structure of a depthwise separable convolution block is shown in Figure 7.



**Figure 7.** Depthwise separable convolution block structure.

Therefore, the PANet structure in the Ghost–YOLOv4 network in this paper was mainly a circular pyramid network structure composed of depthwise separable convolution blocks, up and down sampling, feature fusion, and stacking.

The Yolo Head module of Ghost–YOLOv4 predicted the result on the basis of the characteristic information after the enhanced processing of PANet. Similarly, depthwise separable convolution blocks were also used to replace all convolution kernels in the Yolo Head module with a size of $3 \times 3$, to determine whether the three prediction frames generated by the prediction of each feature layer contained the required feature information, to suppress the non-maximum value and adjust the prior frame, and, finally, to obtain the final prediction frame.

### 2.3. Model Evaluation Method

In order to evaluate the model recognition performance, this paper comprehensively considered Precision, Recall, mAP, model size, network structure parameters, and detection speed FPS [38].

The calculation of Precision and Recall adopts the Macro-F1 calculation rule, and the calculation formulae are as follows. Among them, TP represents the number of positive samples predicted to be positive, FP represents the number of negative samples predicted to be positive, TN represents the number of negative samples predicted to be negative, and FN represents the number of positive samples predicted to be negative.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{1}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{2}$$

The average precision AP is the area under the Precision–Recall curve, and Total images represents the total number of pictures in the data set. The calculation formula is:

$$\text{AP} = \frac{\Sigma \, \text{Precision}}{\text{N (Total images)}} \tag{3}$$

The mAP value is the average value of the AP of the four categories to be detected, and Total classes represents the total category of the target to be detected in the dataset. The calculation formula is:

$$mAP = \frac{\Sigma\ AP}{N\ (Total\ classes)} \qquad (4)$$

## 3. Results

The hardware environment of the experiment was Windows 10, the CPU was Intel Core i7-12700H, the memory capacity was 16 GB, and the disk capacity was 1 T. During the training process, the Nvidia Gtx1070Ti GPU graphics card was used to accelerate the training. The software environment was CUDA10.0, Cudnn10.0, written with the python3.7 program; the open-source deep learning framework Pytorch was used as the development environment; and deep learning libraries, such as Matplotlib and Numpy, were installed for data analysis.

The pictures sets in the training and testing of the experiment had a size of $416 \times 416 \times 3$ and were in JPG format, and the dataset was enhanced with CutMix, Mosaic, and Self-Confrontational Training (SAT) methods [39] using DropBlock regularization to solve the fitting problem and using the class label smoothing method to enhance the generalization ability of the model. A total of 300 iterations were set, and four types of loss values were observed. According to the size of the computer memory, the parameter Batch size was set equal to 8, and the learning rate was 0.001. The learning rate was adjusted according to the cosine annealing learning rate algorithm that comes with Pytorch.

In the early stage of the experiment, although we carried out some work in the process of data set collection and production, the data set was relatively small, and the model trained from 0 was not necessarily good. In order to reduce the calculation pressure and overfitting problem of the model, the transfer learning pre-training weight was introduced into the backbone feature extraction network of Ghost–YOLOv4 [40]. The pre-training weight retained a large amount of parameter information trained by GhostNet on the VOC data set. On the basis of the improved Ghost–YOLOv4 model, the method of migration learning was adopted, and load was used; the load_state_dict function loaded the pre-training weight of GhostNet corresponding to ghostnet_weights.pth. We considered the problem that with more weight layers in the frozen GhostNet network, the accuracy of the model would decline relatively, and the overfitting ratio would increase more. This paper did not freeze the weight parameters of any layer of the GhostNet backbone feature extraction network, and it retrained the weight of all layers of the GhostNet backbone feature extraction network by using the Lentinula edodes log contamination data set. The reason for this is that the more weight layers that are frozen by GhostNet, the fewer parameters that need to be trained in the model; the weaker the calculation and feature extraction ability of the model on the Lentinula edodes log contamination images; the weaker the shared features between the layers due to the weakening of the ability of mutual influence; and the greater the inability to relearn when the original bottom layer features propagate upward layer by layer, resulting in the gradual decline of the feature migration ability of the top layer. That is, the model only transfers the high-level features and cannot realize the gradual abstraction, characterization, and extraction of features from the bottom to the top level, so the recognition rate of the model gradually declines in the end.

For the Lentinula edodes log sundry bacteria contamination data set, the pre-training weight parameters had a significant effect on the improvement of the accuracy of model recognition due to the VOC data set used in the pre-training weights of migration learning, which provided a large number of images so that the model could learn enough features and better fit the parameters and so that the model could obtain good initialization network parameters during migration learning. This reduced the possibility of fitting [41], and it also showed that the transfer learning ability of pre-training weights obtained with sufficient training data in the target domain was stronger than that of directly training small sample data.

The change in the loss value could be observed in real time during training. Figure 8 shows the loss curve of the training process. The train loss is the loss curve of the training set, the val loss is the loss curve of the validation set, and the smooth train loss and smooth val loss are the smoothing of the two. After training multiple epochs, one can see the generated weight file in the logs directory.
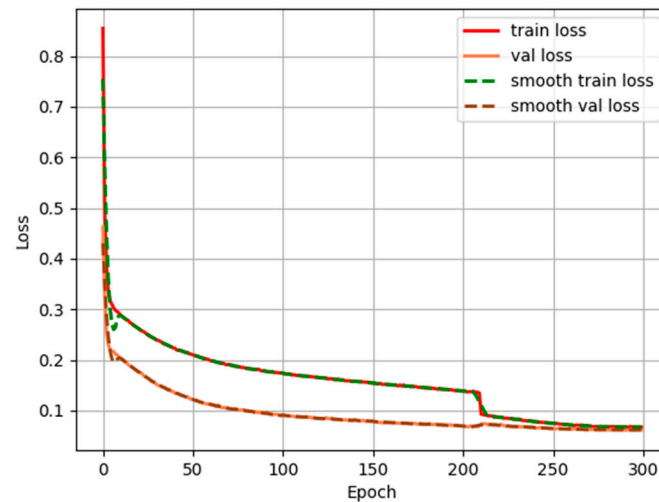


**Figure 8.** Loss value change curve.

In order to prove the effectiveness of the methods in this paper, sufficient comparative experiments were carried out. Using the Lentinula edodes log sundry bacteria contamination test data set and the captured Lentinula edodes log video, comparative experiments were carried out on the original YOLOv4, the improved Ghost–YOLOv4, and the improved Mobilenetv3–YOLOv4 in the same modified way. Table 1 shows the comparison results of various performance indicators of the original YOLOv4 algorithm, Mobilenetv3–YOLOv4, and the Ghost–YOLOv4 detection algorithm. The evaluation indicators were mAP, Precision, Recall, Model size, Network structure parameters, detection speed FPS, etc.

**Table 1.** Comparison of performance indexes of the model.

| Performance Index | YOLOv4 | Mobilenetv3–YOLOv4 | Ghost–YOLOv4 |
|---|---|---|---|
| mAP/% | 93.59 | 92.27 | 93.17 |
| Precision/% | 95.1 | 94.47 | 94.5 |
| Recall/% | 91.46 | 89.11 | 91.02 |
| Model size/MB | 224.29 | 53.77 | 43.4 |
| Network structure parameter quantity | 69,040,001 | 11,729,069 | 11,482,545 |
| FPS/(frames/s) | 23 | 27.68 | 39 |

It can be seen in Table 1 that the improved Ghost–YOLOv4 could reduce the model size to only 19.3% of the original YOLOv4 model, greatly reduce the model parameters from 69,040,001 to 11,482,545, and improve FPS to 1.69 times that of YOLOv4, with the detection accuracy almost unchanged. This shows that the complexity of the improved Ghost–YOLOv4 model was greatly reduced, and the size of the model was only 0.19 times that of the original. On the premise of almost no reduction in accuracy, the real-time performance of the algorithm was effectively improved, and the detection speed was increased by 1.7 times. It is more suitable for lightweight model deployment on embedded devices.

Figure 9 shows the detection effect diagram. It can be seen that the Ghost–YOLOv4 detection model could accurately detect the sundry bacteria contamination species of the Lentinula edodes logs.
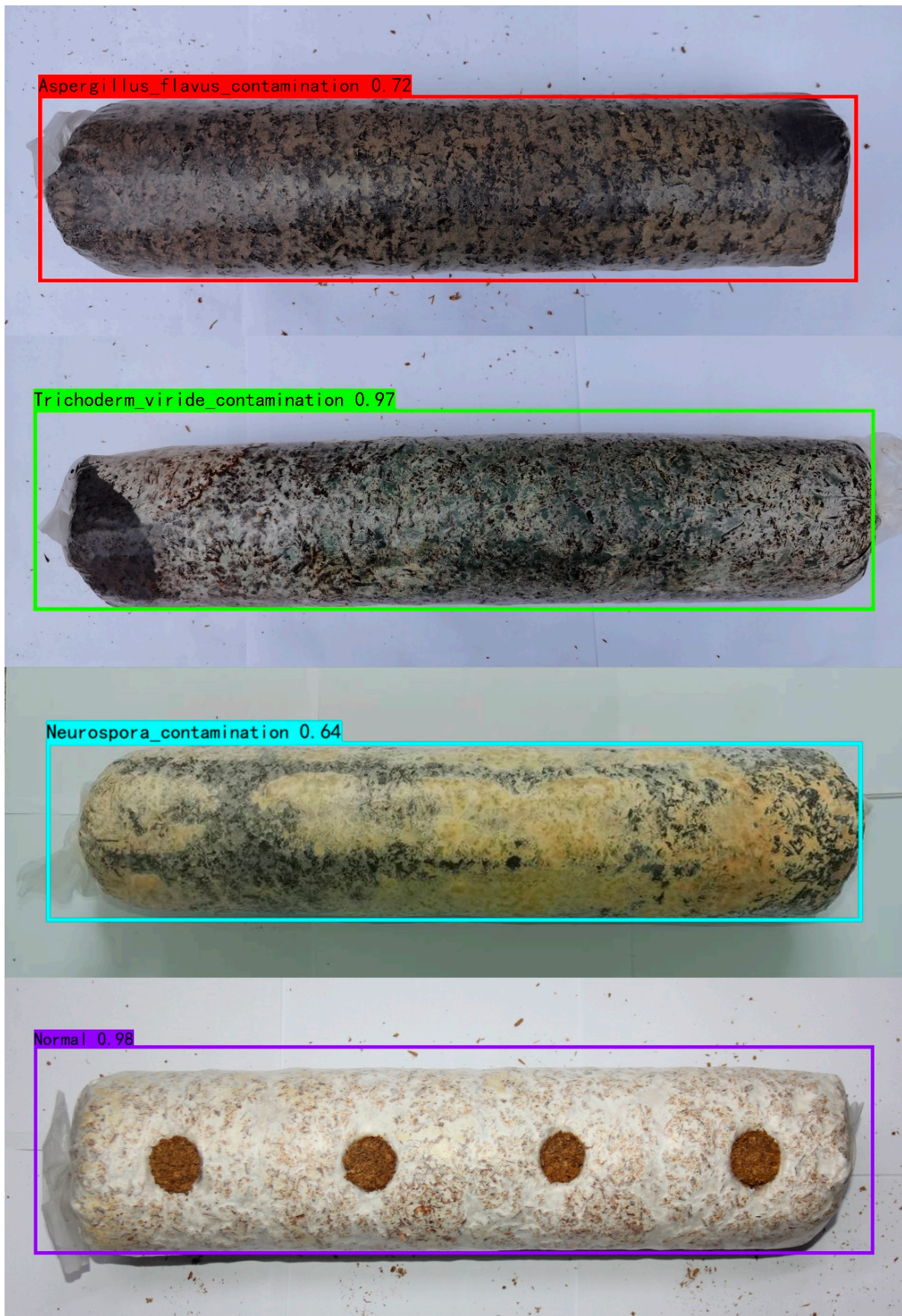
**Figure 9.** Target detection results of sundry bacteria contamination of Lentinula edodes logs.

## 4. Discussion

Aiming to solve the problem of the number of YOLOv4 network parameters being too large and the detection speed of Lentinula edodes log video on embedded equipment being slow, this paper designed a lightweight network model, Ghost–YOLOv4, based on deep learning, which has a high recognition accuracy, meets the requirements of Lentinula edodes log contamination identification, reduces the workload of factory staff, and is of great significance in improving the product quality of Lentinula edodes logs. Reducing the economic losses of edible fungi enterprises has certain practical significance and value. In follow-up work, we will study the development of the Lentinula edodes log contamination identification system, complete the deployment of the lightweight target detection model on embedded equipment, connect the image acquisition equipment in the culture shed (so as to identify the contaminated Lentinula edodes logs in the production process), record the number and status of contaminated Lentinula edodes logs, send an alarm for abnormal conditions, trace the source of the Lentinula edodes log contamination, and improve the product quality of the Lentinula edodes logs.

**Author Contributions:** D.Z. and F.Z. designed the work and guided the fundamental concept to the co-author, and they designed the data processing process, drafted the work, and submitted it to Scientific Reports; X.C. took an image of Lentinula edodes log disease; C.L. and W.W. prepared the data set; Q.W. reviewed the paper and put forward suggestions for revision. All authors have read and agreed to the published version of the manuscript.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

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

## References

1. Geng, L.; Gong, F.P.; Zhang, Y.X. The main contamination forms and their relationship in the industrialized production of Lentinula edodes logs. *Edible Med. Fungi* **2019**, *27*, 283–286.
2. Zhong, Z.L. Causes and control measures of rotten tube of Lentinula edodes in layered cultivation in summer. *Edible Fungi* **2021**, *43*, 68–69.
3. Wan, L.C.; Ren, H.X.; Guo, H.D.; Ren, P.F.; Qu, L.; Chang, Z.H.; Zhao, J.C.; Wang, H.P.; Zhao, Y. Analysis of severe contamination of rhizopus longipectus and key technologies of green prevention and control. *Edible Fungi* **2021**, *43*, 66–68.
4. Liu, Y.N.; Mao, F.R.; Zhu, X.T.; Lin, X.X.; Sun, Z.Y. Contamination and control of main sundry bacteria in the production of Lentinula edodes. *Jilin Veg.* **2017**, *21*, 31–32.
5. Zhang, L.L.; An, F.Y.; Wang, Q. Investigation of main diseases of Lentinula edodes in maling township and screening of control methods. *Anhui Agric. Sci.* **2017**, *45*, 133–135.
6. Chen, M.Z. Mold contamination and control of bag cultivated Lentinula edodes. *Agric. Technol. Serv.* **2016**, *33*, 98–99.
7. Li, Y.L. Industrialized production and contamination control of Lentinula edodes logs. *Agric. Sci. Technol. Inf.* **2016**, *25*, 83–84.
8. Cui, L.H. Isolation, identification and diversity analysis of contaminated fungi on edible fungi cultivation rods. *Liaoning Norm. Univ.* **2018**, *4*, 67–71.
9. Cheng, C.B.; Xu, X.J. Characteristics and comprehensive preventive measures of Aspergillus flavus contamination of Lentinula edodes in summer. *Edible Med. Fungi* **2014**, *22*, 359–360.
10. Liu, Y.; Dong, H.; Wang, L. Trampoline motion decomposition method based on deep learning image recognition. *Sci. Program.* **2021**, *9*, 1215065. [CrossRef]
11. Wang, H.; Huang, D.; Wang, Y. GridNet: Efficiently learning deep hierarchical representation for 3D point cloud understanding. *Front. Comput. Sci.* **2022**, *16*, 1–9. [CrossRef]
12. Elyan, E.; Vuttipittayamongkol, P.; Johnston, P. Computer vision and machine learning for medical image analysis: Recent advances, challenges, and way forward. *Artif. Intell. Surg.* **2022**, *2*, 24–45. [CrossRef]
13. Edna, C.T. A comparative study of fine-tuning deep learning models for plant disease identification. *Comput. Electron. Agric.* **2019**, *161*, 272–279.
14. Rahman, C.R.; Arko, P.S.; Ali, M.E. Identification and recognition of rice diseases and pests using convolutional neural networks. *Biosyst. Eng.* **2020**, *194*, 112–120. [CrossRef]

15. Xiong, F.K.; Lu, L.; Cao, T.R. Crop leaf diseases recognition: A generative adversarial network based approach. *Comput. Mod.* **2020**, *303*, 43–50.

16. Wang, G.W.; Wang, J.X.; Yu, H.Y.; Sui, Y.Y. Research on identification of corn disease occurrence degree based on improved ResNeXt network. *Int. J. Pattern Recognit. Artif. Intell.* **2022**, *36*, 2250005. [CrossRef]

17. Sravan, V.; Swaraj, K.; Meenakshi, K. A deep learning based crop disease classification using transfer learning. *Mater. Today Proc.* 2021, *in press*. [CrossRef]

18. Zi, C.F.; Cao, Z.Y.; Xu, J.J.; Chen, M.; Gao, Y. Research on rice blast recognition based on deep learning. *Mod. Agric. Sci. Technol.* **2022**, *01*, 111–118.

19. Huang, L.S.; Luo, Y.W.; Yang, X.D.; Yang, G.J.; Wang, D.Y. Crop disease identification based on attention mechanism and multi-scale residual network. *J. Agric. Mach.* **2021**, *52*, 264–271.

20. Li, W.Q.; Wang, D.; Ning, Z.T.; Lu, M.L.; Qin, P.F. Survey of fruit object detection algorithms in computer vision. *Comput. Mod.* **2022**, *06*, 87–95.

21. Yang, P.X.; Wang, H.L.; Zong, Q.; Chen, L. Design of automatic fruit grading system based on computer vision. *Shihezi Sci. Technol.* **2022**, *03*, 16–17.

22. Wang, T.S. The development and application of computer vision technology. *Inf. Syst. Eng.* **2022**, *04*, 63–66.

23. Lu, H.T.; Luo, M.K. Survey on new progresses of deep learning based computer vision. *J. Data Acquis. Process.* **2022**, *37*, 247–278.

24. Chen, Z.X.; Tian, S.W.; Yu, L.; Zhang, L.Q.; Zhang, X.Y. An object detection network based on YOLOv4 and improved spatial attention mechanism. *J. Intell. Fuzzy Syst.* **2022**, *42*, 2359–2368. [CrossRef]

25. Dlamini, S.; Kao, C.Y.; Su, S.L.; Jeffrey, K.C. Development of a real-time machine vision system for functional textile fabric defect detection using a deep YOLOv4 model. *Text. Res. J.* **2022**, *92*, 675–690. [CrossRef]

26. Wang, G.B.; Ding, H.W.; Yang, Z.J.; Li, B.; Wang, Y.H.; Bao, L.Y. TRC-YOLO: A real-time detection method for lightweight targets based on mobile devices. *IET Comput. Vis.* **2021**, *16*, 126–142. [CrossRef]

27. Liu, T.; Pang, B.; Zhang, L.; Yang, W.; Sun, X.Q. Sea surface object detection algorithm based on YOLO v4 fused with reverse depthwise separable convolution (RDSC) for USV. *J. Mar. Sci. Eng.* **2021**, *9*, 753. [CrossRef]

28. Han, K.; Wang, Y.; Tian, Q.; Guo, J.; Xu, C.; Xu, C. GhostNet: More features from cheap operations. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 1577–1586.

29. Wei, B.; Shen, X.; Yuan, Y. Remote sensing scene classification based on improved GhostNet. *J. Phys. Conf. Ser.* **2020**, *1621*, 012091. [CrossRef]

30. Zhang, S.; Zhou, X. MicroNet: Realizing micro neural network via binarizing GhostNet. In Proceedings of the International Conference on Intelligent Computing and Signal Processing (ICSP), Xi'an, China, 9–11 April 2021.

31. Cao, Y.J.; Gao, Y.X. Lightweight beverage recognition network based on GhostNet residual structure. *Comput. Eng.* **2022**, *48*, 310–314.

32. Sun, D.L.; Wang, J.C.; Chen, K.; Sun, S.W.; Liu, X.T.; Zhou, W.T. Two-scale pig target detection based on Ghost-YOLOv3-2. *Jiangsu Agric. Sci.* **2022**, *50*, 189–196.

33. Zhang, Z.T.; Hu, X.Q.; Wang, S.Q.; Kang, L.; Ma, Q.Y. Trip-GhostNet for Hyperspectral Image Classification. *J. Phys. Conf. Ser.* **2021**, *2024*, 012006. [CrossRef]

34. Xiang, X.J.; Song, X.M.; Zheng, Y.P.; Wang, H.B.; Fang, Z.Y. Research on embedded face detection based on mobilenet YOLO. *China J. Agric. Mach. Chem.* **2022**, *43*, 124–130.

35. Li, G.Q.; Zhang, J.W.; Zhang, M.; Wu, R.X.; Cao, X.Y.; Liu, W.Z. Efficient depthwise separable convolution accelerator for classification and UAV object detection. *Neurocomputing* **2022**, *490*, 1–16. [CrossRef]

36. Jiang, Z.T.; Huang, Y.S.; Hu, L.R. Single image super-resolution: Depthwise separable convolution super-resolution generative adversarial network. *Appl. Sci.* **2020**, *10*, 375. [CrossRef]

37. Hu, G.; Wang, K.J.; Liu, L.L. Underwater acoustic target recognition based on depthwise separable convolution neural networks. *Sensors* **2021**, *21*, 1429. [CrossRef] [PubMed]

38. Zhou, W.; Niu, Y.Z.; Wang, Y.W.; Li, D. Rice pests and diseases identification method based on improved YOLOv4-GhostNet. *Jiangsu J. Agric. Sci.* **2022**, *38*, 685–695.

39. Zhu, X.D. Research on pedestrian detection method based on YOLOv5. *Agric. Equip. Veh. Eng.* **2022**, *60*, 4.

40. Wong, L.J.; Michaels, A.J. Transfer learning for radio frequency machine learning: A taxonomy and survey. *Sensors* **2022**, *22*, 1416. [CrossRef] [PubMed]

41. Xie, L.Y.; Xia, Z.J.; Zhu, S.H.; Zhang, D.Q.; Zhao, F.K. Analysis and research on over fitting of image recognition based on convolutional neural network. *Softw. Eng.* **2019**, *22*, 27–29.