International Journal for Housing Science and Its Applications
Publish August 5, 2025. Volume 46, Issue 3 Pages 3765-3772

https://doi.org/10.70517/ijhsa463318

Research on automatic style migration method based on transfer learning in digital media art

Jiayi Yu[1,*]
[1] The Shanghai Conservatory of Music, The Institute of Digital Media Art, Shanghai, 200031, China
Corresponding authors: (e-mail: Aabc18321858305@126.com).

Abstract Image style migration is a research hotspot in the field of digital media art and artificial intelligence computer vision. The automatic style migration method based on transfer learning (TLST) proposed in this paper undergoes style extraction, style learning and style migration, obtains the style feature matrix, calculates the correlation between the features, and completes the remapping of the content of the original style image. The efficiency of the TLST algorithm is verified by comparing with other models and the application study on real devices. Specific results show that the Precision, Recall and F1 values of TLST are above 0.8, thus, TLST outperforms several other neural network models. The method in this paper reaches a maximum accuracy of about 100% after several trainings, which is better in artist painting recognition. Deploying TLST on MSP432P401R and ARM Cortex-M7 platforms reduces the average inference time from 156.90 seconds to 146.80 seconds and from 63.021 seconds to 59.324 seconds, respectively. This shows that TLST is able to reduce redundant computation and decrease inference latency.

Index Terms transfer learning, style migration, digital media art, feature matrix, remapping

I. Introduction

Digital technology promotes industrial change, and industrial development drives employment and domestic demand. Digital media art uses artificial intelligence, big data, cloud computing and other digital technologies to analyze user habits, and then design or produce humanized products that meet user needs and cater to user preferences. So that the creative industry clusters, traditional culture driven by high-end science and technology to achieve a perfect fusion, and promote the modernization of the media creative industry [1]-[4]. A report shows that the overall volume of China's media industry has been as high as more than 2 trillion yuan, accounting for 1/7 of the global media industry, mainly based on mobile content, movies, online advertising, newspapers, games, book sales, radio and television, Internet audio-visual program content and other industries [5]. The volume expansion of the digital media industry is getting bigger and bigger, and digital media art has become a visual object that can be seen everywhere in the public's life, and digitized paintings and images silently come into all aspects of the audience's life in the form of rich displays, and people can quickly associate animation, online exhibitions, and 3D or VR character images in their daily lives, and digital media art has become one of the most promising contemporary digital media art has also become one of the most promising emerging industries in contemporary times [6]-[9].

Under this demand, the number of professional designers in the digital media art industry is seriously insufficient, and the automated design process is particularly important. Among them, the design of the style module takes a lot of time, and the style migration technology accelerates the design speed of this module [10]. Style migration is the process of migrating the style of one image to another image so that the other image has the same content as the original. But with different artistic styles. Style migration techniques have been widely studied and applied in computer vision and computer graphics, and are used to generate movie special effects, fine art creation, and virtual reality [11]-[13]. At the same time, the style migration technology also reveals shortcomings in the application process. The analysis of the original style is not comprehensive, resulting in semantic distortion of the style migration works. In the face of the fusion of multiple styles, the gradient generated by different styles leads to the weakening of features between styles, and there are cross-border barriers to style migration under different media [14]-[17].

Transfer learning is the use of knowledge learned on one task to improve performance on another related task. Compared to traditional machine learning methods, migration learning can take full advantage of the similarities between different tasks, thus reducing the dependence on large amounts of labeled data, improving the generalization ability of the model, and providing a new path for automatic style migration [18].

Aiming at the traditional image style migration effect is relatively rough and the problem of poor robustness, and the development of artificial intelligence technology and a variety of deep learning model network this paper proposes the TLST method. First of all, through the TLST method of convolutional network loading model, calculate the value of the convolution, obtain the style extraction feature matrix, to lay the foundation for the next style learning process. After completing the style extraction and style learning, the original style image is evaluated and filtered by continuous iteration and thus style extraction, and the style feature matrix dataset is established. Finally, the trained dataset is applied to style migration to complete digital media art creation. In order to verify the application effect of the TLST method, the experimental results were comparatively analyzed and evaluated by comparing different neural network models.

II. Basics

Image style migration is a research hotspot in the field of computer vision, aiming at the remapping process of transplanting the style, texture and other features in one image to another image to generate a new image.

II. A.Transfer learning

Simply put, transfer learning is a machine learning method that reuses a pre-trained model for another learning task. If you want to train a deep neural network to recognize different breeds of cats and dogs, training it directly on the imageNet dataset would require millions of labeled images and massive GPU resources and training time. By choosing to use transfer learning, we can use pre-selected classification networks trained by Google or Microsoft, and only need to train individual neural network layers to achieve the same goal. In this case, only a few thousand images are needed and the training can be done in a short time using an ordinary workstation. However, when using migration learning, the pre-trained network should not be too far from the current task. When the task gap is too large. The learning effect will be poor [19].

II. B.Realization of style migration

Artistic image style migration. It consists of 3 main parts. Style texture feature extraction. Target image content extraction, style and content resynthesized into a new image. The content and style of an image are not completely unrelated, when making the content of one image have the style of another image. There does not exist an image that perfectly matches both content and style. Because image style and content can be represented separately in a convolutional neural network, the two representations are independently manipulated to produce new, perceptually meaningful pictures. The loss function minimized in the process of synthesizing images contains both content loss function and style loss function, which can be adjusted to coordinate between content and style to produce visually compelling images [20]. The overall implementation idea is as follows:

(1) Generate a random picture, the content of this picture can be generated from the original content picture, thus speeding up the training speed.

(2) Calculate the content loss function between the generated image and the content image as:

$$L_{content}(\vec{p},\vec{x},l)=\frac{1}{2}\sum_{i,j}(F_{ij}^{l}-P_{ij}^{l})^{2} \tag{1}$$

(3) Calculate the style loss function between the generated image and the stylized image as:

$$L_{style}(\vec{a},\vec{x})=\sum_{l=0}^{L}w_{l}E_{l} \tag{2}$$

(4) Minimize the loss function as:

$$L_{total}(\vec{p},\vec{a},\vec{x})=\alpha L_{content}(\vec{p},\vec{x})+\beta L_{style}(\vec{a},\vec{x}) \tag{3}$$

In this paper, the style migration network is used as VGG19 model this network has been pre-trained and is mainly used to extract style features and calculate the network loss function. The network loss includes style loss and content loss, the style loss needs to use the Gram matrix.

II. C.Gram matrix

The Gram matrix has the following functional form:

$$G_{ij}^{l}=\sum_{k}F_{ik}^{l}F_{jk}^{l} \tag{4}$$

The Gram matrix is mainly used to calculate the correlation between features, also known as the eccentricity covariance matrix, the diagonal elements of this matrix reflect the amount of each feature that appears in the image. The Gram matrix helps to capture the overall style of an image, to measure the difference in style between two images, simply compare the difference in the Gram matrix between them. The Gram matrix is calculated. The association of different features between the style image and the content image is also grasped to get the style of the work. By calculating the loss function from the Gram matrix, it is possible to compare the difference in style between the style image and the content image [21].

III. Automatic style migration method design

III. A. Presentation of the problem

One of the main challenges facing traditional image style migration methods in digital media art environments is how to adapt to changing scene characteristics under resource constraints. This chapter proposes a transfer learning-based automatic style migration (TLST) approach to address this challenge. By optimizing the inference paths that the samples pass through, the automatic style migration method based on migration learning significantly reduces the resource overhead and effectively ensures the reconstruction quality of image migration. When applied to the task of digital media art scene understanding, the method can effectively balance the performance and computational cost.

III. B. Methodological design

III. B. 1) Style Extraction

In this paper, TensorFlow core open source library is utilized to develop and train machine learning models for style extraction of images using TLST method. TLST is utilized to load the model and compute the convolution. Using the self.vgg_layers function, the image style feature values are constantly extracted hierarchically based on the markers; then np.reshape, tf.reshape, input.shape and other functions are comprehensively used to redefine the feature matrix. After many iterations of processing, the TLST convolution value is obtained, which can be used for further style feature matrix extraction.

Subsequently the convolved values are obtained by TLST, i.e., the image style feature matrix is extracted to form the digital material. Define the image style matrix gram_matrix of input_tensor type, according to the TLST convolution value, use input_tensor.shape, tf.matmul, tf.reshape and other functions to complete the numerical value of the style extraction matrix. The obtained style extraction feature matrix serves as a preliminary material for transferring from image material to digital material for style migration, and can be applied to the following style learning process.

As the first step in the style migration algorithm, the style extraction initially extracts the key feature information matrices from the manually delineated image material according to the TLST punctuation, which provides the original digital material for the subsequent machine learning. After the style extraction procedure, the image information in the painting is abstracted into digital information [22].

III. B. 2) Style learning

Image style learning based on the style migration algorithm, in essence, it is iterative style extraction of the original style image, and evaluation and screening of the extracted style feature matrix [23]. At present, there is a relatively unified set of standards for evaluating the effect of image style learning, and the loss function is shown in equations (5) and (6):

$$L_{style}(\vec{s},\vec{x},l)=\frac{1}{4N_l^2M_l^2}\sum_{i,j}(G_{ij}^l-A_{ij}^l)^2 \tag{5}$$

$$L_{content}(\vec{c},\vec{x},l)=\frac{1}{2}\sum_{i,j}(C_{ij}^l-X_{ij}^l)^2 \tag{6}$$

Equation (5) describes the style loss, where $\vec{s}$ is the style image, $\vec{x}$ is the generated image, l is the convolutional layer, the subscript i represents the number of channels, the subscript j is the location of the markers, A is the feature matrix of the style image $\vec{s}$, G is the feature matrix of the generated image $\vec{x}$, N_l is the number of feature channels in the l th layer, and M_l is the height and width product of the convolution of the l th layer. Equation (6) describes the content loss, where $\vec{c}$ is the content image, $\vec{x}$ is the generated image,

l is the convolution layer, i represents the number of channels, the subscript j is the location of the markers, G is the feature matrix of the generated image $\vec{x}$ and X is the feature matrix of the content image $\vec{c}$.

In Eq. G_{ij}^{l} is defined as:

$$G_{ij}^{l} = \sum_{k} F_{ik} F_{jk} \tag{7}$$

Equation (7) describes the image texture information and color information, and F represents the response of the filter at the corresponding node, i.e., the value corresponding to the punctuation. The A_{ij}^{l}, X_{ij}^{l} definition is the same as G_{ij}^{l}.

The style loss assessment mainly compares the difference between the style image and the generated image by the difference between the marker points to reflect the efficiency of style imitation. The content loss assessment compares the difference between the content image and the generated image by marker points to reflect the efficiency of content retention.

The style learning process continuously corrects the deviations present in the feature matrix obtained from style extraction and forms the dataset. The image style loss and content loss can be evaluated to reflect the effectiveness of learning. In the program, the results of these two formulas are considered comprehensively, and the image is punctuated to extract the feature information matrix and compared to train the data set quickly. In addition, when there exists a more significant dataset bias, human intervention is required to recalibrate the marker points or adjust the parameters to correct the dataset.

The matrix formed by machine learning is corrected on the basis of style loss and content loss assessment. This is supplemented by manual labeling and iterative modification of the parameters and marker points for style extraction. Iteratively perform style extraction loss assessment-machine learning matrix correction to complete style learning.

III. B. 3) Style migration

In the design of this paper, after completing the style extraction and style learning based on the style migration algorithm, we have obtained the style feature matrix of digital media art content such as characters, plants, animals, grass, trees, sky, etc., which is a huge dataset. The trained dataset can be applied to the next style migration, and its technical process is shown in Fig. 1. After three steps of style extraction, style learning and style migration, the digital media art design based on style migration algorithm is finally completed [24].

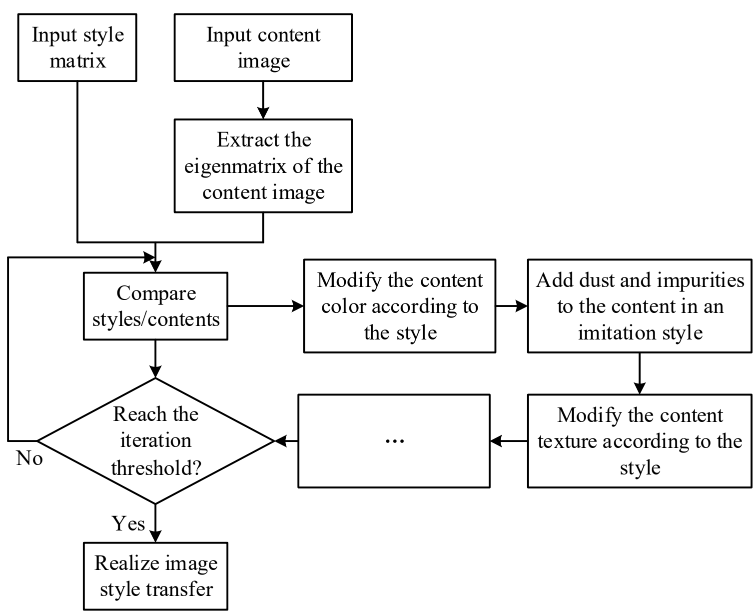


Figure 1: Technical Process of style transfer

IV. Experiments and analysis of results

IV. A. Experimental data set and data processing

In order to train the transfer learning-based automatic style migration (TLST) method to recognize painters, the WikiArt dataset is used, the entire dataset includes over 100,000 digital media artworks by 2,300 painters that span a very wide breadth of time and styles, the dataset has abstraction, realism, figure, landscape, ink, and oil paintings. The vast majority of artists in the entire dataset have fewer than 50 paintings, so to ensure sufficient sample size for training the network, the experiments only use artists with 400 or more paintings in the dataset. To ensure a balanced number of paintings per artist, only 400 paintings were taken from these artists. Thus the dataset consists of 400 paintings from 23 artists (9200 paintings in total). The experiment splits this dataset into a training set validation set and a test set by splitting each artist painting by 80-10-10. Thus, 320 paintings per artist in the dataset were used as the training set, 40 paintings as the validation set, and 40 paintings as the test set. This dataset is much larger than the dataset used in previous work.

The paintings in the dataset have different shapes and sizes, so the images are modified. The images were zeroed and normalized, and any portion of the paintings were randomly cropped while training the network, with a 224 x 224 pixel crop for each input image; this randomness increased the diversity of the training data and helped to avoid overfitting. When validating and testing the images, the experiment also performs random 224 × 224 pixel cropping of the images to ensure stability and repeatability of the results. The experiment assumes that the artist's style is ubiquitous in the image and not just limited to a specific region, so random cropping of paintings can contain enough information to determine the digital media art style.

IV. B. Experimental setup and environment

Model training and testing are done in Py Torch framework. The experiments are done on cloud platform with GTX1080Ti GPU, 16G RAM and 200GB storage.

The experiments train two models by Adam update rule. Neural networks of ResNet-50 and DenseNet-201 for migration learning, using the default Adam parameter, learning-rate=10-3, the experiments can observe the accuracy and loss of the training and validation datasets throughout the training phase, and when the neural network performance no longer improves, the learning-rate is reduced by a factor of 10 until the finally the neural network training accuracy no longer improves.

IV. C. Comparison and analysis of experimental models

IV. C. 1) Comparison of experimental results

Comparison of the results of the experiment is shown in Table 1.The experiment compares 5 neural network models and it can be seen that Top-1 has Precision, Recall and F1 accuracy. The experiments are compared with other models which demonstrate the accuracy of 3 neural network classifications.Precision, Recall and the F1 of are defined as follows:

$$\Pr ecision = \frac{TruePositives}{TruePositives + \mathrm{False}Positives} \tag{8}$$

$$\mathrm{Re}\,call = \frac{TruePositives}{TruePositives + \mathrm{False}Negatives} \tag{9}$$

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \tag{10}$$

where TruePositives and FalsePositives denote the number of positive samples correctly and incorrectly recognized, and TrueNegetives and FalseNege-tives denote the number of negative samples correctly and incorrectly recognized, respectively. The table compares the accuracy of different models of CNNs in key metrics, TLST outperforms Baseline CNN, ResNet-18 from scratch and ResNet-18 from transferlearning 3 neural network models, and also outperforms the DenseNet-201 model, the experimental results show that it is not the deeper neural network that works better.

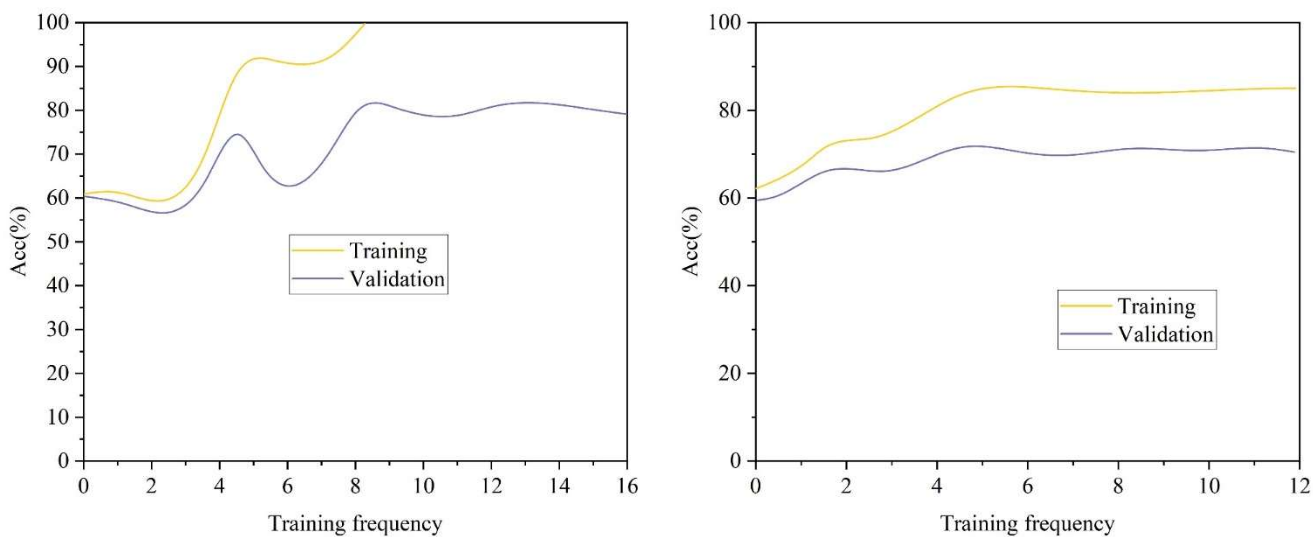
Table 1: The artist classification results summarized

Model	Top-1				
	Tranin Acc	Test	F1	Precision	Recall
Baseline CNN	0.442	0.430	0.364	0.422	0.369
ResNet-18 from scratch	0.521	0.516	0.514	0.520	0.514
ResNet-18 from transferlearning	0.911	0.778	0.772	0.779	0.775
TLST	1.0	0.811	0.806	0.810	0.814
DenseNet-201 from transferlearning	0.852	0.714	0.711	0.714	0.709

IV. C. 2) Analysis of experimental results

The change in accuracy of the two neural networks in training is shown in Fig. 2, as shown in Fig. (a) the experiment found that the TLST based on migration learning started to train the model, and the accuracy improvement curves of the two models were similar for the training set and the valid set. The experiment is divided into four stages, each stage 4 times epochs training, each stage is completed after the learning rate decreased by 10 times, as shown in Figure (a) the first stage of the training set and the effective set of the accuracy has been improved, in which the second stage of the accuracy improvement is not obvious, the third stage of the accuracy improvement is rapid, it can be seen that the different learning rate of the model has a different effect of improving, the fourth stage said that it is no longer improved.

Figure (b) shows that the initial accuracy of DenseNet-201 model is very high, and then the accuracy improves very slowly, and the accuracy reaches up to about 90% after several trainings.The first stage of the model that DenseNet-201 starts to learn is obviously improved, and there is a curve increase in the accuracy in the second stage when the learning-rate is reduced by 10 times. There is no improvement in the third stage and the accuracy is relatively low. Experimentally verified that TLST is better in artist painting recognition compared to DenseNet-201 neural network.



(a) TLST (b) DenseNet-201

Figure 2： Accuracy of different models

IV. D. Parameter sensitivity analysis

In this section, the sensitivity of the TLST model to the reward function parameters γ and t on the Set5 dataset is evaluated, and the experimental results are shown in Table 2. The results show that the parameters γ and t have significant effects on the performance and computational efficiency of the super-resolution model. First, the γ parameter is mainly used to control the gradient strength of the reward function. A large negative value (e.g., γ = -100) increases the penalty effect of the bias on the reward, thus tightly constraining the model optimization path and helping to reduce the number of residual blocks used. However, too large a penalty may limit the model's degrees of freedom, leading to a decrease in image detail recovery. In contrast, smaller negative values (e.g., γ = -10) relax the constraints on the bias and allow the model to optimize over a larger solution space, thus enhancing image detail restoration. Second, the t parameter acts as a threshold regulator and directly affects the number of residual blocks used. Larger values of t (e.g., t = 100) allow the model to use more residual blocks, thus enhancing the reconstruction capability of the model, but also significantly increasing the computational cost. Smaller t-values (e.g., t=10), on the other hand, limit the number of residual blocks used, which significantly improves the inference efficiency, but may lead to insufficient image detail recovery.

Experimental results further validate the role of the above mechanisms. When γ = -10 and t = 40, the model achieves the best performance with a PSNR of 37.54 and only 27 residual blocks are used. This suggests that with reasonable parameter tuning, the model can achieve a good balance between super-resolution quality and computational efficiency. However, under some extreme parameter settings, e.g., γ = -100 and t = 120, although the PSNR remains high at 37.422, the number of residual blocks used increases to 32, which significantly reduces the inference efficiency. Similarly, although the number of residual blocks used decreases to 26, the PSNR decreases to 37.437, failing to achieve optimal performance. Taken together, the tuning of γ and t plays a key role in balancing model performance and efficiency. The experiments show that as γ increases from -100 to -10, the PSNR gradually improves and the number of residual blocks used gradually decreases, indicating that appropriately lowering the penalty for bias helps optimize the super-resolution effect. While the change of t value has a more direct impact on the performance and efficiency, t=40 strikes a better balance between the two, which ensures the improvement of PSNR and reduces the computational overhead.

Table 2: Parametric sensitivity analysis

γ	t	PSNR	Blocks Used
-100	40	37.452	29
-50	40	37.39	28
-10	40	37.54	27
-100	120	37.422	32
-100	60	37.418	30
-100	20	37.437	26

IV. E. Deployment testing of edge devices

In order to verify the efficiency of the TLST algorithm on real devices, this section deploys TLST on two platforms, Texas Instruments MSP432P401R and ARM Cortex-M7, and analyzes its inference performance in detail to obtain the test results shown in Table 3. It can be seen that on the MSP432P401R platform (100MHz clock frequency), the average inference time of TLST is reduced from 156.90 seconds to 146.80 seconds, and the reduction in inference time for a single sample ranges from 3.687 seconds to 13.642 seconds. On the ARMCortex-M7 platform (250MHz clock frequency), the inference time is reduced from 63.0205 seconds to 59.324 seconds, and the reduction in inference time for a single sample ranges from 1.573 seconds to 7.461 seconds, both with an average speedup ratio of 6.6%.

Table 3: Deployment test for edge equipment

Texas Instruments MSP432P401R						
	Data0	Data1	Data2	Data3	Data4	Data5
Baseline	385.639	105.46	92.346	96.254	103.89	157.83
Ours	371.958	92.985	74.52	90.784	103.28	147.28
ARM Cortex-M7						
Baseline	154.92	42.553	37.938	38.471	41.196	63.045
Ours	150.378	38.202	30.56	36.311	41.082	59.412

The above results show that TLST is able to effectively reduce the inference latency by reducing redundant computations through an intelligent path selection mechanism, and its lightweight design also ensures the algorithm's adaptability and usability on low-power embedded devices. However, in scenarios with higher real-time requirements (e.g., traffic monitoring or public safety tasks), it is still important to further reduce the inference latency. In the future, the performance of the algorithm in ultra-high real-time scenarios can be further improved by combining FPGA or GPU hardware acceleration, as well as optimizing the path selection strategy and network structure, to fully exploit its potential for application in urban edge computing scenarios.

V. Conclusion

This paper is oriented to adapt to changing scenarios with limited resources and proposes a self TLST method based on migration learning. Using the WikiArt dataset of more than 100,000 digital media artworks as the research object, model training and testing are completed under the Py Torch framework to verify the superiority of the performance of this paper's method. The results show that.

By comparing different models in Precision, Recall and F1 value key indexes, the performance of TLST is better than other models, and its effect is better in artist painting recognition. In the parameter sensitivity analysis, when γ = -10 and t = 40, the PSNR reaches 37.54, and the model in this paper achieves the best performance. Therefore, when the parameters are reasonably adjusted, the method in this paper can balance the super-resolution quality and computational efficiency. Deploying the TLST algorithm on real devices, it can be found that TLST not only arbitrarily reduces the redundant computation and lowers the inference delay, but also guarantees the adaptability and availability on low-power embedded devices.

References

[1] Gong, Y. (2021). Application of virtual reality teaching method and artificial intelligence technology in digital media art creation. Ecological Informatics, 63, 101304.
[2] Marcos, A. (2020). Digital media art: creating, intervening and investigating in the intersection between media, art and computing. 2nd DCT Virtual Week," A cup of tea with Masters".
[3] Liu, S., & Li, T. (2020). AVG comprehensive practice reform of digital media art based on fuzzy theory. Journal of Intelligent & Fuzzy Systems, 38(4), 3697-3706.
[4] Sugita, I. W., Setini, M., & Anshori, Y. (2021). Counter hegemony of cultural art innovation against art in digital media. Journal of Open Innovation: Technology, Market, and Complexity, 7(2), 147.
[5] Hong, Y., & Zhu, Y. (2023). Evaluation of the high-quality development of China's media industry. Heliyon, 9(8).
[6] McMillan, S. J., & Childers, C. C. (2017). A decade of change and the emergence of digital media: Analysis of trade press coverage of the advertising industry, 2005–2014. Journal of Interactive Advertising, 17(1), 51-64.
[7] Jiang, R., Wang, L., & Tsai, S. B. (2022). An empirical study on digital media technology in film and television animation design. Mathematical Problems in Engineering, 2022(1), 5905117.
[8] Wu, Y. (2022). Application of artificial intelligence within virtual reality for production of digital media art. Computational Intelligence and Neuroscience, 2022(1), 3781750.
[9] Zhu, C., & Zhu, P. (2022). Application of network digital media interaction for exhibition using the internet technology. Mobile Information Systems, 2022(1), 1474918.
[10] Suh, S. (2020). Non-boundaries of style represented in fashion Instagram: a social media platform as a digital space–time. Fashion and Textiles, 7(1), 33.
[11] Li, S. (2023). Application of artificial intelligence-based style transfer algorithm in animation special effects design. Open Computer Science, 13(1), 20220255.
[12] Peng, X. (2025). Research on the Generation of Artistic Patterns in Fashion Design Based on Style Transfer. International Journal of High Speed Electronics and Systems, 2540460.
[13] Vasiliu, V., & Sörös, G. (2019, October). Coherent rendering of virtual smile previews with fast neural style transfer. In 2019 IEEE International Symposium on Mixed and Augmented Reality (ISMAR) (pp. 66-73). IEEE.
[14] Yu, J. (2025). Design of a neural network-based automated style migration technique in digital media art. J. COMBIN. MATH. COMBIN. COMPUT, 127, 9561-9576.
[15] Zhang, S., Qi, Y., & Wu, J. (2025). Applying deep learning for style transfer in digital art: enhancing creative expression through neural networks. Scientific Reports, 15(1), 11744.
[16] Zhang, Z., Wei, F., Liang, G., & Wang, X. (2024). Graphic Style Transfer Technology in "Multimedia communication": Application of Deep Residual Adaptive Networks in Graphic Design. National Journal of Antennas and Propagation, 6(3), 109-117.
[17] Zheng, S. (2024). Deep learning-based artistic style transformation algorithm in visual communication. International Journal of System Assurance Engineering and Management, 1-10.
[18] Hosna, A., Merry, E., Gyalmo, J., Alom, Z., Aung, Z., & Azim, M. A. (2022). Transfer learning: a friendly introduction. Journal of Big Data, 9(1), 102.
[19] Yongxu Shen,Chang Wang,Zhihui Sun,Faxiang Zhang,Shaodong Jiang,Xu Liu & Fengxia Gao. (2025) .A Fast recognition method for DVS perimeter security intrusion signals based on transfer learning. Optical Fiber Technology,93,104227-104227.
[20] Igeon Hwang & Taeyeon Oh. (2025) .Design and experimental research of on device style transfer models for mobile environments. Scientific Reports,15(1),13724-13724.
[21] Ren Guo & Ying Wang. (2025) .Eigenvalue properties of gram matrices associated with simplexes and polygons. Journal of Geometry,116(2),7-7.
[22] Wang Qi & H Alamusi. (2022) .Design of Watercolor Cultural and Creative Products Based on Style Transfer Algorithm. Mathematical Problems in Engineering,2022.
[23] Zhijie Lin,Jingjing Chen,Xiaolong Ma,Chao Li,Huiming Zhang & Lei Zhao. (2024) .Unsupervised content and style learning for multimodal cross-domain image translation. Scientific Reports,14(1),29469-29469.
[24] Honggang Zhao,Beinan Zhang & Yi Jun Yang. (2025) .Contrastive attention and fine-grained feature fusion for artistic style transfer. Journal of Visual Communication and Image Representation,110,104451-104451.