International Journal of Multiphysics
Volume 18, No. 2, 2024
ISSN: 1750-9548

Landscape Painting Style Transfer and Feature Extraction Model based on Convolutional Neural Networks

Rui Bian[1]
[1]China Celadon Institute, Lishui University, Lishui 323000, China; bianrui2025@126.com

Abstract

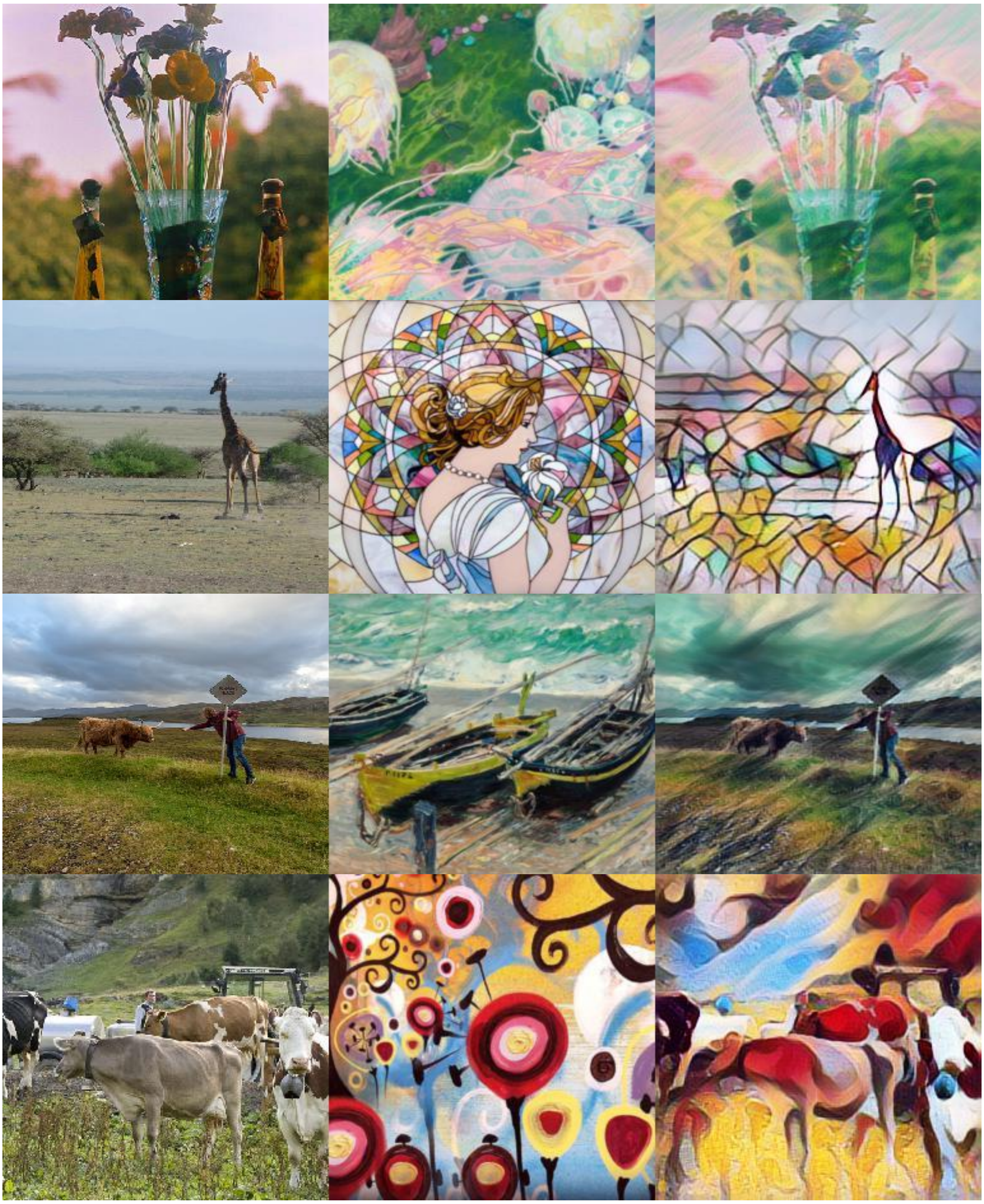
In computer-based painting research, style transfer and feature extraction is a challenging subject. Most research use manual segmentation to choose local regions, which results in inefficient final extracted characteristics and an inability to adequately identify the artist's aesthetic approach. To address this problem, this paper proposes a novel style loss and content loss guided two-channel VGG network for landscape painting style transfer and feature extraction. We convolve the input image and then use the feature maps from the third to the fifth layer of the two-channel network. A loss function for content and style is built up from the higher layers to the lower layers and decoded to the next layer by a decoder after each layer is matched until the final synthetic map is obtained, achieving feature extraction, feature constraint addition and parameter control for traditional Chinese paintings with local style transfer. It is observed that the feature information such as lines, textures and frequencies in landscape paintings are significantly different from other images compared to other styles of paintings, and therefore these feature information can be extracted and constrained and learned with style loss and content loss. The final experimental results show that the proposed method improves image style transfer and feature extraction for landscape paintings to some extent.

Keywords: Style Transfer; Feature Extraction; Landscape Painting; Convolutional Neural Networks; Image Processing

1 Introduction

Neural networks are a class of algorithms from the field of machine learning that were inspired by research into artificial neural networks [1-3]. The core idea of the algorithm is to extract higher-level feature information from the raw data input. For example, in intelligent computer image processing [4-5], lower-level features can recognise concepts such as edges of an image, while higher-level features can recognise human-related concepts such as numbers, letters or faces. It has been shown that the higher-level visual feature information extracted by neural networks is generalisable and generalisable, so that these extracted higher-level visual features can be applied to a variety of tasks. Currently, most of the neural network algorithms are derived from artificial neural networks (ANNs), in particular convolutional neural networks (CNNs) [6-7], and the network models are quite adaptable for image class high level feature extraction. Therefore, the use of a CNNs model maximises the extraction of higher-level semantic information in images.

Image style transfer is not an emerging image processing technology for practical purposes. Prior to the emergence of neural network-based style transfer techniques, traditional image style transfer techniques tended to treat style transfer as a generalised problem and task, i.e. by synthesising the texture features of an image. Specifically, i.e., texture features in a stylised image were obtained and their texture features were transferred to the content image. Efros et al. [8] gave a simple texture feature synthesis algorithm that could stitch and recombine the texture features of all sample images together; while, based on the idea of analogy, Hertzmann et al. [9] used the relationship of feature mapping between images to synthesise images with completely new texture features images. However, traditional image style transfer algorithms do not extract high-level feature information from the image, so when the style transfer is performed on images with complex textures, complex colours or messy structures, the resulting image is relatively coarse and difficult to apply in real-world scenes.



(a) Content (b) Style (c) Output

Figure 1: Schematic representation of neural network image style transfer.

Through a brief review and discussion of traditional image style transfer techniques, the authors observe that while traditional image style transfer techniques can reproduce the stylistic characteristics of stylised images well, they often have limitations in terms of flexibility, stylistic variety and the construction of the image. As a result, new needs have emerged to break down and remove these limitations, and in due course this has led to the emergence of neural network image style transfer techniques. Due to the rapid development of artificial neural networks, Gatys et al. [10] have made a significant contribution to the research on image style transfer and have used an innovative neural network-based approach to image style transfer on this basis. It is also demonstrated that the neural network is able to effectively extract the basic content information and style feature information of an image, and that by using this neural network to independently process the higher level feature information of an image, image style transfer can be carried out efficiently to produce admirable artistic results, as depicted in Figure 1.The basic idea is to use a pre-trained VGG network to iteratively update and optimise the synthetic image, with the aim of matching the desired high level abstract feature semantic distribution of the content and style images, and then iteratively update and optimise the input randomised noisy image through techniques such as gradient descent, and finally generate the original content image with a new style.

Artistic stylisation [11] has been an important research topic in computer vision for a long time because to its vast array of potential applications. Through neural network-based image style transfer techniques [12-14], we can extract stylistic information from an artistic image and assign its stylistic features to a specified real image, thus producing a stylised, unrealistic image. For example, combining an image of a Van Gogh painting of a starry sky with a natural image creates a stylised image of a Van Gogh starry sky. This technique allows other images that do not have such a style to have such a style, a technique known as style transfer. However, most of the current mainstream approaches to style transfer are limited to "one image, one style", which lacks variety and is limited by the model, which needs to be retrained after the target image has changed, making it less practical.

This research offers a novel style loss and content loss guided two-channel VGG network for landscape painting style transfer and feature extraction based on these observations. We convolve the input image before utilising the feature maps from the third to fifth layer of the two-channel network. From the top layer to the bottom layer, loss functions for content and style are constructed, and each layer is matched and decoded to the next layer until the final synthetic map is obtained, enabling feature extraction, feature constraint addition, and parameter control transfer of native styles for Chinese paintings. It was observed that feature information such as lines, textures and frequencies in landscape paintings are significantly different from other styles of painting compared to other images, so these feature information can be extracted, constrained and learned with loss of style and loss of content. The final experimental results show that the method improves image style transfer and feature extraction for landscape paintings to some extent.

The main contributions of this paper are as follows:

(1) A style transfer approach based on style loss and content loss constraints is suggested, which employs a two-channel VGG19 network to convolve the input image before employing the feature mappings from layer three to layer five. The content and style loss functions are constructed from higher to lower levels and decoded to the next layer by a decoder after each layer is matched until the final synthetic map is generated. Based on the original algorithm, feature extraction of landscape paintings, the addition of feature constraints and control of parameters, and local style transfer are implemented.

(2) Due to the special nature of landscape painting, the styles of paintings from different painters are generally different, but belonging to the same Chinese landscape painting means that there must be similarities, and each trained model is able to share weights, but multiple models are inefficient. The proposed method is therefore able to perform fast stylised transfer of multiple styles on a single model, highlighting its superiority.

2 Related Works

2.1 Image style transfer method based on image optimisation

Image style transfer aims to transform an image from its original style to another style while maintaining the integrity of the image's content. The challenge of style transfer stems from the realistic representation of images, and the earliest stage of style transfer is typically considered to be the transfer of image textures [15-16]. Image textures can be utilised as a depiction of the abstract concept of style and are strongly related to texture generation and transfer. These methods frequently rely on low-level statistical data, and the use of such low-level characteristics for style transmission is typically ineffectual and restricted. The optimal style limited transfer should contain the image's high-level semantics, which necessitates the capacity to identify the image's semantics. Presently, semantic transfer for image style transfer is becoming increasingly popular.

(1) Based on Gram matrix: Gaty et al [10] utilized the feature information matrix representation of the VGG network in the middle layer and found that the neural network can extract content feature information from any image and can extract style feature information from any image by constructing a Gram matrix. Based on this observation, they first used the constructed Gram matrix to obtain the style feature information in the image, followed by image reconstruction, and then optimised the reconstructed image using gradient descent algorithms (e.g. stochastic gradient descent algorithm, adaptive gradient algorithm, Adam's algorithm, etc.) so that the Gram matrix of its content image approximates the Gram matrix of its style image, and the neural network high level features in the content image with the feature information of the content image to finally obtain the stylised resultant image. Under the details of their algorithm, assuming that the target image is I , the content image is I_c and the style image is I_s ,the total loss function formula of the algorithm is expressed as follows：

$$L_{total}\left(I_c,I_s,I\right)=\alpha L_c(I,I_c)+\beta L_s(I,I_s) \tag{1}$$

where α is the equalization weighting factor of the image content loss function $L_c(I,I_c)$, and β is the equalization weighting factor of the image style loss function $L_s(I,I_s)$. While the image content loss L_c is defined by the squared Euclidean distance between the content feature representation F^l of the content image at layer l in VGG and the feature representation F^l of the stylized image I initialized with the noise image:

$$L_c=\sum_{l\in\{l_c\}}\left\|F^l\left(I_c\right)-F^l(I)\right\|^2 \tag{2}$$

where $\{l_c\}$ denotes the set of VGG model layers used to compute the content loss. For the style loss L_s , the style losses I_s and I are defined based on the squared Euclidean distance between the representations of the Gram matrix:

$$L_s=\sum_{l\in\{l_s\}}\left\|G\left(F^l\left(I_s\right)'\right)-G\left(F^l(I)'\right)\right\|^2 \tag{3}$$

where G represents the Gram matrix of the content image and style image, and $\{l_s\}$ represents the set of VGG model layers used to compute the style loss.

The use of a VGG-based neural network model for image style migration is not the only approach, and other pre-trained neural networks (such as ResNet [17]) can be used to achieve similar results. Furthermore, the Gram matrix-based style representation is not the only option. There are other style feature representations that are derived from Gram matrices. Li et al. have argued to some extent for a matching method of Gram matrices, equivalent to minimising a specific maximum mean difference [18], and go on to give different kernel functions, such as Gaussian kernel functions, first-order linear kernel functions, and higher-order kernel functions, for the purpose of improving image quality and reducing image style loss. In addition, the Gram matrix-based style migration algorithm has the disadvantage of requiring manual adjustment of parameters, and the instability of its model iterative optimisation process can cause image texture synthesis errors. In order to improve the above problems, Risser et al [19] added an additional histogram loss function to the algorithm, thus avoiding image texture synthesis errors due to the instability of iterative optimisation. They also proposed an initial solution for adaptive parameter tuning, i.e. preventing gradient extremes by extreme gradient normalisation. Based on content awareness, Yin proposed a synthesis that can effectively control the image content and style method [20], which led to a significant improvement in the resolution of the synthesised images.

(2) Based on Markov Random Field (MRF): MRF-based non-parametric image synthesis is a classic framework for traditional image style transfer [21]. Li and Wand [22] first proposed a neural network style transfer algorithm based on Markov random field. The core idea is to replace the loss function based on Gram matrix with the loss function based on Markov random field. The image style feature map is divided into several regions, and then matched to find and approximate the closest style region. Given the target image I , the content image I_c and the style image I_s , the total loss function based on Markov random field is expressed as follows:

$$L_s = \sum_{l \in \{l_s\}} \sum_{i=1}^{m} \left\| \psi_i \left(F^l (I) \right) - \psi_{NN(i)} (F^l (I_s)) \right\|^2 \tag{4}$$

where $\psi\left(F^l(I)\right)$ is the set of all regional patches in the feature map $F^l(I)$. ψ_i represents the i -th region patch of the target image, and $\psi_{NN(i)}$ represents the most similar style patch in the i -th region patch of the style image. The best matching $\psi_{NN(i)}$ is obtained by calculating the normalized cross-correlation overall style Patch in the style image I_s . m is the total number of local patches. Since their algorithm maps style feature information at the patch level, it can better maintain the structure and arrangement of images.

Due to the Markov random field based loss, it image synthesis works very well for photorealistic styles, or more specifically, when the shape and perspective of the content image and the style image are identical. However, when there are significant changes in perspective and structure, the regions in the image Patch do not match well, affecting the image's preservation of depth information and fine detail.

2.2 Image style transfer method based on model optimization

Although image style migration techniques based on image optimisation can achieve excellent stylisation results, it also has certain drawbacks. The biggest problem in image optimisation algorithms exists in the form of low computing speed. Currently, image style migration based on model optimisation is the most commonly used style migration technique. Compared to other image optimisation-based image style migration algorithms, model optimisation has a higher computational speed, significantly reduces computational costs and produces better style migration results. Model optimisation is a data-driven style migration algorithm, which often requires large amounts of data to feed the model for training. A variety of model optimisation-based image style migration algorithms have been proposed by domestic and international scholars, which can be fused with other types of algorithms, which has led to an increase in the efficiency of image style migration every year. The model-optimisation-based style migration approach largely solves the problem of computational speed and cost by using a network-generated model that has been trained to produce stylised result maps, i.e. optimising the forward neural network g for one or more style image I_s by using a large number of image I_c .

$$\begin{cases} \theta^* = \arg\min_{\theta} L_{Total}\left(I_c, I, g_\theta * (I_c)\right) \\ \qquad I^* = g_\theta * (I_c) \end{cases} \tag{5}$$

Based on the number of image styles that can be produced by a single forward neural network g, the model-based optimisation style migration algorithms can then be subdivided into: single model single style style migration algorithms, single model multiple style style migration algorithms, and single model arbitrary style migration algorithms.

(1)Single-model single-style style transfer algorithm: The first two algorithms for image style migration based on model optimisation were proposed by Johnson and Ulyanov respectively. Both approaches share the same idea, that is, a forward neural network is pre-trained and a stylised resultant image is generated by that model. They differ only in their neural network architectures, with Johnson's architecture design being based on Radford et al.'s residual neural network [23], while Ulyanov uses a multi-scale model architecture. Both loss functions are similar to the algorithm of Gatys et al. and both use Gram matrices for stylised modelling. Since then, Ulyanov et al. [24] have discovered that normalising a single image, as opposed to a group of photographs, improves the quality of the stylized image more. This method of normalising a single image is known as Instance Normalisation (IN), which corresponds to Batch Normalisation (BN) when the batch size is set to 1. IN also has a faster network convergence rate for style migration and better visual performance compared to BN. IN is a style normalisation format that allows for style normalisation of each image [25].

(2)Single-model multi-style style transfer algorithm: The single model single style model mentioned above requires separate neural network training for different styles of images, which is very time consuming and inflexible. Many paintings (e.g. Impressionist paintings) have a similar approach, except that the paint varies. Objectively speaking, it is not necessary to utilise a separate neural network to train for each style of painting. Therefore, a single-model multi-styles approach has been introduced to blend multiple styles in a deeper way, thus making single-model single-styles more flexible. There are two general ways of dealing with this problem: simply binding only a small number of parameters in the network to each style; or still utilising a single network like single model single style, but combining styles and content as input. Dumoulin et al [26] showed that in convolutional neural networks, various styles of images can be well simulated using the same convolutional parameters, simply by affine transformations of the parameters in the IN layer. To this end, they give a multi-style migration model based on the theory of conditional instance normalisation (CIN) of a multi-style migration model, defined as follows:

$$CIN\left(F\left(I_c\right), S\right) = \gamma^s \left(\frac{F\left(I_c\right) - \mu\left(F\left(I_c\right)\right)}{\sigma\left(F\left(I_c\right)\right)} \right) + \beta^s \tag{6}$$

where F is the input feature representation and s is the index of the desired style in a set of style images. As shown in the above equation, after normalising the feature representation $F\left(I_c\right)$, the scaling and shifting of the parameters γ and β is done to each styles conditionally, i.e. each style is attainable by affine transformation of the parameters. In addition, the approach of Dumoulin et al. can be expanded to merge many styles into a single style outcome by merging the affine parameters of the various styles.

Another multi-style algorithm, invented by Chen et al [27], has the idea of explicitly separating style and content, and learning the corresponding content and style features by using separate network modules. Specifically, they used an intermediate layer convolutional filter called "StyleBank" to learn different image styles. The style information is then integrated with the StyleBank layer, which was tied to a set of layer parameters. In addition, the neural net network can learn image content information in other network modules, and this content information can be shared between individual styles. In addition, the other modules of the neural network are capable of learning visual content data, which can be shared between styles. Their system is adaptable and supports progressive training. In the picture content module of a fixed network model, just the " StyleBank " layer needs to be trained in order to learn new styles.

3 Methodology

This research finds that rescaling the style loss can link the majority of parametric and non-parametric style migration approaches in order to improve style migration and feature extraction for landscape painting. The benefits of both approaches are blended to generate superior results. This chapter extends the original method by adding landscape-specific stylistic characteristics and limitations utilising content loss and style loss. Observation reveals that elements such as lines and textures in landscape paintings differ significantly from those in other photos; hence, these traits are retrieved and applied as constraints to the original approach. The final experiment proves that the proposed has some improvement for style migration and feature extraction in landscape painting.

3.1 Landscape painting feature extraction

(1) Line feature extraction

The most fundamental element of Chinese painting is the line. The use of line by painters to extract, summarise, and abstract natural objects and settings. The two-dimensional plane is used to express the three-dimensional space. Line is also the basic modelling tool in traditional Chinese landscape painting, with colour being a secondary feature, and without reference to the light and shade of the malefactor. Whether landscape, birds or flowers, painters always use lines of varying thickness to outline and then complement them with colour, and the line is also one of the forms of expression of the emotional character of traditional Chinese landscape painting. A diagram of lines in a session as shown in Figure 2. Therefore, this paper uses curvature to express the fluidity of the lines, and the calculation equation is expressed as follows:

$$F_{line} = \frac{\left(1+f_x^2\right)f_{yy} + \left(1+f_y^2\right)f_{xx} - 2f_x f_y f_{xy}}{\left(1+f_x^2+f_y^2\right)^{3/2}} \tag{7}$$

where x and y denote the coordinates of a pixel in the image and $f(x,y)$ is the grey value of the pixel. f_x, f_y, f_{xy}, f_{xx}, f_{yy} are the first, second and mixed partial derivatives of $f(x,y)$ respectively and f_{line} is the Gaussian curvature of the pixel.

Figure 2: Schematic diagram of lines in landscape painting.

(2) Texture Feature Extraction

Texture generally refers to the pattern on the surface of an object and is used in everyday life to show the smoothness of an object. It reflects the cyclical nature of colour changes on the surface of an object. For artistic drawing, texture is present in the contours and is a representation of the line drawing as well as the background. For computers, texture is a visual feature that consists of a myriad of pixels that follow certain rules. Texture features can be divided into four types: (1) statistical texture features (2) model texture features (3) signal processing texture features (4) structural texture features. There are many methods for extracting texture, including statistical analysis-based methods, image-structure-based methods and model-based methods. This paper uses the LBP method to extract texture:

$$LBP(x_c, y_c) = \sum_{p=0}^{p-1} 2^p S(i_p - i_c) \tag{8}$$

where (x_c, y_c) represents the central element of the 3x3 field, his pixel value is i_p and i_c represents the other pixel values in the field. $S(x)$ is the symbolic function and is defined as follows:

$$S(x) = \begin{cases} 1 & x \geq 0 \\ 0 & else \end{cases} \tag{9}$$

(3) Frequency feature extraction

The frequency of an image is a spatial frequency that indicates how the greyscale in the image fluctuates in planar space. Different frequency information on the Fourier spectrum play distinct functions in the structure of an image. Low frequency information forms the basic grey level of the image and plays a lesser role in determining the image structure; medium frequency information determines the basic structure of the image and forms the main edge structure of the image; high frequency information forms the edges and details of the image and is an enhancement of the image content based on medium frequency information. The frequency map is extracted using the Fourier transform in this article.

3.2 Image style transfer and feature extraction based on dual-channel VGG

In convolutional neural network-based style migration, there are two types of images - style images and content images. The core principle is to copy the style from the style image and apply it to the content image. The basic working principle of a convolutional neural network is to use convolutional kernels to perform convolutional operations. For example, there is an image and a convolution kernel. We slide the convolutional kernel onto the image and take the weighted sum of the inputs covered by the convolutional kernel as the output via a non-linear transformation function. Each convolutional kernel has its own set of weight values which do not change during the convolutional operation. In this way, the input image can be convolved with multiple convolutional kernels in a convolutional neural network to produce convolutional kernel mappings. These convolutional kernel mappings are then convolved with more convolutional kernels to generate more feature mappings. Convolution at lower levels corresponds to lower level features, such as lines or speckles. As you move to higher levels of convolution, the features become increasingly complex. Convolutional neural networks can extract image features, which are the basis for style migration. When performing an image style shift, one is not training a neural network. Instead, the loss function needs to be optimised by changing the pixel values of a randomly noisy image. Simply put, when training a neural network, the weights and biases of the network model are updated, but in a style shift, the weights and biases need to be kept constant.

(1) Content loss and style loss

The loss function is important in order to keep the weights and biases constant. Given a target image x, a content image c and a style image s, the equation of the image content loss function for style migration based on a two-channel VGG convolutional neural network is expressed as follows:

$$L^{l}_{content}(c,x)=\sum_{i,j}\left(F^{l}_{ij}(x)-F^{l}_{ij}(c)\right)^{2} \tag{10}$$

The image content loss is a function that defines the squared distance between the input image x and the content image c. Let cnn be a trained deep convolutional neural network. Suppose x is an arbitrary image, then $cnn(x)$ is the neural network provided to image x. Let $F^{l}_{ij}(x)\in cnn(x)$ and $F^{l}_{ij}(c)\in cnn(x)$ describe respectively the input image x and the content image c in l intermediate feature representation of the layer network.

Similarly, the equation for computing the image style loss function can be expressed as follows:

$$L^{l}_{style}(s,x)=\sum_{ij}\left(G\left(F^{l}_{ij}(x)'\right)-G\left(F^{l}_{ij}(c)'\right)\right)^{2} \tag{11}$$

where G denotes the Gran matrix of the content image and the target image. The equation for calculating the total loss function of style migration for the two-channel VGG convolutional neural network is as follows:

$$L^{l}_{total}(c,s,x)=aL^{l}_{content}(x,c)+bL^{l}_{style}(x,s) \tag{12}$$

where a is the balanced weight coefficient of the image content loss function, and b is the equilibrium weight coefficient of the image style loss function weight coefficients.

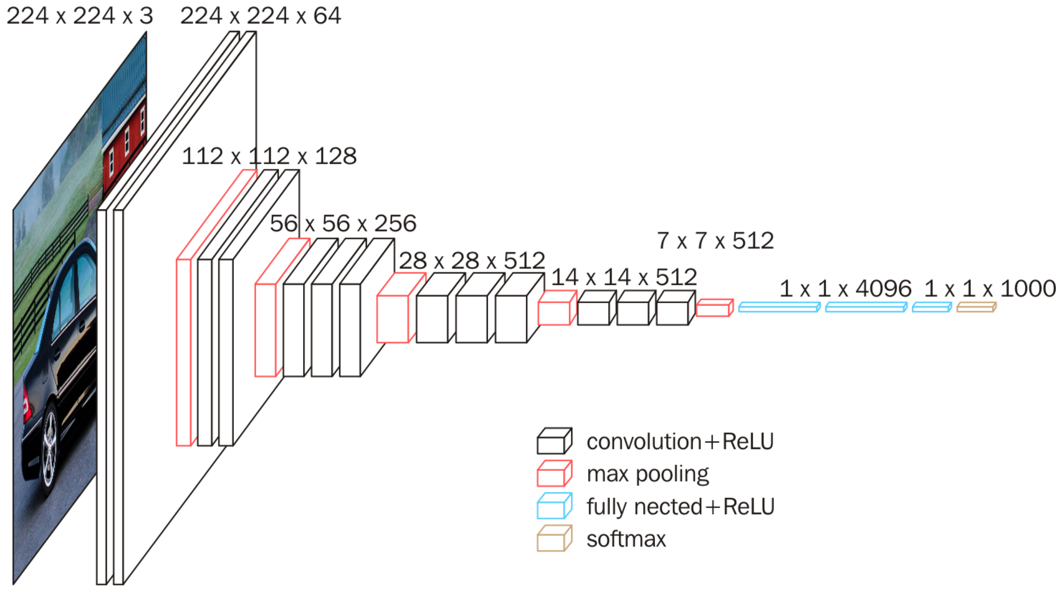


Figure 3: Schematic diagram of vgg16 network structure.

(2) VGG-16 network for feature extraction

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

Figure 4: Schematic diagram of vgg16 network structure.

Before elaborating on the VGG-16 network, knowledge of convolutional neural networks is first required. A colour image is usually divided into three channels of RGB, representing the colours red, green and blue respectively. For a black and white image, each pixel location is assigned a pixel value between 0 and 255 to represent the intensity of that pixel, but for a colour image, each pixel location is given three pixel values to represent the red, green and blue luminance of that pixel, all of which are also between 0 and 255. The colours in a colour image can be represented by different combinations of the three RGB channel values or by blending them in different proportions. The convolution process is based on a small matrix, also called a convolution kernel, which is continuously moved through each layer of the pixel matrix in steps, multiplying the number of passes by the number in the corresponding position of the convolution kernel, and then summing them to create a new matrix. Starting with a random initial value, each value in the convolution kernel represents a parameter of the neuron being trained. When training the network, these parameter values will be continuously updated via backward propagation until the optimal parameter value is determined. After the convolution operation there will be fewer dimensions, which means that the resulting matrix will be smaller than the original matrix.

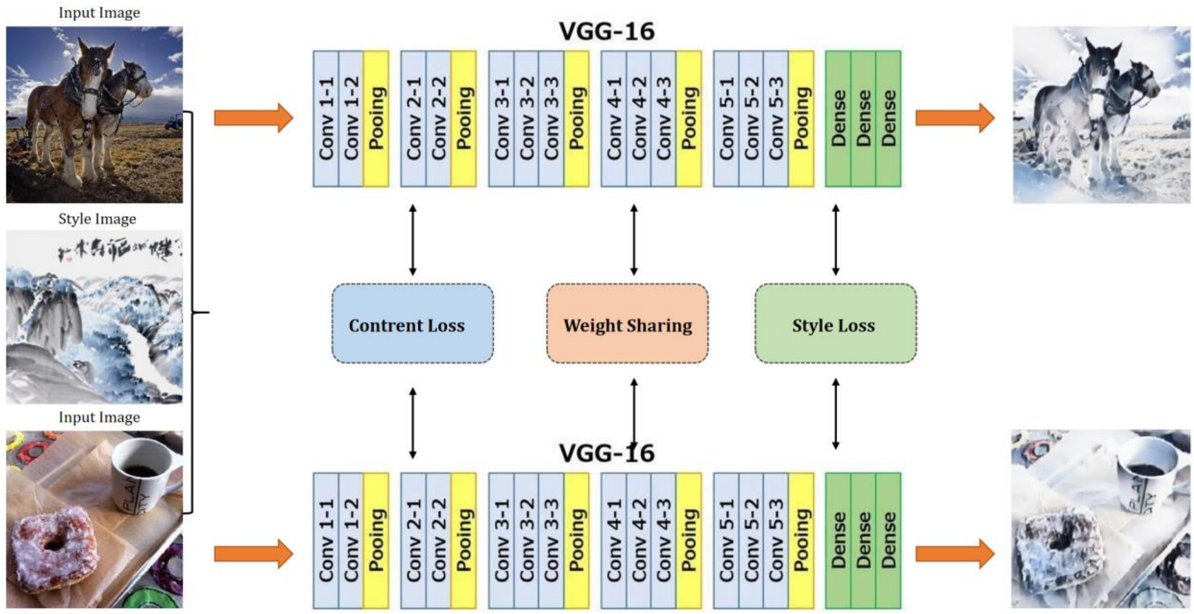


Figure 5: Schematic diagram of image style transfer and feature extraction based on dual-channel VGG16.

AlexNet before the VGG-16 network (as shown in Figure 3 and Figure 4) has made great breakthroughs in neural networks, using CUDA to accelerate the training speed of convolutional neural networks, and using ReLU in the selection of activation functions, while The VGG-16 network uses the principle that several small convolution kernels can replace a large convolution kernel, and calculates through continuous 3×3 convolution kernels, replacing the 11×11 and 7×7 in its previous work AlexNet. convolution kernel. For a given receptive field (the local size of the input image relative to the output), stacking small kernels outperforms larger ones. According to the calculation of the receptive field below, using three 3x3 convolution kernels can be equivalent to a 7x7 convolution kernel, and two 3x3 convolution kernels can be equivalent to a 5×5 convolution kernel. A point at a certain position on the feature map has the same receptive field on the input image, but it deepens the depth of the network compared to the previous network, thereby improving performance.

In fact the use of the VGG16 network in the style migration task does not ultimately output a 1000-dimensional vector for predicting object categories, but rather the pre-trained VGG16 network to extract the features of the image to build the loss function in style migration. The following explains how the loss is constructed using the image features extracted by the VGG16 network: as shown in Figure 5, the objective of style migration can be temporarily divided into two, firstly the generated image needs to contain all the objects in the input content map, and secondly the overall style of the generated image and the provided style map needs to match approximately, so the loss can be divided into two parts, i.e. content loss and style loss. But obviously content loss and style loss should not be calculated in the same way. We already know that for a pre-trained VGG16 network, what is obtained at the end of each convolutional layer is a picture feature in the form of a feature map, while the fine content of a picture can be regarded as a relatively low level feature and the texture of a picture as a relatively high level feature. By observing the feature maps at the end of each convolutional layer in the VGG16 network, it can be found that: the feature maps obtained in the first few convolutional layer stages retain the local features of the pictures better, while in the later stages of the network the local features of the pictures are already very insignificant, but retain the stylistic features of the pictures relatively well, so it can be presumed that a reasonable loss function should be calculated by the first few layers of the VGG16 feature map to calculate the difference in content, while using the feature map at the end of all the convolutional layers of VGG16 to calculate the difference in style, and that the two calculations should be different.

4 Experiment and Results

4.1 Datasets

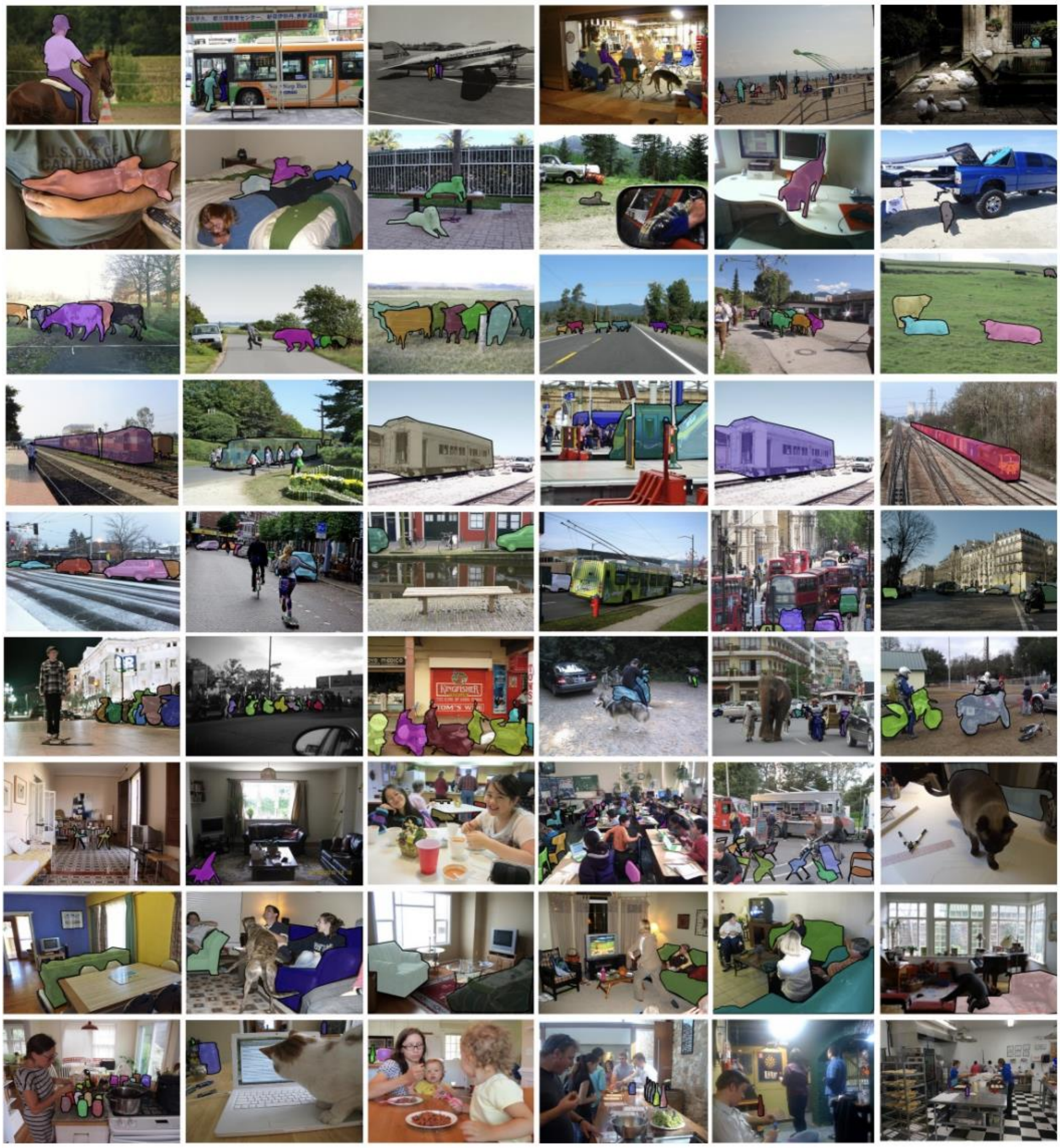


Figure 6: Samples of images in the MS COCO 2017 dataset.

The training set data in this article contains 10,000 real scene images, 5,000 of which are selected from MS COCO2017 (as shown in Figure 7) val images, and the other 5,000 are selected from 2017 Test images. For the selection of style images, this article mostly adopts the previous style transfer work. Choose more style pictures. In the specific training, first resize the pictures in the training set to the size of 256*256, and then take the batch size as 10 to train 5 epochs, use the Adam optimization with the learning rate set to 10^{3}, and train with strength When controlling the model, set the proportion of style loss in the loss function to 1e-3, so that the final model according to the different values of α can cover the strength between 1e-3-1e-5.

4.2 Experimental Setup

In this experiment, a convolutional neural network built using the TensorFlow framework was utilised. TensorFlow is an open source software library for machine learning and numerical computing. TensorFlow employs nodes to represent mathematical calculations, and graph edges to represent multidimensional data arrays in a data graph or tensors transferred between them. Its architecture is adaptable, enabling users to extend computing to one or more CPUs or GPUs on desktops, servers, or mobile devices via an API. The computer operating system is Windows 10 Professional Edition, and the experimental hardware equipment: CPU of Intel(R) Core(TM) i7-6700HQ, GPU of NVIDIA GeForce GTX 1080, the memory size is 8G, and the graphics library CUDA 9.0 launched by NVIDIA is used for the experiment. GPU accelerated computing.

4.3 Experimental results and analysis

(a) Content image (b) Style image (c) Generated image

Figure 7: Landscape image and Chinese print composite effect image.

This experiment uses an image of two boats on the coast as the content image and an image with similar content image in the style of Chinese prints as the style image. After several experiments, the authors' results show that adding content loss and style loss to the VGG-16 neural network has a better improvement on the performance of VGG-16. Figure 7 depicts the original content image, style image, and style migration image. Due to the method's optimisation enhancement, image noise artefacts are drastically minimised and the image seems softer. The image noise artefacts have been drastically reduced, rendering the image softer.

(b) Content image (b) Style image (c) Generated image

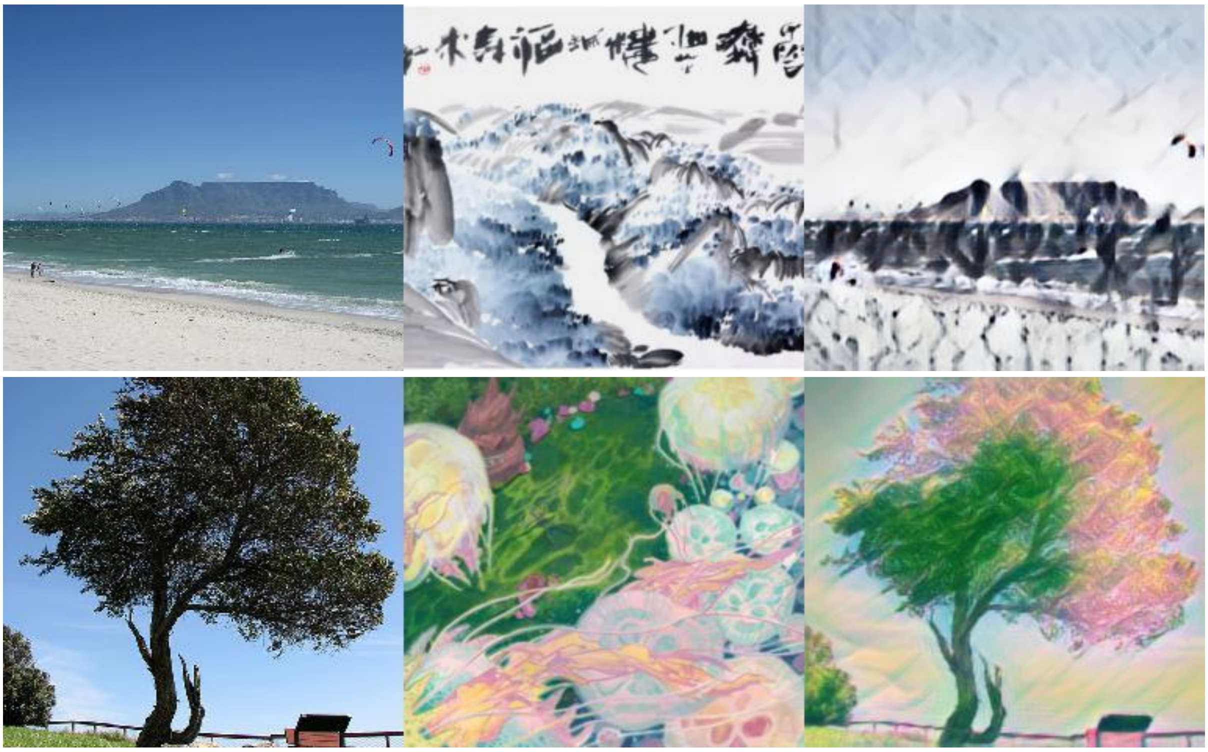
Figure 8: Landscape image and Chinese print composite effect image.

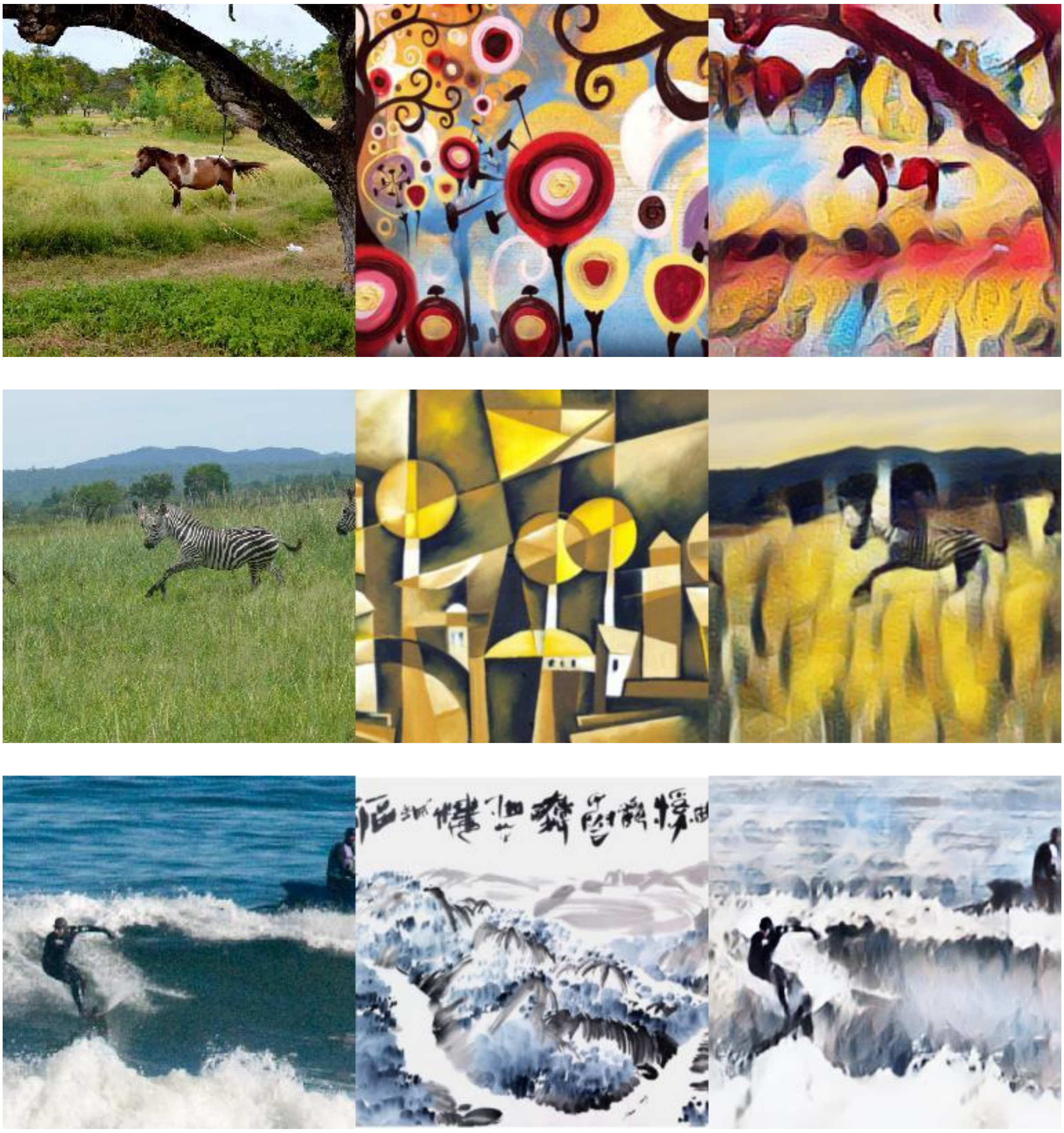
The experimental image data input: a natural photo, a group of Chinese prints. Through experiments and theoretical analysis, it is still the best choice to operate the fourth layer of VGG-16 neural network. The experiment proves that if the difference between the main body and the background of the content image is not clear, the method in this paper still has a certain improvement ability. As shown in Figure 8, it is the content image of this experiment. In Fig. 8, a composite image of our method is shown. The synthesized image obtained by the method in this paper is more natural on the whole, and its color, contrast, saturation, etc. are better than the original image, and it has achieved good results in the details of the image.

Nowadays, there is no accepted standard evaluation system for the evaluation of computer image style migration algorithms. Since most of them are subjective evaluations, in this experiment the authors found 10 art practitioners and asked them to evaluate the composite effect image shown in Figure 9. These 10 art practitioners were people who had studied painting styles and were aware of the differences between the landscape painting style and other painting styles. When the subjects did not know the provenance of the images, they were asked to evaluate the synthetic images of the original method and the synthetic images of the present method. The results of the evaluation are shown in Table 1, it can be found that the overall rating of this method is high, so this method can have an optimal effect on the migration of styles in landscape painting to some extent.

Table 1: Composite image ratings from 10 art practitioners.

Type	Landscape	Personage	Food	Architecture	Animal
Traditional machine learning	0.4	0.2	0.5	0.6	0.5
Manual method	0.5	0.5	0.6	0.6	0.4
Ours	**0.9**	**0.6**	**0.7**	**0.8**	**0.9**





(c) Content image (b) Style image (c) Generated image

Figure 9: Landscape image and Chinese print composite effect image.

5 Conclusion

In this paper, we propose a novel style loss and content loss guided two-channel VGG network for landscape painting style transfer and feature extraction. We convolve the input image and then use the feature maps from the third to fifth layers of the two-channel network. Construct the loss function of content and style from the upper layer to the lower layer. After each layer is matched, it will be decoded by the decoder to the next layer until the final composite image is obtained. Feature extraction, feature constraint addition, and parameter control are used to migrate the local style of Chinese painting. . Observing that feature information such as line, texture, and frequency in landscape paintings are significantly different from other styles of painting, these feature information can be extracted and constrained and study. The final experimental results show that this method improves the image style transfer and feature extraction of landscape paintings to a certain extent.

Conflicts of Interest

The authors do not have any possible conflicts of interest.

Funding Statement

This work was not supported by any foundation.

References

[1] Jin, X. B., Wang, Z. Y., Kong, J. L., Bai, Y. T., Su, T. L., Ma, H. J., & Chakrabarti, P. (2023). Deep Spatio-Temporal Graph Network with Self-Optimization for Air Quality Prediction. Entropy, 25(2), 247.
[2] Chen J, Li T, Zhang Y, et al. Global-and-local attention-based reinforcement learning for cooperative behaviour control of multiple UAVs[J]. IEEE Transactions on Vehicular Technology, 2023.
[3] Huang Z, Zhang P, Liu R, et al. An Improved YOLOv3-Based Method for Immature Apple Detection[J]. IECE Transactions on Internet of Things, 2023, 1(1): 9-14.
[4] W. Cai et al., "A Novel Hyperspectral Image Classification Model Using Bole Convolution With Three-Direction Attention Mechanism: Small Sample and Unbalanced Learning," in IEEE Transactions on Geoscience and Remote Sensing, vol. 61, pp. 1-17, 2023, Art no. 5500917, doi: 10.1109/TGRS.2022.3201056.
[5] Ding, Y., Zhang, Z., Zhao, X., Cai, W., Yang, N., Hu, H., ... & Cai, W. (2022). Unsupervised self-correlated learning smoothy enhanced locality preserving graph convolution embedding clustering for hyperspectral images. IEEE Transactions on Geoscience and Remote Sensing, 60, 1-16.
[6] Z. Zhang, Y. Ding, X. Zhao, S. Li, N. Yang, Y. Cai and Y. Zhan, " MultiReceptive Field: An Adaptive Path Aggregation Graph Neural Framework for Hyperspectral Image Classification," Expert Systems with Applications, early accepted,2023.
[7] Jianming Zhang, Wei Wang, Chaoquan Lu, Jin Wang, Arun Kumar Sangaiah. Lightweight deep network for traffic sign classification. Annals of Telecommunications, 2020, vol. 75, no. 7-8, pp. 369-379. DOI: 10.1007/s12243-019-00731-9.
[8] Efros, A. A., & Freeman, W. T. (2001, August). Image quilting for texture synthesis and transfer. In Proceedings of the 28th annual conference on Computer graphics and interactive techniques (pp. 341-346).
[9] Hertzmann, A., Jacobs, C. E., Oliver, N., Curless, B., & Salesin, D. H. (2001, August). Image analogies. In Proceedings of the 28th annual conference on Computer graphics and interactive techniques (pp. 327-340).
[10] Gatys, L. A., Ecker, A. S., & Bethge, M. (2015). A neural algorithm of artistic style. arXiv preprint arXiv:1508.06576.
[11] Zhou, J., Li, L., & Yu, Z. (2021). The transfer of stylised artistic images in eye movement experiments based on fuzzy differential equations. Applied Mathematics and Nonlinear Sciences, 7(2), 477-484.
[12] Kwon, G., & Ye, J. C. (2022). Clipstyler: Image style transfer with a single text condition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 18062-18071).
[13] Yang, S., Jiang, L., Liu, Z., & Loy, C. C. (2022). Pastiche master: exemplar-based high-resolution portrait style transfer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 7693-7702).
[14] Zhang, Y., Tang, F., Dong, W., Huang, H., Ma, C., Lee, T. Y., & Xu, C. (2022, July). Domain enhanced arbitrary image style transfer via contrastive learning. In ACM SIGGRAPH 2022 Conference Proceedings (pp. 1-8).
[15] Aminuddin, A., & Ernawan, F. (2022). AuSR2: Image watermarking technique for authentication and self-recovery with image texture preservation. Computers and Electrical Engineering, 102, 108207.

[16] Cai, W., Wei, Z., Song, Y., Li, M., & Yang, X. (2021). Residual-capsule networks with threshold convolution for segmentation of wheat plantation rows in UAV images. Multimedia Tools and Applications, 80, 32131-32147.

[17] Wu, Z., Shen, C., & Van Den Hengel, A. (2019). Wider or deeper: Revisiting the resnet model for visual recognition. Pattern Recognition, 90, 119-133.

[18] Li, Y., Wang, N., Liu, J., & Hou, X. (2017). Demystifying neural style transfer. arXiv preprint arXiv:1701.01036.

[19] Risser, E., Wilmot, P., & Barnes, C. (2017). Stable and controllable neural texture synthesis and style transfer using histogram losses. arXiv preprint arXiv:1701.08893.

[20] Yin, R. (2016). Content aware neural style transfer. arXiv preprint arXiv:1601.04568.

[21] Efros, A. A., & Leung, T. K. (1999, September). Texture synthesis by non-parametric sampling. In Proceedings of the seventh IEEE international conference on computer vision (Vol. 2, pp. 1033-1038). IEEE.

[22] Li, C., & Wand, M. (2016). Combining markov random fields and convolutional neural networks for image synthesis. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2479-2486).

[23] Radford, A., Metz, L., & Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434.

[24] Ulyanov, D., Vedaldi, A., & Lempitsky, V. (2017). Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 6924-6932).

[25] Huang, X., & Belongie, S. (2017). Arbitrary style transfer in real-time with adaptive instance normalization. In Proceedings of the IEEE international conference on computer vision (pp. 1501-1510).

[26] Dumoulin, V., Perez, E., Schucher, N., Strub, F., Vries, H. D., Courville, A., & Bengio, Y. (2018). Feature-wise transformations. Distill, 3(7), e11.

[27] Chen, D., Yuan, L., Liao, J., Yu, N., & Hua, G. (2017). Stylebank: An explicit representation for neural image style transfer. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1897-1906).