Artistic Image Style Conversion Based on Multi-Scale Feature Fusion Network

Huizhou Li[1*], Wubin Zhu[2]
[1]School of Fine Arts and Design, Hefei Normal University, Hefei, 230601, China
[2]Zhejiang Uniview Technologies Co., Ltd, Hangzhou, 310051, China
E-mail: lihuizhou@hfnu.edu.cn, wubing202403@163.com
[*]Corresponding author

Keywords: CNN, MSFF network, AM, artistic images, style conversion

Received: March 29, 2024

To enhance the efficiency and quality of artistic image style conversion, this study improves the convolutional neural network style conversion algorithm by introducing a multi-scale feature fusion network, comprehensively considering different convolutional features. Then, combined with attention mechanism, important features of artistic images are extracted. It occupied less conversion time, CPU usage, and memory usage in the artistic image style conversion, with better conversion performance. The research method had high peak signal-to-noise ratio and structural similarity index when converting different artistic styles. The highest peak signal-to-noise ratios for converting to Van Gogh artistic style, Ukiyo-e style, Monet style, and Cézanne style were 22.892, 17.844, 21.647, and 22.291, respectively, and the highest structural similarity index values were 0.842, 0.783, 0.845, and 0.843, respectively. The research has achieved effective conversion of target styles while preserving content in images, improving the quality and effectiveness of artistic image style conversion, and promoting the image processing technology.

Povzetek: Študija izboljšuje algoritem za pretvorbo umetniških slik v drugačne stile s pomočjo konvolucijskih globokih nevronskih mrež.

1 Introduction

With the progress of artificial intelligence and computer vision technology, artistic image style conversion technology has gradually become a research hotspot. The artistic image style conversion technology aims to convert the style of one image into another, which has broad application prospects [1]. However, existing methods have certain limitations in processing large-scale image data and performing style conversion while preserving content. Traditional methods suffer from style distortion and slow computational speed. Therefore, the research on artistic image style conversion based on Multi-scale Feature Fusion (MSFF) networks is of great significance. The MSFF is a network structure that utilizes deep learning techniques to fuse image features extracted at different scales. By integrating feature information, the accuracy and effectiveness of image processing tasks can be improved, which is suitable for fields such as image style conversion, semantic segmentation, etc [2]. The feature information of various scales is fused, which can retain the content information while approximating the target style information into the generated image, improving the quality and efficiency of style conversion [3]. Therefore, the research aims to use MSFF to achieve more accurate and efficient artistic image style conversion, and improve its quality and efficiency. The innovation lies in the introduced attention mechanism, which help the network focus its attention on more important features. The research provides an effective solution for the development of artistic image style conversion technology, with significant scientific research and practicality. It can promote the progress and application of related technologies. This study has four parts. The first reviews the literature, summarizing the existing results of MSFF networks and artistic image style conversion. The second mainly discusses the improved Convolutional Neural Network (CNN), MSFF network, and the attention mechanism for artistic image style conversion. The third mainly compares the research methods. The last part summarizes the achievements and shortcomings.

2 Related works

MSFF network is a crucial and widely used research direction, which integrates feature information at various scales and improve image processing and analysis performance. Zhou et al. built an unsupervised dense network ground on MSFF, and residual modules to address the multi-focus image fusion. It performed better, providing an efficient solution [4]. Deng et al. designed an efficient and lightweight MSFF multi-tasking strategy to address the challenges of cell segmentation and counting. A new up-sampling method, norm combination loss function, and coordinated multi-tasking training discriminator were introduced to achieve non-point-based cell counting and segmentation tasks based on cell count and global segmentation annotations. Compared with

traditional methods, the research method had fewer parameters and better performance. The speed increased by nearly ten times [5]. Wang et al. built a method ground on feature fusion and hybrid strategy to address the significant challenge of re-identifying individuals. The ResNet50 backbone was improved and implemented with a deep kernel pooling strategy and a mixed loss function. In three datasets including CUHK3, the research method had higher recognition accuracy, surpassing multiple advanced methods [6]. Wang et al. developed a MSFF network framework to solve the difficult single image crowd counting. This network combined encoder decoder, dense dilated convolutional block, and channel attention mechanism to improve the accuracy of density maps. It was superior to existing methods. The ablation study confirmed the effectiveness of each component [7]. Shen et al. built a hyper-spectral classification strategy ground on a three-dimensional MSFF strategy and channel attention mechanism to address the difficulties of traditional 2D or 3D deep CNNs in hyper-spectral image classification. The proposed method had significant progress in hyper-spectral data classification, solving the challenges of traditional methods in dealing with limited training samples and excessive parameters [8].

Applying algorithms to solve image related problems is an important and widely used method that can achieve functions such as image recognition, processing, and analysis, which has great value and role in fields such as computer vision, medical imaging, and security monitoring. Sun et al. developed a strategy to improve the structure and weight initialization of the deep CNN to solve image classification problems. The variable length gene coding strategy was used to represent network building blocks and depth. The new connection weights were introduced to initialize the representation scheme. It could improve computational efficiency, which was superior to existing designs in terms of classification error rate and weight quantity [9]. Bi et al. designed a genetic performance program with knowledge transfer to address the high computational cost of current large-scale image classification. The new fitness function and set were used to represent the effective image classification set established by the strategy. It could achieve better classification ability in a shorter computation time, which had significant advantages over baseline genetic performance program algorithms and other algorithms [10]. In response to the high time consumption of fractal image compression, Li et al. developed a specific update strategy to improve the computational time in fractal image compression. Experimental results showed that while maintaining image quality, the research method had higher encoding efficiency. It could effectively reduce encoding time [11]. Alkishriwo proposed an adaptive multi-resolution image decomposition strategy to optimize image compression without reducing image quality, which conducted multi-resolution decomposition in different directions. The designed method performed excellently compression ratio, bringing new solutions to the image compression [12]. Tade and Vyas proposed a hybrid depth classifier to classify tone mapped images in various visualization applications. The research method was superior to other image quality evaluation methods. It had the potential to solve color mapping challenge in high dynamic range environments, providing the best quality images for specific visualization applications [13]. The summary of the related works is shown in Table 1.

Table 1: Related works summary table

Author	Main method	Key result	Limitation
Zhou et al. [4]	Unsupervised dense networks	Extract source image features	May require a significant amount of computing resources
Deng et al. [5]	Efficient and lightweight multi-scale feature fusion multi-task model	Fewer parameters, better performance, and nearly ten times faster.	Small object detection may not be accurate enough
Wang et al. [6]	A method based on feature fusion and hybrid strategy	Higher recognition accuracy	Weak generalization ability
Wang et al. [7]	Multi-layer feature fusion network framework	Improved the accuracy of density maps	May lead to over-fitting
Shen et al. [8]	Hyper-spectral image classification method	Remarkable progress in hyper-spectral data classification	Unclear applicability of non hyperspectral image data
Sun et al. [9]	An improved structure for deep convolutional neural networks	Improved computational efficiency	Maybe too much search space
Bi et al. [10]	Divide-and-conquer genetic performance program	Better classification performance in less computation time	Need to design fitness functions
Li et al. [11]	Specific update search algorithm	Higher coding efficiency	Image compression may not be ideal for complex textures
Alkishriwo [12]	Image decomposition	Excellent performance in	Sensitive to parameter

	algorithm based on adaptive multi-resolution	peak signal-to-noise ratio	selection
Tade and Vyas [13]	A hybrid depth classifier	Solved color tone mapping in high dynamic range environments	Unable to determine the special classification effect

In summary, integrating feature information from different scales can improve image processing and analysis performance. Given the style distortion and slow computational speed of traditional methods for artistic image style conversion, this study utilizes the MSFF network for artistic image style conversion, achieving more accurate image artistic style conversion.

3 Feature extraction and style conversion of artistic images ground on MSFF Network

To improve the efficiency and quality of artistic image style conversion, the CNN is first improved to better extract image features. Then, the MSFF network is used to fuse features at various scales to improve the expression ability. The attention mechanism is adopted to fix on the more important features of artistic images.

3.1 Image feature extraction based on improved CNN

Image style conversion algorithms ground on deep learning use CNN to extract image features. Then the U-shaped network structure is used for style conversion. The high-level convolutional features of the input content image and the target style image are calculated by the encoder. The style conversion algorithm is combined to form a fused feature map, which is ultimately mapped back to the original pixel space by the decoder to get the target style conversion image [14]. Figure 1 displays the process.

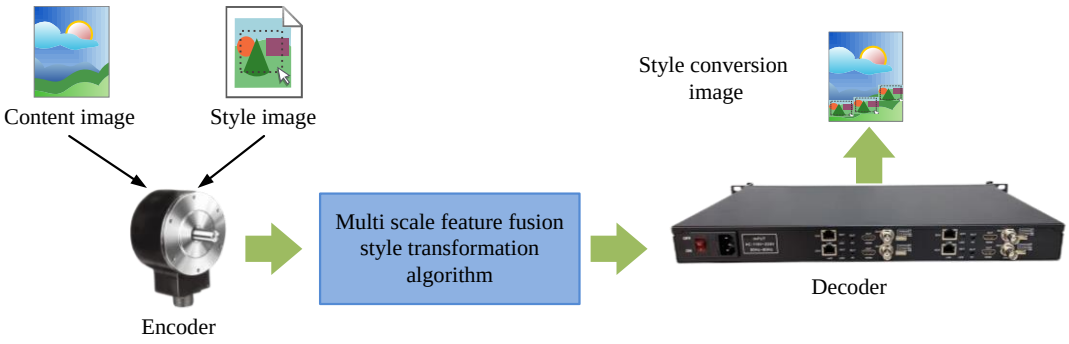

Figure 1: Artistic image style conversion process

In Figure 1, the features extracted by convolution at different levels exhibit different characteristics. As the network depth increases, the extracted overall contour features are more blurred and representative. In view of this, an improved CNN style conversion algorithm has been proposed in the study to address issues such as detail deterioration and local structural distortion caused by input images with complex spatial structures. The algorithm structure includes an encoder, a conversion network, and a decoder. A novel feature detection strategy is used to grasp features with fewer parameters. By decomposing the large convolutional kernels in the conversion network, the parameters are reduced and the conversion speed is improved [15]. The adaptive normalization method is used to process the output of Convolutional Layer (CL) to better preserve the semantic information of content images. This algorithm achieves fast conversion of multiple styles, enhances the structural features, and significantly improves the detail effects. In style conversion, the encoder extracts feature under various CLs. The encoder adopts a pre-trained Visual Geometry Group (VGG) structure, as shown in Figure 2.

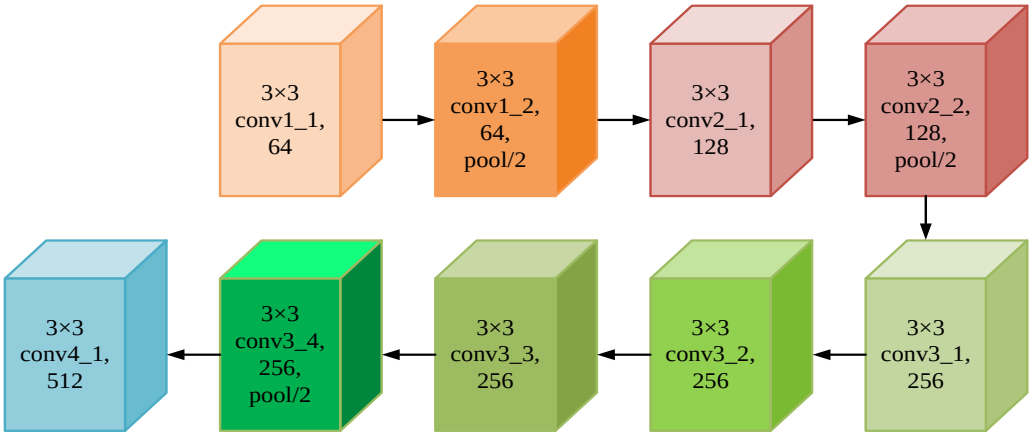

Figure 2: Encoder structure

In Figure 2, from conv1_1 to conv4_1, all convolution kernels have a size of 3×3. Each CL is followed by a Relu activation function. After conv_2, conv2_2, and conv3_4, there is a max pooling layer for down-sampling. In feature fusion, the content feature map and style feature map are output for fusion in the conv2_1, conv3_1, and conv4_1 section to avoid fusing the conv_1 results and prevent affecting the quality of style conversion [16-17]. To reduce parameter calculations, the large convolutional kernel in the CL of the conversion network is decomposed into two 5×5 convolutional kernels instead of the 9×9 convolutional kernel, keeping the receptive field unchanged and increasing the network depth and learning ability.

In the conversion network, an Adaptive Instance Normalization (AdaIN) is introduced to automatically match the feature statistics of content images and style images. The mean and variance of content feature information are aligned with the mean and variance of style feature images to obtain the target feature map h. After AdaIN, the content Loss Function (LF) and Style LF are obtained, as shown in equation (1).

$$\begin{cases} L_{C,AdaIN} = \left\| d\left(g\left(h\right)\right) - h \right\|_2 \\ L_{S,AdaIN} = \left\| \mu\left(G\left(\tilde{y}\right)\right) - \mu\left(G\left(y\right)\right) \right\|_2 + \left\| \sigma\left(G\left(\tilde{y}\right)\right) - \sigma\left(G\left(y\right)\right) \right\|_2 \end{cases} \quad (1)$$

In equation (1), $L_{C,AdaIN}$ and $L_{S,AdaIN}$ are the content LF and style LF, respectively [18]. The overall content perception LF and style perception LF for image style conversion are displayed in equation (2).

$$\begin{cases} L_C\left(x,\tilde{y}\right) = \sum_{l=1}^{L} \frac{1}{H^l W^l N^l} \left\| M^l\left(x\right) - M^l\left(\tilde{y}\right) \right\|_2^2 + \sum_{l=1}^{L} \left\| d\left(\tilde{y}^l\right) - h^l \right\|_2 \\ L_S\left(y,\tilde{y}\right) = \sum_{l=1}^{L} \left(\left\| G^l\left(y\right) - G^l\left(\tilde{y}\right) \right\|_F^2 + \left(\begin{array}{l} \left\| \mu\left(G^l\left(\tilde{y}\right)\right) - \mu\left(G^l\left(y\right)\right) \right\|_2 \\ + \left\| \sigma\left(G^l\left(\tilde{y}\right)\right) - \sigma\left(G^l\left(y\right)\right) \right\|_2 \end{array} \right) \right) \end{cases} \quad (2)$$

In equation (2), $L_C\left(x,\tilde{y}\right)$ and $L_S\left(y,\tilde{y}\right)$ represent the overall content perception LF and style perception LF of image style conversion, respectively. Therefore, the total LF of the entire network training is obtained. It is trained and optimized by the random gradient descent method, as shown in equation (3).

$$L = \alpha L_C + \beta L_S + \gamma L_R \quad (3)$$

In equation (3), α and β represent the weight values of content loss and style loss, respectively. L_R is the regularization term. γ represents the weight value of L_R [19]. Afterwards, the decoder parameters can be derived through the LF. After multiple training, the optimal decoder parameters can be obtained. The style conversion network training and CNN improvement are completed to extract image features. The improved CNN architecture aims to improve the extraction efficiency and quality of image features, as shown in Figure 3.

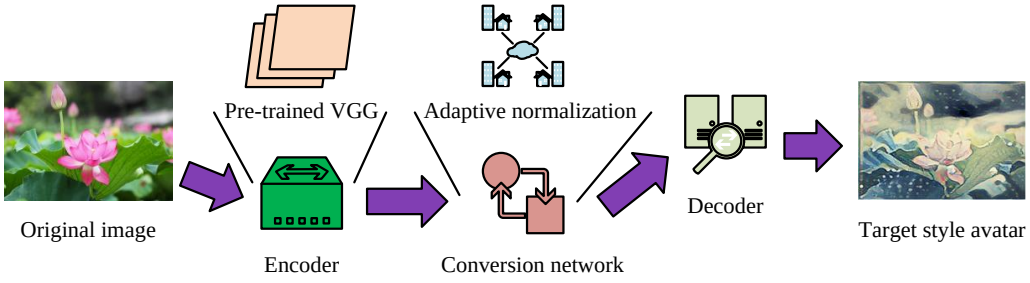

Figure 3: Improved CNN architecture

The architecture includes three main parts: encoder, conversion network and decoder. The encoder uses the pre-trained VGG network structure to extract image features through multiple convolution layers and pooling layers. The conversion network reduces the number of parameters by reducing large convolution kernels, and optimizes the feature representation by adaptive normalization methods. The decoder is responsible for mapping the fused features back into the pixel space to generate the target style image.

3.2 Feature fusion based on the MSFF network

The features extracted by convolutional networks at different levels have different effects. Low level convolution can grasp the detailed information, which is beneficial for expressing local features. High level convolution focuses more on the overall structural features of the image, such as shape and contour. The existing image style conversion algorithms mainly focus on converting high-level features into images. Although this can better express overall features, it may not achieve satisfactory results in terms of local details [20]. Accordingly, the MSFF network is introduced. Taking into account different levels of convolutional features comprehensively, the decoder takes into account both low and high information during the image generation to obtain more satisfactory detail results. The artistic image style conversion based on MSFF network first extracts content and style image features by the encoder, and merges feature maps through MSFF. Finally, the target image is generated through the decoder. The core of MSFF network is to integrate features of different scales to enhance feature expression ability. The network uses convolution kernels with different sizes to extract features in parallel through the multi-scale feature extraction layer, and then concatenation operations are carried out through the MSFF layer to integrate features. The dimensionality reduction layer is used to reduce the number of channels in the fusion layer to avoid dimensional disasters. The MSFF is displayed in Figure 4.

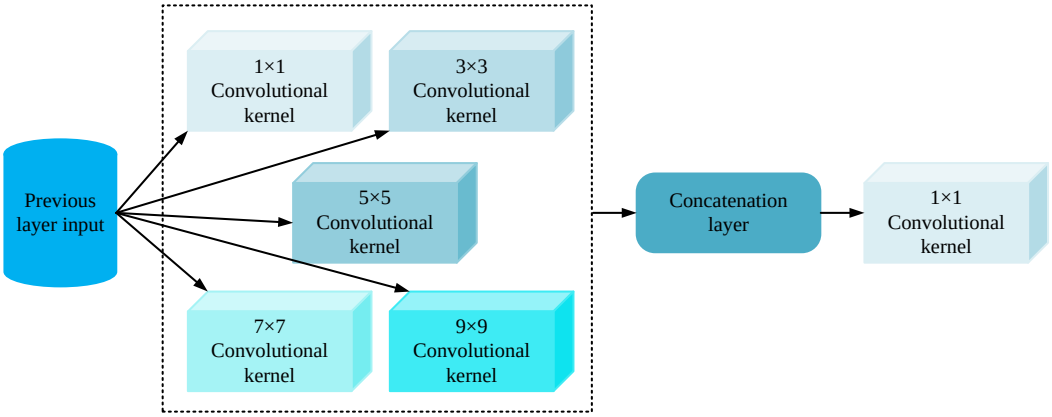

Figure 4: MSFF module structure

From Figure 4, the MSFF module aims to extract features at various scales in the image and fuse them, including a MSFF feature extraction layer, a MSFF layer, and a dimensionality reduction layer. The multi-scale feature extraction layer extracts feature at various scales through multiple convolution kernels of various sizes, among which the 1x1 convolution kernel is used to preserve shallow information to improve the image quality. When selecting the feature extraction scales, a comprehensive consideration should be given to network parameters and over-fitting. The core function of the MSFF module is to extract image features at various scales and integrate these features together [21]. Multiple convolutional kernels with different sizes can extract features from different scales and perform nonlinear representations in the fusion layer. When determining the feature extraction scales, network parameters and over-fitting need to be balanced to achieve the best results. After each CL, the nonlinear mapping ability is enhanced through nonlinear layers. The input of all multi-scale feature extraction layers is X, and there are m CLs in this layer. Different layers have different convolution kernel sizes. Equation (4) represents the i-th CL in the first MSFF module.

$$f_{1_i}(X) = \sigma_{1_i}\left(W_{1_i} * X + B_{1_i}\right) \tag{4}$$

In equation (4), W_{1_i} and B_{1_i} are the weights and biases of the CL, respectively. $*$ refers to the convolution operation. σ_{1_i} is the nonlinear element after the i-th CL, which presented in equation (5).

$$\sigma_{1_i}(x) = \max(0, x) \tag{5}$$

In equation (5), x stands for the input value of the nonlinear element. The MSFF layer is to fuse the feature maps output by multiple scale feature extraction layers to supply the next layer for processing. This layer consists of concatenation operations, which overlay feature maps of various scales and channels together [22]. The channels in the fused feature map are equal to the total channels in each CL of the MSFF extraction layer. The fusion principle is displayed in Figure 5.

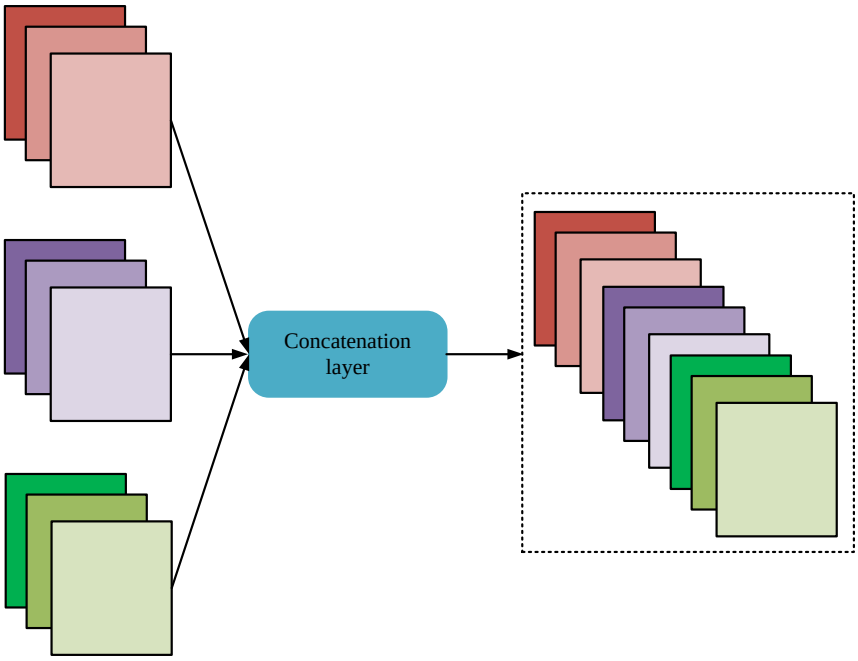

Figure 5: Fusion of MSFF layers

From Figure 5, the MSFF layer mainly integrates three different types of features. Assuming that the multi-scale feature extraction layer has m CLs, the MSFF layer in the first MSFF module is displayed in equation (6).

$$f^1(X)=\sum_{i=1}^{m} f_{1_i}(X)=\sum_{i=1}^{m}\sigma_{1_i}\left(W_{1_i} * X + B_{1_i}\right) \quad (6)$$

In equation (6), X refers to the input value of the multi-scale feature extraction layer. The dimensionality reduction layer is to reduce the MSFF layer's dimensionality, that is, to reduce its channel count. In a multi-feature extraction layer, each scale CL typically requires a certain number of convolution kernels, as different convolution kernels can extract different features. Although there are many channels in the CL at each scale, it may not cause dimensionality issues when used in cascading. However, after parallel use and fusion, the channels in the MSFF layer increase sharply, which may cause dimensional disasters and limit the network size [23]. Therefore, before entering the next multi-feature fusion module, the MSFF layer is dimensionally reduced to reduce the channels and facilitate feature fusion into the next module. The dimensionality reduction layer has a CL with a kernel size of 1×1 and a nonlinear activation unit. The channels in this CL are less than the channels in the multi-feature fusion layer. The 1×1 convolution kernel can retain all information in the multi-feature fusion layer, while reducing the channels in the final output multi-scale feature map, playing a dimensionality reduction role. The dimensionality reduction layer of the first MSFF module is displayed in equation (7).

$$F^1=\sigma\left(W_{1_{m+1}} * f^1(X)+B_{1_{m+1}}\right) \quad (7)$$

In equation (7), $f^1(X)$ stands for the output value of the multi-feature fusion layer. $W_{1_{m+1}}$ and $B_{1_{m+1}}$ stand for the weights and biases of the CL in the dimensionality reduction layer, respectively. σ is used to describe the operation of non-linear activation units in the dimensionality reduction layer. The MSFF module utilizes convolution kernels of various sizes to extract multi-scale features of images. Multiple filter sets with various sizes extract and fuse multi-scale information from images [24]. After each CL, nonlinear activation units are introduced to learn the nonlinear mapping relationship between input and class labels. The dimensionality reduction operation avoids the curse of dimensionality caused by the increase in the channels in the MSFF layer and the limitation on network size, allowing modules to be used in multi-level concatenation. It is suitable for MSFF network artistic image style conversion tasks. The MSFF module performs better when used in cascading. The first and l-th multi-scale feature modules are shown in equation (8).

$$\begin{cases} F^1 = \sigma\left(W_{1_{m+1}} * \sum_{i=1}^{m} \sigma_{l_i}\left(W_{1_i} * X + B_{1_i}\right) + B_{1_{m+1}}\right) \\ F^l = \sigma\left(W_{l_{m+1}} * \sum_{i=1}^{m} \sigma_{l_i}\left(W_{l_i} * F^{l-1} + B_{l_i}\right) + B_{l_{m+1}}\right) \end{cases} \quad (8)$$

In equation (8), F^1 and F^l represent the first and l-th multi-scale feature modules, respectively. F^{l-1} stands for the output value of the previous MSFF module. $W_{l_{m+1}}$ and $B_{l_{m+1}}$ represent the weights and biases of the CL in the dimensionality reduction layer, respectively. The MSFF network includes multiple cascaded multi-feature fusion modules and a CL with a kernel size of 3×3, mainly achieving artistic image style conversion. These modules map the multi-scale features of one artistic image style to another artistic image style. Finally, a multi-scale feature of an artistic image style is transformed into the desired artistic image through 3×3 CLs [25-26]. Assuming that L MSFF modules and a reconstruction layer (i.e. convolutional layer) are used in the network, the mathematical expressions of the first L modules are shown in equation (9).

$$\begin{cases} f^1(X) = F^1 \square\ X, l = 1 \\ f^l(X) = F^l \square\ f^{l-1}, l = 2..L \end{cases} \quad (9)$$

In equation (9), $\square$ is used to describe the set of feature extraction, representation, and dimensionality reduction operations of the MSFF module on the input. The final reconstruction layer, also known as the CL, is responsible for fusing features at different scales together, as shown in equation (10).

$$F(X) = W_{L+1} * f^L(X) + B_{L+1} \quad (10)$$

In equation (10), $f^L(X)$ stand for the output of the L-th MSFF module in the network, thus achieving the artistic image style conversion of the MSFF network. The overall structure of the Improved CNN-Multi-Scale Feature Fusion Network (ICNN-MFFN) is presented in Figure 6.

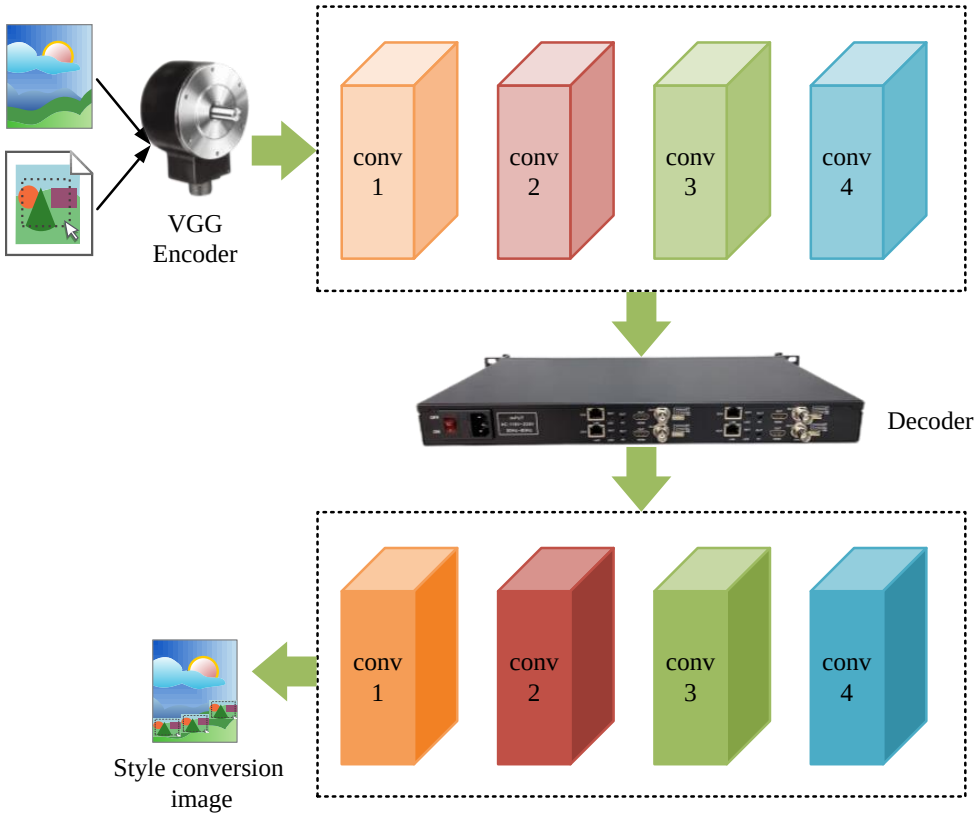

Figure 6: MSFF network structure

From Figure 6, the MSFF network structure utilizes feature information from different scales. An appropriate fusion strategy effectively combines this feature information into a network structure. The features at different scales are fused, which can better generalize targets, thereby more effectively achieving tasks including image segmentation and object detection. This network structure can effectively integrate information from different scales in images, improving the performance and robustness.

3.3 Artistic image style conversion based on introduced attention mechanism

The visual attention mechanism is an important method for humans to obtain key information. In complex scenes, humans prioritize capturing the target area and concentrate their attention to obtain more detailed information. This mechanism helps humans suppress useless information and quickly obtain information on key areas. The attention mechanism in computer vision is comparable to humans, focusing on key regions in images [27]. The visual attention mechanism based on deep learning is implemented through a mask mechanism, using weights to mark important features of the image, and forming attention through neural network learning. Soft attention focuses on regions or feature channels, and obtains attention weights through neural network learning, while strong attention focuses on pixel level details. Each pixel may generate attention, which is typically achieved through reinforcement learning.

The channel attention mechanism in CNNs is used to measure the importance of each feature channel, stimulate important channel information, suppress useless channel information, and highlight key areas in the image. This mechanism can improve the deep CNNs [28]. An Efficient Channel Attention Network (ECA) network is proposed. The ECA is displayed in Figure 7.

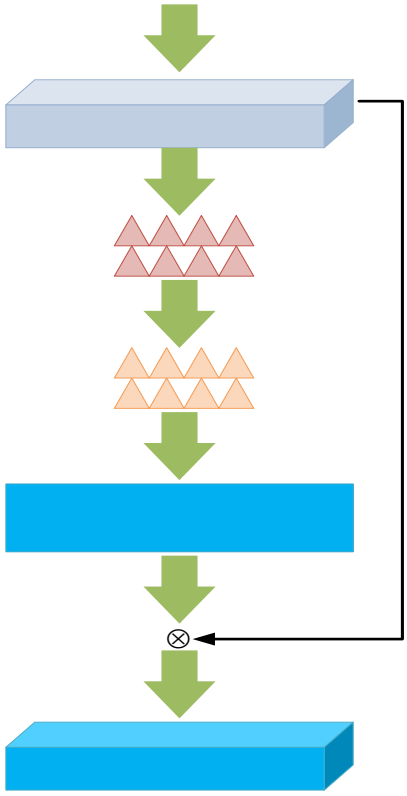

Figure 7: ECA network

From Figure 7, the ECA network emphasizes the importance of direct correspondence between channels and weights by independently learning the weights of each channel, while avoiding dimensionality reduction operations. Meanwhile, by designing a one-dimensional convolutional kernel with adaptive size selection, cross channel information exchange is achieved, which improves the effectiveness of channel attention and ensures both performance and model efficiency [29-30]. The channel attention module first performs a compression operation, as shown in equation (11).

$$z = \frac{1}{T \times N} \sum_{i=1}^{T} \sum_{j=1}^{N} u_c(i, j) \tag{11}$$

In equation (11), z represents the compressed feature map. $u(c)$ represents the spatial graph convolution output data. Excitation conversion is performed on the feature graph, as shown in equation (12).

$$s = \sigma\left(W_z \delta\left(W_1 z\right)\right) \tag{12}$$

In equation (12), s represents the converted feature map. σ represents the *Sigmoid* activation function. δ stands for the $\mathrm{Re}\,Lu$ activation function. The ICNN-MFFN-Attention Mechanism (ICNN-MFFN-AM) is designed. Figure 8 displays the structure.

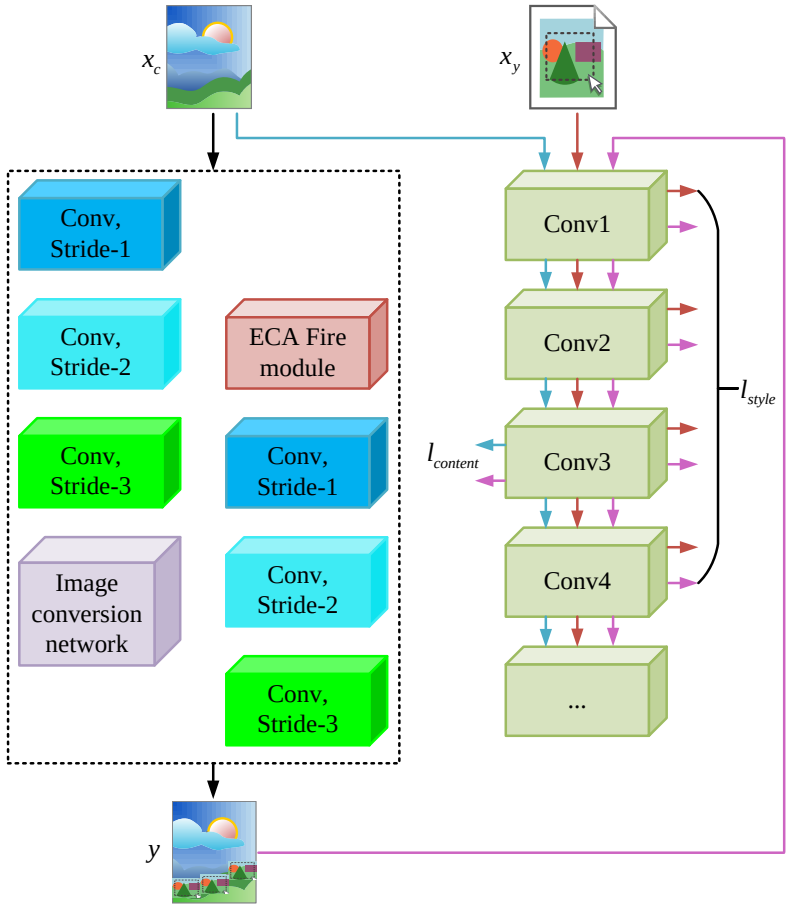

Figure 8: Lightweight image style conversion algorithm with attention mechanism

From Figure 8, the network structure of the algorithm includes an image conversion network and a content and style representation network. The image conversion network has an encoder, an ECA-Fire module, and a decoder. The encoder-decoder structure is applied to reduce computational complexity and increase receptive field. The main body is composed of multiple ECA-Fire modules. The content and style representation network is a pre-trained VGG-16 network used to grasp content and style features of images, and define content loss and style loss. During the training, based on pre-selected images read from the dataset, they are input into the network to calculate content and style loss. The image conversion network parameters are updated through backpropagation. A lightweight style transfer model with a specific style is ultimately generated [31]. The image conversion network consists of five ECA-Fire modules, with residual connections used between the first and second modules, as well as between the fourth and fifth modules. The network uses a abundant small convolution kernels with sizes of 3*3 or 1*1, while the first and last layers use 9*9 convolution kernels. The input is a color 3-channel content image with a resolution of 256*256. Down-sampling is achieved through a CL with a stride size of 2. The corresponding up-sampling is achieved through a CL with a stride size of 1/2 to adjust the channels and resolution of the output image to match the input image. This operation reduces computational complexity, which can effectively increase the size of the receptive field. Down-sampling can conveniently use larger CLs for feature extraction, and the increase in effective receptive field also helps to improve the quality of image style conversion.

The content and style representation network is essentially a pre-trained VGG-16 network used to grasp features from content images, converted images, and style images. These three types of images are input into the network and their activation responses in a certain layer of the network are extracted, which are called feature maps [32]. Content loss is not an accurate pixel level loss, but the mean square error in the feature maps extracted from the converted image and the content image in the network, representing their content similarity. After the CLs in the content and style representation network, the feature map size is represented as $C_j \times H_j \times W_j$. C_j is the channels, H_j is the height, and W_j is the width. The LS is defined as the mean square error between the features of the content image and the features of the converted image, as expressed in equation (13).

$$l_{content}(x_c, y) = \frac{1}{H_j \times W_j \times C_j} \left\| \varphi_j(x_c) - \varphi_j(y) \right\|_2^2 \quad (13)$$

In equation (13), $l_{content}$ represents the CLF. x_c and y are content images and conversion images. φ_j represents the j-th CL of the content and style network φ. Style loss is applied to constrain the distinctions in the converted image y and the style image x_s, aiming to preserve style features such as color, texture, and common patterns. The Gram matrix is defined to represent the style information of the feature map, as expressed in equation (14).

$$G_j^{\varphi}(x)_{c,c'} = \frac{1}{H_j \times W_j \times C_j} \sum_{h=1}^{H_j} \sum_{w=1}^{W_j} \varphi_j(x)_{h,w,c} \varphi_j(x)_{h,w,c'} \quad (14)$$

In equation (14), $\varphi_j(x)$ represents a C-dimensional vector. All elements are composed of a set of feature maps $H_j \times W_j$ to form a row vector. It meas that $\varphi_j(x)$ is converted into a two-dimensional vector ϕ of $C_j \times H_j W_j$, and then solved with its transposed inner product to obtain the Gram matrix. The diagonal elements of the Gram matrix represent the feature map information itself. Other elements represent the correlation information between different feature maps, which can be used to measure the importance of features within themselves and between different features. The square of the difference norm in the Gram matrices of the converted image y and the style image x_s is calculated. The differences calculated at each layer are added to obtain the final style loss, as expressed in equation (15).

$$l_{style}(x_s, y) = \left\| G_j^{\varphi}(x_s) - G_j^{\varphi}(y) \right\|_F^2 \quad (15)$$

The total LS is composed of a linear combination of content loss and style loss, expressed as equation (16).

$$l_{total}(x_c, x_s, y) = \lambda_1 l_{content}(x_c, y) + \lambda_2 l_{style}(x_s, y) \quad (16)$$

In equation (16), λ_1 and λ_2 represent the weight coefficients of content loss and style loss, respectively. The total LS is iteratively optimized, aiming to minimize the total loss value and ultimately generate a lightweight style conversion model with a specific style. The stylized images generated by this model are comparable in quality to other models, but have advantages in terms of size and speed, making it more convenient to achieve real-time image style conversion.

4 Analysis of artistic image style conversion based on MSFF network

To analyze the effect of the research method on artistic image style conversion, it is first compared with other advanced methods and applied to artistic image style conversion. The research method performs well in various artistic image style conversions, with good visualization results.

4.1 Algorithm Performance Analysis

To analyze the artistic image style conversion effect of the research method, the performance of ICNN-MFFN-AM is first compared. It is compared with ICNN-MFFN and Cycle-Consistent Generative Adversarial Networks (CycleGAN). Among them, CycleGAN can achieve image conversion of different styles through adversarial training between two generators and two discriminators. The experimental environment for all algorithms is consistent, as displayed in Table 2.

Table 2: Experimental environment

Number	Software and hardware projects	Specific information
(1)	CPU	12th Gen Intel(R) Core(TM) i5-12400F
(2)	GPU	NVIDA GeForce RTX 3060 Ti
(3)	RAM	12G
(4)	Operating system	Windows 10 64 bit operating system
(5)	Anaconda version	4.6.11
(6)	Hard disk	1TB
(7)	CUDA version	9.0
(8)	Python version	3.7.3

The experiment uses a memory hardware parameter of 12GB DDR4 RAM, which ensures the memory requirement when processing large-scale image datasets. The GPU hardware model is NVIDIA GeForce RTX 3060 Ti, which provides efficient parallel computing power to accelerate the training and reasoning process of deep learning models. The Anaconda version 4.6.11 is used to manage the Python environment and dependency packages, with Python version 3.7.3 as the basis for programming language and scripting, and PyTorch 0.4.0 deep learning framework for building and training CNN models. The Content image dataset is MSCOCO 2017, which is widely used for computer vision tasks that contains 118,287 images of everyday life scenes. The

Style Image dataset of WikiArt is adopted. WikiArt is a dataset containing many artistic style images, downloaded from Kaggle, with a quantity of approximately 80000. The preprocessing steps ensure the consistency and availability of the dataset, ensure that all images are in RGB format, adjust the size uniformly to 256x256 pixels, remove damaged or inconsistent images, establish a pair of content images and style images, and ensure that there are enough samples for style conversion training. A two-sample t test is performed on the experimental results to verify whether the performance difference between the proposed method and the existing method is statistically significant. 95% confidence intervals are calculated to evaluate the reliability of the experimental results. To ensure the reliability and consistency of the experimental results, a fixed random seed is used in the experiment to repeat the random initialization process. The experiment is repeated for many times under the same conditions, and the mean value and standard deviation of the results are calculated. In the comparative experiment, all algorithms have the same backbone network VGG-16. Four GPU servers are used to compute nodes. Each node processes one type of image. The environment configuration of each node is consistent. The usage and loss of each node are different, resulting in differences in model training time. To control variables, the detection speed of the training model is tested at the same computing node to ensure the rigor. 300 images are randomly selected for testing. The batch size is 2. The learning rate is 1×10-4, with a total of 50000 iterations. The time for converting images using three methods is shown in Figure 9.

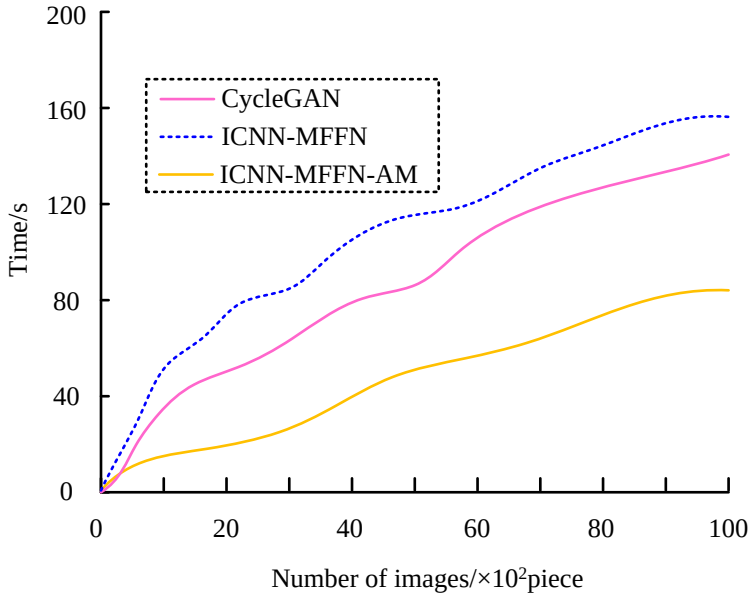

Figure 9: Artistic image style conversion time

From Figure 9, as the images increased, the conversion time of the three algorithms also has increased. However, the conversion time of ICNN-MFFN-AM was significantly shorter than that of CycleGAN and ICNN-MFFN. When converting 8000 images, the conversion time of ICNN-MFFN-AM, CycleGAN, and ICNN-MFFN was 61.2s, 118.5s, and 137.6s, respectively. ICNN-MFFN-AM had higher efficiency in artistic image style conversion. The CPU and memory usage during the artistic image style conversion process using the three algorithms are shown in Figure 10.

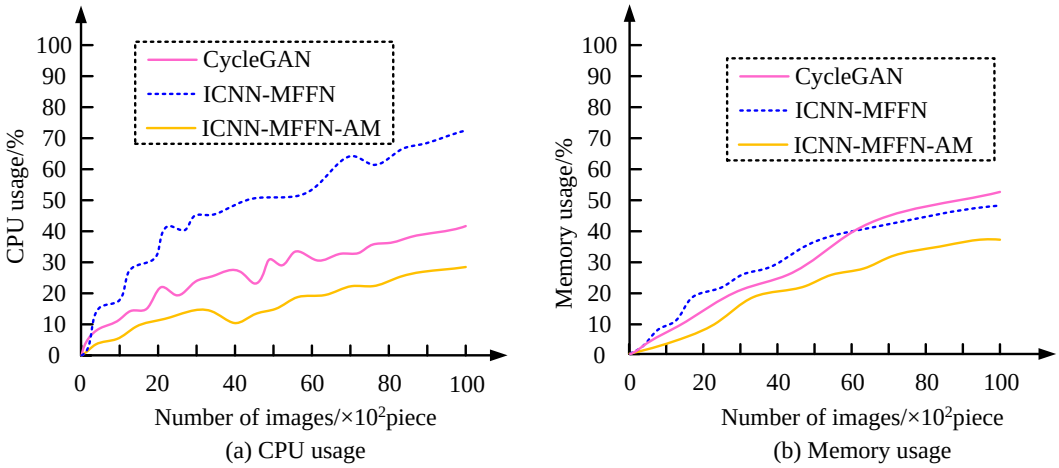

Figure 10: CPU usage and memory usage during the artistic image style conversion

Figures 10 (a) and 10 (b) respectively show the CPU and memory usage for converting image artistic styles. From Figure 10, when there were more converted images, the CPU and memory usage also increased and gradually stabilized. Overall, ICNN-MFFN-AM had lower CPU and memory usage, which meant that ICNN-MFFN-AM had the best artistic image style conversion performance.

4.2 The quality conversion results of different artistic image styles

Artistic image style conversion is a digital image processing technique. It uses computer vision and deep learning algorithms to re-render an image (called a "content image") with the artistic style of another image (called a "style image"). The effects of converting ordinary photos into Van Gogh style, Ukiyo-e style, Monet style, and Cézanne style are compared. The Structural Similarity Index (SSI) and Peak Signal-to-Noise Ratio (PSNR) are used to assess the similarity and distortion in the generated and the source domain image. Firstly, an ablation experiment is conducted on Van Gogh's artistic style conversion. The results are shown in Figure 11.

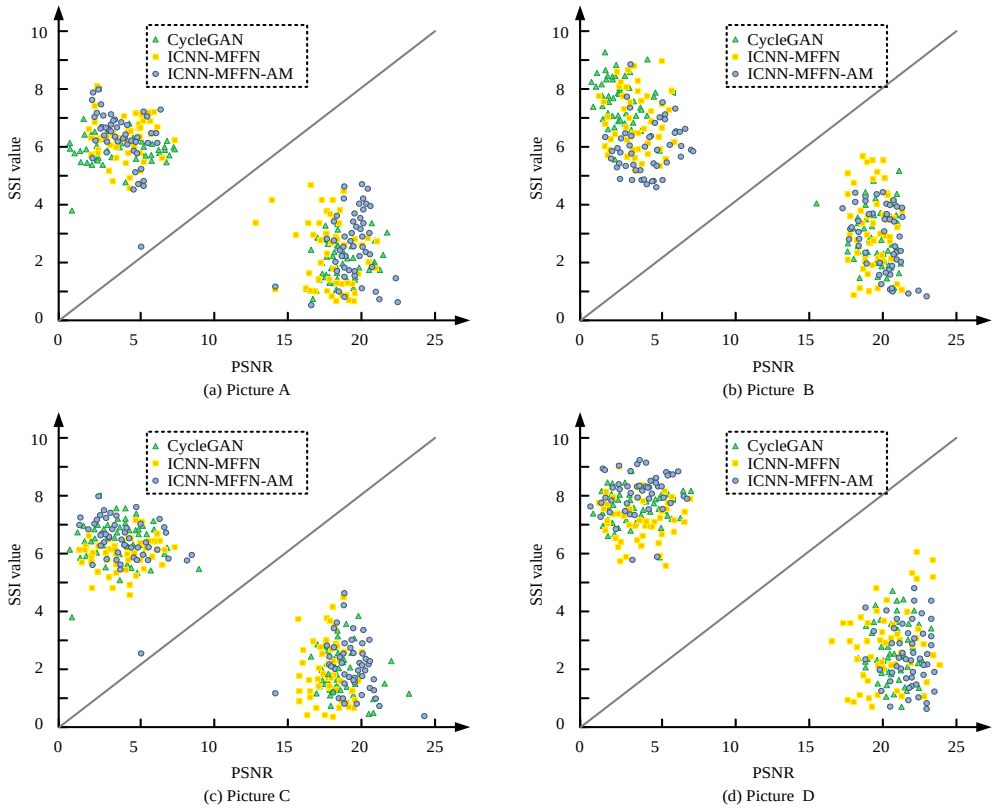

Figure 11: The result of Van Gogh style conversion

Figures 11 (a), 11 (b), 11 (c), and 11 (d) represent the average PSNR and SSI values of converting images A, B, C, and D into Van Gogh artistic style images using three algorithms 50 times, respectively. PSNR evaluates the distortion by comparing the ratio between mean square error and the maximum pixel value, with higher values indicating better image quality. SSI is based on similarities in brightness, contrast, and structure. The larger values indicating that the generated image is closer to the source domain image. From Figure 11, the PSNR and SSI values of the three algorithms were ICNN-MFFN-AM, CycleGAN, and ICNN-MFFN in descending order. ICNN-MFFN-AM preserved high quality of the content images in converting the four content images into Van Gogh artistic style images. The error was smaller, which could better reflect the subjective evaluation of image quality, while retaining the structural information. The style conversion of Ukiyo-e is shown in Figure 12.

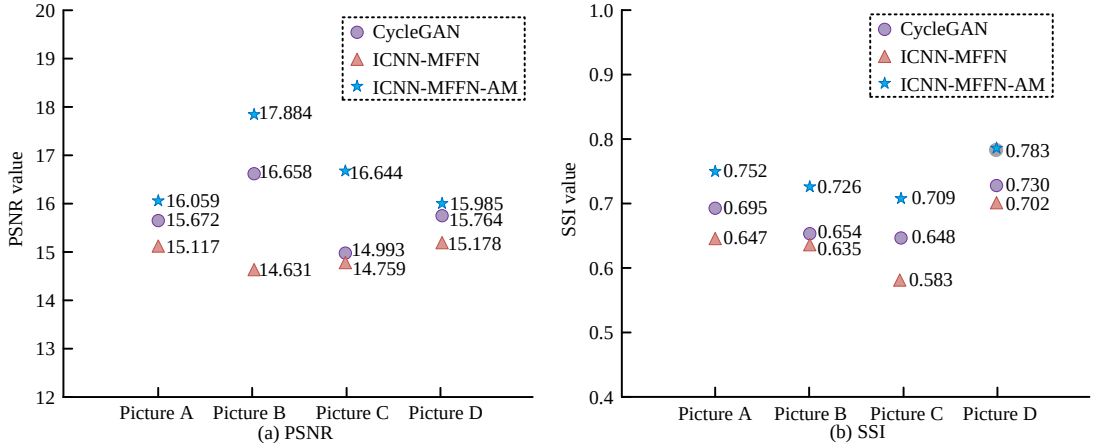

Figure 12: Result of Ukiyo-e artistic style conversion

Figures 12 (a) and 12 (b) respectively represent the PSNR and SSI values of three algorithms for converting four content images into Ukiyo-e style images. From Figure 12, the PSNR values of the four content images converted by ICNN-MFFN-AM were 16.259, 17.884, 16.644 and 15.985, respectively. The standard deviation of the PSNR values for ICNN-MFFN-AM was 0.131. The SSI values were 0.752, 0.756, 0.709 and 0.783, respectively, and the standard deviation of SSI values of ICNN-MFFN-AM was 0.193. The PSNR and SSI values for ICNN-MFFN-AM were significantly higher than those of CycleGAN and ICNN-MFFN, and the standard deviations were lower than those of CycleGAN and ICNN-MFFN. The result of Monet style conversion is shown in Figure 13.

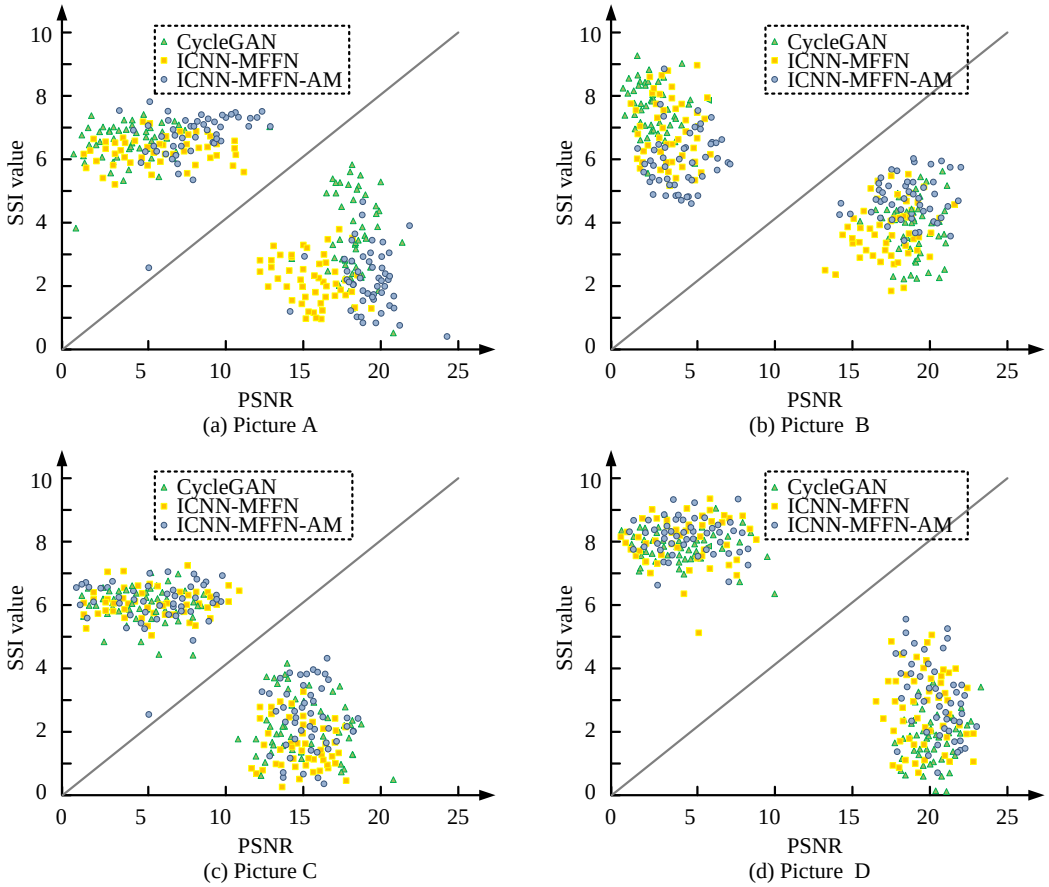

Figure 13: Monet style conversion results

Figures 13 (a), 13 (b), 13 (c), and 13 (d) present the average PSNR and SSI values of images A, B, C, and D converted into Monet artistic style images using three algorithms 50 times, respectively. From Figure 13, compared with CycleGAN and ICNN-MFFN, ICNN-MFFN-AM also had higher PSNR and SSI values, indicating that CycleGAN retained more information. It had smaller errors when converting content images into Monet artistic style images. The result of the Cézanne style conversion is shown in Figure 14.

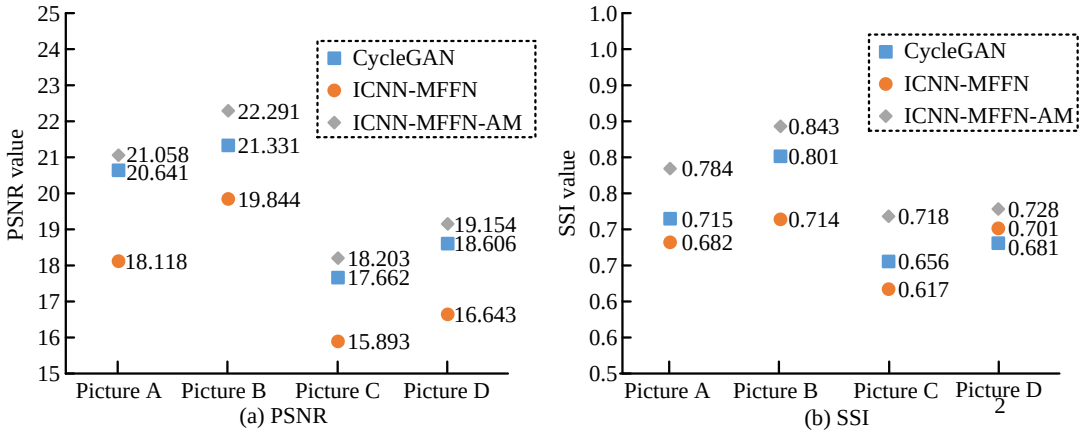

Figure 14 Cézanne style conversion results

Figures 14 (a) and 14 (b) respectively represent the PSNR and SSI values of three algorithms for converting four content images into Cézanne style images. From Figure 14, the PSNR values of image B converted by ICNN-MFFN-AM, CycleGAN and ICNN-MFFN were 22.291 (P<0.05), 21.331 (P<0.05) and 19.844 (P<0.05), respectively. The standard deviations of PSNR values for ICNN-MFFN-AM, CycleGAN and ICNN-MFFN were

0.122, 0.231 and 0.063, respectively. The SSI values of ICNN-MFFN-AM, CycleGAN and ICNN-MFFN were 0.843 ($P<0.05$), 0.801 ($P<0.05$) and 0.714 ($P<0.05$), respectively. The standard deviations of SSI values for ICNN-MFFN-AM, CycleGAN and ICNN-MFFN were 0.041, 0.063 and 0.084, respectively. ICNN-MFFN-AM was more likely to inherit the color and texture information of the source image when converting styles, achieving the best image conversion effect. The above research results indicate that ICNN-MFFN-AM has superior image conversion performance and significant advantages in image conversion tasks. The visualization results of ICNN-MFFN-AM converting four content images into Van Gogh style, Ukiyo-e style, Monet style, and Cézanne style are shown in Figure 15.

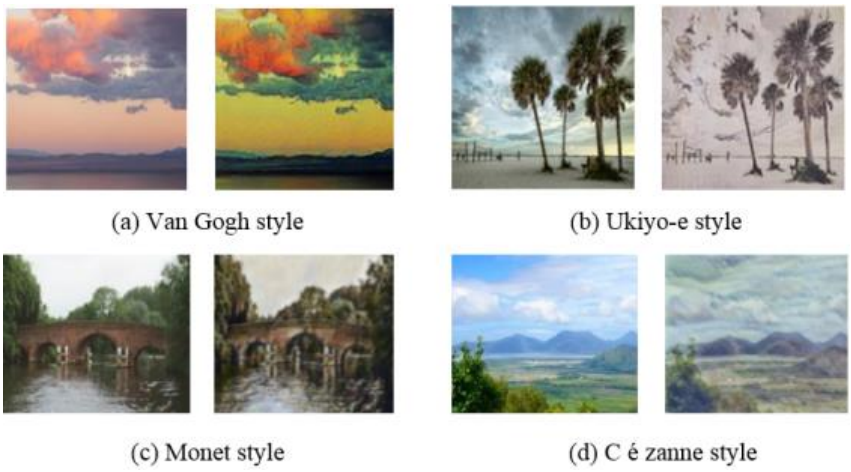

Figure 15: Visualization results of artistic image style conversion

Figures 15 (a), 15 (b), 15 (c), and 15 (d) present the results of ICNN-MFFN-AM converting four content images into Van Gogh style, Ukiyo-e style, Monet style, and C é zanne style. From Figure 15, the ICNN-MFFN-AM could naturally convert content images into different artistic styles while retaining the original content.

5 Conclusion

The artistic image style conversion technology aims to convert the style of one image into another image, which has broad application prospects. However, existing methods have certain limitations in processing large-scale image data and performing style conversion while preserving content. This study was based on the MSFF network and attention mechanism to convert artistic image styles. The main contributions of the research included an improved CNN structure, which enhanced the feature extraction capability through adaptive normalization and multi-scale feature fusion technology. An efficient channel attention mechanism was introduced, which enabled the network to focus more on the key features of the image, and improved the naturalness and accuracy of style conversion. The results showed that when converting 8000 images, the conversion time of ICNN-MFFN-AM, CycleGAN, and ICNN-MFFN was 61.2s, 118.5s, and 137.6s, respectively. The ICNN-MFFN-AM had higher efficiency in artistic image style conversion, and lower CPU and memory usage. The PSNR and SSI values of the three algorithms in descending order were ICNN-MFFN-AM, CycleGAN, and ICNN-MFFN. After converting content images into different artistic styles, ICNN-MFFN-AM could naturally convert them into the required artistic style while retaining the original content. The artistic image style conversion method based on MSFF network proposed in the study has achieved significant improvements in image conversion quality and speed. In the future research, from the perspective of practical application, the research will study how to combine user interaction and allow users to guide the style conversion process to generate an image style that is more in line with user expectations.

6 Discussion

The proposed artistic image style conversion realizes efficient style conversion of content images through multi-scale feature fusion network and attention mechanism. First of all, compared with the unsupervised dense network proposed by Zhou et al. [4], the research method is more refined in feature extraction. The introduced attention mechanism pays more attention to the important features of artistic images, so as to better retain the details of content images in the style conversion. Compared with the traditional CNN style conversion algorithm, this study significantly improves the detail effect and overall quality of style conversion by improving the CNN structure, and using adaptive normalization and multi-scale feature fusion techniques. In addition, by reducing the large convolution kernel in the conversion network, the number of parameters is

reduced, the conversion speed is increased, and the slow operation speed in the traditional method is solved. Compared with advanced methods such as CycleGAN, the research method showed lower consumption in conversion time, CPU usage, and memory usage, indicating higher efficiency. Thanks to the design of the multi-scale feature fusion network, it allows the network to consider local details and overall structure at different levels simultaneously, thus achieving a better balance in the task of artistic image style conversion. The multi-scale feature fusion network combined with attention mechanism provides a new perspective in the artistic image style conversion. This combination not only improves the quality and efficiency of style conversion, but also achieves a deeper understanding and expression of artistic style through more detailed feature extraction and fusion.

7 Fundings

The research is supported by: The Philosophy and Social Science Research Project in Universities of Anhui Province in 2023, (No. 2023AH040158); Horizontal scientific research project of Hefei Normal University in 2023, (No. HXXM2023018); The Project of Supporting Outstanding Young Talents in Universities of Anhui Province in 2019, (No. gxyq2019065).

References

[1] Li Dong, Zheng Liang, and Yue Wang. Graph convolutional network-based image matting algorithm for computer vision applications. IET image processing, 16(10):2817-2825, 2022. https://doi.org/10.1049/ipr2.12528

[2] Gangming Zhao, Kongming Liang, Chengwei Pan, Fandong Zhang, Xianpeng Wu, Xinyang Hu, and Yizhou Yu. Graph convolution based cross-network multiscale feature fusion for deep vessel segmentation. IEEE transactions on medical imaging, 42(1):183-195, 2023. https://doi.org/10.1109/TMI.2022.3207093

[3] Hao-Hsiang Yang, Kuan-Chih Huang, and Wei-Ting Chen. LAFFNet: A lightweight adaptive feature fusion network for underwater image enhancement. IET image processing, 15(3):774-785, 2021. https://doi.org/10.48550/arXiv.2105.01299

[4] Ding Zhou, Xin Jin, Qian Jiang, Li Cai, Shin-jye Lee, and Shaowen Yao. MCRD-Net: An unsupervised dense network with multi-scale convolutional block attention for multi-focus image fusion. IET image processing, 16(6):1558-1574, 2022. https://doi.org/10.1049/ipr2.12430

[5] Lijia Deng, Shui-Hua Wang, and Yu-Dong Zhang. ELMGAN: A GAN-based efficient lightweight multi-scale-feature-fusion multi-task model. Knowledge-based systems, 252:109434.1-109434.12, 2022. https://doi.org/10.1016/j.knosys.2022.109434

[6] Yongjie Wang, Wei Zhang, and Yanyan Liu. Multi-scale feature fusion network for person re-identification. IET image processing, 14(17):4614-4620, 2020. https://doi.org/10.1049/iet-ipr.2020.0008

[7] Luyang Wang, Yun Li, Sifan Peng, Xiao Tang, and Baoqun Yin. Multi-level feature fusion network for crowd counting. IET computer vision, 15(1):60-72, 2021. https://doi.org/10.1049/cvi2.12012

[8] Jinyue Shen, Zhouzhou Zheng, Yingwei Sun, Mengmeng Zhao, Yankang Chang, Yuyi Shao, and Yan Zhang. HAMNet: Hyperspectral image classification based on hybrid neural network with attention mechanism and multi-scale feature fusion. International journal of remote sensing, 43(11/12):4233-4258, 2022. https://doi.org/10.1080/01431161.2022.2109222

[9] Yanan Sun, Bing Xue, Mengjie Zhang, and Gary G. Yen. Evolving deep convolutional neural networks for image classification. IEEE transactions on evolutionary computation, 24(2):394-407, 2020. https://doi.org/10.1109/TEVC.2019.2916183

[10] Ying Bi, Bing Xue, and Mengjie Zhang. A divide-and-conquer genetic programming algorithm with ensembles for image classification. IEEE transactions on evolutionary computation, 25(6):1148-1162, 2021. https://doi.org/10.1109/TEVC.2021.3082112

[11] Yunping Zheng, Xiangpeng Li, and Mudar Sarem. Fast fractal image compression algorithm using specific update search. IET image processing, 14(9):1733-1739, 2020. https://doi.org/10.1049/iet-ipr.2019.0522

[12] Osama A.S. Alkishriwo. Image compression using adaptive multiresolution image decomposition algorithm. IET image processing, 14(14):3572-3578, 2020. https://doi.org/10.1049/iet-ipr.2019.1699

[13] Sunil L. Tade, and Vibha Vyas. Hybrid deep emperor penguin classifier algorithm-based image quality assessment for visualisation application in HDR environments. IET image processing, 14(11):2579-2587, 2020. https://doi.org/10.1049/iet-ipr.2019.1371

[14] Jiawei Yuan, Hai-Lin Liu, Yew-Soon Ong, and Zhaoshui He. Indicator-based evolutionary algorithm for solving constrained multi-objective optimization problems. IEEE transactions on evolutionary computation, 26(2):379-391, 2022. https://doi.org/10.1109/TEVC.2021.3089155

[15] Guilherme Paim, Hussam Amrouch, Leandro M. G. Rocha, Brunno Abreu, Eduardo Antônio César da Costa, Sergio Bampi, Jörg Henkel. A framework for crossing temperature-induced timing errors underlying hardware accelerators to the algorithm and application layers. IEEE transactions on computers, 71(2):349-363, 2022. https://doi.org/10.1109/TC.2021.3050978

[16] Zhenshou Song, Handing Wang, Cheng He, and

Yaochu Jin. A kriging-assisted two-archive evolutionary algorithm for expensive many-objective optimization. IEEE transactions on evolutionary computation, 25(6):1013-1027, 2021. https://doi.org/10.1109/TEVC.2021.3073648

[17] Dawei Zhan, and Huanlai Xing. A fast kriging-assisted evolutionary algorithm based on incremental learning. IEEE transactions on evolutionary computation, 5(5):941-955, 2021. https://doi.org/10.1109/TEVC.2021.3067015

[18] Abhinav Tomar, Lalatendu Muduli, and Prasanta K. Jana. A fuzzy logic-based on-demand charging algorithm for wireless rechargeable sensor networks with multiple chargers. IEEE transactions on mobile computing, 20(9):2715-2727, 2021. https://doi.org/10.1109/TMC.2020.2990419

[19] Jian-Yu Li, Zhi-Hui Zhan, Hua Wang, and Jun Zhang. Data-driven evolutionary algorithm with perturbation-based ensemble surrogates. IEEE transactions on cybernetics, 51(8):3925-3937, 2021. https://doi.org/10.1109/TCYB.2020.3008280.

[20] Hanbo Zheng, Yonghui Sun, Xinghua Liu, Calvin Laurent Tcheteu Djike, Jinheng Li, Yang Liu, Jianchao Ma, Kai Xu, and Chaohai Zhang. Infrared image detection of substation insulators using an improved fusion single shot multibox detector. IEEE transactions on power delivery, 36(6):3351-3359, 2021. https://doi.org/10.1109/TPWRD.2020.3038880

[21] Zi-Han Zhang, Xiao-Jun Wu, and Tianyang Xu. FPNFuse: A lightweight feature pyramid network for infrared and visible image fusion. IET image processing, 16(9):2308-2320, 2022. https://doi.org/10.1049/ipr2.12473

[22] Wenjun Tan, Pan Liu, Xiaoshuo Li, Shaoxun Xu, Yufei Chen, and Jinzhu Yang. Segmentation of lung airways based on deep learning methods. IET image processing, 16(5):1444-1456, 2022. https://doi.org/10.1049/ipr2.12423

[23] Kavita Bhosle, and Vijaya Musande. Evaluation of deep learning CNN model for recognition of devanagari digit. Artificial intelligence and applications, 1(2):114-118, 2023. https://doi.org/10.47852/bonviewAIA3202441

[24] Rui Guo, Yong Zhou, Jiaqi Zhao, Yiyun Man, Minjie Liu, Rui Yao, and Bing Liu. Point cloud classification by dynamic graph CNN with adaptive feature fusion. IET computer vision, 15(3):235-244, 2021. https://doi.org/10.1049/cvi2.12039

[25] Padmaprabha Preethi, and Hosahalli Ramappa Mamatha. Region-based convolutional neural network for segmenting text in epigraphical images. Artificial intelligence and applications, 1(2):119-127, 2023. https://doi.org/10.47852/bonviewAIA2202293

[26] Zhong Qu, Xue Shang, Shu-Fang Xia, Tu-Ming Yi, and Dong-Yang Zhou. A method of single-shot target detection with multi-scale feature fusion and feature enhancement. IET image processing, 16(6):1752-1763, 2022. https://doi.org/10.1049/ipr2.12445

[27] Zunlin Fan, Naiyang Guan, Zhiyuan Wang, Longfei Su, Jiangang Wu, and Qianchong Sun. Unified framework based on multiscale transform and feature learning for infrared and visible image fusion. Optical engineering, 60(12):123102-1-123102-16, 2021. https://doi.org/10.1117/1.OE.60.12.123102

[28] Zhilin He. Improved genetic algorithm in multi-objective cargo logistics loading and distribution. Informatica, 47(2), 2023. https://doi.org/10.31449/inf.v47i2.3958

[29] Ziyu Chen, Huaiyu Zhuang, Jia Han, Yani Cui, and Jiaxian Deng. Multi-scale single image dehazing based on the fusion of global and local features. IET image processing, 16(8):2049-2062, 2022. https://doi.org/10.1049/ipr2.12467

[30] Wen Yang, Ming Zhan, Zhijun Huang, and Wei Sha. Design and development of mobile terminal application based on Android. Informatica, 47(2), 2023. https://doi.org/10.31449/inf.v47i2.4008

[31] Shuhui Zhang, Chenglin Zheng, and Xi Chen. SyPSE: A symbolic computation toolbox for process systems engineering part I- architecture and algorithm development. Industrial & engineering chemistry research, 60(45):16304-16316, 2021. 10.1021/acs.iecr.1c02151

[32] Syed Ahmed Nadeem, Eric A. Hoffman, Jessica C. Sieren, Alejandro P. Comellas, Surya P. Bhatt, Igor Z. Barjaktarevic, Fereidoun Abtin, and Punam K. Saha. A CT-based automated algorithm for airway segmentation using freeze-and-grow propagation and deep learning. IEEE transactions on medical imaging, 40(1):405-418, 2021, https://doi.org/10.1109/TMI.2020.3029013