

Arbitrary Style Transfer via Multi-Adaptation Network

Yingying Deng
School of Artificial Intelligence,
UCAS & NLPR, Institute of
Automation, CAS
dengyingying2017@ia.ac.cn

Fan Tang*
NLPR, Institute of Automation, CAS
tfan.108@gmail.com

Weiming Dong*
NLPR, Institute of Automation, CAS
& CASIA-LLVision Joint Lab
weiming.dong@ia.ac.cn

Wen Sun
Institute of Automation, CAS &
School of Artificial Intelligence, UCAS
sunwen2017@ia.ac.cn

Feiyue Huang
Youtu Lab, Tencent
garyhuang@tencent.com

Changsheng Xu
NLPR, Institute of Automation, CAS
& CASIA-LLVision Joint Lab
csxu@nlpr.ia.ac.cn

ABSTRACT

Arbitrary style transfer is a significant topic with research value and application prospect. A desired style transfer, given a content image and referenced style painting, would render the content image with the color tone and vivid stroke patterns of the style painting while synchronously maintaining the detailed content structure information. Style transfer approaches would initially learn content and style representations of the content and style references and then generate the stylized images guided by these representations. In this paper, we propose the multi-adaptation network which involves two self-adaptation (SA) modules and one co-adaptation (CA) module: the SA modules adaptively disentangle the content and style representations, i.e., content SA module uses position-wise self-attention to enhance content representation and style SA module uses channel-wise self-attention to enhance style representation; the CA module rearranges the distribution of style representation based on content representation distribution by calculating the local similarity between the disentangled content and style features in a non-local fashion. Moreover, a new disentanglement loss function enables our network to extract main style patterns and exact content structures to adapt to various input images, respectively. Various qualitative and quantitative experiments demonstrate that the proposed multi-adaptation network leads to better results than the state-of-the-art style transfer methods.

KEYWORDS

Arbitrary style transfer; Feature disentanglement; Adaptation

ACM Reference Format:

Yingying Deng, Fan Tang, Weiming Dong, Wen Sun, Feiyue Huang, and Changsheng Xu. 2020. Arbitrary Style Transfer via Multi-Adaptation Network. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20)*, October 12–16, 2020, Seattle, WA, USA.

*Co-corresponding authors

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](https://permissions.acm.org).

MM '20, October 12–16, 2020, Seattle, WA, USA

© 2020 Association for Computing Machinery.

ACM ISBN 978-1-4503-7988-5/20/10...\$15.00

<https://doi.org/10.1145/3394171.3414015>

'20), October 12–16, 2020, Seattle, WA, USA. ACM, New York, NY, USA, 9 pages.
<https://doi.org/10.1145/3394171.3414015>

1 INTRODUCTION

Artistic style transfer is a significant technique that focuses on rendering natural images with artistic style patterns and maintaining the content structure of natural images at the same time. In recent years, researchers have applied convolutional neural networks (CNNs) to perform image translation and stylization [3, 31]. Gatys et al. [3] innovatively used deep features extracted from VGG16 to represent image content structure and calculate the correlation of activation maps to obtain image style patterns. However, this optimization-based method is also time-consuming. Based on [3], many works either speed up the transfer procedure or promote the generation quality [3, 10, 12, 22, 24, 27, 30]. Johnson et al. [10] used feed-forward neural networks to achieve the purpose of real-time style rendering. Gatys et al. [4] improved the basic model of [3] to obtain higher-quality results and broaden the applications.

To expand the application of style transfer further, many works focus on arbitrary style transfer methods [6, 9, 16, 17, 20, 23, 25, 28]. AdaIN [8] and WCT [17] align the second-order statistics of style image to content image. However, the holistic style transfer process makes the generated quality disappointing. Patch-swap based methods [1, 28] aim to transfer style image patches to content image according to the similarity between patches pairs. However, when the distributions of content and style structure vary greatly, few style patterns are transferred to the content image through style-swap [1]. Yao et al. [28] improved the style-swap [1] method by adding multi-stroke control and self-attention mechanism. Inspired by self-attention mechanism, Park et al. [20] proposed style-attention to match the style features onto the content features, but may encounter semantic structures of content image distorting problem. Moreover, most style transfer methods use a common encoder to extract features of content and style images, which neglect the domain-specific features contributing to improved generation.

In recent years, some researchers [5, 7, 11, 13, 14, 19, 26, 29, 31, 32] have used Generative Adversarial Networks (GANs) for high-quality image-to-image translation. GAN-based methods can generate high-quality artistic works that can be considered as real. The style and content representation are essential for the translation model. Numerous works [5, 13, 29, 32] focused on the disentanglement of style and content for allowing models to be aware

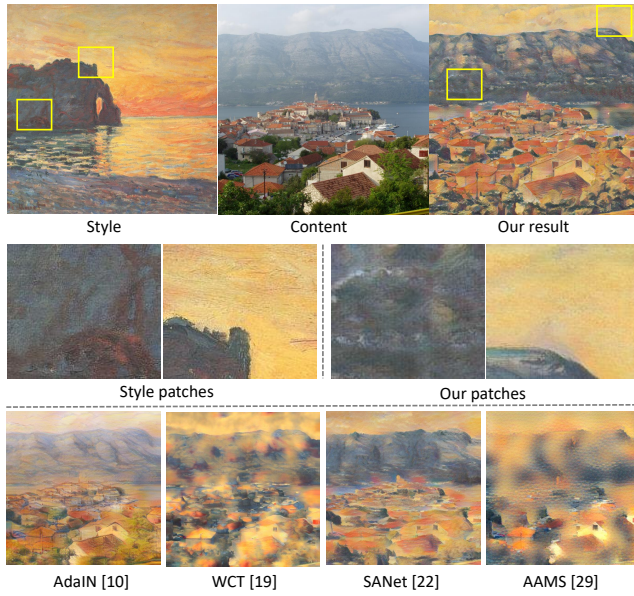


Figure 1: Stylized result using Claude Monet's painting as style reference. Compared with some state-of-the-art algorithms, our result can preserve detailed content structures and maintain vivid style patterns. For example, the mountain and house structures in our result are well preserved. We can observe that the painting strokes and color theme are well transferred by zooming in on similar patches in the style reference and our result.

of isolated factors of deep features. However, the image-to-image translation is difficult to adapt to arbitrary style transfer because of its limitation in unseen domain although it can achieve multi-modal style-guided translation results.

To enhance the generation effect of arbitrary style transfer methods aforementioned, we propose a flexible and efficient arbitrary style transfer model with disentanglement mechanism for preserving the detailed structures of the content images, while transferring rich style patterns of referenced paintings to generated results. As shown in Figure 1, state-of-the-art methods can render the referenced color tone and style into the content image. However, the content structures (outline of the houses) and stroke patterns are not well preserved and transferred. In this work, we propose multi-adaptation network that involves two *Self-Adaptation* (SA) modules and one *Co-Adaptation* (CA) module. The SA module uses the position-wise self-attention to enhance content representation and channel-wise self-attention to enhance style representation, which adaptively disentangle the content and style representation. Meanwhile, the CA module adjusts the style distribution to adapt to content distribution. Our model can learn effective content and style features through the interaction between SA and CA and rearrange the style features based on content features. Then, we merge the rearranged style and content features to obtain the generated results. Our method considers the global information in content and style image and the local similarity between image patches through the SA and CA procedure. Moreover, we introduce a novel disentanglement loss for style and content representation. The content disentanglement loss makes the content features extracted from

stylized results similar when generating a series of stylized results by using a common content image and different style images. The style disentanglement loss makes the style features extracted from stylized results similar when generating a series of stylized results by using a common style image and different content images. The disentanglement loss allows the network to extract main style patterns and exact content features to adapt to various content and style images, respectively. In summary, our main contributions are as follows:

- A flexible and efficient multi-adaptation arbitrary style transfer model involving two SA modules and one CA module.
- A novel disentanglement loss function for style and content disentanglement to extract well-directed style and content information.
- Various experiments illustrate that our method can preserve the detailed structures of the content images, and transfer rich style patterns of reference paintings to the generated results. Furthermore, we analyze the influence of different convolutional receptive field sizes on the CA module when calculating the local similarity between disentangled content and style features.

2 RELATED WORK

Style Transfer. Since Gatys et al. [3] proposed the first style transfer method by using CNNs, many works are devoted to promoting the transfer efficiency and generation effects. Some works [10, 15, 24] proposed real-time feed-forward style transfer networks, which can only transfer one kind of style by training an independent network. Arbitrary style transfer becomes a major research topic to obtain a wide range of applications. Chen et al. [1] initially swapped the style image patch onto content images based on patch similarity and achieved fast style transfer for arbitrary style images. Huang et al. [8] proposed adaptive instance normalization to adjust the mean and variance of content images to style images in a holistic fashion. Li et al. [17] aligned the covariance of style and content images by using WCT and transferred multilevel style patterns to content images to obtain better-stylized results. Avatar-Net [23] applied style decorator to guarantee semantically aligned and holistically matching for combining the local and global style patterns to stylized results. Park et al. [20] proposed a style-attention network to match the style features onto the content features for achieving good results with evident style patterns. Yao et al. [28] achieved multi-stroke style results using self-attention mechanism.

However, the above arbitrary style transfer methods cannot efficiently balance content structure preservation and style pattern rendering. The disadvantages of these methods can be observed in Section 4.2. Therefore, we aim to propose an arbitrary style transfer network that effectively transfers style patterns to content image while maintaining detailed content structures.

Feature Disentanglement. In recent years, researches [5, 7, 11, 13, 14, 19, 29, 31, 32] used generative adversarial networks (GANs), which can be applied to style transfer task in some cases, to achieve image-to-image translation. A significant thought suitable for the style transfer task is that the style and content feature should be disentangled because of the domain deviation. Zhu et al. [7] used two encoders to extract latent code to disentangle style and content

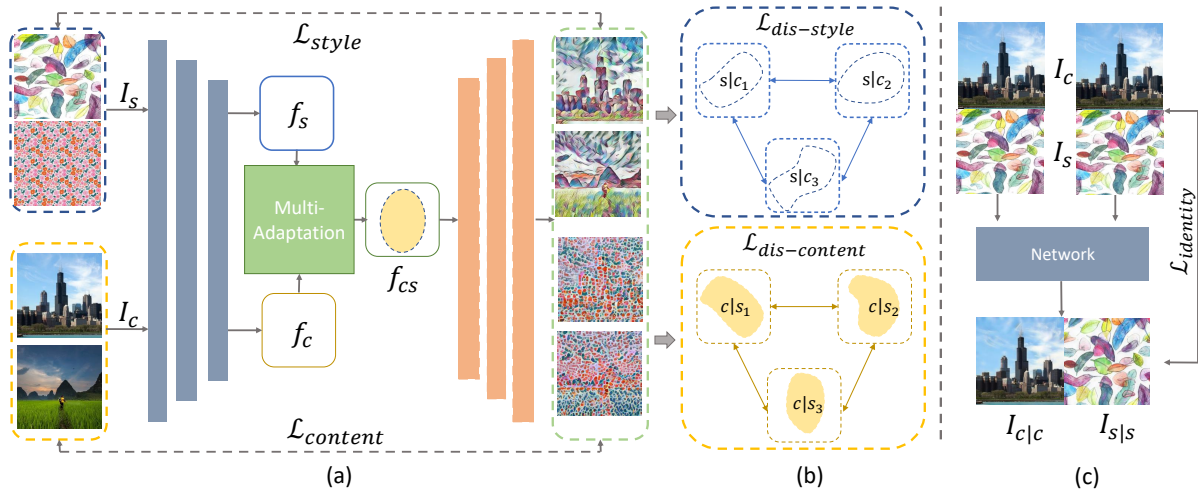


Figure 2: (a) Structure of our network. The blue blocks are the encoder, and the orange blocks are the decoder. Given input content images I_c and style images I_s , we can obtain corresponding features f_c and f_s , respectively, through the encoder. Then we feed f_c and f_s to the multi-adaptation module and acquire the generated features f_{cs} . Finally, we generate the results I_{cs} through the decoder. Moreover, the losses are calculated through a pretrained VGG19. $\mathcal{L}_{content}$ measures the difference between I_{cs} and I_c . $\mathcal{L}_{style}$ calculates the difference between I_{cs} and I_s . (b) Disentanglement loss. $\mathcal{L}_{dis-content}$ determines the content difference among stylized results, which are generated using different style images and a common content image. $\mathcal{L}_{dis-style}$ evaluates the style difference among stylized results, which are generated using different content images and a common style image. (c) Identity loss. $\mathcal{L}_{identity}$ quantifies the difference between $I_{c|c}(I_{s|s})$ and $I_c(I_s)$, where $I_{c|c}(I_{s|s})$ is the stylized result by using two common content (style) images.

representation. Kotovenko et al. [13] proposed a disentanglement loss to separate style and content. Kazemi et al. [11] described a style and content disentangled GAN (SC-GAN) to learn a semantic representation of content and textual patterns of style. Yu et al. [29] disentangled the input to latent code through an encoder-decoder network.

The existing disentanglement network usually adopted different encoders to decouple features through a training procedure. However, the structures of encoders are similar, which are not suitable enough for the disentanglement of style and content. In this paper, we design content and style SA modules to disentangle features specifically by considering the structures of content and texture of style.

3 METHODOLOGY

For the purpose of arbitrary style transfer, we propose a feed-forward network, which contains an encoder-decoder architecture and a multi-adaptation module. Figure 2 shows the structure of our network. We use a pretrained VGG19 network as an encoder to extract deep features. Given a content image I_c and style image I_s , we can extract corresponding feature maps $f_c^i = \mathcal{E}(I_c)$ and $f_s^i = \mathcal{E}(I_s)$, $i \in \{1, \dots, L\}$. However, the encoder is pretrained using ImageNet dataset for classification tasks, which is not suitable enough for style transfer tasks. Meanwhile, only a few domain-specific features can be extracted using a common encoder given the domain deviation between artistic paintings and photographic images. Therefore, we proposed a multi-adaptation module to disentangle style and content representation through the self-adaptation process, and then rearrange the disentangled style distribution according to

content distribution through the co-adaptation process. We can obtain stylized features f_{cs} through the multi-adaptation module. Section 3.1 describes the multi-adaptation module in detail. The decoder is a mirrored version of the encoder, and we can obtain the generated result $I_{cs} = \mathcal{D}(f_{cs})$. The model is trained by minimizing three types of loss functions described in Section 3.2.

3.1 Multi-adaptation Module

Figure 3 shows the multi-adaptation module, which is divided into three parts: position-wise content SA module, channel-wise style SA module, and CA module. We disentangle the content and style through two independent position-wise content and channel-wise style SA modules. We can disentangle the corresponding content/style representation $f_c(f_s)$ to $f_{cc}(f_{ss})$ through the content/style SA module. Then the CA module rearranges the style representation based on content representation and generates stylized features f_{cs} .

Position-wise Content Self-adaptation Module. Preserving the semantic structure of the content image in the stylized result is important, so we introduce position attention module in [2] to capture long-range information adaptively in content features. Given a content feature map $f_c \in \mathbb{R}^{C \times H \times W}$, $\hat{f}_c$ denotes the whitened content feature map, which removes textural information related to style by using whitening transform in [17]. We feed $\hat{f}_c$ to two convolution layers, and generate two new feature maps $\hat{f}_{c1}$ and $\hat{f}_{c2}$. Meanwhile, we feed f_c to another convolution layer to generate new feature map f_{c3} . We reshape $\hat{f}_{c1}$, $\hat{f}_{c2}$ and f_{c3} to $\mathbb{R}^{C \times N}$, where $N = H \times W$. Then, the content spatial attention map $A_c \in \mathbb{R}^{N \times N}$ is formulated

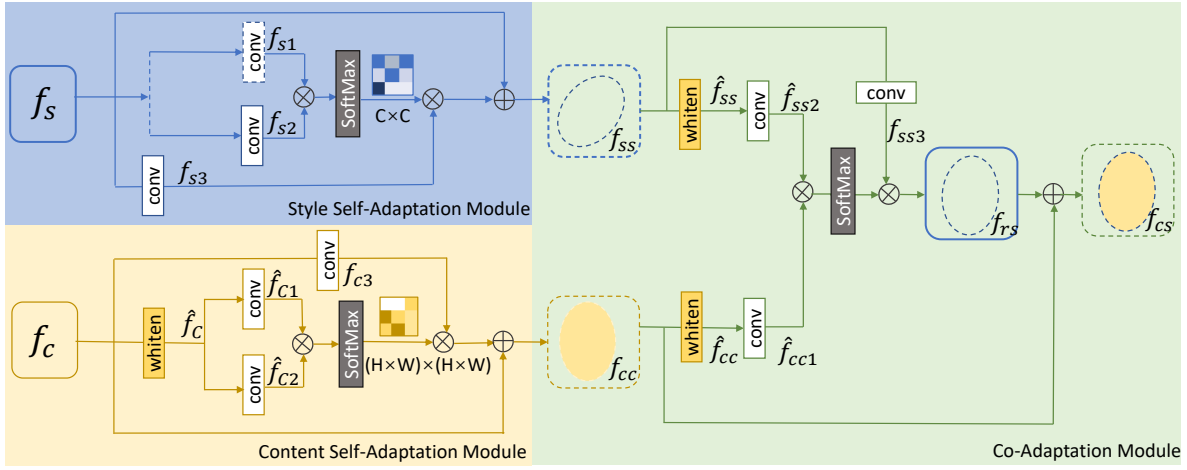


Figure 3: Multi-adaptation Network. We disentangle the content and style through two independent SA modules. We can disentangle the corresponding content/style representation $f_c(f_s)$ to $f_{cc}(f_{ss})$ through the content/style SA module. Then, the CA module rearranges the style distribution based on content distribution and generates rearranged features f_{rs} . Finally, we merge f_{cc} and f_{rs} to obtain stylized features f_{cs} .

as follows:

$$A_c = \text{SoftMax}(\hat{f}_{c1}^T \otimes \hat{f}_{c2}), \quad (1)$$

where $\otimes$ represents the matrix multiplication; and $\hat{f}_{c1}^T \otimes \hat{f}_{c2}$ denotes the position-wise multiplication between feature maps $\hat{f}_{c1}$ and $\hat{f}_{c2}$. Then, we obtain the enhanced content feature map f_c through a matrix multiplication and element-wise addition:

$$f_{cc} = f_{c3} \otimes A_c^T + f_c. \quad (2)$$

Channel-wise Style Self-adaptation Module. Learning style patterns (e.g., texture and strokes) of the style image is important for style transfer. Inspired by [3], the channel-wise inner product between the vectorized feature maps can represent style, so we introduce channel attention module in [2] to enhance the style patterns in style images. The input style features do not need to be whitened, which is different from the content SA module. We feed the style feature map $f_s \in \mathbb{R}^{C \times H \times W}$ to two convolution layers and generate two new feature maps f_{s1} and f_{s2} . Meanwhile, we feed f_s to another convolution layer to generate new feature map f_{s3} . We reshape f_{s1} , f_{s2} and f_{s3} to $\mathbb{R}^{C \times N}$, where $N = H \times W$. Then, the style spatial attention map $A_s \in \mathbb{R}^{C \times C}$ is formulated as follows:

$$A_s = \text{SoftMax}(f_{s1} \otimes f_{s2}^T), \quad (3)$$

where $f_{s1} \otimes f_{s2}^T$ represents the channel-wise multiplication between feature maps f_{s1} and f_{s2} . Then, we adjust the style feature map f_s through a matrix multiplication and an element-wise addition:

$$f_{ss} = A_s^T \otimes f_{s3} + f_s. \quad (4)$$

Co-adaptation Module. Through the SA module, we obtain the disentangled style and content features. Then, we propose the CA module to calculate the correlation between the disentangled features, and recombine them adaptively onto an output feature map. The generated results can not only retain the prominent content structure, but also adjust semantic content with the appropriate style patterns based on the correlation. Figure 3 shows the CA process. Initially, the disentangled style feature map f_{ss} and content

feature map f_{cc} are whitened to $\hat{f}_{ss}$ and $\hat{f}_{cc}$, respectively. Then, we feed $\hat{f}_{cc}$ and $\hat{f}_{ss}$ to two convolution layers to generate two new feature maps $\hat{f}_{cc1}$ and $\hat{f}_{ss2}$. Meanwhile, we feed feature map f_{ss} to another convolution layer to generate a new feature map f_{ss3} . We reshape $\hat{f}_{cc1}$, $\hat{f}_{ss2}$, and f_{ss3} to $\mathbb{R}^{C \times N}$, where $N = H \times W$. Then, the correlation map $A_{cs} \in \mathbb{R}^{N \times N}$ is formulated as follows:

$$A_{cs} = \text{SoftMax}(\hat{f}_{cc1}^T \otimes \hat{f}_{ss2}), \quad (5)$$

where the value of A_{cs} in position (i, j) measures the correlation between the i -th and the j -th position in content and style features, respectively. Then, the rearranged style feature map f_{rs} is mapped by:

$$f_{rs} = f_{ss3} \otimes A_{cs}^T. \quad (6)$$

Finally, the CA result is achieved by:

$$f_{cs} = f_{rs} + f_{cc}. \quad (7)$$

3.2 Loss Function

Our network contains three loss functions in training procedure.

Perceptual Loss. Similar to AdaIN [8], we use a pretrained VGG19 to compute the content and style perceptual loss. The content perceptual loss $\mathcal{L}_{content}$ is used to minimize the content difference between generated and content images, where

$$\mathcal{L}_{content}^i = \|\phi_i(I_{cs}) - \phi_i(I_c)\|_2. \quad (8)$$

The style perceptual loss $\mathcal{L}_{style}$ is utilized to minimize the style difference between generated and style images:

$$\mathcal{L}_{style}^i = \|\mu(\phi_i(I_{cs})) - \mu(\phi_i(I_c))\|_2 + \|\sigma(\phi_i(I_{cs})) - \sigma(\phi_i(I_c))\|_2, \quad (9)$$

where $\phi_i(\cdot)$ denotes the features extracted from i -th layer in a pretrained VGG19, $\mu(\cdot)$ denotes the mean of features, and $\sigma(\cdot)$ is the variance of features.

Identity Loss. Inspired by [20], we introduce the identity loss to provide a soft constraint on the mapping relation between style and content features. The identity loss is formulated as follows:

$$\mathcal{L}_{identity} = \|I_{c|c} - I_c\|_2 + \|I_{s|s} - I_s\|_2, \quad (10)$$

where $I_{c|c}$ indicates the generated results using one natural image as content and style images simultaneously and $I_{s|s}$ signifies generated results using one painting as content and style images simultaneously.

Disentanglement Loss. The style features should be independent from the target content to separate the style and content representation. That is, the content disentanglement loss renders the content features extracted from stylized results similar when generating a series of stylized results by using a common content image and different style images. The style disentanglement loss renders the style features extracted from stylized results similar when generating a series of stylized results by using a common style image and different content images. Therefore, we propose a novel disentanglement loss as follows:

$$\begin{aligned} \mathcal{L}_{dis_content}^i &= \|\phi_i(I_{c|s_1}) - \phi_i(I_{c|s_2})\|_2, \\ \mathcal{L}_{dis_style}^i &= \|\mu(\phi_i(I_{s|c_1})) - \mu(\phi_i(I_{s|c_2}))\|_2 \\ &\quad + \|\sigma(\phi_i(I_{s|c_1})) - \sigma(\phi_i(I_{s|c_2}))\|_2, \end{aligned} \quad (11)$$

where $I_{c|s_1}$ and $I_{c|s_2}$ are the generated results by using a common content image and different style images, and $I_{s|c_1}$ and $I_{s|c_2}$ represent the generated results by using a common style image and different content images. The total loss function is formulated as follows:

$$\begin{aligned} \mathcal{L} &= \lambda_c \mathcal{L}_{content}^j + \lambda_{dis_c} \mathcal{L}_{dis_content}^j + \lambda_{id} \mathcal{L}_{identity} \\ &\quad + \lambda_s \sum_{i=1}^L \mathcal{L}_{style}^i + \lambda_{dis_s} \sum_{i=1}^L \mathcal{L}_{dis_style}^i. \end{aligned} \quad (12)$$

In general, the loss functions constrain the global similarity between generated results and content/style images. The two SA modules calculate the long-range self-similarity of input features to disentangle the global content/style representation. The CA module rearranges the style distribution according to content distribution by calculating the local similarity between the disentangled content and style features in a non-local fashion. Therefore, our network can consider the global content structure and local style patterns to generate fascinating results.

4 EXPERIMENTS

4.1 Implementation Details

We use MS-COCO [18] as content dataset and WikiArt [21] as style dataset. The style and content images are randomly cropped to 256×256 pixels in the training stage. All image sizes are supported in the testing stage. We use *conv1_1*, *conv2_1*, *conv3_1*, and *conv4_1* layers in the encoder (pre-trained VGG19) to extract image features. The features of *conv4_1* layer are fed to the multi-adaptation module to generate the features f_{cs} . Furthermore, we use layer *conv4_1* to calculate the content perceptual and disentanglement loss and *conv1_1*, *conv2_1*, *conv3_1*, and *conv4_1* layers to calculate the style perceptual and disentanglement loss. The convolution kernel sizes utilized in the multi-adaptation module are all set to 1×1 . The

weights λ_c , λ_s , λ_{id} , λ_{dis_c} , and λ_{dis_s} are set to 1, 5, 50, 1, and 1, respectively.

4.2 Comparison with Prior Work

Qualitative Evaluation. We compare our method with four state-of-the-art works: AdaIN [8], WCT [17], SANet [20] and AAMS [28]. Figure 4 shows the stylized results. AdaIN [8] and WCT [17] adjust content images according to the second-order statistics of style images globally, but they ignore the local correlation between the content and style. Their stylized results have similar repeated textures in different image locations. AdaIN [8] adjusts the mean and variance of the content image to adapt to the style image globally. Inadequate textual patterns may be transferred in stylized images although the content structures are well preserved (3rd and 7th rows in Figure 4). Moreover, the results may even present different color distributions from style images (1st, 4th, 5th, and 8th rows in Figure 4). WCT [17] improves the style performance of AdaIN by adjusting the covariance of the content image through whitening and coloring transform operation. However, WCT would introduce content distortion (2nd, 4th, 6th, 7th, and 8th rows). SANet [20] uses style attention to match the style features to the content features, which can generate attractive stylized results with distinct style texture. However, the content structures in stylized results are unclear without feature disentanglement (2nd, 4th, and 8th rows in Figure 4). Moreover, the use of multi-layer features leads to repeated style patches in the results (eyes in the 3rd row in Figure 4). AAMS [28] also adopts self-attention mechanism, but the use of self-attention is not effective enough. In the results, the main structures of the content images are clear, but the other structures are damaged and the style patterns in the generated image are not evident (2nd, 3rd, 4th, 5th, and 8th rows in Figure 4).

Disentangled content and style features in multi-adaptation network can well represent a domain-specific characteristic, which is different from the abovementioned methods. Therefore, the results generated by our method can further preserve the content and style information. Moreover, our method can generate good results, which have distinct content structures and rich style patterns, by adaptively adjusting the disentangled content and style features. The content images can be rendered by corresponding style patterns based on their semantic structures (1st row in Figure 4).

User Study. We conduct user studies to compare the visual performance of ours and the aforementioned SOTA methods further. We select 20 style images and 15 content images to generate 300 results for each method. Initially, we show each content-style pair to the participants. Then we show two results (one is by our method and the other is randomly selected from one of the SOTA methods) to them. We ask the participants four questions: (1) which stylized result further preserves the content structures, (2) which stylized result further transfers the style patterns, (3) which stylized result has improved visual quality overall, (4) when selecting the images in question (3), which factor is mainly considered: content, style, both, or neither? We ask 30 participants to do 50 rounds of comparisons and obtain 1500 votes for each question. Figure 5 shows the statistical results. Figure 5(a) presents a Sankey diagram, which is used to demonstrate the direction of data flow. For example, among the users who select our method in style, most people also select our

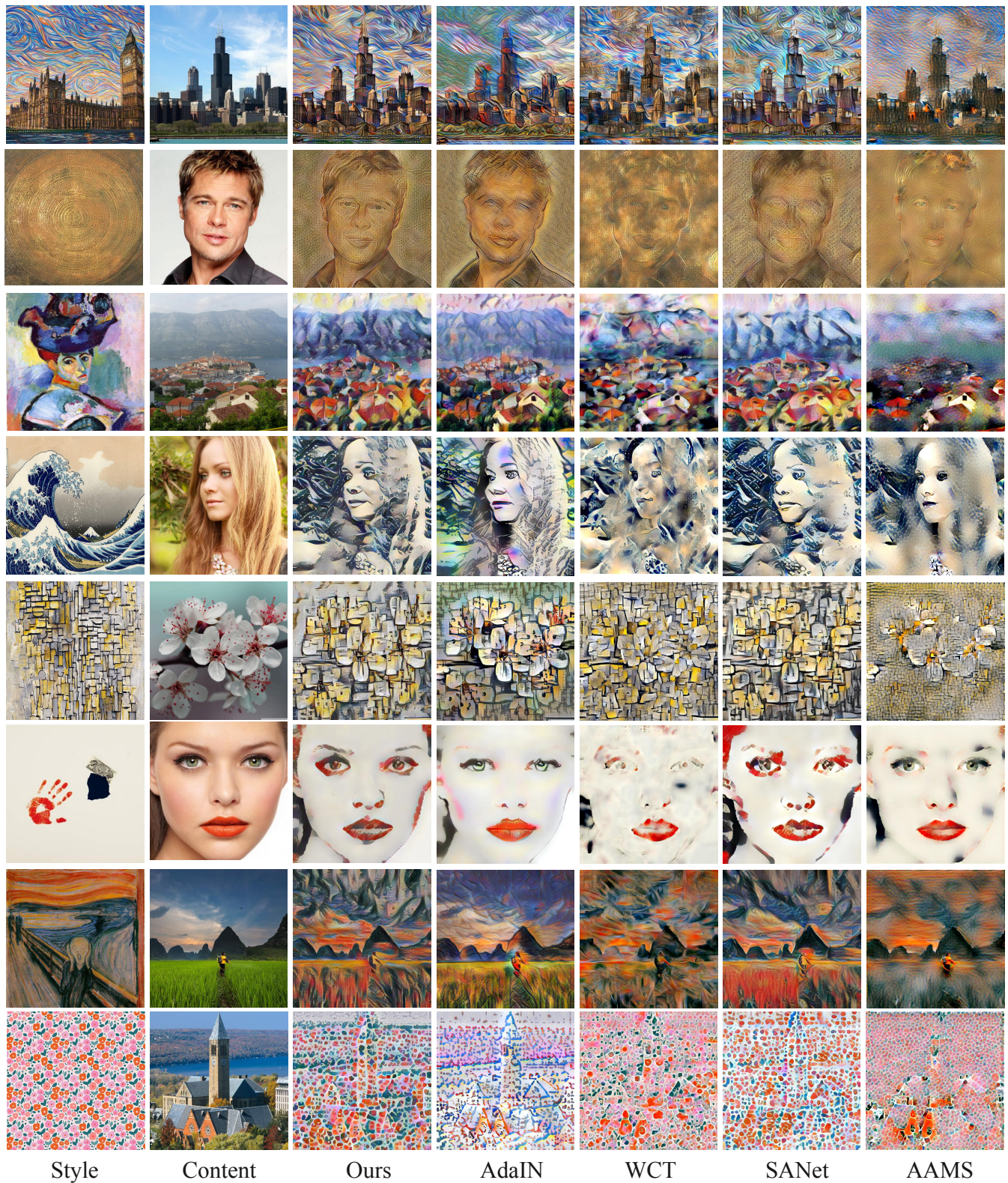


Figure 4: Comparison of stylized results with SOTA methods. The first column shows style images, and the second column shows content images. The remaining columns are stylized results by our methods, AdaIN [8], WCT [17], SANet [20], and AAMS [28].

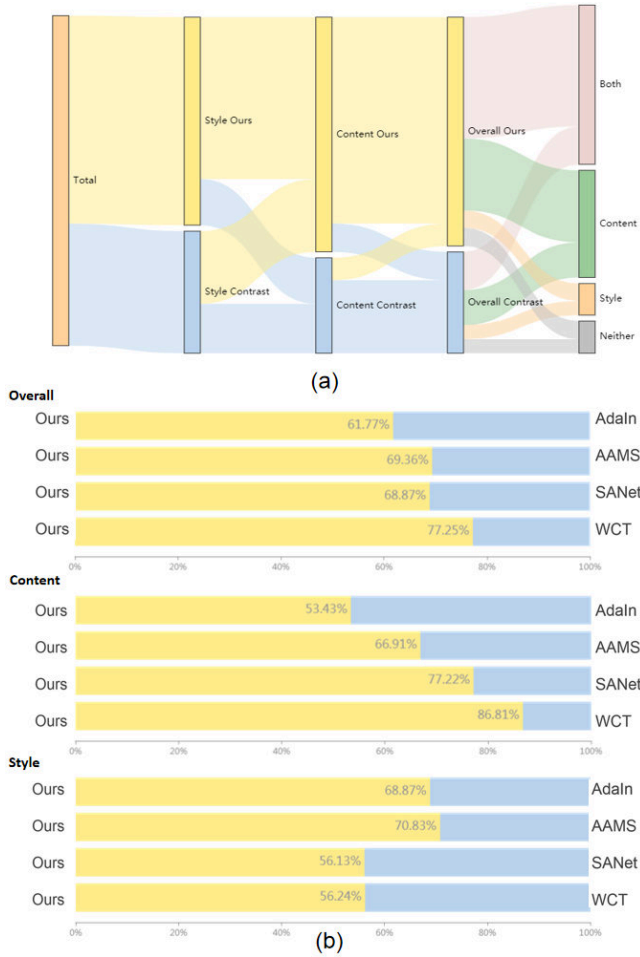


Figure 5: User study results: (a) the Sankey diagram shows the overall results of users' preference between our results and the contrast results, (b) detailed results with each contrast method.

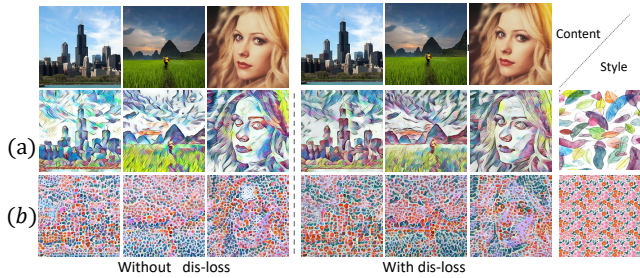


Figure 6: Comparison of stylized results with/without disentanglement loss. (a) With style and content disentanglement, different content images have unified style patterns, which is the key component of style image (purple feathers). (b) With style and content disentanglement, the content structures are more visible.

content, and a few select the content of the contrast methods. We can conclude from Figure 5(a) that regardless of the aspects of content, style, or overall effect, our method obtains the majority

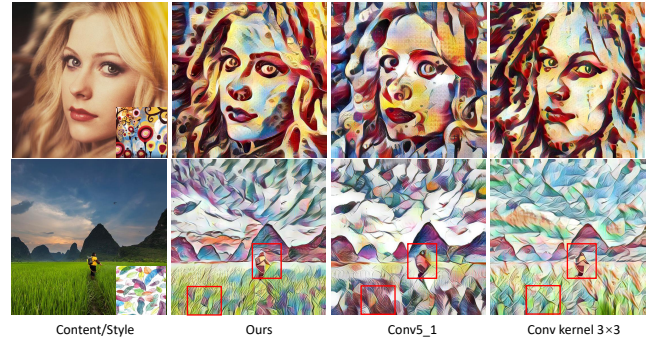


Figure 7: Comparison of stylized results with different receptive field size. The meticulous structures of results in 3rd and 4th columns are not well-preserved compared with our results (see details in red box).

votes. The participants are more impressionable to content than style when selecting results with improved visual performance overall.

Subsequently, we compare our method with each comparison method in the aspect of content, style, and overall separately in Figure 5(b). The overall performance of our method is better than that of AdaIN [8], has clear advantages in style and comparable content preservation ability. Our results, compared with those of WCT [17], have clear advantages in content and comparable style patterns. Our results, compared with those of AAMS [28], have clear advantages in content and style.

Quantitative Evaluation. To quantify the style transferring and content preservation ability of our method, we first introduce an artist classification model to evaluate how well the stylized results are rendered by every artist's style. We select 5 artists each with 1000 paintings and divide them into training and testing sets in the ratio of 8:2. Then we fine-tune the pretrained VGG19 model by using the training set. We generate 1000 stylized images for each method. We feed the stylized images to the artist classification model to calculate the accuracy. Second, we use a content classification model to quantify the content preservation effects of different methods. We randomly select 5 classes from the ImageNet dataset. Then we fine-tune the pretrained VGG19 model by using the training set of ImageNet. We use the corresponding validation set (five classes, each class includes 50 content images) to generate 1000 stylized images for each method. We feed the stylized images to the content classification model to calculate the accuracy.

Table 1 shows the classification results. The style and content classification accuracy values of our method are relatively high, which illustrates that our method can obtain a trade-off between content and style. Although the SANet achieves the highest style classification accuracy, the content classification accuracy is too low to obtain attractive results. The content classification accuracy of AdaIN is high, but the style classification accuracy is low. In general, the content/style classification results of each method are consistent with user study results. The small statistical difference is because participants may be influenced by the effect of generated results when they select improved content/style results.

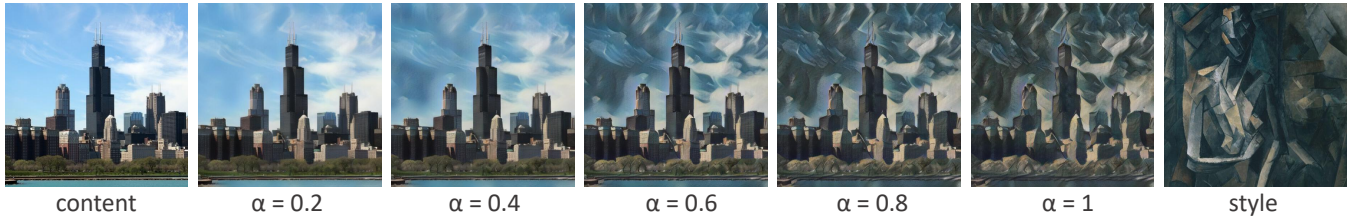


Figure 8: Trade-off between content and style.



Figure 9: Style interpolation.

Table 1: Classification accuracy.

	AdaIN	WCT	SANet	AAMS	Ours
style(%)	55.7	61.5	65.2	57.9	62.9
content(%)	42.0	22.8	29.1	33.0	34.6

4.3 Ablation Study

Verify the effect of disentanglement loss. We compare the generated results with and without disentanglement loss to verify the effect of disentanglement loss. As shown in Figure 6, using disentanglement loss can generate results with the key style patterns of style image (purple feathers, Figure 6(a)) or more visible content structures (Figure 6(b)), compared with the stylized results without disentanglement loss. With disentanglement loss, the stylized results can preserve unified style patterns and salient content structure.

The Influence of Convolutional Receptive Field Size. The receptive field size of convolutional operation can influence the generated results when calculating the local similarity between disentangled content and style features in the CA module. There are two factors that can change the receptive field size. First, the convolution kernel size is fixed to 3×3 in the encoder; the deeper the model is, the larger receptive field we can obtain. Therefore, we use *conv5_1* by replacing *conv4_1* of the encoder layer to obtain a large receptive field. Second, we feed the content and style features to two convolutional layers in the CA module and calculate their correlation. The convolutional kernel size used in this module is related to the receptive field, which influences the size of the region used to calculate the correlation. Thus, we change the convolutional kernel size from 1×1 to 3×3 to obtain a large receptive field.

As shown in Figure 7, the stylized results using *conv5_1* or kernel size 3×3 are transferred to more local style patterns (e.g., circle patterns in the 1st row and feather patterns in the 2nd row), and the content structures are highly distorted. The results prove that a large receptive field pays further attention to the local structure and will spatially distort the global structure.

4.4 Applications

Trade-off between content and style. We can adjust the style pattern weights in the stylized results by changing α in the following function:

$$I_{cs} = \mathcal{D}(\alpha f_{cs} + (1 - \alpha)f_c). \quad (13)$$

When $\alpha = 0$, we obtain the original content image. When $\alpha = 1$, we obtain the fully stylized image. We change α from 0 to 1. Figure 8 shows the results.

Style Interpolation. For a further flexible application, we can merge multiple style images into one generated result. Figure 9 presents the examples. We can also change the weights of different styles.

5 CONCLUSIONS AND FUTURE WORK

In this paper, we propose a multi-adaptation network to disentangle global content and style representation and adjust the style distribution to content distribution by considering the long-range local similarity between the disentangled content and style features. Moreover, we propose a disentanglement loss to render the style features independent from the target content feature for constraining the separation of style and content features. Our method can achieve a trade-off between content structure preservation and style pattern rendering. Adequate experiments show that our network can consider the global content structure and local style patterns to generate fascinating results. We also analyze the effect of receptive field size in the CNNs on the generated results.

In future work, we aim to develop a style image selection method to recommend appropriate style images for a given content image based on the global semantic similarity of content and style for additional practical applications.

ACKNOWLEDGMENTS

This work was supported by National Natural Science Foundation of China under nos. 61832016 and 61672520, and by CASIA-Tencent Youtu joint research project.

REFERENCES

- [1] Tian Qi Chen and Mark Schmidt. 2016. Fast patch-based style transfer of arbitrary style. In *Constructive Machine Learning Workshop, NIPS*.
- [2] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. 2019. Dual attention network for scene segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 3146–3154.
- [3] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. 2016. Image style transfer using convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2414–2423.
- [4] Leon A Gatys, Alexander S Ecker, Matthias Bethge, Aaron Hertzmann, and Eli Shechtman. 2017. Controlling perceptual factors in neural style transfer. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 3985–3993.
- [5] Abel Gonzalez-Garcia, Joost Van De Weijer, and Yoshua Bengio. 2018. Image-to-image translation for cross-domain disentanglement. In *Advances in Neural Information Processing Systems (NeurIPS)*. 1287–1298.
- [6] Shuyang Gu, Congliang Chen, Jing Liao, and Lu Yuan. 2018. Arbitrary style transfer with deep feature reshuffle. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 8222–8231.
- [7] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. 2018. Multimodal unsupervised image-to-image translation. In *European Conference on Computer Vision (ECCV)*. Springer, 172–189.
- [8] Xun Huang and Belongie Serge. 2017. Arbitrary style transfer in real-time with adaptive instance normalization. In *IEEE International Conference on Computer Vision (ICCV)*. IEEE, 1501–1510.
- [9] Yongcheng Jing, Xiao Liu, Yukang Ding, Xinchao Wang, Errui Ding, Mingli Song, and Shilei Wen. 2020. Dynamic Instance Normalization for Arbitrary Style Transfer. In *Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI)*. AAAI Press.
- [10] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. 2016. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision (ECCV)*. Springer, 694–711.
- [11] Hadi Kazemi, Seyed Mehdi Iranmanesh, and Nasser Nasrabadi. 2019. Style and content disentanglement in generative adversarial networks. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 848–856.
- [12] Nicholas Kolkin, Jason Salavon, and Gregory Shakhnarovich. 2019. Style Transfer by Relaxed Optimal Transport and Self-Similarity. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 10051–10060.
- [13] Dmytro Kotovenko, Artsiom Sanakoyeu, Sabine Lang, and Bjorn Ommer. 2019. Content and style disentanglement for artistic style transfer. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. IEEE, 4422–4431.
- [14] Dmytro Kotovenko, Artsiom Sanakoyeu, Pingchuan Ma, Sabine Lang, and Bjorn Ommer. 2019. A Content Transformation Block for Image Style Transfer. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 10032–10041.
- [15] Chuan Li and Michael Wand. 2016. Precomputed real-time texture synthesis with markovian generative adversarial networks. In *European Conference on Computer Vision (ECCV)*. Springer, 702–716.
- [16] Xueting Li, Sifei Liu, Jan Kautz, and Ming-Hsuan Yang. 2019. Learning linear transformations for fast image and video style transfer. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 3809–3817.
- [17] Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. 2017. Universal style transfer via feature transforms. In *Advances in Neural Information Processing Systems (NeurIPS)*. 386–396.
- [18] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision (ECCV)*. Springer, 740–755.
- [19] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. 2017. Unsupervised image-to-image translation networks. In *Advances in Neural Information Processing Systems (NeurIPS)*. 700–708.
- [20] Dae Young Park and Kwang Hee Lee. 2019. Arbitrary Style Transfer With Style-Attentional Networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 5880–5888.
- [21] Fred Phillips and Brandy Mackintosh. 2011. Wiki Art Gallery, Inc.: A case for critical thinking. *Issues in Accounting Education* 26, 3 (2011), 593–608.
- [22] Falong Shen, Shuicheng Yan, and Gang Zeng. 2018. Neural style transfer via meta networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 8061–8069.
- [23] Lu Sheng, Ziyi Lin, Jing Shao, and Xiaogang Wang. 2018. Avatar-net: Multi-scale zero-shot style transfer by feature decoration. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 8242–8250.
- [24] Dmitry Ulyanov, Vadim Lebedev, Andrea Vedaldi, and Victor Lempitsky. 2016. Texture Networks: Feed-Forward Synthesis of Textures and Stylized Images. In *International Conference on International Conference on Machine Learning (ICML)* (New York, NY, USA). JMLR.org, 1349–1357.
- [25] Hao Wang, Xiaodan Liang, Hao Zhang, Dit-Yan Yeung, and Eric P Xing. 2017. ZM-NET: Real-time zero-shot image manipulation network. (2017). arXiv:arXiv:1703.07255
- [26] Miao Wang, Xu-Quan Lyu, Yi-Jun Li, and Fang-Lue Zhang. 2020. VR content creation and exploration with deep learning: A survey. *Computational Visual Media* 6, 1 (2020), 3–28.
- [27] Hao Wu, Zhengxing Sun, and Weihang Yuan. 2018. Direction-Aware Neural Style Transfer. In *Proceedings of the 26th ACM International Conference on Multimedia* (Seoul, Republic of Korea). Association for Computing Machinery, New York, NY, USA, 1163–1171.
- [28] Yuan Yao, Jianqiang Ren, Xuansong Xie, Weidong Liu, Yong-Jin Liu, and Jun Wang. 2019. Attention-aware multi-stroke style transfer. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 1467–1475.
- [29] Xiaoming Yu, Yuanqi Chen, Shan Liu, Thomas Li, and Ge Li. 2019. Multi-mapping Image-to-Image Translation via Learning Disentanglement. In *Advances in Neural Information Processing Systems (NeurIPS)*. 2990–2999.
- [30] Yuheng Zhi, Huawei Wei, and Bingbing Ni. 2018. Structure Guided Photorealistic Style Transfer. In *Proceedings of the 26th ACM International Conference on Multimedia*. Association for Computing Machinery, 365–373.
- [31] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision (CVPR)*. IEEE, 2223–2232.
- [32] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. 2017. Toward multimodal image-to-image translation. In *Advances in Neural Information Processing Systems (NeurIPS)*. 465–476.