Learning Dynamic Style Kernels for Artistic Style Transfer

Wenju Xu
OPPO US Research Center
Palo Alto, CA, USA
xuwenju123@gmail.com

Chengjiang Long
Meta Reality Lab
Burlingame, CA, USA
clong1@meta.com

Yongwei Nie
South China University of Technology
Guangzhou, Guangdong, China
nieyongwei@scut.edu.cn

Abstract

Arbitrary style transfer has been demonstrated to be efficient in artistic image generation. Previous methods either globally modulate the content feature ignoring local details, or overly focus on the local structure details leading to style leakage. In contrast to the literature, we propose a new scheme “style kernel” that learns spatially adaptive kernels for per-pixel stylization, where the convolutional kernels are dynamically generated from the global style-content aligned feature and then the learned kernels are applied to modulate the content feature at each spatial position. This new scheme allows flexible both global and local interactions between the content and style features such that the wanted styles can be easily transferred to the content image while at the same time the content structure can be easily preserved. To further enhance the flexibility of our style transfer method, we propose a Style Alignment Encoding (SAE) module complemented with a Content-based Gating Modulation (CGM) module for learning the dynamic style kernels in focusing regions. Extensive experiments strongly demonstrate that our proposed method outperforms state-of-the-art methods and exhibits superior performance in terms of visual quality and efficiency.

1. Introduction

Artistic style transfer [48] refers to a hot computer vision technology that allows us to recompose the content of an image in the style of an artistic work. Figure 1 shows several vivid examples. We might have ever imagined what a photo might look like if it were painted by a famous artist like Pablo Picasso or Van Gogh. Now style transfer is the computer vision technique that turns this into a reality. It has great potential values in various real-world applications and therefore attracts a lot of researchers to constantly put efforts to make progress towards both quality and efficiency.

Most of existing style transfer works [6, 15, 18, 26, 33] either globally modulate the content feature ignoring local details or overly focus on the local structure details leading to style leakage. In particular, [15, 18] seeks to match global statistics between content and style images, resulting in inconsistent stylizations that either remain large parts of the content unchanged or contain local content with distorted style patterns. SANet [33] and AdaAttN [31] improve the content similarity by learning attention maps that match the semantics between the style and stylized images, but these methods tend to distort object instances when improper style patterns are involved. Recently, IEC [5] adopts contrastive learning to pull close both the content and style representations between input and output images. MAST [16]
seeks to balance the style and content via manifold alignment. Although both IEC and MAST have achieved some success in most cases, they are still far from satisfactory to well balance two important requirements, i.e., style consistency and structure similarity.

In this paper, we develop a novel style transfer framework in a manner of encoder-decoder structure, as illustrated in Figure 2, which contains two important modules: (1) a Style Alignment Encoding (SAE) module enhanced by Content-based Gating Modulation (CGM), which is used to generate content-style aligned features, and (2) a Style Kernel Generation (SKG) module used to transform the output features of SAE to convolutional kernels. In such a framework, we propose a new style transfer scheme, “style kernel”, which treats the features of the style images as dynamic convolutional kernels and then transfer the style information to the content image by convolving the content image by the learned dynamic style kernels. This allows fine-grained local interactions between the style and content features, the flexibility of which makes both style transferring and content structure preserving much easier.

To enforce the global correlation between the content and style features, we simulate the self-attention mechanism of Transformer [41] in the SAE module. This treats the content and style features as the queries and keys of the self-attention operator respectively and computes a large attention map the elements of which measure for each local content context the similarity to each local style context. With the attention map, we can aggregate for each pixel of the content image all the style features in a weighted manner to obtain content-style aligned features. We also design a Content-based Gating Modulation (CGM) operator that further thresholds the attention map and zeros the places where similarities are too small, seeking an adaptive number of correlated neighbors as a focusing region for each query point. Then, we use another set of convolutions that constitute the SKG module to further transform the content-style aligned features, and view (reshape) the output of SKG as dynamic style kernels. In order to improve the efficiency of the dynamic convolutions, SKG predicts separable local filters (two 1D filters and a bias) instead of a spatial 2D filter that has more parameters. The learned style kernels already integrate the information of the given content and style images. We further apply the learned dynamic style kernels to the content image feature for transferring the target style to the content image.

We shall emphasize that unlike “style code” in AdaIN [15], DRB-GAN [48], and AdaConv [2] modeled as the dynamic parameters (e.g. mean, variance, convolution kernel) which are shared over all spatial positions globally without distinguishing the semantic regions, the learned dynamic kernels via our “style kernel” are point-wisely modeled upon the globally style-content aligned feature, and therefore is able to make full use of the point-wise semantic structure correlation to modulate content feature for better artistic style transfer. Our learned dynamic style kernels are good at transferring the globally aggregated and locally semantic-aligned style features to local regions. This ensures our style transfer model that works well on consistent stylization with well preserved structure, overcoming the shortages of existing style transfer methods which either globally modulate the content feature ignoring local details (e.g. AdaIN, WCT [28]) or overly focus on the local structure details leading to style leakage (e.g. AdaAttn [31] and MAST [16]).

The main contributions of this work are 3-fold as follows:

• We are the first to propose the “style kernel” scheme for artistic style transfer, which converts the globally style-content alignment features into position point-wisely dynamic convolutional kernels to convolve the features of content images together for style transfer.

• We design a novel architecture, i.e. SAE and CGM, to generate the dynamic style kernels by employing the correlation between the content and style image adaptively.

• Extensive experiments demonstrate that our method produces visually plausible stylization results with fine-grained style details and coherence content with the input content images.

2. Related Works

Artistic Style Transfer was initially solved by Gatys et al. [12] a pre-trained neural network to synthesize stylizations in an iterative optimization manner, and then a bunch of neural style transfer methods [3,5,11,18,31,38,39] were developed. In particular, AdaIN [15] introduces the adaptive instance normalization to globally match the statistics between the content and style features. For the same purpose, WCT [28] utilizes the whitening and coloring transforms, and LST [27] introduces a linear transformation matrix predicted on content and style features. [7,22,42,51] seek different ways to match the feature distribution. SANet [33] seeks to align the semantics between the content and style features via a self-attention network. StrTr2 [8] resorts to the transformers [41,47]. To further preserve the content details, AdaAttn [31] perform attentive normalization on per-point basis, IEC [5] employs stylization relations with contrastive learning and MAST [16] introduces manifold alignment to balance the style and content. However, emphasizing on content details prevents to transfer adequate style information to some degree, and results in style leakage. Recently, dynamic network based style transfer methods [3,15,18,38] extend the model generalization. As a generic extension of AdaIN, AdaConv [2] introduces a parameter network that predicts network parameters used by
another network for style transfer. DRB-GAN [48] models “style code” as the global mean and variance for dynamic convolutions to conduct style transfer. Different from all the above methods, we propose a new scheme “style kernel” to learn dynamic convolutional kernels from the style-content alignment feature to point-wisely modulate the content feature for artistic style transfer.

**Self-Attention Mechanism** is introduced in Transformer [41] as a new attention mechanism. Similar to non-local neural networks [13, 21, 41, 45], Transformer directly works on sequences of image patches to aggregate information for long-range dependencies. The recent researches demonstrate it has been successfully applied in various vision tasks like image recognition [4, 10, 14, 29], object detection [1, 44, 46], image captioning [9], image enhancement [49, 50], and text conditioned image generation [35, 37]. Our model takes the self-attention mechanism to align the semantics of the style feature to that of the content image.

**Dynamic Network** consists of convolution layers for which the filter parameters are predicted by another sub-network. Recently, there is rapid progress in vision applications [17, 32, 36, 48, 52] that benefits from dynamic network for image/video enhancement. [19] employs dynamic upsampling filters for high-resolution image reconstruction. [52] predicts filters from spatio-temporal features for video de-blurring. Differently, our StyleKernel learns spatially adaptive kernel for per-pixel stylization.

3. Our Method

While existing dynamic kernel based methods [2, 15, 48] can efficiently generate stylizations, we observe that their visual-quality is limited in two aspects. (1) As the dynamic kernels are predicted based on the style feature without distinguishing the content-style correlation, the transferred style only globally reflects the style characters of the style image, ruining local content details. (2) The completeness of semantic regions is not well preserved. This is because the dynamic kernels used by the layers are spatially shared. To alleviate these problems, we propose to align the semantics of the style image to those of the content image and transform the content features for per-pixel stylization in a separable convolution manner. The pipeline of our proposed model is shown in Figure 2. The content image \( I_c \) and style image \( I_s \) are first fed into a fixed VGG encoder to produce the content and style feature maps \( Z_c \in \mathbb{R}^{H \times W_c \times C_{in}} \) and \( Z_s \in \mathbb{R}^{H \times W_s \times C_{in}} \), where \( H, W \) and \( C_{in} \) represent the height, width and channel size of the feature map, respectively. Then in SAE, we align the content and style features by using the content features \( Z_c \) as the queries and the style features \( Z_s \) as the keys in a self-attention mechanism. The CGM module is conducted on the attention map to seek a focusing region for each query point. Thereafter, we predict by SKG the dynamic style kernels \( F \) from the content-style aligned features \( Z_{cs} \). Finally, we group and shuffle (GS) \( Z_c \) to break down the correlation within the channel-wise feature and apply convolutions parameterized by the learned kernels to the grouped and shuffled \( Z_c \) to obtain \( Z_{cs} \) which is further decoded to the target stylized image by a decoder.

### 3.1. Style Alignment Encoding

This module takes the content and style features as input, aligns between them by learning content-style attention, and finally attentively adjusts the attention map by...
Content & Style Attribution

**Content-Style Alignment Attention.** By taking the content feature $Z_c$ as the query, and the style feature $Z_s$ as the key and value, we compute the content-style alignment attention that seeks to aggregate information from the style feature map according to the matched content information. This warps the style feature to align with the content feature. To this end, we calculate the pairwise feature correlation $A^{s\rightarrow c} \in \mathbb{R}^{H_s \times W_s \times H_c \times W_c}$. This results in an attention map $A \in \mathbb{R}^{H_s \times W_s \times H_c \times W_c}$ by:

$$A(u, v) = \text{softmax}(\alpha M^{s\rightarrow c}) = \text{softmax}(\alpha \frac{Z_c(u)^T Z_s(v)}{||Z_c(u)|| ||Z_s(v)||}),$$  

where $\alpha$ is the coefficient that controls the sharpness of the softmax. $Z_c(u)$ and $Z_s(v)$ stands for the feature of $Z_c$ and $Z_s$ at position $u \in \mathbb{R}^{H_s \times W_s}$ and $v \in \mathbb{R}^{H_c \times W_c}$. The $u$-th row of the attention map $A$ represents similarities between the $u$-th node of the content feature and all the nodes of the style feature.

To align the semantics of the style feature to that of the content feature, we aggregate information in $Z_s$ based on the attention score map $A$ and obtain $Z_{cs} \in \mathbb{R}^{H_c \times W_c \times C}$ by calculating their weighted average:

$$Z_{cs}(u) = \sum_v A(u, v) \cdot Z_s(v).$$  

where $Z_{cs}(u)$ represents the feature at position $u$ of $Z_{cs}$. In the following, we denote the $Z_{cs} \in \mathbb{R}^{H_c \times W_c \times C}$, which is obtained by reshaping.

In Eq. 2, all style features are taken into consideration when computing the content-style aligned features $Z_{cs}$. We find that this yields inferior results because many irrelevant features are involved into the computation. We therefore introduce the Content-based Gating Modulation (CGM) operator which creates dynamic thresholds that further filter out irrelevant style features.

**Content-based Gating Modulation** is used to further adapt the attention map by the content feature $Z_c$. Taking $Z_c$ as input, we employ a convolution network $E_f$ to generate a scale parameter $\lambda$ and a bias parameter $\beta$. We then perform a row average operator on the attention matrix $A$ by:

$$A_r(i) = \frac{1}{H_c} \sum_{j=1}^{H_c} A_{i,j}, \forall i \in [1, W_c].$$

where $A_r$ is the obtained column vector after the average operator. Then, we obtain the threshold matrix $T$ by the following equation:

$$T = A(\lambda A_r + \beta).$$

Finally, we update the attention $A$ by:

$$\bar{A} = A \cdot \text{Sign}(A - T),$$

where the dot indicates element-wise multiplication and $\text{Sign}$ is a function returning 1 given positive input and 0 otherwise. We use $\bar{A}$ instead of $A$ in Eq. 2 to compute $Z_{cs}$.

$\bar{A}$ is a dynamically learned mask, with which our model creates focused areas and filter out irrelevant style features. This allows our model to aggregate features from fairly correlated features, which is essential to preserve the completeness of semantic regions. In Figure 3 we compare the intermediate visualizations and stylized results. We see that the CGM works to recognize focused areas, where the style information is aggregated by the query point. By contrast, the attention learned by the SANet is distributed over the entire image, including visually less correlated points, such as corners and background.

**3.2. Style Kernel Generation**

We propose the Style Kernel (SK) Generation network to further transform the content-style aligned features of SAE to dynamic style kernels. It takes the globally content-style
aligned feature $Z_{cs}$ as input to predict the dynamic style filters $F, B = \{F_1^{ij}, F_2^{ij}, B^{ij}\}$.

Given the content-style aligned feature $Z_{cs}$, we take two convolution layers to predict the filters $F = \{F_1^{ij}\}$ and $F_2 = \{F_2^{ij}\}$ of the sizes $k \times 1$ and $1 \times k$, respectively, and one bias vector $B = \{B^{ij}\} \in \mathbb{R}^{H_c \times W_c \times C \times 1}$. Finally, we conduct convolution operations on the content feature $Z_c$ with the learned dynamic style kernels $F_1, F_2$ and $B$. The output after the convolutions is the feature called $Z_{cs}$:

$$Z_{cs} = \frac{Z_c - \mu}{\sigma} \ast F_1 \ast F_2 + B,$$

where $\mu$ and $\sigma$ are the affine parameters in normalization layer and $\ast$ is the convolution operator. $Z_{cs}$ is then passed to the decoder to generate the stylized image $I_{cs}$.

**Remarks.** Note that different from “style code” [48] modeled as the parameters (e.g. mean and variance) which are shared over all spatial positions globally without taking different semantic regions into consideration, our learned dynamic style kernels are position point-wisely inferred from the globally style-content aligned feature, and therefore are able to fully explore the point-wise semantic structure correlation to modulate local content feature for better artistic style transfer. Our dynamic style kernels make it possible to transfer both the globally aggregated content-style aligned features to local regions in the content images. This ensures our new scheme “style kernel” works well on consistent stylization with well preserved structure, and alleviates the shortages of existing style transfer methods [15, 16, 28, 31] which either globally modulate the content feature ignoring local details or overly focus on the local structure details leading to style leakage.

### 3.3. Loss Functions

The loss function is formulated with an adversarial loss $L_{adv}$, a reconstruction loss $L_{rec}$, a REMD loss $L_{REMD}$, a style loss $L_{sty}$ [20] and a content loss $L_{cont}$ [33] as:

$$L = L_{adv} + \lambda_{rec} L_{rec} + \lambda_{cont} L_{cont} + \lambda_{sty} L_{sty} + \lambda_{REMD} L_{REMD},$$

where $\lambda_{adv}$, $\lambda_{rec}$, $\lambda_{cont}$, $\lambda_{sty}$ and $\lambda_{REMD}$ are the hyperparameters used to balance the loss during training.

**Reconstruction Loss** $L_{rec}$ is defined as:

$$L_{rec} = \lambda_{rec1} (||I_{cc} - I_c||_2 + ||I_{ss} - I_s||_2) + \lambda_{rec2} \sum_{l=1}^{L} (||\phi_l(I_{cc}) - \phi_l(I_c)||_2 + ||\phi_l(I_{ss}) - \phi_l(I_s)||_2),$$

where $I_{cc}$ and $I_{ss}$ are the reconstructed images when both the input content and style images are $I_c$ and $I_s$, respectively. $\phi_l$ refers to the $l_{th}$ layer in VGG. We chose features from Relu2_1, Relu3_1, Relu4_1 and Relu5_1 layers.

**REMD Loss** $L_{REMD}$ [25] adapts the relaxed earth mover distance (REMD) to align the manifold surface of style features. It is formulated as:

$$L_{REMD} = \max \left( \frac{1}{W_{cc} H_{cc}} \sum_{i} \min_{j} C_{ij}, \frac{1}{W_{ss} H_{ss}} \sum_{i} \min_{j} C_{ij} \right),$$

where $C_{ij}$ denotes the pair-wise cosine distance matrix between the $i_{th}$ and $j_{th}$ feature vector in $Z_{cs}$ and $Z_s$.

### 4. Experiments

We train our proposed method on content images from the MS-COCO [30] dataset, and style images from WikiArt [23] database. Each dataset has around 80k images. The training images are first resized to 512, and then $256 \times 256$ patches are randomly cropped from the images as inputs. Note that our model can be applied to images of any resolution at the testing stage. Our model is implemented by PyTorch [34]. The Adam [24] is adopted as the optimization solver. We train our model for 160k iterations with a batch size of 16. The learning rate is set to 0.0001.

The hyperparameters in loss functions are set to $\lambda_{cont} = 1$, $\lambda_{rec} = 1$, $\lambda_{sty} = 1$, $\lambda_{rec1} = 20$, $\lambda_{rec2} = 0.5$, $\lambda_{REMD} = 3$, and $\lambda_{adv} = 1$. The encoder is a pretrained VGG-16 network [40], whose parameters are fixed during model training. A multi-scale discriminator is adopted from [43]. We update the discriminator one time after two generator iterations.

### 4.1. Comparison with State-of-the-Art Methods

**Qualitative Evaluation.** As shown in Figure 4, we present qualitative results of different style transfer methods *i.e.*, AdaConv [2], DRB-GAN [48], statistics matching based method like AdaIN [15], attention based methods including
Table 2. Quantitative comparison with state-of-the-art methods. We compute the average style loss and the LPIPS score of different methods to indicate how well the style and content are transferred.

<table>
<thead>
<tr>
<th>Method</th>
<th>AdaIN’17</th>
<th>LST’19</th>
<th>AvatarNet’18</th>
<th>SANet’19</th>
<th>IEC’21</th>
<th>DRB-GAN’21</th>
<th>AdaAttN’21</th>
<th>MAST’21</th>
<th>AdaConv’21</th>
<th>StrTr²’22</th>
<th>Ours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sty Loss</td>
<td>1.77</td>
<td>2.67</td>
<td>5.91</td>
<td>1.41</td>
<td>1.81</td>
<td>1.51</td>
<td>1.14</td>
<td>0</td>
<td>-</td>
<td>1.04</td>
<td>1.07</td>
</tr>
<tr>
<td>LPIPS</td>
<td>0.37</td>
<td>0.33</td>
<td>0.34</td>
<td>0.36</td>
<td>0.31</td>
<td>0.33</td>
<td>0.32</td>
<td>0.32</td>
<td>0.42</td>
<td>0.33</td>
<td>0.30</td>
</tr>
</tbody>
</table>

AdaAttN [31] and IEC [5], and feature modification methods like StrTr² [8] and MAST [16]. For a fair comparison, we deploy all competing methods for style transfer on images of the smaller dimension resized to 512 while the aspect ratio is preserved. Note that we use the public released code and follow the default configurations for testing.

The comparison in Figure 4 shows the outperformance of our method in terms of visual quality. These images produced by our method faithfully reflect the style characters (e.g., stroke sizes and colors) with no artifact in the regions, and most importantly, they preserve the structural similarity of the content images. On the contrary, AdaIN [15] and MAST [16] fail to generate sharp details and fine strokes. AdaConv [2] yields distorted patterns as it cannot always recover the original style patterns. We also observe non-negligible artificial structures in those images obtained by StrTr² [8], DRB-GAN [48] and AdaIN [15]. These methods struggle to preserve the consistency in semantic regions such as the sky. In addition, AdaAttN [31] and IEC [5] and MAST [16] tend to overly preserve the structural similarity and fail to transfer the style to the content image, which makes the results look like the content images.

Quantitative Evaluation. (1) Style Loss. Following WCT [28], we adopt the style loss to measure the style consistency between the generated stylizations and the style references. The results of different methods are reported in Table 2. As we can observe, our method achieves the lowest style loss, which indicates that our method mostly transfers the style information to the output. (2) LPIPS. We conduct LPIPS to measure the stability and consistency of rendered video clips by following IEC [5]. This metric is to compute the average perceptual distances between consecutive frames and the lower LPIPS score indicates a better stable and consistent performance. We synthesize 10 video clips
Figure 5. Qualitative performance comparison on stylized results of different model variants. The results of SANet are also listed as reference. Please zoom in to observe the detailed difference.

Figure 6. Effectiveness of the Content-based Gating Module (CGM). The results of AdaAttN are also listed as reference.

Figure 7. Impact of grouped shuffling: we randomly take 100 content-style pairs to create stylizations. From their feature maps extracted by VGG network, we measure the mean activation value channel-wisely and report the overall distribution.

for each method. In Table 2 we observe that our approach produces the lowest LPIPS score among all methods.

**Efficiency Analysis.** In Table 1, we compare the inference time of different methods on image resolutions of 256 × 256 and 512 × 512. All experiments are conducted using a single Tesla V100 GPU. Our method can achieve 25 fps on 512×512 images, which is comparable with SOTA methods such as AdaAttN and IEC. It worth pointing out that the operation of our style kernel takes FLOPs cost of $H_c \times W_c \times C_{in} \times (k+k+1)$ while vanilla convolution operation has FLOPS cost of $H_c \times W_c \times C_{out} \times (C_{in} \times k \times k+1)$. In Table 3, we compare the inference time given different kernel sizes. Note we set k=3 as a trade-off between the efficiency and performance.

Table 3. Results of different sizes of style kernel.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Sty Loss↓</th>
<th>LPIPS↓</th>
<th>$T_{256}$ ↓</th>
<th>$T_{512}$ ↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>k=1</td>
<td>1.04</td>
<td>0.30</td>
<td>0.015</td>
<td>0.034</td>
</tr>
<tr>
<td>k=3</td>
<td>0.98</td>
<td>0.30</td>
<td>0.017</td>
<td>0.035</td>
</tr>
<tr>
<td>k=5</td>
<td>0.97</td>
<td>0.31</td>
<td>0.021</td>
<td>0.044</td>
</tr>
</tbody>
</table>

4.2. Ablation Study

We conduct ablation studies to highlight the effectiveness of different modeling components used by our method.

**Effectiveness of Style Kernel.** This module works to fully explore the point-wise semantic structure correlation to modulate local content features for better artistic style transfer. As shown in Figure 5, our dynamic style kernels make it possible to transfer both the globally aggregated content-style aligned features to local regions in the content images. This ensures our new scheme “style kernel” works well on consistent stylization with well preserved structure.

**Effectiveness of Grouped Shuffling.** To verify this module, we report the statistic distribution of activation strength in Table 7. The distribution gap between the results and real artworks is significantly reduced via using grouped shuffling. Without using GS module, the model fails to produce stylizations consistent to the style reference as shown in Figure 5. This demonstrates that the GS shrinks the statistic gap between the generated images and the style images.

**Effectiveness of CGM.** We summarized the results of our model w/ and w/o CGM in Figure 6. The model w/o CGM produces images containing regions with corrupted patterns and the stylization is not consistent to the style reference. While our model w/ CGM faithfully preserves the completeness of semantic regions. This is because our CGM aims to select focusing regions, where query points can aggregate style information from fairly correlated nodes.

4.3. More Discussions

**Generalization of CGM.** Our content-based gating modulation can be flexibly deployed as a plug-in module. To ver-
ify the generalization, we conduct experiments by replacing the attention module in SANet and IEC with our CGM module. The refactored models are denoted as SANet + CGM and IEC + CGM. We train these models with the default settings and compare the performances in Figure 8. We can observe there are nonneglectable artifacts and distorted regions in the results of original SANet and IEC. By plugging-in our CGM, the artifacts are removed and the semantic regions are well preserved. This further proves the effectiveness of our CGM in aggregating semantically consistent information.

Real-world Video Style Transfer. To verify the performance on real-world video stylization, we collect a 1080P video clip consist of 1377 images and compare our method with DRB-GAN [48]. As shown in Figure 9, our approach outperforms existing style transfer methods in terms of style consistency and stability. This can be attributed to the fact that our style kernels are good at transferring the globally aggregated and locally semantic-aligned style features to local regions.

Social Impact and Future Work. Our model achieves impressive image stylizations with well preserved content structure and consistent style characteristics, as shown in Figure 10. This can definitely benefit our society. In the future, we will incorporate text information for text conditioned image stylization.

5. Conclusion

In this paper, we present a new scheme “style kernel” for artistic style transfer. Our method learns an attention map with focusing regions using the proposed content-based gating operator. The style feature is then aligned to match the semantics in the content feature based on the learned attention map. In the style kernel generation module, the dynamic parameters “style kernel” are learned from the content-style aligned feature and then applied to modulate the content feature for style transfer. Extensive experimental results demonstrate the remarkable performance of our model in generating synthetic style images with better quality than the state-of-the-art.

Acknowledgement

This research was sponsored by Prof. Yongwei Nie’s Natural Science Foundation of China (62072191).
References


[31] Songhua Liu, Tianwei Lin, Dongliang He, Fu Li, Meiling Wang, Xin Li, Zhengxing Sun, Qian Li, and Errui Ding. Adaattn: Revisit attention mechanism in arbitrary neural style transfer. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 6649–6658, 2021. 1, 2, 5, 6


