IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 7, 2021, accepted January 17, 2021, date of publication January 27, 2021, date of current version April 14, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3054969

Image Style Transfer Algorithm Based on Semantic Segmentation

ZHIJIE LIN[1], ZHIZHONG WANG[2], HAIBO CHEN[2], XIAOLONG MA[3], CHUAN XIE[4], WEI XING[2], LEI ZHAO[2], WEI SONG[1]

[1]School of Information and Electronic Engineering, Zhejiang University of Science and Technology, Hangzhou 310023, China
[2]College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China
[3]School of Management, Huzhou University, Huzhou 313000, China
[4]Hangzhou Vocational and Technical College, Hangzhou 310018, China

Corresponding authors: Wei Xing (wxing@zju.edu.cn), Lei Zhao (cszhl@zju.edu.cn), Wei Song (esongok@126.com)

This work was supported in part by the Construction and Verification of Digital Identity System for Folk Cultural Relics under Grant 2020YFC1523201, in part by the Research on Key Methods of Intelligent Extraction, Understanding and Generation of Elements in Large Ruins under Grant 2020YFC1523101, in part by the Collaborative Processing and Intelligent Computing Engine of Digital Cultural Heritage under Grant 2020YFC1522701, in part by the Research on Key Technology of Image Inpainting based on Background Knowledge Constraint under Grant LY21F020005, in part by the Management Oriented Network Mechanism Design and Data Mining Method under Grant LGF19F020003, in part by the Research on Key Techniques of Image Inpainting based on Weak Supervised Learning under Grant LY19F020049, in part by the Research on Image Semantic Segmentation Technology under Grant LY20G010003, and in part by the Research on Pattern Creative Design Technology under Grant 2019C03137.

ABSTRACT Most of the existing image style transfer algorithms transfer the whole image style as a whole. Style feature is a set of correlation matrix based on style image, namely Gram matrix. Each matrix is a global description of the style image. This kind of methods can perform well in the insensitive semantic scenes (such as the style transfer between landscape photos), but in the sensitive semantic scenes (such as the style transfer between portrait photos), the problem of semantic mismatch will be highlighted, such as transferring the background texture of the style image to the foreground of the target image. Although the existing research takes the manually annotated semantic image as an input of the algorithm, and then guides the style transfer based on the semantic information, and finally achieves good results in the style transfer between portraits. But there are still two problems: first, semantic images need to be manually annotated, which costs human resources. In practical applications, large-scale image style transfer is often needed. Second, the details of the synthesized image are fuzzy, and the definition is not enough. We propose an image style transfer algorithm based on semantic segmentation to resolve semantic mismatching in image style transfer. Our algorithm extracts the semantic information of style image and content image automatically through a semantic segmentation network and uses the semantic information to guide the style transfer. Our algorithm builds a semantic segmentation network based on mask R-CNN, introduces semantic information, and then makes style transfer on the patch level, realizes the style transfer between similar objects (consistent semantic information). Experiments on Celeba and Wikiart show that our method could automatically extract the semantic information of style image and content image. Compared with the state-of-art approaches in this field, our method can effectively avoid semantic mismatch in the process of image style transfer. That is, it can maintain semantic consistency in the process of style transfer.

INDEX TERMS Image style transfer, semantic segmentation, semantic mismatching, feature extraction, fine semantic guidance, mask R-CNN.

I. INTRODUCTION

Most of the existing image style transfer approaches transfer image styles as a whole. Taking Gatys' method as an example, the style feature is a set of correlation matrices calculated based on style image, namely Gram matrix. Each matrix is a global description of the style image. This kind of method can perform well in insensitive semantic scenes (such as style transfer between landscape photos), but in the insensitive semantic scenes (such as style transfer between portraits), the problem of semantic mismatching will be highlighted, such as transferring the background texture of the style image to the foreground of the target image. Champanard took the manually annotated semantic image as an input of

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

the algorithm, then guided the style transfer based on the semantic information, and finally achieved good results in the style transfer between portraits. However, Champanard's algorithm has two problems: first, semantic images need to be manually annotated, which requires a lot of human resources and material resources. In practical applications, large-scale image style migration is often needed. For example, a large number of promotional posters are automatically designed for e-commerce websites. Obviously, the algorithm is not suitable for this scenario; second, the details of the synthesized images are fuzzy, and the clarity is not enough.

We propose an image style transfer algorithm based on semantic segmentation (the whole method is abbreviated as SST) algorithm based on semantic segmentation. The algorithm builds a semantic segmentation network on the mask R-CNN, introduces semantic information, and then performs style migration at the image block level to realize the style transfer between similar objects (with consistent semantic information). To prove the effectiveness of SST algorithm, we do a comparative experiment on Celeba

II. RELATED WORK

Mordvintsev *et al.* [1] iteratively optimized the image to make the CNN feature of the image consistent with the target CNN feature, and the resulting image is the stylized image. This style transfer method combined with visual texture modeling [2] technology leads to the online image style transfer (IOB-NST) algorithm based on image optimization, which lays the foundation of the NST field. The basic idea of IOB-NST is as follows: firstly, the style and feature are modeled, then the content feature and style feature are extracted from the content image and the style image, respectively, and then combined into the target feature. Finally, the stylized image is constructed iteratively to make its feature and target feature match. In general, the IOB-NST algorithm is the same idea. The difference is the way of modeling style. Because of the iterative optimization steps, the IOB-NST algorithm generally has a large amount of calculation and low efficiency.

The algorithm proposed by Gatys *et al.* [3] is an important representative of IOB-NST. By reconstructing the features of the middle layer of the VGGNET19 network, Gatys and others found that the deep convolution neural network can extract content information from any photo and style information from any artistic image. According to this finding, Gatys *et al.* Reconstructed the content information by reducing the difference of high-level features between the content image and the target image and reconstructed the style information by reducing the difference of the statistical features of the Gram matrix between the style image and the target image. Gatys and other algorithms do not need a training set and do not require the type of style, which solves the problems of the IB-AR algorithm before deep learning. But the algorithm also has its own shortcomings because CNN features inevitably lose low-level information, so the algorithm cannot maintain the consistency of fine structure in the process of stylization and will also lose a lot of detailed information; because of the limitation of Gram matrix representation style, the algorithm can not synthesize high realistic images. In addition, the stroke, semantic, and depth information in the content image are not considered in the algorithm, which are important indicators to evaluate the visual quality.

Gram matrix is not the only choice for statistical features of coding style. There are many style representation methods based on the Gram matrix. Li *et al.* [4] treated style migration as domain adaptation and proposed a new style representation method. Given training sets and test sets from different distributions, the goal of domain adaptation is to train a model based on labeled training sets in the source domain so that the model can predict unlabeled test data sets in the target domain. The basic idea of domain adaptation is to establish the relationship between the source domain samples and the target domain samples by minimizing the differences between the distributions. The maximum mean difference (MMD) is usually used to measure the difference between the two distributions. Li *et al.* Have proved that the Gram matrix matching the style image and the target image is equivalent to minimizing MMD with quadratic polynomial kernel. Therefore, the use of other kernel functions can also be used for style transfer, such as linear kernel, polynomial kernel, Gaussian kernel, and so on. In addition to MMD, there is also a modeling method based on BN (batch normalization). Li *et al.*'s main contribution is to prove theoretically that the style matching based on Gram matrix is equivalent to minimizing MMD based on quadratic polynomial kernel, which provides a reasonable explanation for NST and makes the theory of NST clearer. However, Li *et al.*'s algorithm still does not solve the shortcomings of Gatys' algorithm.

The algorithm based on the Gram matrix is not stable enough in the optimization process, so it needs to adjust the parameters manually, which is very cumbersome [7]–[12]. Risser *et al.* [5] found that two feature maps with great differences in mean and variance may still have the same Gram matrix, which is one of the main reasons for the instability of the Gram matrix. Inspired by this discovery, Risser *et al.* Introduced a histogram loss function to match the entire histogram of a feature graph in the optimization process. In addition, Risser *et al.* Also proposed an experimental scheme of automatic parameter adjustment, which can explicitly avoid the occurrence of extremum in the gradient through standardized operation. The above-mentioned NST algorithms only compare the similarity between the content image and the target image in the CNN feature space [13]–[17]. However, due to the inevitable loss of the low-level information of the image, the distortion and irregular defects often appear in the composite image. In order to preserve the consistency of fine structure in the process of stylization, Li and Wand [6] imposed additional restrictions on low-level features in pixel space. Specifically, the Laplacian loss function was introduced to calculate the Euclidean distance of Laplacian filter responses between the content image and the target image. Laplacian filter is used to calculate the second

derivative of pixels, which is widely used in the field of edge detection.

Li *et al.*'s algorithm can keep the consistency of fine structure in the process of stylization, but it still does not consider the semantic, depth information, stroke, and so on.

Li *et al.* [7] first proposed an NST algorithm based on Markov random fields (MRF). Li *et al.* found that although the statistical feature-based NST algorithm captures the correlation of pixel features, it ignores the spatial structure of the image, which makes it not ideal in the migration of photorealistic styles. Therefore, he proposed a new style loss function, which introduced the prior knowledge of MRF based on patch.

The advantage of Li's algorithm is that it can deal with high realistic style transfer, especially when the shape and perspective of content image and style image are similar. The effect is very good. However, when the shape and perspective of the content image and style image are too different, the patch will be mismatched, and the migration effect is not ideal. In addition, the ability of the algorithm to retain details and depth information is not strong. Texture synthesis is often used for feature extraction [28]. We use mask R-CNN network to extract fine semantic features and utilize them to guide semantic style transfer.

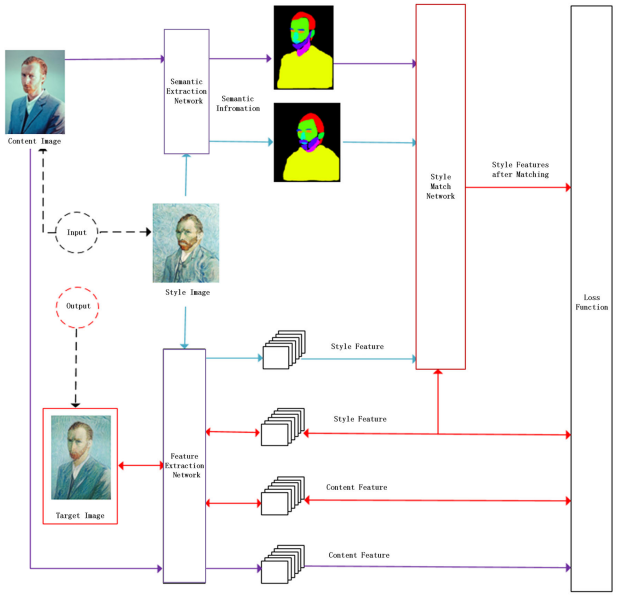


FIGURE 1. Overall structure of image style transfer algorithm based on semantic segmentation.

III. METHOD

In order to facilitate the following description, this paper agreed that the target image to be synthesized is represented by I_g, the input content images are represented by I_c, and the input style images are represented by I_s. Figure 1 is the overall structure of the algorithm. The algorithm belongs to an online image style transfer algorithm based on image optimization. The whole network inputs content image and style image, then outputs stylized target image. The target image is a variable to be optimized. The optimization algorithm adjusts the target image continuously so that the value of the loss function gradually decreases. When the optimization algorithm ends, the target image is the stylized image. When calculating the loss function, firstly, the semantic information of the content image and style image is extracted by semantic extraction network, then the content feature of the content image, style image and content and style feature of the target image are extracted by feature extraction network, and then the target image is matched based on semantic information (the target image is constrained by the semantics of content image) Finally, the loss function is calculated based on the content and style features of the target image, the matched style features and the content features of the content image. The design details of each part will be introduced in detail below.

A. SEMANTIC EXTRACTION NETWORK

We extract semantic information based on a semantic segmentation network. A semantic segmentation network is used to classify input images at the pixel level. In traditional classification networks, an image corresponds to a label, such as a dog or a cat. The output of the semantic segmentation network is a mask image with the same resolution as the input image, and the value of each pixel is the category label corresponding to the input pixel. As shown in Figure 2, the input image is on the left, and the output of the semantic segmentation network is on the right. It can be seen that red pixels correspond to hair, and green pixels correspond to faces. Each pixel has its own category label (i.e., different colors).

FIGURE 2. Input (left) and output (right) of semantic segmentation network.

If the input image is represented by I, the semantic segmentation network is represented by S, and a mask image is represented by m, then the relationship between the three can be expressed as following:

$$m = S(I) \tag{1}$$

where I is the three-dimensional array, m is the three-dimensional array, S generally is denoted by the convolution neural network, there is no explicit representation, W, H, D are the length, width, and channel number of the input image respectively, and K is the number of categories that can be recognized by the semantic network S.

The semantic extraction network is built on the basis of mask R-CNN [18]. Mask R-CNN is an algorithm that integrates detection, classification, and semantic segmentation. Based on fast R-CNN [19], the algorithm adds a branch for predicting semantic mask (mask), which only increases a small amount of computational overhead, but significantly improves the detection performance.

Mask R-CNN has two features that are suitable for style migration, and the semantic extraction network in this paper also retains these two features. First, the RoIAlign layer [18] is introduced to avoid the loss of location information during downsampling to a certain extent, which improves the prediction accuracy of the semantic mask; and the semantic mask determines the process of style matching in the style migration network. Secondly, mask R-CNN decouples the classification and semantic segmentation, and their prediction results are independent of each other. This not only improves the performance of the two tasks but also greatly simplifies the process of style matching based on semantics. Figure 3 shows the structure of the semantic extraction network. It can be seen that the whole network is mainly composed of four subnetworks: FPN (feature pyramid network [20]), RPN (region proposal network [19]), FCN (fully volatile network [21]), and classification regression network.

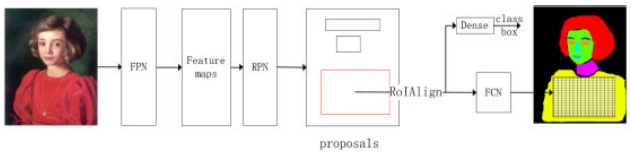


FIGURE 3. Semantic segmentation network takes the source image as an input and outputs a semantic image corresponding to the input image.

B. FEATURE EXTRACTION NETWORK

In order to realize the style transfer, it is necessary to extract the content features and style features of the content image respectively, and input these features into a composite network to obtain a new image which combines the content features and the style features. Similarly, if the input image is represented by I, the content feature extraction network is represented by the function F_c, the style feature extraction network is represented by the function F_s, and the content feature and style feature of the input image are represented by c and s, respectively. The input image I is also a three-dimensional array, and functions F_c and F_s are generally implemented by convolution neural network. It should be noted that content features and style features are generally expressed in the form of array vectors, that is, each element is a three-dimensional array, and these elements are accessed through subscript index. Generally speaking, each element of content feature and style feature corresponds to feature map of different layers of convolutional neural network. This combination of low-level visual information and high-level semantic information can more completely express the information contained in the original image. In addition, it is convenient to define the loss function by using array vector. According to Gatys *et al.*'s method [1], after feature extraction by a deep artificial neural network, the style and content of the image are separable to a certain extent. Generally speaking, the feature map of the high-level network corresponds to the content feature of the image, while the feature map of the low-level network corresponds to the style feature. Based on this study, Gatys *et al.* Successfully synthesized new images with content and style from two images using a multi-level feature map and Gram matrix. Based on Gatys *et al.*'s research, this paper retains the organization of content features, abandons the Gram matrix, and directly uses the network feature map as the style feature. A feature extraction network based on vggnet19 [22] is proposed in this paper.

C. STYLE MATCHING NETWORK

The starting point of this paper is to do style transfer based on semantic information, that is, to transfer the same kind of objects. For example, the style of clothes in graph a is transferred to the clothes of graph B, and the style of hair in graph a is transferred to the hair of graph B so as to migrate one by one. If the traditional coarse-grained style migration algorithm is used, such as the algorithm of Mordvintsev *et al.* [1], it is difficult to achieve style migration at the semantic level. According to the research of Li *et al.* [7], the style matching based on patch can achieve the migration quality similar to that of Mordvintsev *et al.* [1] Therefore, this paper introduces semantic information into image block granularity and makes matching between image blocks based on style features and semantic information, so as to achieve the purpose of style transfer at the semantic level. Style features have a hierarchical structure, including low-level high-resolution features and high-level low-resolution features. To integrate semantic information, it is necessary to downsampling the semantic mask of the original image, as described in the equation 2.

$$m_l = downsampling(m, s_l) \tag{2}$$

where l represents the network layer number, m_l represents the semantic mask of the network layer, and s_l represents the downsampling ratio of m_l. This value is determined by the resolution of the input image and the output resolution of the network layer. We take the stride size of convolution kernel as two and take it as an example to illustrate how to calculate the downsampling ratio.

$$s_l = \frac{m}{2^l} \tag{3}$$

where l represents the network layer number.

Then, the style feature and the semantic feature is spliced together in the feature dimension to form a new style feature, which integrates the traditional style features and semantic information. Both have a hierarchical structure, and the subsequent content will explain the algorithm principle with a certain layer as the representative, that is, the style characteristics s_l corresponding to the network layer and the style features s_l^i corresponding to the network layer integrating

semantic information. In addition, it should be noted that the semantic information m_l of the target image comes from the content image. The process of fusion s and m is described in equation 4. Before splicing, in order to unify the magnitude of $m_l$and s_l and facilitate the subsequent calculation of distance, the two are normalized, respectively. In addition, hyperparameters are introduced to balance the influence of traditional features and semantic information on style. When $\lambda = 0$, only traditional features were used for style migration. When $\lambda = 10000$,only semantic information was used for style migration; users can set different values according to the actual application scenarios.

$$s_l^i = norm(s_l)||\lambda \times norm(m_l) \tag{4}$$

With the complete style features, we can do matching on the image block granularity. The function ψ is used to extract the features of the image block $K \times K$. For the convenience of distinguishing, s_l^X represents the style feature of the target image X and s_l^a represents the style feature of the style image. For the $i-th$ image block of the layer l, the matching image block is calculated by function N, as shown in equation 5. The formula shows that the image block j with the largest cosine similarity is found in the style feature layer l of the style image, and the image block j matches the image block i. in other words, in the style image, the image block i closest to the image block i style is the image block j. Note that the image block i belongs to the target image and the image block j belongs to the input style image.

$$N_l^i = \arg\max_j \frac{\phi_i(s_l^X) \cdot \phi_j(s_l^a)}{|\phi_i(s_l^X)| \cdot |\phi_j(s_l^a)|} \tag{5}$$

In order to realize the matching function N efficiently, the style features of image blocks are expressed in the form of a one-dimensional vector. That is, the output of the function ϕ is a one-dimensional vector. Then the style features of all image blocks in the layer l can form a two-dimensional matrix, which is represented by P. Then the cosine similarity between all image blocks of s_l^X and all image blocks of s_l^a can be calculated. Note that matrix multiplication is used in the formula, which includes the cosine similarity between all image block pairs. After obtaining the cosine similarity between all image block pairs, the image blocks can be quickly matched according to equation 6.

$$N_l^i = \arg\max_j D_l(i,j) \tag{6}$$

where $D_l = P_l^X \times P_l^a$. In this paper, we do not use features to extract the style features extracted from the network directly but use the matching and recombined style features. Inspired by the paper [23], this paper divides the style matching network into two sub-networks: semantic information fusion sub-network (Fig 4) and style matching sub-network (Fig 5).

The semantic information fusion sub-network is responsible for introducing semantic information into the style. That is, the semantic mask is spliced with the feature map of the network layer (Reference). Figure 4 shows the workflow of the network. While the feature extraction network extracts image features, the semantic mask is also downsampling in the same proportion, and pairwise splicing is made to form a new feature $semx_y$ which integrates semantic information. With this feature, you can do style matching. The sub-network of style matching has two inputs and one output, which inputs the style features of the style image

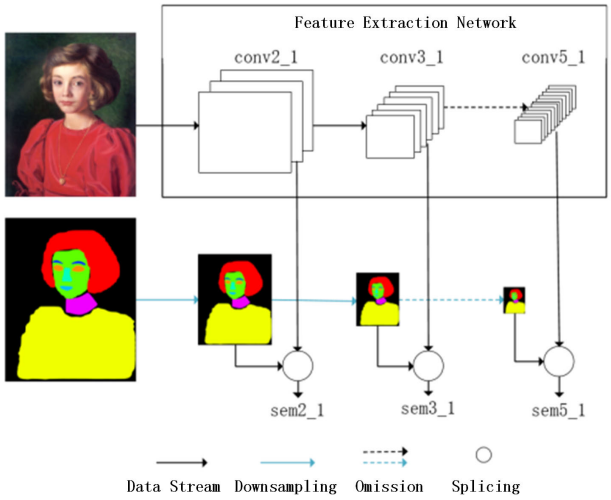


FIGURE 4. Semantic information fusion subnet in order to realize the semantic style transfer.

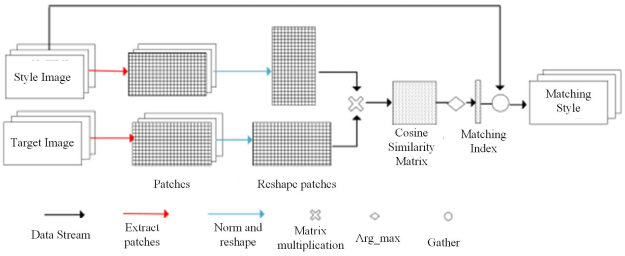


FIGURE 5. *sem*3_1 style matching subnet used to realize style transfer.

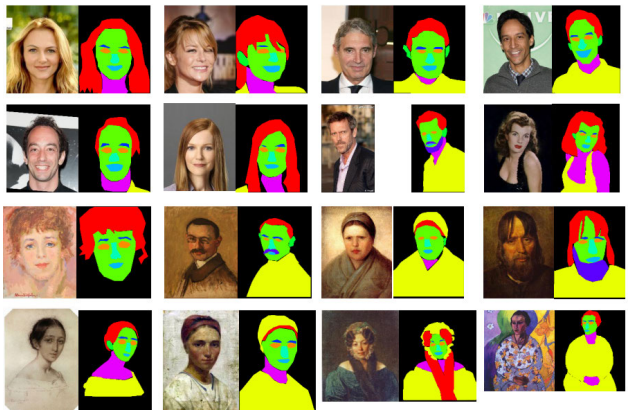


FIGURE 6. Portrait sample dataset (semantic segmentation) used in our approach.

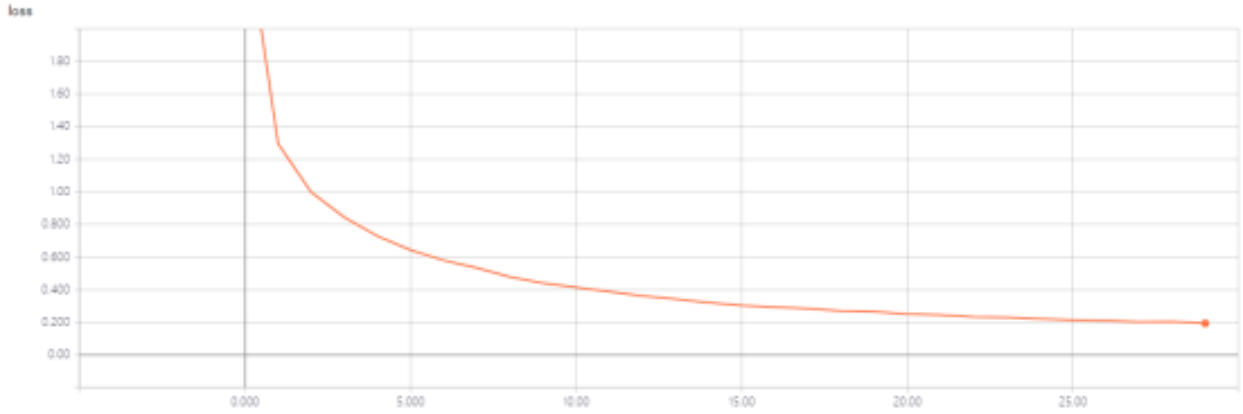


FIGURE 7. Relationship between semantic extraction network training loss and Epoch.

FIGURE 8. Relationship between semantic extraction network validation loss value and Epoch.

and the target image, and outputs the features after matching and recombination. Since the style features come from the feature map of a multi-layer network, only $sem3_1$ is used in Figure 5 as an example to illustrate the workflow of the network. Matching is based on image blocks. The features of the target image and the style image are divided into the same number of image blocks, and the cosine similarity matrix of the two image blocks is calculated. Based on the matrix, the most similar image blocks in the style image are found for each image block of the target image. Finally, these matching image blocks are reconstructed into a complete network layer feature map. The style features of the target image after matching are presented. With the matching style features, the loss function can be calculated, and then the target image can be obtained by minimizing the loss function.

After matching styles in the image block granularity, the stylized image can not be obtained directly, and the stylized target image needs to be generated iteratively based on the image optimization method. This process is driven by the optimization algorithm for image reconstruction loss. According to the research of Gatys *et al.* [3], the loss of image reconstruction in style transfer includes two parts: the loss of content reconstruction and the loss of style reconstruction, and there is a constraint relationship between them so that neither side can be optimized without restriction. Therefore, Gatys *et al.* Expressed the reconstruction loss in the form of formula 7. α and β are the weights of content reconstruction loss and style reconstruction loss, respectively, which are used to balance the influence of both on the overall reconstruction loss. The loss function of this design is still used in this paper.

$$L = \alpha L_c + \beta L_s \tag{7}$$

The loss of content reconstruction follows the design of Gatys *et al.*, as shown in formula 8. Note, c_l^X and c_l^P are the content feature of the target image and the content feature of the content image, respectively.

$$L_c = \sum_l ||c_l^X - c_l^P||^2 \tag{8}$$

L_s uses the Gram matrix, but the Gram matrix loses the location information and is difficult to combine with the

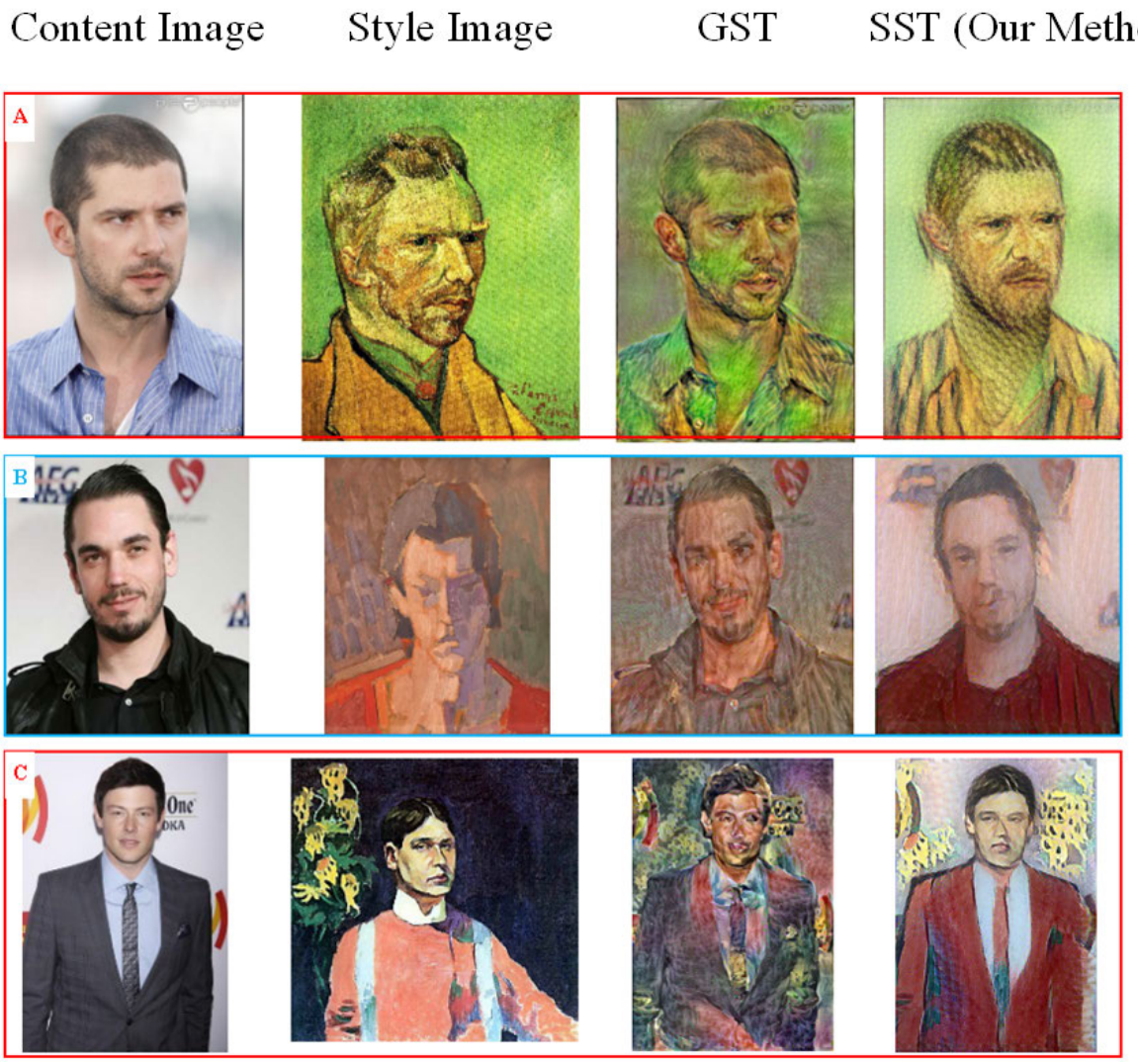


FIGURE 9. Comparison of GST and SST (our approach) results of image style transfer.

semantic information. So we gives up the Gram matrix during constructing L_s in this paper and uses the characteristic graph and semantic mask of convolution layer, such as formula 9. Where l is the number of the network layer, i is the number of the image block, and S_l^X is the number of the image block so that the target image X combines the style features of semantic information (layer l), s_l^a means that the style image integrates the style features of semantic information (layer l), and the function $\phi_i(s_l^X)$ denotes from s_l^X to extract the features of image block $i-th$. It can be seen from the formula that the essence of L_s is the sum of the squares of the Euclidean distance between the style features of the target image and the style features of the style image, but this distance is calculated at the image block level and involves the style matching between image blocks.

$$L_s = \sum_l \sum_i ||\phi_i(s_l^X) - \phi_{N_L(i)}(s_l^a)||^2 \tag{9}$$

IV. EXPERIMENT AND ANALYSIS

A. EXPERIMENTAL ENVIRONMENT

Following the idea of Champanard [1], this experiment focuses on the typical semantic sensitive scene, that is, the style transfer between portraits. Semantic extraction network needs pre-training, but there is no portrait data set for semantic segmentation. Therefore, this paper randomly extracts some portrait images from Celeba [25] and Wikiart [24]] data sets and labels 554 of them with labels [27] (Figure 6 shows some examples). It should be noted that what we do is transfer learning. On the basis of the trained mask R-CNN model, we use the new data to fine-tune the model. Although the coco dataset [26] (which uses mask R-CNN trained on the dataset) does not define the categories in the portrait dataset, the dataset itself contains a large number of images of characters R-CNN has learned the features related to portrait in the training process, so it does not need a lot of annotation images to fine-tune the model.

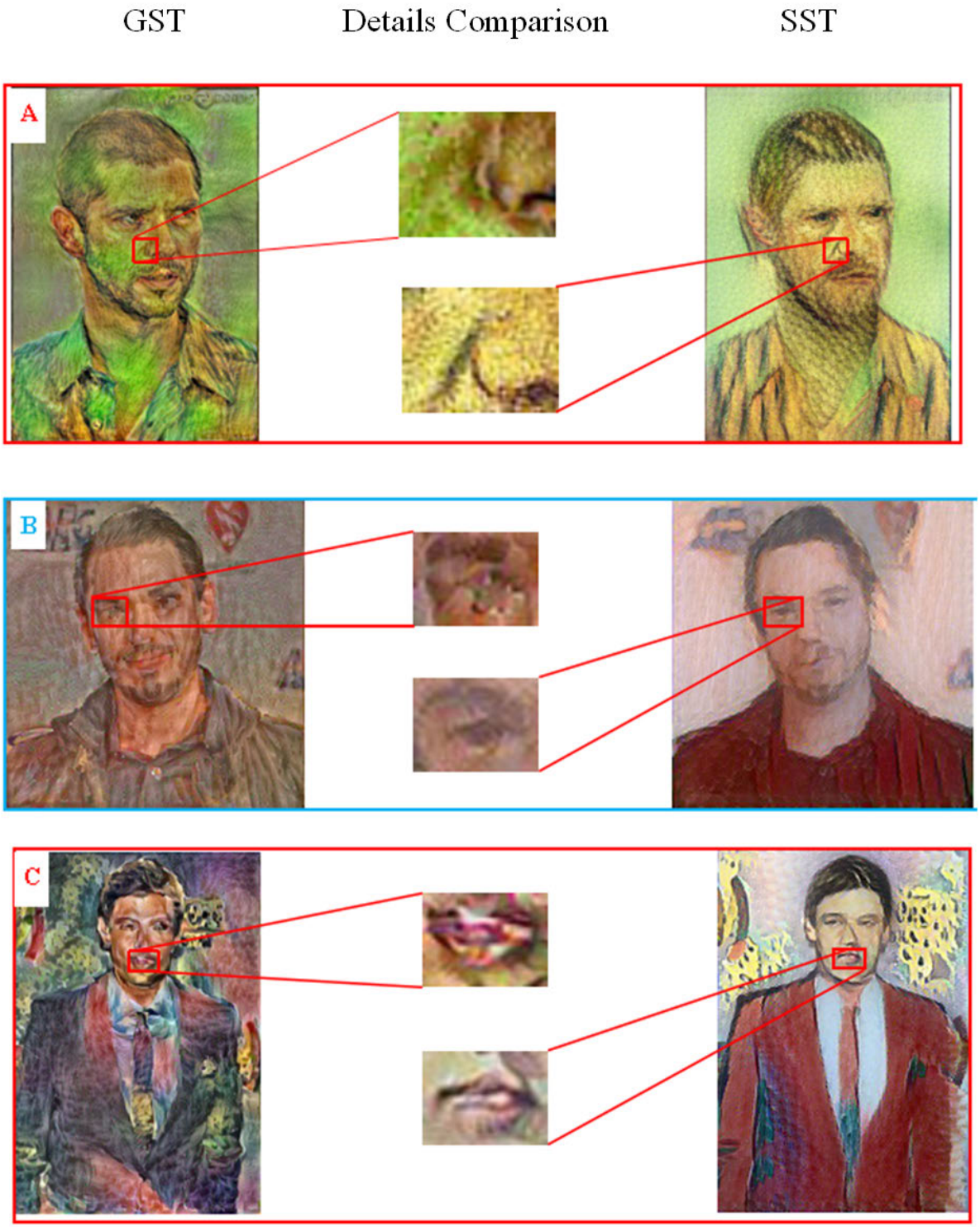


FIGURE 10. Comparison of GST and SST (our approach) details of image style transfer.

B. EXPERIMENTAL DESIGN

In order to verify that adding semantic information can improve the quality of style transfer, this paper designs a comparative experiment with GST [3], extracts images from Wikiart [24] and Celeba [25], synthesizes new images using SST and GST respectively, and then compares the visual effects after migration. Since there is no universally accepted quantitative evaluation index in the field of style transfer, this

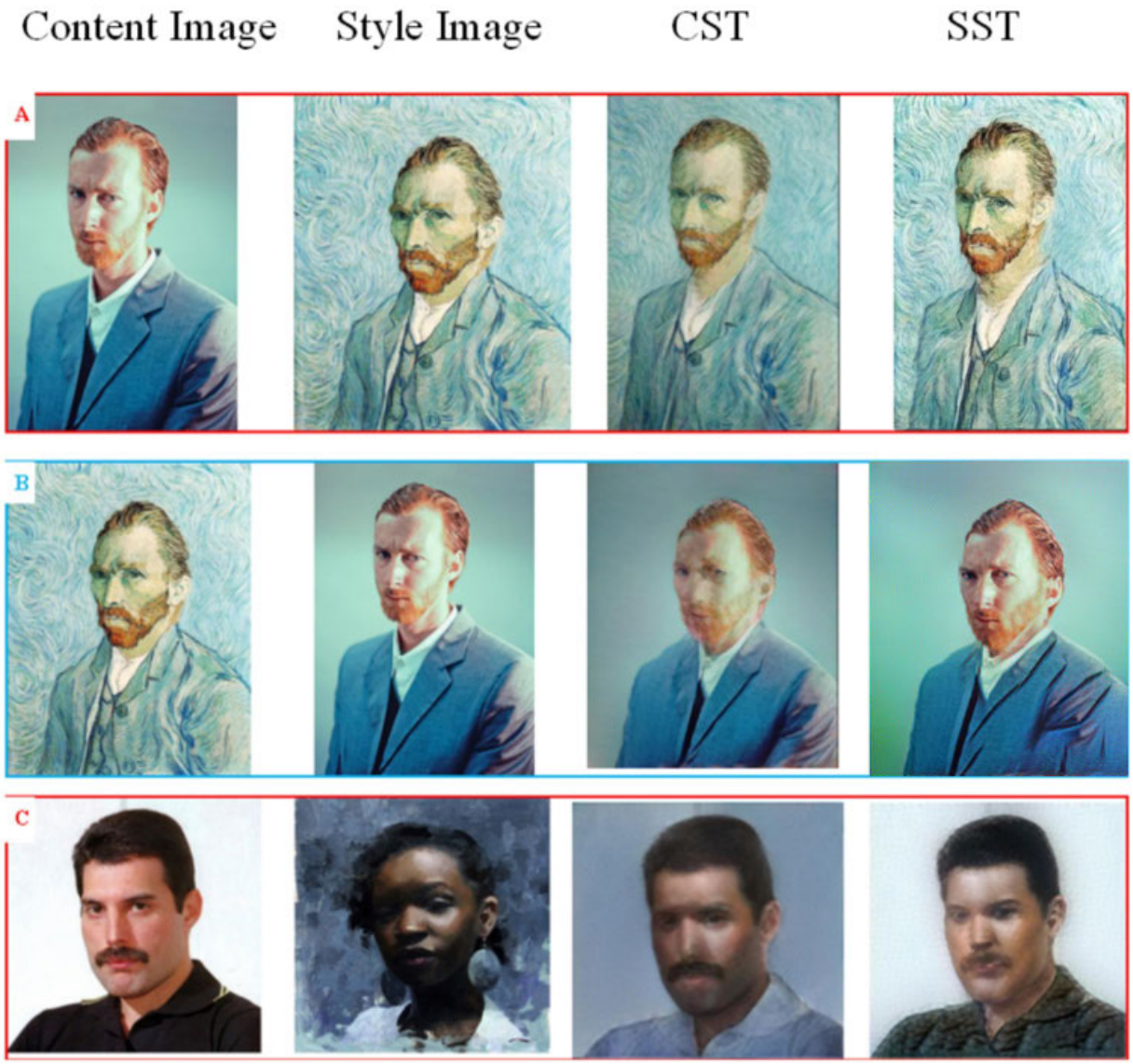


FIGURE 11. Comparison of CST and SST (our approach) results of image style transfer.

paper also uses the Convention to give the composite graph of the two algorithms as an example to evaluate the quality of the transfer. In order to prove that the proposed algorithm has a higher quality of style transfer than the existing algorithms based on semantic information, this paper designs a comparative experiment with CST [1]. As far as we know, CST is the only image style migration algorithm that uses semantic information. In this experiment, we will use this algorithm to recombine the composition graph published in the CST paper to compare the synthesis effect of the two.

C. EXPERIMENTAL RESULTS AND ANALYSIS

Fig 7 and Fig 8 are the change curve of the loss value (loss) relative to the period (epoch) in the process of semantic extraction network training. From the graph, we can find that with the increase of the period, the training loss value gradually decreases, while the verification loss value first decreases and then increases. That is to say, there is overfitting. At about the sixth period, the verification loss value is the minimum, and the value is about 1.55. The model used for semantic segmentation in subsequent experiments is also the model at this time.

Figure 9 and figure 10 compare the synthesis effect of GST and SST (there are three contrast examples of A, B, and C). It can be seen that there are two obvious differences between the two. First, GST does not distinguish the style of the object, which is more obvious on the images of a and C. for example, a migrates the background color (green) of the style image to multiple places of the content image, and C transfers the flowers in the background of the style image to the clothes of the content image. In contrast, the algorithm in this paper does

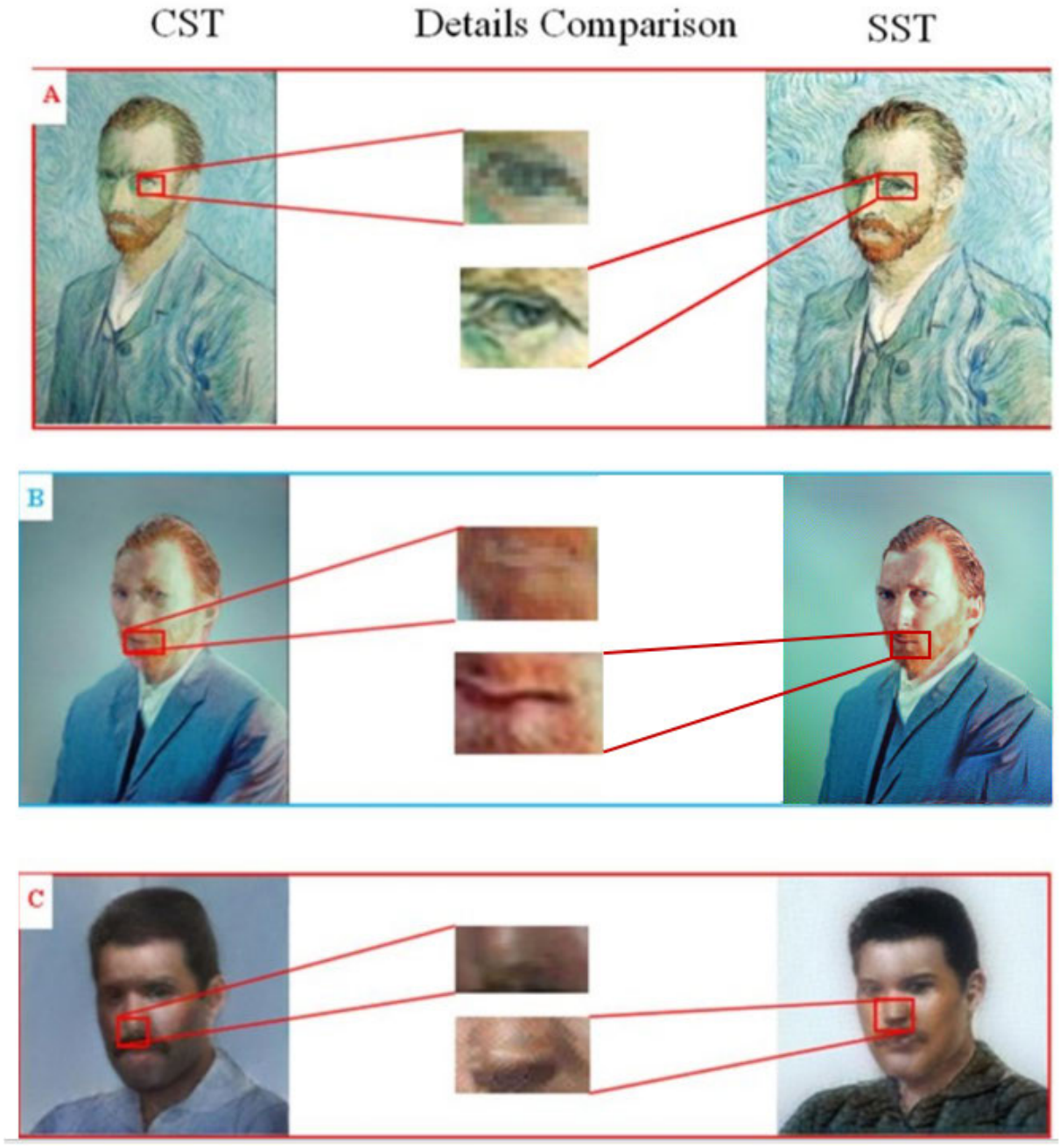


FIGURE 12. Comparison of CST and SST (our approach) details of image style transfer.

not have this problem. When the style is transferred, clothes correspond to clothes (for example, the style map of C) For example, the pink dress corresponds to the dark red dress in the target image), and the background corresponds to the background (for example, the green background of A's style image corresponds to the light green background in the composite image). Secondly, the facial features of the GST generated image are fuzzy, and some of them even distort the facial features, which is more obvious in the images of B and C. For example, the eyes of the GST composite image of B have become lumped (refer to Fig 10 for details), and the mouth of the composite image of C has been distorted, which is quite different from the content image (refer to the figure for details) The algorithm in this paper can deal with the details of facial features very well, and can achieve the coordination of facial features as a whole, and there will be no fuzzy and caking (refer to the details of Fig. 10 for comparison). From comparing the above two points, we can find that the introduction of semantic information can achieve style transfer between similar objects, which can significantly

improve the quality of style transfer, especially in the application scenarios that are sensitive to semantic information.

Figure 11 and figure 12 compare the synthesis effect of CST and SST. From the figure, we can find an obvious advantage of SST. That is, the synthesized image is relatively clear, especially the facial features. The typical features are the eyes of a, the mouth of B, and the nose of C (refer to the figure for details) In the composite image of CST, a's eyes are lack of wrinkle details, B's mouth is basically covered by a beard, C's nose boundary is relatively fuzzy, and there is no prominent three-dimensional feeling. In the composition diagram of SST, these details are reflected (refer to Fig. 12). Therefore, SST is better than CST in the clarity of the composite graph. From Figure 12, we can see that our algorithm has better image transfer quality in the portrait image. Specifically, in the eyes, nose, mouth, and other key feature points, the detailed enlarged images of the results generated by our algorithm are clear to those of the results generated by the CST algorithm. We can also see the style on the clothes that our method learns the style of clothes in the style image, but the style of clothes in the CST method was not transferred. This phenomenon is particularly obvious in Figure 12.B.

Of course, compared with other algorithms, SST also has shortcomings. Compared with GST, the algorithm in this paper is not widely used and can only be used as a supplement of GST in specific scenarios. Compared with CST, some style information is lost due to the introduction of semantic information. For example, in Fig. 11, the composition graph of the algorithm in this paper has fewer buttons, and the composition graph of B has too many wrinkles. Therefore, the algorithm in this paper has some room for improvement, which will also be a direction of my follow-up research. Most of the existing image style transfer algorithms transfer the whole image style as a whole. Style feature is a set of correlation matrix based on style image, namely Gram matrix. Each matrix is a global description of the style image. In other words, the semantic correspondence between content image and style image is not considered in the process of migration. Our method extracts the semantic information of the content image and style image and inputs the semantic information (in the form of a semantic map) into the subsequent style transfer network. During the style transfer, the content with the same semantic will be matched first so as to realize the style transfer of semantic awareness and achieve high-quality image style transfer.

V. CONCLUSION

In order to solve the problem of semantic mismatching in image style migration, an image style migration algorithm based on semantic segmentation is proposed. The algorithm automatically extracts the semantic information of style image and content image through a semantic segmentation network and uses the semantic information to guide the style migration. The experiments on Celeba and Wikiart data sets show that compared with the current classic in this field, the algorithm can automatically extract the semantic information of style image and content image. Our algorithm can effectively solve the problem of semantic mismatch in the process of image style transfer, and it can maintain semantic consistency in the process of style transfer. In the future, we plan to apply our feature extraction method to content-based image retrieval [29]. For the style transfer between non-art images, we plan to use the correspondence between images as a constraint for style transfer.

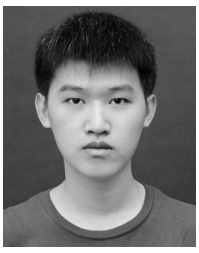
REFERENCES

[1] A. Mordvintsev, C. Olah, and M. Tyka, "Inceptionism: Going deeper into neural networks," Tech. Rep., 2015.
[2] Y. Jing, Y. Yang, Z. Feng, J. Ye, Y. Yu, and M. Song, "Neural style transfer: A review," *IEEE Trans. Vis. Comput. Graphics*, vol. 26, no. 11, pp. 3365–3385, Nov. 2020.
[3] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2414–2423.
[4] Y. Li, N. Wang, J. Liu, and X. Hou, "Demystifying neural style transfer," in *Proc. IJCAI*, Aug. 2017, pp. 2230–2236.
[5] Y. Jing, Y. Liu, Y. Yang, Z. Feng, Y. Yu, D. Tao, and M. Song, "Stroke controllable fast style transfer with adaptive receptive fields," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 238–254.
[6] C. Li and M. Wand, "Combining Markov random fields and convolutional neural networks for image synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2479–2486.
[7] S. Li, X. Xu, L. Nie, and T.-S. Chua, "Laplacian-steered neural style transfer," in *Proc. 25th ACM Int. Conf. Multimedia*, Oct. 2017, pp. 1716–1724.
[8] D. Chen, L. Yuan, J. Liao, N. Yu, and G. Hua, "StyleBank: An explicit representation for neural image style transfer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1897–1906.
[9] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1501–1510.
[10] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 694–711.
[11] L. Zhao, A. Li, S. Lin, W. Xing, and D. Lu, "SpatialGAN: Progressive image generation based on spatial recursive adversarial expansion," in *Proc. ACM Multimedia Conf.*, 2020, pp. 2336–2344.
[12] X. Li, S. Liu, J. Kautz, and M.-H. Yang, "Learning linear transformations for fast arbitrary style transfer," in *Proc. CVPR*, 2019, pp. 1–9.
[13] H. Chen, L. Zhao, L. Qiu, Z. Wang, H. Zhang, W. Xing, and D. Lu, "Creative and diverse artwork generation using adversarial networks," *IET Comput. Vis.*, vol. 14, no. 8, pp. 650–657, Dec. 2020.
[14] L. Zhao, Q. Mo, S. Lin, Z. Wang, Z. Zuo, H. Chen, W. Xing, and D. Lu, "UCTGAN: Diverse image inpainting based on unsupervised cross-space translation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 5741–5750.
[15] N. Kolkin, J. Salavon, and G. Shakhnarovich, "Style transfer by relaxed optimal transport and self-similarity," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 10051–10060.
[16] Z. Wang, L. Zhao, H. Chen, L. Qiu, Q. Mo, S. Lin, W. Xing, and D. Lu, "Diversified arbitrary style transfer via deep feature perturbation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 7789–7798.
[17] F. Luan, S. Paris, E. Shechtman, and K. Bala, "Deep photo style transfer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4990–4998.
[18] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.
[19] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.
[20] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. CVPR*, 2017, vol. 1, no. 2, p. 4.

[21] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.
[22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. ICLR*, 2015, pp. 1–14.
[23] Z. Wang, L. Zhao, S. Lin, Q. Mo, H. Zhang, W. Xing, and D. Lu, "GLStyleNet: Exquisite style transfer combining global and local pyramid features," *IET Comput. Vis.*, vol. 14, no. 8, pp. 575–586, Dec. 2020.
[24] (Dec. 8, 2018). *WikiArt DB/OL*. [Online]. Available: http://wikiart.org/
[25] (Dec. 8, 2018). *Large-Scale CelebFaces Attributes (CelebA) Dataset [DB/OL]*. [Online]. Available: http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html
[26] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 740–755.
[27] (Dec. 8, 2018). *LabelMe [CP/DK]*. [Online]. Available: http://labelme.csail.mit.edu
[28] L. Armi and S. Fekri-Ershad, "Texture image analysis and texture classification methods—A review," 2019, *arXiv:1904.06554*. [Online]. Available: http://arxiv.org/abs/1904.06554
[29] F. Tajeripour, M. Saberi, and S. F. Ershad, "Developing a novel approach for content based image retrieval using modified local binary patterns and morphological transform," *Int. Arab J. Inf. Technol.*, vol. 12, no. 6, pp. 574–581, 2015.

ZHIJIE LIN graduated from the Hangzhou University of Electronic Science and Technology, received the master's and Ph.D. degrees from Zhejiang University. He currently works with the Hangzhou University of Science and Technology, where he is mainly engaged in research and application of deep learning, machine learning, and image processing.

ZHIZHONG WANG is currently pursuing the Ph.D. degree with the College of Computer Science and Technology, Zhejiang University. His research interests include computer vision and deep learning.



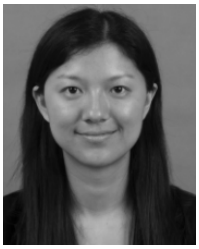
HAIBO CHEN is currently pursuing the Ph.D. degree with the College of Computer Science and Technology, Zhejiang University. His research interests include computer vision and deep learning.

XIAOLONG MA received the Ph.D. degree in management science and engineering from the Shanghai University of Finance and Economics, China, in 2016. He is currently an Associate Professor with the School of Economics and Management, Huzhou University, Zhejiang, China. His interests include image processing, network mechanism design, and data mining.

CHUAN XIE is currently an Associate Professor with the Hangzhou Vocational and Technical College, where he is engaged in deep learning, image processing, especially the research and application of image style migration, image restoration, 3D modeling, and rendering.

WEI XING received the Ph.D. degree from Zhejiang University, in 2009. He is currently an Associate Professor with the College of Computer Science and Technology, Zhejiang University. His research interests include computer vision, image processing, and deep learning.

LEI ZHAO received the Ph.D. degree from Zhejiang University, in 2009. Then he did two years postdoctoral training at the College of Computer Science and Technology of Zhejiang University. He is currently an Assistant Professor with Zhejiang University. His research interests include computer graphics, computer vision, image processing, and deep learning.



WEI SONG graduated from Chongqing University. She currently works with the Zhejiang University of Science and Technology, where she is mainly engaged in research of digital image processing.