



Article

RAHC_GAN: A Data Augmentation Method for Tomato Leaf Disease Recognition

Hongxia Deng , Dongsheng Luo , Zhangwei Chang, Haifang Li and Xiaofeng Yang

Department of Information and Computer, Taiyuan University of Technology, TaiYuan 030024, China; denghongxia@tyut.edu.cn (H.D.); changzhangwei0572@link.tyut.edu.cn (Z.C.); lihaifang@tyut.edu.cn (H.L.); xianfengyang0042@link.tyut.edu.cn (X.Y.)

* Correspondence: luodongsheng0369@link.tyut.edu.cn

Abstract: Accurate recognition of tomato diseases is of great significance for agricultural production. Sufficient and insufficient training data of supervised recognition neural network training are symmetry problems. A high precision neural network needs a large number of labeled data, and the difficulty of data sample acquisition is the main challenge to improving the performance of disease recognition. The tomato leaf data augmented by the traditional data augmentation methods based on geometric transformation usually contain less information, and the generalization is not strong. Therefore, a new data augmentation method, RAHC_GAN, based on generative adversarial networks is proposed in this paper, which is used to expand tomato leaf data and identify diseases. In this method, continuous hidden variables are added at the input of the generator, and the purpose is to continuously control the size of the generated disease area and to supplement the intra class information of the same disease. Additionally, the residual attention block is added to the generator to make it pay more attention to the disease region in the leaf image; a multi-scale discriminator is also used to enrich the detailed texture of the generated image and finally generate leaves with obvious disease features. Then, we use the images generated by RAHC_GAN and the original training images to build an expanded data set, which is used to train four kinds of recognition networks, AlexNet, VGGNet, GoogLeNet, and ResNet, and the performance is evaluated through the test set. Experimental results show that RAHC_GAN can generate leaves with obvious disease features, and the generated expanded data set can significantly improve the recognition performance of the classifier. Furthermore, the results of the apple, grape, and corn data set show that RAHC_GAN can also be used as a method to solve the problem of insufficient data in other plant research tasks.



Citation: Deng, H.; Luo, D.; Chang, Z.; Li, H.; Yang, X. RAHC_GAN: A Data Augmentation Method for Tomato Leaf Disease Recognition. *Symmetry* **2021**, *13*, 1597.

<https://doi.org/10.3390/sym13091597>

Academic Editor: Basil Papadopoulos

Received: 8 July 2021

Accepted: 30 August 2021

Published: 31 August 2021

Keywords: data augmentation; generative adversarial network; tomato disease recognition; insufficient training data

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

Tomato is one of the crops planted all over the world and is rich in nutritional value [1]. Most studies have shown that the intake of tomatoes or tomato products has an inhibitory effect on some cancers (such as lung cancer, gastric cancer, etc.), which is beneficial for multiple health outcomes in humans [2,3]. However, in the cultivation process of tomatoes, due to the influence of climate, environment, pests, and other factors, they will be infected with different kinds and degrees of diseases. These diseases are highly infectious, widely spread, and harmful; are not conducive to the growth of tomatoes; and also affect the yield and quality of fruit. Therefore, it is of great significance to accurately identify tomato diseases and carry out early prevention and timely treatment. With the development of deep learning, convolutional neural networks (CNN) have made major breakthroughs in the field of disease recognition. Most researchers [4–8] improved the architecture of neural network models and applied them to the recognition of plant diseases, which has achieved good results. However, a neural network with good performance usually requires a large

number of parameters, and a large amount of data is required to make these parameters work correctly. To obtain accurate disease data, manual collection and labeling are needed, which are time-consuming, labor-intensive, and error-prone. Therefore, in the field of tomato disease recognition, it is often impossible to obtain enough data to train the neural network, which is also the main reason that affects the further improvement of tomato disease recognition accuracy.

In the field of computer vision, data augmentation is a typical and effective method to deal with small data sets, including supervised and unsupervised methods. The unsupervised data augmentation method expands samples by geometric transformation [9–12] (such as flipping, clipping, rotation, scaling, etc.), which is simple and operable, so it is widely used in the data processing of diseased leaves. Liu et al. [13] used conventional image rotation, brightness adjustment, and principal component analysis methods to expand the image, solve the problem of insufficient images of apple disease, and achieve good recognition results. Abas et al. [14] used rotation, translation, scaling, and other methods to enhance the plant image data and used an improved VGG16 network to classify plants, so as to solve the problem of overfitting caused by too few plant samples. This kind of generation method can generate a large amount of data, which solves the problem of insufficient data to a certain extent but still has the following shortcomings: the image expanded by geometric transformation contains less information and is temporary, and a partially expanded image is not conducive to the training of the model, is not applicable to some tasks, and even reduces the performance of the model. At the same time, due to the large randomness of the generated data, there is no consistent target, which will produce a large number of redundant samples. In order to generate images that are more beneficial to specific tasks, an unsupervised image generation method based on learning has been proposed [15–18]. This kind of method uses different geometric transformation sequences to transform images, finds the most beneficial combination of classification results through reinforcement learning to enhance data, and adaptively adjusts it according to the current training iteration, finally generating a customized, high-performance data augmentation plan. However, this method is time-consuming and resource-intensive for training, so it has limitations on some tasks.

GAN [19] (generative adversarial network) is another data augmentation method based on unsupervised methods. It has been proven to be able to generate images with a similar distribution to the original data, showing subversive performance in image generation [20–23]. Clément et al. [24] used a GAN-based image augmentation method to enhance the training data and segmented the apple disease on tree crown with a smaller data set. Compared with the results without image generation, the F1 value was increased by 17%. The DCGAN (deep convolution generative adversarial network) [25] used a CNN (convolutional neural network) to replace the multilayer perceptron in GAN to improve the quality of the generated images. Purbaya et al. [26] used a regularization method to improve DCGAN and generated plant leaf data to prevent the model from overfitting. Although DCGAN improves the quality of the generated images, it cannot control the classes of the generated images, limiting the generation of multi-class images. To this end, ACGAN (generative adversarial network with an auxiliary classifier) [27] is proposed, which adds class information constraints to the generator and an auxiliary classifier to the discriminator to classify the output image, so that it has the function of image classification and can guide the generator to generate images of different classes. In addition, in order to generate higher quality data, many new architectures based on GAN have been proposed and applied to the field of plant science [25,28–32]. For the problem of insufficient data for apple diseases, Tian et al. [33] used CycleGAN [34] to generate images, which enriched the diversity of training data. Zhu et al. [35] proposed a data augmentation method based on cGAN, adding condition information to the input of generator to control the class of the generated image. In addition, dense connection was added to the generator and discriminator to enhance the information input of each layer and significantly improve the classification performance. Liu et al. [36] added dense

connections in GAN, and improved the loss function to generate grape leaves to achieve better recognition results in the same recognition network. However, the goal of GAN is to generate images that are as real as possible. It learns the overall information of the image, ignoring the key local disease information, and cannot generate images with clear disease spots. At the same time, due to the lack of detail texture, most of the images generated are of poor quality.

In order to solve the problem of insufficient data for tomato disease identification and insufficient disease features of expanded data, this paper proposes an improved ACGAN method RAHC_GAN for data augmentation (Figure 1a) and establishes an expanded data set of original data and generated data using RAHC_GAN (Figure 1b), so as to meet the data required for neural network training and improve the performance of the classifier in tomato leaf disease recognition. The innovations and contributions are as follows:

- Adding a set of continuous hidden variables to the input of the generative adversarial network to improve the diversity of the generated images. Due to the intra-class differences of the same tomato disease being small, the traditional generative adversarial network has difficulty learning the intra-class differences, and then it will generate similar images, resulting in the phenomenon of mode collapse [37]. In order to avoid this phenomenon, we use the hidden variable and class label to generate tomato leaves with different diseases. The class label is used to generate specific disease classes, and the hidden variable is used to improve the potential changes in the same class. For each class of disease, we capture the potential differences within the class by changing the value of the hidden variable, such as the area size and severity of the disease, so as to supplement the information within the class and enrich the diversity of the generated images.
- The fusion of residual attention block and multi-scale discriminator to enrich disease information of generated pictures. We add residual attention blocks to the generator. The residual network deepens the network depth and avoids network degradation. The attention mechanism makes the generator pay more attention to the disease information in the leaves from the perspectives of channel and space and guides the generation of tomato diseased leaves with obvious disease features. In addition, we introduce a multi-scale discriminator, which can capture different levels of information in the image, enrich the texture and edges of the generated leaves, and make the generated leaves more complete, richer in detail, and clearer in texture.
- Using RAHC_GAN to expand the training set to meet the large amount of data needed for neural network training. We use the expanded data set as the training set, train four kinds of recognition networks (AlexNet, VGGNet, GoogLeNet, and ResNet) through transfer learning, and use the test set for performance evaluation. The experimental results show that in multiple recognition networks, the performance of the expanded data set is better than that of the original training set.

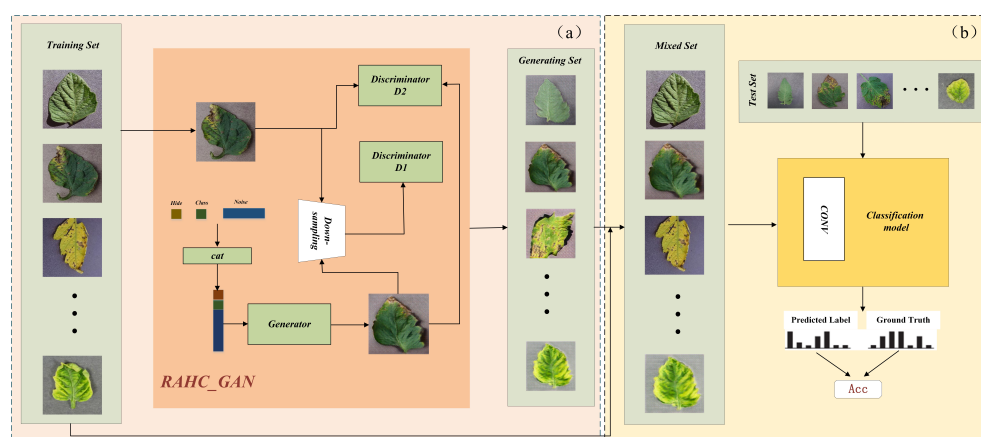


Figure 1. Schematic of the proposed strategy. This strategy consists of two parts: (a) using the proposed method, RAHC_GAN, to generate tomato diseased leaves; (b) the generated data are extended to the training set to train the classifier, so as to improve the disease recognition performance of the classifier.

The rest of this paper is organized as follows: Section 2 introduces related work and the model we proposed. Then, Section 3 shows the experimental results and analyses. Section 4 shows the related discussions. Section 5 contains the conclusions and future prospects.

2. Materials and Methods

2.1. Related Works

2.1.1. ACGAN

The working principle of GAN is that the generator generates a fake image, and the discriminator determines whether the input image is real or fake and achieves balance through mutual games. In the traditional GAN, the input of the generator is a random noise, so it is impossible to generate images of specified classes. In order to generate images of different classes accurately, it is necessary to train a generator for each class, which is cumbersome and involves a heavy workload. To solve this problem, ACGAN was proposed. It connects the class label c with the random noise z and adds a classifier in the discriminator to guide the training to control the generator to generate specified class images, which overcomes the limitation of GAN in multi-class data generation.

The main goal of our work is to generate six different tomato leaf disease images and use them as an extended data set to train the disease recognition network to improve the accuracy of disease recognition. However, in tomato leaves, the early features of some diseases are not obvious, and the similarity between the same diseases is high. Therefore, relying on the class label alone is not enough to generate diseased leaves with obvious features.

2.1.2. CBAM

The CBAM (convolutional block attention module) [38] is a lightweight attention module composed of a channel attention module and a spatial attention module. For a given feature map, CBAM infers the attention map along the channel and space dimensions and then multiplies the attention map with the input feature map to refine the features, so that the network focuses on the parts that are more effective for the task, thereby improving the performance of the model. The network structure of CBAM is shown in Figure 2.

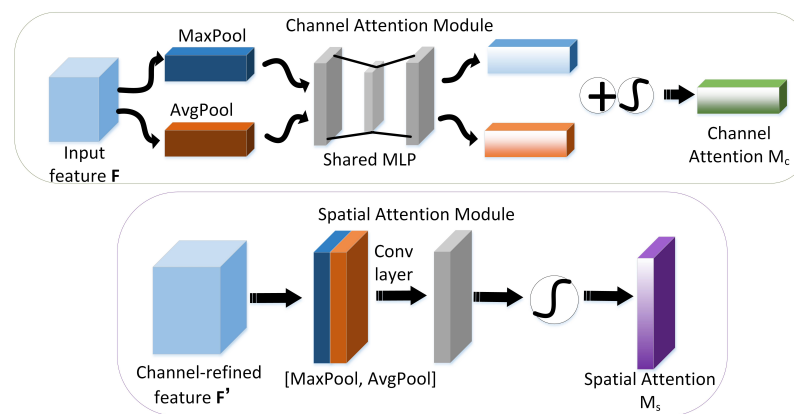


Figure 2. The network structure of CBAM.

For the input feature map F , the channel attention map M_c is first generated through the channel attention module and dot-multiplied with the original feature map F to obtain F' . Then the spatial attention map M_s is generated by the spatial attention module. The final output F'' is obtained by dot multiplication with F' . The calculation process is shown in Equations (1) and (2).

$$F' = M_c(F) \otimes F \quad (1)$$

$$F'' = M_s(F') \otimes F' \quad (2)$$

The channel attention module aggregates the spatial information of the feature map through average pooling and maximum pooling and generates average pooling features F_{avg}^c and maximum pooling features F_{max}^c . Then, the channel attention map $M_c \in R^{C \times 1 \times 1}$ is generated through the shared network. The calculation process of the channel attention module is shown in Equation (3).

$$M_c(F) = \sigma(MLP(AvgPool(F)) + MLP(MaxPool(F))) \quad (3)$$

The spatial attention module generates $F_{avg}^s \in R^{1 \times H \times W}$ and $F_{max}^s \in R^{1 \times H \times W}$ through the average pooling layer and the maximum pooling layer, and then performs convolution operations to generate the space attention map. The calculation process of the spatial attention module is shown in Equation (4).

$$M_s(F) = \sigma(f^{7 \times 7}([AvgPool(F)]; MaxPool(F))) \quad (4)$$

2.2. Building RAHC_GAN for Tomato Leaf Disease Recognition

2.2.1. The Overview of RAHC_GAN

In the leaf disease recognition network, there is a strong symmetric relationship between sufficient and insufficient training data. Sufficient data make the neural network achieve better performance. Therefore, in the case of insufficient training data, we used the method of GAN to enhance the data. Inspired by Odena, A [27], this paper improved the network architecture of ACGAN and proposed a generative adversarial network, RAHC_GAN, which is used to generate tomato leaf images with obvious disease features, so as to improve the recognition accuracy of the recognition network. The overall network structure of RAHC_GAN is shown in Figure 3. RAHC_GAN uses the hidden variable mechanism to continuously control the generation of the diseased area; through the generator with a residual block and attention mechanism, more obvious leaf features were extracted to generate tomato leaves with prominent disease, and the multi-scale discriminator is used to capture the high-frequency information lost in the image and enrich the detail texture of the image.

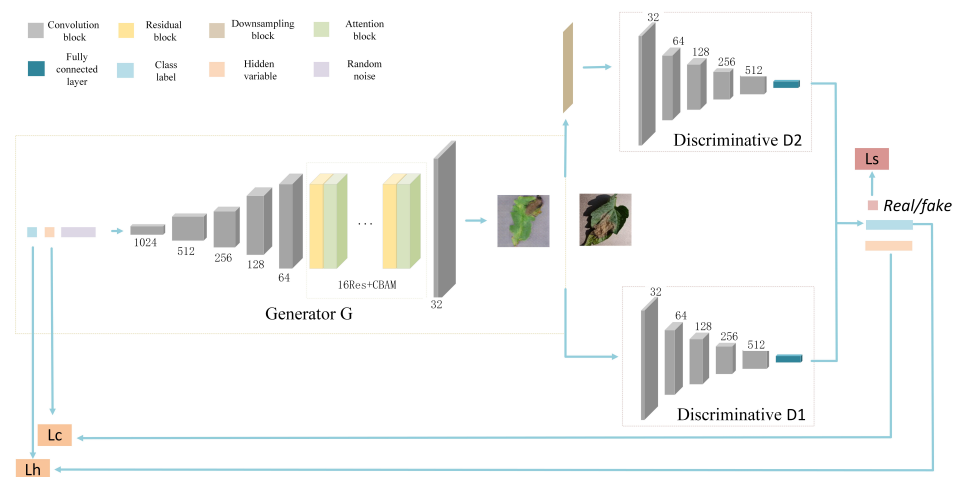


Figure 3. The network structure of RAHC_GAN. The input is composed of class label c , hidden variable h , and random noise z . The image is down-sampled using a multi-scale discriminator strategy, and the outputs are real probability, class label, and reconstructed hidden variables.

2.2.2. The Generative Network G

In order to control the disease class of the generated leaves and enhance the generator's attention to the diseased area, this paper used the generator network structure as shown in Figure 4 (a). It extracts high-level features in the image through the residual attention block, so it can generate images that are more conducive to recognition tasks.

The generator is composed of multiple deconvolution layers. The input of the generator is a random vector composed of class label, hidden variable, and random noise. After six layers of deconvolution operations, the output tensor is obtained. Finally, a $128 \times 128 \times 3$ image is generated through the tanh activation layer.

The original ACGAN controls the class of generated images through a set of discrete class labels. Inspired by this, we added a set of continuous variables as a supplement to the input information. Since the class label is a discrete value of 0, 1, we defined a $1 \times 1 \times 6$ hidden variable $Hide \in [-1, 1]$. Similar to the class label, the six dimensions represent six classes of diseases respectively, since this paper studies a single disease of a single leaf image and does not consider the existence of multiple diseases in one image. Therefore, for the hidden variable of a specified class of disease, we set its dimension to a random value of -1 to 1 and set other dimensions to 0 to generate the image of this class, so that the size of the disease area and the disease severity of the generated image can change with the continuous change of the hidden variables. Thus, the introduced hidden variables can guide the generation of images with different disease sizes and regions and increase the diversity of generated images.

However, the existing generators can generate images, considering that the deeper network is beneficial to the improvement of model performance. In this paper, 16 residual blocks [39] are added before the last deconvolution layer of the generator. On one hand, the network depth is deepened to enable the generator to obtain high-level features of images, and on the other hand, the degradation problem caused by networks with a too deep depth is avoided.

Accurate recognition of tomato diseases requires accurate disease features. Some of the early-stage features of the diseases are relatively small. However, the existing GAN often ignores the tiny disease features when generating images. In order to pay attention to the disease area of the image, we add a CBAM block before the accumulation layer of each residual block to help the model judge more beneficial features from two aspects: channel and space. Channel attention makes the network pay attention to the diseased area, and spatial attention makes the network know which parts to pay attention to, so that the generator focuses on the diseased area and de-noises irrelevant information to generate disease images with obvious disease features.

2.2.3. The Discriminative Network D

The discriminator is composed of sequentially stacked convolutional layers, and the input is a real or generated image. The input image is subjected to a six-layer convolution operation to obtain a feature vector, and the feature vector is transformed into a one-dimensional vector through the flatten operation of the sigmoid function and soft-max function, respectively, which is used to indicate the probability of the input image as a real image and the probability of being classified. In addition, for the additional hidden variable $h = (h_0, \dots, h_n)$ introduced at the input, our purpose is to independently control the inter-class details of the same disease. In this way, similar to the class label c , the hidden variable h should be independent, similar to random noise. Therefore, for the image generated by using a hidden variable, after the feature extraction of the discriminator, it should be possible to separate the various components $h' = (h'_0, \dots, h'_n)$ of h [40], which is called the restored hidden variable. Correspondingly, since the hidden variable at the input takes the value $[-1, 1]$, we use the tanh function to flatten the feature vector to obtain the restored hidden variable and the hidden variable loss to constrain it. The overall architecture of the discriminator is shown in Figure 4b.

The images generated by traditional GAN often have the problem of a fuzzy texture structure and lack of high-frequency information. In order to capture the image distribution at different scales and recover the high-frequency information lost in the image, we proposed a multi-scale parallel learning discriminator architecture [41], which used two discriminant scales, namely the original 128×128 image scale and the down sampling 64×64 image scale. We input them into the corresponding discriminator for feature extraction and subsequent operation and integrate multi-scale data by feature fusion. Among them, the discriminator with a smaller input size has a larger field and guides the generator to generate whole continuous plant leaves, and the discriminator with a larger input size guides the generator to generate a smaller area, such as the boundary of the lesion. Since images with different resolutions contain different levels of visual features, the proposed multi-scale discriminator can increase the discriminator's receptive field and make the generated images more detailed and contain a clearer texture.

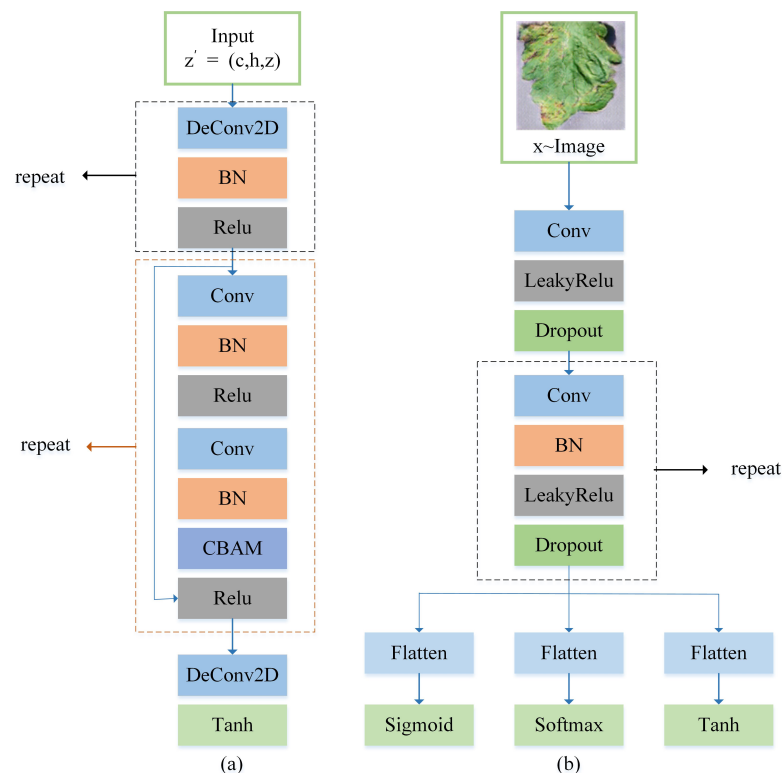


Figure 4. The network structure of generator (a) and discriminator (b).

2.2.4. Loss Function

Adversarial Loss: In order to make the generator “cheat” the discriminator and generate indistinguishable images, the adversarial loss function of the original GAN is used, as shown in Equation (5):

$$L_s = E_{x \sim p_{data}(x)} [\log D[x]] + E_{z \sim p_z(z)} [\log [1 - D[G[z]]]] \quad (5)$$

where x represents the real data sampled from the real data $p_{data}(x)$, and z represents the d -dimensional noise vector sampled from the Gaussian distribution p_z .

Classification Loss: For a given class label c , the generator needs to generate the corresponding class image X_{fake} . The real image pair (c, X_{real}) is given by the training data, and the classification loss is shown in Equation (6).

$$L_c = E[\log P[C = c | X_{real}]] + E[\log P[C = c | X_{fake}]] \quad (6)$$

Hidden variable loss: Due to the instability of the GAN's training, pattern collapse often occurs. Therefore, we added hidden variable loss to GAN to enhance the diversity of generated images and reduce the occurrence of mode collapse. The definition of hidden variable loss is shown in Equation (7):

$$L_h = \sum_{i=0}^n (h'_i - h_i)^2 \quad (7)$$

where $h = (h_0, \dots, h_n)$ represents the hidden variable input, and $h' = (h'_0, \dots, h'_n)$ represents the restored hidden variable obtained after RAHC_GAN.

Specifically, mode collapse means that different input vectors generate the same image through the generator. For different hidden variables h_1, h_2 , they are connected with a label vector and random noise as the input of the generator z_1, z_2 , further generating images. Then, the discriminator extracts the features of the generated images and tries to restore the hidden variables through the fully connected layer. If there is a mode collapse phenomenon in GAN, the images generated $G(z_1) = G(z_2)$ after different inputs z_1, z_2 are passed through the generator. When the image $G(z_1) = G(z_2)$ is input to the discriminator, the discriminator extracts the same features and, through the full connection layer restoration, the restored hidden variables $h'_1 = h'_2$. While the input hidden variables $h_1 \neq h_2$, at this time, the hidden variable loss calculated by Equation 7 will be very large. In order to reduce the loss, the RAHC_GAN will continuously optimize the generated images and generate different images for different inputs, which can improve the diversity of generated images and reduce the occurrence of model collapse.

Full Objective: Finally, the objective functions of the generator and discriminator are defined as Equations (8) and (9), respectively:

$$L_G = L_s + L_c + \lambda L_h \quad (8)$$

$$L_D = -L_s + L_c \quad (9)$$

where λ is a hyperparameter, and $\lambda = 1$ is used in all experiments.

2.2.5. Recognition Model for Tomato Leaf Disease Identification

In this paper, a convolutional neural network is applied to tomato disease recognition, mainly to explore whether the image generated by RAHC_GAN is beneficial to the improvement of recognition performance, so the performance of different recognition networks is not compared. Moreover, in order to explore the performance of the expanded data set on different recognition models, we conducted experiments on four classic recognition networks, AlexNet [42], VGGNet [43], GoogLeNet [34], and ResNet [39]. The main algorithm is shown in Table 1.

Table 1. Tomato leaf disease recognition algorithm based on RAHC_GAN data augmentation.

Algorithm 1	
INPUT: Training set X , Test set X_c , the epoch of RAHC_GAN E_d , the epoch of recognition network E_c , the batch size of RAHC_GAN b_h , the batch size of recognition network b_c	
1.	Initialize RAHC_GAN network parameters;
2.	for epoch 1 \leftarrow to E_d do
3.	//RAHC_GAN training
4.	for epoch = 1, ..., E_d do
5.	Sampling b_h group data (x^z, y) , where $x^z \sim (c, h, z)$, $y \sim p_g(y)$; Sampling b_h group real data $(x^r, y) \sim p_r(X, y)$;
6.	Update the discriminator network by Equations (9);
7.	Update the generator network by Equations (8);
8.	end
9.	// RAHC_GAN completes training;
10.	Generate data $X_g \sim (x_g, y)$ through trained RAHC_GAN
11.	$X_e = X \cup X_g$
12.	Initialize the network parameters of the recognition network;
13.	// recognition network training
14.	for epoch \leftarrow 1 to E_c do
15.	Sampling b_c group data $(x_e, y) \sim p_e(X_e, y)$;
16.	Update the recognition network through gradient descent;
17.	Sampling b_c group data $(x_c, y) \sim p_r(X_c, y)$, use recognition network to calculate the recognition accuracy;
18.	end
OUTPUT: Recognition accuracy rate of test set X_c	

3. Results

3.1. Data Set

In this paper, tomato data from the Plant Village data set are used for experimentation. The Plant Village data set is the only public data set on plant leaf disease research at present. We selected six classes of data from tomato diseases, including five classes of diseased leaf data and healthy leaf data, as shown in Figure 5. In order to maintain the data balance, we selected 1250 images for each class of diseased leaves from the Plant Village data set and used an 80–20% train-test distribution [44] to divide them into a training set (Train (origin)) and test set (Test). The specific information of the data set is shown in Table 2. In Table 2, Train (origin + augment) represents the expanded data set, which is composed of Train (origin) and the data generated by the augmentation method. Specifically, for each class of data in Train (origin + augment), it includes 1000 original data points in Train (origin) and 1000 data points generated by the augmentation method.

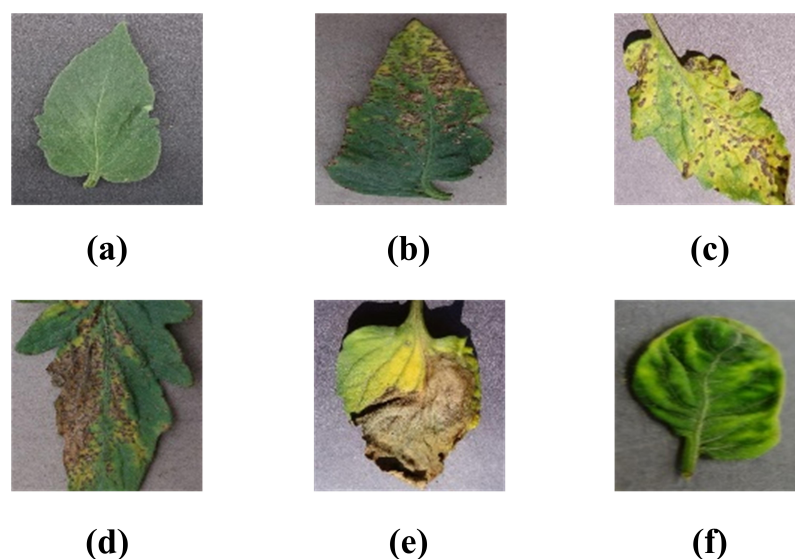


Figure 5. Six classes of tomato diseases. (a) Healthy; (b) Spot disease; (c) Septoria leaf spot; (d) Bacterial spot; (e) Late blight; (f) Yellow leaf curl virus disease.

Table 2. The tomato data set information.

Class	Train (Origin)	Train (Origin + Augment)	Test
Healthy	1000	2000	250
Spot disease	1000	2000	250
Septoria leaf spot	1000	2000	250
Bacterial spot	1000	2000	250
Late blight	1000	2000	250
Yellow leaf curl virus disease	1000	2000	250
Total	6000	12000	1500

3.2. Experimental Setup

All the models in the experiments are running based on the pytorch framework under Ubuntu 18.04.2 LTS, and the Python version is 3.6.9.

For RAHC_GAN, Adam is used as the optimizer, the learning rate is set to 0.0002, the output image size is 128×128 , and the training stops after 500 epochs. The training data set is Train (origin).

In all experiments, the recognition network is trained from 0, and the network weights are randomly initialized. The final fully connected layer output of the recognition network is modified to 6, corresponding to the number of classes of tomato diseases. The training is stopped after 50 epochs. In our work, for all the recognition results obtained by expanding the data using data augmentation methods, the training set is the expanded data set Train (origin+augment), and the test set (Test) remains unchanged.

The recognition accuracy Acc of the test set is used as an index to evaluate the classification effect, and the calculation process is shown in Equation (10). The original data set contains 1000 original images, and the expanded data set contains 2000 images (1000 original images, 1000 generated images). In the whole evaluation process, the test set is kept unchanged, and the data generated are only used to expand the training set:

$$Acc = \frac{\sum_{i=1}^N t_i}{\sum_{i=1}^N n_i} \times 100\% \quad (10)$$

where N represents the number of classes, t_i is the number of correct predictions for the i -th class, and n_i is the total number of samples in all classes.

3.3. RAHC_GAN Model Exploration

3.3.1. Number of Residual Blocks for Plant Image Generation

In order to explore the influence of the number of residual blocks on tomato disease recognition results, we conducted experiments on different numbers of residual blocks based on the original ACGAN, and the results are shown in Table 3. In Table 3, the first row represents the recognition results obtained after data expansion using ACGAN, and the second to fifth rows represent the results of adding 4, 8, 16, and 32 residual blocks into ACGAN, respectively.

Table 3. Results of tomato disease recognition under different numbers of residual blocks (%).

	AlexNet	VGGNet	GoogLeNet	ResNet
ACGAN	94.7	92.4	94.9	96.8
Res4	95.3	94.6	95.0	97.1
Res8	95.0	93.9	95.1	96.9
Res16	95.3	94.9	95.3	97.6
Res32	94.5	94.2	94.1	98.0

It can be seen from Table 3 that compared with the original ACGAN, the network with the residual blocks has better performance (except for the application of Res32 on AlexNet and GoogLeNet), which shows that the addition of the residual blocks is beneficial to deepening the generation network and improving the generation results. In addition, we noticed that the performance of ACGAN with 32 residual blocks is better than that of ACGAN with 16 residual blocks on the ResNet network, but the performance on other networks is not as good as the results with 16 residual blocks. Considering the performance and longer training time adding more residual blocks, in this paper, 16 residual blocks are added to the generator to deepen the depth of the generator.

3.3.2. Different Scale of Discriminators for Plant Image Generation

In order to explore the influence of the discriminator scale on tomato disease recognition results, we conducted experiments on discriminators with different scales based on the original ACGAN, and the results are shown in Table 4.

Table 4. Results of tomato disease recognition under different discriminator scales (%).

	AlexNet	VGGNet	GoogLeNet	ResNet
2D(64 128)	94.8	95.1	95.5	97.6
2D(128 256)	94.7	94.4	93.9	96.9
3D(32 64 128)	93.7	94.8	95.4	97.3
3D(64 128 256)	94.3	93.9	95.0	97.2

Table 4 shows the tomato disease recognition results of discriminators with different scales. The scale of the image generated by the generator is 128×128 , and we down-sampled and up-sampled, respectively, to obtain 64×64 and 256×256 scale images. The first two rows of Table 4 indicate that two discriminators are used to input images with scales of 64×64 and 128×128 and 128×128 and 256×256 , respectively. The results show that the performance of down-sampling is better than that of up-sampling. On this basis, we further introduced a discriminator and down-sampled the image again to obtain an image with a scale of 32×32 . We combined three discriminators and input images with scales of 32×32 , 64×64 , and 128×128 and images of 64×64 , 128×128 , and 256×256 , respectively. The results show that for the combination of three discriminators, two

down samplings of the image generated by the generator will achieve better results, but the results are still lower than that of using down sampling once; thus, we used multi-scale discriminators with scales of 64×64 and 128×128 to enrich the detailed texture of the generated images.

3.3.3. The Recognition Performance on Different Expansion Numbers of Training Set

By using the images generated by RAHC_GAN to expand the training set, the recognition accuracy of the classifier can be improved. In order to explore the optimal number of expanded images, we used RAHC_GAN to generate images and evaluated the performance of the classifier in four cases (expanding the data by 0.5 times, 1 times, 2 times, and 3 times the number of original training sets). The experimental results are shown in Figure 6.

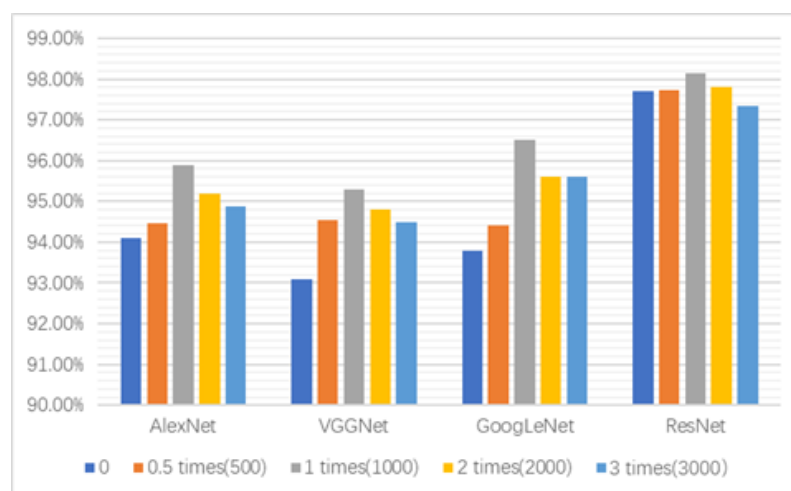


Figure 6. Results of tomato disease recognition under different expansion ratios.

From Figure 6, we know that expanding different amounts of data on the original data set can improve the performance of the classifier, and after expanding the data by 1 time, the classifier obtains the best performance. However, when the expansion ratio is increased to 2 times and 3 times the original data set, the performance of the classifier does not improve, and the overall performance shows a downward trend. It is worth noting that on ResNet, the performance of the model after the data are expanded by 3 times is lower than that of original data set, which indicates that a greater expansion of the amount of data is not better. A larger expansion of the data may “contaminate” the data set and have a negative impact on accuracy [45]. Therefore, for all the experiments in this paper, we chose to expand the data set by 1 time on the basis of the original data set as the expanded data set for further experiments.

3.3.4. The Effect of Hidden Variables

In order to illustrate the control effect of hidden variables on disease generation, we fixed the random noise z and class c and explored its influence on the generated images by changing the values of the hidden variable. Each dimension of the hidden variable h ($[h_0, \dots, h_5]$) controls a disease, and we control the generation of the disease by changing the value of a certain dimension. Since this work studies a single disease class of a single leaf image and does not consider the existence of two diseases on a single image, we change the size of a certain dimension in the range of -1 to 1 with an increasing value of 0.25 and set the values of other dimensions to 0 for the experiment. The results are shown in Figure 7. From Figure 7, we know that when the hidden variable takes different values, the area size and the severity of leaf disease in the generated images are not exactly the same. With the increase of the hidden variable, the disease area of spot, septoria leaf spot, bacterial spot, and late blight gradually increases, indicating that the disease suffered by the leaves is becoming more and more serious [46]. Moreover, when the value of

the hidden variable increases, the septoria leaf spot becomes deeper and deeper, and the leaf color difference of yellow leaf curl virus disease becomes bigger and bigger, which indicates that the hidden variable can capture the potential difference information within the class, so it can control the severity of the disease. Therefore, the proposed hidden variable can improve the intra-class difference of diseased leaves, thereby increasing the diversity of generated images.

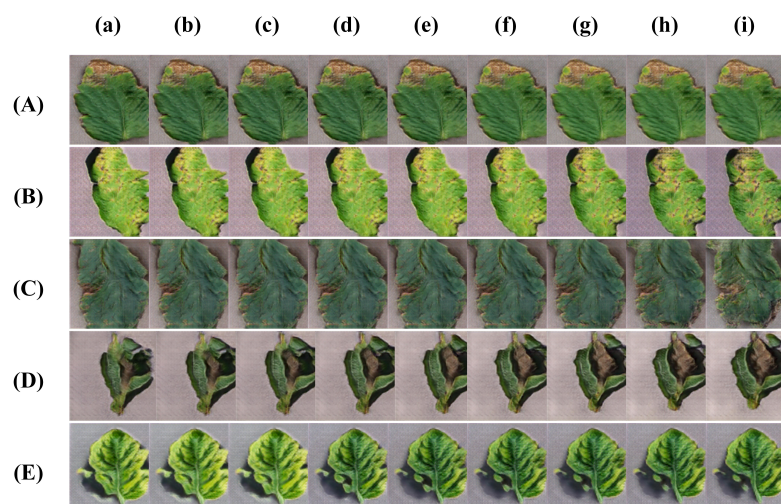


Figure 7. Images generated by different values of hidden variables. The rows represent the class of disease: (A) Spot disease, (B) Septoria leaf spot, (C) Bacterial spot, (D) Late blight, (E) Yellow leaf curl virus disease. The columns represent the values of different hidden variables: (a) -1 , (b) -0.75 , (c) -0.5 , (d) -0.25 , (e) 0 , (f) 0.25 , (g) 0.5 , (h) 0.75 , (i) 1 .

We added hidden variable loss to the full objective loss of RAHC_GAN, which is to enhance the diversity of generated images and reduce the occurrence of mode collapse. In order to explore the importance of the newly added hidden variable loss relative to the overall loss, we set the value of the hyperparameter λ (Equation (8)) ranging from 0 to 1 and the experiment with 0.2 as the step size. In addition, according to the study of Jin et al. [47], we set $\lambda = 6$ to make it equal to the number of classes. We trained RAHC_GAN with different λ values, expanded the generated data to the training set, and used it for the training of AlexNet, VGGNet, GoogLeNet, and ResNet; then, the test sets were recognized. According to the recognition accuracy, the generation quality under different λ values was judged. The results are shown in Figure 8. It can be seen from Figure 8 that when $\lambda = 1$, the accuracy of each recognition network reaches the highest, which shows that when the hidden variable loss is the same as the proportion of adversarial loss and classification loss, the disease generation ability of the network is optimal.

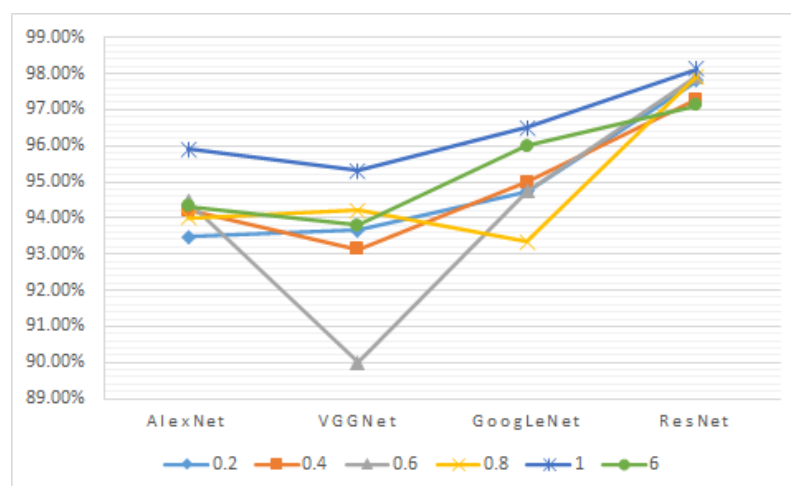


Figure 8. The recognition results of different values of λ .

3.4. The Comparative Experiment

In order to evaluate the effect of disease images generated by RAHC_GAN on recognition performance, we compared it with six other generation methods, including conventional augmentation, random erasing [9], ACGAN [27], DCGAN [25], SAGAN [48], and MSGAN [49]. We used different generation methods to generate images and expanded them to the original training set to obtain the expanded training set. Then, we performed training and testing on four mainstream recognition networks of AlexNet, VGGNet, GoogLeNet, and ResNet. The experimental results are shown in Table 5.

Table 5. The recognition accuracy of different generation methods on 4 recognition networks (%).

	AlexNet	VGGNet	GoogLeNet	ResNet
Baseline	94.1	93.1	93.8	97.7
Conventional	94.5	93.0	95.7	97.7
Random erasing	94.6	93.0	94.8	97.6
ACGAN	94.7	92.4	94.9	96.8
DCGAN	94.9	95.1	94.7	96.6
SAGAN	93.6	92.1	95.2	97.3
MSGAN	95.0	93.5	95.8	97.5
RAHC_GAN	95.9	95.3	96.5	98.1

In Table 5, the baseline indicates that no augmentation method was used to expand the data, that is, only the Train (origin) was used to train the model. It can be seen from Table 5 that the effect of various generation methods on the ResNet recognition network is in a declining state, which may be because we have used the pre-training model for training, and it is difficult to improve the performance. Since conventional augmentation only changes the location and direction of the image, less information is learned, and the improvement in recognition accuracy is limited. Random erasing erases random areas in the image, but it may erase the diseased area, which makes the recognition network learn incomplete disease features. Therefore, its accuracy is lower than that of conventional augmentation. When ACGAN, DCGAN, and MSGAN were used as the generation network to expand the data set, the performance of some recognition networks was improved. However, because the generated disease features are not obvious or there are overlaps between diseases, the performance of some recognition networks declined. On the other hand, RAHC_GAN adds a residual attention mechanism to the generator to enhance the transmission of disease information in the network, which is conducive to generating obvious disease features. Compared with the baseline, the performance of RAHC_GAN on the four classification networks improved by 1.8%, 2.2%, 2.7%, and 0.4%, respectively.

In addition, compared with other data augmentation methods, the expanded data set generated by RAHC_GAN can improve the accuracy of the four recognition networks, which has obvious advantages.

In order to intuitively compare the quality of six classes of leaf disease images generated by different generation methods, we visualized the generated disease images, as shown in Figure 9. In Figure 9, different rows represent leaf disease images generated by different methods, and different columns represent different leaf diseases. It can be seen that the images generated by DCGAN and MSGAN are fuzzy and poor in quality; the images generated by SAGAN are clearer, but the disease features are not obvious. Under the same training setting, due to the insufficient ability of the generator to learn disease features, ACGAN has difficulty generating obvious spot diseases, while the images generated by RAHC_GAN have high resolution and obvious disease features. In addition, combined with Table 5, we can see that MSGAN, which generates relatively fuzzy images, has achieved better results in the recognition of tomato leaf diseases, while SAGAN with clearer images reduced the performance of the model, because the GAN can not balance image generation and class classification well [50]. In contrast, RAHC_GAN adds a residual attention block to make the disease information of the generated images more obvious, which helps the discriminator to classify correctly. The multi-scale discriminators introduced can enrich the detail texture of the generated images, so that the generated images are clear with obvious disease features.

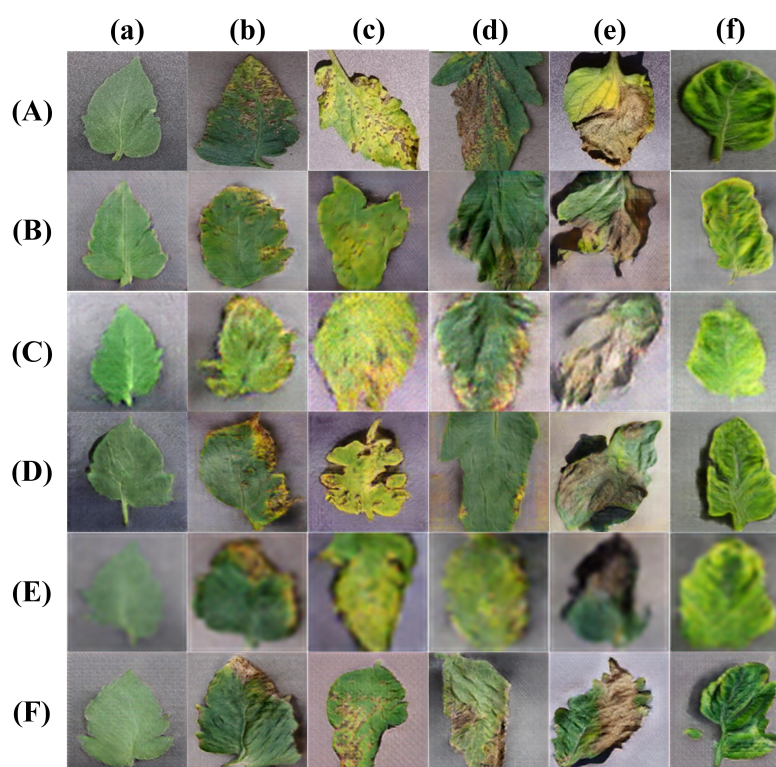


Figure 9. Diseased leaves generated by different methods: (a) Healthy, (b) Spot disease, (c) Septoria leaf spot, (d) Bacterial spot, (e) Late blight, (f) Yellow leaf curl virus disease; (A) Original data, (B) ACGAN, (C) DCGAN, (D) SAGAN, (E) MSGAN, (F) RAHC_GAN.

In order to further illustrate that the images generated by RAHC_GAN have obvious disease features, we used Train (Origin) as the training set and the generated images as the test set to conduct experiments on different network architectures. Since each class in the origin test set (Test) contains 250 images, we also generate 250 images for each class and use them as the test set. The results are shown in Figure 10. From Figure 10, we know that on different classifiers, the recognition accuracy of leaf diseases reached the highest after adding RAHC_GAN (except for the result on ResNet), indicating that

the images generated by RAHC_GAN are more similar to the original images. We noticed that the accuracy of RAHC_GAN on the ResNet classifier is lower than SAGAN, which may be because the pre-trained model used is more sensitive to the leaves generated by SAGAN. In addition, the images generated by DCGAN and MSGAN are relatively fuzzy, and the disease feature are not obvious, so they cannot be classified correctly. On the whole, compared with other generation methods, the images generated by RAHC_GAN can be recognized more accurately, indicating its effectiveness in generating leaf disease features.

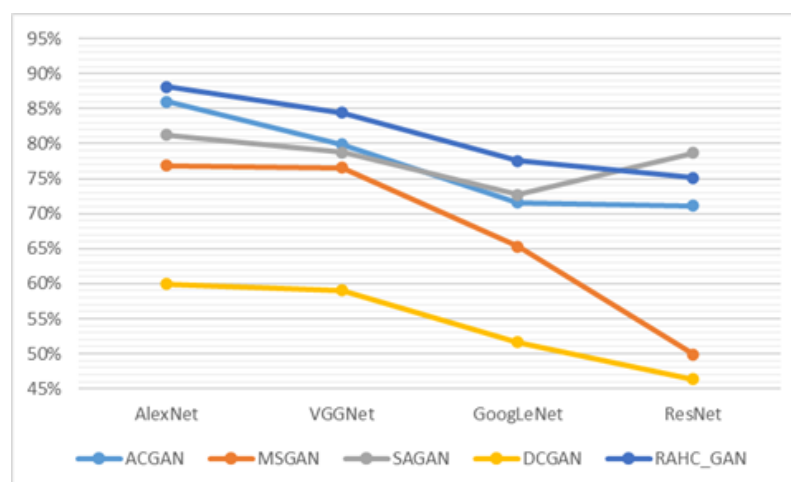


Figure 10. Test accuracy of images generated by different generation methods.

3.5. The Ablation Experiment

The data augmentation method RAHC_GAN proposed in this paper mainly adopts three improvement strategies to improve the baseline model ACGAN: the addition of residual attention blocks, the introduction of hidden variables, and the use of multi-scale discriminators. In order to verify the effectiveness of these three improvement strategies, we trained a generated adversarial network based on different network configurations and combined the generated images with the original training set as an extended training set, which is trained on four recognition networks of AlexNet, VGGNet, GoogLeNet, and ResNet, and used the test set to evaluate the performance of different networks. The results are shown in Table 6.

Table 6. Ablation experiment. R is the 16-layer residual block, A is the attention mechanism of CBAM, and H is the hidden variable (%).

	AlexNet	VGGNet	GoogLeNet	ResNet
ACGAN	94.7	92.4	94.9	96.8
+R+A	94.9	93.6	95.5	97.5
+R+H+A	95.7	94.7	96.1	97.7
RAHC_GAN	95.9	95.3	96.5	98.1

It can be seen from Table 6 that after adding the residual attention block to ACGAN, the recognition accuracy of AlexNet, VGGNet, GoogLeNet, and ResNet has been improved, indicating that the residual attention block can improve the disease generation ability of the network. On this basis, by adding the hidden variable input and hidden variable loss function, the performance of the four recognition networks has been greatly improved. This is because the hidden variable has a control effect on the generation of diseases and can supplement intra-class information in the generated images. In addition, the constraint of the hidden variable's loss increases the diversity of the generated images, so that the recognition network can learn more disease characteristics. Finally, the multi-scale discriminator strategy is applied to RAHC_GAN, and the performance is also improved on

the four recognition networks, because the multi-scale discriminator enriches the detailed texture of the generated images and can generate complete local disease information. It can be seen that the three proposed improvement strategies are beneficial to the generation of diseased images, which can improve the recognition performance of the classifier.

3.6. The Recognition Performance on Other Plants

RAHC_GAN can generate leaves with obvious disease features on tomato plants. At the same time, we hoped that it could also have a good disease generation ability on other plants to improve the performance of the recognition network. In order to verify whether RAHC_GAN has a good ability to generate other plants, we chose apple, grape, and corn leaves in the Plant Village data set. The apple data include four diseases (black spot, black rot, cedar-apple rust, and healthy), the grape data include four diseases (black rot, wheel spot, brown spot, and healthy), and the corn data include four diseases (gray leaf spot, common rust, healthy, and northern leaf blight), as shown in Figure 11. We use the same method for image generation, data expansion, and disease identification on these three plants. The specific information of the data set and the experimental results of RAHC_GAN on these three types of data sets are shown in Tables 7 and 8, respectively.

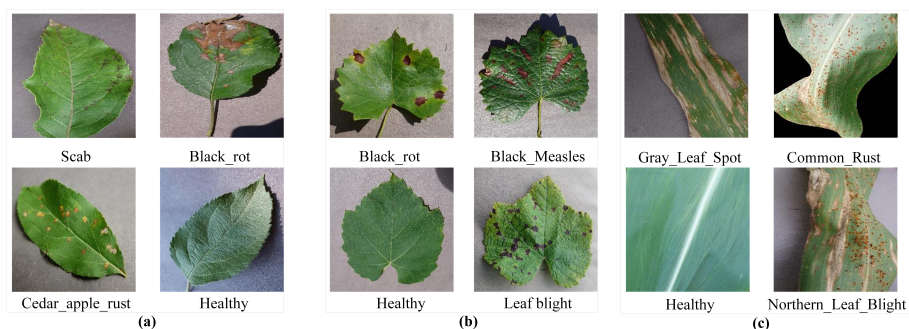


Figure 11. Sample display of apple (a), grape (b), and corn (c).

Table 7. The data set display of apple, grape, and corn.

Apple			Grape			Corn		
Class	Train	Test	Class	Train	Test	Class	Train	Test
Scab	500	125	Black_rot	500	125	Gray_leaf_spot	200	50
Black_rot	500	125	Black_Measles	1000	250	Common_rust	500	125
Cedar_apple_rust	200	50	Healthy	200	50	Healthy	500	125
Healthy	1000	250	Leaf_blight	500	125	Northern_leaf_blight	500	125

Table 8. The recognition accuracy on apple, grape, and corn data(%). a,b,c represent three kinds of recognition networks, AlexNet, VGGNet, and GoogLeNet, respectively.

	Apple			Grape			Corn		
	a	b	c	a	b	c	a	b	c
baseline	97.5	97.5	99.5	98.7	98.4	99.5	96.7	96.9	98.4
ACGAN	97.8	97.8	99.6	97.1	98.7	99.3	97.7	97.4	98.6
RAHC_GAN	98.0	99.1	99.8	99.1	98.7	99.8	98.1	98.1	98.8

From Table 8, we can see that after using RAHC_GAN for data augmentation, the performance of the three recognition models on apple, grape, and corn data has been improved in varying degrees, proving that the RAHC_GAN method has a generalization ability on different data sets. It is worth noting that for apple and corn data, using ACGAN for expansion can improve the recognition performance of the classifier. However, for grape data, on the AlexNet and GoogLeNet recognition networks, using ACGAN for expansion

will reduce the recognition performance. This is because the four classes of apple and corn disease data are quite different in shape and lesion (Figure 11a,c), while the shapes of the four grape disease data sets (Figure 11b) are very similar, with small intra-class differences, and the difference of the four diseases is only in the disease spots. In this case, ACGAN cannot accurately focus on the lesion area, which leads to an indistinct lesion in the generated images and even reduces the recognition performance of the model. In contrast, because RAHC_GAN adds the residual attention module, it can accurately pay attention to and learn the disease information in the image. Furthermore, the multi-scale discriminator can enrich the detailed texture of the disease spot area and generate the images with obvious disease features. Therefore, in grape data, RAHC_GAN can improve the recognition performance of the classifier.

4. Discussion

Deep learning training requires a large amount of data. When the data are insufficient, a variety of methods can be used for data augmentation, such as flipping, cropping, etc. Instead of this method, GAN does not require the manual selection of transformation operations, but through the confrontation of generator and discriminator, it simulates the distribution of the original data and generates a group of data. Experimental results show that the extended data generated by GAN can improve the recognition performance of plant diseases. However, because the variability of the image generated by GAN is limited, it is impossible to capture all the possible changes in the image, so the performance improvement is limited. At the same time, the images generated by GAN have limitations. Although adding data can improve the recognition performance, adding too much data in the training set may destroy the original data set and cause performance degradation. Therefore, in future work, we will continue to explore more effective data augmentation methods to generate higher-quality images while maintaining better recognition performance.

In addition, because GAN is a kind of neural network, its training also depends on data. When the amount of data in the training set is too small, the image of the specified class cannot be generated accurately by noise. Therefore, we will also try to use image to image conversion generation instead of noise to image conversion for data augmentation.

5. Conclusions

In the training of a recognition network, there is a strong symmetric relationship between sufficient and insufficient training samples. In order to solve the problem of insufficient data, we proposed a generative adversarial network model for tomato disease leaf generation, called RAHC_GAN, and evaluated it on four disease identification networks: AlexNet, VGGNet, GoogLeNet, and ResNet. The experimental results show that the expanded data set can meet the large amount of data required for neural network training, which makes the data set expanded by RAHC_GAN have a greater improvement in recognition accuracy under different classification models. At the same time, we found that under the same experimental settings, VGGNet, as the classification network with the most network parameters, has the lowest recognition performance among the four networks without data augmentation, and the performance improvement is the most obvious after using RAHC_GAN for data augmentation, which shows that on the neural network with large parameters, sufficient data can significantly improve its recognition performance.

In comparison with other data augmentation methods, the data set expanded by RAHC_GAN has better performance on the four classification networks, indicating that RAHC_GAN can generate tomato diseased leaf images with obvious disease features from random noise and can effectively solve the problem of insufficient training data. The results of ablation experiments show that in the generation of leaves with the same disease, the newly added hidden variable can learn the potential intra-class change information, so as to enrich the class information of the same disease. The addition of residual attention blocks can make the generator pay attention to the disease area in the image and guide

it to generate leaves with obvious disease features. The introduction of a multi-scale discriminator can enrich the detailed texture of the generated images.

The application of RAHC_GAN on apple, grape, and corn data from the PlantVillage data set shows that the proposed method has good generalization ability. In addition to solving the problem of insufficient data for tomato leaf identification, it can also be used for other plant research tasks lacking diseased leaf data.

Author Contributions: Conceptualization, H.D., D.L., Z.C., H.L. and X.Y.; methodology, H.D. and D.L.; validation, D.L.; formal analysis, H.D. and D.L.; investigation, Z.C.; data curation, D.L.; writing—original draft preparation, D.L.; writing—review and editing, H.D. and D.L.; visualization, D.L. and Z.C.; supervision, H.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China grant number 61976150; the Natural Science Foundation of Shanxi Province, China grant number 201801D121135, 201901D111091; and Key R&D projects in Jinzhong City, China grant number Y192006.

Data Availability Statement: The PlantVillage data set is available at the following link: <https://github.com/spMohanty/PlantVillage-Dataset/tree/master/raw>.

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

References

- Giovannucci, E. Tomatoes, Tomato-Based Products, Lycopene, and Cancer: Review of the Epidemiologic Literature. *J. Natl. Cancer Inst.* **1999**, *91*, 317–331. [\[CrossRef\]](#)
- Li, N.; Wu, X.; Zhuang, W.; Xia, L.; Zhou, Y. Tomato and lycopene and multiple health outcomes: Umbrella review. *Food Chem.* **2020**, *343*, 128396. [\[CrossRef\]](#)
- Mazidi, M.; Ferns, G.A.; Banach, M. A high consumption of tomato and lycopene is associated with a lower risk of cancer mortality: Results from a multi-ethnic cohort. *Public Health Nutr.* **2020**, *23*, 1–7. [\[CrossRef\]](#)
- Shijie, J.; Peiyi, J.; Siping, H.; Haibo, S. Automatic detection of tomato diseases and pests based on leaf images. In Proceedings of the 2017 Chinese Automation Congress (CAC), Jinan, China, 20–27 October 2017; pp. 2510–2537. [\[CrossRef\]](#)
- Liu, J.; Wang, X. Correction to: Early recognition of tomato gray leaf spot disease based on MobileNetv2-YOLOv3 model. *Plant Methods* **2021**, *17*, 19. [\[CrossRef\]](#)
- Mohanty, S.P.; Hughes, D.P.; Salathe, M. Using Deep Learning for Image-Based Plant Disease Detection. *Front. Plant Sci.* **2016**, *7*, 1419. [\[CrossRef\]](#)
- Khandelwal, I.; Raman, S. Analysis of Transfer and Residual Learning for Detecting Plant Diseases Using Images of Leaves. In *Computational Intelligence: Theories, Applications and Future Directions*; Springer: Singapore, 2019; Volume II, pp. 295–306. [\[CrossRef\]](#)
- Ma, A.; As, B.; Sa, C.; As, D.; Sg, A. ToLeD: Tomato Leaf Disease Detection using Convolution Neural Network. *Procedia Comput. Sci.* **2020**, *167*, 293–301. [\[CrossRef\]](#)
- Zhong, Z.; Zheng, L.; Kang, G.; Li, S.; Yang, Y. Random Erasing Data Augmentation. *Proc. AAAI Conf. Artif. Intell.* **2017**, *34*, 13001–13008. [\[CrossRef\]](#)
- Takahashi, R.; Matsubara, T.; Uehara, K. Data Augmentation Using Random Image Cropping and Patching for Deep CNNs. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *30*, 2917–2931. [\[CrossRef\]](#)
- Zhang, H.; Cisse, M.; Dauphin, Y.; Lopez-Paz, D. mixup: Beyond Empirical Risk Minimization. *arXiv* **2017**, arXiv:1710.09412.
- Chou, H.P.; Chang, S.C.; Pan, J.Y.; Wei, W.; Juan, D.C. Remix: Rebalanced Mixup. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020. [\[CrossRef\]](#)
- Bin, L.; Zhang, Y.; He, D.; Li, Y. Identification of Apple Leaf Diseases Based on Deep Convolutional Neural Networks. *Symmetry* **2017**, *10*, 11. [\[CrossRef\]](#)
- Mohamad, A.H.A.; Ismail, N.; Ahmad, I.M.Y.; Nasir Taib, M. VGG16 for Plant Image Classification with Transfer Learning and Data Augmentation. *Int. J. Eng. Technol.* **2018**, *7*, 90–94. [\[CrossRef\]](#)
- Shi, Y.; Qin, T.; Liu, Y.; Lu, J.; Shen, D. Automatic Data Augmentation by Learning the Deterministic Policy. *arXiv* **2019**, arXiv:1910.08343.
- DeVries, T.; Taylor, G.W. Dataset Augmentation in Feature Space. *arXiv* **2017**, arXiv:1702.05538.
- Cubuk, E.D.; Zoph, B.; Mane, D.; Vasudevan, V.; Le, Q.V. AutoAugment: Learning Augmentation Strategies From Data. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 113–123. [\[CrossRef\]](#)
- Ho, D.; Liang, E.; Stoica, I.; Abbeel, P.; Chen, X. Population Based Augmentation: Efficient Learning of Augmentation Policy Schedules. In Proceedings of the 36th International Conference on Machine Learning (ICML), Long Beach, CA, USA, 9–15 June 2019; pp. 4843–4856.

19. Goodfellow, I.J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Networks. *Commun. ACM* **2014**, *63*, 139–144. [\[CrossRef\]](#)
20. Bowles, C.; Chen, L.; Guerrero, R.; Bentley, P.; Gunn, R.; Hammers, A.; Dickie, D.A.; Hernández, M.; Wardlaw, J.; Rueckert, D. GAN Augmentation: Augmenting Training Data using Generative Adversarial Networks. *arXiv* **2018**, arXiv:1810.10863.
21. Shrivastava, A.; Pfister, T.; Tuzel, O.; Susskind, J.; Webb, R. Learning from Simulated and Unsupervised Images through Adversarial Training. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2242–2251. [\[CrossRef\]](#)
22. Tran, N.T.; Tran, V.H.; Nguyen, N.B.; Nguyen, T.K.; Cheung, N.M. On Data Augmentation for GAN Training. *IEEE Trans. Image Process.* **2021**, *30*, 1882–1897. [\[CrossRef\]](#)
23. Niu, S.; Li, B.; Wang, X.; Lin, H. Defect Image Sample Generation With GAN for Improving Defect Recognition. *IEEE Trans. Autom. Sci. Eng.* **2020**, *17*, 1611–1622. [\[CrossRef\]](#)
24. Douarre, C.; Crispim-Junior, C.F.; Gelibert, A.; Tougne, L.; Rousseau, D. Novel data augmentation strategies to boost supervised segmentation of plant disease. *Comput. Electron. Agric.* **2019**, *165*, 104967. [\[CrossRef\]](#)
25. Radford, A.; Metz, L.; Chintala, S. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *arXiv* **2015**, arXiv:1511.06434.
26. Purbaya, M.E.; Setiawan, N.A.; Adji, T.B. Leaves image synthesis using generative adversarial networks with regularization improvement. In Proceedings of the International Conference on Information and Communications Technology, Yogyakarta, Indonesia, 6–7 March 2018; pp. 360–365. [\[CrossRef\]](#)
27. Odena, A.; Olah, C.; Shlens, J.B. Conditional image synthesis with auxiliary classifier GANs. In Proceedings of the 2017 International Conference on Machine Learning (ICML), International Convention Centre, Sydney, Australia, 11–15 August 2017; pp. 4043–4055.
28. Tang, X.L.; Du, Y.M.; Liu, Y.W.; Li, J.X.; Ma, Y.W. Image Recognition With Conditional Deep Convolutional Generative Adversarial Networks. *Zidonghua Xuebao/Acta Autom. Sin.* **2018**, *44*, 855–864. [\[CrossRef\]](#)
29. Karras, T.; Aila, T.; Laine, S.; Lehtinen, J. Progressive Growing of GANs for Improved Quality, Stability, and Variation. *arXiv* **2017**, arXiv:1710.10196.
30. Zhao, Y.; Chen, Z.; Gao, X.; Song, W.; Zhang, Z. Plant Disease Detection using Generated Leaves Based on DoubleGAN. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2021**,. [\[CrossRef\]](#)
31. Arsenovic, M.; Karanovic, M.; Sladojevic, S.; Anderla, A.; Stefanovic, D. Solving Current Limitations of Deep Learning Based Approaches for Plant Disease Detection. *Symmetry* **2019**, *11*, 939. [\[CrossRef\]](#)
32. Qin, X.H.; Wang, Z.; Jiepeng, Y.; Zhou, Q.; Zhao, P.; Wang, Z.y.; Huang, L. Using a one-dimensional convolutional neural network with a conditional generative adversarial network to classify plant electrical signals. *Comput. Electron. Agric.* **2020**, *174*, 105464. [\[CrossRef\]](#)
33. Tian, Y.; Yang, G.; Wang, Z.; Li, E.; Liang, Z. Detection of Apple Lesions in Orchards Based on Deep Learning Methods of CycleGAN and YOLOV3-Dense. *J. Sens.* **2019**, *2019*, 1–13. [\[CrossRef\]](#)
34. Szegedy, C.; Wei, L.; Jia, Y.; Sermanet, P.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 1–9. [\[CrossRef\]](#)
35. Zhu, F.; He, M.; Zheng, Z. Data augmentation using improved cDCGAN for plant vigor rating. *Comput. Electron. Agric.* **2020**, *175*, 105603. [\[CrossRef\]](#)
36. Liu, B.; Tan, C.; Li, S.; He, J.; Wang, H. A Data Augmentation Method Based on Generative Adversarial Networks for Grape Leaf Disease Identification. *IEEE Access* **2020**, *8*, 102188–102198. [\[CrossRef\]](#)
37. Arjovsky, M.; Bottou, L. Towards Principled Methods for Training Generative Adversarial Networks. *arXiv* **2017**, arXiv:1701.04862.
38. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–198. [\[CrossRef\]](#)
39. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [\[CrossRef\]](#)
40. Chen, X.; Duan, Y.; Houthoofd, R.; Schulman, J.; Sutskever, I.; Abbeel, P. InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets. In Proceedings of the 30th International Conference on Neural Information Processing Systems, NIPS'16, Barcelona, Spain, 5 December 2016; Volume 29, pp. 2180–2188.
41. Guo, Y.; Li, H.; Zhuang, P. Underwater Image Enhancement Using a Multiscale Dense Generative Adversarial Network. *IEEE J. Ocean. Eng.* **2020**, *45*, 862–870. [\[CrossRef\]](#)
42. Krizhevsky, A.; Sutskever, I.; Hinton, G. ImageNet Classification with Deep Convolutional Neural Networks. *Neural Inf. Process. Syst.* **2012**, *25*. [\[CrossRef\]](#)
43. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *Comput. Sci.* **2014**.
44. Brahimi, M.; Arsenovic, M.; Laraba, S.; Sladojevic, S.; Boukhalfa, K.; Moussaoui, A. Deep Learning for Plant Diseases: Detection and Saliency Map Visualisation. In *Human and Machine Learning*; Zhou, J., Chen, F., Eds.; Springer International Publishing: Cham, Switzerland, 2018; pp. 93–117, ISBN 978-3-319-90402-3. [\[CrossRef\]](#)
45. Jha, G.; Cecotti, H. Data augmentation for handwritten digit recognition using generative adversarial networks. *Multimed. Tools Appl.* **2020**, *79*, 35055–35068. [\[CrossRef\]](#)

-
46. Rao, S.; Danish, S.; Keflemariam, S.; Tesfagergish, H.; Tesfamariam, R.; Habtemariam, T. Pathological Survey on Disease Incidence and Severity of Major Diseases on Tomato and Chilli Crops Grown in Sub Zoba Hamelmalo, Eritrea. *Int. J. Res. Stud. Agric. Sci.* **2016**, *2*, 20–31.
 47. Jin, Y.; Zhang, J.; Li, M.; Tian, Y.; Zhu, H.; Fang, Z. Towards the Automatic Anime Characters Creation with Generative Adversarial Networks. *arXiv* **2017**, arXiv:1708.05509.
 48. Zhang, H.; Goodfellow, I.; Metaxas, D.; Odena, A. Self-Attention Generative Adversarial Networks. *PMLR* **2018**, *97*, 7354–7363.
 49. Mao, Q.; Lee, H.Y.; Tseng, H.Y.; Ma, S.; Yang, M.H. Mode Seeking Generative Adversarial Networks for Diverse Image Synthesis. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 12744–12753. [[CrossRef](#)]
 50. Li, C.; Xu, K.; Zhu, J.; Zhang, B. Triple Generative Adversarial Nets. *arXiv* **2017**, arXiv:1703.02291.