

Article

Using a Hybrid Convolutional Neural Network with a Transformer Model for Tomato Leaf Disease Detection

Zhichao Chen, Guoqiang Wang *, Tao Lv and Xu Zhang

Electronic Engineering College, Heilongjiang University, Harbin 150080, China; 2211802@s.hlju.edu.cn (Z.C.); 2211800@s.hlju.edu.cn (T.L.); 2211797@s.hlju.edu.cn (X.Z.)

* Correspondence: wangguoqiang@hlju.edu.cn

Abstract: Diseases of tomato leaves can seriously damage crop yield and financial rewards. The timely and accurate detection of tomato diseases is a major challenge in agriculture. Hence, the early and accurate diagnosis of tomato diseases is crucial. The emergence of deep learning has dramatically helped in plant disease detection. However, the accuracy of deep learning models largely depends on the quantity and quality of training data. To solve the inter-class imbalance problem and improve the generalization ability of the classification model, this paper proposes a cycle-consistent generative-adversarial-network-based Transformer model to generate diseased tomato leaf images for data augmentation. In addition, this paper uses a Transformer model and densely connected CNN architecture to extract multilevel local features. The Transformer module is utilized to capture global dependencies and contextual information accurately to expand the sensory field of the model. Experiments show that the proposed model achieved 99.45% accuracy on the PlantVillage dataset. The 2018 Artificial Intelligence Challenger dataset and the private dataset attained accuracies of 98.30% and 95.4%, and the proposed classification model achieved a higher accuracy and smaller model size compared to previous deep learning models. The classification model is generalizable and robust and can provide a stable theoretical framework for crop disease prevention and control.

Keywords: Transformer; deep learning; tomato disease; cycle-consistent generative adversarial network



Citation: Chen, Z.; Wang, G.; Lv, T.; Zhang, X. Using a Hybrid Convolutional Neural Network with a Transformer Model for Tomato Leaf Disease Detection. *Agronomy* **2024**, *14*, 673. <https://doi.org/10.3390/agronomy14040673>

Academic Editor: Gniewko Niedbała

Received: 14 January 2024

Revised: 8 February 2024

Accepted: 23 February 2024

Published: 26 March 2024



Copyright: © 2024 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

As a major cash crop for global agricultural development, tomato production has increased in recent years according to statistics and surveys. However, due to various factors, such as climatic variations and human-made operations, tomatoes are susceptible to lesions that affect their yield and economy [1]. Traditional detection methods require agricultural workers to detect the lesions via eye identification using specialized knowledge of the lesion area. As computer vision technology continues to be developed, deep learning has been gradually applied to the field of crops due to its ease of operation and high accuracy. The application of deep learning technology for disease identification has been widely used, and a sufficient number of crop-diseased leaves is a prerequisite to ensure good performance in disease identification [2].

Training a deep learning model requires large amounts of data for support. However, there are often insufficient agricultural crop disease images and they are difficult to obtain. Moreover, the designed deep learning classification model sometimes suffers from overfitting problems and generalization problems due to the imbalance of class distribution among datasets. Data augmentation [3] can effectively enhance model generalizability. Traditional augmentation typically involves spatial transformations of existing images, like rotation, skew, and color space modifications. However, traditional image enhancement cannot fully enrich the spatial hierarchy and distribution. In recent years, a generative adversarial network [4] has overcome the limitations of traditional augmentation, which aimed to generate images by matching real disease characteristics as closely as possible.

Using unsupervised image-to-image translation, generative adversarial networks (GANs) can transform healthy crop leaf images in the dataset to synthetic images exhibiting tomato disease features. It is a data enhancement technique for small-sample disease classification and identification. Convolutional neural networks (CNNs) are the most common deep learning architectures, with models like VGG16 [5], ResNet [6], and MobileNet [7] being widely used in medical treatment, biology, and agriculture for target detection and recognition. As artificial intelligence has improved, the emergence of the Transformer model has sparked great interest in computer vision [8]. CNNs utilize convolutional kernels sliding on the input image to capture the local features of the image. In the opposite direction, the Transformer utilizes a self-attention mechanism to model a global context and extract global features from images. Its unique structure captures contextual dependencies through global modeling across the image. Researchers have used the Transformer model in combination with a CNN model to extract local features using CNN and model global dependencies using the Transformer's self-attention mechanism, achieving good results in tasks such as image classification.

As science and technology continue to advance, deep learning techniques for image recognition and detection have become essential tools in the field of agricultural disease prevention [9]. Some researchers have designed various deep learning models to recognize crop diseases. The emergence of generative adversarial networks (GANs) has shown that they have better performance in image augmentation, attracting experts' attention and rapidly advancing crop science development. Liu et al. [10] proposed the Leaf GAN model to generate new images of grapevine leaf disease. Moreover, the Xception model was utilized to identify grape leaf diseases, and the results showed that the method achieved an average identification accuracy of 98.70%. Douarre et al. [11] used C-GAN to augment tomato disease leaf images and proposed an improved DenseNet121 model, achieving higher recognition accuracy. Zhu et al. [12] also proposed a C-GAN-based data augmentation method using conditional information to constrain synthesized image categories. Dense connections were introduced to the generator and discriminator to enhance feature propagation, significantly improving classification performance. Tian et al. [13] adopted a CycleGAN for image-to-image translation to generate apple sample images and combined it with YOLO for apple disease detection. The GAN model transformed healthy apple images into diseased ones, achieving a detection accuracy of 95.57%. Quan Huu Cap et al. [14] proposed the Leaf GAN model with self-attention mechanisms. The model only translates relevant regions from images with various backgrounds and enhances the diversity of training images. In comparison, LeafGAN improved the diagnostic performance by 7.4% over baseline models. Abbas et al. [15] developed a deep learning model for detecting tomato leaves. They synthesized tomato leaf images using a conditional GAN, and then trained a DenseNet121 model on both synthetic and real images through transfer learning. They trained a DenseNet121 model on synthetic and real images via transfer learning. The model achieved 97.11% classification accuracy in the public PlantVillage tomato leaf dataset.

Sagar and Dheeba [16] compared several deep learning methods, including VGG16, ResNet50, Inception-V3, and Inception-Resnet. They utilized transfer learning to improve model generalization effectively. On the validation set, they found that the ResNet50 model achieved the best accuracy of 98.2%. Widiyanto et al. [17] designed a CNN model for identifying four tomato disease classes, training the model on 1000 images per class from the dataset. The model acquired 96.6% accuracy in recognizing five classes. Albogamy [18] developed a model composed of convolutional, batch normalization, max pooling, and dropout layers, along with an automated assisted diagnostic system. Experiments were carried out on a public dataset of 15 classes of diseases. The model achieved 96.4% accuracy on the validation set. Zhao et al. [19] introduced attention mechanisms into deep convolutional neural network models. Their network consisted of residual and attention modules. The trained model extracted complex features of different diseases to classify tomato pathologies better. Through multiple experiments, their model achieved an average accuracy of 98.81% on tomato datasets. Agarwal et al. developed a CNN model with three

convolutional layers, three max-pooling layers, and two fully connected layers to classify ten categories of tomato plants (nine diseased plants and one healthy plant). Their model achieved 91.2% classification accuracy on the PlantVillage dataset. Elhassouny and Smarandache [20] designed an intelligent mobile application utilizing the MobileNet network model to detect nine tomato diseases. Their model achieved an accuracy of 90.3%. Ahmed et al. [21] proposed a tomato leaf disease detection method based on lightweight transfer learning, where a network composed of the trained MobileNetV2 architecture and classifier extracts features to effectively detect disease regions, achieving an accuracy of 99.30% with a model size of 9.60 MB. Yang et al. [22] proposed a new three-branch Swin Transformer classification network for disease classification. Under a multi-task classification strategy, three-branch networks built on the Swin Transformer backbone extract preliminary features to improve learning ability for feature extraction, achieving an overall disease classification accuracy of 99.00%. Li et al. [23] proposed a lightweight apple leaf disease identification model based on Vision Transformer to extract real features of crop disease. An improved patch embedding approach retained more edge information, promoting information exchange between patches in the Transformer. Depthwise separable convolutions and linear complexity multi-head attention substantially reduced the model parameters and FLOPs. When trained on an apple leaf disease dataset with a complex background, ConvViT achieved a recognition accuracy of 96.85%.

To address the problems of an insufficient sample size of tomato disease leaf data, poor sample diversity, and imbalance between classes, an improved model based on a cycle-consistent generative adversarial network is proposed to replace the generator in the original model with a Transformer and a convolutional neural network, which effectively captures the feature information of tomato diseases. The Transformer, as a generator, can learn the global style information of tomato diseases and migrate it to standard tomato images, thus realizing style migration and generating new disease samples. The method expands the scale of the tomato disease dataset, and the diversity and quality of the generated samples are also improved, which provides rich training samples for the subsequent disease classification model and uses the original dataset and the newly synthesized dataset as the training sets for the subsequent classification task. In addition, this study also proposes a tomato disease classification model integrating a Transformer and a CNN. The model utilizes a convolutional neural network to extract local features of the diseases. At the same time, the Transformer module analyzes the whole image and learns the global distribution features of the diseases. The Transformer's self-attention mechanism can model the positional information of diseases within an image as well as the correlations between different regions, enabling the model to understand global disease patterns better. Compared to using either a CNN or Transformer alone, the hybrid model combines the advantages of both, jointly modeling local details and global structures to increase the model's feature characterization capabilities. Moreover, the proposed classification model has smaller parameters and computational loads, making it easier to deploy on mobile or embedded devices. The performance of the proposed model has been evaluated on the PlantVillage dataset [24], 2018 AI Challenger dataset, and self-built tomato dataset, achieving accuracies of 99.45%, 98.63%, and 98.45%, respectively. Using the CyTrGAN augmented dataset further improved the accuracy to 99.60%, 98.91%, and 96.40%.

In this study, we make the following contributions:

1. The categories in the three datasets, which are unbalanced and have inadequate sample amounts, are used to augment the tomato dataset by synthesizing new tomato disease images based on an improved cycleGAN network (CyTrGan) to improve the robustness and generalization of the subsequent classification models.
2. A hybrid model with high classification accuracy and small parameters composed of a Transformer and densely connected networks is proposed.
3. This paper compares the recognition performance of the proposed Dual Vision Transformer model (DVT) with seven classical models and recent works in the literature on the tomato disease image classification task. The results show that the proposed clas-

sification model can achieve better classification performance. In addition, through visualization experiments, we also found that the model can better learn the detailed features of tomato disease images.

2. Materials and Methods

2.1. CyTrGan as Data Augmentation

To relieve overfitting and increase the generalization capability and robustness of the classification model, this study employs a GAN for data augmentation by generating new synthetic images to expand the dataset scale rather than simply applying conventional data augmentation techniques, like flipping and random noise addition, to existing images. Compared to simple traditional data augmentation, the new samples produced by GAN are more prosperous and more diverse, which helps enhance the model's adaptability to different data distributions and strengthen model generalization. The GAN comprises two models: a generator and a discriminator. The generator produces synthetic pictures, while the discriminator differentiates between real and synthetic pictures. The generator aims to fool the discriminator with the synthesized images, whereas the discriminator tries to ensure the generated images are as close to the real ones. The method proposed in this paper is based on CycleGAN [25], using a combination of a Transformer and CNN as the generator to generate tomato disease images [26]. The overall framework is illustrated in Figure 1, comprising a pair of generator and discriminator networks.

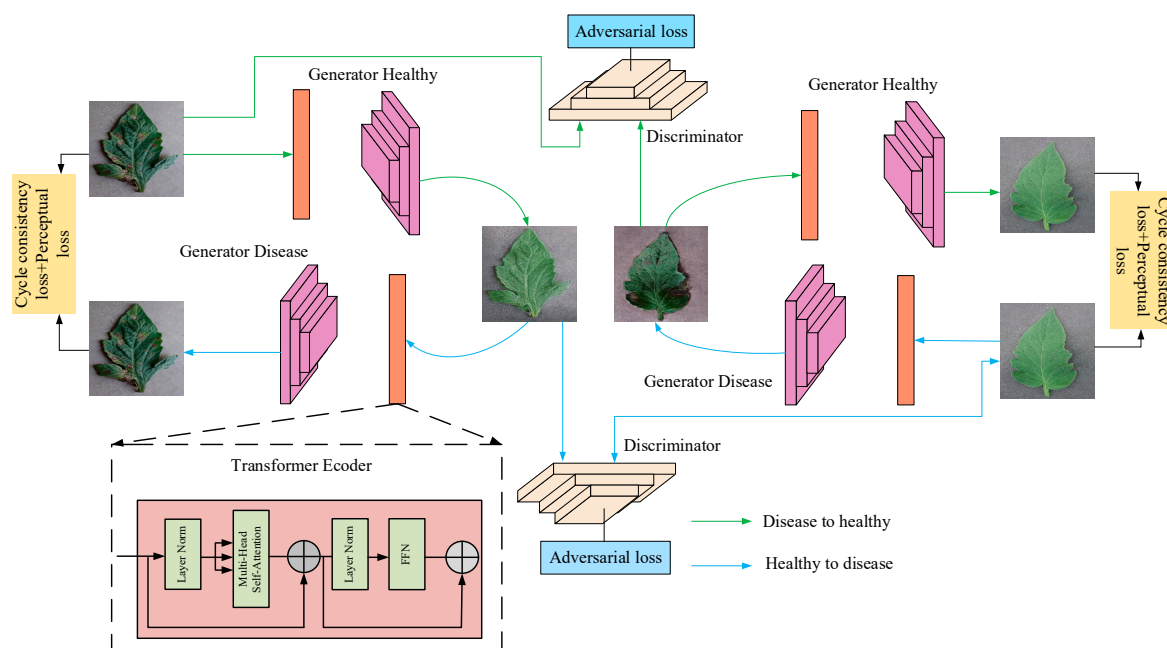


Figure 1. The framework of CyTrGAN.

The CyTrGan generator consists of a visual Transformer encoder and three CNN upsampling layers, which divide an input-sized image into patches of sizes to reduce computation and improve efficiency in image processing effectively. The patches are linearly projected into a high-dimensional feature space to extract patch features. Position encodings are introduced to the patch features to provide positional information for subsequent self-attention layers. Each encoder layer contains a feedforward fully connected network and a multi-head self-attention module. The self-attention module can model global dependencies between images. The encoder's output is three convolutional blocks with progressive upsampling. Each upsampling block consists of convolution layers, normalization layers, and ReLU activation functions. The kernel size, stride, and padding for all convolution layers are set to 3, 2, and 1, respectively. Upsampling is achieved via transposed convolutions to recover the original RGB image gradually.

The discriminator network adopts a convolutional neural network architecture to discriminate between real and synthetic images. It contains four main components: The first is the initial convolution block where the input images pass through a convolutional layer with a kernel size of 4×4 , stride of 2, and padding of 1, producing 64 output channels to extract initial low-level features, with LeakyReLU as the activation function. The second is three downsampling modules, each containing a set of conv2d layers and downsampling operations. The first module has conv2d kernels with a size of 4×4 , stride of 1, and input/output channels of 64/128. The second module downsamples with a stride of 2 and input/output channels of 128/256. The third module also downsamples, with input/output channels of 256/512. Finally, there is a conv2d layer with a kernel size of 4×4 , a stride of 1, a padding of 1, an input channel of 512, and outputs a single channel that passes through sigmoid activations to produce a probability output in the range of 0–1. During forward propagation, real images undergo data augmentation to prevent discriminator overfitting. The discriminator then extracts multi-scale features through the above modules and outputs the probability that the input is a real image.

The generator synthesizes images containing tomato disease characteristics and healthy characteristics into images with disease features imposed on a healthy background and images with healthy features imposed on a diseased background. The discriminator network extracts multi-level features through convolutions and captures global information by downsampling to distinguish the generated health appearance image from the real health image and calculates the probability that the image is real. Therefore, the process can be seen as a cyclic adversarial process.

Loss Functions

- (1) Adversarial loss: Adversarial loss allows the generated disease pictures to match the distribution of the target area as much as possible and could enhance the quality of the disease pictures generated. For $G_1: X \rightarrow Y$, the generator G_1 expects the generated image $G_1(x)$ to be as similar as possible to the real data, y , in the y -domain. The discriminator D needs to decide whether the input image of a disease is a real image or a synthetic image of disease, and the goal of the generator G_1 is to minimize L_{adv} , and that of the discriminator is to maximize it. Both of them are playing games with each other; similarly, $G_2: Y \rightarrow X$ is opposite to the above.

$$L_{GAN}(G, D_Y, X, Y) = E_{y \sim p_{data}(y)} [\log(D_Y(y))] + E_{x \sim p_{data}(x)} [\log(1 - D_Y(G(x)))] \quad (1)$$

$$L_{GAN}(F, D_X, Y, X) = E_{x \sim p_{data}(x)} [\log D_X(X)] + E_{y \sim p_{data}(y)} [\log(1 - D_X(F(y)))] \quad (2)$$

- (2) Cycle-consistent loss: The cycle-consistency loss enables the generator to learn the distribution characteristics of diseased images. However, when an image is mapped from the source domain (X) to the target domain (Y) and back to the source, it should preserve similarity with the original image. The same applies when mapping between two diseased images. The cycle consistency loss ensures the cyclic transformations could recover the image to its original state. It acts as a regularization, with the regularization strength being controlled by the coefficient.

$$L_{cyc}(G, F) = E_{x \sim p_{data}(x)} [\|F(G(x)) - x\|_1] + E_{y \sim p_{data}(y)} [\|G(F(y)) - y\|_1] \quad (3)$$

- (3) Perceptual loss [27]: Perceptual loss has been widely applied as an effective loss in image generation tasks, such as super-resolution and style transfer. It acts as a loss function that can approximate human image similarity evaluation. It compares two different but similar-looking images and extracts high-level semantic features. Perceptual loss utilizes a pre-trained VGG-16 network on the Imagenet dataset to extract texture and structural information, while the fully connected layers extract semantic information. This study regards perceptual loss as a supplement to cycle consistency loss to improve the quality of the generated diseased images. It is typically calculated

by passing the input and target images through a pre-trained neural network to obtain their feature representations. These are input to the loss function to compute the Euclidean or Manhattan distance between them.

$$L_{Perceptual}(G, F) = \|\phi(x) - \phi(F(G(x)))\|_2^2 + \|\phi(y) - \phi(G(F(y)))\|_2^2 \quad (4)$$

where x is the input image and y is the target image.

(4) Total loss:

$$L_{cyclegan}(G, F, D_x, D_y) = L_{GAN}(G, D_y, X, Y) + L_{GAN}(F, D_x, X, Y) + \lambda L_{cyc}(G, F) + \mu L_{perceptual}(G, F) \quad (5)$$

2.2. Tomato Disease Classification with Dual Vision Transformer Model

This study aims to establish a hybrid model with high classification accuracy and small parameters composed of a Transformer and densely connected networks. A CNN model has significant advantages in extracting local features, but there are limitations in capturing localized feature representations because their perceptual fields are typically small. In contrast, the self-attention mechanism inherent to the Transformer demonstrates strength in capturing long-range dependencies and contextual relationships, yet lacks sensitivity to localized, fine-grained details. Combining a convolutional network and Transformer has advantages while maximally retaining global and regional feature information.

The proposed model implements the Transformer model to extract global features, with additional convolutional layers preceding the Transformer to enable the network to capture more comprehensive and enriched representations. The model structure includes four stages, each comprising convolutional modules and Transformer structures. It also includes global average pooling layers and linear classifier modules. The first step extracts local features of diseased regions through multi-scale dense convolutional neural networks. The extracted feature maps are then partially input into the Transformer as the second step. The proposed multi-stage design divides the complex diseased images into four stages for feature extraction, ensuring sufficient information input, to some extent, compared to a single-layer design. Compared to a traditional CNN, introducing the Transformer structure through learning global features and token embeddings changes the feature sequence and dimensions of the model to extract disease features. Modeling multi-level local features and lightweight multi-head self-attention reduces the computational costs significantly. The four stages correspond to downsampling factors of 4, 8, 16, and 32, respectively, with a decreasing feature map height/width and increasing channels while keeping the feature dimensions per stage unchanged. Global average pooling and linear classifiers aggregate global information from the feature maps and output disease classification.

Design of Convolutional Blocks

The proposed Dual Vision Transformer (DVT) consists of local feature extraction units, lightweight multi-head attention modules, and a Residual Feed-Forward Network, as shown in Figure 2 and Table 1.

- (1) Dense network (local feature extraction module): As shown in Figure 2a, convolutional structures are utilized 3×3 to densely connect and extract local features from the input image, with a stride of 2 and a channel number of 14 or 16 to lower the image size and computational cost. To better represent the model hierarchically, a supplementary layer comprising a 2×2 convolutional layer and normalization are added before each stage to reduce the model size for projection to higher dimensions.
- (2) ConvNext Block [28]: Compared to ResNet residual blocks, this bottleneck structure has fewer parameters and is more efficient, enhancing the model's perception of global contexts. Using depthwise convolutions instead of regular convolutions reduces parameters. Applying LayerNorm to the channel dimension (N, H, W, C) is more efficient. Two linear layers are used instead of 1×1 convolutions to implement pointwise convolution, improving feature competition and classification performance.

- (3) **Lightweight Multi-Head Attention [29]:** In the original self-attention module, the input $x \in R^{n \times d}$ linear transformation is for $Q \in R^{n \times d_k}$ (query), $K \in R^{n \times d_k}$ (key), and $V \in R^{n \times d_v}$ (value), where $n = H \times W$ is the number of patches. To reduce the computational complexity, depthwise separable convolutions (DWCONVs) are used to extract the key and value instead of directly using the input feature maps, significantly reducing computations. As shown in Figure 2b, the step of inputting the tensor from $n = H \times W$ to $n \times d$ is omitted. And d, d_k , and d_v are the dimensions of the input, keys (queries), and values, respectively. In calculating attention, a learnable bias parameter, B , is added, which helps the module learn positional information. Finally, the output is not the concatenation of multi-head attention, but a linear mapping added to the input, which reduces parameters. Layer normalization is applied to queries before attention calculation, helping to stabilize training. Overall, compared to standard multi-head self-attention, this module has specific improvement while retaining the merits of self-attention.

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (6)$$

$$\text{LWAttention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}} + B\right)V' \quad (7)$$

- (4) **FFN (Residual Feed-Forward Network):** An FFN consists of two linear layers with GELU activation in between, where the first layer expands the dimension by four times and the second layer reduces it correspondingly. The advantage of FFN modules lies in their small number of parameters. An FFN uses the 1×1 convolution to lower and raise the dimension, which is the middle part of using DWCONV, and the overall number of parameters will be less than the large kernel of the standard convolution. The residual connection can effectively prevent the gradient from disappearing in the deep network and play the role of regularization. FFN facilitates training deeper architectures through residual connections, which mitigate gradient vanishing, and provides a regularization effect. Standard convolutions can be replaced with an FFN for hybrid models to reduce parameters and improve computational efficiency while enhancing representational capacity.

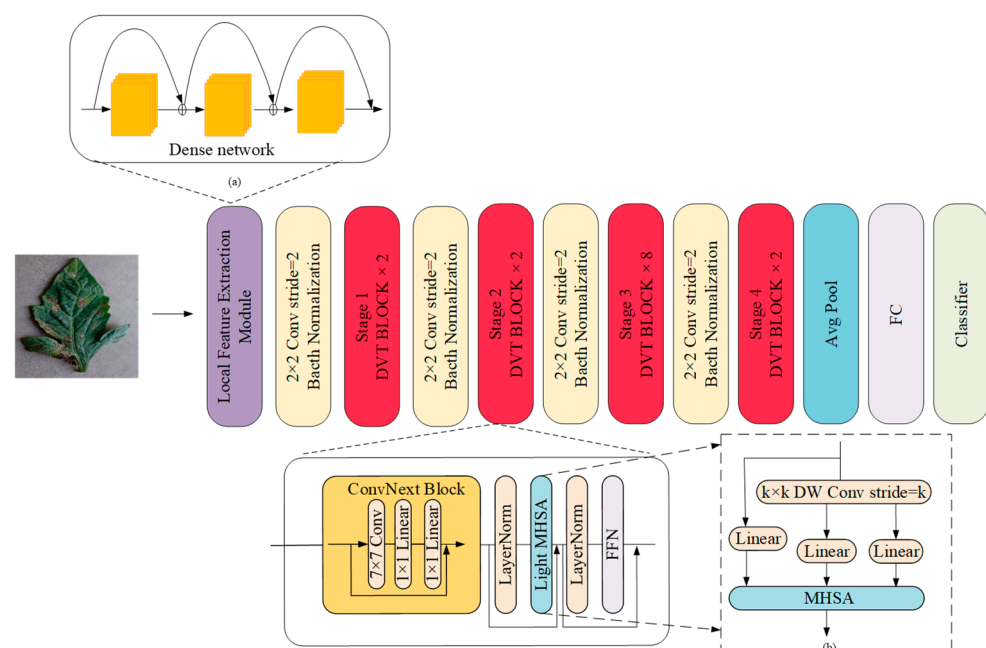


Figure 2. The framework of DVT. (a) Dense network (b) Lightweight Multi-Head Attention.

Table 1. The architecture of the DVT classification model.

Output Size	Layer Name	DVT-Ti	DVT-S	DVT-B
112 × 112	LFEM	3 × 3, 14, stride = 2	3 × 3, 16, stride = 2	3 × 3, 16, stride = 2
56 × 56	Patch Aggregation	[3 × 3, 14] × 3	[3 × 3, 16] × 3	[3 × 3, 16] × 3
	ConvNext	2 × 2, 42, stride = 2	2 × 2, 48, stride = 2	2 × 2, 64, stride = 2
Stage 1	Bottleneck	$\begin{bmatrix} 3 \times 3, 42 \\ H_1 = 1, K_1 = 8 \\ R_1 = 3.6 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 48 \\ H_1 = 1, K_1 = 8 \\ R_1 = 3.6 \end{bmatrix} \times 3$	$\begin{bmatrix} 3 \times 3, 64 \\ H_1 = 1, K_1 = 8 \\ R_1 = 3.6 \end{bmatrix} \times 3$
	LMHSA			
	FFN			
28 × 28	Patch Aggregation	2 × 2, 84, stride = 2	2 × 2, 96, stride = 2	2 × 2, 128, stride = 2
	ConvNext			
Stage 2	Bottleneck	$\begin{bmatrix} 3 \times 3, 84 \\ H_1 = 1, K_1 = 8 \\ R_1 = 3.6 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 96 \\ H_1 = 1, K_1 = 8 \\ R_1 = 3.6 \end{bmatrix} \times 3$	$\begin{bmatrix} 3 \times 3, 128 \\ H_1 = 1, K_1 = 8 \\ R_1 = 3.6 \end{bmatrix} \times 3$
	LMHSA			
	FFN			
14 × 14	Patch Aggregation	2 × 2, 168, stride = 2	2 × 2, 192, stride = 2	2 × 2, 256, stride = 2
	ConvNext			
Stage 3	Bottleneck	$\begin{bmatrix} 3 \times 3, 168 \\ H_1 = 1, K_1 = 8 \\ R_1 = 3.6 \end{bmatrix} \times 8$	$\begin{bmatrix} 3 \times 3, 192 \\ H_1 = 1, K_1 = 8 \\ R_1 = 3.6 \end{bmatrix} \times 10$	$\begin{bmatrix} 3 \times 3, 256 \\ H_1 = 1, K_1 = 8 \\ R_1 = 3.6 \end{bmatrix} \times 12$
	LMHSA			
	FFN			
7 × 7	Patch Aggregation	2 × 2, 336, stride = 2	2 × 2, 384, stride = 2	2 × 2, 512, stride = 2
	ConvNext			
Stage 4	Bottleneck	$\begin{bmatrix} 3 \times 3, 336 \\ H_1 = 1, K_1 = 8 \\ R_1 = 3.6 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 384 \\ H_1 = 1, K_1 = 8 \\ R_1 = 3.6 \end{bmatrix} \times 3$	$\begin{bmatrix} 3 \times 3, 512 \\ H_1 = 1, K_1 = 8 \\ R_1 = 3.6 \end{bmatrix} \times 3$
	LMHSA			
	FFN			
1 × 1	FC		1 × 1, 1280	
1 × 1	Classifier		1 × 1, 10(8, 11)	
Params(M)		8.06	14.05	26.20
FLOPs(G)		1.1	1.4	3.78

3. Implementation and Model Assessment

3.1. Tomato Dataset and Experimental Setup

To evaluate the performance of the model proposed in this study, experiments are conducted using three datasets: the PlantVillage dataset, the 2018 AI Challenger dataset, and a newly constructed dataset combining images collected online and a real greenhouse. These datasets are shown in Table 2. The PlantVillage dataset comprises nine tomato disease classes and one healthy class, totaling 186,110 images. The 2018 AI Challenger dataset contains 11 tomato disease classes, totaling 13,199 images. The self-built dataset has eight tomato disease classes with 3820 images. In the two experiments, all photos are resized to 256 × 256 pixels. For classification experiments, the required datasets are divided into training and validation sets at a ratio of 8:2. The training set is used to train the network, and the validation set is used to evaluate the model's performance. The framework used in the experiment is Pytorch, with a hardware configuration of RTX3090 (24 GB). The specific experimental parameters are shown in Table 3.

Table 2. Category distribution of three tomato datasets.

Category	Our Private	PlantVillage	2018 AI Challenger
Bacterial spot	555	2127	13
Early blight	440	1000	792
Late blight	545	1909	1569
Leaf mold	480	952	755
Mosaic virus	325	373	298
Healthy	310	1591	1381
Spectoria leaf spot	790	1771	1403
Yellow leaf curl virus	375	5357	4442
Two-spotted spider mite	—	1676	975
Target spot	—	1404	74
Powery mildew	—	—	1497
Total	3820	18,160	13,199

Table 3. Experimental parameters of the proposed model.

Parameter	CyTrGAN	DVT
Optimization algorithm	Adam	Adam
Batch size	1	64
Learning rate	0.0001	0.0001
Epochs	200	100
Dropout rate	0.5	0.5
Momentum	0.9	0.9
RMSprop	0.999	0.999

3.2. Evaluation Metrics

We conducted two groups of experiments. In the first set, we utilized the CyTrGAN model with 200 training epochs to generate synthetic tomato disease images across ten categories. Through continuous mutual confrontation between the generator and discriminator, the generator synthesized images approximating the real data distribution. To validate our results, we compared the model with two advanced cyclegan baseline models. The experiments showed that the proposed model generated more explicit novel disease images with CyTrGAN. We selected varying numbers of hundreds to thousands of synthetic images for data augmentation, supplementing the imbalanced categories in the original dataset to form a new dataset. The designed Dual Vision Transformer (DVT) classification model was trained on the original and augmented datasets separately, validating the generalization capabilities of the DVT classification model.

This paper evaluated image differences using FID between generated and original images with structural similarity. FID calculates the Frenchet distance between the mean and covariance of the features of the generated image and the real image. A lower FID value indicates that the generated image is more similar to the real image, and the quality of the generated image is higher.

$$FID = \|\mu_p - \mu_q\|_2 + \text{Tr}\left(\sum_p + \sum_q - 2\sqrt{\sum_p \sum_q}\right) \quad (8)$$

Six metrics were used to assess the performance of the classification experimental models: accuracy, precision, recall, F1-score, model parameter size, and FLOPs. Model parameters determine the model size. FLOPs measure the model complexity. The smaller model has lower hardware requirements and greater adaptability.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (9)$$

$$Precision = \frac{TP}{TP + FP} \quad (10)$$

$$Recall = \frac{TP}{TP + FN} \quad (11)$$

$$F1 - Score = \frac{2 \times precision \times recall}{precision + recall} \quad (12)$$

where TP is true positives, TN is true negatives, FP is false positives, and FN is false negatives.

3.3. Experimental Results and Analysis

3.3.1. Performance of CyTrGAN

The experiment comprehensively compared the effects of tomato leaf disease images generated using the proposed CyTrGAN model and two baseline CycleGAN generators from qualitative and quantitative perspectives.

We analyzed the overall visual effect and detailed reproduction of the synthesized images from a qualitative angle. As shown in Figure 3, the synthetic images produced by

the Resnet and Unet generator networks have problems such as blurring, insufficiently rich textures, and unclear contours. Many of the characteristics of tomato diseases cannot be reproduced well. There is severe image distortion in some areas, which is detrimental for agricultural technicians to identify disease characteristics accurately. In contrast, the tomato disease images generated using CyTrGAN integrating deep features and shallow details are markedly superior to those generated using CycleGAN in preserving the structure and detail features of the original images, facilitating judgment on more information.

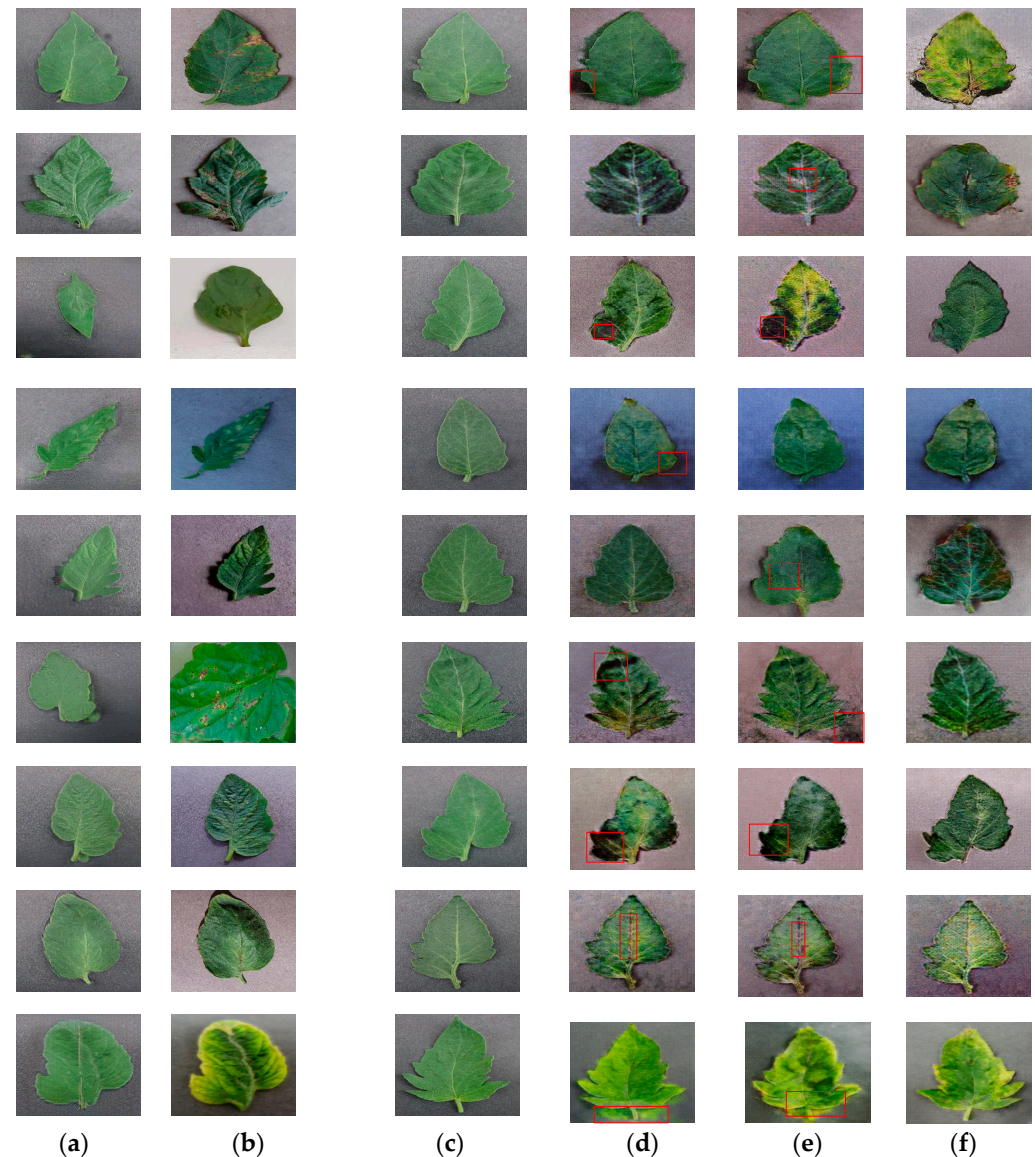


Figure 3. Comparison between original images and generated images: (a) synthesized healthy images, (b) origin disease images, (c) origin healthy images, (d) Resnet generator, (e) Unet generator, (f) CyTrGAN generator.

From a quantitative point of view, we used the FID objective evaluation metric to quantify the quality of tomato leaf disease images generated using different baseline models. As shown in Table 4, the CyTrGAN model has lower FID scores than CycleGAN's two baseline models, indicating that the real images are more similar to the synthesized images and the generated images are more realistic. The CycleGAN model outperformed the CyTrGAN model for two types of diseases, namely mosaic virus and two-spotted spider mite, and the CyTrGAN model outperformed the CycleGAN model for the rest of the diseases. The CyTrGAN model also outperformed the CycleGAN model on the remaining diseases, indicating

that there is a large variability in the images of different types of diseases. On the bacterial spot, early blight, late blight, leaf mold, septoria leaf spot, target spot, and yellow leaf curl virus, they outperformed the lowest results in the comparison models by 7.31, 29.67, 10.92, 17, 2.55, 2.5, 8.83, and 24.21, respectively. The quality and structural information of the CyTrGAN synthesized images are closer to the original images. Therefore, the introduction of the Transformer module and the perceptual loss function enhances the overall performance of the model.

Table 4. FID value of the images synthesized using the two CycleGAN generators and the CyTrGAN generator.

Class	CycleGAN/Resnet	CycleGAN/Unet	CyTrGAN
Bacterial spot	115.26	94.18	86.87
Early blight	131.44	121.17	91.50
Healthy	97.51	160.18	91.72
Late blight	177.40	176.92	166
Leaf mold	193.11	178.70	161.70
Mosaic virus	106.05	97.50	104
Septoria leaf spot	108.15	103	100.45
Two-spotted spider mite	91.53	97.62	94.70
Target spot	187.06	93.63	92.13
Yellow leaf curl virus	139.70	135.60	111.39

In summary, the proposed CyTrGAN model generated higher quality tomato disease synthetic images with richer details than the two CycleGAN baselines, whether analyzed from qualitative or quantitative aspects. The experiment validated the efficacy of CyTrGAN in tomato disease image generation tasks.

3.3.2. Turing Study

To verify the quality of the images generated using the generator, a visual Turing test was proposed for botanist and computer vision research in the context of not providing information about the disease in the pictures. A total of 50 real and 50 synthesized images were selected from each type of diseased tomato disease images, and a total of 500 images were tested. To make the results of the Turing test accurate and reliable, all of the images were resized to 100×100 . The problems faced by the two experts were as follows:

- (1) Confirming whether the images are real or synthetic (Accuracy1);
- (2) Determining the category to which the diseased tomato images belong (Accuracy2).

The results of the experiments are shown in Table 5. The generated images successfully capture the disease features of the actual photos, and the images generated by the two CyTrGAN models proposed in the paper are indistinguishable from the authenticity of the tomato leaf disease images for botanist and vision research, with the highest accuracy reaching 53% and the lowest earning 47%. Therefore, the texture mechanisms of the generated synthetic disease samples and the actual disease samples have similarities and matches.

Table 5. Turing experiment on each component.

Botanist	R-R	R-G	G-R	G-G						
Accuracy1	51%	49%	53%	47%						
Accuracy2	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9
	96%	97%	100%	96%	94%	98%	98%	97%	97%	95%
vision researcher	R-R	R-G	G-R	G-G						
Accuracy1	53%	48%	52%	49%						
Accuracy2	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9
	95%	95%	100%	98%	96%	99%	98%	95%	95%	97%

R-R denotes the number of real images recognized as real images, R-G denotes the number of real images recognized as synthesized images, G-R denotes the quality of synthesized images recognized as real images, G-G denotes the quality of synthesized images recognized as synthesized images, and C0–C9 represents the category of each tomato disease.

3.3.3. DVT as Classification Model

This experiment evaluated the classification performance of the model on three datasets using the PlantVillage tomato dataset as the primary analysis target. In contrast, the 2018 AI Challenger and self-built datasets were used to verify the classification capability of the DVT model. Seven mainstream deep learning models were chosen for comparison, including VGG16, VGG19, Resnet50, Resnet101, MobilenetV2, Vision Transformer, and Swin Transformer. As shown in Table 6, the proposed DVT-B model with the most significant parameters achieved the highest recognition accuracy of 99.76%. DVT-Ti and DVT-S also reached 99.45% and 99.67% accuracies with relatively small parameters of 8.06 M and 14.02 M, respectively. Compared to other mainstream models, the DVT model demonstrated outstanding performance in classification tasks with fewer parameters and FLOPs, providing various model options for different recognition tasks. Figure 4 shows the accuracy and loss curves during the training of the seven baseline and DVT models, from which it can be seen that the model stabilized after 100 epochs. The proposed DVT model attained higher accuracy and lower loss than the mainstream classifiers. Figures 5 and 6 present the validation accuracy and loss curves of the DVT model on the 2018 AI Challenger and self-built datasets, respectively. Tables 7–9 also demonstrate the three datasets' per-class precision, recall, and F1-score.

Table 6. Comparison with tomato recognition models on PlantVillage dataset.

Model	Accuracy (%)	Parameters (M)	FLOPs (MFLOPs)	Model Size (MB)
VGG16	96.46%	138.36	15,470.26	198
VGG19	96.04%	143.67	19,632.06	548
Resnet50	83.72%	25.56	4133.74	97.7
Resnet101	81.44%	44.55	7866.44	170
Mobilenetv2	95.76%	3.50	327.55	13.6
Vision Transformer	98.82%	85.81	16,862.87	327
Swin Transformer [30]	99.10%	27.53	4371.13	105
DVT-Ti	99.45%	8.06	1103.28	31
DVT-S	99.67%	14.02	1950.51	54
DVT-B	99.76%	26.20	3781.19	100

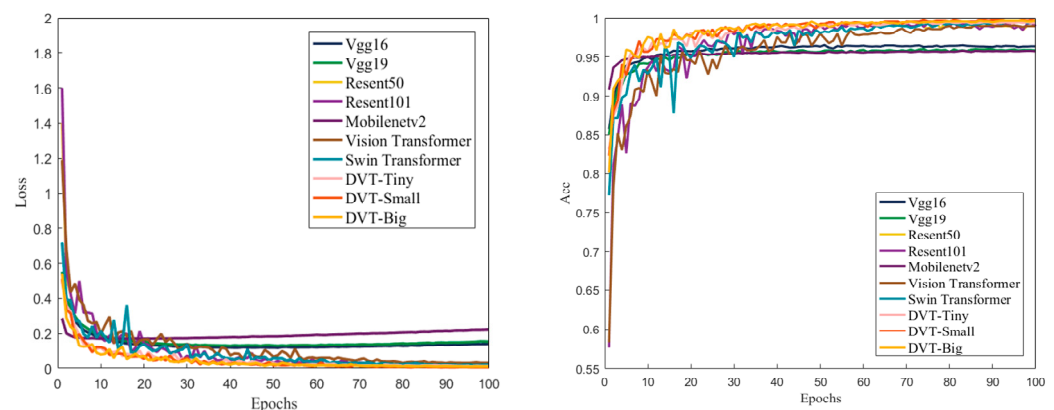


Figure 4. The accuracy and loss of different models on the PlantVillage disease dataset.

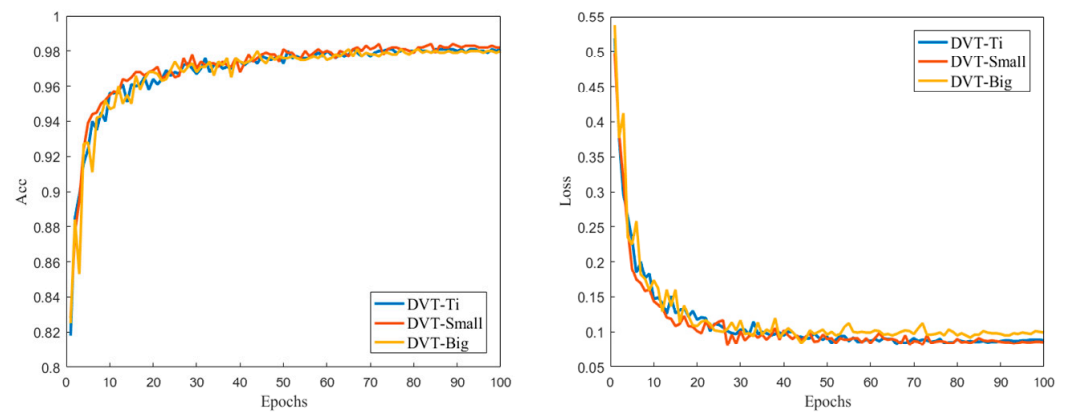


Figure 5. The accuracy and loss of different models on the 2018 AI Challenger disease dataset.

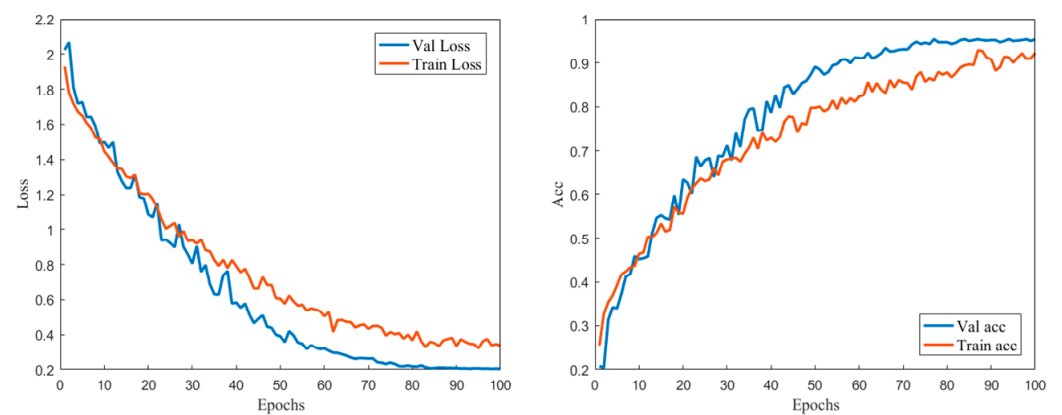


Figure 6. The accuracy of our model on the private dataset; the right is the loss of our model on the validation set.

Table 7. Precision, recall, and F1-score for different disease classes of 2018 AI Challenger.

	Class	F1-Score	Precision	Recall
Bacteria spot	0	1.00	1.00	1.0
Early blight	1	0.95	0.95	0.95
Healthy	2	0.99	0.99	0.99
Late blight	3	0.97	0.97	0.97
Leaf mold	4	0.97	0.98	0.97
Powdery mildew	5	1.00	1.00	1.00
Septoria leaf spot	6	0.98	0.99	0.98
Spider mite	7	0.96	0.95	0.95
Target spot	8	0.71	0.86	0.77
Mosaic virus	9	0.98	0.97	0.97
Yellow leaf curl virus	10	0.99	0.99	0.99

Table 8. Precision, recall, and F1-score for different disease classes of private dataset.

	Class Label	F1-Score	Precision	Recall
Bacterial spot	0	0.95	0.97	0.93
Early blight	1	0.95	0.95	0.95
Late blight	2	0.95	0.92	0.99
Leaf mold	3	0.93	0.93	0.93
Mosaic virus	4	0.95	0.95	0.95
Healthy	5	0.95	0.94	0.94
Septoria spot	6	0.97	0.99	0.99
Yellow leaf curl virus	7	0.96	0.96	0.96

Table 9. Precision, recall, and F1-score for different disease classes of PlantVillage.

Label	Label	F1-Score	Precision	Recall
Bacterial spot	0	1.00	1.00	1.00
Early blight	1	0.98	0.98	0.98
Healthy	2	0.98	0.98	0.98
Late blight	3	1.00	0.99	0.98
Leaf mold	4	1.00	1.00	1.00
Mosaic virus	5	0.98	0.99	0.99
Septoria leaf spot	6	0.99	0.97	0.98
Spider mites	7	1.00	1.00	1.00
Target Spot	8	1.00	1.00	1.00
Yellow leaf curl virus	9	1.00	1.00	1.00

The confusion matrix in Figure 7 shows that most disease categories were correctly identified by the model, indicating strong recognition capability for common diseases. However, the identification performance was poorer for rare classes with fewer samples, which could have been better with some misclassifications. This was primarily due to the imbalanced sample distribution among different disease categories and insufficient samples for some diseases. This prevented the model from obtaining adequate information to learn the characteristics of those rare classes. This suggests that the model's generalization capability was further enhanced by collecting more samples for rare diseases and balancing the sample distribution across categories. As shown in Table 10, the experiments' classification performance on the three original datasets and augmented datasets were compared using CyTrGAN (authentic + synthesized images). This method increased the classification accuracy of the PlantVillage, 2018 AI Challenger, and self-built datasets to 99.65%, 98.91%, and 97.60%, respectively, with noticeable improvements in precision, recall, and F1-score across all categories compared to the original datasets. The classification performance gains demonstrate that CyTrGAN-based data augmentation effectively prevents overfitting, mitigates class imbalance, improves generalization, and enhances model robustness.

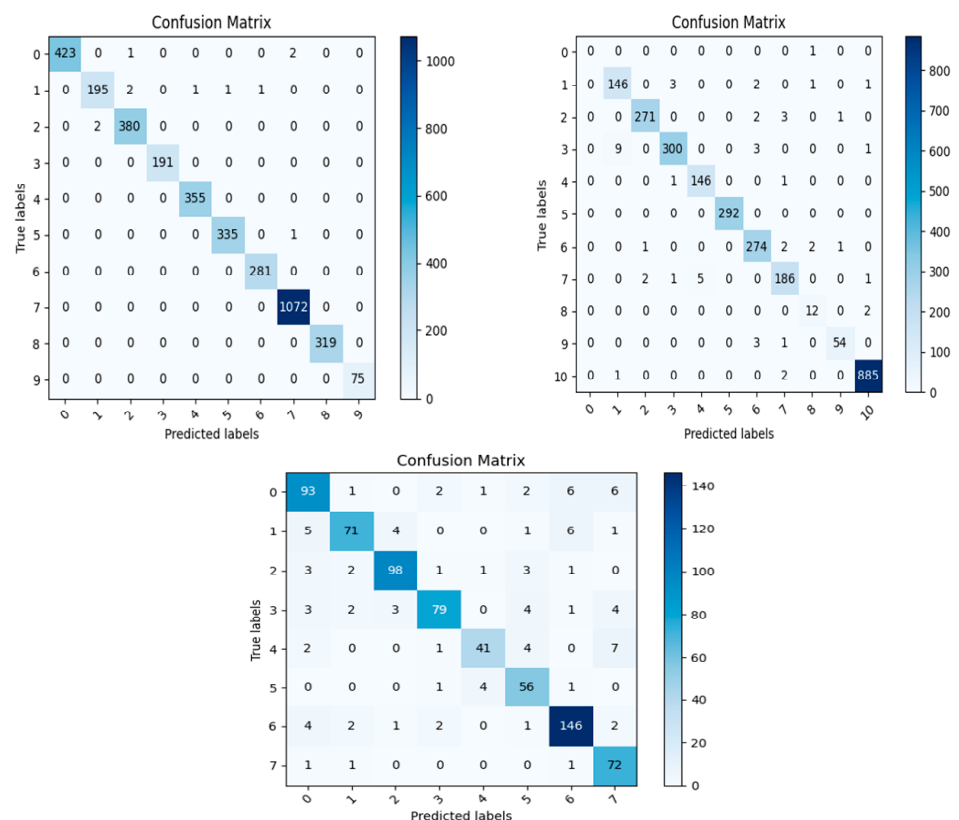
**Figure 7.** Confusion matrix of tomato dataset classification using the proposed method.

Table 10. Accuracy, precision, recall, and F1-score for the model with and without augmentation.

Dataset	Accuracy	Precision	Recall	F1-Score
8 classes				
Origin dataset	95.4%	0.95	0.94	0.94
Origin dataset + synthetic mages	97.60%	0.97	0.97	0.97
10 classes				
Origin dataset	99.45%	0.98	0.99	0.99
Origin dataset + synthetic mages	99.65%	0.99	0.99	0.99
11 classes				
Origin dataset	98.30%	0.97	0.98	0.97
Origin dataset + synthetic mages	98.91%	0.99	0.98	0.98

3.3.4. Performance Comparison

Table 11 compares the proposed model with existing tomato leaf disease classification models. The experimental results show that the model achieved a recognized accuracy of 99.45% while maintaining a small model size and computational cost, which surpasses the current classification models in the literature. Specifically, the proposed method improves the identification accuracy by over 2% compared to the latest research while significantly reducing the computational complexity (FLOPs). This is mainly attributed to the model combining the advantages of a CNN and Transformer, absorbing their advantages in local and global modeling. Meanwhile, the network design ensures high efficiency, a small model size, and a low computational cost. Considering the classification performance and deployment costs, the model demonstrated advantages in tomato leaf disease detection tasks, making it especially suitable for porting to embedded or mobile hardware devices to provide convenient diagnosis for agricultural production. A further analysis in Table 11 shows that the proposed small model occupies merely 29 MB of space, whereas existing works require at least twice the amount of storage to produce similar accuracy. Although some models are smaller than the proposed model, their accuracy could be lower. This further validates the efficacy of the different elements of the proposed structure.

Table 11. Comparison of results with recent works in the literature.

Reference	Dataset	Image Count	Class	Accuracy (%)	Model Size (MB)
Agarwal et al. [31]	PlantVillage	18,160	10	98.70	0.208
Gonzalez-Huitron et al. [32]	PlantVillage	18,160	10	95.24	138.4
Tm et al. [33]	PlantVillage	18,160	10	94.85	156.78
Maeda-Gutierrez et al. [34]	PlantVillage	18,160	10	99.39	23.06
Abbas et al. [15]	PlantVillage	16,012	10	97.11	27.58
Ahmed et al. [21]	PlantVillage	18,160	10	99.30	9.6
Our method	PlantVillage	18,160	10	99.45	29

3.3.5. Visualization of Results

This paper utilized the Grad-CAM technique to visualize the model's attention areas for disease detection to visually observe the image classification model's disease characteristic identification process. As shown in Figure 8, the red areas of heatmaps indicated the highest attention by the model, followed by the yellow regions. By extracting semantic characterization from the images, the model focused primarily on diseased spots and symptomatic areas on the leaves, which are the most relevant regions for plant disease identification. It was shown that the proposed model could pay attention to the diseased parts of the tomato instead of irrelevant background information, thereby making correct disease identifications. Through the visualization of Grad-CAM [35], it can be intuitively understood how the DVT classification model works, and it can be seen that it is fully capable of locating the characteristic areas of the disease and making accurate identification.

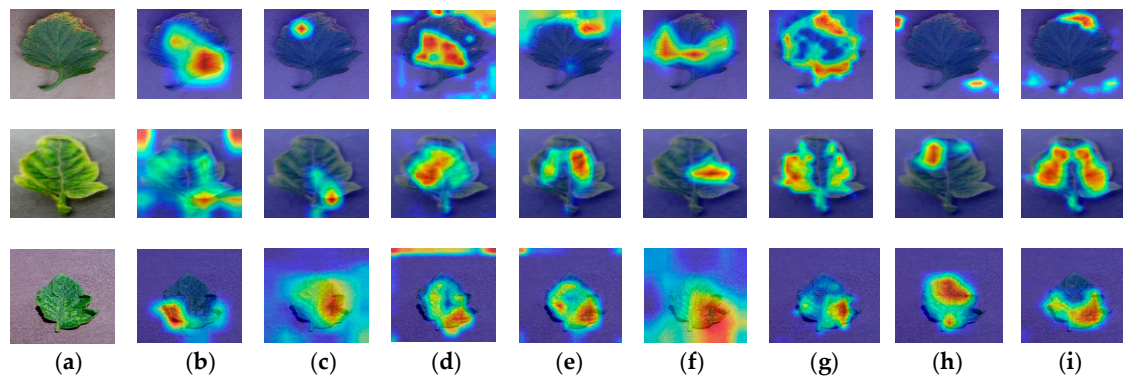


Figure 8. Visualization of Grad-CAM analysis: (a) origin images, (b) VGG16, (c) VGG19, (d) Resnet50, (e) Resnet (101), (f) MobilenetV2, (g) Vision Transformer, (h) Swin Transformer, (i) DVT.

4. Conclusions

Deep learning has shown excellent results in crop disease identification. To resolve the category imbalance, this paper proposed an improved cyclic generative adversarial network (CyTrGAN) based on existing diseased tomato leaves for data augmentation and used the proposed generator to generate a synthetic tomato disease image to alleviate the previous GAN model to produce images with low quality and the presence of severe distortion. The data augmentation technique effectively improved the classification models' generalization capability and robustness. The proposed DVT classification model comprises densely connected networks and Transformer modules. A dense network has advantages in extracting local features from images and capturing critical local disease information. The Transformer module models global contextual dependencies between different image regions through the self-attention mechanism, strengthening the understanding of the international structure. Their combination forms multi-scale feature representations containing both local details and global structures, more comprehensively depicting disease characteristics and enhancing the model's capability in distinguishing complex diseases. The accuracy of the proposed DVT classification model reached 99.45% on the public PlantVillage dataset. The accuracy of the proposed DVT classification model reached 98.30% and 95.4% on the 2018 AI Challenger dataset and self-built private dataset, respectively. The accuracy of the DVT model on the original and the newly combined tomato datasets reached 98.30% and 95.4%, respectively. The accuracy of the DVT model on the original and the newly combined tomato datasets reached 95.4%. The accuracy of the DVT model on the original and the newly combined tomato datasets further increased to 97.90%, 99.63%, and 98.91%, respectively, which has higher identification accuracy compared with other CNN models and further indicates that the experimental model performs well in the task of the multi-categorization of tomato diseases. It can provide powerful support for agricultural disease prevention and control.

As most existing datasets are open-sourced with singular lighting and viewpoint conditions, collecting datasets artificially under relatively pristine environments, an insufficient sample size, and imbalanced category distribution could lead to model overfitting, especially for some rare diseases with insufficient samples, and the authority of the datasets could be improved. Single environments do not reflect complex field conditions, reducing the adaptability of the model. In reality, plant diseases may occur in early, late, and mixed stages. Therefore, the dataset should contain more complex and variable diseases. In the next step, we will construct more diverse, realistic, and larger-scale datasets to improve the generalizability of the model to meet the needs of real-world applications and to deploy the model to a higher level.

Author Contributions: Conceptualization, Z.C.; methodology, Z.C.; software, T.L.; validation, G.W.; investigation, Z.C.; writing—original draft preparation, Z.C.; writing—review and editing, T.L. and X.Z.; supervision, G.W. 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 (No. 51607059).

Data Availability Statement: Data is contained within the article.

Acknowledgments: The authors would like to acknowledge the usage of ChatGPT Tool (OpenAI, San Francisco, CA, USA; accessed on 12 December 2023), an AI-based system, for writing assistance, which helped to improve language fluency and make the content more accessible and understandable.

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

References

1. Panno, S.; Davino, S.; Caruso, A.G.; Bertacca, S.; Crnogorac, A.; Mandić, A.; Noris, E.; Matić, S. A review of the most common and economically important diseases that undermine the cultivation of tomato crop in the mediterranean basin. *Agronomy* **2021**, *11*, 2188. [\[CrossRef\]](#)
2. Saleem, M.H.; Potgieter, J.; Arif, K.M. Plant disease detection and classification by deep learning. *Plants* **2019**, *8*, 468. [\[CrossRef\]](#) [\[PubMed\]](#)
3. Tanner, M.A.; Wong, W.H. The calculation of posterior distributions by data augmentation. *J. Am. Stat. Assoc.* **1987**, *82*, 528–540. [\[CrossRef\]](#)
4. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. *Adv. Neural Inf. Process. Syst.* **2014**, *27*.
5. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:14091556.
6. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016.
7. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
8. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
9. Hassan, S.M.; Jasinski, M.; Leonowicz, Z.; Jasinska, E.; Maji, A.K. Plant disease identification using shallow convolutional neural network. *Agronomy* **2021**, *11*, 2388. [\[CrossRef\]](#)
10. 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. [\[CrossRef\]](#)
11. 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. [\[CrossRef\]](#)
12. Zhu, F.; He, M.; Zheng, Z. Data augmentation using improved cDCGAN for plant vigor rating. *Comput. Electron. Agric.* **2020**, *175*, 105603. [\[CrossRef\]](#)
13. 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*, 7630926. [\[CrossRef\]](#)
14. Cap, Q.H.; Uga, H.; Kagiwada, S.; Iyatomi, H. Leafgan: An effective data augmentation method for practical plant disease diagnosis. *IEEE Trans. Autom. Sci. Eng.* **2020**, *19*, 1258–1267. [\[CrossRef\]](#)
15. Abbas, A.; Jain, S.; Gour, M.; Vankudothu, S. Tomato plant disease detection using transfer learning with C-GAN synthetic images. *Comput. Electron. Agric.* **2021**, *187*, 106279. [\[CrossRef\]](#)
16. Sagar, A.; Dheeba, J. On using transfer learning for plant disease detection. *bioRxiv* **2020**. [\[CrossRef\]](#)
17. Widiyanto, S.; Fitrianto, R.; Wardani, D.T. Implementation of convolutional neural network method for classification of diseases in tomato leaves. In Proceedings of the 2019 4th International Conference on Informatics and Computing (ICIC), Semarang, Indonesia, 16–17 October 2019.
18. Albogamy, F.R. A Deep Convolutional Neural Network with Batch Normalization Approach for Plant Disease Detection. *Int. J. Comput. Sci. Netw. Secur.* **2021**, *21*, 51–62.
19. Zhao, S.; Peng, Y.; Liu, J.; Wu, S. Tomato leaf disease diagnosis based on improved convolution neural network by attention module. *Agriculture* **2021**, *11*, 651. [\[CrossRef\]](#)
20. Elhassouny, A.; Smarandache, F. Smart mobile application to recognize tomato leaf diseases using convolutional neural networks. In *Collected Papers. Volume XI: On Physics, Artificial Intelligence, Health Issues, Decision Making, Economics, Statistics*; Global Knowledge Publishing House: Chennai, India, 2019; p. 431.
21. Ahmed, S.; Hasan, M.B.; Ahmed, T.; Sony MR, K.; Kabir, M.H. Less is more: Lighter and faster deep neural architecture for tomato leaf disease classification. *IEEE Access* **2022**, *10*, 68868–68884. [\[CrossRef\]](#)
22. Yang, B.; Wang, Z.; Guo, J.; Guo, L.; Liang, Q.; Zeng, Q.; Zhao, R.; Wang, J.; Li, C. Identifying plant disease and severity from leaves: A deep multitask learning framework using triple-branch Swin Transformer and deep supervision. *Comput. Electron. Agric.* **2023**, *209*, 107809. [\[CrossRef\]](#)
23. Li, X.; Li, S. Transformer help CNN see better: A lightweight hybrid apple disease identification model based on transformers. *Agriculture* **2022**, *12*, 884. [\[CrossRef\]](#)
24. Hughes, D.; Salathé, M. An open access repository of images on plant health to enable the development of mobile disease diagnostics. *arXiv* **2015**, arXiv:151108060.

25. Zhu, J.-Y.; Park, T.; Isola, P.; Efros, A.A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017.
26. Alimanov, A.; Islam, M.B. Retinal Image Restoration using Transformer and Cycle-Consistent Generative Adversarial Network. In Proceedings of the 2022 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS), Penang, Malaysia, 22–25 November 2022.
27. Johnson, J.; Alahi, A.; Fei-Fei, L. Perceptual losses for real-time style transfer and super-resolution. In Proceedings of the Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; Proceedings, Part II. Springer: Berlin/Heidelberg, Germany, 2016.
28. Woo, S.; Debnath, S.; Hu, R.; Chen, X.; Liu, Z.; Kweon, I.S.; Xie, S. Convnext v2: Co-designing and scaling convnets with masked autoencoders. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023.
29. Guo, J.; Han, K.; Wu, H.; Tang, Y.; Chen, X.; Wang, Y.; Xu, C. CMT: Convolutional neural networks meet vision transformers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022.
30. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021.
31. Agarwal, M.; Gupta, S.K.; Biswas, K. Development of Efficient CNN model for Tomato crop disease identification. *Sustain. Comput. Inform. Syst.* **2020**, *28*, 100407. [[CrossRef](#)]
32. Gonzalez-Huitron, V.; León-Borges, J.A.; Rodriguez-Mata, A.; Amabilis-Sosa, L.E.; Ramírez-Pereda, B.; Rodriguez, H. Disease detection in tomato leaves via CNN with lightweight architectures implemented in Raspberry Pi 4. *Comput. Electron. Agric.* **2021**, *181*, 105951. [[CrossRef](#)]
33. Tm, P.; Pranathi, A.; SaiAshritha, K.; Chittaragi, N.B.; Koolagudi, S.G. Tomato leaf disease detection using convolutional neural networks. In Proceedings of the 2018 11th International Conference on Contemporary Computing (IC3), Noida, India, 2–4 August 2018.
34. Maeda-Gutiérrez, V.; Galván-Tejada, C.E.; Zanella-Calzada, L.A.; Celaya-Padilla, J.M.; Galván-Tejada, J.I.; Gamboa-Rosales, H.; Luna-Garcia, H.; Magallanes-Quintanar, R.; Guerrero Mendez, C.A.; Olvera-Olvera, C.A. Comparison of convolutional neural network architectures for classification of tomato plant diseases. *Appl. Sci.* **2020**, *10*, 1245. [[CrossRef](#)]
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. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.