

An Improved DCGAN-Based Recognition Enhancement Method for American *Hyphantria cunea* Larvae Net Curtain Image Dataset

Shaomin Teng,^{a,b} Chengming Wang,^c Shunyi Shang,^c Yuxuan Tuo,^c and Decheng Wang^{a,*}

The fall webworm (*Hyphantria cunea*) poses a significant threat to agriculture, as its larvae feed on leaves and form silken webs, which can severely impact plant growth. However, the lack of specific image datasets for the larvae's webs hinders the use of image recognition technologies in pest prevention and control. To address this issue, an enhancement method is proposed here based on an improved Deep Convolutional Generative Adversarial Network (DCGAN). This method generates a diverse set of high-quality web images, significantly expanding the existing dataset. Experimental results demonstrated that this enhanced dataset improved the robustness of recognition networks, enabling better automatic identification and precision spraying to control *Hyphantria cunea*. This approach not only advances automated pest monitoring in agriculture but also offers new possibilities for applying similar technologies to the identification of other plant pests.

DOI: 10.15376/biores.19.4.9271-9284

Keywords: American *Hyphantria cunea* larvae net curtain; Generative adversarial network; Data enhancement; Convolutional Neural Network; Checkerboard effect; Plant pest control; Patent

Contact information: a: College of Engineering, China Agricultural University, Beijing 100083, China; b: Menoble Co., Ltd., Beijing 100083, China; c: School of Mechanical and Automotive Engineering, Liaocheng University, Liaocheng 252059, China; *Corresponding author: wdc@cau.edu.cn

INTRODUCTION

The American *Hyphantria cunea*, also known as the autumn curtain moth, is a key target for forestry control because of its high reproduction rate and rapid spread (Yang *et al.* 2008). The damage of American *H. cunea* is primarily caused by larvae feeding on leaves, and the larvae start to feed a few hours after hatching and spit out silk and form a net curtain. The whole larval stage feeds heavily, causing low resilience of the tree and even death of the whole plant in severe cases (Haijun *et al.* 2006). The larval stage has a clear net curtain, which becomes the best period for control. Chemical control is currently the most effective method. The common way is to manually spray the chemical agent on a large scale. However, this method is inefficient and causes serious environmental pollution. There is an urgent need for the emergence of an intelligent spraying technology to achieve accurate automated on target spraying operations. Accurate target identification is a prerequisite for achieving on-target spraying. In recent years, with the development of neural networks, many deep learning-based methods have been widely promoted and applied in the field of pest and disease identification in agriculture and forestry, while deep learning algorithms require huge data sets as training support (Ding *et al.* 2019). For the acquiring of images of American *H. cunea* larvae net curtains, there are problems, such as

the difficulty of acquiring images of net curtains in higher and deeper parts of the bush, high manual acquisition workload, and large differences in images under different lighting conditions, which make it difficult to form a sufficiently large database.

To expand the original dataset and enhance the generalization ability of neural network models (Yang and Li 2021), many methods have been proposed. Some literature presented enhanced datasets using geometric transformations of the original images (including various operations such as deformation, cropping, mirroring, scaling, and rotation) (De Andrade 2019); some literature adjusted the brightness and contrast of the original images randomly or added random noise to the original images, increasing the number of samples in the dataset (Lopes *et al.* 2017), using the method of adding Gaussian noise to the images to generate new images; some other literature adopted the method of randomly intercepting or randomly masking a part of the images (Sun *et al.* 2017), using the replacement of different regions of the images to generate new images. These methods do not take full advantage of the intrinsic characteristics of the original samples, resulting in a trained neural network model with limitations and poor generalization ability. To solve this problem, automatic image generation was created and in 2004, a method to generate new datasets using neural networks was first proposed in the literature (Zhou and Jiang 2004). Since then, it has been one of the key research directions in the field of machine vision (Radford *et al.* 2015; Isola *et al.* 2016; Grant-Jacob *et al.* 2022). Meanwhile, the rapid development of deep learning has greatly facilitated the development of image generation techniques. The proposal of Generative Adversarial Networks (GAN) provides a completely new solution (Goodfellow *et al.* 2014). The GAN has been continuously improved and has been applied in many fields such as generating audio (Yamamoto *et al.* 2019), high resolution images (Karras *et al.* 2017), and image style conversion (Yang *et al.* 2022). In this paper, an improved DCGAN (Deep Convolutional Generative Adversarial Network) is purposefully designed based on the characteristics of American *Hyphantria cunea* larvae net curtain images, which enables the existing dataset to be enhanced, and the use of the enhanced dataset to avoid the occurrence of overfitting during training and improves the generalization ability of the model.

EXPERIMENTAL

Preparation of the Training Set

The experimentally taken pictures of the partly real net screen are shown in Fig. 1, and the resolution of the pictures was 960×720 . In this paper, a series of real images represented in Fig. 1 are cropped into thousands of 64×64 resolution images and manually picked and classified. The infected leaf images were sorted out, some of which are shown in Fig. 2. In this paper, these images were manually sorted again, and similar net curtain images were grouped into a category. There were 12 categories of American *Hyphantria cunea* larvae net curtain images that were sorted out.

The Improvements of the Checkerboard Artifacts

In the original DCGAN, when the images generated by the deconvolution network were carefully observed (Fig. 3(a)), a distinct Checkerboard Artifact (Kingma and Ba 2014; Cao *et al.* 2023) was observed, which is due to the uneven and overlapping pixels of the image caused by the deconvolution operation, and the visual transition was not smooth due to the different color shades of the adjacent parts of the image. To alleviate the

checkerboard effect, in this work the deconvolution layer was eliminated in the original DCGAN network and instead a resized convolution layer was used, consisting of an upsampling 2D operation and a forward convolution (Conv2D) operation with a step size of 1. Figure (3) shows the comparison of the images generated by using resize convolution and deconvolution at different epochs. The training process using the original deconvolution layer for the upsampling operation is shown in Fig. 3(a), and the training process after changing to the resize convolution layer is shown in Fig. 3(b).



Fig. 1. Real net screen image



Fig. 2. Infected leaf

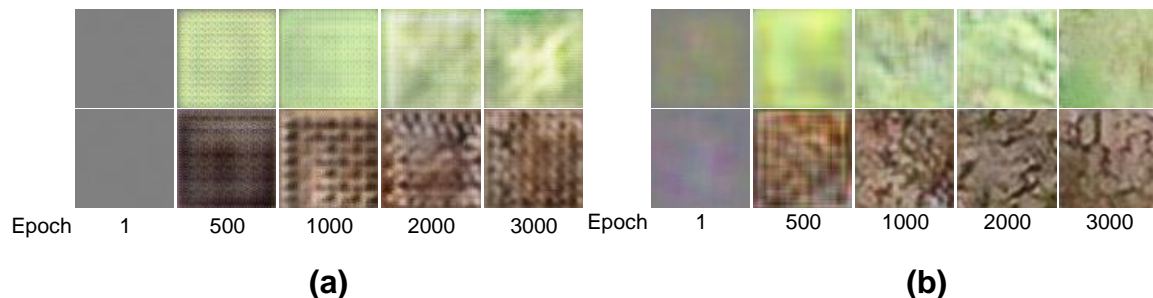


Fig. 3. Comparison of training process before and after improvement: (a) images generated using deconvolution at different epochs, and (b) images generated using resize convolution at different epochs

Table 1. Comparison of FID Indicators for Using Different Convolution Methods

Using Different Convolution Methods	FID Score
Using resize convolution	147.30
Using deconvolution	288.41

From Table 1, it can be judged that the use of resize convolution was much better than the use of deconvolution.

Measures Related to Improving Network Stability

To prevent the network from overfitting, to prevent the parameters from relying too much on the training data, and to increase the generalization ability of the parameters to the dataset, dropout layers were added to both the generator and the discriminator in this paper (Srivastava *et al.* 2014; Park and Kwak 2016). This paper compared the training effect of adding the Dropout layer and not adding the Dropout layer, as shown in Fig. 4.

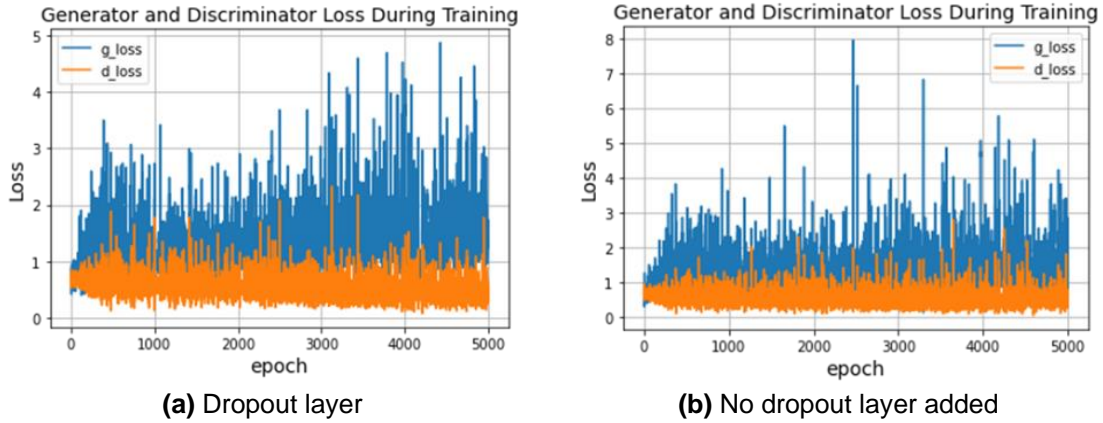


Fig. 4. Comparison of loss values with (a) and without the dropout layer (b)

Table 2 shows the FID metric score for the images generated with and without dropout.

Table 2. Comparison of FID Indicators for Using Different Convolution Methods

Whether to use Dropout	FID Score
Using dropout	147.30
Without dropout	148.19

In this paper, different loss functions were used: Binary cross entropy loss, Categorical cross entropy loss, KL divergence loss, Mean square error (MSE) loss, and Mean absolute error loss. Experiments were conducted using these loss functions separately to compare their effects, as shown in Fig. 5.

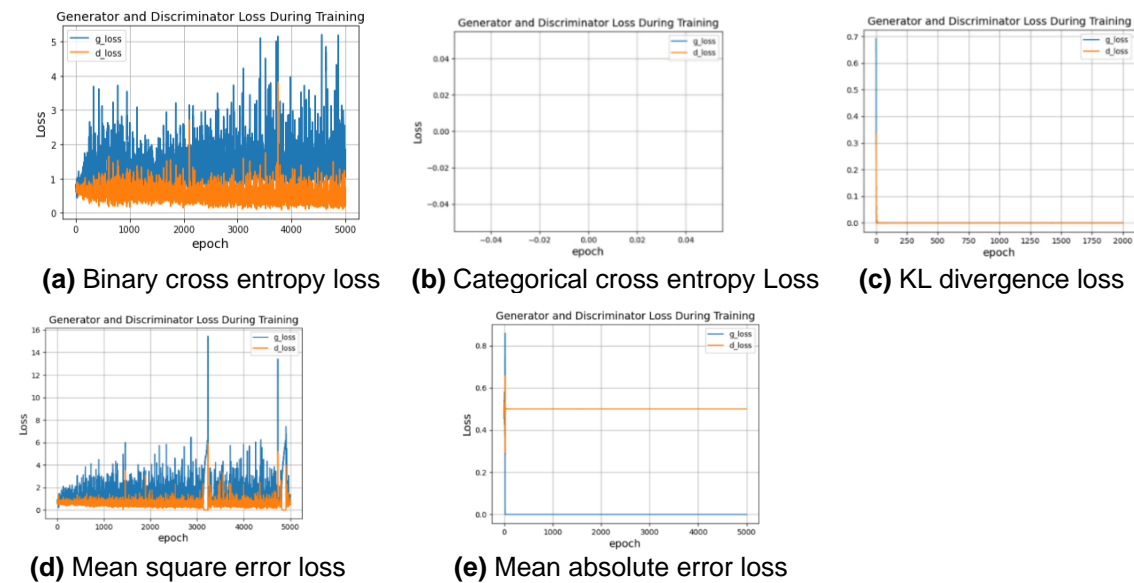


Fig. 5. Comparison of the effect of different loss functions

From Fig. 5, all the loss functions except the MSE loss function showed gradient disappearance when applied. Initially, it was thought that the learning rate was not adjusted

properly, which led to anomalies in the training process when using other loss functions, but after adjusting the learning rate several times, it was found that the size of the learning rate does not affect the result of its gradient vanishing. However, by comparing the use of MSE loss with binary cross entropy loss, it was not difficult to find that when using MSE loss, the loss value of the generator has two large abrupt changes, which indicates the network is less stable. As shown in Fig. 6, it is also clear from its training process that the quality of the generated images was very poor when the epoch was either 3200 or 4800.

As shown in Table 3, by comparing the FID metric score of the generated images using MSE loss and Binary cross entropy loss it is also evident that using Binary cross entropy loss worked well.

Table 3. Comparison of IS Metrics and FID Metrics for Images Generated by Different Loss Functions

Using Different Loss Functions	FID Score
Using MSE loss	264.2
Using binary cross entropy loss	251.7

After the above comparison, the loss function was finally determined as the binary cross entropy loss. This paper used the Adam (Kingma and Ba 2014; Cao *et al.* 2023) algorithm to update the parameters.

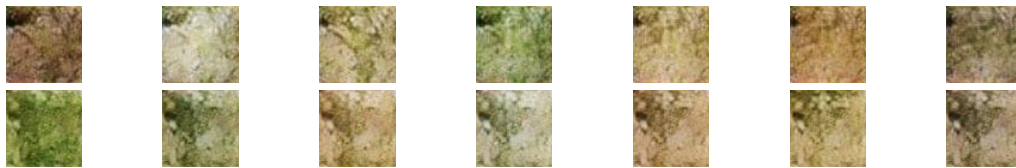
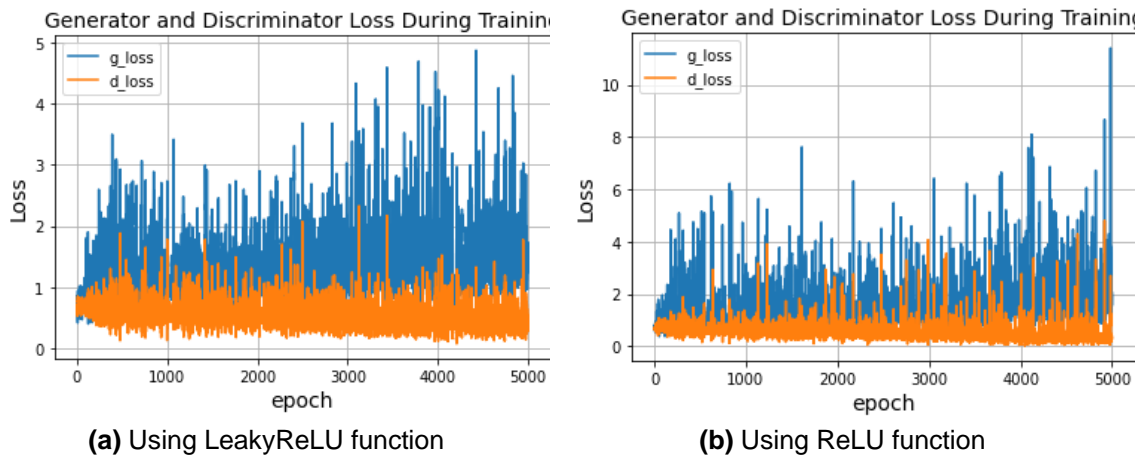
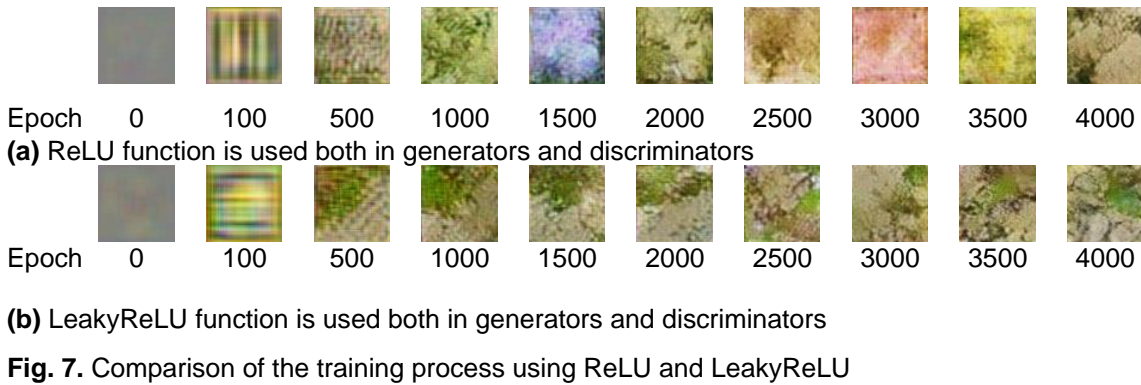
To verify the effect of using LeakyReLU and ReLU, the procedure was modified on the activation function only, and the training results are shown in Fig. 7. Figure 7(a) shows the image generated when both the generator and discriminator used the ReLU function, and Fig. 7(b) shows the image generated when both the generator and discriminator used the LeakyReLU function. The change in loss values when using the LeakyReLU function and when using the ReLU function is shown in Fig. 8. As shown in Fig. 8, the training process of the neural network was more stable when using the LeakyReLU function.

However, when using the ReLU function, images that were different from the training data but matched the characteristics of the American white moth screen image were generated, as shown in Fig. 9. In other words, training using the ReLU function not only expanded the number of datasets, but also expanded the variety of datasets. In this paper, LeakyReLU and ReLU functions were used to train and collect the final resulting qualified images, respectively.

Next, BN layers were added at different locations of the generator and discriminator to further explore the impact of BN layers. Adding an upsampling layer to the activation function was considered.



Fig. 6. Images are generated using different loss functions



The generator was divided into multiple modules. The convolution layer and the activation function layer of the discriminator were treated as two modules. As shown in Fig. 10, the loss variation of the network when BN layers are added to the first module, second module, third module, and fourth module of the generator network only; to the first module, second module, third module, and fourth module of the discriminator network only.

As shown in Fig. 10, the loss value change was anomalous when adding the BN layer to the rest of the positions except for adding the BN layer to the second module and the third module. Therefore, as shown in Table 4, this work only compares the FID index scores of the generated images when the BN layer is added in the second module, the third module, and when no BN layer is added.

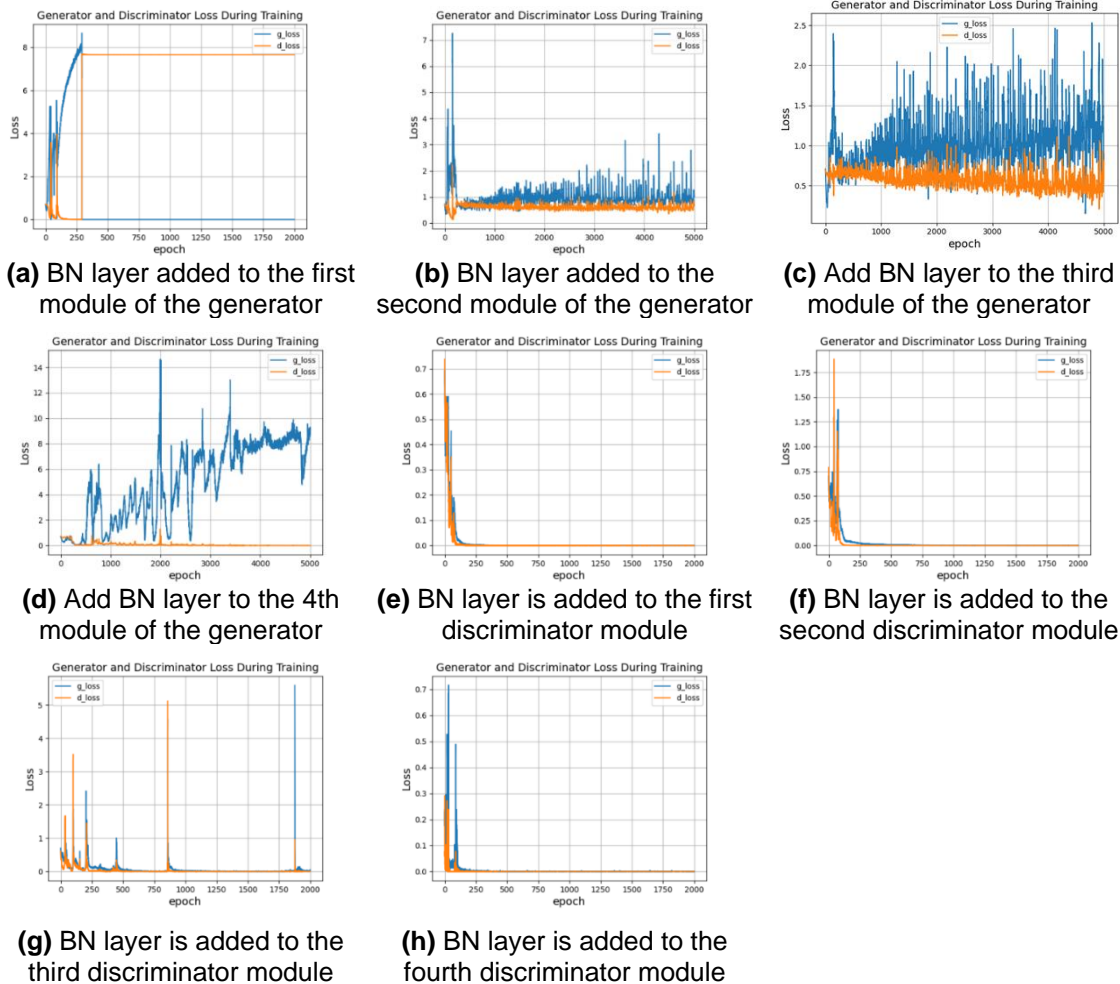


Fig. 10. Variation of loss values when adding BN layers at different locations

Table 4 indicates that the generated image was closest to the real image when BN layer was not used. Figure 11 shows the images generated during the training process for the three cases, and it can be seen that the addition of the BN layer made the network converge more slowly. In summary, the BN layer was not used in this paper.

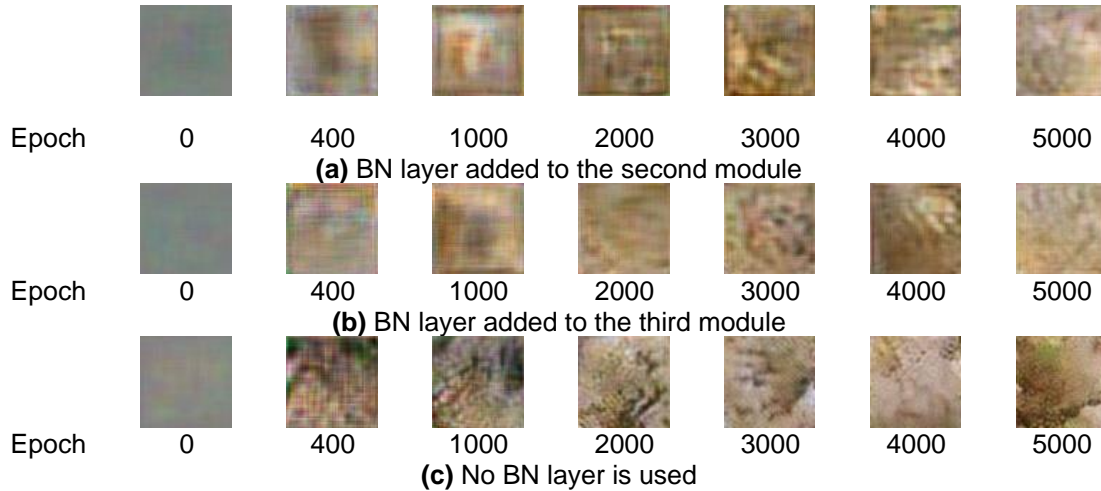
Gan-generated Image Quality Evaluation Validation

IS uses a pre-trained Inception network to classify the generated images and evaluates how confident the network is in its classifications. High confidence in predictions (low entropy) and a wide variety of predicted classes contribute to a higher score. However, IS has limitations because it does not directly compare generated images to real ones and may give high scores even to low-quality images if they appear diverse.

FID compares the statistical distribution (mean and covariance) of generated images to real images in the feature space of a pre-trained network. A lower FID score means the generated images are more similar to real images in terms of quality and diversity. FID is widely preferred because it provides a more direct and reliable measure of image similarity and quality by accounting for differences in the visual features of both datasets.

Table 4. FID Scores or Different Cases of Adding BN Layers

Location of the Added BN layer	FID Score
Add the BN layer to the second module	278.06
Add the BN layer to the third module	247.97
No BN layer is used	147.29

**Fig. 11.** The training process for different cases of adding BN layers

The GAN has two evaluation metrics: the IS metric and the FID metric. The IS metric scores a single dataset by comparing it to a single dataset, with larger values indicating higher image quality and category richness. The FID gives a score by comparing the real image to the generated image, with lower scores indicating that the generated image is closer to the real image.

The specificity of the American *Hyphantria cunea* larvae net curtain images is illustrated in Fig. 6. The images shown in Fig. 6 belong to a certain class of training set, in which the images are relatively very similar. But even so, when the images in the training set shown in Fig. 1 were divided into two parts and their FID scores were evaluated, the FID scores were still as high as 122. To explore further, the paper used the same method described above for other category datasets and evaluated their FID scores. It was finally found that the FID scores between different images in the training set of the same category ranged from 100 to 350.

**Fig. 12.** Images contained in a certain type of training set

Improved Generative and Discriminative Networks

LeakyReLU was used as the activation function at the end of each resize convolution layer in the hidden layer. The structure of the generator network is shown in

Fig. 13(a), and the structure of the discriminator network is shown in Fig. 13(b). The architecture of the generator and discriminator are shown in Fig. 14(a) and Fig. 14(b), respectively.

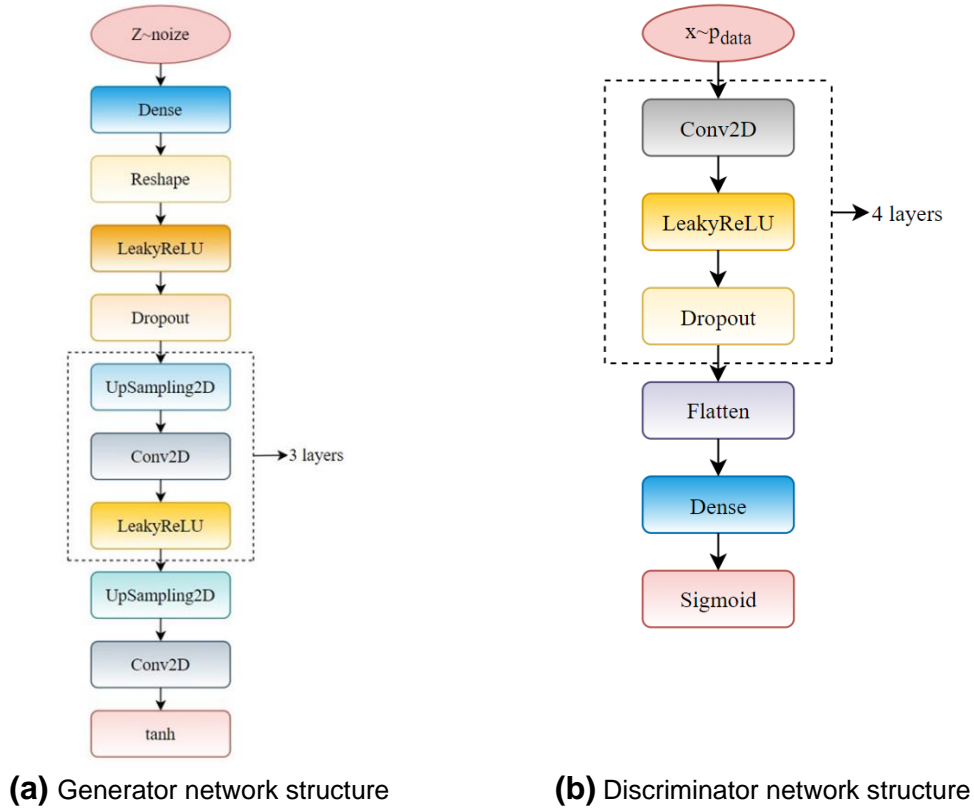


Fig. 13. Generator network structure

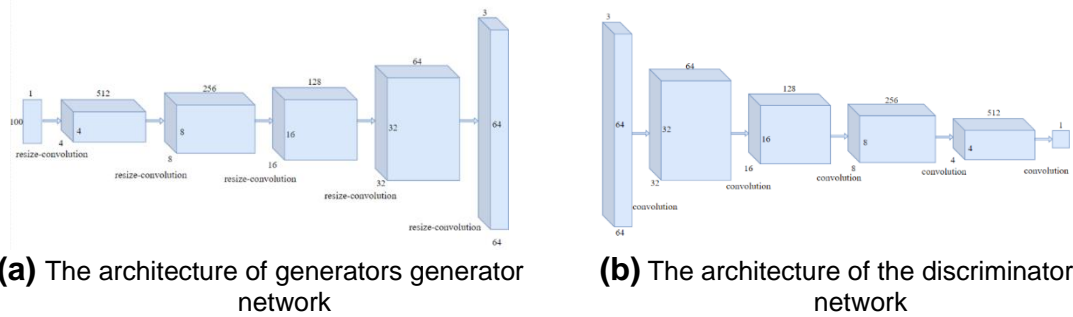


Fig. 14. Architecture of the generator and discriminator

Experimental Platform and Parameter Settings

The computer operating system used in this paper was Windows 64bit system, and the hardware used in this experiment was Intel(R) Core(TM) i76700 CPU, 16 GB RAM, and NVIDIA GeForce RTX 2080 Ti 14 GB, and the software environment is TensorFlow GPU 2.0.0 (Google LLC; Mountain View, CA, USA) and Keras 2.3.1 (Google LLC; Mountain View, CA, USA). The batch size of the training process was set to 64, and the

LeakyReLU slope parameter was set to 0.2. The learning rate of the optimizer Adam was set to $1e05$, the parameter β_1 was set to 0.5, and epsilon was set to $1e05$.

RESULTS AND DISCUSSION

The training process is shown in Fig. 15. The comparison between the generated image and the original image is shown in Fig. 16.

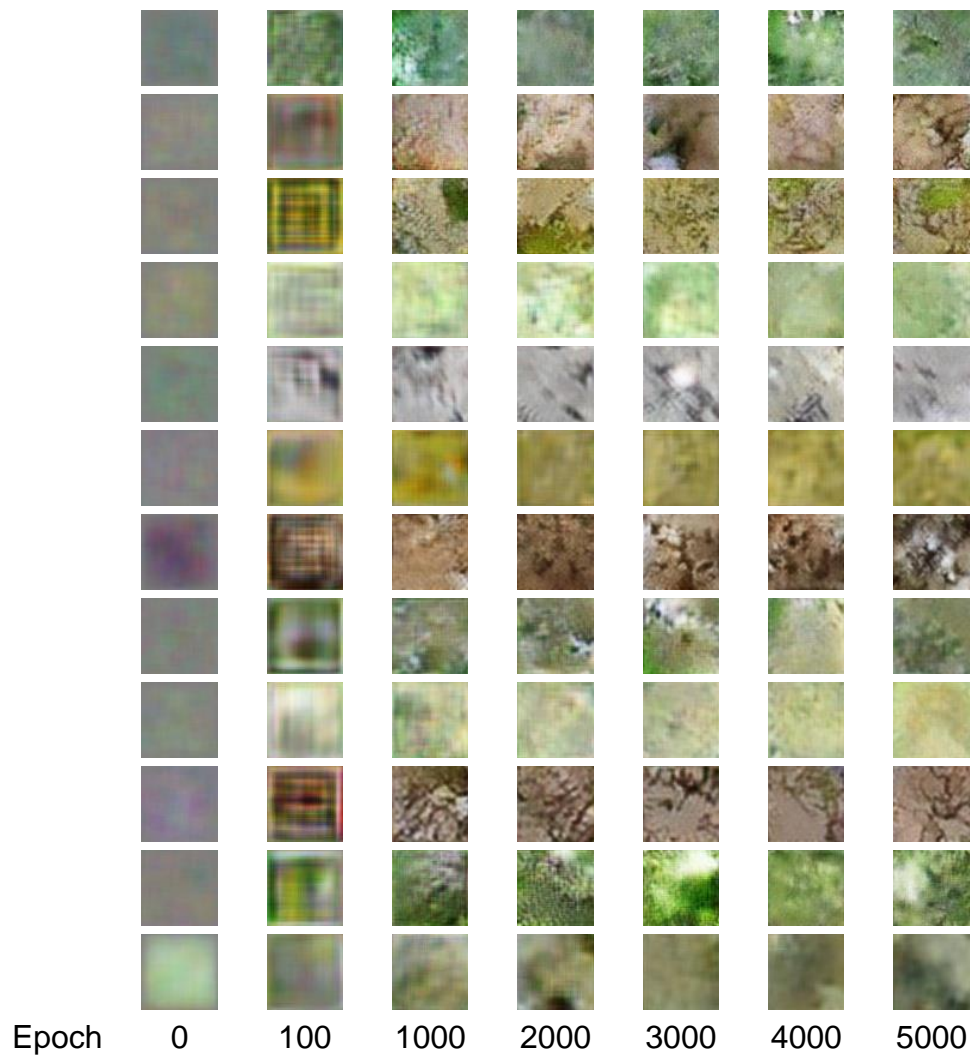


Fig. 15. GAN training process

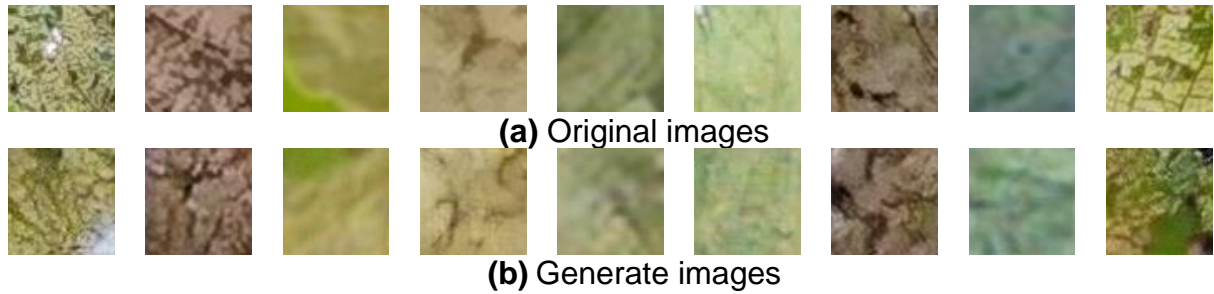
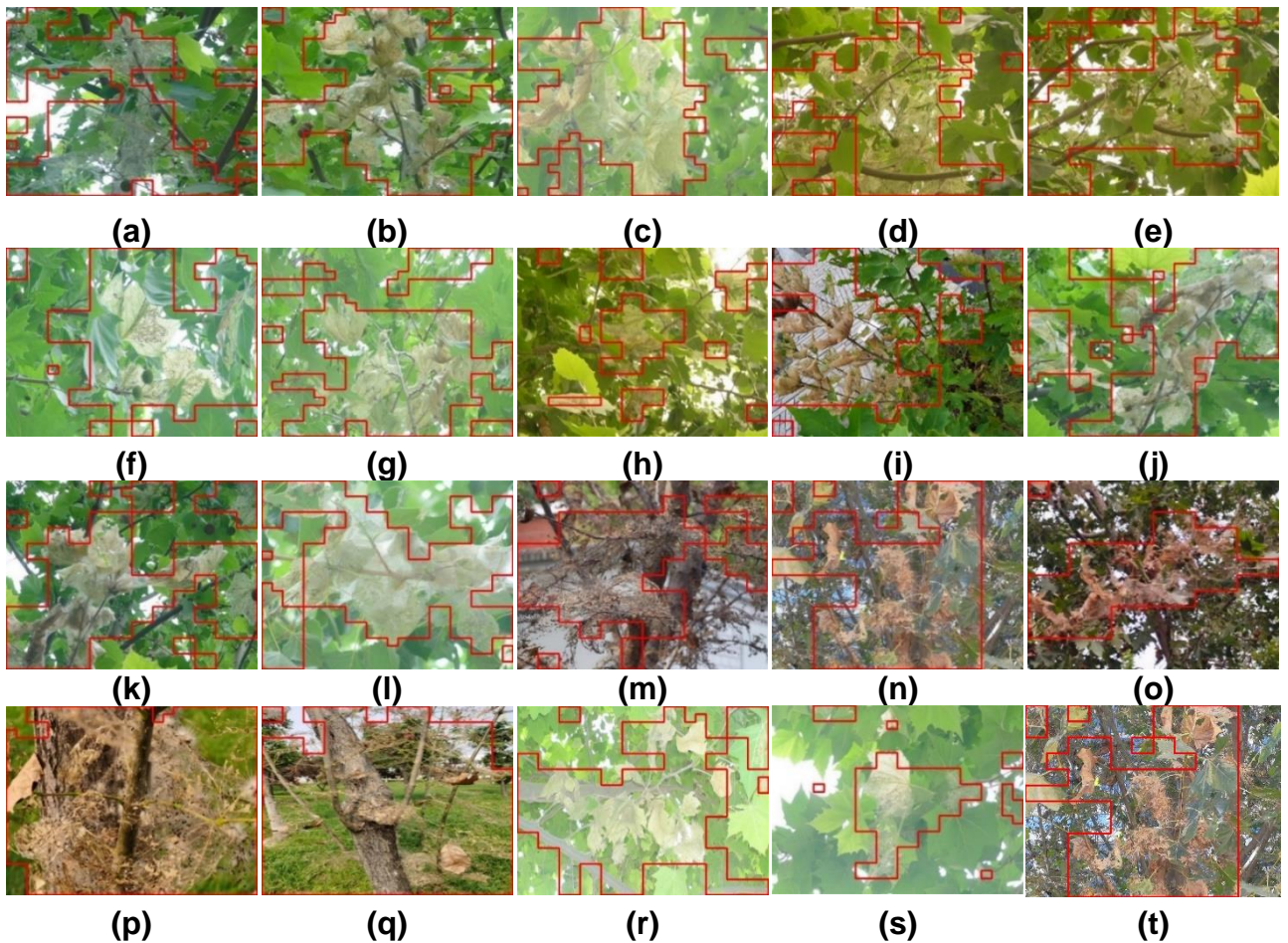


Fig. 16. Comparison of the generated images with the original images

The expanded images were picked separately to remove the images with too high repetition to obtain twelve categories of expanded datasets, at which time the expanded datasets reached the 20,000 level, and there were no real images from the original collection in the expanded datasets. Using this dataset to train the net curtain recognition algorithm based on the convolutional neural network proposed in the literature (Gao *et al.* 2020), the recognition results of the trained algorithm for real net curtain images are shown in Fig. 17.



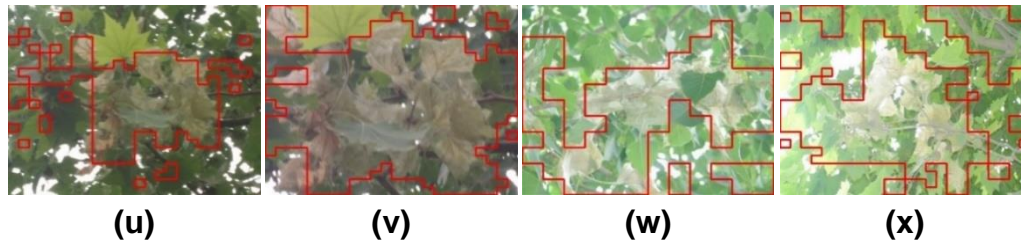


Fig. 17. Recognition effect diagram

The recognition results showed that the image recognition algorithm trained using the expanded net curtain dataset of this method can achieve good recognition under different scenes and lighting conditions. The expanded mesh dataset includes the expanded images obtained by training the GAN with four cropped images from Fig. 17(a) through (k), excluding the image parts in (l) through (x). The recognition images (a) through (k) achieved a high accuracy rate, which shows that the images generated by the algorithm met the requirements and reached the quality standards. Figures 17(l) through (x) also obtained high recognition rates, indicating that the expansion of the dataset improved the generalization ability to recognize the localized mesh curtain. Figures 17(d) and (e) are images taken at different angles, and (j) was obtained from (k) after flipping. All four images achieved good recognition results, which indicates that the algorithm has a strong generalization ability for the recognition of images at different angles after training with the expanded dataset.

Future Directions in Pest Control

The current trajectory of pest control in agriculture is heavily influenced by advancements in deep learning and image processing. Key future developments include refining image generation algorithms, integrating multimodal data sources, enabling real-time deployment, addressing ethical and environmental concerns, and fostering collaborative research efforts. Embracing these avenues will lead to more effective and sustainable pest management practices in agriculture.

CONCLUSIONS

1. *Improved DCGAN for Dataset Expansion:* An image data enhancement algorithm based on improved DCGAN (Deep Convolutional Generative Adversarial Network) was proposed to expand the American *Hyphantria cunea* larvae net curtain dataset. The collected original images were cropped to a resolution of 64×64 to handle the large resolution and complex composition of the larvae net curtain images.
2. *Training for Color Differences:* Images with significant color differences under various conditions were trained separately, ensuring the expanded dataset maintained high image quality.
3. *Algorithm Optimization:* The deconvolution layer was eliminated, and a resize convolution layer was introduced to reduce the checkerboard effect and accelerate training. A dropout layer was added to improve the stability of training. Using the LeakyReLU function instead of the ReLU function avoided neuron necrosis.

4. *Enhanced Neural Network Generalization*: The improved DCGAN network was trained to generate the final expanded dataset. Using this dataset to train existing recognition algorithms significantly improved the generalization ability of the neural network, achieving high recognition accuracy.

ACKNOWLEDGMENTS

Thank you to the reviewers for their patience and professionalism, and for assistance with this work.

Funding

This paper was supported by the National Key Research and Development Program, and the subject number is 2022YFD2001905.

REFERENCES CITED

- De Andrade, A. (2019). "Best practices for convolutional neural networks applied to object recognition in images," *arXiv* 1910.13029. DOI: 10.48550/arXiv.1910.13029
- Ding, J., Li, X., Kang, X., and Gudivada, V. N. (2019). "A case study of the augmentation and evaluation of training data for deep learning," *Journal of Data and Information Quality* 11(4), article 20. DOI: 10.1145/3317573
- Gao, Y., Zhao, Y., Ji, Y., Zhao, D., Wang, C., and Sun, Q. (2020). "A screen location method for treating American *Hyphantria cunea* larvae using convolutional neural network," *Mathematical Problems in Engineering* 2020, article ID 3874546. DOI: 10.1155/2020/3874546
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). "Generative adversarial nets," in: *Advances in Neural Information Processing Systems*, Curran Associates, Montreal, Canada, pp. 2672-2680. DOI: 10.5555/2969033.2969125
- Grant-Jacob, J. A., Praeger, M., Eason, R. W., and Mills, B. (2022). "Generating images of hydrated pollen grains using deep learning," *IOP SciNotes* 3(2), article 024001. DOI: 10.1088/2633-1357/AC6780
- Gu, X., Liu, J., Zou, X., and Kuang, P. (2017). "Using checkerboard rendering and deconvolution to eliminate checkerboard artifacts in images generated by neural networks," in: *Proceedings of the 14th International Computer Conference on Wavelet Active Media Technology and Information Processing*, Chengdu, China, pp. 197-200. DOI: 10.1109/ICCWAMTIP.2017.8301478
- Isola, P., Zhu, J. Y., Zhou, T., and Efros, A. A. (2016). "Image-to-image translation with conditional adversarial networks," *arXiv* 1611.07004. DOI: 10.48550/arXiv.1611.07004
- Karras, T., Aila, T., Laine, S., and Lehtinen, J. (2017). "Progressive growing of GANs for improved quality, stability, and variation," *arXiv* 1710.1196. DOI: 10.48550/arXiv.1710.1196
- Kingma, D., and Ba, J. (2014). "Adam: A method for stochastic optimization," *arXiv* 1412.6980. DOI: 10.48550/arXiv.1412.6980

- Liu, H., Luo, Y., Wen, J., Zhang, Z., Feng, J., and Tao, W. (2006). "Pest risk assessment of *Dendroctonus valens*, *Hyphantria cunea*, and *Apriona swainsoni* in Beijing," *Frontiers of Forestry in China* 1(3), 328-335. DOI: 10.1007/s11461-006-0025-5
- Lopes, A. T., de Aguiar, E., De Souza, A. F., and Oliveira-Santos, T. (2017). "Facial expression recognition with convolutional neural networks: Coping with few data and the training sample order," *Pattern Recognition* 61, 610-628. DOI: 10.1016/j.patcog.2016.07.026
- Radford, A., Metz, L., and Chintala, S. (2015). "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv* 1511.06434. DOI: 10.48550/arXiv.1511.06434
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research* 15(1), 1929-1958. DOI: 10.5555/2627435.2670313
- Sugawara, Y., Shiota, S., and Kiya, H. (2019). "Checkerboard artifacts free convolutional neural networks," *APSIPA Transactions on Signal and Information Processing* 8, 1-9. DOI: 10.1017/ATSIP.2019.2
- Sun, X., Lv, M., Quan, C., and Ren, F. (2017). "Improved facial expression recognition method based on ROI deep convolutional neural network," in: *Proceedings of the 7th International Conference on Affective Computing and Intelligent Interaction*, San Antonio, TX, USA, IEEE Computer Society, pp. 256-261. DOI: 10.1109/ACII.2017.8273609
- Yamamoto, R., Song, E., and Kim, J. M. (2019). "Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with a multi-resolution spectrogram," in: *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, Brighton, UK. DOI: 10.48550/arXiv.1910.11480
- Yang, R., Wang, Y., Xu, Y., and Zhang, M. (2022). "Application of neural network in pixel art creation: Bi-directional conversion between photo and pixel art with GAN base model," in: *Proceedings of the 2nd International Conference on Consumer Electronics and Computer Engineering*, Guangzhou, China, IEEE, pp. 14-16. DOI: 10.1109/ICCECE54139.2022.9712735
- Yang, Y., and Li, C. (2021). "Quantitative analysis of the generalization ability of deep feedforward neural networks," *Journal of Intelligent and Fuzzy Systems* 40(3), 4867-4876. DOI: 10.3233/JIFS-201679
- Yang, Z. Q., Wang, X. Y., Wei, J. R., Qu, H. R., and Qiao, X. R. (2008). "Survey of the native insect natural enemies of *Hyphantria cunea* (Drury) (Lepidoptera: Arctiidae) in China," *Bulletin of Entomological Research* 98(3), 293-302. DOI: 10.1017/S0007485308005609
- Zhou, Z.-H., and Jiang, Y. (2004). "NeC4.5: Neural ensemble based C4.5," *IEEE Transactions on Knowledge and Data Engineering* 16(6), 770-773. DOI: 10.1109/TKDE.2004.11

Article submitted: July 30, 2024; Peer review completed: August 31, 2024; Revised version received: September 26, 2024; Accepted: September 27, 2024; Published: October 16, 2024.

DOI: 10.15376/biores.19.4.9271-9284