Article

An Approach for Rice Bacterial Leaf Streak Disease Segmentation and Disease Severity Estimation

Shuo Chen [1], **Kefei Zhang** [1,2,*], **Yindi Zhao** [1], **Yaqin Sun** [1], **Wei Ban** [1], **Yu Chen** [1], **Huifu Zhuang** [1], **Xuewei Zhang** [1], **Jinxiang Liu** [1] **and Tao Yang** [1]

[1] School of Environment and Spatial Informatics, China University of Mining and Technology, Xuzhou 221116, China; stdchensh@cumt.edu.cn (S.C.); zhaoyd@cumt.edu.cn (Y.Z.); syqin@cumt.edu.cn (Y.S.); banwei@whu.edu.cn (W.B.); chenyu@cumt.edu.cn (Y.C.); huifuzhuang@163.com (H.Z.); tb19160015b2@cumt.edu.cn (X.Z.); liujinxiang@cumt.edu.cn (J.L.); tyang@cumt.edu.cn (T.Y.)

[2] Satellite Positioning for Atmosphere, Climate and Environment (SPACE) Research Center, RMIT University, Melbourne, VIC 3001, Australia

* Correspondence: PROFKZHANG@CUMT.EDU.CN

Abstract: Rice bacterial leaf streak (BLS) is a serious disease in rice leaves and can seriously affect the quality and quantity of rice growth. Automatic estimation of disease severity is a crucial requirement in agricultural production. To address this, a new method (termed BLSNet) was proposed for rice and BLS leaf lesion recognition and segmentation based on a UNet network in semantic segmentation. An attention mechanism and multi-scale extraction integration were used in BLSNet to improve the accuracy of lesion segmentation. We compared the performance of the proposed network with that of DeepLabv3+ and UNet as benchmark models used in semantic segmentation. It was found that the proposed BLSNet model demonstrated higher segmentation and class accuracy. A preliminary investigation of BLS disease severity estimation was carried out based on our BLS segmentation results, and it was found that the proposed BLSNet method has strong potential to be a reliable automatic estimator of BLS disease severity.

Keywords: rice bacterial leaf streak; leaf disease recognition; lesion segmentation; semantic segmentation; deep learning; convolutional neural network; disease severity estimation

1. Introduction

Rice (*Oryza sativa* L.) is one of the three main crops in Asia and is widely planted. Due to the complex characteristics of rice growing in relatively hot and humid conditions, it is vulnerable to diseases. Rice bacterial leaf streak (BLS) is one of the most serious diseases affecting rice production, occurring early in the growth cycle, spreading quickly and causing severe damage [1]. The timely monitoring and prediction of the occurrence and development of BLS is of great significance to maintain rice production. The leaves are usually damaged by BLS and the traditional disease severity estimation of BLS depends on the lesion area as a proportion of the total leaf area. The estimation of leaf damage is largely dependent on the level of experience of agronomists and farmers, which is labor intensive and time consuming, and the estimations are often subjective and unreliable [2].

In the early studies of plant disease recognition, feature descriptors were designed to extract internal features from pictures obtained in the laboratory or the agricultural production environment using edge detection, color space transform, feature space transform theories, etc. [3–5]. These features were usually classified using a support vector machine (SVM) and linear discrimination, among others. For example, a prototype system for rice plant disease was proposed by Prajapati et al. [6], in which the background was removed and SVM was used to classify the types of diseases according to color, shape, and texture features of the digital pictures of infected rice. In another example, four types of local descriptors were compared in soybean leaf disease recognition and the effectiveness of

the proposed local descriptors, combined with the bag of visual words approach, were evaluated using classic 10-fold cross-validation and correct classification rate metric [7].

Deep learning methods have been applied successfully in image recognition [8] and semantic segmentation [9]. This technology has been used recently by Sethy et al. for plant disease recognition [10–12]. In addition, Lu et al. used a convolutional neural network (CNN) algorithm to identify 10 common rice disease types from 500 natural images of diseased and healthy rice leaves and stems [13]. There are some well-known proposed networks that have been investigated and show good potential and performance for plant disease recognition, e.g., MoblieNet and self-structured artificial neural network models, which were used to classify citrus disease images captured using smartphones [14]. Saleem et al. compared the classification performance of 19 CNN models applied in Plant Village Dataset, including AlexNet and OverFeat, the best CNN model was selected and the performance of the model was improved further by various deep learning optimizers, and the results showed that Xception combined with the Adam optimizer achieved the highest accuracy [15]. Kabir et al. implemented numerous popular CNN architectures and found that the Xception and DenseNet models performed better in multi-label plant disease classification [16].

Different agricultural environments have different characteristics, based on which Karlekar et al. proposed a two-stage soybean disease classification method named SoyNet. The background of images in SoyNet was excluded from the whole picture and then CNN was used to classify the soybean disease image [17]. Aiming to improve the accuracy of crop diseases classification, Nanehkaran et al. first segmented crop disease symptoms by color hue, saturation, and brightness, and then used CNN to classify disease types [18]. Sethy et al. used a transfer learning strategy and SVM to classify rice leaf diseases from the features extracted by eleven types of CNN models, and their results showed that the deep features plus SVM method outperformed the transfer strategy method [10]. Marino et al. introduced the unsupervised adversarial learning method into potato defect detection to overcome the shortcoming of accuracy deterioration caused by new labels. Their results showed that the domain self-adaption method can improve classification accuracy significantly [19].

Few studies have been carried out to date in the field of disease severity estimation. Typical examples of these include the study by Kalpana et al., in which the rice disease severity degree was predicted by using the fuzzy verdict method [20]. PD2 SE-Net was proposed to assist plant disease and severity automatic estimation based on CNN, inspired by the ResNet and ShuffleNet-V2 [21]. Esgario et al. designed a multi-task CNN method for the stress type classification of coffee leaves [22]. Prabhakar et al. used the ResNet-101 pre-trained in the Plant Village dataset to determine healthy, mild, middle, and serious disease severities for tomato early blight investigation [23]. These studies mostly built models directly to determine crop disease type and the level of severity. Even though these models were successful, they have certain shortcomings in fine quantitative evaluation and precise assessment, e.g., if the criteria changed, the model needs re-training due to the fact that directly building the relationship between disease severity and samples and direct model is lack of interpretation.

To solve the problems mentioned above, semantic segmentation theory has been applied in the study of agricultural disease severity estimation and other relevant agricultural applications. The primary task of semantic segmentation is to assign proper labels to every pixel in pictures so that it can acquire the disease lesion zone percentage required for disease severity estimation. The rice panicle segmentation approach based on the simple linear iterative super-pixel region generating (SLIS) technique was proposed by Xiong et al. The proposed method was robust to the field environment, degree of capture, crop growth factor, and other variables when compared with HSeg and other segmentation algorithms [24]. For different types of plants disease severity investigations, Kerkech et al. used a fully convolutional network (FCN) to fuse information to achieve pixel-wise segmentation for shadow, ground, healthy and diseased vine via registering visible and

near infrared vine imageries acquired by an unmanned aerial vehicle (UAV) [25]. Wang et al. used the VGG16 model and an encoder–decoder structure to segment diseased leaf spots in corn, and this overcame the influence of complex backgrounds and solved the labor-intensive problem of feature extraction template design [26]. Tusubira et al. evaluated Cassava brown streak (CBS) loss according to the segmented pixels number of necrotized and non-necrotized cells based on CBS lesion segmentation results [27]. In the area of weed identification, Abdalla et al. segmented images taken in the field and efficiently distinguished crops from weed species [28].

In general, the development of agricultural disease recognition and severity estimation undergoes two stages, i.e., the manual crafted features-based machine learning method and the deep learning-based method, both of which can be summarized as a classification problem. The former methods can be regarded as shallow classifiers, and the latter methods can be considered as deep classifiers according to the complex degree of the proposed models. After the data acquisition stage, e.g., by color cameras, mobile devices, and multispectral or hyperspectral imaging devices. The former methods need pre-processing, segmentation, feature extraction and classification modules [29]. A systematic summary about classical classification methods used in the first study stage of disease recognition and severity estimation was also given in this literature review. The performance of this type of method largely depends on the degree of feature extraction. Deep learning-based methods rises with the hardware computing power, which do not rely on manual feature extraction and have excellent generalization ability [30], having been applied in many types of crops disease recognition studies [31–33]. Wang et al. used apple black rots data from PlantVillage dataset; the data were further labeled into different disease levels by agronomist experts. The performance of disease classification accuracy of VGG16, VGG19, Inception-V3 and ResNet50 models were compared and found that VGG-16 combined with the transfer learning strategy achieved 90.4% accuracy on the test samples [34]. Verma et al. used tomato data from PlantVillage Dataset and label three stages (early, middle and end) for tomato late blight severity estimation study. Two types model design methods, i.e., transfer learning-based and feature extraction-based, were compared, and it was demonstrated that AlexNet achieved the best accuracy in both methods [35]. Although great progress has been made in the study of disease classification, existing research about agricultural disease classification mainly directly compared the performance of networks in the area of computer vision. Meanwhile, the number of disease levels used in the studies were small, possibly causing the proposed methods to estimate disease severity roughly and restrict the practical applications of the model. For further research on the fine classification of disease severity, Lin et.al used the semantic segmentation of deep learning to quantify the degree of cucumber powdery mildew through outperformed k-means, random forest and the GBDT method in 20 test samples [36]. For the specific rice disease recognition task, Shrivastava used a pre-trained convolutional neural network as feature extractor and support vector machine as a classifier, then classified four types of rice diseases taken in actual fields; the encouraging results showed that it can be used in the early warning of disease occurrence [37]. Chen et. al combined squeeze-and-excitation (SE block) with MobileNet to propose a model of rice diseases classification, which achieved 99.33% classification accuracy [38]. Das et. al firstly segmented the rice disease zone, then CNN was used to extract features and the dimension reduction method was used to reduce redundant features; finally, various classifiers were used to classify rice disease types [39]. Sethy et al. proposed a model for detecting rice false smut based on the Faster R-CNN object detection method [40]. Among studies of rice disease, most of them concern whether rice disease occurs or classifying multiply types of rice diseases; however, the number of studies about disease lesion segmentation and fine assessment of rice BLS severity estimation is still few, and the background of the analyzed images was usually simple. There is an urgent demand to design a better severity estimation model and improve the robustness of the proposed model for efficient plant protection. With the improvement of the performance of models, some research pays attention to the lightweight model of disease recognition,

which can be deployed in a mobile device. Elhasouny et al. proposed an effective and lightweight mobile application to classify and diagnose ten types of common tomato plant diseases [41].

The studies mentioned above concerning semantic segmentation applications in agriculture have made a beneficial step forward in high accuracy uses of the method. However, to the best of our knowledge, the research on rice BLS lesion segmentation and disease severity estimation is still limited, which is of great significance to the crop protection of rice and the increase in rice yield.

Based on the above discussion, in this study, a new model named BLSNet for rice BLS lesion recognition and segmentation is proposed. The main contribution of this paper can be stated as follows: (1) this is the first step for the evaluation of the capacity of deep learning semantic segmentation for rice BLS lesion segmentation; (2) it fills the research gap of rice BLS lesion segmentation and fine disease severity estimation. The lesion area value can be drawn for BLS disease severity assessment based on the segmentation results. The model was trained and validated using real images captured in the production environment to ensure real application utility in the future. Preliminary studies suggest that the proposed method is effective in estimating rice BLS severity.

2. Materials and Methods

2.1. Data Collection

Data were collected in the fields around Shanji, Xuzhou, JiangSu Province, China (117°34′22″ N, 34°13′58″ E, 30 m above sea level), on 18–19 September 2020 at 10:00–16:00 under different sun light intensities and capture angles. A portable digital camera (D850, Nikon Inc., Tokyo, Japan) was used to capture RGB pictures of paddy rice. Most of the collected lesion images were at the growth stage and ranged from small water-spot-like short spots to vertical long stripes. The initial size of the images was 8256 × 5504 pixels (Figure 1a). Due to the fact that the initial size of one image was large, data were first cropped by the central area determined by the actual size of the 1 × 1 m white box shown in the images (Figure 1b). Then, after randomly cropping the central area, the final size of the images used to train and validate BLSNet was 256 × 256 pixels. The dataset consisted of 109 images, which were labeled as BLS lesion, rice leaves, and three background classes using LabelMe software [42]. Figure 2 shows the data labels.

(a)   (b)

Figure 1. Images showing (**a**) initial sample data and (**b**) cropped sample data determined by the 1 × 1 m white box.

To avoid model overfitting, data augmentation methods were used through random vertical and horizon flip, contrast stretch, brightness stretch, sharpness stretch, salt and pepper noise, Gaussian noise, random rotation (rotation angle = 0–360°), random resize (resize factor = 0.1–1.0), and random blur. These data enhancements were performed using open source OpenCV [43], and Python Image Library (PIL) [44]. A total of 1199 images were acquired using the above data augmentation techniques, ~80% of which were randomly selected as a training dataset, with the remaining ~20% used for model validation.

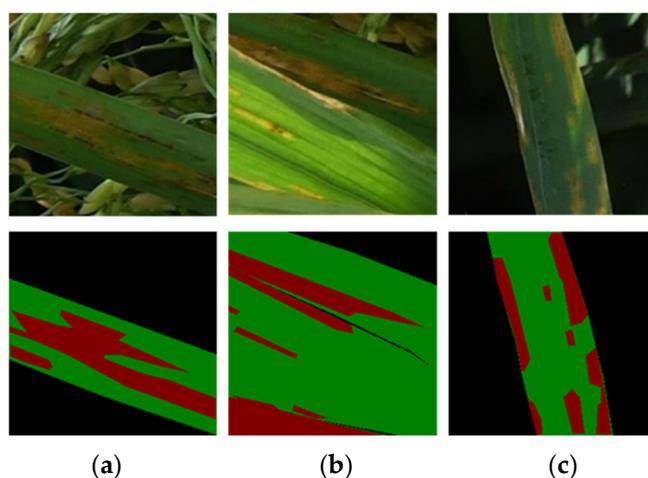(**a**)   (**b**)   (**c**)

Figure 2. Image types and their corresponding labels, (**a**), (**b**) and (**c**), give three examples of the samples used for training and validation (red, green, and black stand for BLS lesion, leaves, background, respectively).

2.2. Disease Severity Label

The disease severity label determination was based on the BLS lesion and the proportion of the BLS lesion of the total leaf area was used in the process. Level 0 included leaves with no lesion; level 1 with below 10% lesion; level 2 = 11–25% lesion; level 3 = 26–45% lesion; level 4 = 46–65%; level 5 = >65% accordingly. Due to the data collection time constraints, BLS was in the intermediate or more serious disease severity classes, so the sample sizes of levels 1 and 2 were relatively small.

For this study, as the trained data were randomly partitioned from the initial data, the proportion of the lesion in the total leaves of each sample was used to investigate disease severity. Table 1 shows the disease severity data distribution in the training and validation datasets.

Table 1. Disease severity data distribution.

Disease Severity Level	Training	Validation
1	34	11
2	109	18
3	351	79
4	280	78
5	194	45
Total	968	231

2.3. Model Architecture

CNN blocks are usually used in semantic segmentation models because CNN can extract and exploit local contextual features. The attention mechanism and multi-scale features methods can discriminate parts and make full use of local and global information, improving the efficiency and accuracy of the model training process. Therefore, both the attention mechanism and multi-features were used in BLSNet for rice BLS lesion segmentation in our research.

Based on the encoder–decoder architecture used in UNet [45], our BLSNet was designed to have five stages or blocks; namely, a down-sample stage, up-sample stage, skip connection with an attention block(SE block), atrous spatial pyramid pooling (ASPP) block [46], and multi-scale spatial channel attention block (MSC block). Figure 3 presents the overall network model architecture.
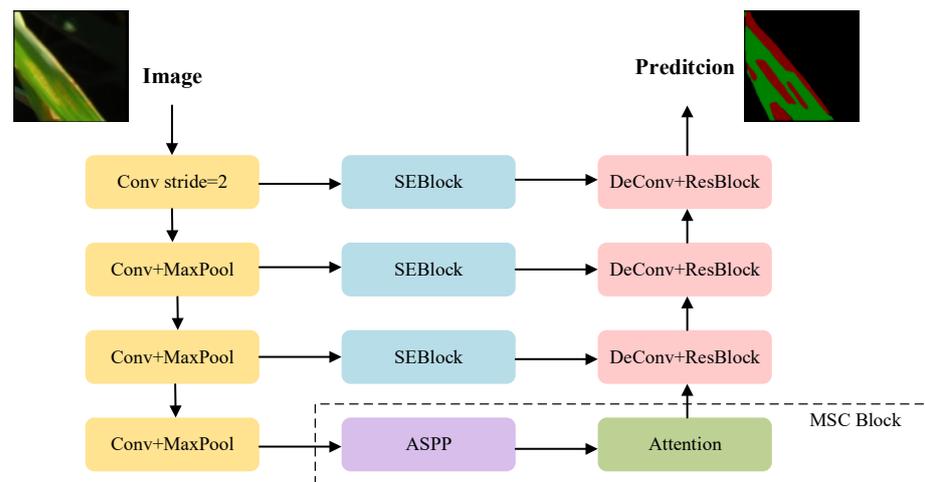
Figure 3. Overall architecture of BLSNet. Conv stride = 2 represents the down-sample block with convolution operation of kernel size of 3 and stride size of 2. Conv + MaxPool stands for the down-sample block composed of convolution operation of kernel size of 3, followed by a maximum pooling of kernel size of 2. DeConv + ResBlock stands for the up-sample block, i.e., DeConv is convolutional transposed operation of kernel size 3, ResBlock uses residual connection. The detailed implementation process is described in the following sections.

Similar to the architecture of UNet, BLSNet was divided into three parts: left, middle, and right. Feature extraction and semantic feature extraction were achieved in the left part. Four down-sample blocks were included in the feature extraction process, while skip connection with attention machine was used in the middle part. According to the level of feature complexity, two types of attention blocks, i.e., SE block and MSC block, were used in BLSNet. Shallow, simple, and edge features were obtained in the 1st, 2nd, and 3rd down-sample block. The simple channel attention block was used to process the shallow, simple, and edge features. High level semantic features were extracted during the 4th down-sample block and combined with the ASPP and MSC blocks. The right part of the BLSNet was the decoded part, in which different level and scale information flows were combined, and picture size was recovered to the original size of 256 × 256 pixels gradually from bottom to top. Convolution blocks used in the whole model were composed of combined operations of convolution, normalization, and activation to improve training performance.

2.3.1. Down-Sample Stage

The down-sample stage was composed of four down-sample blocks, after which the output size was half of the input size, meaning the final features were one sixteenth of the original size. In the first down-sample phase, a convolution of kernel size 3 and stride 2 was adopted to avoid the edge information being ignored for the purpose of maximum pooling adoption in the UNet. In the second, third, and fourth down-sample phases, a convolution of kernel size 1 was used to integrate information, and a maximum pooling operation of stride 2 was used to down-sample and extract salient features. During the down-sample block procedure, an increasing amount of contextual information was drawn as the number of down-samples increased.

2.3.2. Skip Connection Stage with Attention Block

As is shown in Figure 4 [47], in the skip connection stage, feature maps obtained from the 1st, 2nd, and 3rd down-sample stages were processed with the squeeze and excitation attention blocks (SEBlock). The size of feature maps extracted in the 4th down-sample stage containing rich semantic information was 1/16 of the initial input image that contains

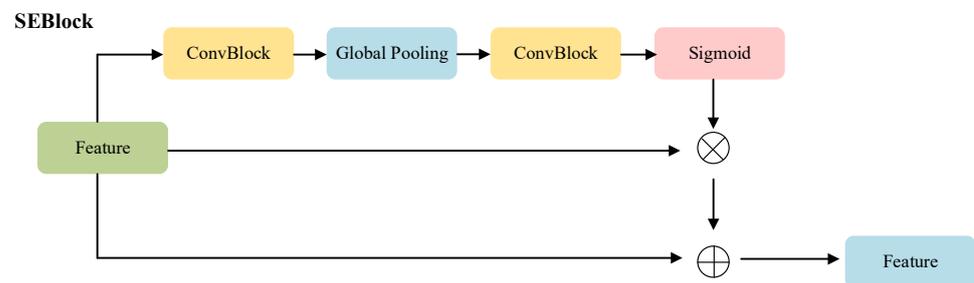rich semantic information, so the MSCBlock was utilized to process the 4th down-sample stage features.

Figure 4. Illustration of the SEBlock.

The SEBlock was composed of three branches: the upper branch generated the weight of every channel, i.e., the channel weights were extracted through the convolutional block and global pooling, followed by a convolution block comprising of two 1×1 operations of convolution. The sigmoid activation function was then used to scale channel weights to a range of 0 to 1. The output features of the SEBlock were the summary of the initial features dotted with the channel weight and the original features.

The MSCBlock consisted of the ASPP block and a spatial channel attention machine.

Figure 5 shows the information flow process of the MSCBlock, in which the higher semantic features are first filtered with ASPP to integrate multi-scale features and are then separately fed into the two branches (i.e., the position attention module (PAM) and channel attention module (CAM)). Figure 6 shows the structure of PAM and Figure 7 presents the diagram of CAM. Through both PAM and CAM branches, the position-enhanced and channel-enhanced features were then added, and a convolutional block with 1×1 convolutional operation was used to reduce channel numbers and refine the semantic contextual content to acquire the final high-level semantic features.

In the PAM branch, the features extracted by ASPP were first processed with the convolutional block, after which three 1×1 convolutional branches were used in the position attention module (PAM) to process the features and generate key feature map B, query feature map C, and value feature map D. The shape of feature maps B and C was then altered and dotted to generate spatial feature relation matrix E. Feature map D was reshaped to operate with E and generate enhanced position feature maps. The outputs of the PAM were the summary of enhanced position feature map and original features.
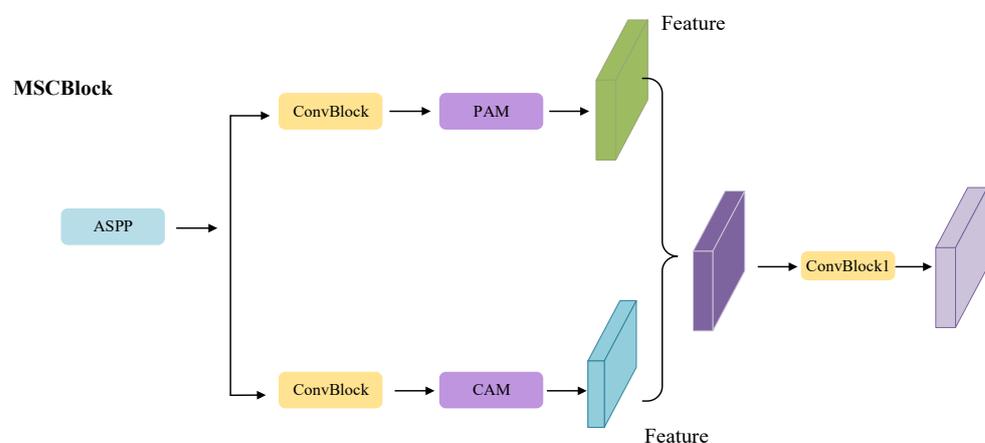

Figure 5. MSCBlock flow diagram. ConvBlock stands for convolution block and is composed of convolutional operation of kernel size 3, batch normalization operation, and Relu operation. ConvBlock1 represents a convolutional block composed of kernel size 1, batch normalization operation, and Relu operation.
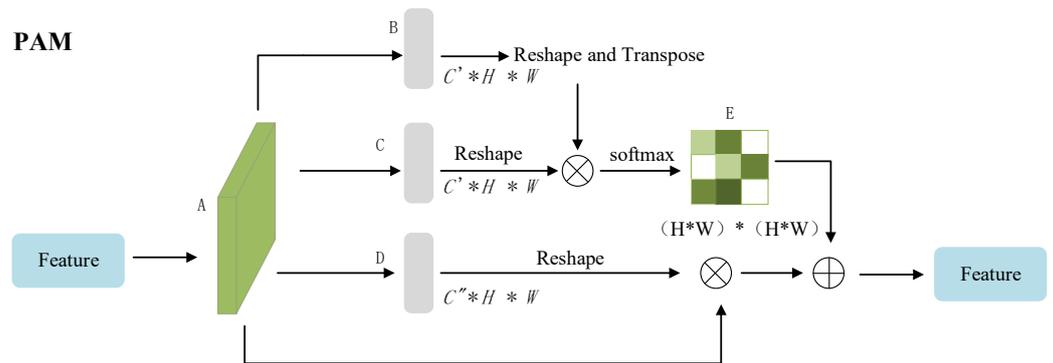
PAM

Figure 6. The structure of the PAM. B = key feature maps, of which dimension is $C'*H*W$, C = query feature maps, of which dimension is $C'*H*W$, D = value feature maps, of which dimension is $C''*H*W$.
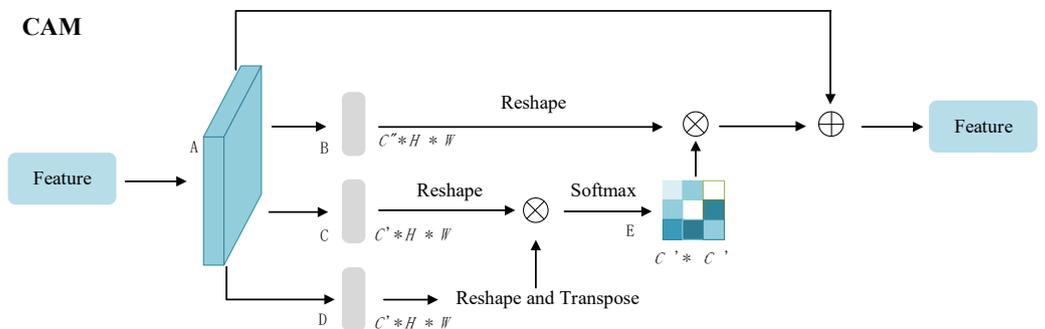
CAM

Figure 7. The structure of the CAM. B = value feature maps, of which dimension is $C''*H*W$, C = query feature maps, of which dimension is $C'*H*W$, D = key feature maps, of which dimension is $C'*H*W$.

Similar to the design of the PAM branch, after the convolutional block three 1×1 convolutional branches were used in the channel attention module (Figure 7) to generate value feature maps B, query feature maps C and key feature maps D. Channel correlation feature maps E were obtained by C and D. Specifically, the shapes of C and D were first altered and matrix multiplication was performed. The final enhanced position feature maps were acquired using matrix multiplication of E with feature maps B. The outputs of the CAM were the summary of the enhanced position feature map and original features.

The ASPP block (Figure 8) consisted of five branches, four of which were two-dimensional dilation convolutions with a kernel size of 3, and dilation rates of 1, 3, 6, and 9, respectively; the last branch utilized average pooling to obtain global semantic information.
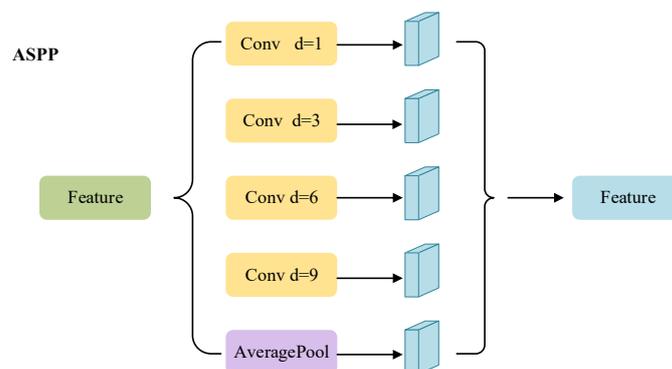

Figure 8. The ASPP block. Conv d means convolution blocks with different dilation rates.

2.3.3. Up-Sample Stage

As the size of the final feature map was 1/16 of the initial feature map, to recover the same size of the input images, an up-sample operation for the feature maps was required. The up-sample stage of the proposed network had four up-sample blocks, with the 1st, 2nd, 3rd, and 4th up-sample stages corresponding to the respective down-sample stages. During the up-sample stage, from bottom to top, a deconvolution of stride 2 was first used to double the size of the highest-level feature maps generated from the MSCBlock. Then, in the 3rd, 2nd, and 1st up-sample blocks, features enhanced by a skip connection stage were concatenated with up-sampled features. Then, a residual operator composed of a convolution of kernel size 3 and kernel size 1 was used to process the concatenated features to prevent the network performance from degradation. After four up-sample operations, the size of the feature map was the same as the input image. A convolution of kernel size 1 was used to make the final channel size the same as the category numbers to be distinguished. Each pixel label was assigned by the indices of the maximum values among the possibilities in the three channels.

2.4. Experimental Process

Our model was implemented in the PyTorch deep learning framework. The model was trained using 16 GB of memory powered by a Tesla V100 Graphics Processing Unit (GPU) under Ubuntu18.04 LTS system environment. The number of batch sizes for training and validating was set to 32, according to the capacity of the GPU and the size of sample images. A "*Loss*" value was computed according to the multi classes cross entropy loss function, calculated in Equation (1), where y_n indicates the class label, $\hat{y}$ represents the predicted class label, and K represents the number of total categories. The parameters of the proposed model were updated using backward gradient propagation algorithm.

$$Loss = -\frac{1}{K} \sum_{n=1}^{K} [y_n log \hat{y}] \tag{1}$$

The model was trained for a maximum of 300 epochs, and the performance on the validation dataset was recorded after every two training epochs. Adam optimizer was chosen as the network optimizer. The initial learning rate was set to 0.001, and the poly learning rate strategy was used to update the learning rate dynamically. For parameter initialization, the parameters of the convolutional layer were initialized using the Kaiming initialization method, while other parameters were set using the default values provided in the PyTorch framework.

The BLSNet was compared with UNet and DeepLabv3+ based on segmentation accuracy, pixel accuracy, BLS lesion segmentation accuracy, rice segmentation accuracy, background segmentation accuracy, and the mean interaction over union (mIoU). The mIoU was the ratio of the intersection area and the total area between the input label mask and the predicted result mask, calculated using Equation (2). The mIoU reflects the quality of the segmentation on a scale of 0 to 1, where 1 means the predicted result was exactly the same as the label, which is the ideal situation for semantic segmentation. In Equation (2), k represents the total number of categories, which in this case was set to 3, and p_{ij} represents the number of categories i predicted as category j.

$$mIoU = \frac{1}{k+1} \sum_{i=0}^{k} \frac{p_{ii}}{\sum_{j=0}^{k} p_{ij} + \sum_{j=0}^{k} p_{ij} - p_{ii}} \tag{2}$$

Pixel accuracy (PA) represents the proportion of all the correctly predicted pixels to the total pixels, as is shown in Equation (3). k indicates the number of classes, and p_{ii}

represents the corrected predicted pixels, p_{ij} represents the number of categories i predicted as category j.

$$PA = \frac{\sum_{i=0}^{k} p_{ii}}{\sum_{i=0}^{k} \sum_{j=0}^{k} p_{ij}} \tag{3}$$

Class accuracy (CA) was usually used to measure the accuracy of each category prediction results and was computed using Equation (4):

$$CA = \frac{p_{ii}}{\sum_{j=0}^{k} p_{ij}} \tag{4}$$

3. Results

3.1. Training Results

Figure 9 shows the training accuracy curves from the mIoU and different category segmentations. All the criteria for the BLSNet model were the highest at the beginning, and the number of epochs required to achieve the convergence of BLSNet was the smallest. BLSNet showed an overall upward trend in the training and validation process, proving the stability of the method. By contrast, in the training process of UNet, two obvious oscillations were observed that reflect the fluctuation of UNet, which is not acceptable in practical applications.
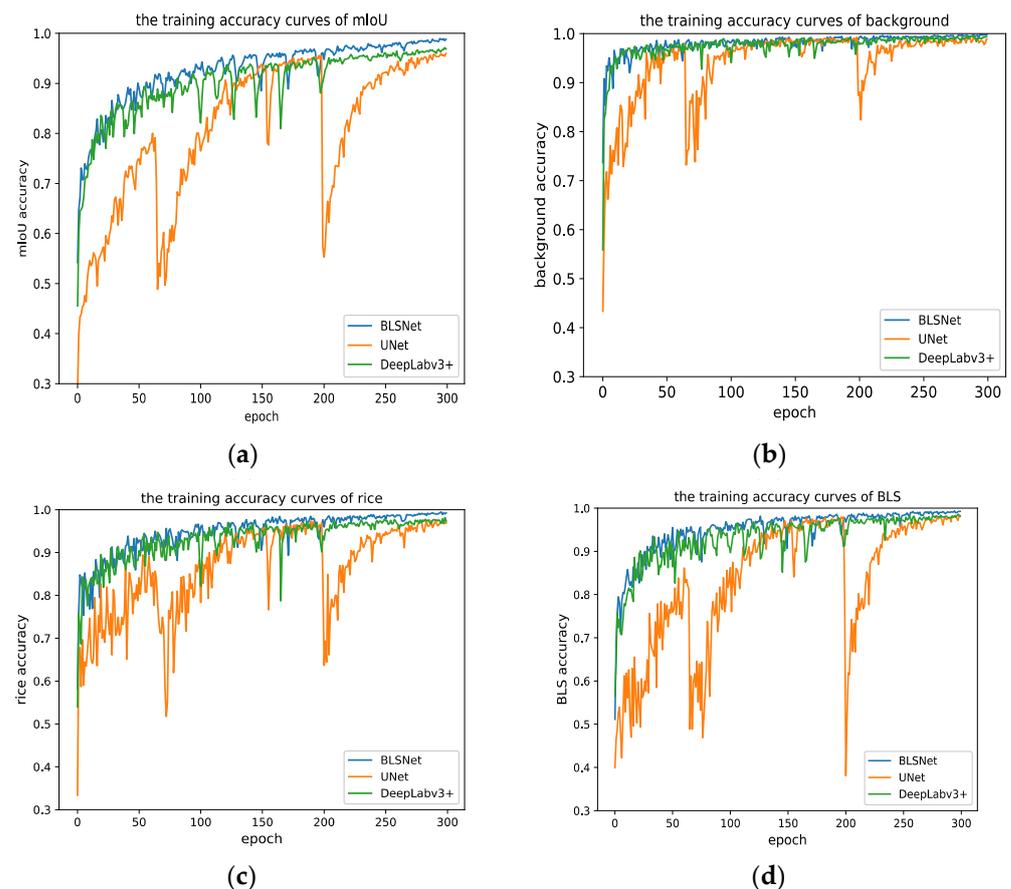

Figure 9. Training accuracy curves of different models. (**a**) The training curves of mIoU of three models; (**b**) the training curves of segmentation accuracy of background of three models; (**c**) the training curves of segmentation accuracy of rice of three models; (**d**) the training curves of segmentation accuracy of BLS of three models.

3.2. Validation Results

In the training process, the performance of the trained models used in the validation dataset was recoded every two epochs. From Figure 10, it can be seen that compared with the UNet and DeepLabV3+ methods, the change of BLSNet was smallest, indicating the stability and transfer capacity of this method.
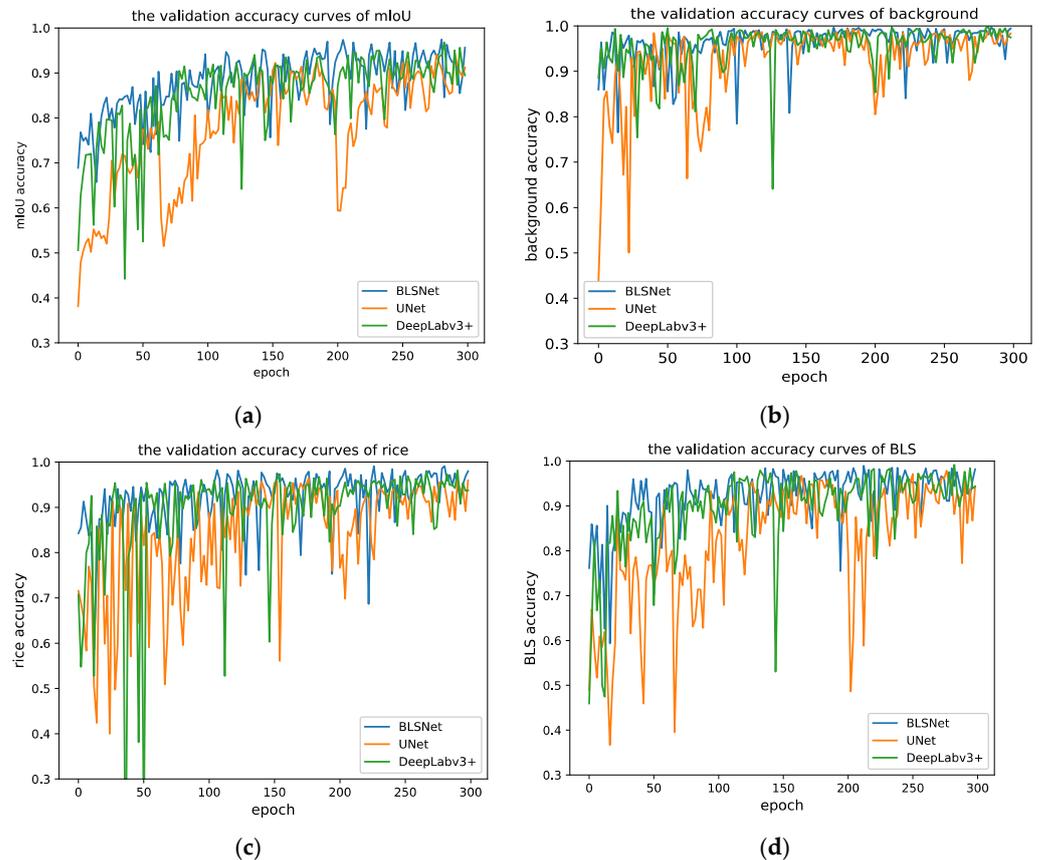

Figure 10. Validation curves of criteria for different models. (**a**) the validation curves of mIoU of three models; (**b**) the validation curves of segmentation accuracy of background of three models; (**c**) the validation curves of segmentation accuracy of rice of three models; (**d**) the validation curves of segmentation accuracy of BLS of three models.

The mIoU and CA values from the three models are listed in Table 2 for comparison. BLSNet performed better than UNet and DeepLabV3+ in terms of mIoU, and the segmentation accuracy of the BLS disease lesion, rice leaves and background outperformed the next best method by 4%, 4%, 2%, and 1% in terms of accuracy, respectively.

Table 2. Accuracy comparison of the proposed BLSNet model with UNet and DeepLabV3+.

Indicators	UNet	DeepLabV3+	BLSNet
mIoU	0.912	0.895	**0.956**
Background	0.983	0.975	**0.994**
BLS	0.944	0.944	**0.982**
Rice	0.959	0.938	**0.980**

The bold in each row in the table is the highest accuracy achieved by model.

4. Discussion

4.1. Ablation Study on Attention Block

Ablation studies were performed to investigate the effectiveness of each attention block used in the skip connection stage. In this experiment, the first, second, third, and

fourth attention blocks in the skip connection were removed separately, and the first, second, and third attention blocks were removed together. For mIoU, the segmentation accuracy of the background, rice, and BLS were compared independently (Table 3), and we found that the performance of segmentation by applying ablation models was mostly worse than the highest accuracy achieved in BLSNet, demonstrating the effectiveness of the attention mechanism in the segmentation process.

Table 3. Accuracy comparison in ablation study when removing attention blocks.

Indicator	BLSNet	Without1	Without2	Without3	Without4	Without1–3
mIoU	**0.956**	0.949	0.952	0.951	0.935	0.957
Background	**0.982**	0.979	**0.982**	0.978	0.957	0.980
BLS	**0.982**	0.979	**0.982**	0.978	0.957	0.980
Rice	**0.980**	0.974	**0.980**	0.976	0.968	**0.980**

Without1 stands for the ablation BLSNet model which removed the first attention block in the skip connection stage, without2 stands for removing the second attention block in BLSNet, without3,4,1–3 represent for removing the third, the fourth, the first to the third attention block. The bold in each row in the table is the accuracy achieved by the BLSNet model and the highest accuracy achieved by ablation model.

The performance of the ablation BLSNet and the initial BLSNet model in terms of the mIoU was evaluated. The lowest accuracy was obtained in the model without the fourth attention block, and a possible reason for this is that the fourth attention block was mainly effective in dealing with high-level semantic features containing rich semantic contextual and spatial information. This attention block was also useful in dealing with distinguishable features for further segmentation. This finding is beneficial to the further development of the model architectural design for agricultural diseases segmentation; that is, it can pay more attention on a full utilization of high level semantic and contextual features.

The second attention block did not significantly improve the performance of the BLSnet model. A possible reason for this is that the channel number of the output features of the second down-sample block was 128, which has already distinguished the intermediate features reasonably well from the low-level of edge features and the high-level of semantic features.

Comparing the accuracy of different classes achieved by the ablation studies listed in Table 3, it can be seen that the background accuracy was not obviously influenced by removing attention blocks. A possible reason for this is that the background occupies a relatively large area in an image, so the attention blocks did not work efficiently. The accuracy of the BLS segmentation and rice segmentation is apparently decreased by about 3% when the attention blocks were not included, especially the fourth attention block. This verifies the effectiveness of using the MSC block to distinguish high-level semantic features.

4.2. Prediction Results Comparison

To investigate the segmentation performance of the BLSNet, UNet, and DeepLabV3+ models, the images and results of their segmentation are presented in Figure 11. In Figure 11, the first-row sample under dark light conditions was processed using the flip augmentation operation. The segmentation effects of the first-row sample suggest that the performance of all models was similar. The performance of the BLSNet segmentation was relatively consistent and tended to be complete. The second-row sample was augmented by an image rotation operation, and it is clear that some of the leaves' segments are missing in comparison with those in the UNet and DeepLabv3+ models. Again, the segmentation performed by BLSNet tended to be complete, indicating its ability to perform well even when sample rotation occurs. This characteristic is desirable for disease segmentation in the natural agricultural production environment. The third-row sample is enhanced by a scale transformer and the detailed features of lesion segmentation from the BLSNet model were better than those produced by UNet and DeepLabV3+. This suggests that BLSNet is better suited for the adaptation of scale changes of images. In the fourth row of Figure 10, it can be seen that the detailed edges of BLS can be segmented well by the BLSNet model, which

is consistent with the conclusion obtained in the lesion ablation studies, i.e., attention blocks can significantly improve detailed segmentation. In the fifth row of Figure 10, panicles were misclassified as BLS by the BLSNet model. A possible reason for this is that the characteristics of the panicle were similar to the color characteristics of the BLS disease spots, so the network may not be able to discriminate between the fine differences of disease spots and panicles, which ultimately leads to misclassifications.
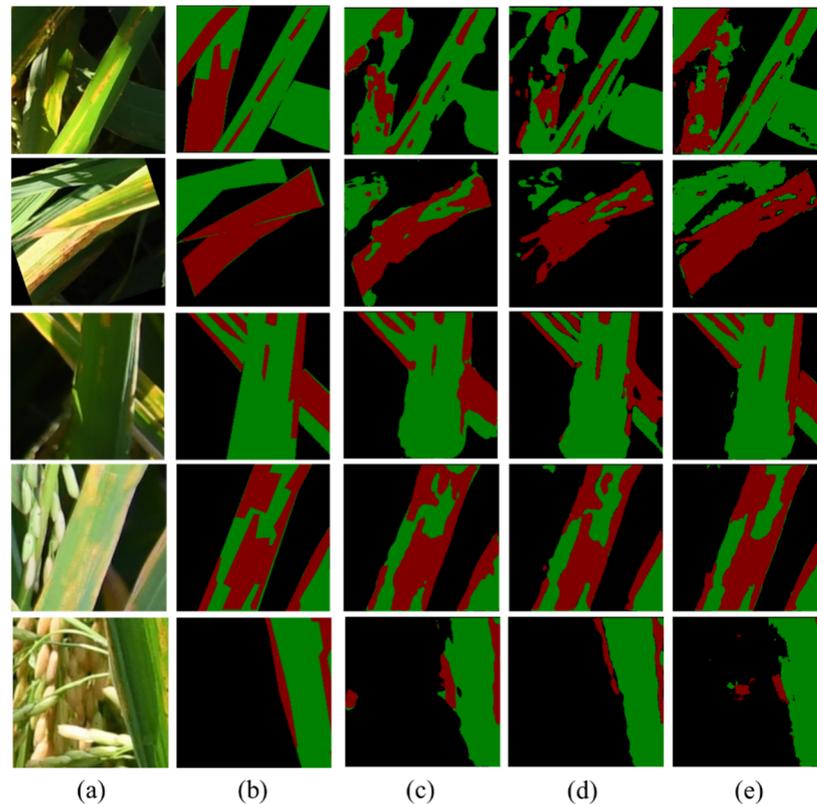

     (a)   (b)   (c)   (d)   (e)

Figure 11. Comparison of prediction results for different models: (**a**) image (**b**) label; (**c**) UNet segmentation result; (**d**) DeepLabV3+ segmentation result; and (**e**) BLSNet segmentation result. Each row lists a representative sample, corresponding rice, BLS, and background label, the segmentation result of UNet, DeepLabv3+, and BLSNet.

4.3. Comparison of Model Prediction Time

To compare the prediction time of different models, the saved parameter files of UNet, DeepLabV3+, and the proposed BLSNet were used to load the trained models for the segmentation of validation dataset images. The segmentation time of each image in the validation dataset was calculated by averaging the prediction time of all images. Table 4 shows that the prediction time for BLSNet was slightly longer than UNet, shorter than DeepLabV3+, but the segmentation accuracy achieved by the BLSNet was the highest. This suggests that the BLSNet is well-suited for real-time BLS lesion segmentation.

Table 4. Model prediction time comparison.

Model	UNet	DeepLabV3+	BLSNet
Prediction Time (s)	0.020	0.028	0.021

4.4. Disease Severity Estimation Comparison

Precise areas of the disease lesion that are needed for the investigation of disease severity can be extracted based on the segmentation results. The trained models were used to segment the validation dataset, and the percentage of the BLS diseased lesion area

in a sample was then recorded to obtain the disease severity in each image according to the criteria outlined in Section 2.2. To compare the effectiveness of BLSNet, UNet, and DeepLabV3+ in determining different levels of BLS disease severity, confusion matrices were used, as shown in Figures 12–14. The classification accuracy of different levels of disease severity is shown in Table 5. In confusion matrices, the number in each row represents the number of the predicted category corresponding to the true category. It can be seen that for level 2 damage severity, UNet achieved the highest accuracy, while the best results of disease severity level determination were achieved by BLSNet for the other four levels. As for the overall classification accuracy, BLSNet was higher than UNet and DeepLabV3+.
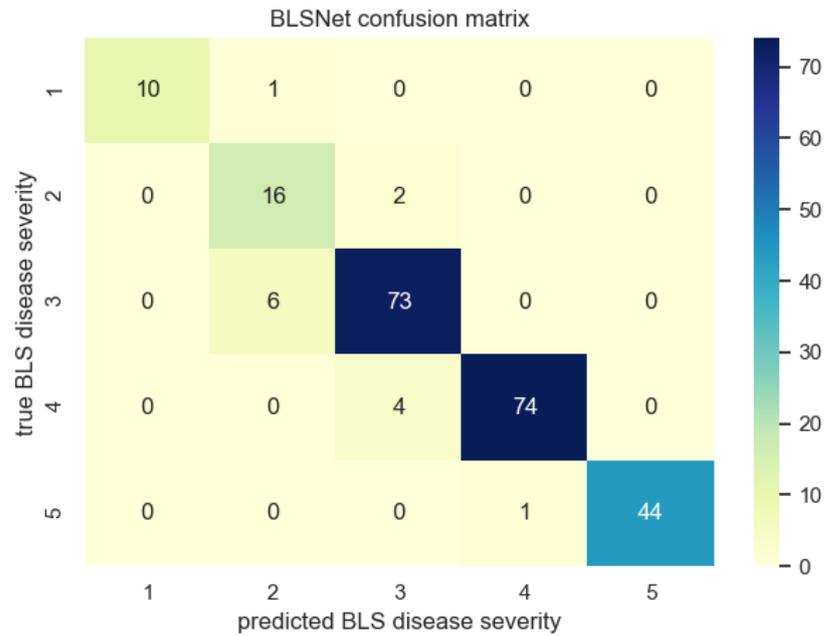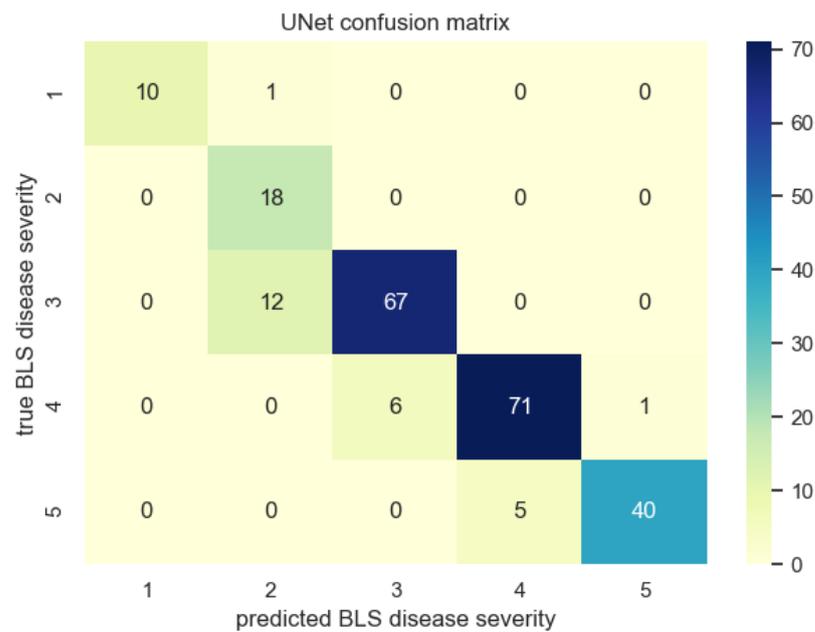

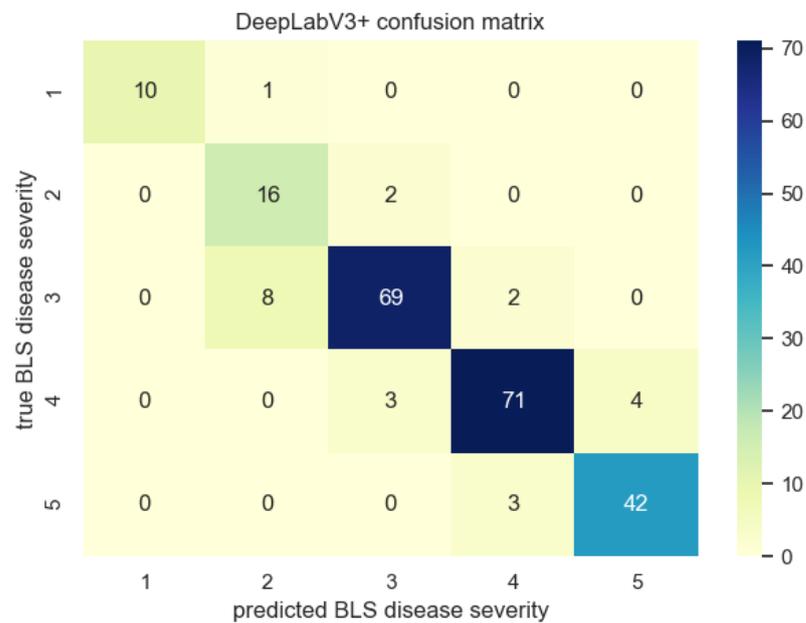**Figure 12.** BLSNet confusion matrix.

Figure 13. UNet confusion matrix.

Figure 14. DeepLabV3+ confusion matrix.

Table 5. Disease severity level estimation accuracy comparison.

Severity Level	BLSNet	UNet	DeepLabV3+
1	**0.91**	0.91	0.91
2	0.89	**1.00**	0.89
3	**0.92**	0.85	0.87
4	**0.95**	0.91	0.91
5	**0.95**	0.89	0.93
Average accuracy	**0.94**	0.89	0.90

The bold in each row in the table is the highest accuracy achieved by model.

To improve our understanding of the causes for such a phenomenon, misclassification problems were further analyzed. A total of 14 samples were misclassified by BLSNet, among which the disease severity of 11 samples was underestimated. This underestimation problem is likely due to the fact that lesion segmentation zones tended to be clustered due to the attention mechanism. This led to the estimated area of disease infection being smaller than the actual area. Twenty-three samples were misclassified by the DeepLabV3+ model correspondingly. The possible reason for this misclassification was that the receptive fields of ASPP used in DeepLabV3+ were large and lacked sufficient attention blocks, increasing the difficulty of efficiently detecting BLS lesion zones. It is clear that the performance of different levels of disease classification differed in the UNet model: the accuracy of classification for levels 2 and 3 were 1 and 0.85, respectively, demonstrating that UNet was not always able to determine BLS disease severity well at certain levels. The results obtained from the disease severity investigation suggest that the proposed BLSNet model could be used in practical applications to automatically determine BLS disease severity with a high level of accuracy.

Although higher classification accuracy was achieved by the BLSNet in BLS severity estimation overall, the accuracy of level 1 and 2 determination was lower than that of levels 3–5. The capability to identify low level BLS disease severity was limited, i.e., segmenting small lesion zones needs to be improved in BLSNet. The performance of the proposed model can be improved through the amount of data used, by using a higher volume of different level BLS disease severity data. A priori knowledge of plant protection and other relevant agriculture information, as well as full usages of computer vision, could help with the design and implementation of an effective severity estimation process. In addition,

automatic labeling, semi-supervised training, and assisted labeling methods could be used to improve the speed of agricultural sample processing, helping to overcome the workload problem of manual semantic labeling.

5. Conclusions

This paper proposed BLSNet for rice BLS lesion segmentation and disease severity estimation based on the semantic segmentation technique and attention mechanism and multi-scale methods. The BLSNet was trained by images taken in the actual fields. The accuracy of segmentation of the BLSNet was the highest compared with benchmark DeepLabV3+ and UNet model. In the case of level 5 disease severity, the accuracy of classification of the proposed model was higher in level 4, demonstrating the effectiveness of disease severity estimation of the proposed models.

The highlight of this paper is the application of the semantic segmentation of the deep learning method into the rice BLS segmentation and severity estimation. To our knowledge, this is the first work to take advantage of the pixel-wise classification of semantic segmentation to obtain fine rice BLS lesion areas, which are used for disease severity determination. The result of comparative experiments proved the introduction of the attention mechanism and multi-scale features aggregation can improve disease classification accuracy; the finding of the work provides references for the deep learning architecture design of other relevant plant disease classification.

It is worth noting that there are still some limitations in this method. The accuracy of low level disease degree estimation needs improving. Due to the restriction of data collection time and experimental site, only one type, i.e., BLS rice disease segmentation and severity estimation, was studied. In the future study of lesion segmentation and disease severity level estimation, an advanced segmentation model can be developed for more types of plants, more types of diseases and more levels of disease severity. Multiple types of data sources can also be considered, e.g., multispectral and hyperspectral data of which spectral characteristics can be used to show the growth state of crops, and aerial imagery which be used for effective plant diseases monitoring in the large scale. In theory, the accuracy of the model can be improved by increasing the amount of data; the data crowdsourcing method can be used to label adequate crop disease samples. In terms of the size of the model, work can also be carried out to reduce the parameter amount of the model and the proposed lightweight model for mobile and IoT devices in the actual field.

Author Contributions: Conceptualization, S.C., Y.Z.; methodology, S.C., W.B.; software, S.C., Y.C.; validation, S.C., Y.S.; formal analysis, S.C., W.B. and Y.Z.; investigation, S.C., H.Z.; resources, S.C., J.L., X.Z., T.Y.; data curation, S.C., T.Y.; writing—original draft preparation, S.C., K.Z.; writing—review and editing, S.C., K.Z.; visualization, X.Z., J.L.; supervision, K.Z.; project administration, K.Z. and Y.S. 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: The data presented in this study are available within the article.

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

References

1. NIÑO-LIU, D.O.; Ronald, P.C.; Bogdanove, A.J. Xanthomonas oryzae pathovars: Model pathogens of a model crop. *Mol. Plant Pathol.* **2006**, *7*, 303–324. [CrossRef]
2. Joshi, A.A.; Jadhav, B. Monitoring and controlling rice diseases using image processing techniques. In Proceedings of the 2016 International Conference on Computing, Analytics and Security Trends (CAST), Pune, India, 19–21 December 2016; pp. 471–476.
3. Wang, L.; Yang, T.; Tian, Y. Crop disease leaf image segmentation method based on color features. In Proceedings of the International Conference on Computer and Computing Technologies in Agriculture, Boston, MA, USA, 18–20 October 2008; pp. 713–717.
4. Revathi, P.; Hemalatha, M. Homogenous segmentation based edge detection techniques for proficient identification of the cotton leaf spot diseases. *Int. J. Comput. Appl.* **2012**, *47*, 18–21. [CrossRef]
5. Zhang, M.; Meng, Q. Automatic citrus canker detection from leaf images captured in field. *Pattern Recognit. Lett.* **2011**, *32*, 2036–2046. [CrossRef]
6. Prajapati, H.B.; Shah, J.P.; Dabhi, V.K. Detection and classifcation of rice plant diseases. *Intell. Decis. Technol.* **2017**, *11*, 357–373. [CrossRef]
7. Pires, R.D.L.; Gonçalves, D.N.; Oruê, J.P.M.; Kanashiro, W.E.S.; Rodrigues, J.F.; Machado, B.B.; Gonçalves, W.N. Local descriptors for soybean disease recognition. *Comput. Electron. Agric.* **2016**, *125*, 48–55. [CrossRef]
8. 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.
9. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
10. Sethy, P.K.; Barpanda, N.K.; Rath, A.K.; Behera, S.K. Deep feature based rice leaf disease identification using support vector machine. *Comput. Electron. Agric.* **2020**, *175*, 105527. [CrossRef]
11. Singh, U.P.; Chouhan, S.S.; Jain, S.; Jain, S.; Jain, S. Multilayer convolution neural network for the classification of mango leaves infected by anthracnose disease. *IEEE Access* **2019**, *7*, 43721–43729. [CrossRef]
12. Yin, H.; Gu, Y.H.; Park, C.; Park, J.; Yoo, S.J. Transfer learning-based search model for hot pepper diseases and pests. *Agriculture* **2020**, *10*, 439. [CrossRef]
13. Lu, Y.; Yi, S.; Zeng, N.; Liu, Y.; Zhang, Y. Identification of rice diseases using deep convolutional neural networks. *Neurocomputing* **2017**, *267*, 378–384. [CrossRef]
14. Barman, U.; Choudhury, R.D.; Sahu, D.; Barman, G.G. Comparison of convolution neural networks for smartphone image based real time classification of citrus leaf disease. *Comput. Electron. Agric.* **2020**, *177*. [CrossRef]
15. Saleem, M.H.; Potgieter, J.; Arif, K.M. Plant disease classification: A comparative evaluation of convolutional neural networks and deep learning optimizers. *Plants* **2020**, *9*, 1319. [CrossRef] [PubMed]
16. Kabir, M.M.; Ohi, A.Q.; Mridha, M. A Multi-plant disease diagnosis method using convolutional neural network. *arXiv* **2020**, arXiv:2011.05151.
17. Karlekar, A.; Seal, A. SoyNet: Soybean leaf diseases classification. *Comput. Electron. Agric.* **2020**, *172*, 105342. [CrossRef]
18. Chen, J.; Chen, J.; Zhang, D.; Sun, Y.; Nanehkaran, Y.A. Using deep transfer learning for image-based plant disease identification. *Comput. Electron. Agric.* **2020**, *173*, 105393. [CrossRef]
19. Marino, S.; Beauseroy, P.; Smolarz, A. Unsupervised adversarial deep domain adaptation method for potato defects classification. *Comput. Electron. Agric.* **2020**, *174*, 105501. [CrossRef]
20. Kalpana, M.; Karthiba, L.; Kumar, A.V.S. Disease severity diagnosis for rice using fuzzy verdict method. In *New Trends in Computational Vision and Bio-Inspired Computing: Selected Works Presented at the ICCVBIC 2018*; Coimbatore, I., Smys, S., Iliyasu, A.M., Bestak, R., Shi, F., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 873–880.
21. Liang, Q.; Xiang, S.; Hu, Y.; Coppola, G.; Zhang, D.; Sun, W. PD2SE-Net: Computer-assisted plant disease diagnosis and severity estimation network. *Comput. Electron. Agric.* **2019**, *157*, 518–529. [CrossRef]
22. Esgario, J.G.M.; Krohling, R.A.; Ventura, J.A. Deep learning for classification and severity estimation of coffee leaf biotic stress. *Comput. Electron. Agric.* **2020**, *169*, 105162. [CrossRef]
23. Prabhakar, M.; Purushothaman, R.; Awasthi, D.P. Deep learning based assessment of disease severity for early blight in tomato crop. *Multimed. Tools Appl.* **2020**, *79*, 28773–28784. [CrossRef]
24. Xiong, X.; Duan, L.; Liu, L.; Tu, H.; Yang, P.; Wu, D.; Chen, G.; Xiong, L.; Yang, W.; Liu, Q. Panicle-SEG: A robust image segmentation method for rice panicles in the field based on deep learning and superpixel optimization. *Plant Methods* **2017**, *13*, 104. [CrossRef]
25. Kerkech, M.; Hafiane, A.; Canals, R. Vine disease detection in UAV multispectral images using optimized image registration and deep learning segmentation approach. *Comput. Electron. Agric.* **2020**, *174*, 105446. [CrossRef]
26. Wang, X.F.; Wang, Z.; Zhang, S.W. Segmenting Crop Disease Leaf Image by Modified Fully-Convolutional Networks. *Intell. Comput. Theor. Appl.* **2019**, 646–652. [CrossRef]
27. Tusubira, J.F.; Akera, B.; Nsumba, S.; Nakatumba-Nabende, J.; Mwebaze, E. Scoring Root Necrosis in Cassava Using Semantic Segmentation. *arXiv* **2020**, arXiv:2005.03367.

28. Abdalla, A.; Cen, H.; Wan, L.; Rashid, R.; Weng, H.; Zhou, W.; He, Y. Fine-tuning convolutional neural network with transfer learning for semantic segmentation of ground-level oilseed rape images in a field with high weed pressure. *Comput. Electron. Agric.* **2019**, *167*, 105091. [CrossRef]

29. Kaur, S.; Pandey, S.; Goel, S. Plants disease identification and classification through leaf images: A survey. *Arch. Comput. Method Eng.* **2019**, *26*, 507–530. [CrossRef]

30. Saleem, M.H.; Potgieter, J.; Arif, K. Plant disease detection and classification by deep learning. *Plants* **2019**, *8*, 468. [CrossRef]

31. Jiang, P.; Chen, Y.; Liu, B.; He, D.; Liang, C. Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks. *IEEE Access* **2019**, *7*, 59069–59080. [CrossRef]

32. Wu, H.; Wiesner-Hanks, T.; Stewart, E.L.; DeChant, C.; Kaczmar, N.; Gore, M.A.; Nelson, R.J.; Lipson, H. Autonomous detection of plant disease symptoms directly from aerial imagery. *Plant Phenome J.* **2019**, *2*, 1–9. [CrossRef]

33. Li, D.; Wang, R.; Xie, C.; Liu, L.; Zhang, J.; Li, R.; Wang, F.; Zhou, M.; Liu, W. A recognition method for rice plant diseases and pests video detection based on deep convolutional neural network. *Sensors* **2020**, *20*, 578. [CrossRef]

34. Wang, G.; Sun, Y.; Wang, J. Automatic image-based plant disease severity estimation using deep learning. *Comput. Intell. Neurosci.* **2017**, *2017*. [CrossRef]

35. Verma, S.; Chug, A.; Singh, A.P. Application of convolutional neural networks for evaluation of disease severity in tomato plant. *J. Discret. Math. Sci. Cryptogr.* **2020**, *23*, 273–282. [CrossRef]

36. Lin, K.; Gong, L.; Huang, Y.; Liu, C.; Pan, J. Deep learning-based segmentation and quantification of cucumber powdery mildew using convolutional neural network. *Front. Plant Sci.* **2019**, *10*, 155. [CrossRef] [PubMed]

37. Shrivastava, V.K.; Pradhan, M.K.; Minz, S.; Thakur, M.P. Rice plant disease classification using transfer learning of deep convolutional neural network. In Proceedings of the International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences, ISPRS-GEOGLAM-ISRS Joint Int. Workshop on "Earth Observations for Agricultural Monitoring", New Delhi, India, 18–20 February 2019.

38. Chen, J.; Zhang, D.; Suzauddola, M.; Nanehkaran, Y.A.; Sun, Y. Identification of plant disease images via a squeeze-and-excitation MobileNet model and twice transfer learning. *IET Image Process.* **2021**, *15*, 1115–1127. [CrossRef]

39. Das, A.; Mallick, C.; Dutta, S. Deep Learning-Based Automated Feature Engineering for Rice Leaf Disease Prediction. In Proceedings of the Computational Intelligence in Pattern Recognition, Singapore, 18–19 November 2020; pp. 133–141.

40. Sethy, P.K.; Barpanda, N.K.; Rath, A.K.; Behera, S.K. Rice false smut detection based on Faster R-CNN. *Indones. J. Elect. Eng. Comput. Sci* **2020**, *19*. [CrossRef]

41. Elhassouny, A.; Smarandache, F. Smart mobile application to recognize tomato leaf diseases using Convolutional Neural Networks. In Proceedings of the 2019 International Conference of Computer Science and Renewable Energies (ICCSRE), Agadir, Morocco, 22–24 July 2019; IEEE Computer Society: Washington, DC, USA, 2019; pp. 1–4.

42. Russell, B.C.; Torralba, A.; Murphy, K.P. LabelMe: A database and web-based tool for image annotation. *Int. J. Comput. Vis.* **2008**, *77*, 157–173. [CrossRef]

43. Bradski, G.; Kaehler, A. *Learning OpenCV: Computer Vision with the OpenCV Library*; O'Reilly Media, Inc.: California, CA, USA, 2008.

44. Clark, A. *Pillow (PIL Fork) Documentation*, 2015; Release.

45. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Cham, Switzedland, 5–9 October 2015; pp. 234–241.

46. Chen, L.-C.; Papandreou, G.; Schroff, F.; Adam, H. Rethinking atrous convolution for semantic image segmentation. *arXiv* **2017**, arXiv:1706.05587.

47. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR), Salt Lake City, UT, USA, 18–22 June 2018; IEEE Computer Society: Washington, DC, USA, 2018; pp. 7132–7141.