Digital Camouflage Generation Based on an Improved CycleGAN Network Model

Leixiang Xia

School of Computer Science and Engineering
Xi'an Technological University
Xi'an, 710021, Shaanxi, China
E-mail: 2576341766@qq.com

Zhiyi Hu

Engineering Design Institute
Army Research Loboratory
Beijing, 100042, China
E-mail: 763757335 @qq.com

Jun Yu

School of Computer Science and Engineering
Xi'an Technological University
Xi'an, 710021, Shaanxi, China
E-mail: yujun@xatu.edu.cn

Yunshan Xie

School of Computer Science and Engineering
Xi'an Technological University
Xi'an, 710021, Shaanxi, China
E-mail: 2398576240@qq.com

Kuncai Jiang

School of Computer Science and Engineering
Xi'an Technological University
Xi'an, 710021, Shaanxi, China
E-mail: 2992165628@qq.com

Abstract—**This paper proposes a digital camouflage generation method based on an improved CycleGAN to produce camouflage patterns with a high degree of fusion with the background and realistic texture details. Firstly, a SE-ResNet network structure is constructed by combining the residual network ResNet with the channel attention mechanism SENet, enabling flexible adjustment of channel weights to effectively extract crucial channel features and enhance the network's perception capability of important information in images. Secondly, a color preservation loss is introduced to improve the adversarial loss function, thereby avoiding training instability and fluctuation in pattern quality. Experimental results demonstrate that the camouflage patterns generated using the proposed method achieve a Structural Similarity Index (SSIM) of 0.77 and a Peak Signal-to-Noise Ratio (PSNR) of 18.9, representing improvements of 0.27 and 3.3, respectively, compared to the original CycleGAN. This method can generate digital camouflage patterns with richer details, textures, and high fusion with the background.**

Keywords-Digital Camouflage; CycleGAN; Channel Attention Mechanism; Residual Network

I.   INTRODUCTION

With the rapid development of military science and technology, military reconnaissance methods are exhibiting trends of diversification, intelligence, and high-precision resolution capabilities, which have significantly improved battlefield information acquisition capabilities, posing greater challenges to military camouflage technology [1]. Against this backdrop, digital camouflage technology, as a part of military camouflage techniques, plays an important role in military reconnaissance countermeasures.

To cope with diversified military reconnaissance techniques, domestic scholars have conducted in-depth research on digital camouflage technology. Cai Yunxiang [2] and colleagues utilized fractal dimension estimation based on fractal Brownian motion and a layer-by-layer fuzzy C-means clustering algorithm with a pyramid structure to extract texture and primary color features from background images, achieving the generation of digital camouflage patterns.

However, this digital camouflage the generation method is relatively cumbersome and requires sufficient practical experience to produce high-quality camouflage. To automatically generate digital camouflage patterns, Yang Wuxia [2] et al. extract the main colors of the background using the K-means algorithm on the color grayscale histogram based on the target background image to generate digital camouflage. Jia Qi [3] et al. used Markov random fields and pyramid models to build a digital camouflage design system, initially achieving automation in digital camouflage design. With the development of deep learning technology, Teng Xu [3] et al. combined cyclic adversarial networks with densely connected convolutional networks to quickly and automatically generate digital camouflage patterns, but the generated patterns still lacked richness in details and textures.

To generate camouflage patterns that blend seamlessly with the background and exhibit realistic texture details, this paper enhances a generative adversarial network model based on CycleGAN [4] (Cycle-Consistent Generative Adversarial Network). Building upon the traditional framework of Cycle-Consistent Generative Adversarial Networks, this study introduces a channel attention mechanism into the existing residual network to extract image features. Moreover, it incorporates a color loss function and enhances the adversarial loss function. These modifications effectively resolve issues related to the fine details and textures of the generated patterns.

II. CYCLE-CONSISTENT GENERATIVE ADVERSARIAL NETWORK

A. Network Architecture

CycleGAN is an unsupervised generative adversarial network designed for translating images from one domain to another, such as from class X to class Y, without the need for paired training data. The model's key innovation lies in enforcing bidirectional image translation through a cycle consistency loss. This loss ensures that a translated image can be reconstructed into its original form within the same domain, maintaining fidelity to the original image throughout the translation process.

A complete CycleGAN model consists of two sets of generators and discriminators, each targeting a specific translation direction. Specifically:

The first set includes generator G and discriminator D_Y. Generator G is responsible for translating images from class X to class Y, while discriminator D_Y distinguishes between generated Y-class images and real Y-class images.

The second set includes generator F and discriminator D_X. Generator F translates images from class Y back to class X, while discriminator D_X distinguishes between generated X-class images and real X-class images [5].

During the training process, two sets of generators and discriminators are trained alternately. By optimizing the adversarial loss and cycle consistency loss, the model can gradually learn the mapping between the two image types and generate images that conform to the characteristics and distribution of the target image type.As shown in Figure 1.
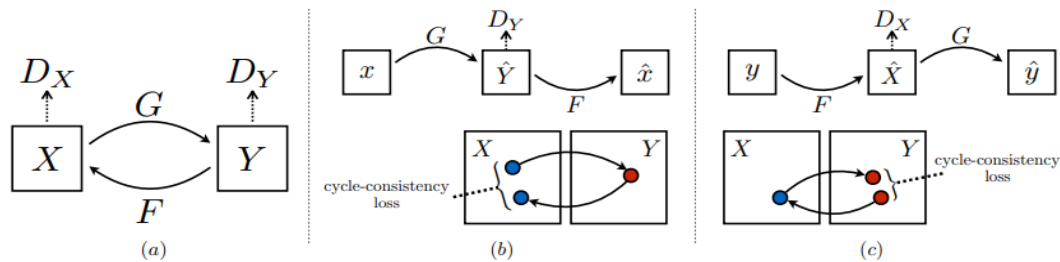

Figure 1.   Structure of CycleGAN

ResNet (Residual Networks) is a deep learning network architecture that addresses the gradient issues in training deep networks by introducing residual blocks and skip connections, thereby enhancing network performance. CycleGAN utilizes multiple basic residual blocks in its generators to construct the network [6]. While ResNet solves the gradient vanishing problem through residual blocks, which enhances the efficiency of data transmission and mitigates the issues of gradient diffusion and network degradation, it does not significantly improve the quality of image generation.

B. Loss function

The loss functions of the CycleGAN network comprise three types: the adversarial loss (referred to as GAN loss in this context), the cycle consistency loss (referred to as cycle loss), and the identity mapping loss (referred to as identity loss). The adversarial loss aims to promote the adversarial learning between the generator and the discriminator, encouraging the generator to produce more realistic samples. CycleGAN employs two adversarial losses, taking the generator G and the discriminator D_Y as an example, which can be expressed as in formula (1); the cycle consistency loss enables translation between two different domains, and its formula is given in (2); the identity mapping loss is primarily designed to train the network's recognition capabilities, and its formula is presented in (3).

$$L_{GAN}\left(G, D_Y, X, Y\right) = E_{y \sim p_{\text{data}}(y)}\left[\lg D_Y(y)\right] + E_{x \sim p_{\text{data}}(x)}\left[\lg\left(1 - D_Y(G(x))\right)\right] \qquad (1)$$

$$L_{cyc}(G, F) = E_{x \sim p_{\text{data}}(x)}\left[\| F(G(x)) - x\|_1\right] + E_{y \sim p_{\text{data}}(y)}\left[\| G(F(y)) - y\|_1\right] \qquad (2)$$

$$L_{\text{identity}}(G, F) = E_{y \sim P_{\text{data}}(y)}\left[\| G(y) - y\|_1\right] + E_{x \sim P_{\text{data}}(x)}\left[\| F(x) - x\|_1\right] \qquad (3)$$

Wherein, $p_{\text{data}}(x)$ represents the sample distribution of type X, and $p_{\text{data}}(y)$ represents the sample distribution of type Y. x represents images of type X, and y represents images of type Y[7]. JS (Jensen-Shannon) divergence is a parameter used in traditional GAN loss to measure the difference between the distributions of real images and generated images. When the two distributions do not overlap, the gradient of the JS divergence may become very small or zero, leading to issues such as gradient vanishing and, in extreme cases, gradient explosion

III. IMPROVED CYCLEGAN NETWORK MODEL

The image generation quality of traditional CycleGAN network models is not high, as the simple stacking of residual blocks in the generator introduces excessive redundant channels. These redundant channels are not conducive to extracting finer-grained information features, thus affecting the quality of image generation. To address this issue, this paper introduces the channel attention mechanism of SENet to optimize the residual blocks and improves the loss function, aiming to enhance the quality of image generation.

A. Improvement of network structure

In the original CycleGAN architecture, convolutional layers and pooling layers are typically employed to extract image features. However, the feature information extracted using this method is often overly redundant. To address this, a channel attention mechanism is introduced. The channel attention mechanism calculates the importance of each channel, enabling the network to focus on more significant channels and filter out redundant information. This paper utilizes the classic channel attention method, SENet [8] (Squeeze-and-Excitation Networks), and combines the SENet model with ResNet to generate SE-ResNet modules, thereby enhancing the network's feature extraction capabilities.
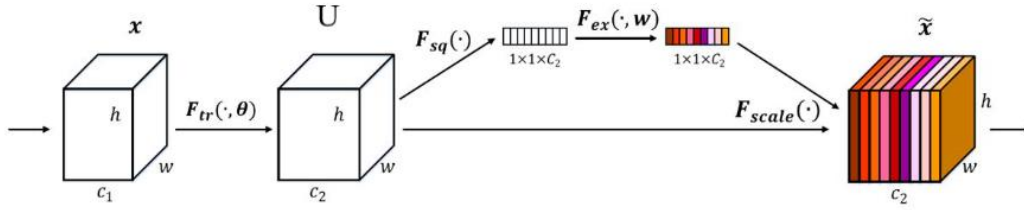
Figure 2.   Schematic diagram of SENet

1) SENet Channel Attention Mechanism

The SENet module consists of two core operations: Squeeze and Excitation. The Squeeze operation compresses the feature channels through global average pooling. The Excitation operation learns the dependencies between channels through fully connected layers, generating weights for each channel, thus achieving the recalibration of feature channels. As shown in Figure 2, the input data X undergoes a convolutional operation $F_{tr}(\cdot, \theta)$ to obtain data U. Subsequently, it passes through the Squeeze operation $F_{sq}(\cdot)$, the Excitation operation $F_{ex}(\cdot, W)$, and the Scale operation $F_{scale}(\cdot)$, where C represents the number of channels, and H and W represent the height and width of a single channel. The specific implementation steps are as follows:

Step One: Squeeze（ F_{sq} ）: Global average pooling is used to compress the 2D features of each channel, with dimensions H×W, into a one-dimensional numerical value, generating a feature description along the channel dimension.

Step Two: Excitation（ F_{ex} ）: In SENet, there is a fully connected layer that takes the feature vector obtained from the previous step as input. This fully connected layer has an intermediate hidden layer to learn the dependencies between channels. Through an activation function (such as ReLU) and a parameterized scaling operation, the fully connected layer can learn the weight of each channel. These weights represent the importance of information in different channels [9].

Step Three: Scale（ F_{scale} ）: The learned channel weights are applied to the input feature map. Each channel's weight is multiplied by the corresponding channel's feature map, reweighting the feature map to enhance the representation of important channels and suppress the representation of less important channels.

During the above process, the Squeeze operation reduces the dimensions of the channel from C×H×W to C×1×1 through global pooling. The Excitation operation utilizes a fully connected layer to generate a weight vector for each channel's feature. Finally, the Scale operation multiplies the output of the Excitation operation with the input feature map. Through these operations, the weight vectors are assigned to each channel of the feature map, weighting the features of different channels accordingly.

2) SE-Resnet Module

SE blocks are plug-and-play and can be integrated into ResNet, resulting in SE-ResNet. The structure of SE-ResNet is illustrated in Figure 3.
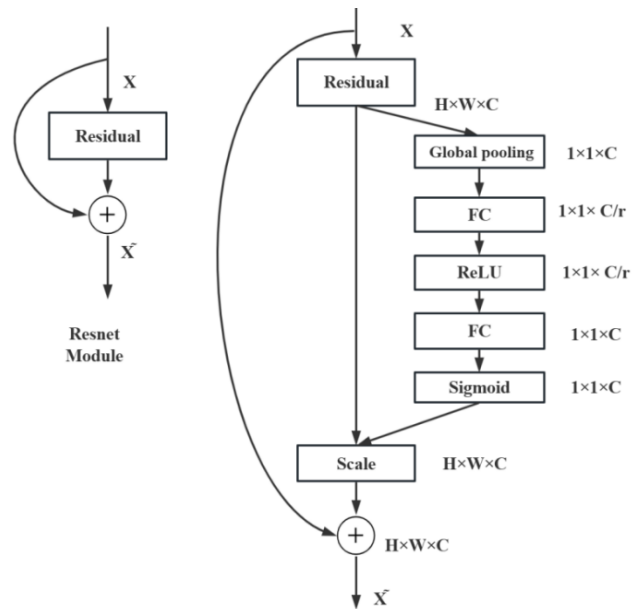

Figure 3.   Diagram of the SE-Resnet module

SE-ResNet's basic structure is similar to that of traditional Residual Networks, composed of multiple residual blocks. The key difference lies in the addition of SE modules within each residual block to extract attention information. As shown in Figure 4, by introducing SE modules, SE-ResNet can more accurately mine feature information from images and enhance the network's utilization of features from different channels.
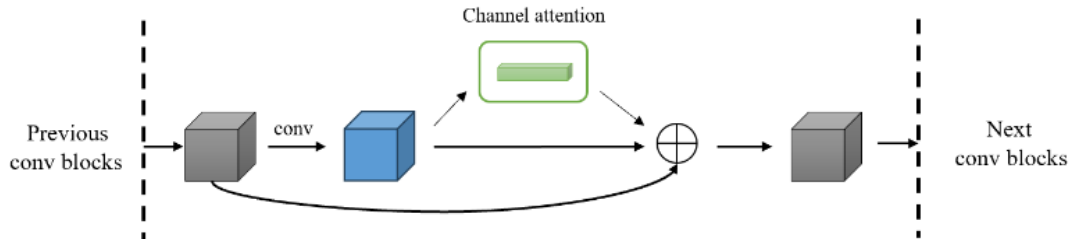

Figure 4. Schematic diagram of the attention mechanism for joining channels

The workflow of SE-ResNet is as follows:

a). Adjust the input image to a uniform size and pass the 3-channel image through a 7×7 convolution kernel to transform it into a 64-channel image. Subsequently, use 3×3 convolutions to extract features and generate feature maps.

b). Through a 9-layer residual structure, the image features are transformed from the source type to the target type. During this process, the input feature information is propagated and output.

c). Utilize deconvolution operations to restore the high-dimensional feature maps from the input, in order to reconstruct the surface features of the image.

d). The final layer performs a convolution operation to modify the dimensionality of the output from the previous step. The network structure of this generator is illustrated in Figure 5.
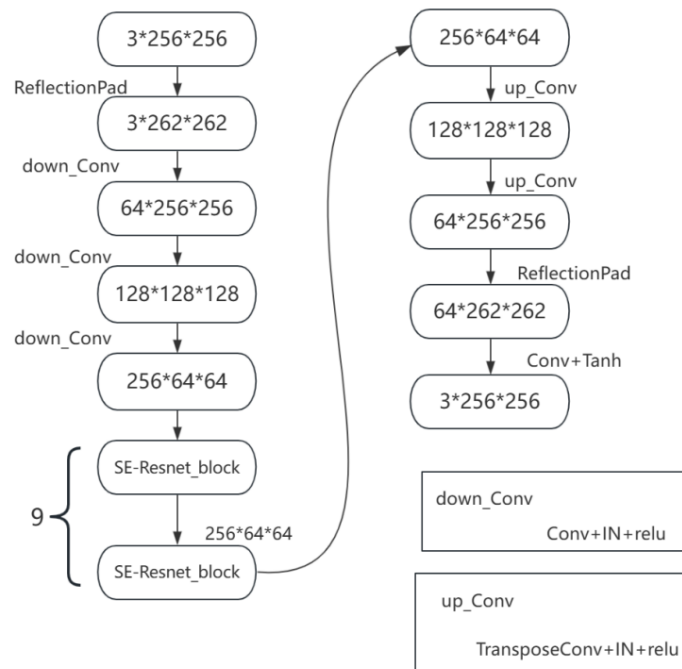

Figure 5. Improving the Generator Network Structure

B. Improvements to the loss function

In this paper, the loss function from WGAN-GP (Wasserstein Generative Adversarial Networks-Gradient Penalty) is adopted instead of the original GAN loss. WGAN-GP focuses more on matching the overall distribution, better preserving global consistency, and making the generated samples more natural. Additionally, a color preservation loss function is also included to ensure consistency in color features before and after image transformation.

1) Improving the Adversarial Loss Function

To avoid gradient issues caused by the JS divergence, this paper uses the WGAN-GP loss instead of the original GAN loss. The Wasserstein distance (also known as the EM distance) in WGAN-GP loss exhibits superior smoothness and smoother curve changes. Even when the generated image distribution does not overlap with the real image distribution, the discriminator D can still accurately calculate the actual distance between them, effectively evaluating the quality of the images produced by the generator G. This allows the use of the backpropagation algorithm to optimize the discriminator D, theoretically overcoming the difficulties of gradient vanishing and gradient exploding.

The meaning of EM distance is explained in equation (4).

$$W\left(p_{\text{data}}, p_{\text{model}}\right) = \sup_{\|f\|_L \leq 1} E_{x \sim p_{\text{data}}}[f(x)] - E_{x \sim p_{\text{model}}}[f(x)] \tag{4}$$

In this context, p_{data} represents the distribution of real data, while p_{model} represents the model distribution output by the generator. $f(x)$ is a Lipschitz continuous function.

To achieve gradient penalty, a gradient penalty term is introduced on the basis of the EM distance. By penalizing the gradient of the discriminator's output for samples, the calculation is performed as shown in formula (5). During each parameter update, samples are taken from the linear interpolation points between real samples and generated samples. The gradients of the discriminator corresponding to these points are then calculated. By applying a penalty term to these gradient norms, the gradient norms are restricted to a reasonable range. Where $\hat{x}$ is the linear interpolation between a real sample x and a generated sample $G(z)$, and λ is the weight coefficient of the gradient penalty.

$$L_{\text{GP}} = \lambda E_{\hat{x} \sim P_{\hat{x}}}\left[\left(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1\right)^2\right] \tag{5}$$

The objective function in the WGAN-GP model is shown in equation (6).

$$L = \underbrace{E_{x \sim P_g}[D(\tilde{x})] - E_{x \sim P_r}[D(x)]}_{\text{Original\_critic\_loss}} + \underbrace{\lambda E_{x \sim P_{\hat{x}}}[\left(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1\right)^2]}_{\text{gradient\_penalty}} \tag{6}$$

In this paper, by reassessing the weight parameters, the images generated from random noise are taken as penalty terms and input into the network, and gradient clipping is applied to address the issue of training instability when using the EM distance. The improved objective function is shown in formula (7), where $P_{\text{penalty}}(x)$ serves as a penalty term positioned between $P_{data}(x)$ and $P_z(z)$, and $P_z(z)$ aims to continuously approach $P_{data}(x)$.

$$V(D,G) = \max_D \left\{ E_{x \sim P_{\text{data}}(x)}[(D(x))] - E_{z \sim P_z(z)}[D(G(z))] - \lambda_1 E_{x \sim \text{penalty}}\left[\left(\|\nabla_x D(x)\|_2 - 1\right)^2\right] \right\} \tag{7}$$

The GAN loss typically encourages the generated samples to be as similar to the real samples as possible on a pixel-level basis. However, this pixel-level similarity can sometimes lead to inconsistencies in the details of the generated images. WGAN-GP (Wasserstein GAN with Gradient Penalty) incorporates a gradient penalty term that constrains the gradient norm of the discriminator, preventing gradient explosion and vanishing issues. This gradient penalty term also enhances the stability of the network and allows it to generate more realistic and clearer samples.

2) Add color retention loss function

To enhance the similarity between the generated images and the original images, a color preservation loss is integrated into the cycle consistency loss. This color preservation loss function is used to calculate the distance between the distributions of real samples and generated samples. Compared to other loss functions, the color preservation loss is relatively less affected by outliers, making it more robust to noise or abnormal situations. Through regularization using the color preservation loss function, the model can be encouraged to select features with fewer non-zero weights, which improves the model's generalization ability and interpretability. The formula for the color preservation loss is shown in equation (8).The formula for the total loss function is shown in equation (9).

$$L_{\text{color}}(G,F) = E_{x-P_{\text{data}}(x)}\left[\| x - G(x)\|_1\right] + E_{x-P_{\text{data}}(x)}\left[\| x - F(G(x))\|_1\right] \tag{8}$$

$$L_{\text{total}}(G,F,D_X,D_Y) = V(D,G) + \lambda L_{\text{cyc}}(G,F) + \mu L_{\text{identity}}(G,F) + L_{\text{color}}(G,F) \tag{9}$$

IV. EXPERIMENTAL RESULTS AND ANALYSIS

The experimental platform used Ubuntu 18.04 system and employed a graphics processing unit (GPU) to accelerate the training speed. The deep learning framework adopted in this experiment is Pytorch 1.4.0. The datasets used in this paper are from UCMerced_LandUse [10], Places365 [11], and others, with pre-training conducted on the public dataset monet2photo.

In this experiment, the images in the dataset were preprocessed by eliminating unqualified images, removing poor learning samples, excluding overly similar learning samples, and fixing the image sizes. Subsequently, data augmentation techniques such as scaling up, random rotation, random flipping, and adding noise were applied to expand the dataset to 4,200 images.

To evaluate the experimental results, Peak Signal to Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) were used in this paper.

Given two image sets I and K of dimensions m×n, for any i and j, PSNR can be defined as equation (10).

$$PSNR = 10\lg\left[\frac{(2^n-1)^2}{MSE}\right] \tag{10}$$

In equation (10), MSE represents the Mean Squared Error, and its expression is given in equation (11).

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left[I(i,j) - K(i,j)\right]^2 \tag{11}$$

The higher PSNR value is, the lower the degree of image distortion and the better the quality of the image.

Luminance, Contrast, and Structure are the three essential elements that constitute an image. Their respective definitions are given in equations (12), (13), and (14):

$$l(x,y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \tag{12}$$

$$c(x, y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \qquad (13)$$

$$s(x, y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \qquad (14)$$

The SSIM algorithm consolidates the calculation formulas for luminance, contrast, and structural similarity to obtain the overall calculation formula for image similarity (12).

$$\text{SSIM}(x, y) = \left[l(x, y)^{a}, c(x, y)^{\beta}, s(x, y)^{\gamma} \right] \qquad (15)$$

SSIM values range from -1 to 1, with SSIM = -1 indicating complete dissimilarity between two images, and SSIM = 1 indicating their absolute identity.

This paper conducted ablation experiments and comparative experiments. In the ablation experiments, three improvements were made to the model with the addition of the color preservation loss function: CycleGAN + improved loss function, CycleGAN + SENet, and CycleGAN + SENet + improved loss function. These experiments aimed to validate the efficacy of the proposed approach. During comparative analysis, the proposed model was assessed alongside two image generation methods, GAN and DRIT, to gauge the image generation quality across different models.

A. Ablation experiments

In the ablation experiments, initially, the color preservation loss function was incorporated, with the outcomes presented in Figure 6. It is evident that upon integrating the color preservation loss function, the colors of the generated images more closely resemble those of the input images.
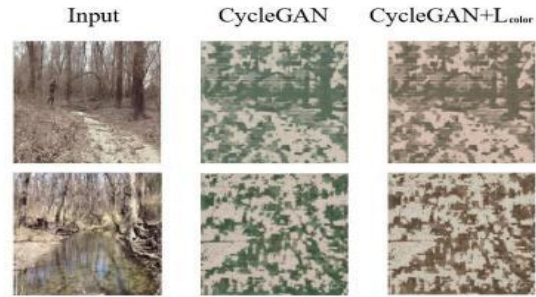
Figure 6.   Comparison of Results before and after Adding Color Preservation Loss

1) CycleGAN + Improved Loss Function

The CycleGAN was combined with the improved loss function. Three models were trained using the original GAN loss, the WGAN objective function, and the WGAN-GP objective function. Figure 7 shows the changes in the loss curves during the training process for the three different loss functions: the original GAN loss, WGAN, and WGAN-GP.
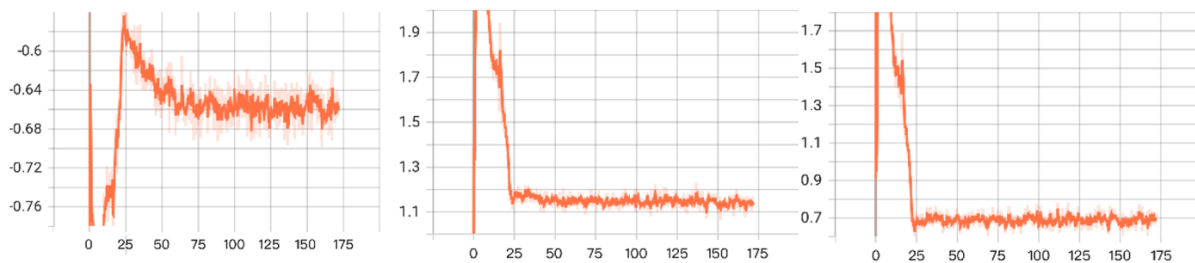

Figure 7.   Loss Curve Variation Chart

As can be seen from Figure 7, compared to the loss function graphs of the original GAN and WGAN, the WGAN-GP loss adopted in this paper achieves a lower loss value when reaching the Nash equilibrium. This phenomenon indicates that the images generated by the WGAN-GP network have a higher quality.

2) CycleGAN + SENet

The CycleGAN was combined with SENet. The curve comparisons of the various loss functions between the original CycleGAN model and the improved CycleGAN model in this paper are shown in Figure 8 and Figure 9, where the horizontal axis represents the number

of epochs (iterations), and the vertical axis represents the loss value.
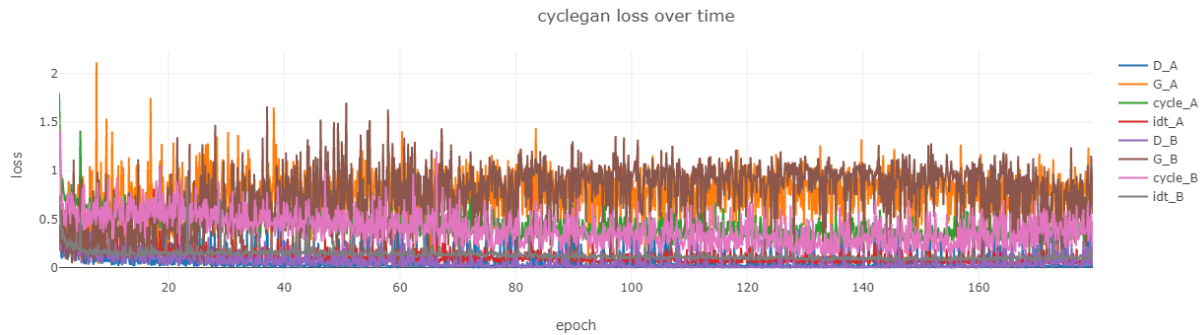

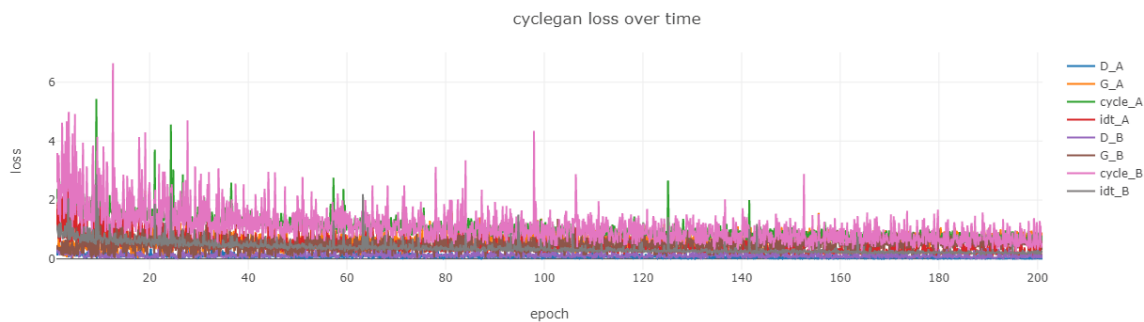Figure 8.   Loss Function Change Trend for the Original CycleGAN Model

Figure 9.   Loss Function Change Trend for the Improved CycleGAN Model

As can be seen from Figures 8 and 9, the loss function of the original CycleGAN model has not yet shown a converging downward trend when reaching 160 epochs, while the loss function of the improved model begins to show a downward trend starting from 40 epochs and converges at 200 epochs.

The ablation experiments were conducted to verify the effects of introducing the SENet network structure and the improved loss function. Using the method of controlled variables, four sets of experiments were performed by adding different improved modules. The final evaluation index data are shown in Table Ⅰ.

TABLE I.   EVALUATION INDEX TABLE OF ABLATION EXPERIMENTS

Model	SSIM	PSNR
CycleGAN	0.50	15.6
CycleGAN+SENet	0.61	16.7
CycleGAN+improved loss function	0.68	15.9
CycleGAN+SENet+improved loss function.	0.77	18.9

According to Table I, the baseline CycleGAN model exhibited the poorest performance. Incorporating the SENet channel attention mechanism led to an SSIM increase of 0.11 and a PSNR increase of 1.1. Compared to the original CycleGAN, the version with enhanced loss functions showed SSIM and PSNR improvements

of 0.18 and 0.3, respectively. By combining these approaches, SSIM reached 0.77, while PSNR achieved 18.9. The experimental results indicate that both methods, when used independently, can enhance the pattern details and texture quality. By combining these two methods in this paper, a more ideal pattern generation effect can be achieved.

B. Comparative experiments

In this paper, two image generation methods, GAN (Generative Adversarial Network) and DRIT (Diversity Regularized Image-to-Image Translation), were selected to conduct comparative experiments and evaluate the effectiveness of image generation. GAN is a form of generative adversarial network that generates images by allowing two neural networks (generator and discriminator) to learn from each other in an adversarial manner. DRIT is a model that can achieve image translation with different styles.
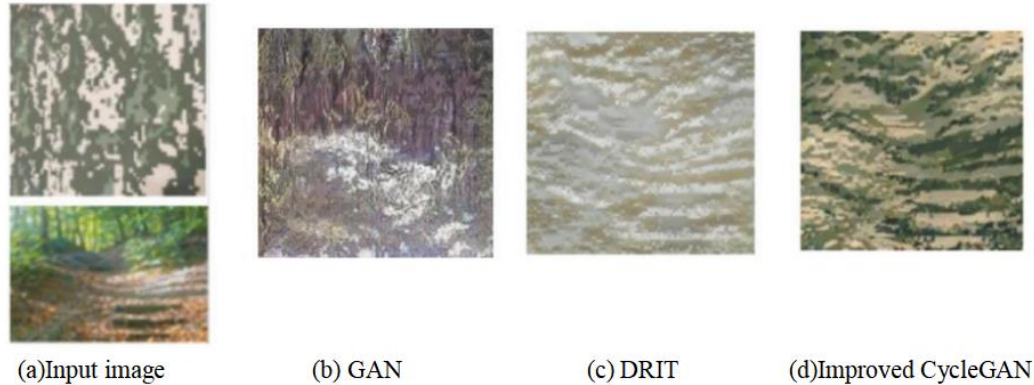

Figure 10. Comparison of images generated by different models

As shown in Figure 10, (a) represents the target-type image and the original image; (b) is the image generated by the GAN model, which performs poorly in terms of color and texture, significantly differing from the original image; (c) depicts the image produced by the DRIT model, which although retains some texture features, differs greatly in color features and appears blurry. (d) is the image generated by the CycleGAN model, which, compared to DRIT, not only preserves texture and color features but also exhibits richer details.

To evaluate the quality of the generated camouflage patterns, two metrics, SSIM and PSNR, were used, and the final evaluation results are presented in Table Ⅱ.

TABLE II.   COMPARATIVE EXPERIMENTAL EVALUATION SCORE TABLE

Generative model	SSIM	PSNR
GAN	0.19	13.5
DRIT	0.48	14.8
CycleGAN+SENet+improved loss function	0.77	18.9

As seen from Table Ⅱ, the SSIM value of our proposed method is 0.58 higher than that of the GAN model and 0.29 higher than the DRIT model. Similarly, the PSNR value of our method is 5.4 higher than the GAN model and 4.1 higher than the DRIT model. Combining the results from Table Ⅱ and Figure 10, it can be concluded that our proposed method achieves superior performance, both in terms of visual perception and the SSIM and PSNR scores.

V.   CONCLUSIONS

This paper investigates a digital camouflage generation method based on an improved CycleGAN. Firstly, the combination of residual networks with channel attention mechanisms is explored, which enables the network to focus more on important channel features. Secondly, the use of WGAN-GP loss instead of the original GAN Loss improves the adversarial loss function, avoiding the common problem of unstable outputs in traditional generative adversarial networks, thus generating more realistic and clearer samples. Lastly, a color preservation loss function is added

to prevent the generator from autonomously modifying the image's hue before and after image translation, avoiding color changes. Experimental results demonstrate that compared to other methods, the proposed approach produces camouflage patterns with more realistic texture details and higher fusion with the background, confirming the effectiveness of this method.

REFERENCES

[1] Yang Di. Evaluation Method of Camouflage Effect in Dynamic 3D Scenes [Dissertation]. Jiangnan University, 2024. DOI: 10.27169/d.cnki.gwqgu.2023.001844.

[2] Teng Xu. Research on Digital Camouflage Generation Based on Generative Adversarial Networks [Dissertation]. Southwest University of Science and Technology, 2021. DOI: 10.27415/d.cnki.gxngc.2021.000342.

[3] Teng Xu, Zhang Hui, Yang Chunming, Zhao Xujian, Li Bo. Digital Camouflage Disguise Generation Method Based on Cycle-Consistent Adversarial Networks [J]. Computer Applications, 2020, 40(02): 566-570.

[4] Luo Li. Research and Application of Adversarial Sample Generation Method Based on CycleGAN [Dissertation]. Guilin University of Electronic Technology, 2023. DOI: 10.27049/d.cnki.ggldc.2023.001423.

[5] Liu, Zunyang et al. "Background dominant colors extraction method based on color image quick fuzzy c-means clustering algorithm." Defence Technology (2020): n. pag.

[6] Lü Lin. Research and Implementation of Image Translation Algorithm Based on Cycle Generative Adversarial Networks [D]. Shandong University, 2024. DOI: 10.27272/d.cnki.gshdu.2023.002296.

[7] Fu JinHuang Shan. Research on Image Denoising Based on Improved Cycle Generative Adversarial Networks [J]. Computer Engineering and Applications, 2023, 59(17): 178-186.

[8] JHU J, SHEN L, ALBANE S, et al. Squeeze-and-excitation networksL [J]. EEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 42(8) 2011-2023.

[9] Li Yongjiang, Yang Zhigang. Image Translation Method Based on Improved CycleGAN [J]. Journal of Harbin University of Commerce (Natural Sciences Edition), 2023, 39(03): 281-286. DOI: 10.19492/j.cnki.1672-0946.2023.03.005.

[10] Li Yongjiang, Yang Zhigang. Image Translation Method Based on Improved CycleGAN [J]. Journal of Harbin University of Commerce (Natural Science Edition), 2023, 39(03): 281-286. DOI: 10.19492/j.cnki.1672-0946.2023.03.005.

[11] Li .Research on Artistic Image Style Transfer Method Based on Generative Adversarial Networks [D]. Guangdong University of Technology, 2022. DOI: 10.27029/d.cnki.ggdgu.2021.000308.