

Article

Lightweight Corn Seed Disease Identification Method Based on Improved ShuffleNetV2

Lu Lu , Wei Liu * , Wenbo Yang, Manyu Zhao and Tinghao Jiang

School of Information and Electronic Engineering, Shandong Technology and Business University, Yantai 264005, China

* Correspondence: liuweis@sdtbu.edu.cn

Abstract: Assessing the quality of agricultural products is an essential step to reduce food waste. The problems of overly complex models, difficult to deploy to mobile devices, and slow real-time detection in the application of deep learning in agricultural product quality assessment requiring solutions. This paper proposes a lightweight method based on ShuffleNetV2 to identify phenotypic diseases in corn seeds and conduct experiments on a corn seed dataset. Firstly, Cycle-Consistent Adversarial Networks are used to solve the problem of unbalanced datasets, while the Efficient Channel Attention module is added to enhance network performance. After this, a 7×7 depthwise convolution is used to increase the effective receptive field of the network. The repetitions of basic units in ShuffleNetV2 are also reduced to lighten the network structure. Finally, experimental results indicate that the number of model parameters are 0.913 M, the computational volume is 44.75 MFLOPs and 88.5 MMAdd, and the recognition accuracy is 96.28%. The inference speed of about 9.71 ms for each image was tested on a mobile portable laptop with only a single CPU, which provides a reference for mobile deployment.

Keywords: image classification; lightweight neural networks; zea mays; ECA attention; CycleGAN



check for updates

Citation: Lu, L.; Liu, W.; Yang, W.; Zhao, M.; Jiang, T. Lightweight Corn Seed Disease Identification Method Based on Improved ShuffleNetV2. *Agriculture* **2022**, *12*, 1929. <https://doi.org/10.3390/agriculture12111929>

Academic Editors: Muhammad Sultan, Redmond R. Shamshiri, Md Shamim Ahamed and Muhammad Farooq

Received: 14 October 2022

Accepted: 10 November 2022

Published: 17 November 2022

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



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Evaluating the quality agricultural products is an issue to which countries have always attached great importance. Recent years have seen the introduction of the concept of precision agriculture, with stricter requirements for the quality assessment of agricultural products, which is important to guarantee accurate identification and effective control of seed pests and diseases, in addition to playing a vital role in grain storage and distribution management, helping to ensure seed quality, avoid food waste, and ensure food security [1,2].

Fusarium graminearum, *F.cepacia*, *F.proliferatum*, and *F.subglutinans* are common causes of root, stalk, and cob rot in maize [3]. Diseased seeds are an important source of primary infestation resulting in plant diseases, leading to the long-distance spread of plant diseases and reducing the germination rate of seeds [4]. Infected seeds are not conducive to storage and can cause other seeds in the storage to become moldy, thus causing huge food losses and further leading to declines in seed quality or rendering these seeds altogether inedible [5]. Traditional grain quality and safety assessments often use microbial experiments (e.g., spore counting, enzyme-linked immunosorbent assays). In spite of their excellent detection accuracy, these methods are time-consuming, labor-intensive, and destructive [6]. As an important basis for evaluating the quality of seeds, the phenotypic detection of seeds is a non-destructive testing method. However, due to the influence of subjective factors in manual testing, the test results vary from person to person, and the detection efficiency is also low, which is easy to misjudge [7,8]. Therefore, quality inspectors urgently need a fast and objective method to detect diseases in corn seeds.

With deep learning's ability to extract features efficiently and accurately, it has become widely used in agriculture, reducing the need for manual feature extraction and analysis and

making great progress in crop disease detection. Using hyperspectral imaging technology and deep convolutional neural networks (DCNNs), Zhang et al. [9] classified corn seeds with different degrees of freezing damage and reached a higher than 90% classification accuracy. Javanmardi et al. [10] used deep convolutional neural networks to classify varieties of maize seeds with an accuracy of 98%. Wang et al. [11] used hyperspectral imaging to identify aged maize seeds, which involved a full-spectrum classification model using the support vector machine (SVM) algorithm, principal component analysis (PCA), and ANOVA to reduce the data's dimensionality and extract the feature wavelengths; they classified maize seeds harvested in different years with a prediction accuracy of 97.5%. Yang et al. [12] used hyperspectral imaging (HSI) combined with sparse auto-encoders (SAEs) and convolutional neural network (CNN) algorithms to classify the mold grades of maize kernels. SAEs and a CNN were combined with an SVM classifier to construct the SAE-CNN-SVM model, and the results showed 99.47% and 98.94% correct recognition rates on the training and test sets, respectively.

A real-time method based on deep convolutional neural networks for the identification of maize leaf diseases was proposed by Mishra et al. [13], and the model was deployed on a Raspberry Pi 3. It was used to identify maize leaf diseases, and their model achieved an accuracy of 88.46%. Meng et al. [14] developed a spectral disease indices (SDIs) monitoring model based on in situ leaf reflection spectra to detect southern corn rust (SCR)-infected leaves and to classify the severity of SCR damage. The performance of the developed SCR-SDIs was evaluated by employing a support vector machine (SVM), and the model achieved an overall accuracy of 87% and 70% for SCR detection and severity classification, respectively. The authors also found that these spectral features were associated with the leaf pigments and water content. Albarrak et al. [15] created a date fruit dataset containing eight categories and used the MobileNetV2 model for date fruit classification. The results showed that the classification accuracy was 99%. Padilla et al. [16] used convolutional neural networks and OpenMP to detect leaf blight, leaf rust, and leaf spot in corn crops, and performed validation experiments on a Raspberry Pi, with measured accuracies of 93%, 89%, and 89%, respectively.

All of the above studies have provided positive results in agricultural product quality assessment and classification. However, hyperspectral imaging data acquired has high physical complexity, and analysis of hyperspectral data requires fast computers, sensitive detectors, and large data storage capacity [17]. In addition, large convolutional neural networks or traditional machine learning models are difficult to deploy in agricultural production environments with limited computing resources.

To make deep learning models flexible for deployment on mobile platforms, scholars have proposed lightweight network structures such as GhostNet [18], MobileNet [19], ShuffleNet [20], etc. Existing domestic and international methods for mobile crop disease identification include application realization on mobile devices such as cell phones, intelligent mobile monitoring robots, aerial monitoring drones, and other ground deployment methods [21,22]. In response to the above research, this paper proposes a lightweight corn seed disease identification method with an improved ShuffleNetV2. We started by using CycleGAN [23] to solve the problem of the unbalanced corn seed disease dataset. Then, the ShuffleNetV2 and ECA [24] modules are combined to improve network performance; the network structure is simplified to speed up network inference. Finally, an experimental evaluation was conducted on a corn seeds dataset [25], and the results showed that the improved model was lighter and had better recognition accuracy compared with ShuffleNetV2, which shows the potential for crop pest recognition on mobile platforms with low computational power.

2. Dataset Preparation

The dataset in this paper is the public Corn Seeds Dataset [25] provided by the laboratory in Hyderabad, India, which classifies corn seeds into four categories, pure, broken, discolored, and silkcut, for a total of 17,801 maps. The number of healthy seeds accounts for 40.8% of the original dataset, and the number of diseased seeds that are broken, discolored,

and silkcute account for 32%, 17.4%, and 9.8% of the total, respectively. The number of corn seeds in each of the four categories is thus extremely unbalanced. Nagar S. et al. [25] used the BigGAN [26] method to generate 5000 pseudograms, but the dataset suffered from classification inaccuracies, making the recognition model network severely over-fitted. Figure 1 shows a preview of the four categories of the Corn Seeds Dataset.

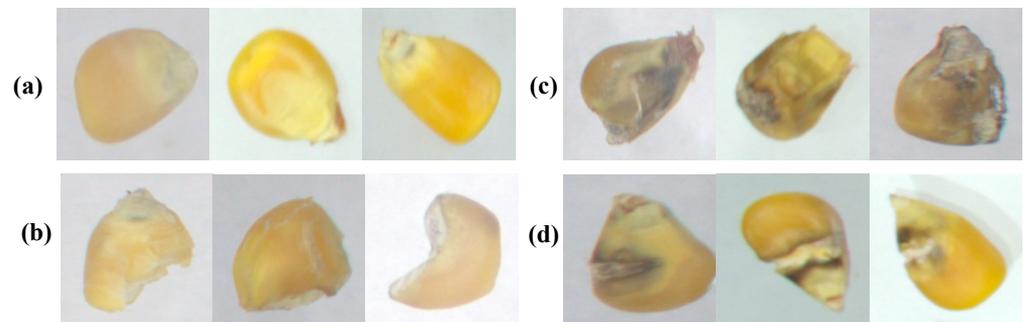


Figure 1. Corn Seeds Dataset: (a) pure, (b) broken, (c) discolored, (d) silkcute.

2.1. CycleGAN Data Augmentation

In this paper, each type of seed in the dataset is first manually corrected according to the original criteria:

- (1) Pure: the seeds are full in appearance, with no visible breakage, mold, black rot, or cracks;
- (2) Broken: the seeds are incomplete in appearance, with visible breaking, accompanied by a few mold infections and discoloration;
- (3) Discolored: large areas of mold infection and black rot on the seed surface causing discoloration of the seeds, accompanied by a small amount of breakage;
- (4) Silkcute: the basic type of seed is intact, with visible cracks on the surface, with a few accompanying discolorations or breakage at the cracks.

Next, we use the Cycle-Consistent Adversarial Networks (CycleGAN) to generate disease seeds in the dataset to solve the dataset's imbalance problem. CycleGAN is an unsupervised generative adversarial network that does not require a one-to-one mapping relationship between training data for image-to-image translation. Figure 2 illustrates the model structure of CycleGAN, where A and B represent two different styles of image domains; a and b represent the images in the A and B domains, respectively; and G and F represent the generators required for the mutual translation process of image domains A and B . The translation process from A to B can be described as follows: a obtains a forged image $G(a)$ with the style of B through generator G , and the forged image $G(a)$ is input to generator F to obtain the reconstructed image $F(G(a))$. The translation of B to A follows the same process as above. The two discriminators, $Discriminator_A$ and $Discriminator_B$, discriminate the forged image and calculate the probabilities $D_A(a)$ and $D_A(F(b))$ that a and $F(b)$ belong to the A domain and similarly obtain the probabilities $D_B(b)$ and $D_B(G(a))$ that b and $G(a)$ belong to the B domain. The computed probabilities between the discriminators D_A and D_B are used to define the adversarial loss of CycleGAN, which ensures that the generator and discriminator evolve with each other, thus allowing the generator to generate more realistic images; the A and B domains and the reconstructed image domains $F(G(a))$ and $G(F(b))$ are mapped using a cyclic consistent loss function to ensure an efficient mapping from domain A to B .

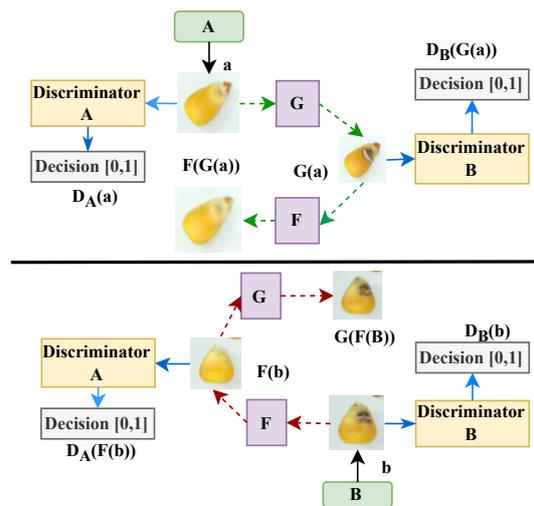


Figure 2. CycleGAN model structure.

2.2. Loss Function

The loss function of CycleGAN consists of the adversarial loss $Loss_{GAN}$ and the cycle consistent loss $Loss_{cycle}$, which ensures an efficient mapping of the two domains. The total adversarial loss is $Loss_{GAN}$ with image domains A and B . G is the mapping generator from image domain A to B with the adversarial loss function denoted as $Loss_{A2B}$; F is the mapping generator from image domain B to A with the adversarial loss function denoted as $Loss_{B2A}$; and the total adversarial loss is $Loss_{GAN}$, as shown in the following Equations (1)–(4):

$$Loss_{GAN} = Loss_{A2B} + Loss_{B2A} \quad (1)$$

$$Loss_{A2B} = \mathbb{E}_{(b \sim p_{data}(b))} [\log D_B(b)] + \mathbb{E}_{(a \sim p_{data}(a))} [\log(1 - D_B(G(a)))] \quad (2)$$

$$Loss_{B2A} = \mathbb{E}_{(a \sim p_{data}(a))} [\log D_A(a)] + \mathbb{E}_{(b \sim p_{data}(b))} [\log(1 - D_A(G(b)))] \quad (3)$$

$$Loss_{Cycle} = \mathbb{E}_{(a \sim p_{data}(a))} [\|F(G(a)) - a\|_1] + \mathbb{E}_{(b \sim p_{data}(b))} [\|G(F(b)) - b\|_1] \quad (4)$$

where $p_{data}(a)$ and $p_{data}(b)$ are the probability distributions of image domains A and B , respectively.

The cycle consistent loss function $Loss_{cycle}$ is shown in Equation (4). For each image a from domain A , the image translation cycle should be able to bring a back to the original image, i.e., $a \approx F(G(a))$. We call this forward cycle consistency. The cycle consistent loss of the B to A domain is similar: $b \approx G(F(b))$ satisfies the backward cycle consistency, thus avoiding the situation of invalid adversarial loss. From the above Equations (1)–(4), the loss function Loss of CycleGAN can be obtained as in Equation (5):

$$Loss = Loss_{GAN} + \lambda \times Loss_{Cycle} \quad (5)$$

where λ is weight of the cyclic consistency loss, controlling for the relative importance of adversarial and cyclic consistency losses.

2.3. Training Results

Table 1 below shows the distribution of pure, broken, discolored and silkcut seeds in this dataset. This paper uses CycleGAN to supplement the discolored and silkcut classes. The two training processes define image domain A as pure and image domain B as discolored or silkcut.

The CycleGAN image translation models for pure and discolored seeds are first trained. In its model training phase, for the input image domains A (pure) and B (discolored), the corresponding forged and reconstructed images are generated by the generative network; then, the gradient of the generative network is calculated and the weight parameters are updated; next, the gradient of the discriminative network is calculated and the weight

parameters are updated; finally, the model parameters are saved. The training process for the pure to silkcut translation is the same as above.

Table 1. Distribution of datasets.

Classification	Original Dataset	After Screening	Participate in Training	CycleGAN Generation	New Dataset
Pure	7265	6972	5473	0	5473
Broken	5670	5489	5489	0	5489
Discolored	3115	2748	2748	2677	5425
Silkcut	1751	1569	1569	4011	5580

The input image is processed by RandomHorizontalFlip, RandomCrop, and Normalize into the generator; the input and output image resolutions are unified at 128×128 , the number of residual blocks is 6, the Adam optimizer is used, the batch size is 1, and a total of 200 iterations are trained from the beginning. The learning rate remains constant at 0.0002 for the first 100 iterations, and decreases linearly in the direction of 0 for the next 100 iterations. The experimental framework was PaddlePaddle 2.1.2 (Baidu; Beijing, China), Python 3.7 (Centrum Wiskunde & Informatica, Netherlands), and an NVIDIA Tesla V100 graphics card for model training.

The experimental procedure is shown in Figures 3 and 4. The image translation process recordings for each 50 iterations of the two training sessions are shown separately. The first row of the figure are all real training images of image domains A and B, and the second row shows the corresponding generated images. The figure clearly shows that as the number of training epochs increases, the healthy maize seeds gradually possess the features of the diseased seeds, while the mold and cracks on the diseased seeds gradually disappear, representing that the CycleGAN network has learned the mapping relationship between the two image domains, completing the translation between pure and discolored (or pure and silkcut).

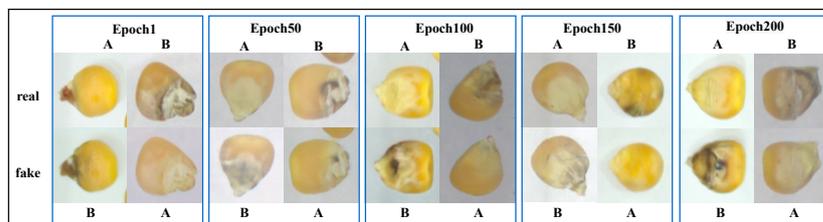


Figure 3. Pure and discolored interconversion training process.

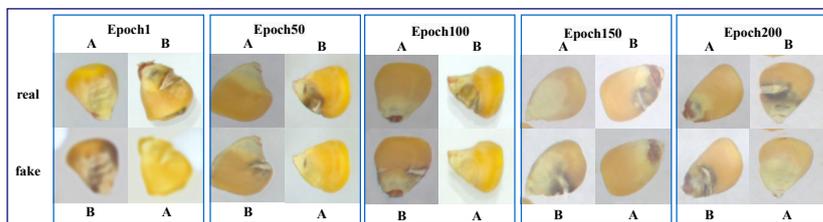


Figure 4. Pure and silkcut interconversion training process.

In the model testing phase, e.g., translating pure seeds to discolored, the latest saved model is first loaded; then, a batch of pure and discolored images are fed into CycleGAN; and after the test is completed, we will obtain a batch of fake images, such as the fake discolored seed images in Figure 5. Finally, we save the resulting fake map and use it to solve the imbalance problem in the dataset.

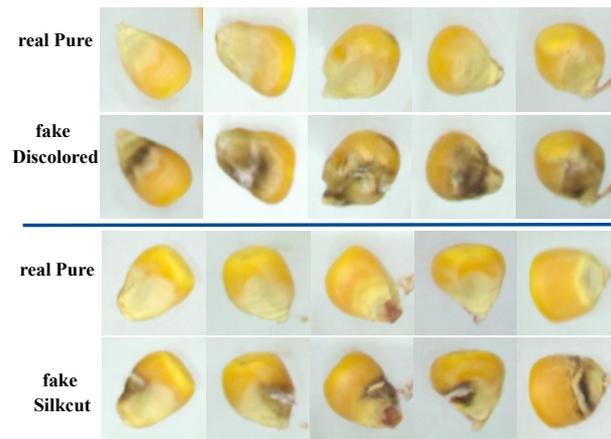


Figure 5. CycleGAN model image translation test results.

Figure 5 shows the results of testing with the trained CycleGAN model, with pure translated into discolored on top and pure translated into silkcult on the bottom. The generated images, noted as fake discolored or fake silkcult, are similar to the discolored and silkcult corn seed features in the real dataset and can be used to balance the dataset.

3. Efficient Channel Attention

Attention mechanisms have been widely used in machine vision in recent years, and attention in neural networks allows the system to pay more attention to focused information and learn better attention weights. The ECA attention mechanism proposed by Wang et al. is an improvement on the SE (Squeeze-and-Excitation) [27] module. The authors found that avoiding dimensionality reduction and a proper local cross-channel interaction strategy helped to improve the performance and efficiency of channel attention.

The ECA structure is shown in Figure 6, with feature χ passing through the Global Average Pooling (GAP) layer to obtain a $1 \times 1 \times C$ feature matrix, and the convolution operation is performed on this feature using a local weight sharing, one-dimensional convolution, followed by a sigmoid activation function to obtain the attention weights; then, the output feature matrix is derived by multiplying this weight with the input feature map. One-dimensional convolution involves an adaptive hyperparameter k (convolution kernel size), which represents the coverage of local cross-channel interactions, as shown in Equation (6):

$$k = \psi(C) = \left\lfloor \frac{\log_2(c)}{\gamma} + \frac{b}{\gamma} \right\rfloor_{\text{odd}}, (\gamma = 2, b = 1) \tag{6}$$

where $|x|_{\text{odd}}$ denotes the oddest number closest to x . ECA attention mechanisms have fewer parameters compared to SE, and appropriate cross-channel information interaction ensures that gains are brought to the network while introducing a small number of parameters.

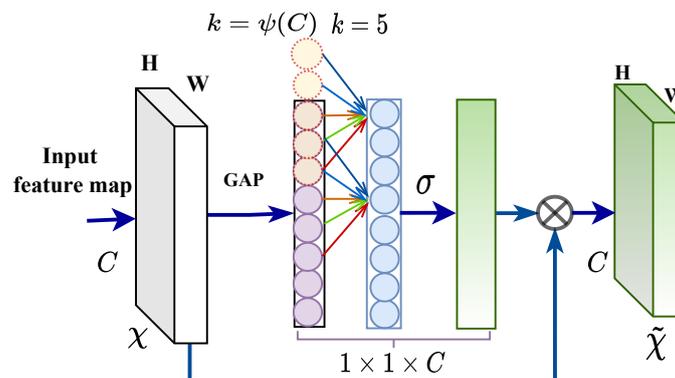


Figure 6. ECA attention mechanism.

4. ECA—ShuffleNetV2 Relevant Theories

The ultimate goal of deep learning development has always been to proceed to practical applications, which makes people more concerned about how to obtain the optimal results with limited resources. In this paper, we compare the Top-1 Accuracy of GhostNet, MobileNet V2, MobileNet V3, and ShuffleNetV2 series networks under CIFAR-10 [28]. The following metrics were counted using the torchstate tool provided by PyTorch: FLOPs, Params, and Memory cost.

To meet the needs of mobile deployments, only lightweight versions of these networks are tested in this article. The detailed model versions are shown in Table 2. We had to make a trade-off between accuracy and speed, and the table shows that MobileNet V2 1× achieved the highest Top-1 Accuracy on CIFAR-10, with ShuffleNetV2 1× following closely behind. However, the FLOPs, Params, and Memory costs of ShuffleNetV2 1× are much lower than those of MobileNet V2 1×. Without a doubt, ShuffleNet V2 1× won this race, and this was further confirmed in subsequent experiments. The subsequent experiments also focused on ShuffleNetV2 and MobileNet V2. To further refine the experiments, the rest of the networks are tested in the final inference speed comparison.

Table 2. Comparison of various parameters of the lightweight model.

Model	FLOPs (M)	Params (M)	Memory Cost (MB)	Top-1 Accuracy (%)
ShuffleNetV2 0.5×	42.63	0.352	10.63	85.7
ShuffleNetV2 1×	149.58	1.257	20.84	88.5
MobileNet V2 0.5×	101.35	0.590	40.12	87.6
MobileNet V2 1×	318.96	2.236	74.25	89.4
MobileNet V3-Small	58.8	1.528	16.20	88.0
GhostNet 0.5×	45.82	1.319	20.03	85.6
GhostNet 1×	149.41	3.914	40.05	87.7

4.1. Depthwise Separable Convolution

Depthwise Separable Convolution (DSC) has been effective in making network structures more lightweight [29]. It consisting of depthwise (DW) and pointwise (PW) convolution and has a relatively low number of parameters and computational cost in extracting features compared to ordinary convolution. One depthwise convolution is computed for only one channel of the input feature map. Pointwise convolution is similar to ordinary convolution with a size of $1 \times 1 \times M$. M is the number of channels in the input feature matrix. Thus, the number of output feature maps is equal to PW's number of convolution kernels. Moreover, PW convolution solves the problem of DW convolution, which is the poor interaction between the feature information of different channels at the same spatial location, shown in Figure 7.

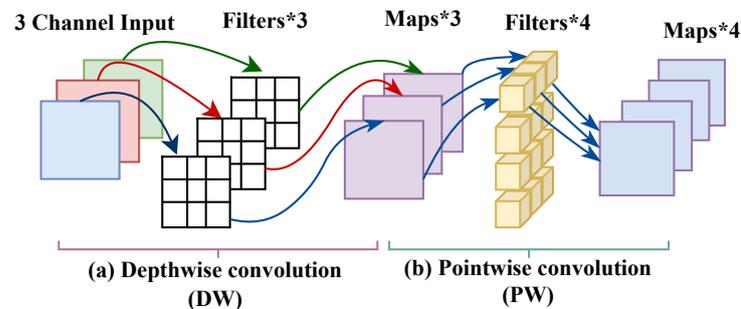


Figure 7. Depthwise separable convolution: (a) DW: Depthwise convolution; (b) PW: Pointwise convolution.

4.2. Grouped Convolution and Channel Shuffle

Grouped convolution discretizes dense convolutional connections to build a sufficiently deep and wide neural network by replicating the grouped convolution. Compared with standard convolution, grouped convolution has less parameters, lower complexity, and helps to facilitate the parallelism in the model, but the lack of information exchange between different

groups weakens the feature extraction ability of the network. ShuffleNet V1 uses the channel shuffle to remedy this deficiency [30]. As shown in Figure 8, Output Features 1 after group convolution are “shuffled” by the channel shuffle operation, which fully integrates the inter-group channel information without increasing the computational effort.

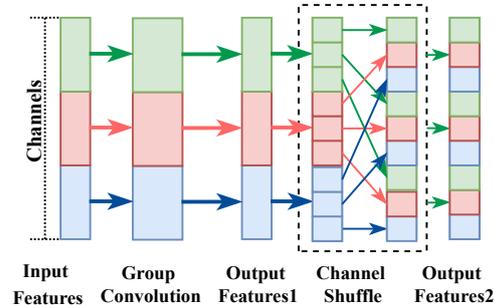


Figure 8. Grouped convolution and channel shuffle.

4.3. ShuffleNet V2

ShuffleNetV2 proposes that FLOPs are an indirect metric that cannot be equated with direct metrics such as speed and accuracy and proposes four guidelines for efficient network design:

- (1) Maintain a constant number of convolutional input and output channel widths to minimize the memory access cost;
- (2) The quantity of groups in group convolution is inversely proportional to the speed of network operation;
- (3) Cautious fragmentation operations and reducing the count of network branches can improve operational efficiency;
- (4) Reducing element-wise operations which have relatively small FLOPs but high memory access costs.

Starting from the four guidelines mentioned above, the authors present an improved ShuffleNetV2 network structure, illustrated in Figure 9. The authors devised the channel split operation shown in Figure 9a. In front of each unit, the input channel C is divided into two equal branches. To avoid fragmentation, one branch is left unchanged. The other branch follows criterion (1) and consists of three convolutional layers with a constant number of channels and no longer uses 1×1 grouped convolution. Then, the two branches are concatenated in the depth direction, followed by a channel shuffling operation to enhance the information interaction between the channels. Figure 9b shows the module with downsampling, and with channel splits removed, twice the number of channels of the output feature map as the input are obtained.

4.4. ECA—ShuffleNetV2

The basic component of a transformer is self-attention, which essentially performs Query-Key-Value operations at a global scale or within a larger window, which is the reason for the superior performance of transformers on downstream tasks [31]. Recently, many scholars have applied large convolution kernels to CNNs. Liu et al. [32] borrowed the large-scale window of the transformer in their paper and changed the size of the convolution kernels in CNNs from 3×3 to 7×7 . They experimentally concluded that 7×7 convolution kernels can achieve better detection on the ImageNet dataset with only a small increase in the number of parameters. Similarly Ding et al. [33] state in their article that large convolution kernels are both more accurate and more efficient at this task.

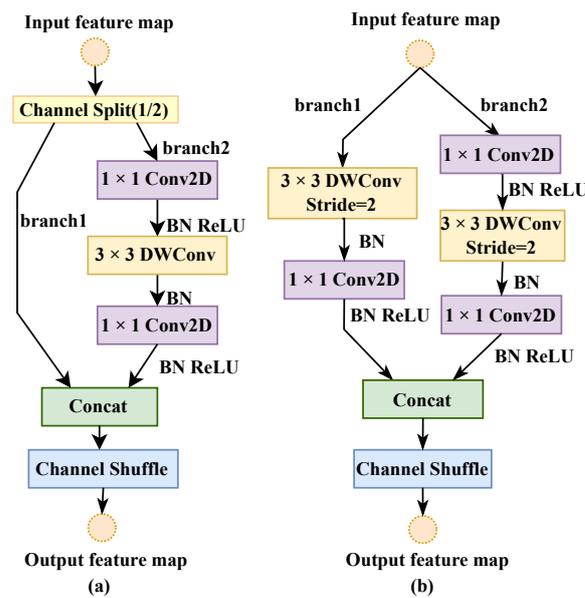


Figure 9. (a) ShuffleNetV2 basic unit, ShuffleNetV2 block 1; (b) unit for spatial down sampling, ShuffleNetV2 block 2. DWConv: depthwise convolution.

In this paper, the two units of ShuffleNetV2 are improved by moving forward the depthwise convolution in the original branch 2, followed by two pointwise convolutions. The specific structure of branch 2 is shown in Figure 10b, where a depthwise convolution of the size 7×7 is used instead of the 3×3 depthwise convolution in ShuffleNetV2, depicted in Figure 10c.

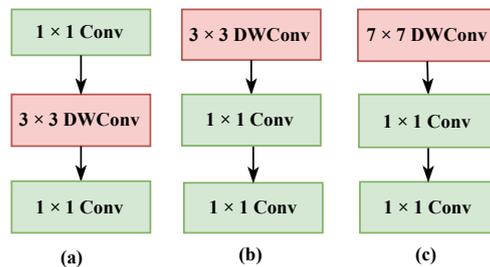


Figure 10. (a) Structure for branch 2 in the ShuffleNetV2 unit; (b) DW convolutional advance of branch 2's structure; (c) structure of branch 2 in the ECA ShuffleNetV2 base unit.

In practical applications, the model needs to be reasonably designed according to the complexity of the task. The deep small kernel network has a large theoretical receptive field, but its effective receptive field is limited. We replace the CNN network with a deep small kernel with a shallow but large kernel and conclude in this paper that the method is effective and feasible.

This is a relatively simple four-classification task, and the model size should be appropriately reduced to improve detection efficiency. Therefore, in this paper, the number of repeats of Block C is reduced. Moreover, the 3×3 DW convolution is connected after the channel shuffling of Block D with the downsampling function to further extract features. The modified unit of the network is shown in Figure 11.

Figure 12 shows a comparison of the ShuffleNetV2 $1 \times$ and the improved ECA-ShuffleNetV2 $1 \times$ network structure, with the input image resolution changed to 160×160 . Since the DW convolution is moved forward in the branch, the number of channels in the output feature layer is changed to $[58, 116, 232, 464, 1024]$ here to guarantee that the number of channels of its output feature matrix is divisible by the number of channels of the input feature matrix. The 3×3 convolutional layer and the max pooling at the beginning of the ShuffleNetV2 are replaced by a 4×4 convolutional, a stride of 4, and the input image is downsampled $4 \times$. We reduce the size of the network by changing the number of iterations of Block C in each stage from (3, 7, 3) to (1, 1, 1). Conv5 is then followed by the ECA attention

module, and the appropriate cross-channel information interaction ensures that a small quantity of parameters are introduced while bringing gains to the network.

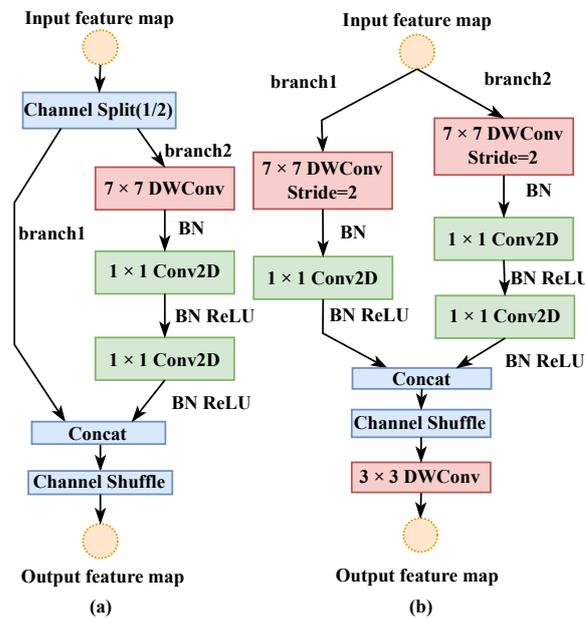


Figure 11. (a) ECA-ShuffleNetV2 basic Block C; (b) ECA-ShuffleNetV2 Block D for spatial down sampling; DWConv: depthwise convolution.

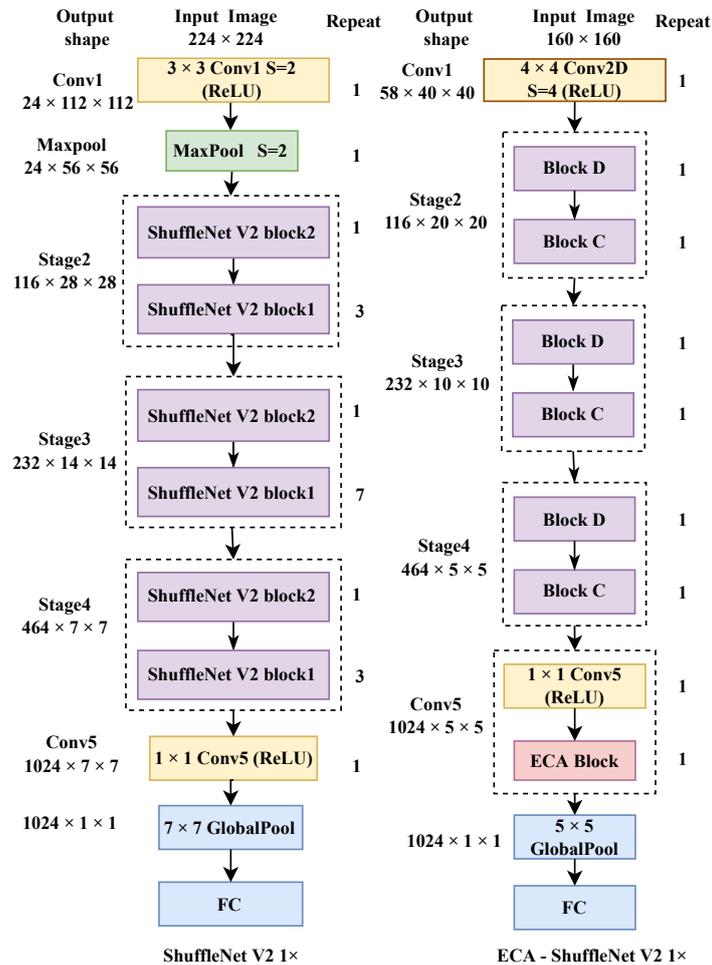


Figure 12. ShuffleNetV2 1× and ECA-ShuffleNetV2 1× network structure.

5. Model Training

5.1. Experimental Details

The dataset in this paper is from the Corn Seeds Dataset provided by the Hyderabad Laboratory in India. The imbalance in the dataset was resolved using CycleGAN, and the distribution of the datasets involved in the training is shown in Table 3, with a total of four categories and 21,967 photos. The balanced dataset was randomly divided into a training set and a validation set according to the 4:1 ratio of each category.

Table 3. Datasets.

No.	Classification	Training Set	Validation Set	Number
1	Pure	4321	1152	5473
2	Broken	4333	1156	5489
3	Discolored	4281	1144	5425
4	Silkcut	4408	1172	5580
Total		17,343	4624	21,967

The experiments were conducted in the Anaconda environment, Python 3.8.1 (Centrum Wiskunde & Informatica, Netherlands), Pytorch 1.10.0 (Facebook Artificial Intelligence Research, Menlo Park, California, United States), CUDA 11.3 (NVIDIA, Santa Clara, California, United States), and trained on an NVIDIA TITAN Xp graphics card. The SGD (stochastic gradient descent) optimizer was used, with parameters set to momentum = 0.9, weight decay = 0.012, and lr = 0.01. The learning rate decay was set to ReduceLROnPlateau with threshold = 0.99, mode = 'min', factor = 0.70, and patience = 3. The input image was resized to 160 × 160 by random horizontal flipping and normalized. The batchsize was set to 200 for a total of 70 epochs of iterative training. Using parameter grouping optimization in training [34], the trainable parameters are divided into two groups based on whether they require L2 regularization processing, where weights in the convolution and fully connected layers are used for L2 regularization. Other parameters, including the biases and γ and β in BN layers, are left unregularized.

5.2. Experimental Results and Analysis

Figure 13 shows the loss and accuracy comparison results of ShuffleNetV2 1×, MobileNet V2, and ECA-ShuffleNetV2 1× models on the train and validation dataset. Obviously, ECA-ShuffleNetV2 1× outperforms the first two in terms of convergence speed and training stability, and has higher accuracy and lower loss value in the training and validation set, which verifies the effectiveness and reliability of the improved method proposed in this paper.

Grade-CAM [35] (Gradient-weighted Class Activation Mapping) is used to visualize the class activation mapping of models as a way of expressing the “visual interpretation” of CNN-based models without the need to modify the model’s structure or retrain it. Filling the regions of interest in the CNN model with highlights allows us to analyze whether the network is learning the correct features. In Grade-CAM heat maps, the brighter the color of an activated region indicates its greater relevance to a particular category.

In Figure 14, comparing the grade-CAM heat maps of corn seeds generated by the three models (ECA-ShuffleNetV2 1× (2, 2, 2); ShuffleNetV2 1× (2, 2, 2); ShuffleNetV2 1× (4, 8, 4)), it is not difficult to see that there are differences in the sensitivity of different models to key features. As can be seen in the heat map, the ECA-ShuffleNet V2 model is able to precisely focus on key features that distinguish corn seed disease categories from one another, such as disease spots and cracks. It indicates that this region was helpful for the model to identify the category of the disease and shows that the model extracted the important disease features well. In contrast, the other two models deal with a large amount of redundant information or ignore valid category feature information, which makes the models less effective at classification.

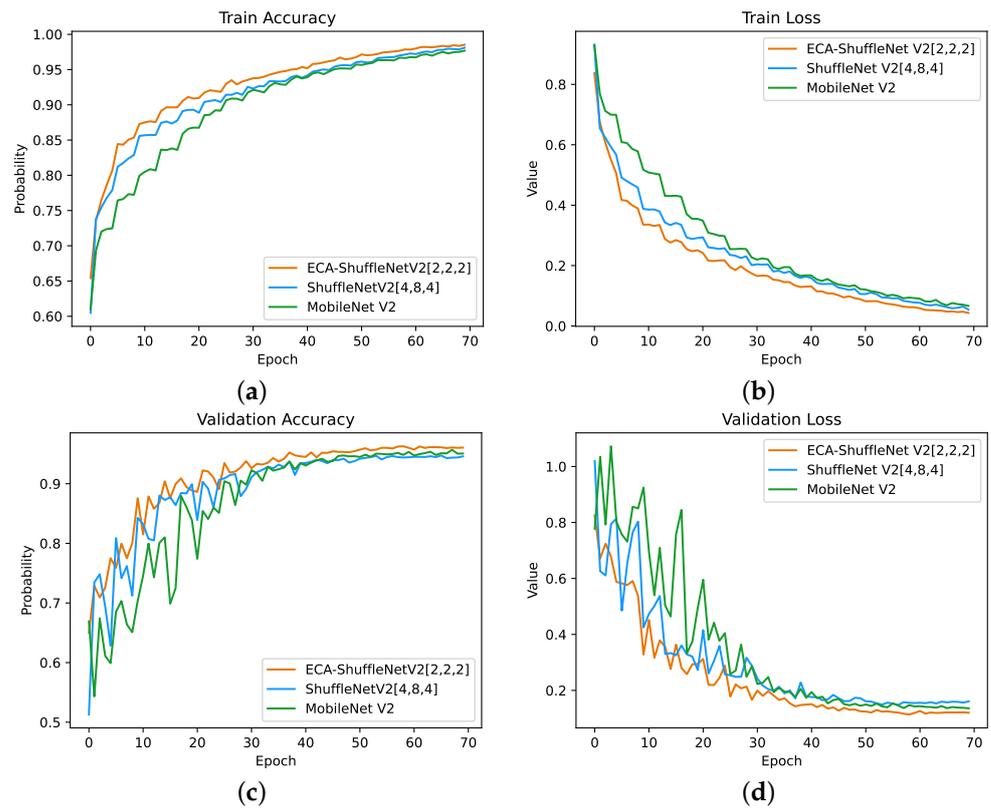


Figure 13. Comparison of accuracy and loss values for ECA-ShuffleNetV2 1×, ShuffleNetV2 1×, and MobileNet V2: (a) Training accuracy comparison convolution; (b) Training loss comparison; (c) Validation accuracy comparison convolution; (d) Validation loss comparison.

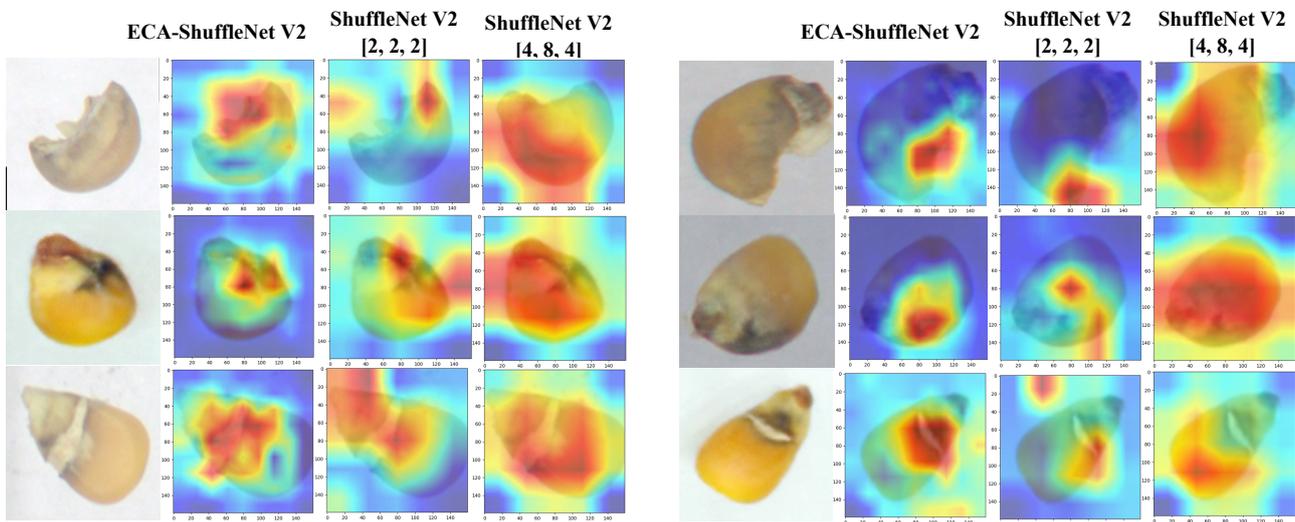


Figure 14. Grad-CAM heat map.

5.3. Ablation Experiments

The results of the ablation experiments of the improved model are shown in Table 4. Model 1 is a ShuffleNetV2 1× network, and Model 2 only reduces the number of iterations of Model block 1 to one, which reduces the depth of the network, reduces the number of parameters and FLOPs by 1/3, and improves accuracy by 0.07%. Model 3 only adds 3 × 3 DW convolution after block 2 of Model 1, which increases accuracy by 0.68% by introducing only a small number of parameters. Model 4 moved the DW convolution in branch 2 forward and modified the output feature layer channels from [24, 116, 232, 464, 1024] to [58, 116, 232, 464, 1024], the accuracy improved to 95.24%. Model 5 replaces

the 3×3 DW convolution with 7×7 DW convolution in the branch, and the accuracy is improved by 0.08. Model 6 adds the ECA attention module after Conv5, with almost no change in the number of parameters and FLOPs and an accuracy increase of 0.87 percentage points, with only a small increase in memory cost. Model 7 replaces the convolution and max pooling layers of Conv1 with an ordinary convolution, a kernel size of 4, and a stride of 4, and the accuracy rose by 0.47%. This method, Model 8 (ECA-ShuffleNetV2), uses the above six improvement strategies simultaneously on the basis of Model 1 (ShuffleNetV2 $1 \times$). Relative to Model 1, the accuracy improved to 96.28% with a decrease of nearly 30% in Params and a compression of nearly half in FLOPs, Memory cost, and Madd, resulting in a good improvement. Model 9 (MobileNet V2 $1 \times$) achieved 95.69% accuracy on this dataset, but Params, FLOPs, Memory cost, and Madd were all greater than those of ShuffleNetV2.

After training, confusion matrices were created for the three algorithms MobileNet V2 $1 \times$, ShuffleNetV2 $1 \times$, and ECA-ShuffleNetV2 $1 \times$, and the performance evaluation of the model was carried out using the values (TP, TN, FP, FN) on the confusion matrix. The confusion matrix for the validation set of the CNN model is shown in Figure 15. The confusion matrix is used to visualize the performance of the CNN model, with each column representing the true label of the sample and the rows of the matrix representing the predicted class of the classifier. These four metrics typically include true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) [36]. In this task, TP and TN correspond to the correct identification of diseased maize seed categories, while FP and FN correspond to misclassification. Model performance was evaluated according to the parameters associated with the confusion matrix, i.e., Accuracy, Precision, Recall, Specificity, F1_Score, which are defined as shown in Table 5. Performance was evaluated on the validation set [37].

Table 4. Results of ablation experiments.

Model	Params (M)	FLOPs (M)	Memory Cost (MB)	Madd (M)	Accuracy (%)
1	1.258	76.32	10.63	150.88	94.74
2	0.848	47.24	6.74	93.36	94.81
3	1.267	77.29	12.85	152.67	95.42
4	1.265	74.89	12.04	147.61	95.24
5	1.356	85.75	10.63	169.74	95.61
6	1.258	76.32	10.65	150.88	94.92
7	1.258	73.51	9.17	145.5	95.31
8	0.913	44.75	5.86	88.50	96.28
9	2.229	162.74	37.98	318.95	95.69

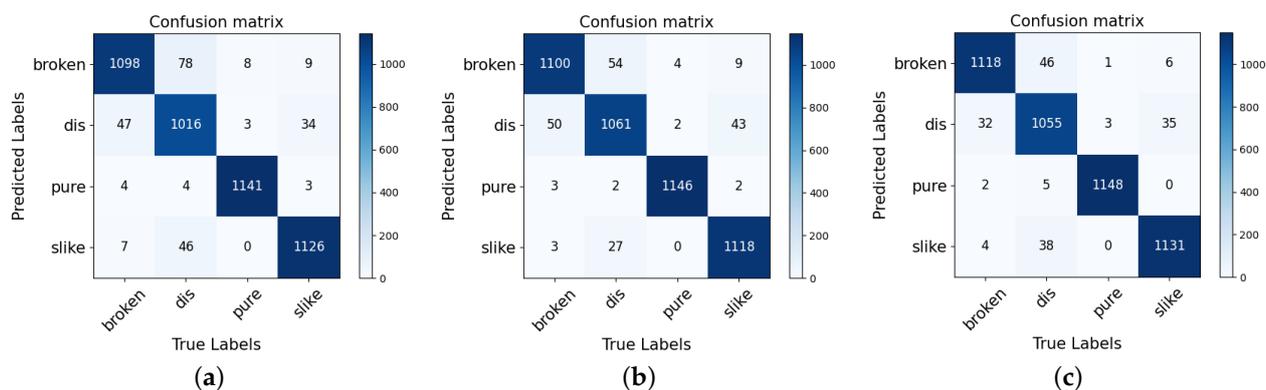


Figure 15. Confusion Matrix for the validation set: (a) ShuffleNetV2 $1 \times$; (b) MobileNet V2; (c) ECA-ShuffleNetV2.

A comparison of the confusion matrix’s performance metrics for the three models is shown in Tables 4 and 6, which include the Accuracy, Precision, Recall, Specificity, F1_Score of the models. Table 4 demonstrates that the model with the highest accuracy in disease identification is the ECA-ShuffleNetV2 $1 \times$ (96.28%) proposed in this paper, followed by MobileNet V2 $1 \times$ (95.69%), and ShuffleNetV2 $1 \times$ (94.74%). Moreover, it can be concluded

that ECA-ShuffleNetV2 1× has excellent detection performance both in terms of speed and accuracy, and this network can extract detailed information about the samples and provide a way of thinking for the processing of similar classification tasks in this area. The experimental results also show that appropriate dataset augmentation has a positive impact on small or unbalanced samples in classifier training. Additionally, CycleGAN effectively contributes to dataset augmentation and increases model accuracy. Meanwhile, the deep learning-based feature extraction method can efficiently and accurately retain the information of corn seed appearance and reduce the information loss caused by manual feature extraction.

Table 5. Performance evaluation metrics.

Metrics	Formula (M)	Evaluation Focus
Accuracy	$\frac{(TP + TN)}{(TP + FP + FN + TN)}$	The ratio of the number of correct predictions made by the classifier to the total number of predictions made by the classifier is measured.
Precision	$\frac{(TP)}{(TP + FP)}$	Denotes the percentage of samples in which the predicted outcome is a positive case and the true case is also a positive case.
Recall	$\frac{(TP)}{(TP + FN)}$	Represents the model's ability to correctly predict the positives out of actual positives.
Specificity	$\frac{(TN)}{(TN + FP)}$	Specificity is the metric that evaluates a model's ability to predict the true negatives of each available category.
F1_Score	$\frac{(2 * TP)}{(2 * TP + FP + FN)}$	F1_Score gives equal weight to both the Precision and Recall for measuring its performance in terms of accuracy.

Table 6. Classification results of 3 models: SN: ShuffleNetV2 1×; MN V2: MobileNet V2 1×; ECA-SN: ECA-ShuffleNetV2 1×.

Name	Precision (%)			Recall (%)			Specificity (%)			F1_Score (%)		
	SN	MN V2	ECA-SN	SN	MN V2	ECA-SN	SN	MN V2	ECA-SN	SN	MN V2	ECA-SN
Broken	92.04	94.26	95.47	94.98	95.16	96.71	97.26	98.07	98.47	93.49	94.71	96.09
Discolored	92.36	91.78	93.78	88.81	92.74	92.22	97.59	97.27	97.99	90.55	92.26	92.99
Pure	99.05	99.39	99.39	99.05	99.48	99.65	99.68	99.80	99.80	99.05	99.44	99.52
Silkcut	95.50	97.39	96.42	96.08	95.39	96.50	98.46	99.13	98.78	95.79	96.38	96.46

From Table 6, it can be seen that all three classifiers have the excellent recognition of healthy corn seeds, with precision over 99%, and poor recognition of discolored seeds. This is due to the fact that, in most cases, there is more than one form of deterioration in maize seeds during storage; for example, the presence of broken or discolored conditions causes the classifier to become tangled up in identifying specific disease conditions. However, the model identifies healthy corn seeds well, which is sufficient to determine whether corn seeds are up to standard. The ECA-ShuffleNetV2 model proposed in this paper has less than 1 M parameters, and the memory cost is only 5.86 MB; meanwhile, the precision of the model in identifying healthy seeds is 99.39%, and the number of misclassified seeds is small, which can basically meet the needs of mobile applications.

Table 7 shows the single image inference speed of the ShuffleNetV2 1×, MobileNet V2, MobileNet V3-Small, GhostNet 0.5×, and ECA-ShuffleNetV2 1× models on a portable laptop. The device is equipped with an Intel(R) Core(TM) i5-7200U CPU and 4G RAM without GPU acceleration. Input image resolution is 160 × 160. To ensure the accuracy of the test, the single image inference time is the average time it takes for 100 images to be inferred by the model. The table clearly shows that the ECA-ShuffleNetV2 1× achieves an inference time of 9.71 ms per image. This is expected, as ShuffleNetV2 was designed from the outset with inference latency in mind and does not focus solely on FLOPs.

Table 7. The single image inference speed comparison for five models. All results are evaluated with single thread. SN: ShuffleNetV2 1×; MN V2: MobileNet V2 1×; MN V3-S: MobileNet V3-Small; GN: GhostNet 0.5×; ECA-SN: ECA-ShuffleNetV2 1×.

Model	SN	MN V2	MN V3-S	GN	ECA-SN
Time (ms)	16.20	28.11	33.89	28.39	9.71

5.4. Related Work

Previously, Ferentinos [38] developed a deep learning model based on a convolutional neural network architecture specifically adapted to plant leaf detection, trained on a publicly available dataset of 87,848 photos, with experimental results showing that the VGG convolutional neural network achieved 99.53% accuracy (top-1 error of 0.47%) in the classification of the test set. Moreover, the inference time of the model is about 2 ms under a single GPU, which is suitable for the deployment of mobile devices.

Common classifier training methods are transfer learning and training from scratch. For the case of supervised learning, the most time-consuming steps are the production of convincing and relevant sample datasets and the significant time spent on parameter training; the overfitting and convergence states of CNN networks are also of concern. In problems similar to the classification of seed pests and diseases, transfer learning can be used to speed up the training of classifiers [39]. Gulzar et al. [40] built on the VGG16 network and trained the model using transfer learning; the model achieved 99% accuracy on the seed classification dataset. Hamid et al. [41] also performed experiments on this dataset using MobileNetV2 for seed classification, and the results showed an accuracy of 98% and 95% for the training and test sets, respectively. Although the classification accuracy is not as good as that of Gulzar et al.'s method, the MobileNetV2 model has better parameter scale, inference speed, and memory usage than VGG16 and is more suitable for applications in mobile scenarios.

The prediction accuracy of supervised deep learning models depends heavily on the amount and diversity of data available during training. Generally, when dealing with complex tasks, the amount of data available for training models is difficult to obtain. In addition, when there are large differences between research tasks or when the problem of imbalance in sample datasets is severe, ordinary simple dataset augmentation algorithms (e.g., padding, random rotating, darkening or brightening/color modification) may not be sufficient for the experimental needs. It is possible to overcome this challenge by using GAN-based dataset augmentation, such as how CycleGAN was used in this paper [42].

6. Conclusions and Future Work

This paper presents an improved ECA-ShuffleNetV2 network for corn seed disease identification. Production of the dataset requires only a low-cost digital camera. On a CPU device, this model has a single-threaded inference speed of about 9.71 ms and a classification accuracy of 96.28% on the validation set. It has 0.913 M parameters and 44.75 M FLOPs. The model is suitable for deployment on mobile devices, such as smart phones and portable laptops available to growers or quality assessment practitioners, as well as on offline mobile monitoring sites.

Compared with traditional machine learning, deep learning is more effective at pest and disease classification. In addition, the corn seed disease recognition model built in this paper has the advantages of high recognition accuracy, fast recognition speed, and relatively small model parameters. This model can quickly extract the characteristics of diseased seeds, which greatly reduces the workload of manual detection. It is possible for relevant departments to collect and produce different data sets of agricultural samples for quality evaluation model training of other cereals according to the actual situation, which has good application prospects.

Using machine vision, we can only obtain phenotypic information about seeds; however, that information does not describe their internal characteristics. Therefore, in the follow-up work, we will combine our method with hyperspectral technology in order to

detect and classify seed diseases and enhance the model's robustness so that it works well on similar tasks.

According to a comparison with related research at this stage, the limited number of seed samples chosen for this study is not representative of all diseases present in Chinese maize seeds today. Particularly, diseased seeds are much less common than healthy seeds, which is a huge test of our work. In the next study, a complete maize seed image acquisition system will be designed to address the problem. To build a comprehensive and accurate data set of maize seed diseases, we will collaborate with relevant departments. We will also ensure its classification quality in order to improve the practical utility of the model.

While the acquisition of datasets is critical, the success of supervised machine learning is not possible without high-quality data annotation. Manual annotation is labor-intensive and time-consuming, so relying only on manual annotation is not a wise decision. Our future research will use unsupervised machine learning to classify and label maize seed diseases. This will further reduce the workload and accelerate the deployment of the model in agricultural quality assessment.

In addition, this experiment did not consider the relationship between external factors, such as different corn growing regions, corn varieties, and climate in China, and corn seed diseases. These external factors are also crucial to the quality assessment of agricultural products and food waste reduction. As a result, follow-up work will include further experiments and investigations related to this topic.

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

Funding: This work was supported by the Yantai Science and Technology Innovation Development Plan Project (Grant No. 2022XDRH015).

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Phupattanasilp, P.; Tong, S.-R. Augmented Reality in the Integrative Internet of Things (AR-IoT): Application for Precision Farming. *Sustainability* **2019**, *11*, 2658.
2. Kong, J.; Wang, H.; Wang, X.; Jin, X.; Fang, X.; Lin, S. Multi-stream hybrid architecture based on cross-level fusion strategy for fine-grained crop species recognition in precision agriculture. *Comput. Electron. Agric.* **2021**, *185*, 106134.
3. Watson, A.; Burgess, L.W.; Summerell, B.; O'Keeffe, K. Fusarium species associated with cob rot of sweet corn and maize in New South Wales. *Australas. Plant Dis. Notes* **2014**, *9*, 142.
4. Sastry, K.S. *Seed-Borne Plant Virus Diseases*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 85–100.
5. Schmidt, M.; Horstmann, S.; Colli, L.D.; Danaher, M.; Speer, K.; Zannini, E.; Arendt, E.K. Impact of fungal contamination of wheat on grain quality criteria. *J. Cereal Sci.* **2016**, *69*, 95–103.
6. Franco-Duarte, R.; Černáková, L.; Kadam, S.; Kaushik, K.S.; Salehi, B.; Bevilacqua, A.; Corbo, M.R.; Antolak, H.; Dybka-Stępień, K.; Leszczewicz, M.; et al. Advances in Chemical and Biological Methods to Identify Microorganisms—From Past to Present. *Microorganisms* **2019**, *7*, 130.
7. Li, J.; Wu, J.; Lin, J.; Li, C.; Lu, H.; Lin, C. Nondestructive Identification of Litchi Downy Blight at Different Stages Based on Spectroscopy Analysis. *Agriculture* **2022**, *12*, 402.
8. Lu, Z.; Zhao, M.; Luo, J.; Wang, G.; Wang, D. Design of a winter-jujube grading robot based on machine vision. *Comput. Electron. Agric.* **2021**, *186*, 106170.
9. Zhang, J.; Dai, L.; Cheng, F. Classification of Frozen Corn Seeds Using Hyperspectral VIS/NIR Reflectance Imaging. *Molecules* **2019**, *24*, 149.
10. Javanmardi, S.; Ashtiani, S.M.; Verbeek, F.J.; Martynenko, A. Computer-vision classification of corn seed varieties using deep convolutional neural network. *J. Stored Prod. Res.* **2021**, *92*, 101800.

11. Wang, Z.; Huang, W.; Tian, X.; Long, Y.; Li, L.; Fan, S. Rapid and Non-destructive Classification of New and Aged Maize Seeds Using Hyperspectral Image and Chemometric Methods. *Front. Plant Sci.* **2022**, *13*, 849495.
12. Yang, D.; Jiang, J.; Jie, Y.; Li, Q.; Shi, T. Detection of the moldy status of the stored maize kernels using hyperspectral imaging and deep learning algorithms. *Int. J. Food Prop.* **2022**, *25*, 170–186.
13. Mishra, S.; Sachan, R.; Rajpal, D. Deep Convolutional Neural Network based Detection System for Real-time Corn Plant Disease Recognition. *Procedia Comput. Sci.* **2020**, *167*, 2003–2010.
14. Meng, R.; Lv, Z.; Yan, J.; Chen, G.; Zhao, F.; Zeng, L.; Xu, B. Development of Spectral Disease Indices for Southern Corn Rust Detection and Severity Classification. *Remote Sens.* **2020**, *12*, 3233.
15. Albarrak, K.; Gulzar, Y.; Hamid, Y.; Mehmood, A.; Soomro, A.B. A Deep Learning-Based Model for Date Fruit Classification. *Sustainability* **2022**, *14*, 6339.
16. Padilla, D.A.; Pajes, R.A.I.; Guzman, J.T.D. Detection of Corn Leaf Diseases Using Convolutional Neural Network with OpenMP Implementation. In Proceedings of the 2020 IEEE 12th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM), Manila, Philippines, 3–7 December 2020; pp. 1–6.
17. Caballero, D.; Calvini, R.; Amigo, J.M. Chapter 3.3—hyperspectral imaging in crop fields: Precision agriculture. In *Hyperspectral Imaging*; Amigo, J.M., Ed.; Data handling in science and technology; Elsevier: Amsterdam, The Netherlands, 2019; Volume 32, pp. 453–473.
18. Han, K.; Wang, Y.; Tian, Q.; Guo, J.; Xu, C.; Xu, C. GhostNet: More Features From Cheap Operations. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 1577–1586.
19. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.-C. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520.
20. Ma, N.; Zhang, X.; Zheng, H.T.; Sun, J. ShuffleNet v2: Practical guidelines for efficient cnn architecture design. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 116–131.
21. Che'Ya, N.; Mohidem, N.A.; Roslin, N.; Saberioon, M.; Tarmidi, Z.; Shah, J.; Ilahi, W.; Man, N. Mobile Computing for Pest and Disease Management Using Spectral Signature Analysis: A Review. *Agronomy* **2022**, *12*, 967.
22. Shendryk, Y.; Sofonia, J.; Garrard, R.; Rist, Y.; Skocaj, D.; Thorburn, A.P. Fine-scale prediction of biomass and leaf nitrogen content in sugarcane using UAV LiDAR and multispectral imaging. *Int. J. Appl. Earth Obs. Geoinf.* **2020**, *92*, 102177.
23. Zhu, J.; Park, T.; Isola, P.; Efros, A.A. Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2242–2251.
24. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 11531–11539.
25. Nagar, S.; Pani, P.; Nair, R.; Varma, G. Automated Seed Quality Testing System using GAN & Active Learning. *arXiv* **2021**, arXiv:2110.00777.
26. Brock, A.; Donahue, J.; Simonyan, K. Large Scale GAN Training for High Fidelity Natural Image Synthesis. *arXiv* **2018**, arXiv:1809.11096.
27. Hu, J.; Shen, L.; Sun, G. 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.
28. Krizhevsky, A. Learning Multiple Layers of Features from Tiny Images. 2009. Available online: <http://www.cs.utoronto.ca/~kriz/learning-features-2009-TR.pdf> (accessed on 13 October 2022).
29. Howard, A.; 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.
30. Zhang, X.; Zhou, X.; Lin, M.; Sun, J. ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6848–6856.
31. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is All you Need. In *Advances in Neural Information Processing Systems*; Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2017; Volume 30.
32. Liu, Z.; Mao, H.; Wu, C.Y.; Feichtenhofer, C.; Darrell, T.; Xie, S. A convnet for the 2020s. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 19–20 June 2022; pp. 11976–11986.
33. Ding, X.; Zhang, X.; Han, J.; Ding, G. Scaling up your kernels to 31×31 : Revisiting large kernel design in cnns. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 19–20 June 2022; pp. 11963–11975.
34. He, T.; Zhang, Z.; Zhang, H.; Zhang, Z.; Xie, J.; Li, M. Bag of Tricks for Image Classification with Convolutional Neural Networks. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 558–567.
35. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. *Int. J. Comput. Vis.* **2020**, *128*, 336–359.
36. Xu, P.; Yang, R.; Zeng, T.; Zhang, J.; Zhang, Y.; Tan, Q. Varietal classification of maize seeds using computer vision and machine learning techniques. *J. Food Process. Eng.* **2021**, *44*, e13846.

37. Feng, J.; Sun, Y.; Zhang, K.; Zhao, Y.; Ren, Y.; Chen, Y.; Zhuang, H.; Chen, S. Autonomous Detection of *Spodoptera frugiperda* by Feeding Symptoms Directly from UAV RGB Imagery. *Appl. Sci.* **2022**, *12*, 2592.
38. Ferentinos, K.P. Deep learning models for plant disease detection and diagnosis. *Comput. Electron. Agric.* **2018**, *145*, 311–318.
39. Ullah, N.; Khan, J.; Khan, M.; Khan, W.; Hassan, I.; Obayya, M.; Negm, N.; Salama, A. An Effective Approach to Detect and Identify Brain Tumors Using Transfer Learning. *Appl. Sci.* **2022**, *12*, 5645.
40. Gulzar, Y.; Hamid, Y.; Soomro, A.B.; Alwan, A.A.; Journaux, L. A Convolution Neural Network-Based Seed Classification System. *Symmetry* **2020**, *12*, 2018.
41. Hamid, Y.; Wani, S.; Soomro, A.B.; Alwan, A.A.; Gulzar, Y. Smart Seed Classification System based on MobileNetV2 Architecture. In Proceedings of the 2022 2nd International Conference on Computing and Information Technology (ICCIT), Tabuk, Saudi Arabia, 25–27 January 2022; pp. 217–222.
42. Shorten, C.; Khoshgoftaar, T.M. A survey on Image Data Augmentation for Deep Learning. *J. Big Data* **2019**, *6*, 60.