RAST: Restorable Arbitrary Style Transfer via Multi-restoration

Yingnan Ma Chenqiu Zhao Xudong Li
Anup Basu
Dept. of Computing Science, University of Alberta, Canada
{ma4, zhao.chenqiu, xudong9, basu}@ualberta.ca

Figure 1: Results generated by our Restorable Arbitrary Style Transfer (RAST) architecture. The content image is shown in the top-left corner. The style images are shown in the top-right corner of stylized images.

Abstract

Arbitrary style transfer aims to reproduce the target image with the artistic or photo-realistic styles provided. Even though existing approaches can successfully transfer style information, arbitrary style transfer still faces many challenges, such as the content leak issue. Specifically, the embedding of artistic style can lead to content changes. In this paper, we solve the content leak problem from the perspective of image restoration. In particular, an iterative architecture is proposed to achieve the Restorable Arbitrary Style Transfer (RAST), which can realize transmission of both content and style information through multi-restorations. We control the content-style balance in stylized images by the accuracy of image restoration. In order to ensure effectiveness of the proposed RAST architecture, we design two novel loss functions: multi-restoration loss and style difference loss. In addition, we propose a new quantitative evaluation method to measure content preservation performance and style embedding performance. Comprehensive experiments comparing with state-of-the-art methods demonstrate that our proposed architecture can produce stylized images with superior performance on content preservation and style embedding.

1. Introduction

Arbitrary style transfer [37, 39, 5] is a long-standing image processing topic, which aims to render an image with referenced arbitrary styles. The styles can be either artistic or photo-realistic. Since Gatys et al. [13] proposed using convolutional neural networks to solve artistic style transfer, neural style transfer [9, 22, 46] has attracted significant attention as an application of computer vision in the field of art. Recently, transformer-based approaches [6, 51] were also involved in neural style transfer by applying the self-attention mechanism and positional encoding. Although ex-

isting algorithms can achieve the delivery of styles, the arbitrary style transfer task still has challenges maintaining a balance between content preservation and style embedding. In artistic style transfer, it is challenging to preserve the content details during artistic style embedding due to style differences. Excessive style embedding can result in changes in content information. For photo-realistic style transfer, the architecture pays more attention to content preservation, which can result in a lack of delicate patterns [20].

We propose a novel framework that tackles the problem of content-style balance from the perspective of image restoration. During the style transformation process, most of the style features do not change the content information. Thus, embedding these style features does not impact content restoration. However, the embedding of certain style features can results in content leak [1], which can lead to the failure of image restoration. Our purpose is to embed more style features into content features without impacting the restoration of content and style images.

Differing from existing methods that minimize content and style differences between input images and stylized images, we achieve style transfer by minimizing the difference between restored images and input images and maximizing the style difference between the stylized images and the input images. This approach can avoid the interference caused by different content information or style information during feature evaluation. Through extensive experiments, both qualitative and quantitative results demonstrate that our proposed framework has better performance on content preservation and style embedding. In addition, our framework can restore the stylized images back to the input images more accurately. The contributions of the this paper are summarized below:

- We propose a quadruple-cycle framework to support iterative learning so that Restorable Arbitrary Style Transfer (RAST) can be achieved. The RAST framework can realize transmission of both content and style information through multiple restorations. Content transmission can guarantee the performance of content preservation. Style transmission can ensure the performance of style consistency.
- We propose multi-restoration loss and style difference loss which are extended from perceptual loss. In particular, the style difference loss can enable our framework to embed more style patterns that do not affect restoration. The multi-restoration loss can achieve content transmission and style transmission through multi-restorations.
- We also propose a novel quantitative evaluation approach that measures the performance of content preservation and style embedding from the perspective of restoration.

The remainder of this paper is organized as below: Section 2 reviews state-of-the-art style transfer methods and indicates the difference between our proposed framework and existing approaches. Section 3 introduces our proposed RAST framework and related loss functions. Device information and data arrangement are summarized in Section 4. We also demonstrate the experimental results in this section. We illustrate the effectiveness of our proposed RAST framework by comparing it with eight state-of-the-art approaches. We evaluate performance using both qualitative and quantitative measures.

2. Related Work

2.1. Image Style Transfer

Image style transfer has been an attractive research topic since the 1990s. Originally, it was proposed as a stroke-based rendering [17, 11] algorithm, which can add strokes on target images with objective guidance. Later on, it was further explored as an image analogy [18, 42, 12] problem to learn the transformation from paired images. Style transfer can also be solved as an image filtering [49] problem by utilizing Gaussian filters [14] and bilateral filters [45]. However, these methods can only learn low-level image features, which cannot guarantee image structures.

In addition to traditional approaches, Gatys et al. [13] proposed a neural-based style transfer approach, which utilized a convolutional neural network to recombine content features and style features from layers. It can achieve iterative optimization with the support of the Gram matrix. Inspired by Gatys et al., the feed-forward neural network [9, 22, 46] has been widely applied to solve style transfer tasks. With the utilization of encoder-decoder architecture, many transformation-based methods were proposed. The AdaIN [21] approach proposed the adaptive instance normalization layer to obtain the mean and variance of features. Based on AdaIN, WCT [31] replaced variance by covariance utilizing whitening and coloring, which was further improved by OptimalWCT[35] with a more general closed-form solution. Similarly, LST [30] proposed linear transformation on cross-domain features for solving universal style transfer. Furthermore, the transformation-based methods could also be improved by neural flows [8, 19, 27]. ArtFlow [1] utilized reversible neural flows, which could solve the content leak problem. In addition to the above transformation-based approaches, image style transfer can also be solved by patch-based methods [10, 29, 41]. StyleSwap [3] replaced each activation patch of a content image by matched style patches. Similarly, Avatar-net [41] proposed a patch-based style decorator, which could use the pattern characteristics to decorate the content features.

With the wide use of attention mechanism [47, 52], attention-based methods [37, 7, 5] were involved in the

style transfer field. By calculating the style attention on feature spaces, SANet [37] could embed matched style features onto content features. Considering features from multiple layers, SANet combined local and global style patterns. Furthermore, IEContraAST [2] explored SANet as the backbone by involving internal-external learning [38, 44] and contrastive learning [23, 39, 50]. With the external learning of a discriminator [15], IEContraAST could learn human-aware style. Also, contrastive learning could ensure the accuracy of content and style transmission. In addition, MANet [7] and MCCNet [5] were proposed employing multi-adaptation and multi-channel correlation techniques respectively, which enhanced the performance of feature fusion. AdaAttN [33] proposed an Adaptive Attention Normalization module, which can learn spatial attention from shallow and deep features. PAMA [36] proposed progressive attentional manifold alignment, which could reposition the style features dynamically by repeated attention operations. Furthermore, transformer-based methods [6, 51, 34] were proposed in the style transfer field by using self-attention mechanism and positional encoding. StyTr^2 [6] proposed content-aware positional coding and modified the transformer structure to fit a style transfer task. Similarly, StyleFormer [51] modified transformer structure utilizing a style bank and parametric composition, which could guarantee the style transformation performance. TxST [34] proposed a text-driven architecture, which could achieve style transfer according to the text description.

2.2. Restoration-based Style Transfer

Restoration-based style transfer [55, 25, 4, 54] was first employed by CycleGAN [55]. Similar to GAN-based [15, 40, 28] approaches, it solved style transfer as a domain adaption problem. Utilizing two generators and two discriminators, CycleGAN could achieve bidirectional image generation from domain A to B and B to A. With the use of pixel-level consistency loss, the restored images could help generators and discriminators to optimize the accuracy of domain adaption. However, CycleGAN still suffered from the geometry structure constraint. Specifically, the images in domains A and B need to have similar geometrical structures. If the geometric structures are different, the adaption performance of color and texture could not be guaranteed. Similarly, DiscoGAN [25] involves restoration with learning cross-domain relations by double generators. Based on CycleGAN, StarGAN [4] was proposed with a classification module and domain classification loss, which could achieve the restoration process by using a single generator. In addition to StarGAN, CAST [54] proposed a different solution to replace the double generators from CycleGAN. It employed contrastive learning to involve multiple style images and utilized a memory bank architecture [16] to store style information, which could also achieve arbitrary style transfer with a single generator.

In contrast, we propose a Restorable Arbitrary Style Transfer (RAST) framework, which handles the content leak issue from the perspective of image restoration. Instead of involving a classification module and style memory bank, the proposed RAST framework achieves multiple restorations via iterative learning. By sharing the same parameters for the attention-based transfer block, our framework can achieve bidirectional multi-restoration with the same transfer block. Differing from CycleGAN, DiscoGAN, StarGAN, and CAST, the proposed RAST framework can achieve transmission of both content and style information. Moreover, we propose multi-restoration loss and style difference loss at the feature level to support our RAST framework. Note that CycleGAN, DiscoGAN, StarGAN, and CAST are mainly guided by an adversarial process to achieve domain adaption. However, our framework is mainly guided by the restoration process to achieve the delivery of content and style information.

3. Proposed Method

Previous arbitrary style transfer approaches usually suffer from the content leak problem. In order to handle this problem, the Restorable Arbitrary Style Transfer (RAST) framework is proposed. Through multi-restoration of content and style images, our framework can not only achieve transmission of content and style information but it also filters out the style features that interfere with image restoration. To ensure effectiveness of the proposed framework, we also design multi-restoration loss and style difference loss to guide the learning process. The overview of our framework is shown in Figure 2. In our transfer block, we employ SANet [37] as the backbone, which can map the correspondence between content feature map and style feature map semantically by calculating style attention. The pre-trained VGG-19 network [43] is utilized as the encoder to obtain feature maps. The decoder is a symmetric VGG-19 network [21], which can decode feature maps into images. Apart from this, multi-scale discriminators [48] are applied as external [2] discriminators to learn human-aware style information.

3.1. Network Architecture

The main architecture of the proposed framework is shown in Figure 2. Assuming $\mathcal{T}$ is the transfer block of the proposed approach, the input images of $\mathcal{T}$ include a content image $\mathbf{I}_c$ and a style image $\mathbf{I}_s$, and the output is the style transfer image $\mathbf{I}_o$, which is a combination of the content part of $\mathbf{I}_c$ and the style part of $\mathbf{I}_s$. Mathematically:

$$\mathbf{I}_o = \mathcal{T}(\mathbf{I}_c, \mathbf{I}_s) \;\equiv\; \mathbf{I}_c \xrightarrow{\mathbf{I}_s} \mathbf{I}_o, \tag{1}$$

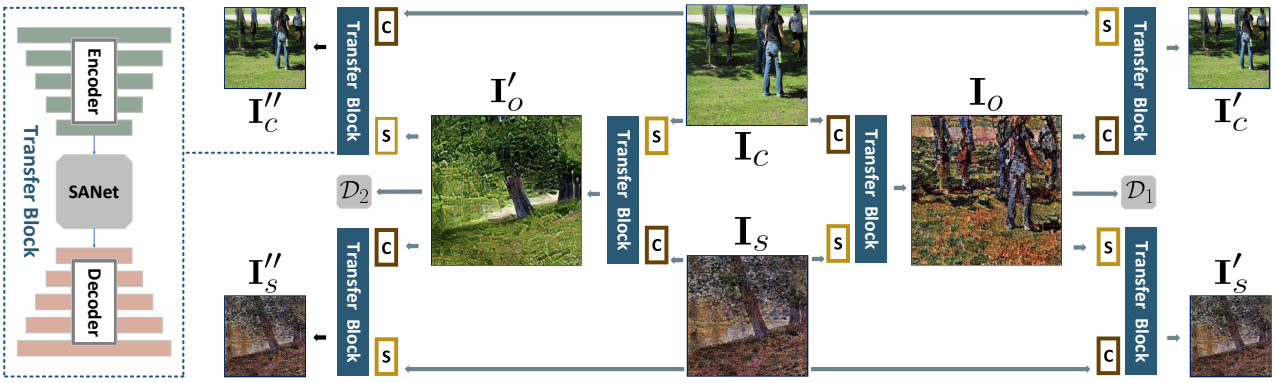

Figure 2: Overview of the proposed Restorable Arbitrary Style Transfer (RAST) framework. It includes one transfer block, which is utilized iteratively for stylization and restoration by sharing the same parameters. It also employs two external discriminators $\mathcal{D}_1$ and $\mathcal{D}_2$, which can deal with realistic-to-artistic and artistic-to-realistic processes, respectively.

where $\mathbf{I}_c \xrightarrow{\mathbf{I}_s} \mathbf{I}_o$ is defined as the notation of the process of style transfer $\mathbf{I}_o = \mathcal{T}(\mathbf{I}_c, \mathbf{I}_s)$ in the proposed approach, which will be used in the remaining part of this section.

To the best of our knowledge, there is no golden standard of what is content or style. Fortunately, in the proposed approach, we re-input the stylized image $\mathbf{I}_o$ of the proposed approach and the style of the content image, with the new output expected to approach the original content image $\mathbf{I}_c$, when the transfer block works appropriately. Mathematically, this can be expressed as:

$$\mathbf{I}_c \xrightarrow{\mathbf{I}_s} \mathcal{T}(\mathbf{I}_c, \mathbf{I}_s) = \mathbf{I}_o \xrightarrow{\mathbf{I}_c} \mathcal{T}(\mathbf{I}_o, \mathbf{I}_c) = \mathbf{I}'_c \approx \mathbf{I}_c, \tag{2}$$

where $\mathbf{I}'_c$ is the restored content image, which is supposed to be close to the original content image $\mathbf{I}_c$. In addition, we can also restore the style image by using $\mathbf{I}_s$ as the content image and $\mathbf{I}_o$ as the style image. The restored style image $\mathbf{I}'_s$ is supposed to approach the original style image $\mathbf{I}_s$, which is mathematically shown in Equation 3, where $\mathbf{I}'_s$ is the restored style image which is supposed to be close to the original style image $\mathbf{I}_s$. Similarly, when we switch $\mathbf{I}_c$ and $\mathbf{I}_s$, we can in total get 4 different restored images, which are supposed to be close to the original input images. Overall, the multiple restored images of the proposed framework are shown as follows:

$$\begin{aligned}
&\mathbf{I}_c \xrightarrow{\mathbf{I}_s} \mathcal{T}(\mathbf{I}_c, \mathbf{I}_s) = \mathbf{I}_o \xrightarrow{\mathbf{I}_c|\mathbf{I}_s} \begin{cases} \mathcal{T}(\mathbf{I}_o, \mathbf{I}_c) = \mathbf{I}'_c \approx \mathbf{I}_c \\ \mathcal{T}(\mathbf{I}_s, \mathbf{I}_o) = \mathbf{I}'_s \approx \mathbf{I}_s \end{cases} \\
&\mathbf{I}_s \xrightarrow{\mathbf{I}_c} \mathcal{T}(\mathbf{I}_s, \mathbf{I}_c) = \mathbf{I}'_o \xrightarrow{\mathbf{I}_s|\mathbf{I}_c} \begin{cases} \mathcal{T}(\mathbf{I}'_o, \mathbf{I}_s) = \mathbf{I}''_s \approx \mathbf{I}_s \\ \mathcal{T}(\mathbf{I}_c, \mathbf{I}'_o) = \mathbf{I}''_c \approx \mathbf{I}_c, \end{cases}
\end{aligned} \tag{3}$$

where $\mathbf{I}'_c$, $\mathbf{I}''_c$, $\mathbf{I}'_s$ and $\mathbf{I}''_s$ are restored content images and style images respectively.

3.2. Loss Function

As discussed in Section 3.1, our framework can achieve restorable arbitrary style transfer through a multi-restoration process. To ensure the accuracy of the restoration, we design the feature-level multi-restoration loss $\mathcal{L}_{multi}$ and style difference loss $\mathcal{L}_{diff}$ to ensure feature consistency and style embedding, respectively. We utilize perceptual loss [22, 21] as the base functions to calculate the feature difference. As shown in Equation 4, f_s can calculate the difference of style features by the mean and standard deviation of the feature maps, where ϕ_i represents the i_{th} layer of the VGG-19 network. Specifically, Relu1_1, Relu2_1, Relu3_1, Relu4_1, and Relu5_1 layers are utilized to capture style feature maps. $\mathbb{E}$ denotes the mean of the feature maps and σ represents the standard deviation of the feature maps. Besides, f_c can calculate the difference of content features as shown in Equation 5, where Relu4_1 and Relu5_1 layers are employed to extract content features.

$$\begin{aligned}
f_s(\mathbf{I}_1, \mathbf{I}_2) = &\sum_{i=1}^{L} \|\mathbb{E}(\phi_i(\mathbf{I}_1)) - \mathbb{E}(\phi_i(\mathbf{I}_2))\|_2 \\
&+ \|\sigma(\phi_i(\mathbf{I}_1)) - \sigma(\phi_i(\mathbf{I}_2))\|_2
\end{aligned} \tag{4}$$

$$\begin{aligned}
f_c(\mathbf{I}_1, \mathbf{I}_2) = &\|\phi_{4_1}(\mathbf{I}_1) - \phi_{4_1}(\mathbf{I}_2)\|_2 \\
&+ \|\phi_{5_1}(\mathbf{I}_1) - \phi_{5_1}(\mathbf{I}_2)\|_2 .
\end{aligned} \tag{5}$$

Based on the above functions, we design the multi-restoration loss $\mathcal{L}_{multi}$, which can not only calculate the feature difference between restored images and input images but also measure the differences among multi-restored images.

$$\begin{aligned}
\mathcal{L}_{multi} = &f_c(\mathbf{I}'_c, \mathbf{I}_c) + f_c(\mathbf{I}''_s, \mathbf{I}_s) + f_s(\mathbf{I}'_s, \mathbf{I}_s) + f_s(\mathbf{I}''_c, \mathbf{I}_c) \\
&+\alpha[f_c(\mathbf{I}'_c, \mathbf{I}''_c) + f_s(\mathbf{I}'_c, \mathbf{I}''_c) + f_c(\mathbf{I}'_s, \mathbf{I}''_s) + f_s(\mathbf{I}'_s, \mathbf{I}''_s)].
\end{aligned} \tag{6}$$

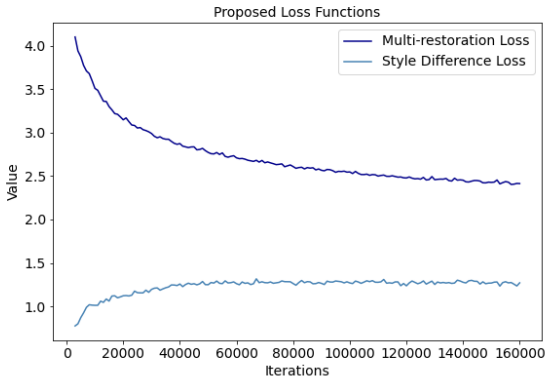

Figure 3: Training performance of proposed loss functions after the initialization stage.

As shown in Equation 6, $\mathcal{L}_{multi}$ first calculates the feature difference between restored images $\mathbf{I}'_{c|s}$, $\mathbf{I}''_{c|s}$ and input images $\mathbf{I}_{c|s}$. Specifically, content feature difference through restoration is calculated between $\mathbf{I}'_c$ and $\mathbf{I}_c$, $\mathbf{I}''_s$ and $\mathbf{I}_s$. The reason is that the content features in $\mathbf{I}'_c$ and $\mathbf{I}''_s$ are transmitted from stylized images $\mathbf{I}_o$ and $\mathbf{I}'_o$ instead of being provided by input images directly. Similarly, style feature difference is calculated between $\mathbf{I}'_s$ and $\mathbf{I}_s$, $\mathbf{I}''_c$ and $\mathbf{I}_c$ since the process from $\mathbf{I}_s$ to $\mathbf{I}'_s$ and the process from $\mathbf{I}_c$ to $\mathbf{I}''_c$ involve style transmission. In addition to the feature loss caused by the content and style transmission, L_{multi} also calculates the feature difference among multi-restored images. As shown in Figure 2, our RAST architecture involves a multi-restoration process. For each input image, two restored images are produced through content transmission and style transmission, respectively. Through the visualization, we observe that the restored image via content transmission is slightly different from the restored image via style transmission. Thus, we calculate the feature difference between $\mathbf{I}'_c$ and $\mathbf{I}''_c$, $\mathbf{I}'_s$ and $\mathbf{I}''_s$. Considering the difference caused by different transmission processes, our architecture can further improve the accuracy of content transmission and style transmission. We employ a hyper-parameter α to provide a different weight to this feature difference caused by different transmission methods. As shown in Figure 4, higher α values can lead to better feature consistency. Differing from standard cycle loss [55], the overall multi-restoration loss $\mathcal{L}_{multi}$ can calculate the feature difference caused by transmission while also taking into account the error caused by different transmission methods, which can be used to replace the existing content loss and style loss in our architecture. It is worth mentioning that the loss of content features and style features are given equal weights.

In addition to the multi-restoration loss, we also design the style difference loss. To avoid the stylized images converging to content images, we design the style difference loss $\mathcal{L}_{diff}$ to maximize the style difference between content images and stylized images. Specifically, we maximize the style feature difference between $\mathbf{I}_o$ and $\mathbf{I}_c$, $\mathbf{I}'_o$ and $\mathbf{I}_s$. The equation of style difference loss $\mathcal{L}_{diff}$ is shown below.

$$\mathcal{L}_{diff} = f_s(\mathbf{I}_o, \mathbf{I}_c) + f_s(\mathbf{I}'_o, \mathbf{I}_s). \tag{7}$$

The combination of style difference loss $\mathcal{L}_{diff}$ and $\mathcal{L}_{multi}$ can promise the embedding of style features from style images. Style difference loss $\mathcal{L}_{diff}$ can ensure the style features in stylized images are different from the style features in content images. Also, the calculation of style consistency in $\mathcal{L}_{multi}$ can guarantee that the embedded style features originate from style images. The results of training with different $\mathcal{L}_{diff}$ weights are shown in Figure 5, where higher weights of $\mathcal{L}_{diff}$ can lead to a richer style in stylized images. Since the purpose of style difference loss is to maximize the difference of style features, we expect an increasing value during the training process as shown in Figure 3. To ensure convergence of the final loss function, we take the reciprocal value of style difference loss $\mathcal{L}_{diff}$ in the final loss.

In addition to the above proposed loss, we also include three existing loss functions: identity loss [37], contrastive

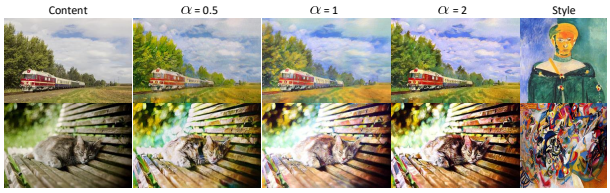

Figure 4: Results of training with different hyper-parameters of multi-restoration loss. The other weights are the same as in Equation 8. The Batch Size is set to 4 for 60000 iterations.

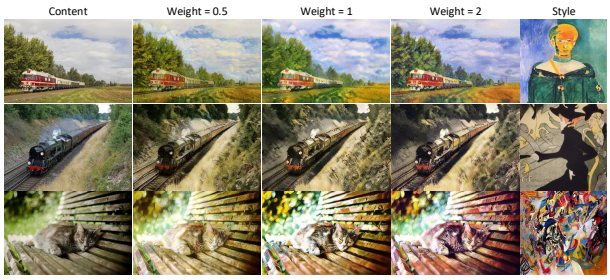

Figure 5: Results of training with different weights of style difference loss. The rest of the weights are the same as in Equation 8. The Batch size is set to 4 for 75000 iterations.

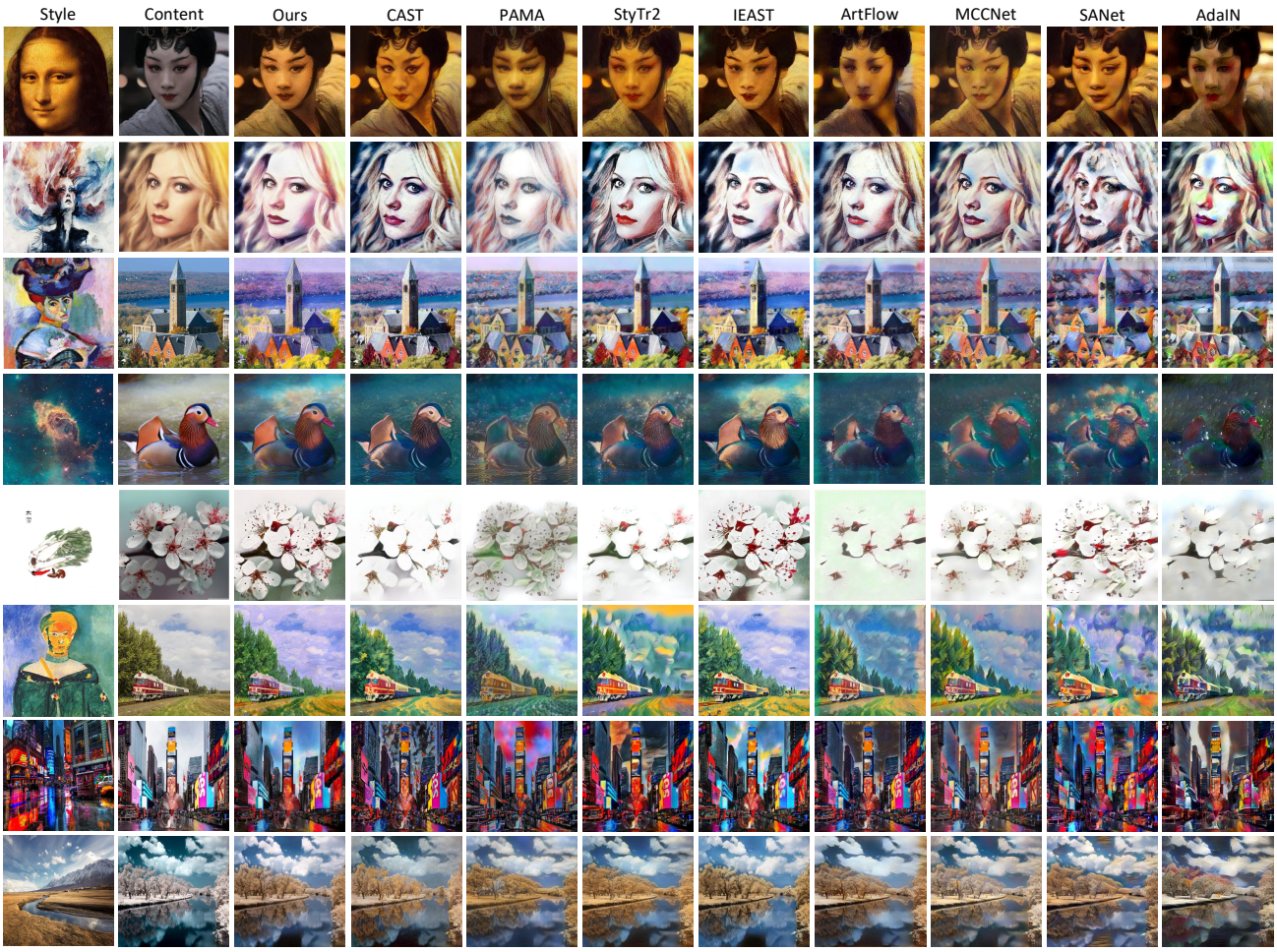

Figure 6: Stylized results for comparisons. The 1^{st} and 2^{nd} columns represent the style images and content images, respectively. The 3^{rd} to 11^{th} columns are stylized results from the proposed architecture and state-of-the-art approaches. The 1^{st} to 6^{th} rows reveal the artistic style transfer. Photo-realistic style transfer are shown in the 7^{th} and 8^{th} rows.

loss [2] and external-adversarial loss [2]. Identity loss $\mathcal{L}_{identity}$ was proposed by SANet [37] to achieve the identity mapping, where the content and the style originated from the same image. As proved by SANet, identity loss can optimize content preservation and improve the accuracy of style embedding. In addition, we include contrastive learning by utilizing contrastive loss $\mathcal{L}_{contra}$ [2]. Taking batch size of 4 as an instance, each content image matches two different style images so that two results with the same content information can be obtained. Similarly, each style image can produce two results sharing the same style. By evaluating the feature difference of associated results, the contrastive loss can learn stylization-to-stylization relations. Finally, we involve the internal-external learning by utilizing external-adversarial loss $\mathcal{L}_{adv}$ [2], which can learn human-aware style information. Differing from IEAST [2], we include two multi-scale discriminators[48] $\mathcal{D}_1$ and $\mathcal{D}_2$, which can deal with realistic-to-artistic and artistic-to-realistic processes respectively.

The final loss function $\mathcal{L}_{final}$ can be summarized as below, where the loss weights are set to $\lambda_1 = 2$, $\lambda_2 = 2$, $\lambda_3 = 1$, $\lambda_4 = 5$ and $\lambda_5 = 0.3$.

$$\begin{aligned}\mathcal{L}_{final} = \lambda_1\mathcal{L}_{multi} + \lambda_2\mathcal{L}_{diff}^{-1} + \\ \lambda_3\mathcal{L}_{identity} + \lambda_4\mathcal{L}_{adv} + \lambda_5\mathcal{L}_{contra}.\end{aligned} \tag{8}$$

Note that we replace the existing content loss and style loss by the proposed multi-restoration loss $\mathcal{L}_{multi}$ and style difference loss $\mathcal{L}_{diff}$ so that the restorable style transfer can be achieved. Utilizing the proposed loss functions, the effectiveness of the proposed RAST architecture can be promised. In addition to the guidance of two proposed loss functions, three state-of-the-art loss functions can further optimize the style transfer performance from different aspects.

Method	Ours	CAST	PAMA	StyTr^2	IEAST	ArtFlow	MCCNet	SANet	AdaIN
LPIPS ($\mathbf{I}_c$,$\mathbf{I}'_c$) ↓	**0.187**	0.324	0.266	0.260	0.328	0.411	0.320	0.455	0.434
LPIPS ($\mathbf{I}_s$,$\mathbf{I}'_s$) ↓	0.250	0.423	0.305	0.312	0.309	0.184	**0.178**	0.436	0.454
$\mathcal{L}_c$ ($\mathbf{I}_c$,$\mathbf{I}'_c$) ↓	**2.021**	4.203	3.594	3.646	4.786	5.549	5.406	7.325	6.952
$\mathcal{L}_s$ ($\mathbf{I}_s$,$\mathbf{I}'_s$) ↓	**0.787**	2.624	1.857	1.222	1.052	0.985	1.334	1.272	1.927
Inference Time (ms/img) ↓	**6**	8	10	538	**6**	280	9	**6**	12

Table 1: Quantitative comparisons with state-of-the-art approaches.

4. Experimental results

To demonstrate the style transfer performance of the RAST architecture, we compare with eight state-of-the-art approaches, including CAST [54], PAMA [36], StyTr^2 [6], IEAST [2], Artflow [1], MCCNet [5], SANet [37] and AdaIN [21]. Qualitative and quantitative comparison results are organized in Sections 4.2 and 4.3, respectively.

4.1. Implementation details

Our proposed RAST architecture is trained with MS-COCO [32] as the content dataset and WikiArt [24] as the style dataset. In the training phase, the smaller dimension of training images are rescaled to 512 and we randomly crop to 256×256 patches. We adopt the Adam optimizer [26] with the learning rate set to 0.0001. The batch size is set to 8 for 160000 iterations on a single Nvidia RTX A6000 GPU. For the testing stage, we randomly choose 10000 content images and 10000 style images from the test sets of MS-COCO and WikiArt, respectively. We resize images to 512×512 so that the evaluation metric can be applied to the same size. The testing stage is finished on a single Nvidia GeForce RTX 2080 GPU. In addition, we utilize widely-used image pairs for visualization involving both artistic style transfer and photo-realistic style transfer. Note that our architecture can deal with testing images of any size.

4.2. Qualitative Comparisons

In Figure 6, we show the qualitative results of our RAST method against eight state-of-the-art approaches. To demonstrate the arbitrary style transfer performance, we include comparisons of both artistic style transfer (the 1^{st}-6^{th} rows) and photo-realistic style transfer (the 7^{th} and 8^{th} rows). To ensure the diversity of experiments, we utilize different types of content images involving portrait, architecture, animal, still life, and landscape. We also adopt style images with various styles. From the comparisons, we observe that AdaIN [21] sometimes generates unreliable results with weak preservation on local details (the 1^{st}, 2^{nd}, 4^{th}, 5^{th} and 6^{th} rows) and produces undesired patterns (the 2^{nd}, 5^{th}, 6^{th} , 7^{th} and 8^{th} rows). SANet [37] brings repetitive patterns (the 1^{st}, 2^{nd}, 3^{rd}, 4^{th}, 6^{th} and 8^{th} rows) and visual artifacts (the 3^{rd} and 7^{th} row). Similar to SANet, MCCNet [5] also suffers from the halation artifact around contours (the 3^{rd}, 4^{th}, 6^{th} and 7^{th} rows). Artflow [1] produces unexpected patterns near the edge of the images (the 1^{st}, 2^{nd}, 3^{rd}, 4^{th}, 6^{th} and 8^{th} rows). IEAST [2] applies repeated patterns in stylized images (the 2^{nd}, 3^{rd} and 6^{th} rows) and faces color distortion problem (the 1^{st}, 5^{th}, 7^{th} and 8^{th} rows).

The above problems have been partially addressed by recent approaches. However, recently proposed methods still suffer from some existing problems, such as the deficiency of delicate patterns and poor content preservation. Insufficient content preservation can lead to the loss of content details and can make the results blurred. From Figure 6, we can observe that StyTr^2 [6] sometimes fails to preserve the content information (the 3^{rd}, 5^{th}, 6^{th} and 7^{th} rows). It also suffers from the color distortion problem (the 6^{th} and 8^{th} rows). PAMA is still not free from the content preservation headache (the 1^{st} and 7^{th} rows). Also, it suffers from insufficient style embedding (the 2^{nd}, 6^{th} and 8^{th} rows). Similarly, CAST [54] cannot produce content-preserved results (the 1^{st}, 5^{th} and 7^{th} rows), which leads to content changes and makes the results blurred. In addition, it faces the color distortion issue for photo-realistic style transfer (the 1^{st}, 6^{th} and 8^{th} row). By contrast, our RAST architecture achieves restorable arbitrary style transfer via multiple restorations, which involves the transmission of both content information and style information. Thus, RAST can achieve superior content preservation performance with promising style embedding performance compared to other state-of-the-art methods.

4.3. Quantitative Comparisons

In addition to qualitative comparisons, we also involve quantitative comparisons. In state-of-the-art methods, there is no golden metric to evaluate the style transfer performance between input images and stylized images. The reason is that the style information between content images $\mathbf{I}_c$ and stylized images $\mathbf{I}_o$ are different. The content information between style images $\mathbf{I}_s$ and stylized images $\mathbf{I}_o$ are also distinct. Due to this limitation, we propose a novel approach to measure the content preservation performance and style consistency performance indirectly by evaluating

Method	Ours	CAST	PAMA	StyTr^2	IEAST	ArtFlow	MCCNet	SANet	AdaIN
Content Preservation Score ↑	**4.374**	3.134	3.110	3.362	3.444	2.854	3.064	2.624	2.174
Style Consistency Score ↑	**3.570**	2.962	3.080	3.188	3.162	2.770	3.062	3.076	2.416
Preference Score ↑	**3.860**	2.982	2.998	3.210	3.332	2.672	3.054	2.810	2.186

Table 2: User study results.

the feature difference between input images $\mathbf{I}_{c|s}$ and restored images $\mathbf{I}'_{c|s}$. Specifically, we measure the content preservation performance by evaluating the feature difference between content images $\mathbf{I}_c$ and restored content images $\mathbf{I}'_c$ since $\mathbf{I}_c$ and $\mathbf{I}'_c$ share the same style features. Also, the content information in $\mathbf{I}'_c$ is transmitted from $\mathbf{I}_c$ following the $\mathbf{I}_c \rightarrow \mathbf{I}_o \rightarrow \mathbf{I}'_c$ process, which only involves content transmission. Similarly, style consistency performance is measured by evaluating the difference between style images $\mathbf{I}_s$ and restored style images $\mathbf{I}'_s$, involving the same content features. For the evaluation, we adopt Learned Perceptual Image Patch Similarity (LPIPS) [53] and perceptual loss [22] as evaluation metrics. The evaluation results of the testing set (10000 image pairs) are shown in Table 1.

From the results, we can observe that the proposed RAST framework can achieve superior content preservation performance for both LPIPS (the 2^{nd} row) and content loss $\mathcal{L}_c$ (the 4^{th} row) evaluation metrics. PAMA and Stytr^2 methods can also achieve promising content preservation. The above results demonstrate that the multi-restoration training indeed improves content consistency. The proposed framework restricts the content changes caused by style information, which can avoid the content leak issue. In addition to content preservation, RAST can also achieve promising style consistency performance, which ranks first for style loss $\mathcal{L}_s$ (the 5^{th} row) and ranks third for LPIPS (the 3^{rd} row). From the results, we can see that the combination of style difference loss $\mathcal{L}_{diff}$ and style transmission loss $\mathcal{L}_{trans-s}$ can effectively make our framework achieve promising style consistency. In addition, the ArtFlow approach can also achieve outstanding style consistency performance with second ranking for both style loss and LPIPS metric. In addition to content preservation and style consistency, we also compare the inference time with state-of-the-art approaches on the testing set. The time is calculated from input images entering the model to returning stylized results, excluding the process of loading and saving images. From Table 1 (the 5^{th} row), we can observe that SANet can achieve outstanding style transfer speed. Our RAST framework and IEAST method can also obtain similar results by utilizing SANet as the backbone.

User Study. To further demonstrate the performance of the proposed framework, we design a user study, which includes 20 sections. For each section, we show participants a different image pair with labeling content and style image. We present the stylized results of the proposed method and eight state-of-the-art approaches in a nine-square grid. The results are arranged randomly in the grid and the names of methods are hidden from the participants. For each stylized result, participants are asked to grade the content preservation performance, the style consistency performance, and the overall performance, separately. The grading scale is set from 1 (bad) to 5 (good). This way, participants can decide the score for the current method after comparing the results of the remaining eight methods. We collect 13500 scores in total from 25 participants. The average scores are shown in Table 2, where the preference score represents the overall performance. Comparing with Table 1, we can recognize that there are two outliers in the user study, IEAST and CAST. The reason is that IEAST can result in some tiny unexpected patterns, like the human eyes, which may not be noticed during the user study process; however, these can result in low scores. In addition, the CAST model produces blurred results, which makes the evaluation results lower. However, this situation may not capture attention during the user study. Overall, we can conclude that our proposed approach can achieve comparable style transfer performance.

5. Conclusion

We proposed the Restorable Arbitrary Style Transfer (RAST) architecture, which handles the content leak problem from the perspective of image restoration. Through multi-restorations, we realized the transmission of both content and style information. Unlike previous methods that minimize the content and style difference between the input images and the stylized images, we focused on minimizing the difference between the restored images and the input images and maximizing the style difference between the stylized images and the input images, with the motivation of avoiding the interference caused by different content information or style information. Furthermore, two new loss functions including style difference loss and multi-restoration loss were proposed to ensure the effectiveness of the RAST architecture. Comprehensive experiments demonstrated that the proposed RAST can achieve superior style transfer performance with comparable content preservation performance and promising style consistency performance.

References

[1] Jie An, Siyu Huang, Yibing Song, Dejing Dou, Wei Liu, and Jiebo Luo. Artflow: Unbiased image style transfer via reversible neural flows. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 862–871, 2021.

[2] Haibo Chen, Zhizhong Wang, Huiming Zhang, Zhiwen Zuo, Ailin Li, Wei Xing, Dongming Lu, et al. Artistic style transfer with internal-external learning and contrastive learning. *Advances in Neural Information Processing Systems*, 34:26561–26573, 2021.

[3] Tian Qi Chen and Mark Schmidt. Fast patch-based style transfer of arbitrary style. *arXiv preprint arXiv:1612.04337*, 2016.

[4] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8789–8797, 2018.

[5] Yingying Deng, Fan Tang, Weiming Dong, Haibin Huang, Chongyang Ma, and Changsheng Xu. Arbitrary video style transfer via multi-channel correlation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1210–1217, 2021.

[6] Yingying Deng, Fan Tang, Weiming Dong, Chongyang Ma, Xingjia Pan, Lei Wang, and Changsheng Xu. Stytr2: Image style transfer with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11326–11336, 2022.

[7] Yingying Deng, Fan Tang, Weiming Dong, Wen Sun, Feiyue Huang, and Changsheng Xu. Arbitrary style transfer via multi-adaptation network. In *Proceedings of the 28th ACM international conference on multimedia*, pages 2719–2727, 2020.

[8] Laurent Dinh, David Krueger, and Yoshua Bengio. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014.

[9] Vincent Dumoulin, Jonathon Shlens, and Manjunath Kudlur. A learned representation for artistic style. *arXiv preprint arXiv:1610.07629*, 2016.

[10] Michael Elad and Peyman Milanfar. Style transfer via texture synthesis. *IEEE Transactions on Image Processing*, 26(5):2338–2351, 2017.

[11] Jakub Fišer, Ondřej Jamriška, Michal Lukáč, Eli Shechtman, Paul Asente, Jingwan Lu, and Daniel Sỳkora. Stylit: illumination-guided example-based stylization of 3d renderings. *ACM Transactions on Graphics (TOG)*, 35(4):1–11, 2016.

[12] Oriel Frigo, Neus Sabater, Julie Delon, and Pierre Hellier. Split and match: Example-based adaptive patch sampling for unsupervised style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 553–561, 2016.

[13] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2414–2423, 2016.

[14] Bruce Gooch, Erik Reinhard, and Amy Gooch. Human facial illustrations: Creation and psychophysical evaluation. *ACM Transactions on Graphics (TOG)*, 23(1):27–44, 2004.

[15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[16] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[17] Aaron Hertzmann. Painterly rendering with curved brush strokes of multiple sizes. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 453–460, 1998.

[18] Aaron Hertzmann, Charles E Jacobs, Nuria Oliver, Brian Curless, and David H Salesin. Image analogies. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 327–340, 2001.

[19] Jonathan Ho, Xi Chen, Aravind Srinivas, Yan Duan, and Pieter Abbeel. Flow++: Improving flow-based generative models with variational dequantization and architecture design. In *International Conference on Machine Learning*, pages 2722–2730. PMLR, 2019.

[20] Kibeom Hong, Seogkyu Jeon, Huan Yang, Jianlong Fu, and Hyeran Byun. Domain-aware universal style transfer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14609–14617, 2021.

[21] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE international conference on computer vision*, pages 1501–1510, 2017.

[22] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016.

[23] Minguk Kang and Jaesik Park. Contragan: Contrastive learning for conditional image generation. *Advances in Neural Information Processing Systems*, 33:21357–21369, 2020.

[24] Sergey Karayev, Matthew Trentacoste, Helen Han, Aseem Agarwala, Trevor Darrell, Aaron Hertzmann, and Holger Winnemoeller. Recognizing image style. *arXiv preprint arXiv:1311.3715*, 2013.

[25] Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jung Kwon Lee, and Jiwon Kim. Learning to discover cross-domain relations with generative adversarial networks. In *International conference on machine learning*, pages 1857–1865. PMLR, 2017.

[26] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[27] Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems*, 31, 2018.

[28] Dmytro Kotovenko, Artsiom Sanakoyeu, Sabine Lang, and Bjorn Ommer. Content and style disentanglement for artistic style transfer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4422–4431, 2019.

[29] Chuan Li and Michael Wand. Combining markov random fields and convolutional neural networks for image synthesis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2479–2486, 2016.

[30] Xueting Li, Sifei Liu, Jan Kautz, and Ming-Hsuan Yang. Learning linear transformations for fast image and video style transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3809–3817, 2019.

[31] Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Universal style transfer via feature transforms. *Advances in neural information processing systems*, 30, 2017.

[32] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[33] 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.

[34] Zhi-Song Liu, Li-Wen Wang, Wan-Chi Siu, and Vicky Kalogeiton. Name your style: An arbitrary artist-aware image style transfer. *arXiv preprint arXiv:2202.13562*, 2022.

[35] Ming Lu, Hao Zhao, Anbang Yao, Yurong Chen, Feng Xu, and Li Zhang. A closed-form solution to universal style transfer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5952–5961, 2019.

[36] Xuan Luo, Zhen Han, Lingkang Yang, and Lingling Zhang. Consistent style transfer. *arXiv preprint arXiv:2201.02233*, 2022.

[37] Dae Young Park and Kwang Hee Lee. Arbitrary style transfer with style-attentional networks. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5880–5888, 2019.

[38] Seobin Park, Jinsu Yoo, Donghyeon Cho, Jiwon Kim, and Tae Hyun Kim. Fast adaptation to super-resolution networks via meta-learning. In *European Conference on Computer Vision*, pages 754–769. Springer, 2020.

[39] Taesung Park, Alexei A Efros, Richard Zhang, and Jun-Yan Zhu. Contrastive learning for unpaired image-to-image translation. In *European conference on computer vision*, pages 319–345. Springer, 2020.

[40] Artsiom Sanakoyeu, Dmytro Kotovenko, Sabine Lang, and Bjorn Ommer. A style-aware content loss for real-time hd style transfer. In *proceedings of the European conference on computer vision (ECCV)*, pages 698–714, 2018.

[41] Lu Sheng, Ziyi Lin, Jing Shao, and Xiaogang Wang. Avatar-net: Multi-scale zero-shot style transfer by feature decoration. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8242–8250, 2018.

[42] YiChang Shih, Sylvain Paris, Connelly Barnes, William T Freeman, and Frédo Durand. Style transfer for headshot portraits. 2014.

[43] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[44] Jae Woong Soh, Sunwoo Cho, and Nam Ik Cho. Meta-transfer learning for zero-shot super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3516–3525, 2020.

[45] Carlo Tomasi and Roberto Manduchi. Bilateral filtering for gray and color images. In *Sixth international conference on computer vision (IEEE Cat. No. 98CH36271)*, pages 839–846. IEEE, 1998.

[46] Dmitry Ulyanov, Vadim Lebedev, Andrea Vedaldi, and Victor Lempitsky. Texture networks: Feed-forward synthesis of textures and stylized images. *arXiv preprint arXiv:1603.03417*, 2016.

[47] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[48] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018.

[49] Holger Winnemöller, Sven C Olsen, and Bruce Gooch. Real-time video abstraction. *ACM Transactions On Graphics (TOG)*, 25(3):1221–1226, 2006.

[50] Haiyan Wu, Yanyun Qu, Shaohui Lin, Jian Zhou, Ruizhi Qiao, Zhizhong Zhang, Yuan Xie, and Lizhuang Ma. Contrastive learning for compact single image dehazing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10551–10560, 2021.

[51] Xiaolei Wu, Zhihao Hu, Lu Sheng, and Dong Xu. Styleformer: Real-time arbitrary style transfer via parametric style composition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14618–14627, 2021.

[52] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In *International conference on machine learning*, pages 7354–7363. PMLR, 2019.

[53] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

[54] Yuxin Zhang, Fan Tang, Weiming Dong, Haibin Huang, Chongyang Ma, Tong-Yee Lee, and Changsheng Xu. Domain enhanced arbitrary image style transfer via contrastive learning. *arXiv preprint arXiv:2205.09542*, 2022.

[55] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017.