

Article

Construction of Apple Leaf Diseases Identification Networks Based on Xception Fused by SE Module

Xiaofei Chao ^{1,2}, Xiao Hu ², Jingze Feng ², Zhao Zhang ², Meili Wang ² and Dongjian He ^{1,3,4,*}

- ¹ College of Mechanical and Electronic Engineering, Northwest A&F University, Yangling 712100, Shanxi, China; cxf@nwafu.edu.cn
- ² College of Information Engineering, Northwest A&F University, Yangling 712100, Shanxi, China; huxiao35@mail2.sysu.edu.cn (X.H.); fjz3334@nwafu.edu.cn (J.F.); zz02.02@nwafu.edu.cn (Z.Z.); wml@nwsuaf.edu.cn (M.W.)
- ³ Key Laboratory of Agricultural Internet of Things, Ministry of Agriculture and Rural Affairs, Yangling 712100, Shanxi, China
- ⁴ Shaanxi Key Laboratory of Agricultural Information Perception and Intelligent Service, Yangling 712100, Shanxi, China
- * Correspondence: hdj168@nwsuaf.edu.cn

Abstract: The fast and accurate identification of apple leaf diseases is beneficial for disease control and management of apple orchards. An improved network for apple leaf disease classification and a lightweight model for mobile terminal usage was designed in this paper. First, we proposed SE-DEEP block to fuse the Squeeze-and-Excitation (SE) module with the Xception network to get the SE_Xception network, where the SE module is inserted between the depth-wise convolution and point-wise convolution of the depth-wise separable convolution layer. Therefore, the feature channels from the lower layers could be directly weighted, which made the model more sensitive to the principal features of the classification task. Second, we designed a lightweight network, named SE_miniXception, by reducing the depth and width of SE_Xception. Experimental results show that the average classification accuracy of SE_Xception is 99.40%, which is 1.99% higher than Xception. The average classification accuracy of SE_miniXception is 97.01%, which is 1.60% and 1.22% higher than MobileNetV1 and ShuffleNet, respectively, while its number of parameters is less than those of MobileNet and ShuffleNet. The minimized network decreases the memory usage and FLOPs, and accelerates the recognition speed from 15 to 7 milliseconds per image. Our proposed SE-DEEP block provides a choice for improving network accuracy and our network compression scheme provides ideas to lightweight existing networks.

Keywords: apple leaf disease; disease identification network; Xception; channel attention; lightweight network



Citation: Chao, X.; Hu, X.; Feng, J.; Zhang, Z.; Wang, M.; He, D. Construction of Apple Leaf Diseases Identification Networks Based on Xception Fused by SE Module. *Appl. Sci.* **2021**, *11*, 4614. <https://doi.org/10.3390/app11104614>

Received: 9 April 2021
Accepted: 10 May 2021
Published: 18 May 2021

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



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Automatic identification of apple leaf diseases is extremely helpful to improve the management of apple orchard and provides a good indication for the early warning and prevention of apple diseases. Traditional expert diagnosis on apple leaf disease takes high cost and has subjective misjudgment risk. Therefore, with the development of computer vision technology, scholars start to use image processing technology to carry out researches on plant disease identification based on image feature extraction and modelling. However, image feature extraction used in image processing mainly relies on manual selection and design [1–4], which is time-consuming and laborious and the accuracy of image recognition needs to be improved. In addition, models for such methods are designed and trained with a specific dataset so it is hard to transfer to new tasks. In recent years, deep learning technology has developed rapidly. Especially, Convolutional Neural Networks (CNN) has been widely applied to plant leaf disease recognition [5–20]. Its end-to-end property

exempts it from the need for complex image preprocessing and manual feature selection, thus it is more efficient than traditional methods and has better generalization ability.

Classic CNN models have been trained and fine-tuned on large dataset, so some studies used transfer learning for plant disease identification. Some of them applied minor modification on the classical CNN network structure. For example, Sun et al. [21] improved the AlexNet and collected an experimental data set composed by PlantVillage dataset [22] and some images collected from the Internet. Their dataset was divided into a training set and a test set after data augmentation. The average classification accuracy on the test set is 99.56%. Wang et al. [23] trained an AlexNet model for tomato disease classification based on transfer learning. The model could quickly and accurately classify 10 categories of tomato leaf diseases, with an average classification accuracy of 95.62% on the validation set. In 2018, Long et al. [13] trained AlexNet and GoogleNet networks by using transfer learning technology for *Camellia oleifera* diseases identification. On the dataset with four kinds of diseases and healthy leaves, the recognition rate was 96.53%, and their results indicated that transfer learning could accelerate the network convergence and improve the classification performance. Ferentinos [11] compared the existing CNN to conduct experiments on the extended version of PlantVillage dataset by using transfer learning. Experiment results proved that VGG had the best classification and recognition accuracy, up to 99.53%. In addition, the experimental results showed that the recognition rate of model trained with images of laboratory background was 32.42% lower than those trained by images with natural growing background on the test set with both natural growing background and laboratory background images. There are also some studies that improve the CNN models for plant leaf disease identification. For example, Liu et al. [12] proposed a CNN model based on AlexNet to identify four common apple leaf diseases and healthy leaves in 2018, and the model could achieve an average recognition accuracy of 97.62%. In 2019, Baranwal et al. [14] designed a convolutional neural network structure based on LeNet-5 architecture, which was used to identify three kinds of apple leaf diseases and healthy leaves. The recognition rate reached 98.54%. In 2019, Jiang et al. [18] proposed a network for real-time detection of apple leaf diseases, and the disease recognition rate of the proposed VGG-INCEP model reached 97.14%. In 2020, Chao et al. [24] proposed a CNN model by fusing dense blocks of Densenet into Xception, the model could identify five apple leaf diseases and healthy leaves, and the average recognition rate was 98.35%. Their experimental results showed that compared with using laboratory background as the training set, using images with complex background as the training set could greatly improve the disease identification accuracy of the model (about 14%).

CNN networks have a global receptive field when extracting features, giving more attention to the principle features of a given task could improve the model performance. This is one of the major studies in the field of computer vision research [25]. The study shows that the spatial attention of the models can be improved by integrating the learning mechanism into CNNs, so as to improve the performance of the CNNs [26–30]. Hu et al. proposed the SE module [25] to add channel attentions to CNNs and it has been widely used. The SE module has a simple structure and only a few parameters. It can be used as an additional module of any CNN network to introduce channel attention to the network, thus improving the network feature extraction ability. There are a few works that use an SE module to add channel attentions to CNN models to improve network feature extraction ability. In 2020, Li et al. [31] proposed the SE-Inception model by integrating the SE (Squeeze and Excitation) module with the Inception to classify Solanaceae diseases with an average recognition accuracy of 98.29%. Tang et al. [32] integrated the SE module [25] into ShuffleNet [33] and carried out disease identification on grape disease from the PlantVillage dataset, with an average recognition rate of 99.14%.

The above studies have proved that CNNs are effective in solving the plant disease identification task, and the integration of the SE module into the CNN networks can improve the disease identification ability of the networks. The SE module can be added to any network to add channel attention to the network. More possible ways to add

an SE module to existing CNN networks can be explored beyond the four approaches suggested by the authors of the SE module. With the popularity of mobile devices, CNN networks can be embedded into mobile applications to provide growers with the ability of disease identification anytime and anywhere. The mobile application requires the network to have a small parameter number, low memory usage, low computation, and faster recognition speed. Existing CNN networks with large quantity of parameters are hard to deploy into the mobile devices. Therefore, we need to look for (or design) lightweight networks with high recognition accuracy. On the one hand, we can verify the performance of existing lightweight networks on the apple disease identification task. On the other hand, the network compression experiment can also be carried out on the existing CNN network with a good recognition rate in the apple disease recognition task.

The main contributions of this paper are summarized as follows:

- In order to meet the application requirements of apple leaf disease identification in a natural growing scenario, we built a dataset with 2975 images of five apple leaf diseases and healthy leaves, where images with different backgrounds were shot in different time periods.
- We propose a new method called SE-DEEP to deeply fuse the SE module with a depth-wise separable convolution layer, where the SE module is inserted between the depth-wise convolution and the pointwise convolution of the depth-wise separable convolution block, therefore, the feature channels from the lower layers can be directly weighted. This makes the model more sensitive to the principal features that are useful for the classification task. Then, we used SE-DEEP to integrate the SE module into Xception network to propose the SE_Xception network.
- To investigate the influence of depth and width on network performance, we designed an experiment to compress the proposed SE_Xception network to get the lightweight network, named SE_miniXception. In the compression experiment we compared two networks by either compressing the depth or compressing the width. The experimental results show that compression on the width has little effect on the network performance, while compression on the depth has a greater effect on the network performance. Then we compressed the SE network width to 0.25-times the original width, and further compressed its depth to get SE_miniXception. The results show that the network compression scheme designed in this paper could achieve high compression ratio, while the compressed network still has ideal performance.

The following sections of this paper are organized as follows. Section 2 introduces our constructed apple disease dataset. In Section 3, we present the proposed SE_Xception network and the light weight SE_miniXception network compressed from SE_Xception network. Section 4 compares the average identification accuracies of the proposed SE_Xception and SE_miniXception and other six popular CNNs. In Section 5, the network compression experiment and its results are discussed. The paper is finally summarized in Section 6.

2. Dataset

2.1. Building the Dataset

The experimental apple leaf images were collected from the apple experimental and demonstration stations of Northwest A&F University in China, which were located in Baishui County of Weinan city, Qianyang County of Baoji city, Luochuan County of Yan'an city, Qian County of Xianyang, and Qingcheng County of Qingyang city. Images were either taken by an ABM-500GE/BB-500GE color digital camera or an Honor V10 mobile phone. The dataset contained color images of 5 apple leaf diseases and healthy leaves, including *Alternaria* leaf spot, gray spot, brown spot, Mosaic, and rust, which were common diseases that have great influence on apple trees. The image resolution was 2448×3264 .

A total of 2975 images containing 5 kinds of diseased and healthy leaves were selected from the collected images as the experimental dataset, which was named ATLDD (Apple Tree Leaf Disease Dataset). The ATLDD contains 279 images with *Alternaria* leaf spot, 540 images with brown spot, 879 images with Mosaic, 430 images with gray spot,

429 images with rust and 418 images of healthy leaves. Examples of the dataset are shown in Figure 1. We can see that the dataset includes the disease images in different disease stages and the images are photographed under different lighting conditions and weather conditions. There is a large discrepancy between different images within each category, and there is a certain similarity between different categories to increase the robustness of our model.

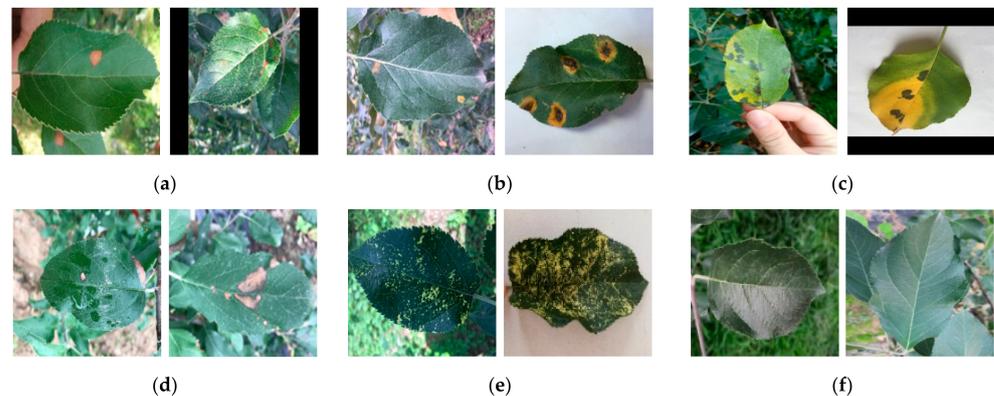


Figure 1. The examples of ATLDD dataset: (a) Alternaria leaf spot; (b) Rust disease; (c) Brown spot disease; (d) Gray spot disease; (e) Mosaic disease; (f) Healthy leaves.

2.2. Dataset Partition and Data Augmentation

(1) Dataset Partition

The dataset was divided into 3 subsets with the ratio of 19:1:4 for training, validation, and testing, respectively. We repeated the above data set division method 5 times, and trained each network 5 times, respectively. For each network, the average accuracy and average F_1 score of these 5 experiments were taken as indicators to measure the network performance.

(2) Data Augmentation

In order to increase the diversity of the dataset, alleviate over-fitting in the training stage, reduce the dependence of the model on certain attributes, and enhance the anti-interference ability of the model under complex conditions, four data enhancement methods were adopted for the images of the training set [12,34]: (a) Introduce angle interference: 90° , 180° , and 270° rotation and horizontal and vertical mirror transformation were used; (b) Introduce light interference: adjust the brightness value of the image (set the enhancement factor as 1.5 and 0.8 respectively, where the enhancement factor of 1 is the original image, and 0 for the complete black image) and the contrast value (set the enhancement factor as 3 and 0.8, respectively, where the enhancement factor of 1 is the original image); (c) Introduce noise interference and add Gaussian noise (the expectation of Gaussian distribution is 0 and the variance is 0.01); (d) Compress and enhance images through Principal Component Analysis. The example pairs of the original image and the enhanced image are shown in Figure 2. After data augmentation, one image can be extended to 14 images.

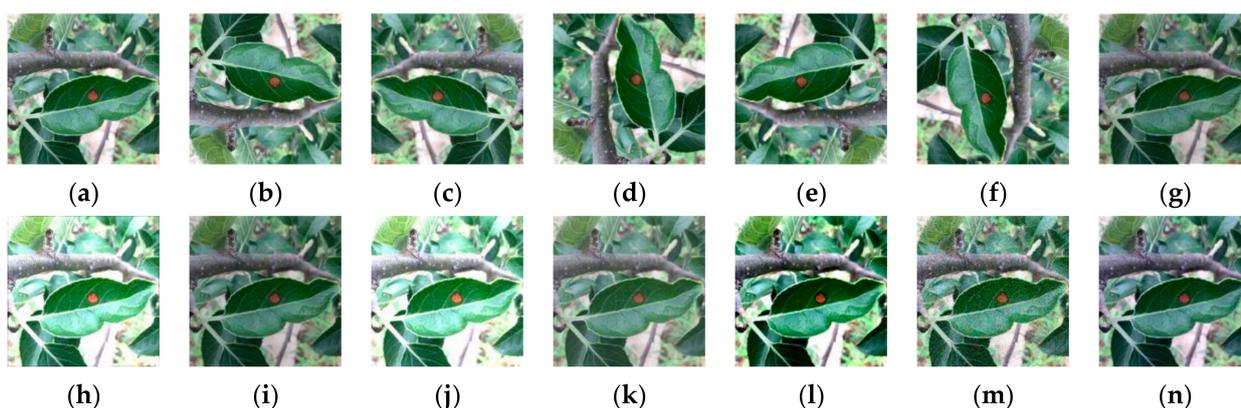


Figure 2. The data augmentation examples of training set: (a) Original image; (b) Vertical mirroring; (c) Horizontal mirroring; (d) Rotating 90°; (e) Rotating 180°; (f) Rotating 270°; (g) Weak sharpening; (h) Strong sharpening; (i) Weak light; (j) Strong light; (k) Weak contrast; (l) Strong contrast; (m) Gaussian noise; (n) PCA jitter.

3. Disease Classification Models of Xception Fused by the SE Module

3.1. Deep Fusion of the SE Module and Depth-Wise Separable Convolution

Xception [35] uses depth-wise separable convolution instead of traditional convolution operation. This helps maintain the classification accuracy while reducing the number of parameters and the amount of computation. The depth-wise separable convolutions used in Xception performs spatial convolution independently in each channel, followed by point-wise convolution. This facilitates the possibility of improving the model from the channel dimension.

Using channel dependency (namely channel correlation) is an important way to improve CNNs [25]. SE module adopts the Squeeze-Excitation-Reweight method. First, the information of all channels is summarized and compressed, then the weight of channels is calculated through the two fully connected layers and activation function. Finally, the weight is multiplied with the original feature by channel-wise, so as to complete the adaptive feature recalibration, namely channel weighting. Hu et al. [25] analyzed the computational complexity of SE module in their paper, and the experiment proved that the number of parameter of SE-ResNet-50 increased by about 10% compared with ResNet-50, while the calculation cost would increase by less than 1%, theoretically. Therefore, it is expected that the fusion of SE module into Xception will not bring obvious changes in the number of parameters and calculation time.

There are two problems to consider when introducing Squeeze-and-Excitation into Xception. Which convolution layers of the network to insert SE modules? How to fuse the SE module into Xception? The author of SENet found that the addition of the SE module into deep layers of CNNs had limited improvement of accuracy on the classification tasks. Therefore, to improve the classification performance without introducing too many extra parameters, we would not add SE modules into the deep layers, but add it to the shallow and middle layers of Xception. The author of SENet proposed four schemes to insert SE modules as independent modules into the CNN models, as illustrated in Figure 3a–d, but these four schemes did not directly weight the feature channels from the low level. Based on depth-wise separable convolution, this paper proposes to insert the SE module between the depth-wise convolution and the point-wise convolution of the depth-wise separable convolution, as shown in Figure 3e. Therefore, the SE module can perform feature recalibration directly on the feature channel from the lower level, which makes the model more sensitive to the important features for the current classification task.

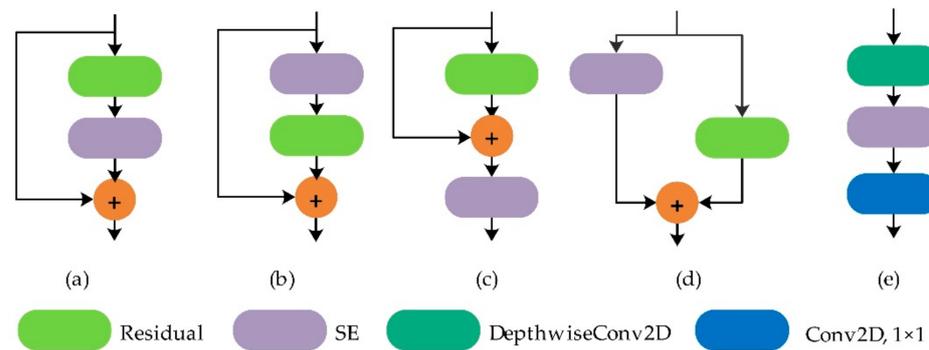


Figure 3. SE-DEEP block: SE model deeply fused to depth-wise separable convolution block: (a) Standard SE block; (b) SE-PRE block; (c) SE-POST block; (d) SE-Identity block; (e) SE-DEEP block (proposed block).

3.2. SE_Xception Network Structure

Conventional convolution considers all spatial information and channel information together. The depth-wise separable convolution used in Xception separates spatial convolution by channel and each channel with the same priority, so that the relationship between different channels is not utilized effectively. The SE module introduces attentions to different channels. Weighting on channels from the previous layer has the potential to improve depth-wise separable convolution. This study introduces the Squeeze-and-Excitation module into Xception to achieve channel weighting. Instead of simply adding SE modules to the network as separate modules, this paper proposes a network that deeply integrates the Xception network and the SE module, namely SE_Xception. In SE_Xception, depth-wise separable convolutions in Xception were reconstructed into depth-wise spatial convolution and point-wise convolution, then the SE module was inserted between depth-wise convolution and point-wise convolution. The feature channels from the low level were directly weighted and then the point-wise convolution was carried out in order to fully model and utilize the dependency relationship between feature channels. The fusion method, namely SE-DEEP block, is shown in Figure 3e. The author of SENet proposed four blocks to fuse the SE module into CNNs, and their experiments proved that standard SE block, SE-PRE module, and SE-Identity block had basically the same degree of improvement on the performance, while those of the SE-POST module were not as good as the aforementioned three schemes.

To verify the effectiveness of the SE-DEEP block proposed in this paper, SE_Xception2 network was designed by referring to the network used in Ni's paper [36] based on the fusion of standard SE block with Xception for animal species recognition. In SE_Xception, the SE-DEEP block is inserted into the shallow and middle layers of the Xception network, as shown in Figure 4, where SE_SeparableConv in the figure is using the SE-DEEP block.

CUDA V8.0.61, Anaconda V4.6.11, Python V3.6.10, Keras V2.2.4, and Keras used TensorFlow V1.4.1 as back-end tools.

4.2. Experimental Setup

Transfer learning and cross validation were used to train the networks. Considering that the pretrained model on similar datasets is helpful to improve the generalization performance and speed up the convergence of the model, all models in this paper are pre-trained on the sub-dataset of PlantVillage with five categories of leaves, and then all the models are migrated to the ATLDD dataset. The input image size is 224×224 . During the pre-training step, batch size is set to 16, epochs to 50 and initial learning rate to 0.001, and an Adam optimizer is used. According to the characteristics of ATLDD data set, the 5-fold validation method described in Section 2.2 is used to divide the data set and train the models. For all of the trainings on ATLDD dataset, batch size, number of epochs, initial learning rate, and the optimizer are the same as the pre-training step, and the validation loss is monitored during the training. If its value does not decline within 2 epochs, the learning rate will be reduced to 1/2 of what it was.

4.3. The Evaluation Index

To evaluate the robustness of the proposed SE_Xception and SE_miniXception networks, they are compared with other classical CNNs on ATLDD dataset. Five experiments were carried out. In the k -th experiment ($k = 1, 2, 3, 4, 5$), the precision ($\text{precision}_{(i)}^k$), the recall ($\text{recall}_{(i)}^k$) and the weighted average of recall and precision ($F_{1(i)}^k$) of i -th class are calculated by Equation (1)~(3). Then F_1^k will be the F_1 score of the k -th experiment of the model, which is calculated by Equation (4). Finally, F_1 is the mean of F_1^k ($k = 1, 2, 3, 4, 5$) of a network calculated by Equation (5) [21]. F_1 is taken as the robustness evaluation standard for all the networks.

$$\text{precision}_{(i)}^k = \frac{n_{ii}^k}{\sum_{j=1}^{n_{cl}} n_{ji}^k} \quad (1)$$

$$\text{recall}_{(i)}^k = \frac{n_{ii}^k}{\sum_{j=1}^{n_{cl}} n_{ij}^k} \quad (2)$$

$$F_{1(i)}^k = 2 \times \frac{\text{precision}_{(i)}^k \times \text{recall}_{(i)}^k}{\text{precision}_{(i)}^k + \text{recall}_{(i)}^k} \quad (3)$$

$$F_1^k = \frac{\sum_{i=1}^{n_{cl}} F_{1(i)}^k}{n_{cl}} \quad (4)$$

$$F_1 = \frac{\sum_{k=1}^K F_1^k}{K} \quad (5)$$

In these equations, n_{ij} refers that the class i is predicted to be the j -th class; n_{cl} is the total number of classes in the data set.

5. Experimental Results

5.1. Performance Analysis

We used the same transfer learning and hyper-parameter finetune method to evaluate the performance of the compared CNN models, including ResNet50 [37], DenseNet201 [38], Xception, MobileNetV1 [39], ShuffleNet [33], SE_Xception2, as well as the proposed SE_Xception and SE_miniXception. Table 1 lists the experimental results of compared models on ATLDD dataset with six types of apple leaves. As illustrated in Table 1, SE_Xception, SE_Xception2, Xception, and SE_miniXception rank among the top four in the aspect of disease classification accuracy. Among them, SE_Xception has the highest classification accuracy (99.40%) and F_1 score (99.10%) with a relatively fast classification speed. The number of parameters of SE_Xception is 14% higher than Xception, but its classification accuracy is

1.99% higher and F_1 score is 2.32% higher than Xception. For SE_Xception2, although there are some improvements compared to Xception, its average recognition accuracy and F_1 score is not as high as those of SE_Xception. Its accuracy is 0.61% higher than Xception and 1.38% lower than SE_Xception, and the F_1 score is 1.58% lower than those of SE_Xception.

Table 1. Experimental results of compared models for ATLDD dataset.

Model Name	Number of Parameters (M)	FLOPs (M)	Training Time for 50 Epochs (min)	Recognition Speed (ms/Image)	Accuracy (%)	F_1 Score (%)
ResNet50	23.60	47.63	72	15	95.56	94.99
DenseNet201	18.33	38.73	209	36	96.88	96.34
Xception	20.87	42.24	92	10	97.41	96.78
SE_Xception	23.85	48.15	122	15	99.40	99.10
SE_Xception2	21.20	42.88	103	13	98.02	97.52
MobileNetV1	3.23	42.24	41	6	95.41	94.64
ShuffleNet	1.95	6.67	106	20	95.79	95.49
SE_miniXception	0.61	0.13	23	7	97.01	96.29

The accuracy and F_1 score of the proposed SE_miniXception lightweight model are 97.01% and 96.29%, respectively. Its accuracy is still higher than ResNet50, DenseNet201, MobileNetV1, and ShuffleNet, but its number of parameters is only 0.61M, which is only about 3% of those of ResNet50, DenseNet201, and 18.89% of those of MobileNetV1, 31.41% of those of ShuffleNet. The number of FLOPs of SE_miniXception model is 0.13M, which is only 0.26% of SE_Xception and is the lowest in the table. The classification speed of SE_miniXception is 7 ms/image, which is only a little slower than MobileNet and faster than the rest of the compared models in Table 1.

Results in Table 1 shows that deep fusion of SE modules into Xception network with channel weighting method significantly improves the accuracy and robustness of model classification. Moreover, when compressing the SE_Xception network, the classification performance of SE_miniXception stays at a good level, and its memory usage and number of FLOPs drops greatly compared to SE_Xception. These imply the high robustness for the SE_Xception network with the proposed SE-DEEP blocks.

5.2. Comparison of Model Convergence Performance

Select the best model in the cross-validation experiment for each network and plot its accuracy curve on the validation set, as shown in Figure 6. SE_Xception has the fastest convergence speed and highest accuracy rate among all of the models on the validation set. SE_miniXception has similar convergence performance with MobileNetV1 and SE_Xception2. In addition, the learning curve of SE_Minixception fluctuates slightly more than that of SE_Xception, and the convergence speed is slower than the latter, but it converges around 20 epochs. To conclude, the convergence speed of SE_Xception is better than all the other compared models, and the compressed SE_miniXception model still has an ideal convergence.

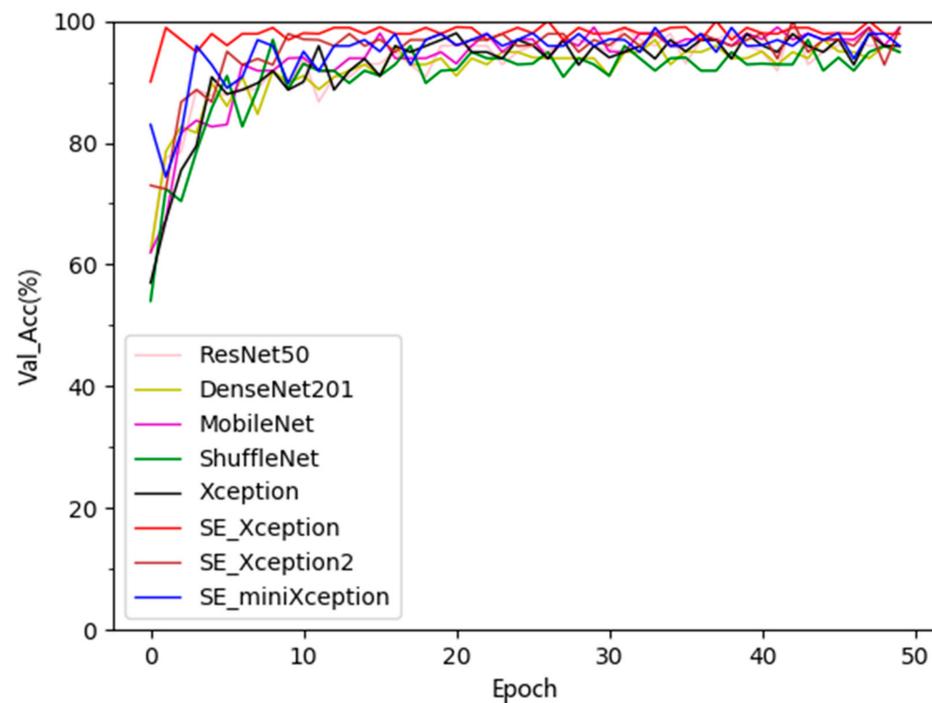


Figure 6. Comparison of convergence of SE_Xception and compared CNN models on the validation set.

5.3. Lightweight Model Analysis

The number of model parameters is an important indicator of whether the model can be deployed to the mobile platform [39]. In order to reduce memory consumption and to improve the model recognition speed to meet the requirements for mobile applications, it is necessary to minimize the amount of model parameters and FLOPs (floating point operations). We designed a compression experiment using the same dataset and experimental methods, inspired by the network compression method of MobileNets [39], we compressed SE_Xception by reducing the number of depth-wise separable convolution blocks, and also reducing the width of the convolution layers in the model by multiply α to the width of the input channels [39]. We designed several lightweight models, and compared their average recognition accuracy on ATLDD, number of parameters, and FLOPs.

Firstly, in order to answer the question of whether to compress the width or the depth, we designed two networks have similar number of parameters. We removed the last six depth-wise separable convolutional blocks from SE_Xception to get the first network, we called this compressed network the SE_Xception_shallow. Then we designed the second network, named SE_Xception_0.75, by reducing the width of SE_Xception to 0.75-times those before. Table 2 compares the accuracy, number of parameters, and number of FLOPs of these two compressed networks and the baseline SE_Xception. We can see from Table 2 that these two compressed models have a similar number of parameters and FLOPs, but the accuracy of SE_Xception_0.75 only drops a little bit compared to SE_Xception, and is 1.47% higher than SE_Xception_shallow. Therefore, we have a conclusion that making SE_Xception thinner is better than making it shallower.

Table 2. Comparison of shallower and narrower SE_Xception.

Model Name	Number of Parameters (M)	FLOPs (M)	Accuracy (%)
SE_Xception	23.85	48.15	99.40
SE_Xception_shallow	11.74	23.73	97.79
SE_Xception_0.75	13.47	27.29	99.26

Secondly, since compressing width is better than compressing depth, we further compressed the width of SE_Xception with a scale of 0.25-times and 0.50-times to get SE_Xception_0.25 and SE_Xception_0.50. Then we compared these two networks with SE_Xception_0.75, baseline SE_Xception, MobileNetV1, and ShuffleNet. Table 3 shows the accuracy, number of parameters, and number of FLOPs of these models. The numbers of parameters and FLOPs drop off quickly, while the accuracy drops off smoothly until the width scale reaches 0.1. The identification accuracy of SE_Xception_0.25 is still higher than most of the compared networks in Table 1, although its numbers of parameters and FLOPs are far less than SE_Xception. Furthermore, the number of parameters of SE_Xception_0.25 is only 13.2% of that of SE_Xception_shallow, while its classification accuracy is 0.7% higher than that of SE_Xception_shallow. This implies that for the Xception structure in apple leaf disease identification task, the depth is more important than the width of the network. Furthermore, the number of parameter and FLOPs of SE_Xception_0.25 is less than those of MobileNetV1 and ShuffleNet, but its average classification accuracy is 3.08% and 2.70% higher than these two popular light weight networks.

Table 3. Comparison of compressed SE_Xception and other light-weight models.

Model Name	Number of Parameters (M)	FLOPs (M)	Accuracy (%)
SE_Xception	23.85	48.15	99.40
SE_Xception_0.75	13.47	27.29	99.26
SE_Xception_0.50	6.04	12.31	98.59
SE_Xception_0.25	1.55	3.21	98.49
SE_Xception_0.1	0.27	0.59	95.78
MobileNetV1	3.24	6.67	95.41
ShuffleNet	1.95	3.99	95.79
SE_miniXception (SE_Xception_0.25_shallow)	0.61	1.27	97.01

To further explore the possibility of compressing both depth and width of the network, we further delete six depth-wise separable convolution blocks from the latter part of SE_Xception_0.25, and name it as SE_Xception_0.25_shallow. We find that the accuracy of SE_Xception_0.25_shallow drops by 1.48% than SE_Xception_0.25, but it is still higher than these of MobileNetV1 and ShuffleNet, as illustrated in Table 3. Therefore, we finally select SE_Xception_0.25_shallow as our light weight model, and name it as SE_miniXception. With relatively high classification accuracy, SE_miniXception reduces the number of parameters of SE_Xception from 23.85M to 0.61M, so that the parameter compression ratio reached 39:1. The FLOPs of SE_miniXception is only 0.26% of SE_Xception, and much less than other compared networks in Table 1. These laid a foundation for SE_miniXception network to be embedded in the mobile platform.

Actually, one can choose to use SE_Xception_0.25 or SE_Xception_0.25_shallow as the final lightweight model according to the requirements of the platform on the model identification accuracy, the number of model parameters, and FLOPs.

5.4. Experiment on Data Augmentation and Transfer Learning

In order to verify the effects of data augmentation and transfer learning technologies, we carried out the ablation experiment.

(1) Experiment on Data Augmentation Technology

With no transfer learning, we compared accuracy convergence curves of two SE_Xception models. One of them uses data augmentation and the other model does not. The results are shown in Figure 7a. It can be seen that the data augmentation technology can effectively improve the accuracy and convergence performance of the model.

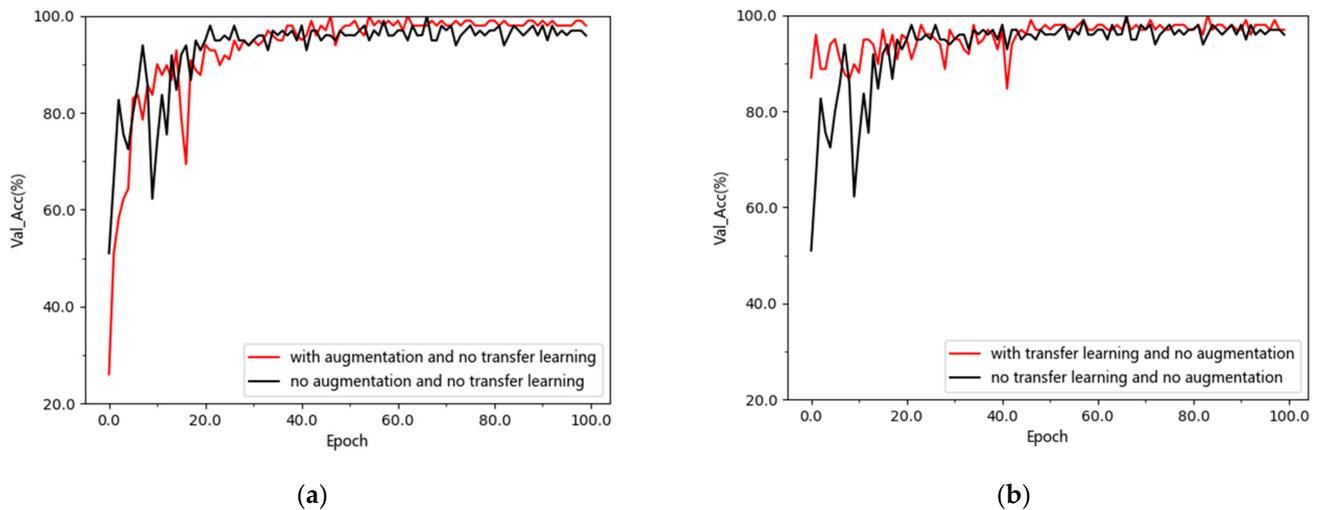


Figure 7. Effects of data augmentation and transfer learning on model convergence performance and recognition accuracy: (a) The effect of data augmentation technology on network convergence and classification accuracy without transfer learning; (b) The effect of transfer learning on network convergence and classification accuracy with no data augmentation technology.

(2) Experiment on Transfer Learning Technology

The effect of transfer learning on model performance is verified on the SE_Xception model without data augmentation. Figure 7b shows that transfer learning can improve the starting point of convergence of the model.

To summarize, data augmentation can improve the final accuracy and transfer learning can improve the starting point of convergence. The combination of these two technologies can effectively improve the final performance of the model.

6. Conclusions and Future Works

In order to add channel attentions to the CNN network, we proposed a deep fusion scheme of SE module into Xception network, and constructed the SE_Xception network, where SE module is inserted between depth-wise convolution and point-wise convolution of a depth-wise separable convolution layer, and feature channels from the lower level are weighted. The proposed SE_Xception network is better than other classical deep convolutional neural networks in terms of classification accuracy, robustness, and convergence performance on the ATLDD dataset. On the premise of maintaining the network classification performance, we propose a lightweight model SE_miniXception by compressing the width and depth of SE_Xception. Compared with MobileNet and ShuffleNet, which are two popular lightweight models, the number of parameters and FLOPs of SE_miniXception are greatly reduced, so that the model complexity and training time have minimized, and disease prediction speed has been significantly improved, while the classification accuracy of SE_miniXception on ATLDD dataset is still higher than those of MobileNet and ShuffleNet. The experimental results prove that the proposed SE-DEEP block can improve the model's performance on apple leaf disease classification through feature channel adaptive weighting. The SE_miniXception model provides technical support for the design of mobile terminal disease identification system. The scheme of SE-DEEP block and the conclusion of light-weight experiment could provide some reference value for the identification of other crop diseases.

In the future, we will extend and improve the existing data set, balance the number of images of different disease types in the dataset, and add more images with natural growing background, so as to improve the model performance on disease identification in the actual planting scenario. Furthermore, we plan to add spatial attentions and channel attentions together to the CNN networks to improve their identification accuracy.

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

Funding: This research was partially funded by Key Research and Development Program of Shaanxi Province of China (2019NY-167), and the Undergraduate Training Program for Innovation and Entrepreneurship of Northwest A&F University(X202010712368).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

Acknowledgments: Appreciations to Jifeng Ning and Jinglei Tang for their inspired discussions, to Lyuwen Huang for the sharing of apple leaf disease images, and to Hao Feng for providing apple leaf disease diagnosis standard.

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

References

1. Camargo, A.; Smith, J.S. An image-processing based algorithm to automatically identify plant disease visual symptoms. *Biosys. Eng.* **2009**, *102*, 9–21. [[CrossRef](#)]
2. Kurniawati, N.N.; Abdullah, S.N.H.S.; Abdullah, S.; Abdullah, S. Investigation on Image Processing Techniques for Diagnosing Paddy Diseases. In Proceedings of the 2009 International Conference of Soft Computing and Pattern Recognition, Malacca, Malaysia, 4–7 December 2009; pp. 272–277.
3. Khairnar, K.; Dagade, R. Disease Detection and Diagnosis on Plant using Image Processing—A Review. *Int. J. Comput. Appl.* **2014**, *108*, 36–38. [[CrossRef](#)]
4. Tian, Y.; Li, T.; Li, C.; Po, Z.; Sun, G.; Wang, B. Method for recognition of grape disease based on support vector machine. *Trans. CSAE* **2007**, *23*, 175–180. [[CrossRef](#)]
5. Kawasaki, Y.; Uga, H.; Kagiwada, S.; Iyatomi, H. Basic study of automated diagnosis of viral plant diseases using convolutional neural networks. In Proceedings of the International Symposium on Visual Computing, Las Vegas, NV, USA, 14–16 December 2015; pp. 638–645.
6. Lee, S.H.; Chan, C.S.; Wilkin, P.; Remagnino, P. Deep-Plant: Plant Identification with convolutional neural networks. In Proceedings of the International Conference on Image Processing, Quebec City, QC, Canada, 27–30 September 2015; pp. 452–456. [[CrossRef](#)]
7. Mohanty, S.P.; Hughes, D.P.; Salathe, M. Using Deep Learning for Image-Based Plant Disease Detection. *Front. Plant. Sci.* **2016**, *7*, 1419. [[CrossRef](#)] [[PubMed](#)]
8. Sladojevic, S.; Arsenovic, M.; Anderla, A.; Culibrk, D.; Stefanovic, D. Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification. *Comput. Intell. Neurosci.* **2016**, *2016*, 3289801. [[CrossRef](#)] [[PubMed](#)]
9. Amara, J.; Bouaziz, B.; Algergawy, A. A deep learning-based approach for banana leaf diseases classification. In Proceedings of the Datenbanksysteme für Business, Technologie und Web (BTW 2017)-Workshopband, Stuttgart, Germany, 6–10 March 2017; pp. 79–88.
10. Wang, G.; Sun, Y.; Wang, J. Automatic Image-Based Plant Disease Severity Estimation Using Deep Learning. *Comput. Intell. Neurosci.* **2017**, *8*. [[CrossRef](#)]
11. Ferentinos, K.P. Deep learning models for plant disease detection and diagnosis. *Comput. Electron. Agric.* **2018**, *145*, 311–318. [[CrossRef](#)]
12. Liu, B.; Zhang, Y.; He, D.; Li, Y. Identification of apple leaf diseases based on deep convolutional neural networks. *Symmetry* **2018**, *10*, 11. [[CrossRef](#)]
13. Long, M.; Ouyang, c.; Liu, H.; Fu, Q. Image recognition of Camellia oleifera diseases based on convolutional neural network & transfer learning. *Trans. CSAE* **2018**, *34*, 194–201. [[CrossRef](#)]
14. Baranwal, S.; Khandelwal, S.; Arora, A. Deep Learning Convolutional Neural Network for Apple Leaves Disease Detection. *SSRN Electr. J.* **2019**, 260–267. [[CrossRef](#)]
15. Boulent, J.; Foucher, S.; Théau, J.; St-Charles, P.-L. Convolutional Neural Networks for the Automatic Identification of Plant Diseases. *Front. Plant. Sci.* **2019**, *10*, 941. [[CrossRef](#)] [[PubMed](#)]
16. Geetharamani, G.; Pandian, J.A.; Agarwal, M.; Gupta, S.K. Identification of plant leaf diseases using a nine-layer deep convolutional neural network. *Comput. Electr. Eng.* **2019**, *76*, 323–338. [[CrossRef](#)]
17. Hasan, M.; Tanawala, B.; Patel, K. Deep Learning Precision Farming: Tomato Leaf Disease Detection by Transfer Learning. *SSRN Electr. J.* **2019**. [[CrossRef](#)]

18. Jiang, P.; Chen, Y.; Bin, L.; 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](#)]
19. Ma, J.; Du, K.; Zheng, F.; Zhang, L.; Sun, Z. Disease recognition system for greenhouse cucumbers based on deep convolutional neural network. *Trans. CSAE* **2018**, *34*, 186–192. [[CrossRef](#)]
20. Zhang, J.; Kong, F.; Wu, J.; Zhai, Z.; Han, S.; Cao, S. Cotton disease identification model based on improved VGG convolution neural network. *J. China Agric. Univ.* **2018**, *23*, 161–171. [[CrossRef](#)]
21. Sun, J.; Tan, W.; Mao, H.; Wu, X.; Chen, Y.; Wang, L. Recognition of multiple plant leaf diseases based on improved convolutional neural network. *Trans. CSAE* **2017**, *33*, 209–215. [[CrossRef](#)]
22. Hughes, D.; Salathé, M. An open access repository of images on plant health to enable the development of mobile disease diagnostics. *arXiv* **2015**, arXiv:1511.08060.
23. Wang, Y.; Zhang, H.; Liu, Q.; Zhang, Y. Image Classification of tomato leaf diseases based on transfer learning. *J. China Agric. Univ.* **2019**, *24*, 124–130. [[CrossRef](#)]
24. Chao, X.; Sun, G.; Zhao, H.; Li, M.; He, D. Identification of Apple Tree Leaf Diseases Based on Deep Learning Models. *Symmetry* **2020**, *12*, 7. [[CrossRef](#)]
25. Hu, J.; Shen, L.; Albanie, S.; Sun, G.; Wu, E. Squeeze-and-Excitation Networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
26. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
27. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the International Conference on Machine Learning, Lille, France, 6–11 July 2015; pp. 448–456.
28. Bell, S.; Zitnick, C.L.; Bala, K.; Girshick, R. Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2874–2883.
29. Newell, A.; Yang, K.; Deng, J. Stacked hourglass networks for human pose estimation. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 483–499.
30. Jaderberg, M.; Simonyan, K.; Zisserman, A.; Kavukcuoglu, K. Spatial transformer networks. *arXiv* **2015**, arXiv:1506.02025.
31. Li, Z.; Yang, Y.; Li, Y.; Guo, R.; Yang, J.; Yue, J. A solanaceae disease recognition model based on SE-Inception. *Comput. Electron. Agric.* **2020**, *178*, 105792. [[CrossRef](#)]
32. Tang, Z.; Yang, J.; Li, Z.; Qi, F. Grape disease image classification based on lightweight convolution neural networks and channelwise attention. *Comput. Electron. Agric.* **2020**, *178*, 105735. [[CrossRef](#)]
33. Zhang, X.; Zhou, X.; Lin, M.; Sun, J. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6848–6856.
34. Shorten, C.; Khoshgoftaar, T.M. A survey on Image Data Augmentation for Deep Learning. *J. Big Data* **2019**, *6*, 1–48. [[CrossRef](#)]
35. Chollet, F. Xception: Deep Learning with Depthwise Separable Convolutions. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
36. Ni, L.; Zou, W. Recognition of Animal Species Based on Improved Xception by SE Module. *Navig. Control.* **2020**, *19*, 106–111. [[CrossRef](#)]
37. 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.
38. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
39. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.