IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 24 May 2024, accepted 14 August 2024, date of publication 2 September 2024, date of current version 10 September 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3452677

RESEARCH ARTICLE

Image Style Conversion Model Design Based on Generative Adversarial Networks

KE GONG[1] AND ZHU ZHEN[2]
[1]Higher Vocational College, Jilin Provincial Institute of Education, Changchun 130022, China
[2]Art Department, Jilin Police College, Changchun 130117, China

Corresponding author: Ke Gong (guessgong_k@163.com)

ABSTRACT The purpose of image style conversion is to transfer the style of one image to another, so that the target image retains the original content and the style of the reference image. An image style conversion technique based on generative adversarial network is proposed. This study innovatively adopts dual synthesizer and dual discriminator structure to improve the quality and efficiency of style conversion, and introduces extended convolution to enhance feature extraction. Combined with a well-designed loss function to optimize the style conversion process, a convolutional module reconstruction generator network including linear computation is added. The experimental results showed that in the training time test, the research method maintained a training time of less than 97 seconds when the number of input style types increased to 18. When conducting image style loss testing, the research method found that the image style loss value was lower compared to other techniques when the input size was 1080p and the pixel count was 10M. In the analysis of pixel loss during image style conversion, the research method shows that the pixel loss after processing virtual images is only 3.5k out of 10M, which hardly affects the expression of image content. The designed image style conversion model can accomplish the task of image style conversion with high quality and high efficiency.

INDEX TERMS Generative adversarial networks, image processing, linear calculation, convolution, image style.

I. INTRODUCTION

With the rapid development of computer technology, image processing technology has been widely applied in many fields. Among them, image style conversion, as an important image processing task, aims to transfer the style features of one image to another, thereby achieving modification and optimization of image style [1], [2]. The traditional image style conversion methods mainly rely on image processing techniques and machine learning algorithms. However, these methods often require a large amount of manual involvement and parameter adjustment. Meanwhile, its effectiveness is poor when dealing with complex scenes. Balancing the original image content and target image style in image style conversion is a challenging issue [3], [4]. With the increase of data size, traditional image style conversion models face problems such as insufficient computing resources and long training time. Some models may cause distortion in the generated image during image style conversion, making it difficult to perfectly integrate the original content and target style [5]. Some advanced methods require a large amount of computing resources and long training time, which is not practical enough. Generative Adversarial Network (GAN) is an adversarial learning framework composed of a generator and a discriminator. Through continuous iterative training, the generator can generate increasingly realistic images, while the discriminator can more accurately distinguish between generated images and real images. In image style conversion tasks, GANs can effectively capture the style features of the input image and transfer them to the output image [6]. In this context, the research attempts to propose an image style conversion model based on GAN, and then optimize the model according to the characteristics of image style conversion. It is expected to design an innovative image style conversion technology, providing feasible technical references for the image processing industry.

The associate editor coordinating the review of this manuscript and approving it for publication was Joewono Widjaja.

The main contributions of the research are as follows: 1. The adversarial network structure of twin synthesizer and double discriminator is innovatively adopted in the research, which effectively improves the quality and efficiency of style conversion. 2. Expansion convolution is introduced into generator network to enhance feature extraction capability while avoiding large increase in the number of model parameters. 3. A compound loss function including adversarial loss, cyclic consistent loss and edge consistent loss is designed to optimize the style transfer process. 4. The attention mechanism is added to the generator network to improve the model's ability to recognize and transform key features in images.

The research mainly consists of four parts. The first part discusses and summarizes the relevant research results on image style conversion and GAN. The second part mainly designs the image style conversion model based on GAN, and elaborates on the optimization method involved. The third part analyzes the effectiveness of the research method. The final part summarizes the paper.

II. RELATED WORKS

Different image styles can present different image information content. With the development of image processing technology, more scholars have realized the importance of image style conversion technology. Some scholars have conducted relevant research on image style conversion techniques. Yang et al. proposed an unsupervised continuous kernel transformation method for the style conversion of X-ray computed tomography images. The network was subjected to adaptive strength normalization, converting images along the interpolation path between two kernel domains. The experimental results showed that the proposed method effectively performed image style conversion [7]. Jang et al. proposed a StyleCarri-based method to address the style conversion in comic images. Shape exaggeration blocks were used to modulate the energy efficiency of rough feature maps. Layer mixing styles were used to finely exchange photo styles, and then generated detailed style results. The experimental results showed that the proposed method had good realism for comic images [8]. Li et al. proposed a transfer learning method to address the style conversion in road surface images. The framework was constructed using data transmission and model transmission. Image fusion was used to synthesize the labeled data of the new scene, and domain adaptation was adopted to complete feature transfer. The experimental results showed that the proposed method had good model accuracy [9]. Gal et al. proposed a wavelet-based method for style conversion in image editing. The potential representation of frequency perception was strengthened. Spectral deviation extracted high-frequency content to reduce input of image structures that couldn't be learned. The experimental results showed that the proposed method effectively completed image style conversion [10]. Li et al. proposed a deep learning method for hyper-graph style conversion in industrial design. A deep Convolutional Neural Network (CNN) was used to establish an emotion recognition model and generate images with emotional preferences. The experimental results indicated that the research method effectively provided scheme references for industrial designers [11].

Some scholars have conducted relevant research on GAN. Wang et al. proposed a GAN method for movie rendering style conversion. Edge features and self attention were fused, and the edge feature extraction network was inserted into the model for edge extraction of the original image. Then, a perceptual loss was used for network optimization. The experimental results showed that the proposed method had good rendering style conversion effects [12]. Gao et al. proposed a GAN technology for automatic detection of epilepsy. The training set for the seizure period data was balanced. An one-dimensional CNN was used to process the signal, reducing the training parameters for deep structures. The experimental results indicated that the proposed method had good detection performance [13]. Daihong et al. proposed a GAN method for solving image super-resolution. Multi-scale pyramid modules were used to extract high-frequency information features. A bi-cubic interpolation was used for high-resolution image reconstruction, and mean square error was added to reconstruct the loss function. The experimental results showed that the proposed method obtained better quality super-resolution results [14]. Kench and Cooper proposed a technique based on GAN for automatic generation of three-dimensional images. A single representative two-dimensional image was used for three-dimensional generation, and the unified information density concept was adopted to ensure the quality of nodes. The experimental results showed that the proposed method had faster automatic generation efficiency [15]. Lei et al. proposed a method based on GAN for structural health monitoring. The deep convolutional network and GAN were combined to establish realistic assumptions about possible lost signals. The training generator extracted features from the data set and used adversarial loss to process high and low frequency features. The experimental results showed that the proposed method had good data processing accuracy [16]. Maeda et al. proposed a method based on GAN for road damage detection. The Poisson mixing technology and growth were combined to manually generate road damage training data and distinguish the authenticity of the images. The experimental results showed that the proposed method could accurately detect road damage [17].

In summary, GAN has been applied in many fields. It has been confirmed that it can perform image processing, but the research on image style conversion is still relatively limited. In view of this, the study proposes an image style conversion model based on GAN, aiming to provide more feasible technical references for image processing.

III. DESIGN OF IMAGE STYLE CONVERSION MODEL BASED ON GENERATIVE ADVERSARIAL NETWORKS

The image style model can provide rich content variation for fields such as image processing and design by transforming images from one style to another. This section focuses

on the technical means used in the image style conversion model.

A. AN IMAGE STYLE CONVERSION MODEL BASED ON DUAL GENERATORS AND DISCRIMINATORS FOR GENERATIVE ADVERSARIAL NETWORKS

Image style conversion refers to converting the style features of an image to a different style from the original image, while preserving the main information of the image during the conversion process [18], [19]. Image style conversion technology is mainly applied in art and design, entertainment production, and special image detection [20], [21]. Style conversion involves rich image processing and computation, which may result in slower computation speed, especially when processing high-resolution images [22], [23]. Preserving the main information of the image while fully expressing the style image features during the conversion is a challenge [24], [25]. Over-emphasizing style may lead to loss of content information, while retaining content too much may result in unclear style [26]. GAN can automatically learn feature representations of content and style images, capture more complex and rich image features, and generate high-quality conversion results. On the basis of generating adversarial networks, an image style conversion model is established. The GAN consists of a generator and a discriminator, where the generator is responsible for generating images, and the discriminator is responsible for judging the authenticity of the image. The main structure of the GAN is shown in Figure 1.

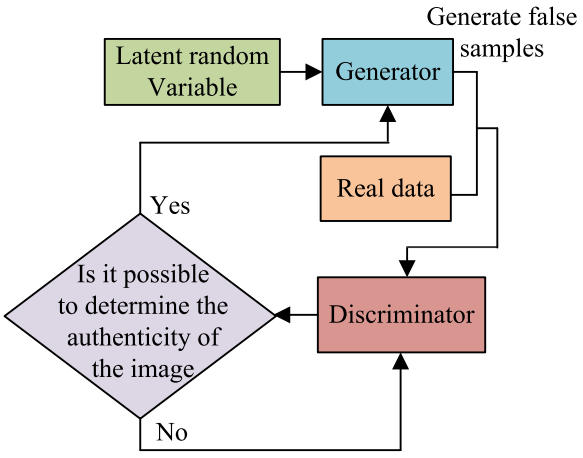

FIGURE 1. The main structure of generative adversarial networks.

In Figure 1, potential random variables are input into the generator. The generator processes the input data, generates corresponding fake data samples, and attempts to deceive the discriminator by inputting them into the discriminator. Meanwhile, real data samples are input into the discriminator and then compared with the data output by the generator. The discriminator determines the authenticity of the data generated by the generator. It sends the judgment results back to the generator and discriminator for training. This operation loops multiple times until Nash equilibrium is reached and the loop ends. However, the commonly used U-net generator structure may lose key information during the training, making it difficult for a single discriminator to effectively control local and global structures when dealing with large-scale image style conversions. To improve the performance and applicability of style conversion models, the study introduces dual generators and discriminators to reconstruct GAN. The GAN structure of dual generators and discriminators is shown in Figure 2.

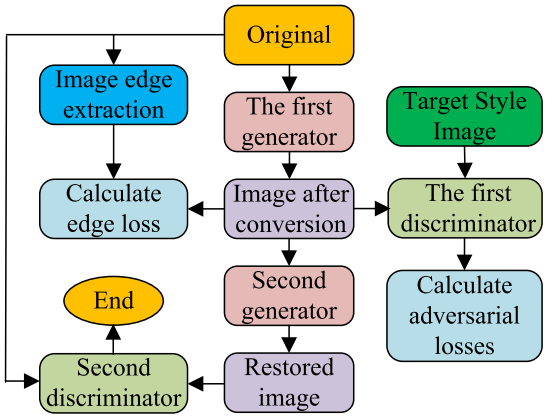

FIGURE 2. Dual generators and discriminators generate adversarial network structures.

In Figure 2, the GAN with dual generators and discriminators converts the image style to the target style through the first generator after inputting the original image. The first conversion result and the original image are subjected to edge extraction to calculate the edge loss value. The first discriminator compares the first conversion result with the target style, and outputs adversarial loss. The second discriminator converts the image style to the original image based on the first conversion result. The second discriminator compares the second style conversion with the original image and outputs adversarial loss. Specifically, the first generator is responsible for converting the input original image into the target style image, while the second generator converts to the original image style based on the first conversion. This bi-directional conversion strategy helps to better balance retention and conversion of content and style. During model training, the discriminator and generator are trained using the obtained loss values. The two generators have the same structure, with a symmetrical structure from input to output and the same number of convolutional layers in the up-sampling and down-sampling sections. Skip connections connect the scores corresponding to up-sampling and down-sampling, achieving higher feature extraction while preserving more image details [27]. In the last three convolutional layers of the down-sampling section, extended convolution is used to improve feature space resolution. The receptive field calculation of extended convolution is shown in equation (1).

$$f = \left(n_p - 1\right) \times (k - 1) + k \tag{1}$$

In equation (1), f represents the convolutional kernel receptive field. n_p represents the expansion rate. k represents

the size of the convolution kernel. Extended convolution does not increase model parameters when increasing the receptive field, which can reduce the amount of data required for image style conversion. Two discriminators can be distinguished based on their functions as primary discriminators and secondary discriminators. The design of the discriminator focuses on capturing the global and local features of the image. The main discriminator pays attention to the global information of the image to ensure the overall consistency of the style. The secondary discriminator, on the other hand, focuses on the local content of the image to ensure the quality of details. While retaining the original content, it fully shows the converted style characteristics. The structural hierarchy of two discriminators is shown in Figure 3.

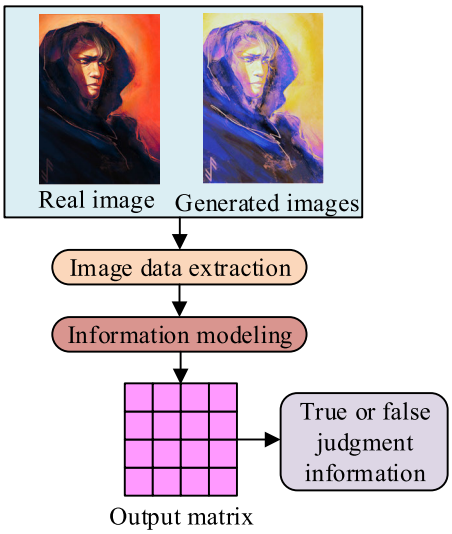

FIGURE 3. Discriminator structure hierarchy.

In Figure 3, the discriminator extracts the input image data and model information based on high-frequency structural information. In the model, key areas of the image are subjected to terminal illumination to extract information output matrix. The value of the output matrix is used to determine the authenticity of the image. In the modeling and constructing output matrices, parameter redundancy caused by irrelevant information is reduced, improving the training effectiveness of the model. The convolution layer of the generator and discriminator uses a convolution kernel of size 3×3, with the fill set to 1 and the step size to 2. In the design of the loss function, a combination of antagonistic loss, cyclic consistent loss and edge consistent loss is used to ensure that the style and content of the generated image are consistent with the target image. The antagonistic loss enables the image generated by the generator to deceive the discriminator. The cyclic coincidence loss ensures the reversibility of the style conversion process. The edge consistent loss ensures that the edge information of the style conversion image is consistent with the original image. The total loss function during model training is shown in equation (2).

$$L(G, F, D_X, D_Y) = L_{GAN} + \lambda_1 L_{cyc}(G, F) + \lambda_2 L_{edge}(X, G) \tag{2}$$

In equation (2), $L(G, F, D_X, D_Y)$ represents the total loss function value. L_{GAN} represents the value of the adversarial loss function. $L_{cyc}(G, F)$ represents the cyclic consistent loss function. $L_{edge}(X, G)$ represents the edge consistent loss function. G and F represent generators. D_X and D_Y represent discriminators. X and Y represent different image domains. λ represents the coefficient of the loss function value. The adversarial loss function consists of the adversarial losses from two different generators, as shown in equation (3).

$$L_{GAN} = L_{GAN}(G, D_Y, X, Y) + L_{GAN}(F, D_X, X, Y) \tag{3}$$

In equation (3), $L_{GAN}(G, D_Y, X, Y)$ represents the adversarial loss function of the main discriminator during the image conversion from the X domain to the Y domain. $L_{GAN}(F, D_X, X, Y)$ represents the generated adversarial loss function of the secondary discriminator when the image is converted from the Y domain to the generator. The cyclic consistent loss function is shown in equation (4).

$$L_{cyc}(G, F) = E_{x \sim p(x)}\left[\|F(G(x)) - x\|_1\right] + E_{y \sim p(y)}\left[\|F(G(y)) - y\|_1\right] \tag{4}$$

In equation (4), E represents the mathematical expectation. x and y represent data samples. The edge consistent loss function is established in the edge extraction network and implemented by the Canny edge algorithm. When performing edge extraction, there is no need to overly focus on noise, only to accurately extract strong and weak edges. Therefore, differential first-order partial derivatives are used to calculate image gradients. The edge loss function is shown in equation (5).

$$L_{edge}(x, G) = \|E(x) - E(G(x))\|_2 \tag{5}$$

In equation (5), $E(x)$ represents the result obtained from edge extraction of the original image. $E(G(x))$ represents the result obtained from edge extraction of the image after style conversion. When performing image style conversion, the model is first trained. Then the image style conversion is performed using the trained model.

B. GENERATIVE ADVERSARIAL NETWORK IMAGE STYLE CONVERSION MODEL DESIGN BASED ON LINEAR MODULE OPTIMIZATION

The image style conversion model needs to extract useful feature representations from the original image and the style image when performing feature extraction. These features not only contain the specific content of the image, but also the artistic style of the image [28], [29]. Convolutional operations can effectively extract image features. Adding linear modules to the convolutional layer can generate more feature

maps [30], [31]. Linear modules are inserted into the GAN to optimize the design of image style conversion models. The output features of the convolutional layer are shown in Figure 4.

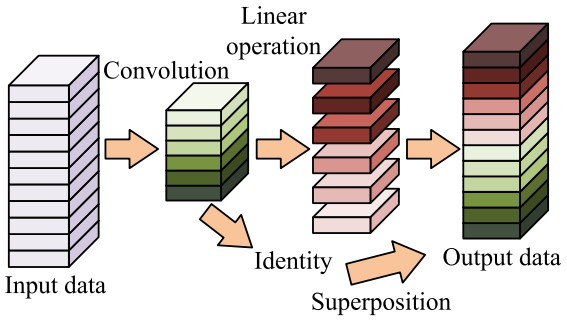

FIGURE 4. Convolutional layer output feature process.

In Figure 4, the original information is convolved after being input, followed by linear transformation to obtain the output feature map. The form of the output feature map is similar to the shadow of the intrinsic feature map. The final data output is concatenated by the feature information obtained from convolution operation and the shadow features obtained from linear operation. When performing feature output, the number of parameters in the convolution kernel and bias term is determined by the size of the feature map. The output feature is shown in equation (6).

$$Y_t = X_R \times f_j + b \tag{6}$$

In equation (6), Y_t represents an output feature with multiple channels. X_R represents input data. f_j represents the convolution kernel. b represents the bias term. The output of each convolution layer is first passed through a linear transformation consisting of a learnable weight matrix and bias vector. Then, the result of the linear transformation is spliced with the original convolution feature map to get the final output feature map. To simplify the algorithm, the bias term is removed. The hyper-parameters used in convolution operations are consistent with those used in regular convolution. The shadow feature map obtained through linear operation is shown in equation (7).

$$y_{ij} = \Phi_{i,j}\left(y_i'\right) \tag{7}$$

In equation (7), y_{ij} represents the shadow feature map. $\Phi_{i,j}$ represents linear operation. y_i' represents the i-th intrinsic feature map in the intrinsic feature map. In the implementation, the weight matrix and bias vector of the linear module are initialized to zero to avoid introducing unnecessary noise. The structure of the input image is better preserved. Then the convolutional module is modified to reconstruct the generator network, as shown in Figure 5.

In Figure 5, the generator network after modifying the convolutional module still includes up-sampling and down-sampling parts, while retaining skip connections. The convolutional layer begins the down-sampling phase, followed by instance normalization and max pooling. The output after instance normalization is concatenated with the input of residual blocks. After the residual module, an attention layer is added. During down-sampling, serial skip connections are used for module connections. After the data enters the up-sampling layer, high-level and low-level features are fused to output a three-channel image. To learn effective features at a higher intensity while reducing ineffective feature learning, attention mechanisms are incorporated into down-sampling. The attention mechanism weights the feature map by learning the importance of different channels in the input feature map to highlight key information and suppress unimportant information. The content of adding attention mechanism is shown in Figure 6.

In Figure 6, the attention mechanism mainly includes three stages: compression, stimulation, and weighting. The compression operation is performed in spatial order and represents the two-dimensional input feature channels in real numbers. Real numbers are required to contain global information to a certain extent. The output dimension of the compression operation needs to correspond to the number of feature channels of the operation input, to ensure that the network has global characteristics at the channel level. The stimulation operation is similar to the gating mechanism, which learns the relationships between channels and generates weights for different channels. The importance of feature channels is represented by the weights. Weighting operation is performed on the channel dimension, weighting the feature information obtained from previous operations to obtain the final feature result. The number of feature maps generated by the convolution module during runtime matches that of existing images, which can reduce the computational cost of image style conversion. The improved convolution theory acceleration is shown in equation (8).

$$r_s = \frac{n \cdot h' \cdot w' \cdot c \cdot k^2}{\frac{n}{s} h' \cdot w' \cdot c \cdot k^2 + (s-1)\frac{n}{s} h' \cdot w' \cdot d^2} \tag{8}$$

In equation (8), r_s represents the theoretical acceleration ratio. d represents the edge length of the linear operation kernel. h' represents the width of the output data. w' represents the height of the output data. c represents the number of input feature channels. s represents the number of linear operations. The compression ratio is shown in equation (9).

$$r_c = \frac{n \cdot c \cdot k^2}{\frac{n}{s} \cdot c \cdot k + (s-1) \cdot \frac{n}{s} \cdot d^2} \tag{9}$$

In equation (9), r_c represents the compression ratio. To reduce the feature dimension during feature extraction, the maximum pooling is selected as the pooling layer. When performing maximum pooling, feature points within the maximum domain are selected. The main methods of pooling and convolution operations are similar. The pooling process focuses on filter size and ignores internal values. The

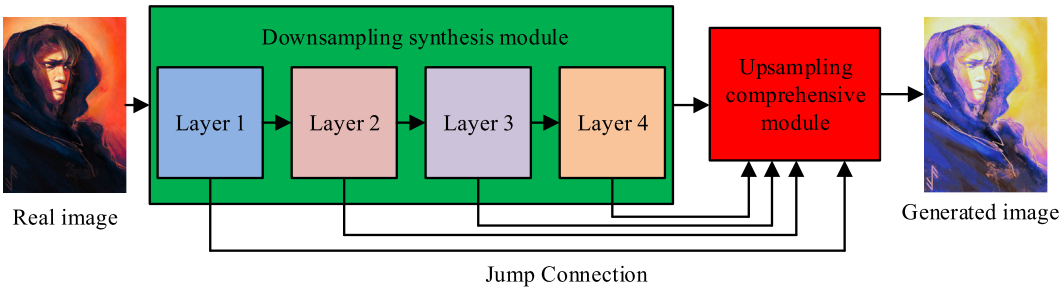

FIGURE 5. The generator network after modifying the convolutional module.

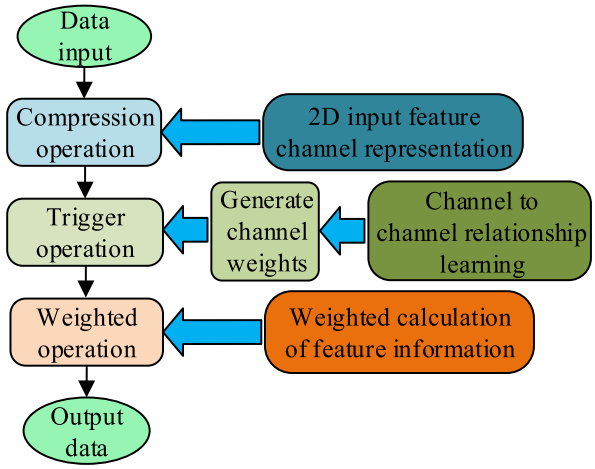

FIGURE 6. Content of attention mechanism.

maximum pooling calculation is shown in equation (10).

$$MaxPooling\left(x_i, i, j\right) = \max_{(p,q)\in pooling\ \text{window}} x_i\left[i+p, j+q\right] \quad (10)$$

In equation (10), x_i represents the input data. (i, j) represents the upper left position of the pooling window. (p, q) represents the position in the pooling window. $MaxPooling\left(x_i, i, j\right)$ represents the maximum value found in the pooling window. To reduce the bias and loss of the image style conversion model during training, the gradient descent method is selected for optimization. To improve the calculation speed, some data is processed simultaneously. The objective function is shown in equation (11).

$$J_{batch}\left(\theta\right) = \frac{1}{b}\sum_{i=1}^{b} L\left(y^{(i)}, h_\theta\left(x_i^{(i)}\right)\right) \quad (11)$$

In equation (11), $J_{batch}\left(\theta\right)$ represents the gradient descent objective function value. b represents the batch size. $L\left(y^{(i)}, h_\theta\left(x_i^{(i)}\right)\right)$ represents the loss of a single sample. θ represents the parameter to be optimized. The gradient is shown in equation (12).

$$\nabla J_{batch}\left(\theta\right) = \frac{1}{b}\sum_{i=1}^{b} \nabla L\left(y^{(i)}, h_\theta\left(x^{(i)}\right)\right) \quad (12)$$

In equation (12), $\nabla J_{batch}\left(\theta\right)$ represents the gradient, which is the change direction of the loss function under the current parameters. $\nabla L\left(y^{(i)}, h_\theta\left(x^{(i)}\right)\right)$ represents the gradient of a single sample loss with respect to the parameters to be optimized. The learning rate is flexibly adjusted at different stages using first-order momentum and second-order momentum control. The first-order momentum is shown in equation (13).

$$v_t = \beta v_{t-1} + \left(1-\beta\right)\nabla J_t\left(\theta\right) \quad (13)$$

In equation (13), v_t represents the first-order momentum. β represents the attenuation coefficient. v_{t-1} represents the first-order momentum of the previous step. $\nabla J_t\left(\theta\right)$ represents the current step gradient. The second-order momentum is shown in equation (14).

$$s_t = \rho s_{t-1} + \left(1-\rho\right)\nabla J_t\left(\theta\right) \quad (14)$$

In equation (14), s_t represents the second-order momentum. ρ represents the attenuation coefficient. s_{t-1} represents the second-order momentum of the previous step. The complete running process of the designed image style conversion model is shown in Figure 7.

In Figure 7, the designed image style conversion model can be mainly divided into two parts: model training and image style conversion. After the model starts running, the first step is to input the target style image, followed by model training. The model was trained using the Adam optimizer with an initial learning rate of 0.0002 and a batch size of 64. The learning rate attenuation strategy was used during training, reducing the learning rate by 0.1 per 100 epochs. During the training process, the image style features are extracted and the model parameters are optimized. When the preset training frequency or duration is reached, the model training is completed. The image style features are extracted and the model parameters are optimized. When the preset training frequency or duration is reached, the model training is completed. Afterwards, the original image that needs to undergo style conversion is input, and the generator generates the converted image. The discriminator identifies and outputs the loss value. Then, the generator restores the image, and the discriminator performs discrimination again and outputs the loss value. If the converted image does not meet the conversion requirements, the loss information is output and the model is retrained. If the image meets the conversion requirements after style conversion, the style conversion image result is output. The image style conversion is completed.

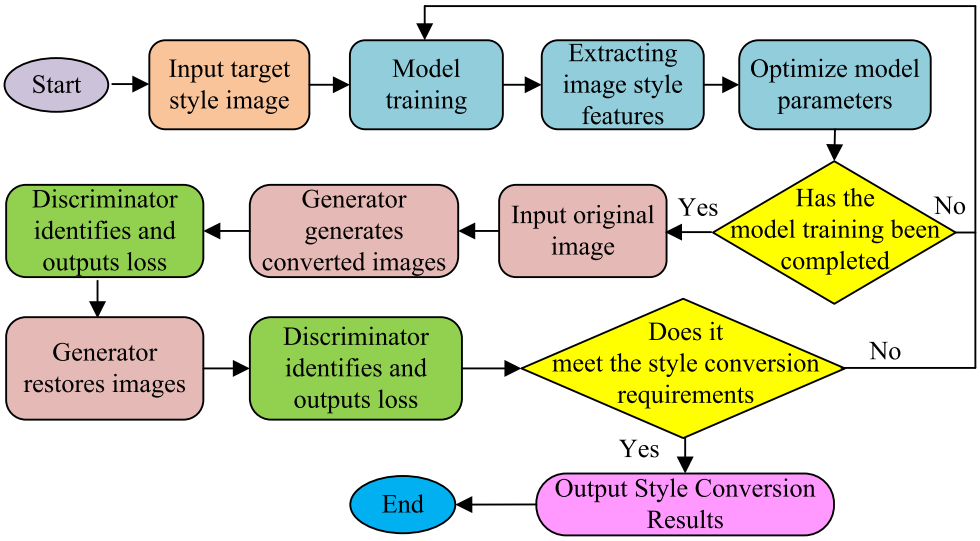

FIGURE 7. The complete running process of image style conversion model.

IV. EFFECTIVENESS ANALYSIS OF IMAGE STYLE CONVERSION MODEL BASED ON GENERATIVE ADVERSARIAL NETWORKS

With the rapid development of computer vision and deep learning technology, image style conversion has become a highly focused research field. This section analyzes the designed image style conversion model from two perspectives: performance testing and application analysis.

A. PERFORMANCE TESTING OF IMAGE STYLE TRANSFORMATION MODEL BASED ON GENERATIVE ADVERSARIAL NETWORKS

To analyze the designed image style conversion model, the WikiArt data set and CelebA data set are selected as the testing data sets for performance testing. The research method, abbreviated as Optimized Generative Adversarial Network (OGAN), is compared with Watchdog Generative Adversarial Network (WGAN) and Instance Normalized Network (INN). The basic software and hardware environment of the experiment is shown in Table 1.

TABLE 1. Basic environmental parameters of the experiment.

Parameter variables	Parameter selection
Operating system	Windows11
Software environment	Pytorch
System running memory	64GB
CPU main frequency	3.30GHz
Graphics card model	NVIDIA GeForce Titan X
CPU	Intel(R) Core (TM) i5-13400

The training time of different methods is tested, as shown in Figure 8.

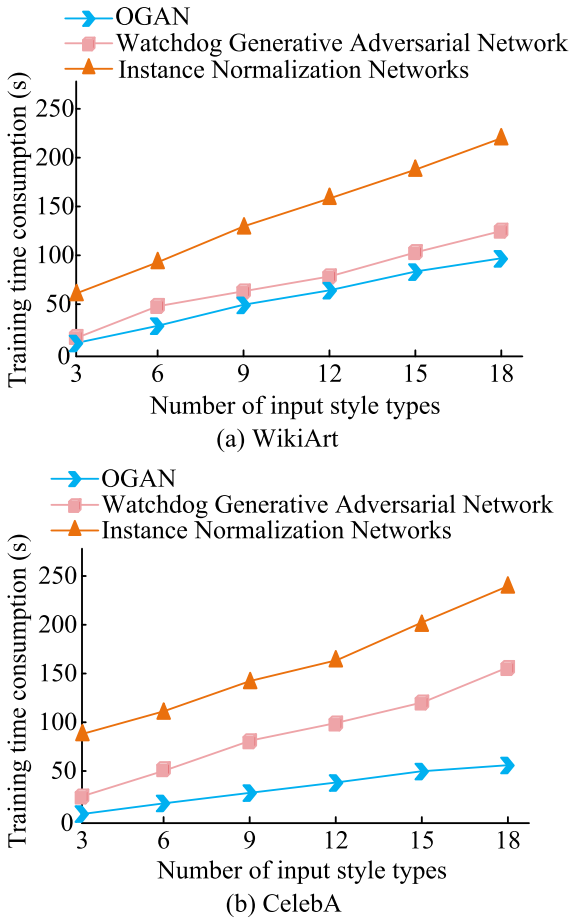

FIGURE 8. Training time consumption.

In Figure 8, the training time of different methods increased with the number of input style types. In Figure 8 (a), in the WikiArt data set, the WGAN was 23s when the number of input style types was 3. When the number of input style types increased to 18, the training time was 125s. The INN was 54s when the number of input style types was 3. When

the input style types increased to 18, the training time was 221s. The training time for OGAN with 3 input style types was 14s. When the number of input style types increased to 18, it was 97s. In Figure 8 (b), in the CelebA data set, the training time of the WGAN was 26s when the number of input style types was 3. The training time when the number of input style types increased to 18 was 152s. The training time of the INN was 89s when the number of input style types was 3. When the number of input style types increased to 18, it was 239s. For OGAN with 3 input style types, it took 9s. When the number of input style types increased to 18, it was 56s. This indicates that the research method has faster model pre-training efficiency, which can be deployed faster. In general, when the number of input style types increased to 18, the training time of OGAN model remained below 97 seconds, which significantly reduced the training time compared with other methods. This advantage is mainly due to the dual synthesizer and dual discriminator structure of the model, which allows the model to converge faster and effectively reduces the computational complexity by reducing the number of model parameters. The Peak Signal-to-Noise Ratio (PSNR) is tested, as shown in Figure 9.

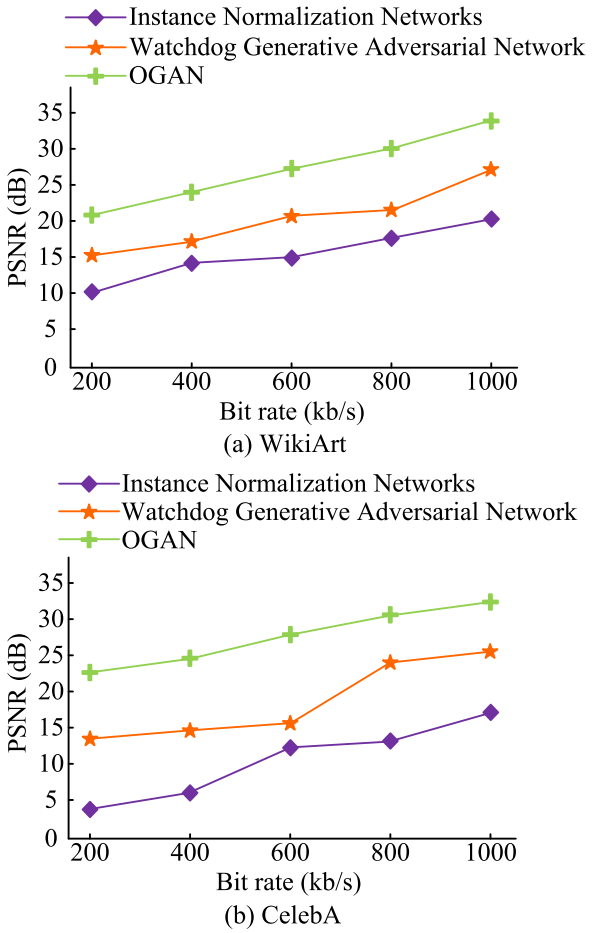

FIGURE 9. Peak signal-to-noise ratio test.

In Figure 9, the PSNR of different methods increased overall with the increase of bit rate. In Figure 9 (a), in the WikiArt data set, the PSNR of the WGAN was 15.1dB at a bit rate of 200kb/s. The PSNR increased to 25.3dB when the bit rate reached 1000kb/s. The INN was 10.1dB at a bit rate of 200kb/s. The PSNR increased to 20.0dB when the bit rate reached 1000kb/s. The PSNR of OGAN was 20.8dB at a bit rate of 200kb/s. The PSNR increased to 34.1dB when the bit rate reached 1000kb/s. In Figure 9 (b), in the CelebA data set, the PSNR of the WGAN was 13.7dB at a bit rate of 200kb/s. The PSNR increased to 25.3dB when the bit rate was 1000kb/s. The INN was 4.1dB at a bit rate of 200kb/s. The PSNR increased to 16.9dB when the bit rate reached 1000kb/s. The PSNR of OGAN at a bit rate of 200kb/s was 22.6dB. The PSNR increased to 32.4dB when the bit rate reached 1000kb/s. This indicates that the data processed by the research method has less distortion. When the bit rate reaches 1000kb/s, the peak signal-to-noise ratio of OGAN model reaches 34.1dB, which indicates that the model has excellent performance in image quality preservation. This is attributed to the use of cyclic consistent loss functions and edge consistent loss functions in the model, which help to retain more image detail and edge information during style transitions. The F1 values of each method are tested, as shown in Figure 10.

In Figure 10, the F1 values of different methods continuously increased with the number of iterations. In Figure 10 (a), in the WikiArt data set, the F1 value of the WGAN was 58.8 at 50 iterations. When the number of iterations increased to 250, the F1 value was 76.1. The INN was 72.4 at 50 iterations. When the number of iterations increased to 250, the F1 value was 86.7. The OGAN was 77.7 at 50 iterations. When the number of iterations increased to 250, the F1 value was 98.2. In Figure 10 (b), in the CelebA data set, the F1 value of the WGAN at 50 iterations was 56.3. When the number of iterations increased to 250, the F1 value was 73.1. The INN at 50 iterations was 67.5. When the number of iterations increased to 250, the F1 value was 81.9. The OGAN was 79.1 at 50 iterations. When the number of iterations increased to 250, the F1 value added to 95.0. This indicates that the research method has better accuracy and recall. When the number of iterations of OGAN model increased to 250, the F1 value remained above 95.0, showing better accuracy and recall rate. This result is due to the introduction of an attention mechanism in the model, which allows the model to pay more attention to key features in the image, thus improving the accuracy of style transformation. The image style loss at runtime is tested, as shown in Figure 11.

In Figure 11, the style loss of different methods increased with the increase of input image pixels. In Figure 11 (a), in the WikiArt data set, the image style loss value of the WGAN was 17.8 when the input image pixel count was 2M. The image style loss value increased to 33.4 when the input image pixel size was 10M. The INN had an image style loss value of 24.1 when the input image pixel count was 2M. The image style loss value increased to 40.2 when the input

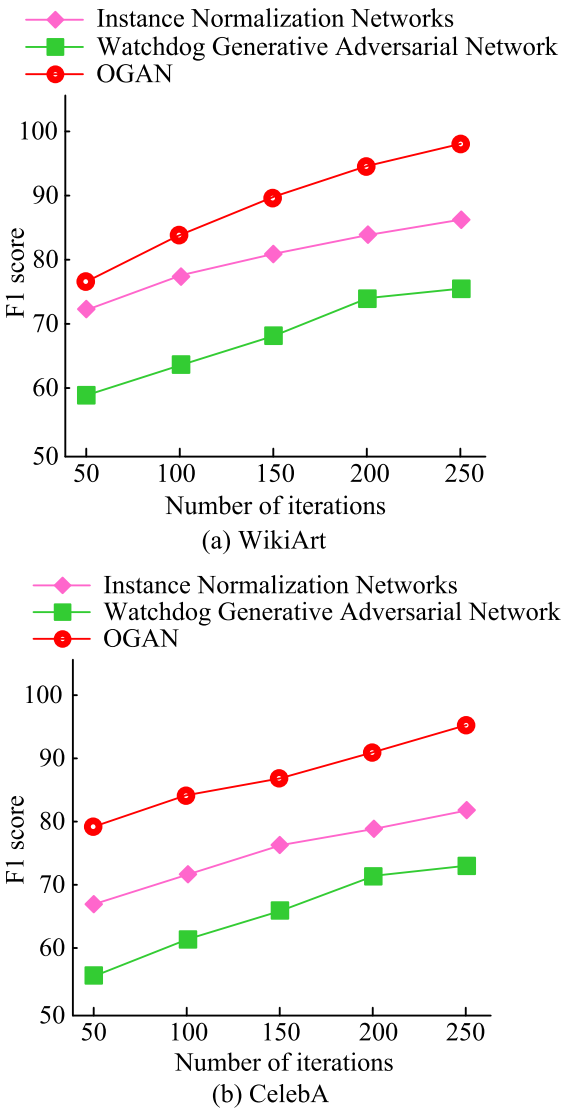

FIGURE 10. F1 value.

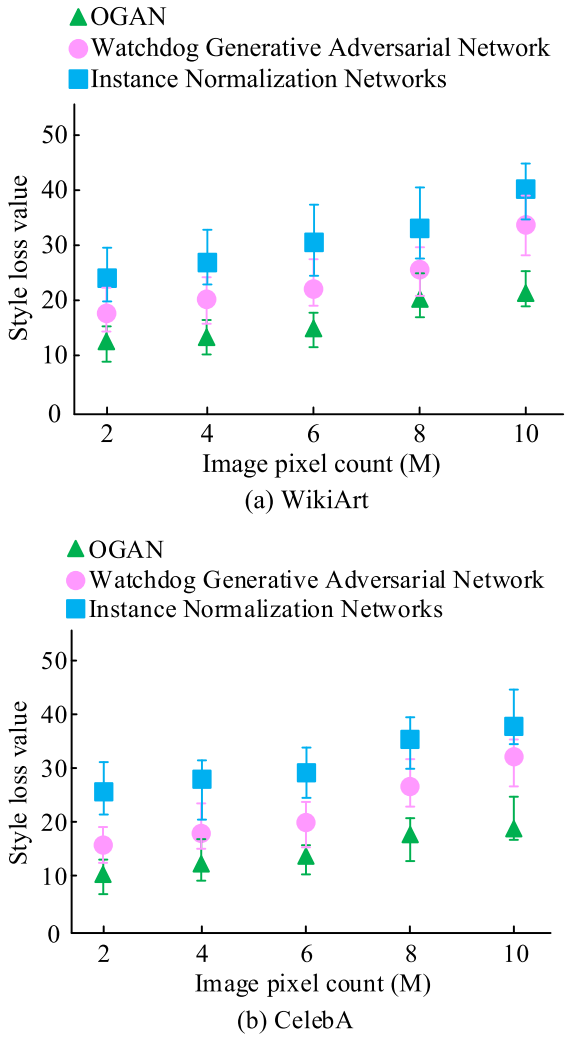

FIGURE 11. Image style loss test.

image pixel count was 10M. When the input image pixel count was 2M, the OGAN was 11.9. The image style loss value increased to 20.9 when the input image pixel count was 10M. In Figure 11 (b), in the CelebA data set, the image style loss value of the WGAN was 16.2 when the input image pixel count was 2M. The loss value increased to 32.1 when the input image pixel count was 10M. The INN had an image style loss value of 26.3 when the input image pixel count was 2M. The image style loss value increased to 37.8 when the input image pixel count was 10M. When the input image pixel count was 2M, the OGAN was 10.0. The image style loss value increased to 18.9 when the input image pixel count was 10M. This indicates that the research method has better accuracy in maintaining image style. The OGAN model maintains an image style loss value below 20.9 when the input size is 1080p and the pixel count is 10M. This result indicates that the model has high accuracy in preserving image style features. It is related to the design of the anti-loss function in the model, which prompts the generator to generate images more in line with the target style characteristics.

B. APPLICATION ANALYSIS OF IMAGE STYLE CONVERSION MODEL BASED ON GENERATIVE ADVERSARIAL NETWORKS

When conducting application analysis, 100 real images and 100 virtual images are used for image style conversion application testing and analysis. The average running time of image style conversion is analyzed, as shown in Figure 12.

In Figure 12, the running time of different methods increased with the increase of image clarity. In Figure 12 (a), when processing real images, the average running time of the WGAN was 194ms when the image clarity was 480p. The average running time increased to 768ms when the image clarity increased to 2160p. The INN at image clarity of 480p was 426ms. The average running time increased to 973ms when the image clarity increased to 2160p. The OGAN

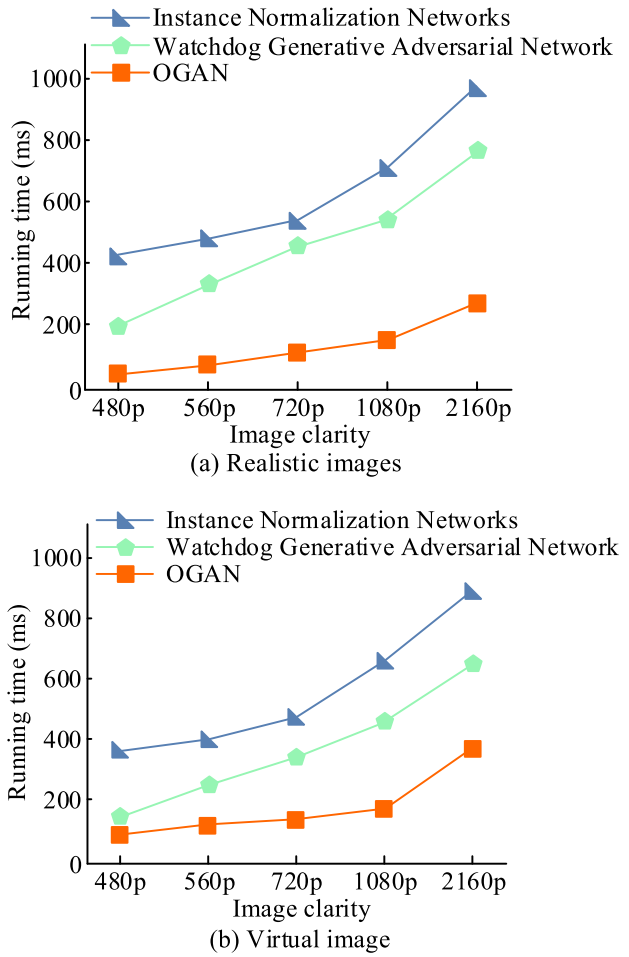

FIGURE 12. Average running time.

at image clarity of 480p was 57ms. The average running time increased to 272ms when the image clarity was 2160p. In Figure 12 (b), the WGAN had an average running time of 159ms when the image clarity was 480p. The average running time increased to 641ms when the image clarity was 2160p. The INN at image clarity of 480p was 378ms. The average running time increased to 889ms when the image clarity was 2160p. The OGAN at image clarity of 480p was 91ms. The average running time increased to 391ms when the image clarity was 2160p. This indicates that the research method has a faster image style conversion speed. The memory usage of the computer model during operation is tested, as shown in Figure 13.

In Figure 13, the memory usage of different methods fluctuated within a certain range during operation. The WGAN fluctuated between 8% and 63% within 100s, with an average memory usage of about 41%. The INN fluctuated between 40% and 68% within 100s, with an average memory usage of about 54%. The OGAN fluctuated between 26% and 37% within 100s, with an average memory usage of approximately 31%. This indicates that the research method has more stable memory usage and lower memory space requirements

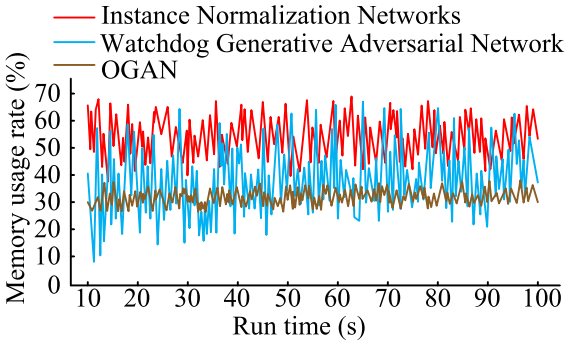

FIGURE 13. Memory usage rate.

during actual operation. The optimized design of OGAN model reduces unnecessary computation and memory usage through efficient data utilization. Introduce other commonly used DualStyleGAN and GigaGAN for comparison. Analyze the pixel loss when performing style conversion on images containing 10M pixels, as shown in Figure 14.

In Figure 14, the pixel loss of different methods in image style conversion increased with the depth of image conversion. As shown in Figure 14 (a), when processing real images, the watchdog generative adversarial network has a pixel loss of 3.7k at 20% conversion completion; The pixel loss of GigaGAN increases to 8.4k when the conversion is fully completed. The pixel loss of the instance normalization network at 20% conversion completion is 6.5k; The pixel loss of DualStyleGAN increases to 6.9k when the conversion is fully completed. The pixel loss of OGAN at 20% conversion completion is 1.1k; The pixel loss increases to 4.8k when the conversion is fully completed. As shown in Figure 14 (b), when processing virtual images, the watchdog generative adversarial network has a pixel loss of 1.9k at 20% conversion completion; The pixel loss of GigaGAN increases to 6.7k when the conversion is fully completed. The pixel loss of the instance normalization network at 20% conversion completion is 8.0k; The pixel loss of DualStyleGAN increases to 8.5k when the conversion is fully completed. The OGAN at 20% conversion completion was 0.7k. The pixel loss increased to 3.5k when the conversion was fully completed. The research method can better maintain pixel integrity during image style conversion. It is related to the extended convolution used in the model, which improves the resolution of feature extraction without increasing the number of parameters. The conversion results of the two example images are analyzed, as shown in Figure 15.

In Figure 15, different methods successfully generated the image style conversion results. In Figure 15 (a), when converting real images, the Maillard style was used as the target style. The conversion results of the WGAN had significant deficiencies overall. There was rich noise and high color saturation in the image. The conversion results of the INN exhibited over-exposure in the bright areas, with much noise in the highlight area. The conversion result of OGAN was

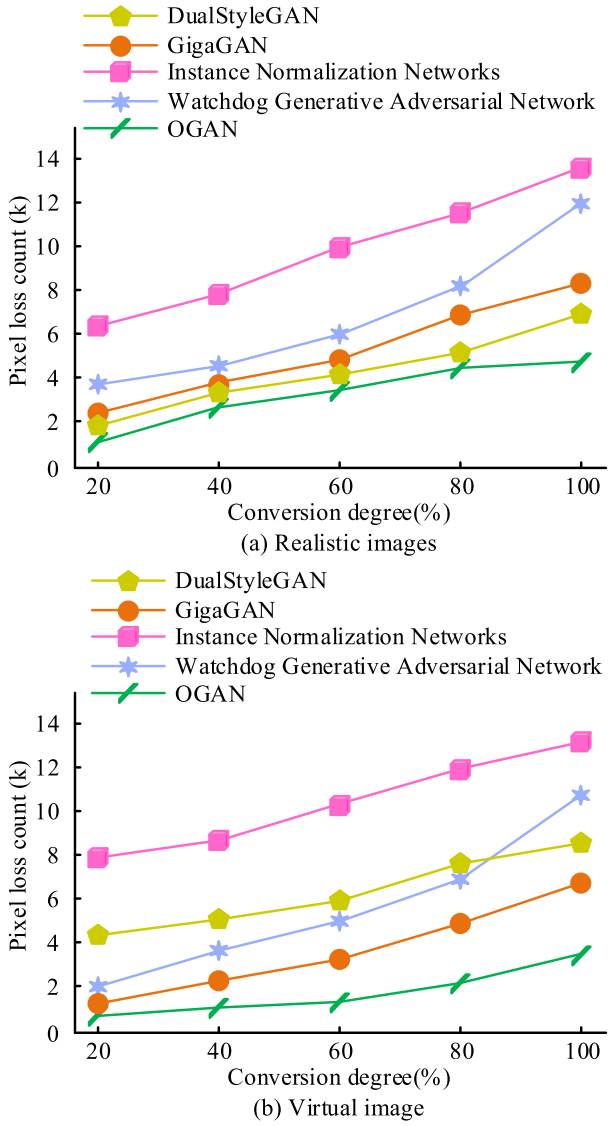

FIGURE 14. Pixel loss.

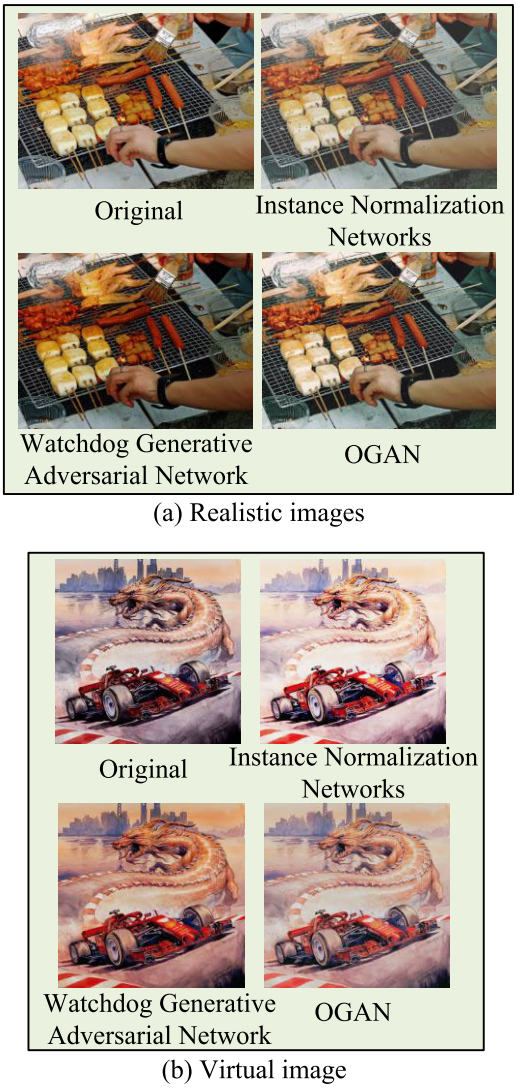

FIGURE 15. Analysis of conversion results.

quite similar to the Maillard style, without too much noise. In Figure 15 (b), the Dunhuang mural style was used as the target style when converting virtual images. The conversion results of adversarial networks generated by Watchdog had low image brightness and lacked dark details. For the INN, the contrast of the conversion results was too low, resulting in an overall explosion. The conversion results of OGAN were quite similar to the style of Dunhuang murals, maintaining good picture quality.

C. DISCUSSION

In the field of image style conversion, maintaining high-quality images is a key indicator of model performance. The optimized generative adversarial network model presented in this study has demonstrated excellent image quality retention ability in multiple tests. The OGAN model uses an innovative twin and dual discriminator architecture, which significantly improves the convergence speed and training efficiency of the model. In the WikiArt dataset, even if the number of input style types is increased to 18, the training time of the OGAN model can still be controlled under 97 seconds, which is particularly significant compared to the traditional single generator model.

The expansive convolution technique is introduced into the subsampling part of the generator in OGAN model, which enhances the ability of the model to capture image details by enlarging the receptive field of the convolution kernel. In image style conversion, the application of this technique can keep the image style loss of 10M pixels below 20.9, which highlights the advantages of the model in processing high-resolution images. By introducing attention mechanisms into the model, OGAN models can more accurately identify and emphasize key features of images while suppressing unimportant information. The application of this mechanism enables the model to process 2160p high definition images

with an average running time of only 272 milliseconds, while maintaining image quality, which is of great value in real-time application scenarios.

OGAN model adopts the combination of counter loss, cyclic consistent loss and edge consistent loss, which not only ensures the reversibility of the style conversion process, but also ensures the high consistency of the edge information of the style conversion image with the original image. In the test of image style loss, the OGAN model's style loss value can be maintained at a low level even in the image processing of 10M pixels, which highlights the importance of loss function design in maintaining image quality. In the training process of OGAN model, Adam optimizer and learning rate attenuation strategy are adopted, and the optimization of these strategies makes the model parameters converge to the optimal state faster. In the experiment, the F1 value of OGAN model can reach more than 95.0 after 250 iterations, which shows the model's efficient performance in style conversion accuracy. It shows that the research method can realize the effective maintenance of image quality through a series of innovative technologies.

V. CONCLUSION

A method based on GAN was proposed to improve the quality of image style conversion. In the process, extended convolution was used in the down-sampling of the generator, the differential first-order deflection was used to calculate the image gradient, and the attention mechanism was inserted in the down-sampling.

The experimental results showed that the research method achieved a PSNR of 34.1dB when the bit rate reached 1000kb/s. The F1 value of the research method remained above 95.0 when the number of iterations increased to 250. In the average running time analysis of image style conversion, the research method had a maximum running time of 272ms when processing real image clarity of 2160p. The memory usage of the research method fluctuated between 26% and 37% during operation. In the analysis of conversion results, the conversion results had less noise and more accurate style types.

This indicates that the research method can effectively convert image styles and has higher conversion efficiency. However, the study only tests a single non-continuous image, which cannot yet determine the consistency and continuity of image style conversion in continuous frame video images. Subsequent research will expand the testing scope and optimize the method.

REFERENCES

[1] L. Zhang, B. Shen, A. Barnawi, S. Xi, N. Kumar, and Y. Wu, "FedDPGAN: Federated differentially private generative adversarial networks framework for the detection of COVID-19 pneumonia," *Inf. Syst. Frontiers*, vol. 23, no. 6, pp. 1403–1415, Jun. 2021, doi: 10.1007/s10796-021-10144-6.

[2] J. Yu and J. Liu, "Multiple granularities generative adversarial network for recognition of wafer map defects," *IEEE Trans. Ind. Informat.*, vol. 18, no. 3, pp. 1674–1683, Mar. 2022, doi: 10.1109/TII.2021.3092372.

[3] A. Kammoun, R. Slama, H. Tabia, T. Ouni, and M. Abid, "Generative adversarial networks for face generation: A survey," *ACM Comput. Surv.*, vol. 55, no. 5, pp. 1–37, Dec. 2022, doi: 10.1145/3527850.

[4] X. Zheng, Y. Liu, P. Wang, and X. Tong, "SDF-StyleGAN: Implicit SDF-based StyleGAN for 3D shape generation," *Comput. Graph. Forum*, vol. 41, no. 5, pp. 52–63, Oct. 2022, doi: 10.1111/cgf.14602.

[5] A. Chen, R. Liu, L. Xie, Z. Chen, H. Su, and J. Yu, "SofGAN: A portrait image generator with dynamic styling," *ACM Trans. Graph.*, vol. 41, no. 1, pp. 1–26, Feb. 2022, doi: 10.1145/3470848.

[6] R. Abdal, P. Zhu, N. J. Mitra, and P. Wonka, "StyleFlow: Attribute-conditioned exploration of StyleGAN-generated images using conditional continuous normalizing flows," *ACM Trans. Graph.*, vol. 40, no. 3, pp. 1–21, May 2021, doi: 10.1145/3447648.

[7] S. Yang, E. Y. Kim, and J. C. Ye, "Continuous conversion of CT kernel using switchable CycleGAN with AdaIN," *IEEE Trans. Med. Imag.*, vol. 40, no. 11, pp. 3015–3029, Nov. 2021, doi: 10.1109/TMI.2021.3077615.

[8] W. Jang, G. Ju, Y. Jung, J. Yang, X. Tong, and S. Lee, "StyleCariGAN: Caricature generation via StyleGAN feature map modulation," *ACM Trans. Graph.*, vol. 40, no. 4, pp. 1–16, Jul. 2021, doi: 10.1145/3450626.3459860.

[9] Y. Li, P. Che, C. Liu, D. Wu, and Y. Du, "Cross-scene pavement distress detection by a novel transfer learning framework," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 36, no. 11, pp. 1398–1415, Jun. 2021, doi: 10.1111/mice.12674.

[10] R. Gal, D. C. Hochberg, A. Bermano, and D. Cohen-Or, "SWAGAN: A style-based wavelet-driven generative model," *ACM Trans. Graph.*, vol. 40, no. 4, pp. 1–11, Jul. 2021, doi: 10.1145/3450626.3459836.

[11] X. Li, J. Su, Z. Zhang, and R. Bai, "Product innovation concept generation based on deep learning and kansei engineering," *J. Eng. Des.*, vol. 32, no. 10, pp. 559–589, Jun. 2021, doi: 10.1080/09544828.2021.1928023.

[12] L. Wang, L. Wang, and S. Chen, "ESA-CycleGAN: Edge feature and self-attention based cycle-consistent generative adversarial network for style transfer," *IET Image Process.*, vol. 16, no. 1, pp. 176–190, Jan. 2022, doi: 10.1049/ipr2.12342.

[13] B. Gao, J. Zhou, Y. Yang, J. Chi, and Q. Yuan, "Generative adversarial network and convolutional neural network-based EEG imbalanced classification model for seizure detection," *Biocybernetics Biomed. Eng.*, vol. 42, no. 1, pp. 1–15, Jan. 2022, doi: 10.1016/j.bbe.2021.11.002.

[14] J. Daihong, Z. Sai, D. Lei, and D. Yueming, "Multi-scale generative adversarial network for image super-resolution," *Soft Comput.*, vol. 26, no. 8, pp. 3631–3641, Feb. 2022, doi: 10.1007/s00500-022-06822-5.

[15] S. Kench and S. J. Cooper, "Generating three-dimensional structures from a two-dimensional slice with generative adversarial network-based dimensionality expansion," *Nature Mach. Intell.*, vol. 3, no. 4, pp. 299–305, Apr. 2021, doi: 10.1038/s42256-021-00322-1.

[16] X. Lei, L. Sun, and Y. Xia, "Lost data reconstruction for structural health monitoring using deep convolutional generative adversarial networks," *Struct. Health Monitor.*, vol. 20, no. 4, pp. 2069–2087, Jul. 2021, doi: 10.1177/1475921720959226.

[17] H. Maeda, T. Kashiyama, Y. Sekimoto, T. Seto, and H. Omata, "Generative adversarial network for road damage detection," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 36, no. 1, pp. 47–60, Jan. 2021, doi: 10.1111/mice.12561.

[18] Z. Cai, Z. Xiong, H. Xu, P. Wang, W. Li, and Y. Pan, "Generative adversarial networks: A survey toward private and secure applications," *ACM Comput. Surv.*, vol. 54, no. 6, pp. 1–38, Jul. 2021, doi: 10.1145/3459992.

[19] J. Chen, S. Li, D. Liu, and W. Lu, "Indoor camera pose estimation via style-transfer 3D models," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 37, no. 3, pp. 335–353, Mar. 2022, doi: 10.1111/mice.12714.

[20] G. Song, L. Luo, J. Liu, W.-C. Ma, C. Lai, C. Zheng, and T.-J. Cham, "AgileGAN: Stylizing portraits by inversion-consistent transfer learning," *ACM Trans. Graph.*, vol. 40, no. 4, pp. 1–13, Jul. 2021, doi: 10.1145/3450626.3459771.

[21] X. Ye, J. Du, and Y. Ye, "MasterplanGAN: Facilitating the smart rendering of urban master plans via generative adversarial networks," *Environ. Planning B, Urban Anal. City Sci.*, vol. 49, no. 3, pp. 794–814, Mar. 2022, doi: 10.1177/23998083211023516.

[22] E. Colleoni and D. Stoyanov, "Robotic instrument segmentation with Image-to-Image translation," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 935–942, Apr. 2021, doi: 10.1109/LRA.2021.3056354.

[23] Ş. Öztürk, U. Özkaya, and M. Barstuğan, "Classification of coronavirus (COVID-19) from X-ray and CT images using shrunken features," *Int. J. Imag. Syst. Technol.*, vol. 31, no. 1, pp. 5–15, Mar. 2021, doi: 10.1002/ima.22469.

[24] Y. Kasten, D. Ofri, O. Wang, and T. Dekel, "Layered neural atlases for consistent video editing," *ACM Trans. Graph.*, vol. 40, no. 6, pp. 1–12, Dec. 2021, doi: 10.1145/3478513.3480546.
[25] M. Hasanvand, "Machine learning methodology for identifying vehicles using image processing," *Artif. Intell. Appl.*, vol. 1, no. 3, pp. 170–178, Jul. 2023, doi: 10.47852/bonviewaia3202833.
[26] L. Gong, W. Wang, T. Wang, and C. Liu, "Robotic harvesting of the occluded fruits with a precise shape and position reconstruction approach," *J. Field Robot.*, vol. 39, no. 1, pp. 69–84, Jan. 2022, doi: 10.1002/rob.22041.
[27] H. Weir, K. Thompson, A. Woodward, B. Choi, A. Braun, and T. J. Martínez, "ChemPix: Automated recognition of hand-drawn hydrocarbon structures using deep learning," *Chem. Sci.*, vol. 12, no. 31, pp. 10622–10633, Aug. 2021, doi: 10.1039/d1sc02957f.
[28] K. Bhosle and V. Musande, "Evaluation of deep learning CNN model for recognition of devanagari digit," *Artif. Intell. Appl.*, vol. 1, no. 2, pp. 114–118, Feb. 2023, doi: 10.47852/bonviewaia3202441.
[29] C. Heinze-Deml and N. Meinshausen, "Conditional variance penalties and domain shift robustness," *Mach. Learn.*, vol. 110, no. 2, pp. 303–348, Feb. 2021, doi: 10.1007/s10994-020-05924-1.
[30] M. Zhang, T. Wang, D. Ceylan, and N. J. Mitra, "Deep detail enhancement for any garment," *Comput. Graph. Forum*, vol. 40, no. 2, pp. 399–411, Jun. 2021, doi: 10.1111/cgf.142642.
[31] J. Du, F. Wu, R. Xing, X. Gong, and L. Yu, "Segmentation and sampling method for complex polyline generalization based on a generative adversarial network," *Geocarto Int.*, vol. 37, no. 14, pp. 4158–4180, Jul. 2022, doi: 10.1080/10106049.2021.1878288.

KE GONG was born in Jilin, China, in 1978. He received the Bachelor of Arts degree and the master's degree in fine arts from Northeast Normal University, in 1996 and 2010, respectively.

Since 2000, he has been a Teacher with the Department of Art and Design, Jilin University of Education, China. He has been an Associate Professor with Jilin Provincial Institute of Education, since 2013. He is the author of four books and more than 15 articles. His research interests include art creation, aesthetic education research, and art teacher training.

ZHU ZHEN was born in Jilin, China, in 1977. She received the B.A. degree in fine arts and the M.A. degree in arts from Northeast Normal University, Jilin, in 2001 and 2008, respectively. From 2001 to 2004, she was a Teaching Assistant at the Department of Visual Communication, Jilin Animation Institute, Jilin. Since 2009, she has been a Lecturer with the Department of Basic Teaching, Jilin Police College, Jilin. In January 2023, she was moved up from a Lecturer to an Associate Professor. She is the author of one book and two textbooks, more than six articles. Her research interests include image style, animation illustration, internet animation model, and brand image.