A fast multi-scale textile pattern generation method combining layered loss and convolutional attention

DOI: 10.35530/IT.076.03.2024126

GUODONG XU
YU CHEN
JIAN HUA

ABSTRACT – REZUMAT

A fast multi-scale textile pattern generation method combining layered loss and convolutional attention

This paper introduces a neural network-based algorithm for the rapid multi-scale synthesis of textile patterns. The algorithm achieves comprehensive textile pattern style reconstruction by utilising low-level feature loss, represented by the Gram matrix, to capture colour and texture, and high-level feature loss, represented by the Wasserstein distance, to capture complex structures and semantic content. The convolutional attention feature enhancement module is incorporated to improve pattern detail clarity and overall visual quality by emphasising significant features and suppressing irrelevant information. Furthermore, the multi-scale optimisation module enhances texture and layering by optimising the image at different scales. Experimental results demonstrate that, compared to existing methods, this approach offers superior visual effects in pattern synthesis and scalability in pattern size. It not only generates high-quality textile patterns but also excels in managing complex textures and maintaining semantic consistency. This method aids designers in extending pattern designs and advances the intelligent development of textile design and production.

Keywords: *textile pattern design, neural network, attention mechanism, style loss, intelligent design*

O metodă rapidă de generare multi-scală a modelelor textile care combină pierderea stratificată și atenția convoluțională

Această lucrare prezintă un algoritm bazat pe o rețea neurală pentru sinteza rapidă multi-scală a modelelor textile. Algoritmul realizează o reconstrucție cuprinzătoare a stilului modelelor textile prin utilizarea pierderii de caracteristici de nivel scăzut, reprezentată de matricea Gram, pentru a capta culoarea și textura, și pierderea de caracteristici de nivel înalt, reprezentată de distanța Wasserstein, pentru a capta structuri complexe și conținut semantic. Modulul de îmbunătățire a caracteristicilor atenției convoluționale este încorporat pentru a îmbunătăți claritatea detaliilor modelului și calitatea vizuală generală prin accentuarea caracteristicilor semnificative și suprimarea informațiilor irelevante. În plus, modulul de optimizare multi-scală îmbunătățește textura și stratificarea prin optimizarea imaginii la diferite scale. Rezultatele experimentale demonstrează că, în comparație cu metodele existente, această abordare oferă efecte vizuale superioare în sinteza modelelor și scalabilitate în dimensiunea modelelor. Aceasta nu numai că generează modele textile de înaltă calitate, dar excelează și în gestionarea texturilor complexe și menținerea coerenței semantice. Această metodă îi ajută pe designeri să extindă modelele și promovează dezvoltarea inteligentă a designului și producției textile.

Cuvinte-cheie*: model textil, rețea neuronală, mecanism de atenție, pierdere de stil, design inteligent*

INTRODUCTION

Pattern design is a crucial aspect of textile design [1]. Traditionally, designers extensively explore sources of inspiration [2], extracting core elements such as structure, perspective, lines, texture, and colour. These elements are then used to upgrade and recreate the artistic essence of the inspiration materials through hand-drawing or software [3], ultimately completing the pattern design. This process tests the designer's artistic imagination, perspective selection, information collection, and abstract expression skills, and is extremely time-consuming and labour-intensive. In recent years, the adoption of new digital technologies has enabled the automatic recombination of core elements from inspiration materials, rapidly generating a large number of pattern drafts for designers to choose from [4]. This approach not only significantly improves design efficiency but also enhances the designer's creativity and imagination. This trend is set to become a new standard in the field of textile pattern design.

In the field of computer image design, the rapid development of deep learning technology based on convolutional neural networks has driven research in textile image generation [2–6]. Li et al. [5] achieved camouflage pattern generation using the CycleGAN algorithm. Wu et al. [6] employed generative adversarial networks (GANs) to create fashionable Dunhuang-style clothing. Liu et al. [7] utilised conditional GANs for innovative designs of traditional Chinese textile patterns. However, GANs face several challenges, including training instability, the need for large datasets, long training times, high computational power requirements, and difficulty in generating high-resolution images [8]. Karagao et al. [9] employed diffusion models for textile pattern generation, while Xie et al. [10] proposed a multi-stage diffusion model that integrates high-level design concepts with low-level clothing attributes for generating and editing fashion design drafts. Although diffusion models produce high-quality images, they demand even greater GPU computational power, limiting their practical application due to high computational costs.

Style transfer, as a significant branch of image generation technology, enables the application of one image's style to another, imbuing it with a new artistic flair to create a novel image. Compared to other image generation technologies, style transfer offers greater flexibility and fewer usage restrictions, making it highly applicable in computer art and visual design. This capability opens up innovative possibilities for textile pattern design. Some researchers have already applied it to textile pattern generation. For instance, Sun et al. [11] developed a fast textile pattern generation algorithm combining Markov Random Fields (MRF) and Gram methods, while Qiu et al. [12] proposed a colour-optimised local pattern style transfer method for fabrics.

Style transfer originates from non-photorealistic rendering (NPR) and is closely related to texture synthesis and texture transfer. Gatys et al. [13] pioneered neural style transfer by extracting image features using convolutional neural networks and constructing Gram matrices to capture image styles, achieving oil painting style transfer through high-level feature processing. This method remains a gold standard today. Subsequently, Johnson et al. [14] introduced a fast style transfer algorithm, enabling rapid image style transfer through iterative optimisation of the generation model. Li et al. [15] proposed a neural style transfer method based on patch matching using MRF. However, due to the high computational cost of numerous patch matches, the method has long running times and struggles with images exhibiting large-scale structural differences. The introduction of the Wasserstein distance improved the measurement of image style distribution differences, enhancing style transfer quality [16–20]. However, its high computational complexity limits its direct application in neural style transfer [16].

Despite these advancements, applying neural style transfer to textile pattern generation still faces limitations. Existing methods are predominantly used for abstract creations like artistic paintings, focusing on the transfer and reconstruction of global style features and local texture features while neglecting the integrity of semantic structure features in patterns [20]. Additionally, during the generation process, local structural information is often incomplete, leading to artefacts in areas of repeated textures, thereby affecting pattern quality. Textile pattern design, unlike abstract art, requires the preservation of the complete regularity of local pattern structures. Capturing and transferring complex style features while maintaining the integrity of local structures and enhancing visual naturalness and generation efficiency remains a challenge.

To address these issues, this paper proposes a multiscale style transfer algorithm for textile pattern generation, incorporating hierarchical style loss optimisation and convolutional attention feature extraction. This approach maintains the integrity of local semantic structures in style images during pattern generation, resulting in improved visual effects. It is well-suited for textile pattern generation and demonstrates higher generation efficiency compared to similar methods.

TEXTILE PATTERN GENERATION ALGORITHM

The textile pattern generation model proposed in this paper consists of four modules: the initialisation module, the feature extraction enhancement module, the loss calculation module, and the multiscale optimisation module, as shown in figure 1. First, the target pattern is input, and style initialisation is performed. Depending on the input image size, a multiscale division is conducted to generate an optimisation pyramid sequence for the pattern. The feature network extracts the pattern's style features, and the channel spatial attention module enhances and integrates these features at the style layer. The loss calculation module computes the feature loss at different levels, and combined with the multiscale optimisation module, iterative optimisation of the pattern is performed at different scales, ultimately generating a new style pattern.

Style loss function

In this paper, the VGG19 network is used as the style feature extractor. It retains 16 convolutional layers and removes the original fully connected layers to reduce computational overhead. All pooling layers are replaced with average pooling to better preserve global information and reduce overfitting. The network layers {conv1_1, conv2_1, conv3_1, conv4_1, conv5_1} are selected as style layers, and the style feature extraction layers are designated as Layer1-5. The shallow layers (close to the input layer) of the network mainly capture primary features such as edges, textures, and colours, while the deep layers capture more complex structures, content objects, and style patterns [25].

The style loss function is a core component of image style feature reconstruction [25], crucial for the overall style effect of the generated image. Previous studies [13–16] typically used a single style loss function for image style reconstruction, which can quickly complete the reconstruction of the pattern style but often fails to adequately represent both low-level and high-level features. This paper employs a hierarchical approach, combining different style losses to simultaneously express low-level and high-level features.

To better reconstruct style features at different levels, this paper uses Gram matrices for the shallow low-level

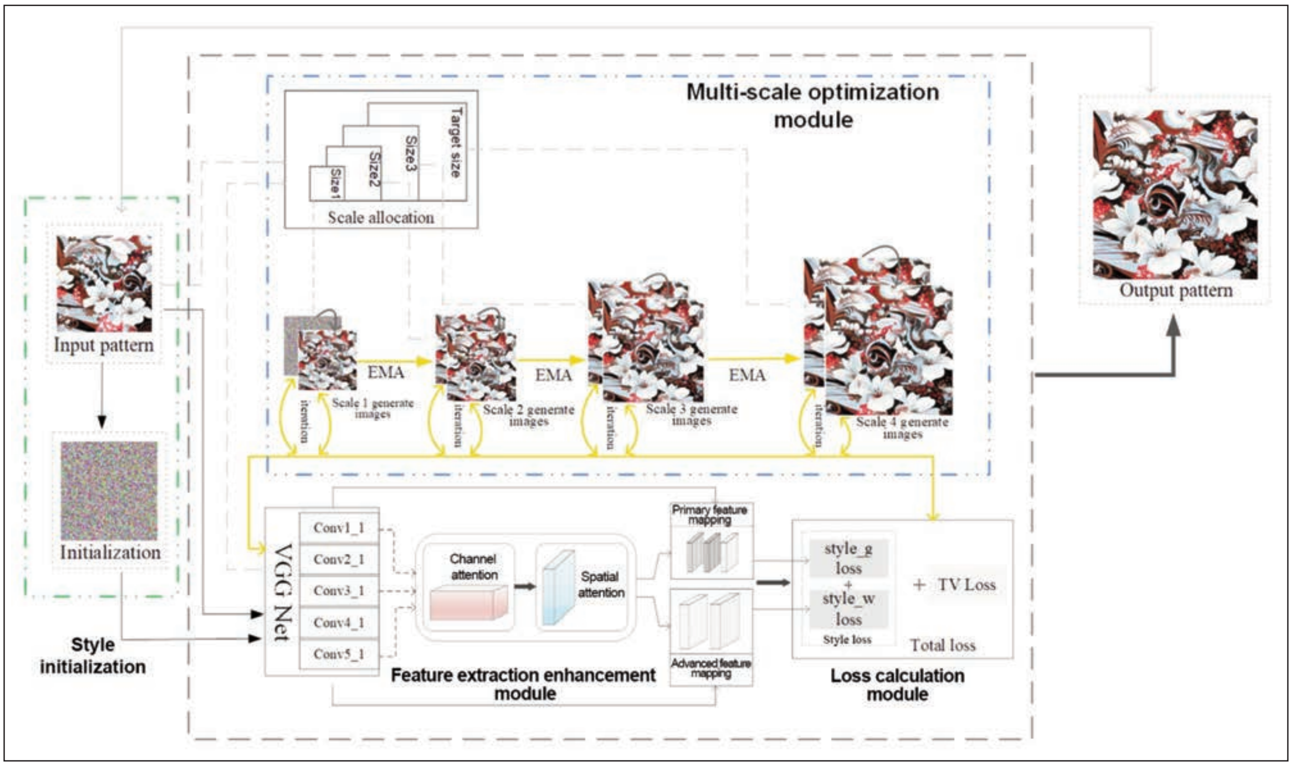

Fig. 1. Overall framework of the algorithm (arrows indicate the flow direction)

features extracted by Layer1, Layer2, and Layer3 in the network. These features mainly express the stylistic elements of the image, with little association to specific spatial structures. Gram matrices effectively capture fundamental texture and colour information [13]. Unlike traditional mean-squared error quantification of Gram matrix differences, shallow features are primarily edges, corners, and colour blocks, and their quantity is far less than the number of pixels. Therefore, this paper normalises features by dividing by the number of features. This approach balances the contribution of features rather than the overall pixel count, which is beneficial for expressing primary features. The feature-normalised Gram matrix is defined as:

$$G_{ij}^{norm} = \frac{\sum_k F_{ik} \cdot F_{jk}}{C} \tag{1}$$

where F_{ik} represents the feature value of the i^{th} channel at position k in the feature map of the input image, F_{jk} is the feature value of the j^{th} channel at the same position, and C denotes the number of channels in the feature map.

To calculate the mean error between the Gram matrices of target features and generated features, gradient normalisation is employed to mitigate numerical issues during parameter updates, thus enhancing the stability of the iterative process. The formula for calculating the shallow layer style loss function based on the Gram matrix is as follows:

$$L_{style_g} = \frac{\sum_{i,j}\left(G_{ij}^{norm}(g) - G_{ij}^{norm}(s)\right)^2}{\sum_{i,j}\left|G_{ij}^{norm}(g) - G_{ij}^{norm}(s)\right| + \epsilon} \tag{2}$$

Where $G_{ij}^{norm}(s)$ and $G_{ij}^{norm}(g)$ represent the feature-normalised Gram matrices of the style image and the generated image, respectively. The term ϵ is a regularisation term to avoid division by zero during computation.

Since Gram matrices do not preserve spatial information [19, 20], they struggle to accurately represent the structure and semantic information of high-level features, making it easy to overlook the content and complex components of the image in style reconstruction. For the high-level features extracted by Layer4 and Layer5, this paper uses the mean and covariance matrices to jointly describe the feature distribution of the target image and the generated image. By calculating the Wasserstein distance under a Gaussian distribution as the style loss, this approach better preserves spatial information and more comprehensively expresses the style information of deep features.

The specific method is as follows. First, compute the mean vector μ and the second-order raw moment matrix S of the input feature map and the target feature map at Layer4 and Layer5. The formulas are as follows:

$$\mu = \frac{1}{HW}\sum_{h=1}^{H}\sum_{w=1}^{W} X_{chw} \tag{3}$$

$$S = \frac{1}{HW}\sum_{h=1}^{H}\sum_{w=1}^{W} X_{chw} X_{dhw} \tag{4}$$

where H and W represent the height and width of the feature map, respectively, and c denotes the channel index.

Using the mean μ and the second-order raw moment S, the covariance matrix Σ of the features is calculated as follows:

$$\Sigma = S - \mu\mu^T + \epsilon I \tag{5}$$

Where ϵI is a regularisation term added to avoid numerical instability in the computation, and I is the identity matrix.

The squared difference of the feature means and the sum of the differences of the covariance matrices are used to approximately calculate the Wasserstein distance, which serves as the style loss for deep features L_{style_w}. The calculation method is as follows:

$$L_{style_w} = ||\mu_x - \mu_t||^2 + Tr\left(\Sigma_x + \Sigma_t - 2\left(\Sigma_t^{1/2}\,\Sigma_x\Sigma_t^{1/2}\right)^{1/2}\right) \tag{6}$$

where μ_x and μ_t represent the mean vectors of the input and target style image features, and Σ_x and Σ_t represent the covariance matrices of the input image features and the target style image features, respectively.

The total style loss function is:

$$\mathcal{L}_{style} = \Sigma_{i=1}^{3} w_l \cdot L_{style_g} + L_{style_w} \tag{7}$$

Where w_l denotes the loss weight of the l layer in the style layers, controlling the balance between the details and the overall structure in the generated image.

Convolutional attention feature enhancement module

The attention mechanism imitates human visual function by focusing on different parts of an observed object to varying degrees, filtering out redundant information, and retaining key information. This mechanism enables networks to concentrate on the main semantic structures of images and the critical texture strokes of style images [21–23]. Inspired by the CBAM attention mechanism [24], this paper incorporates a convolutional attention module into the feature extraction process, which includes both channel attention and spatial attention. The structure is depicted in figure 2.

Channel attention is represented as a diagonal matrix, with each diagonal element representing the channel weight. Feature information is extracted through average pooling and max pooling, then input into a multilayer perceptron (MLP) to learn the relationships between channels. The output of the MLP is element-wise summed, and the channel contribution is regulated by the Sigmoid activation function. The method is detailed as follows:

$$M_{chanle}(F) = \sigma\left(MLP\left(AvgPool(F_s)\right) + MLP\left(MaxPool(F_s)\right)\right) \tag{8}$$

where σ denotes the Sigmoid activation function, and F_s is the input feature map.

Spatial attention is calculated as a full matrix, generating an attention feature map using global information. The design principle of this module is to maintain feature expressiveness while being lightweight and effectively capturing key features. By applying average pooling and max pooling operations to compress the channel dimensions of the feature F_s', followed by convolution concatenation and Sigmoid activation, the final spatial attention feature map is generated. The process is represented as follows:

$$M_{spatial}(F_s') = \sigma\left(conv_{3\times3}\left(\left[AvgPool(F_s');MaxPool(F_s')\right]\right)\right) \tag{9}$$

where $conv_{3\times3}$ denotes a 3×3 convolution.

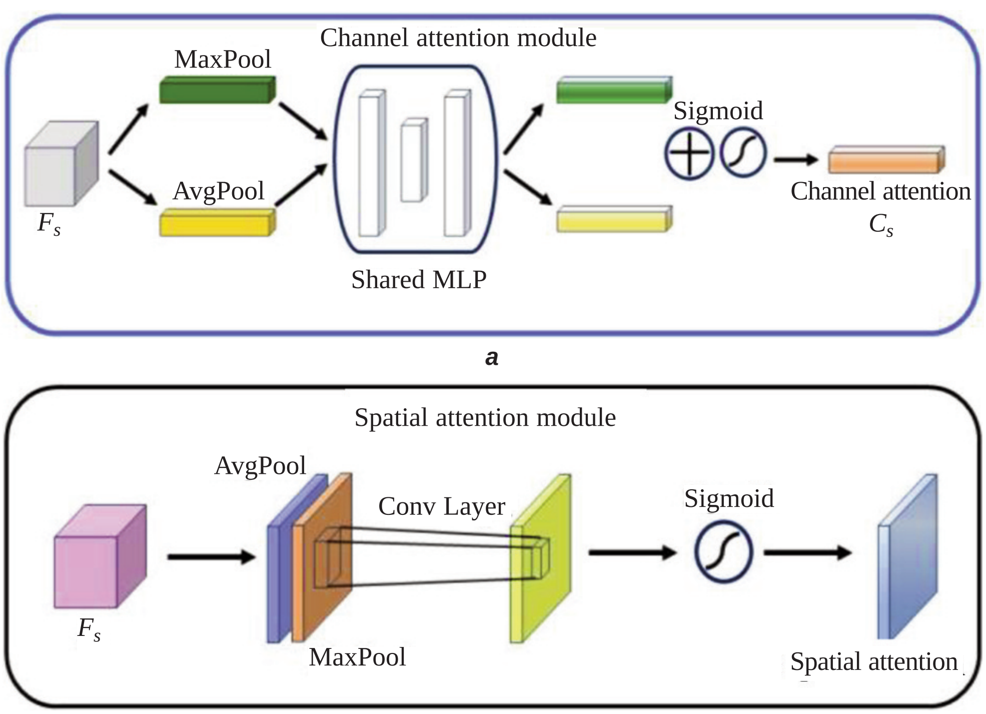

Fig. 2. Convolutional attention feature structure: *a* – Channel attention module structure; *b* – Spatial attention structure

To efficiently optimise the extracted features, inputs are derived from Layer1 and Layer3 in the shallow layers and Layer5 in the deep layers of the feature extraction network. The channel and spatial attention mechanisms are concatenated for feature refinement. The refined output features, combined with other layer features, are then used as style feature inputs into the multiscale optimisation module. Style image synthesis is jointly optimised using the feature loss functions L_{style_g} and L_{style_w} , as well as the total variation (TV) loss. This process is illustrated in figure 1.

Multiscale image optimisation module

The core idea of multiscale optimisation is to progressively optimise the image from low resolution to high resolution. This approach reduces computational cost and the risk of local optima associated with high-resolution optimisation, while maintaining global structure and refining details. First, define the scale range, from the minimum scale factor S_{min} to the maximum scale factor S_{max}, with a series of resolutions, each decreasing by a factor of $\sqrt{2}$. Each scale corresponds to a specific image resolution. Next, generate the initial image using the mean and variance of the style image to preserve the style structure and enhance image diversity.

To further improve the optimisation effect, an exponential moving average (EMA) method with bias correction is introduced during the multiscale optimisation process [26]. This method smooths the image update process, reduces noise and oscillations, and makes the generated images more stable and natural. The EMA update method is:

$$\boldsymbol{A}_t = \alpha \boldsymbol{I}_t + (1 - \alpha)\boldsymbol{A}_{t-1} \tag{10}$$

where $\boldsymbol{A}_t$ represents the EMA value after the th iteration, $\boldsymbol{I}_t$ represents the image after the t^{th} iteration, and α is the decay coefficient.

To ensure continuity and smoothness between different scales, scaling and bicubic interpolation sampling are used to adjust the image. Subsequently, the style image features at the current scale are calculated, and the style target is generated using the L_{style_g} and L_{style_w} losses for feature reconstruction. To optimise the quality and realism of the synthesised image, the total variation (TV) loss function [27] is added. The TV loss reduces noise and smooths the image by penalising local pixel value variations. The TV loss function is expressed as:

$$\mathcal{L}_{TV} = \Sigma_{i,j}\left((I_{i,j} - I_{i+1,j})^2 + (I_{i,j} - I_{i,j+1})^2\right) \tag{11}$$

Combining the L_{style_g}, L_{style_w} and TV loss, the total loss function is used to calculate the overall loss. This loss is backpropagated to update the image, and the optimised image is used as the initialisation for the next scale. The total loss function is expressed as:

$$\mathcal{L}_{toal} = \alpha \mathcal{L}_{style} + \beta \mathcal{L}_{TV} \tag{12}$$

where α and β are the weights of the respective loss terms.

EXPERIMENTS AND RESULTS

This paper utilises a dataset of collected textile patterns as the style images, with each image having dimensions set to 512×512. The experiments were conducted on a computer equipped with an NVIDIA RTX 3060 GPU, an Intel i7 processor. The deep learning framework used was Pytorch, and the programming language was Python. All experimental codes and models were implemented and executed in the aforementioned environment.

Comparison of experimental results of loss functions

This paper compares the reconstruction effects of different loss functions on the style layers through experiments and analyses the rationality of the combined loss function strategy. The experimental results show that using L_{style_g} and L_{style_w} loss functions on different feature layers result in significant differences in image synthesis effects.

Figure 3 shows the synthesis results of these two loss functions. Figure 3, *a* illustrates the results using L_{style_g} loss, and figure 3, *b* illustrates the results using L_{style_w} loss. The first layer of feature mapping mainly reconstructs primary features such as colour and texture; the second and third layers extract relatively complex texture features; the fourth and fifth layers capture high-level features, such as the semantic structure of floral shapes. For the first to third layers of features, the two loss functions do not differ much in the overall structure, but the images synthesised using L_{style_g} loss are smoother, especially in terms of colour and texture. In the fourth layer of feature mapping, the images reconstructed using L_{style_w} loss retain the complete floral structure and appear natural, whereas the images with L_{style_g} loss have distorted and unnatural floral structures. The fifth layer contains the most image feature mappings; using L_{style_g} loss results in a more chaotic structure, while L_{style_w} loss can reconstruct the floral pattern features more naturally.

As presented in table 1, in 500 iterations, the average time per layer for L_{style_g} loss is 43.5 seconds, and for L_{style_w} loss, it is 99.2 seconds. The latter has a higher computational cost but better effects on high-level feature layers. Therefore, to improve efficiency while ensuring the reconstruction effect of high-level feature layers, a combined loss strategy is adopted: L_{style_g} loss is used for the first three layers, and L_{style_w} loss is used for the last two layers.

Figure 4 shows the effects and synthesis time comparison of various loss functions. Figure 4, *a* is the textile print pattern used as the style image. Figure 4, *b* shows the pattern synthesised using L_{style_g} loss, where the floral pattern structure is distorted and messy, appearing unnatural. Figure 4, *c* shows the pattern synthesised using L_{style_w} loss, where the pattern structure is coherent but local textures are unclear, and the colour details are dull. Figure 4, *d* shows the pattern synthesised using the

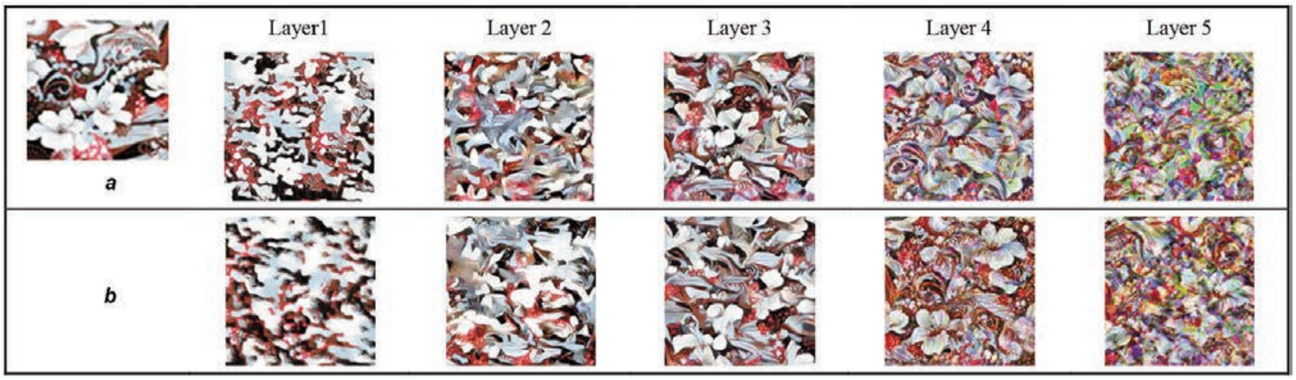

Fig. 3. The effect of different losses on image synthesis at different feature layers

Table 1

SINGLE LAYER TIME CONSUMPTION FOR DIFFERENT LOSS FUNCTIONS

Method	Time				
	Layer 1	Layer 2	Layer 3	Layer 4	Layer 5
L_{style_g}	42 s	41 s	43 s	45	47
L_{style_w}	107 s	88 s	87 s	102 s	112 s

Fig. 4. The impact of different style losses on synthesized patterns

combined loss function, where the floral pattern structure is complete, and the texture and colours are clear and bright. The overall visual effect is good, and the synthesis time is short, indicating that the combined loss strategy is reasonable and effective.

Subjective evaluation of experimental results

To evaluate the effectiveness of the proposed method, it was compared with the approaches of Gatys et al. [13], Li and Wand [15], Heitz et al. [20], and Kolkin et al. [19]. Figure 5 shows the images generated by each method.

Gatys et al.'s method utilises Gram matrices to capture the relative relationships between feature maps. However, this approach often produces artefacts when local region feature combinations are inconsistent (e.g., first row, first column, and second row, second column of figure 5). The semantic structure of the synthesised pattern can be incomplete, leading to a scattered floral structure (e.g., fourth row, first column of figure 5). While effective at maintaining overall style, this method has notable deficiencies in handling details, especially in high-frequency complex texture areas.

Li and Wand's method employs Markov Random Fields (MRFs) for texture synthesis, producing images with relatively clear local textures. However, it struggles with colour processing, often resulting in unnatural colours and noise, particularly in brightly coloured patterns (e.g., first row, second column; second row, second column; and fourth row, second column of figure 5). This noise diminishes the visual quality, making the synthesised images appear less smooth and realistic.

Heitz et al.'s method aims to capture complete feature distributions but suffers from detail loss and inaccurate edge handling due to the randomness of projection directions. Consequently, the synthesised textures differ in detail and edges from the original images, particularly in complex patterns (e.g., first row, third column of figure 5).

Kolkin et al.'s method focuses on local feature matching through self-similarity measures. Each local feature vector is matched independently, resulting in consistent local regions but inconsistent feature matching across the image. This leads to incoherent global structures and a lack of consistent global texture control, causing blurred textures (e.g., first row,

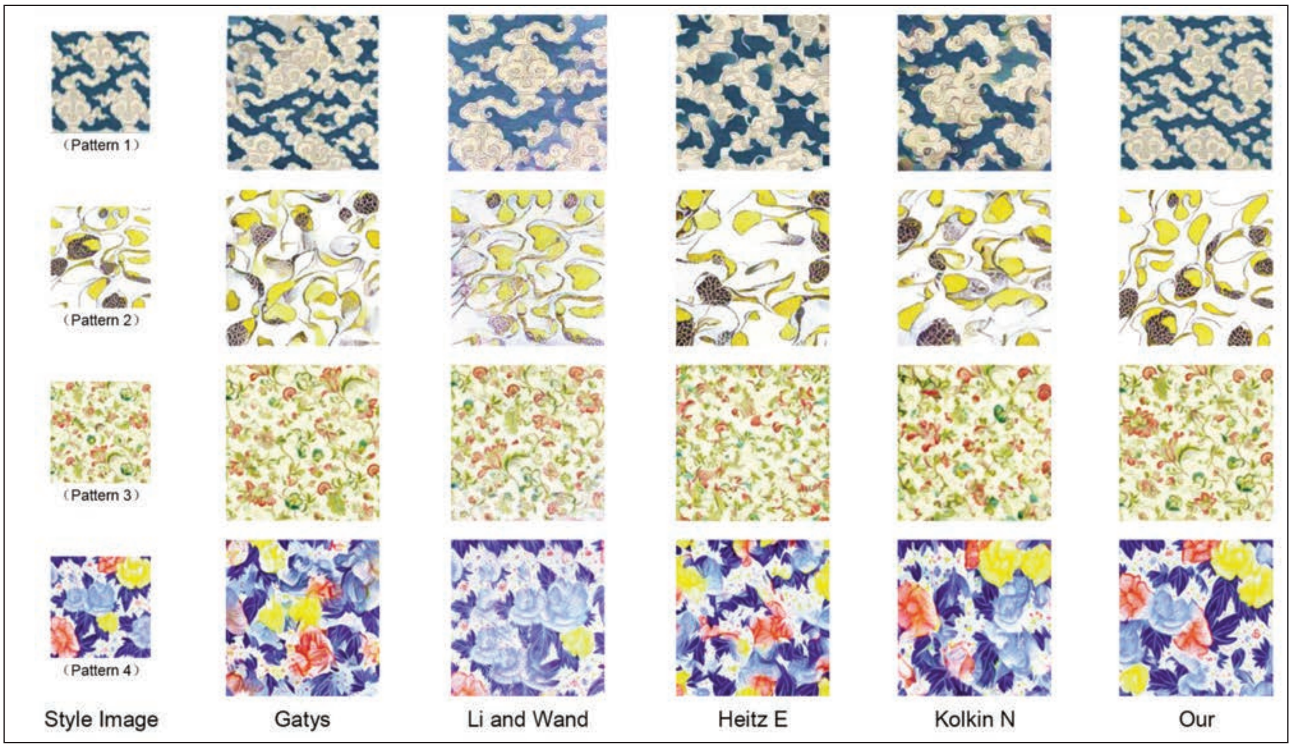

Fig. 5. Examples of fabric patterns generated by different algorithms

fourth column, and second row, fourth column of figure 5), which affects the overall visual effect.

In contrast, the proposed method excels in pattern generation. It employs hierarchical loss to handle different features, resulting in highly distinguishable pattern areas, clear structures, and rich style features. For example, in Pattern 1's high-frequency complex patterns (first row of figure 5), the proposed method produces a more natural synthesis, preserving the details and complexity. In Pattern 2's pattern (second row of figure 5), the targeted reconstruction of low-level and high-level features results in synthesised images with bright overall colours, clear edge details, minimal noise, and no obvious pattern deformation. Overall, the proposed method outperforms other comparative methods in synthesising complex textures and maintaining natural colours, demonstrating its effectiveness in textile pattern synthesis.

Objective Evaluation of Experimental Results

In addition to the subjective assessment of the generated results, an objective quantitative evaluation is also necessary. This paper uses the following metrics for comprehensive evaluation: Structural Similarity (SSIM) [28], Feature Similarity (FSIM) [29], Frechet Inception Distance (FID) [30], and generation efficiency at different resolutions.

Generation Efficiency Performance Evaluation: For the performance evaluation of generation efficiency, this paper compares the pattern generation time of different methods at various resolutions. Table 2 shows the generation time (in seconds) for five methods at four different resolutions. As shown in the table, the proposed method demonstrates higher efficiency at most resolutions, particularly showing a significant time advantage at the 1024×1024 resolution compared to other methods.

The evaluation metrics used provide a comprehensive view of the performance and quality of each method. The Structural Similarity Index (SSIM) is used to measure the similarity between two images, taking into account brightness, contrast, and structural information. The closer the value is to 1, the higher the similarity. The Feature Similarity Index

Table 2

TIME CONSUMPTION FOR TEXTILE PATTERN GENERATION BY DIFFERENT ALGORITHMS

Size	Time				
	Gatys	Li and Wand	Heitz E	Kolkin N	Our
128×128	19	314	86	55	31
256×256	33	693	186	77	48
512×512	78	x	712	118	80
1024×1024	261	x	x	263	242

Note: x indicates that it could not be generated or that it took too long to generate.

Table 3

COMPARISON OF SSIM AND FSIM VALUES OF FABRIC PATTERNS GENERATED BY DIFFERENT ALGORITHMS								
Method	**pattern 1**		**pattern 2**		**pattern 3**		**pattern 4**	
	SSIM	**FSIM**	**SSIM**	**FSIM**	**SSIM**	**FSIM**	**SSIM**	**FSIM**
Gatys	0.053	0.498	0.161	0.42	0.088	0.494	0.037	0.521
Li and Wand	0.950	0.559	0.059	0.443	0.031	0.534	0.043	0.613
Heitz E	0.058	0.519	0.093	0.405	0.091	0.499	0.042	0.505
Kolkin N	0.064	0.533	0.172	0.433	0.086	0.511	**0.065**	0.531
Our	**0.111**	**0.671**	**0.219**	**0.462**	**0.151**	**0.548**	0.054	**0.567**

(FSIM) improves upon SSIM by incorporating gradient and phase consistency, which aligns more closely with human visual perception. The Frechet Inception Distance (FID) evaluates the quality of image generation models, with lower FID values indicating higher quality. FID reflects the similarity between the generated images and real images in terms of statistical features.

Table 3 shows the SSIM and FSIM metrics for various methods across four sets of patterns, while table 4 presents the FID metrics. In these tables, the bold numbers indicate the best values within the same group. In table 3, the proposed method performs excellently overall, especially in the FSIM metric, where it achieves the best values for all patterns. This indicates a significant advantage in preserving pattern features. Additionally, the proposed method obtains the best values in three out of four sets of patterns for the SSIM metric, demonstrating its superior performance in structural similarity. In table 4, the proposed method achieves the lowest FID values in three out of four sets of patterns, and its overall FID value is lower than that of other methods, indicating strong image generation capability. Combining these metrics with generation efficiency, the comprehensive analysis shows that the proposed method has a significant advantage in preserving pattern features and generating high-quality images. This proves its effectiveness and reliability in the task of textile pattern generation.

Table 4

COMPARISON OF FID VALUES OF FABRIC PATTERNS GENERATED BY DIFFERENT ALGORITHMS				
Method	**FID**			
	pattern 1	**pattern 2**	**pattern 3**	**pattern 4**
Gatys	606.743	957.293	254.106	702.38
Li and Wand	**137.743**	1052.671	426.433	343.099
Heitz E	220.682	1109.606	1222.413	942.944
Kolkin N	360.702	988.794	482.411	374.394
Our	374.595	**607.989**	**228.73**	**291.368**

Ablation experiment

To verify the effectiveness of the attention feature module, a series of ablation experiments were conducted. As depicted in figure 6, figure 6, *a* displays two input style patterns: one is a traditional Chinese cloud pattern textile, and the other is a modern colourful fabric texture pattern. Figure 6, *b* presents the generated patterns without the channel-spatial attention module, while Figure 6, *c* shows the results after adding the channel-spatial attention module. Other parameters were kept constant throughout the experiments.

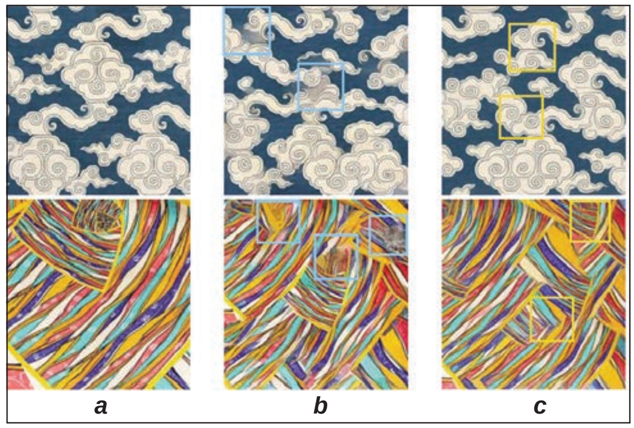

Fig. 6. Attention feature ablation experiment: *a* – input style patterns; *b* – generated patterns without the channel-spatial attention module; *c* – results after adding the channel-spatial attention module

From the experimental results, it is evident that the cloud pattern textile in figure 6, *b* exhibits artefacts at the overlapping cloud elements, and the transitions between pattern elements appear unnatural.

Similarly, in the colourful fabric texture pattern of figure 6, *b*, there are artefacts at the junctions of texture elements, with some areas showing colour overflow. In contrast, the generated patterns in figure 6, *c* show significant improvements in these aspects. The transitions between elements are more natural, and the detail and overall visual quality of the patterns are enhanced. Therefore, the attention feature module has a crucial impact on the generation effect of the algorithm and is an indispensable part of the algorithm.

CONCLUSION AND FUTURE WORK

This study proposes a fast multiscale synthesis algorithm for textile patterns based on neural networks. By combining low-level and high-level feature losses, the algorithm meticulously processes pattern features while preserving their semantic structure, achieving comprehensive style transfer. The convolutional attention feature enhancement module refines features, reduces artefacts, and improves visual quality, while the multiscale optimisation module enhances image texture and layering. Experimental results indicate that this method outperforms existing methods in terms of visual effect and scalability of pattern synthesis. By extracting and applying the style features of inspirational patterns, this method effectively helps designers focus on creativity and concept development, improving design efficiency and promoting the intelligent development of textile design and production. However, this study still has room for improvement. The generation time for high-resolution patterns needs optimisation, and there is a lack of interactive design features. Future research will explore the application of different attention mechanisms and interactive design modules, as well as improve the efficiency of high-resolution pattern generation.

REFERENCES

[1] Kristensen Johnstone, T., *Surface patterns, spatiality and pattern relations in textile design*, Diss. Högskolan i Borås, 2017

[2] Malik, B.S., Naheed Azhar, *Role of inspiration in creating textile design,* In: Journal of Engineering Research and Applications, 2015, 5, 5, 01–07

[3] Lee, A.L., Hyunsook, H., *A review of parametric clothing pattern CAD software methodology*, In: International Journal of Clothing Science and Technology, 2024

[4] Sikka, M.P., Alok, S., Samridhi, G., *Artificial intelligence (AI) in textile industry operational modernisation*, In: Research Journal of Textile and Apparel, 2024, 28, 1, 67–83

[5] Li, M., et al., *The Pattern Design of A Camouflage Suit Based on the Cycle Generative Adversarial Network Algorithm*, In: AATCC Journal of Research, 2023, 24723444231185633

[6] Wu, Q., et al., *ClothGAN: generation of fashionable Dunhuang clothes using generative adversarial networks*, In: Connection Science, 2021, 33, 2, 341–358

[7] Liu, M., Binbin, Z., *Innovative design of Chinese traditional textile patterns based on conditional generative adversarial network*, In: International Conference on Human-Computer Interaction. Cham: Springer International Publishing, 2022

[8] Shi, Q., Luo, R., *Research progress on clothing image generation based on generative adversarial networks*, In: Modern Textile Technology, 2023, 31, 2, 36–46, https://doi.org/10.19398/j.att.202203056

[9] Karagoz, H.F., et al., *Textile Pattern Generation Using Diffusion Models*, 2023, arXiv preprint arXiv:2304.00520

[10] Xie, Z., et al., *Hierarchical Fashion Design with Multi-stage Diffusion Models*, 2024, arXiv preprint arXiv:2401.07450

[11] Sun, Y., Chen, Y., *Fast textile pattern generation combining MRF-based and Gram-based methods*, In: Industria Textila, 2023, 74, 4, 439–445, http://doi.org/10.35530/IT.074.04.202254

[12] Qiu, X., Sun, W.L., Zhenghou, J., *Local style transfer of Nanjing brocade based on color optimization*, In: Silk, 2022, 59, 12, 64–70

[13] Leon, GA., Ecker, A.S., Bethge, M., *Image style transfer using convolutional neural networks*, In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016

[14] Johnson, J., Alexandre, A., Li, F.-F., *Perceptual losses for real-time style transfer and super-resolution*, In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part II 14, Springer International Publishing, 2016

[15] Chuan, L., Wand, M., *Combining markov random fields and convolutional neural networks for image synthesis*, In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2016

[16] Jie, L., Xu, D., Yao, S., *Sliced wasserstein distance for neural style transfer*, In: Computers & Graphics, 2022, 102, 89–98

[17] Youssef, M., *Wasserstein style transfer*, 2019, arXiv preprint arXiv:1905, 12828

[18] Huang, E., Sahil, G., *Style is a Distribution of Features*, 2020, arXiv preprint arXiv:2007, 13010

[19] Kolkin, N., Salavon, J., Shakhnarovich, G., *Style transfer by relaxed optimal transport and self-similarity*, In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019

[20] Heitz, E., et al., *A sliced wasserstein loss for neural texture synthesis*, In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021

[21] Fu, J., et al., *Dual attention network for scene segmentation*, In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019

[22] Yao, Y., et al., *Attention-aware multi-stroke style transfer*, In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019

[23] Deng, Y., et al., *Arbitrary style transfer via multi-adaptation network*, In: Proceedings of the 28th ACM International Conference on Multimedia, 2020

[24] Woo, S., et al., *Cbam: Convolutional block attention module*, In: Proceedings of the European Conference on Computer Vision (ECCV), 2018

[25] Jing, Y., et al., *Neural style transfer: A review*, In: IEEE Transactions on Visualization and Computer Graphics, 2019, 26, 11, 3365–3385

[26] Diederik, K.P., Ba, J., *Adam: A method for stochastic optimization*, 2014, arXiv preprint arXiv:1412, 6980

[27] Chambolle, A., *An algorithm for total variation minimization and applications*, In: Journal of Mathematical Imaging and Vision, 2004, 20, 89–97
[28] Wang, Z., et al., *Image quality assessment: from error visibility to structural similarity,* In: IEEE Transactions on Image Processing, 2004, 13, 4, 600–612
[29] Zhang, L., et al., *FSIM: A feature similarity index for image quality assessment,* In: IEEE Transactions on Image Processing, 2011, 20, 8, 2378–2386
[30] Heusel, M., et al., *Gans trained by a two time-scale update rule converge to a local nash equilibrium*, In: Advances in Neural Information Processing Systems, 2017, 30

Authors:

GUODONG XU, YU CHEN, JIAN HUA

School of Textiles and Fashion, Shanghai University of Engineering Science, China

Corresponding author:

YU CHEN
e-mail: ychen0918@sues.edu.cn