QuantArt: Quantizing Image Style Transfer Towards High Visual Fidelity

Siyu Huang[1*] Jie An[2*] Donglai Wei[3] Jiebo Luo[2] Hanspeter Pfister[1]
[1]Harvard University [2]University of Rochester [3]Boston College
{huang,pfister}@seas.harvard.com {jan6,jluo}@cs.rochester.edu donglai.wei@bc.edu

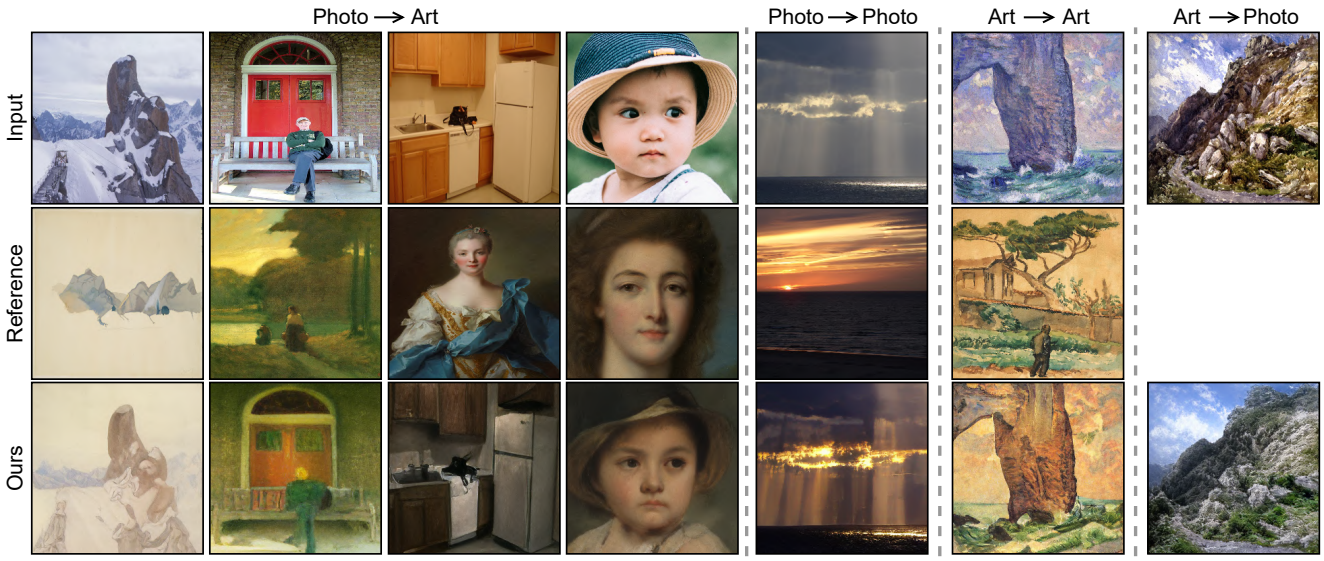


Figure 1. The proposed QuantArt method produces impressive arbitrary style transfer results on various image style transfer tasks.

Abstract

The mechanism of existing style transfer algorithms is by minimizing a hybrid loss function to push the generated image toward high similarities in both content and style. However, this type of approach cannot guarantee visual fidelity, i.e., the generated artworks should be indistinguishable from real ones. In this paper, we devise a new style transfer framework called QuantArt for high visual-fidelity stylization. QuantArt pushes the latent representation of the generated artwork toward the centroids of the real artwork distribution with vector quantization. By fusing the quantized and continuous latent representations, QuantArt allows flexible control over the generated artworks in terms of content preservation, style similarity, and visual fidelity. Experiments on various style transfer settings show that our QuantArt framework achieves significantly higher visual fidelity compared with the existing style transfer methods.

1. Introduction

Image style transfer aims at transferring the artistic style of a reference image to a content image, where the output image should have the style (*e.g.*, colors, textures, strokes, and tones) of the reference and the content information of the content image. Great advances [4, 5, 13, 14, 39, 40, 63] have been made in the area of image style transfer, where the arbitrary style transfer (AST) has become one of the main research focuses. Given a trained model, AST algorithms [8, 26, 33] can perform style transfer on arbitrary unseen content-style pairs in a zero-shot manner, such that it enables more practical applications[1].

Existing AST algorithms, including the statistics-based methods [1, 26, 35, 60] and the patch-based methods [6, 47], deliver remarkable style transfer results by matching the artistic style information of the stylized image and the style reference. However, taking the high-fidelity artwork generation as the ultimate goal of image style transfer, all existing methods can still be improved since there are few mechanisms to guarantee a high artistic fidelity of the stylized image. A few existing work [3, 59] accommodate the adversarial loss [15] into the style transfer framework to enhance the image quality. However, the performance improvement is hindered by the heterogeneous optimization objectives of high image quality and faithful image stylization.

In this work, we introduce visual fidelity as a new eval-

[1]The codes of this paper are available at https://github.com/siyuhuang/QuantArt

uation dimension of style transfer. It is formulated as the similarity between the stylized image and the real artwork dataset, and it is orthogonal to the two widely studied evaluation dimensions including style similarity and content preservation. Motivated by the vector-quantized image representation [11, 45, 52], if the latent feature of generation is closer to one of the cluster centers in the real distribution, it is harder for humans to distinguish it from the real images, *i.e*., having better visual fidelity. We propose to learn an artwork codebook, *i.e*., a global dictionary, to save the discrete cluster centers of all artworks. The continuous representations of images are converted to the discrete encodings in the artwork codebook via vector quantization, ensuring that it is not only close to the given style reference but also close to one of the learned cluster centers in the real distribution.

We further propose a framework called Quantizing Artistic Style Transfer (QuantArt) to achieve flexible control of the three evaluation dimensions mentioned above. QuantArt first extracts both content and style features using separate encoders, respectively. Next, it applies vector quantization to both content and style features to fetch discrete codes in the learned codebooks. Then, the content and style codes are transferred to the stylized feature with a specially designed feature style transfer module called Style-Guided Attention. Before feeding into the decoder, the stylized feature is quantized again with the artwork codebook, ensuring a high visual-fidelity stylization by approaching the cluster centers of the real artwork distribution. By fusing the continuous and quantized stylized features with the content features before the decoder, QuantArt allows users to arbitrarily trade off between the style similarity, visual fidelity, and content reservation of the style transfer results. In the experiments, the proposed method significantly increases the visual fidelity of generations in various image style transfer settings including photo-to-art, art-to-art, photo-to-photo, and art-to-photo (see Fig. 1). The contribution of the proposed method can be summarized as follows:

- We define *visual fidelity* as a new evaluation dimension of style transfer and propose a high visual-fidelity style transfer algorithm based on vector quantization.
- We design a framework based on both discrete and continuous style transfer architectures, which allow users to flexibly control style similarity, content preservation, and visual fidelity of the stylization result.
- The extensive experiments demonstrate that our method achieves higher visual fidelity and comparable style similarity with respect to the state-of-the-art style transfer methods.

2. Related Work

Image style transfer. Image style transfer is a challenging topic that has been studied for decades [20, 27, 34, 55]. Gatys et al. [13] first adopted convolutional neural networks (CNNs) for image style transfer by matching the statistics of content and style features extracted by CNNs. Among the neural style transfer (NST) algorithms [13, 14, 40], arbitrary style transfer (AST) [5, 6, 25, 26, 63] has drawn much attention from researchers in recent years due to its zero-shot image stylization manner.

Existing AST algorithms can be generally categorized into two types: the statistics-based methods [10, 51] and the patch-based methods [12, 16, 37, 54]. The statistics-based methods minimize the distance of global feature statistics between the generation and the style image, where the feature statistics can be Gram matrices [13, 14], histograms [19, 46], wavelets [44, 60], mean-std statistics [9, 26], and covariance matrices [35]. The statistics-based methods are highly efficient in capturing the global style information. The patch-based methods search for appropriate patches in style images to reconstruct the transferred images. StyleSwap [6] and Avatar-Net [47] are two typical patch-based methods, which iteratively swap the content feature patches with the nearest-matched feature patches of the reference image. Compared to the statistics-based approaches, the patch-based methods produce better texture synthesis quality as they directly adopt patches from style images. However, it requires the content and style images to have similar local semantic structures.

In general, the existing AST algorithms aim at matching the styles of the generation and the reference, where the visual fidelity of the generation cannot be guaranteed. In this work, we introduce visual fidelity as a new evaluation dimension of style transfer and propose a novel AST framework, *i.e*., QuantArt, to enhance the visual fidelity of generations via pushing the latent feature toward the centroids of artwork distributions. QuantArt can also alleviate the stylization artifact issue, as the outlier styles are replaced with the nearest style centroid in the latent space.

Photorealistic style transfer. The proposed method can also handle the photorealistic style transfer task. The artistic style transfer algorithms often fail for this task, since the stylizated image would contain warping distortions that are redundant for the photorealism scenario. Motivated by this, several methods [2, 23, 57, 58] have been specially designed. Luan et al. [40] first introduced a locally affine transformation as the photorealism loss term. PhotoWCT [36] proposes a closed-form post-processing algorithm to further smooth the stylized results. WCT^2 [60] eliminates the post-processing stage via the Wavelet Corrected Transfer module. Distinct from these approaches, our method does not require to impose any additional regularization or post-processing step for photorealistic style transfer, thanks to the highly effective quantized image representation.

Vector-quantized image representation. The vector-quantized generative models [11, 61] are originally devel-

oped for compact yet effective image modeling. VQ-VAE [45, 52] devises a vector-quantized autoencoder to represent an image with a set of discrete tokens. VQ-GAN [11] improves VQVAE with the adversarial learning scheme [15]. This work adopts vector quantization as an efficient learnable implementation of artwork distribution clustering. The vector quantization pushes the latent feature to be closer to the real artwork distribution, resulting in higher visual-fidelity image stylization.

3. Visual Fidelity for Image Style Transfer

Image style transfer aims to transfer a content image c, *e.g.*, a photo, to a stylized image y by a given style reference s, *e.g.*, an artwork image. Existing image style transfer algorithms mainly focus on either content preservation or style similarity between the generated and input images. However, we argue that the visual realness of generated images is also a vital factor for style transfer performance. We formulate the three critical performance indicators of image style transfer as

- *Style fidelity*: The style similarity between y and s, which is often evaluated by the Gram matrix [13, 34].
- *Content fidelity*: The content similarity between y and c, which is often evaluated by the perceptual loss [28] or the LPIPS distance [62].
- *Visual fidelity*: The realness of the generated image y. Since all real artwork images belong to a distribution $\mathcal{T}$, a generated image y tends to have a higher visual fidelity if it is closer to distribution $\mathcal{T}$.

Many existing literature [26, 32, 39, 42, 47] shows that there is a trade-off between the style fidelity and content fidelity in image style transfer. Analogous to this, there is a trade-off between visual fidelity and style fidelity. As an example shown in the bottom of Fig. 2, the neural style transfer algorithm ($\alpha = 0$) faithfully renders the style textures of the tree in the reference image. However, it lowers the visual fidelity of the generated image.

To increase the visual fidelity of neural style transfer, in this work we propose a novel framework named QuantArt to learn to cluster the artwork distribution $\mathcal{T}$ in the representation space, where the centroids of all clusters form an *artwork codebook* $\mathcal{Z}_{art}$. When making inferences, we replace the feature map at each position with its nearest centroid in $\mathcal{Z}_{art}$. In this way, the feature of the generated artwork is pushed closer to the real artwork distribution, thus leading to better visual fidelity. This nearest centroid search and replacing operation is implemented by the vector quantization used in [11, 45, 52]. Besides, the idea of pushing the latent feature closer to the centroid of real distribution is partially motivated by the low-temperature sampling used in GANs [15, 24, 30, 66] and diffusion models [17, 22]. As illustrated in the top right part of Fig. 2, one can increase the visual fidelity by pushing the generation y to be close to one of the centroids of the artwork distribution $\mathcal{T}$ but far away from the reference s. We introduce more details of QuantArt in the following Section 4.

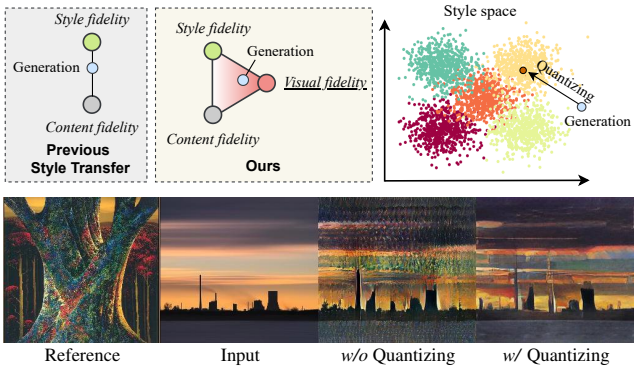


Figure 2. **(Top left)** This work introduces *visual fidelity* as an orthogonal evaluation dimension to content fidelity and style fidelity. **(Top right)** Our style transfer method enables a trade-off between style and visual fidelities via quantizing the generation in the style space. **(Bottom)** An example of image style transfer *w/o* and *w/* latent feature quantizing, respectively.

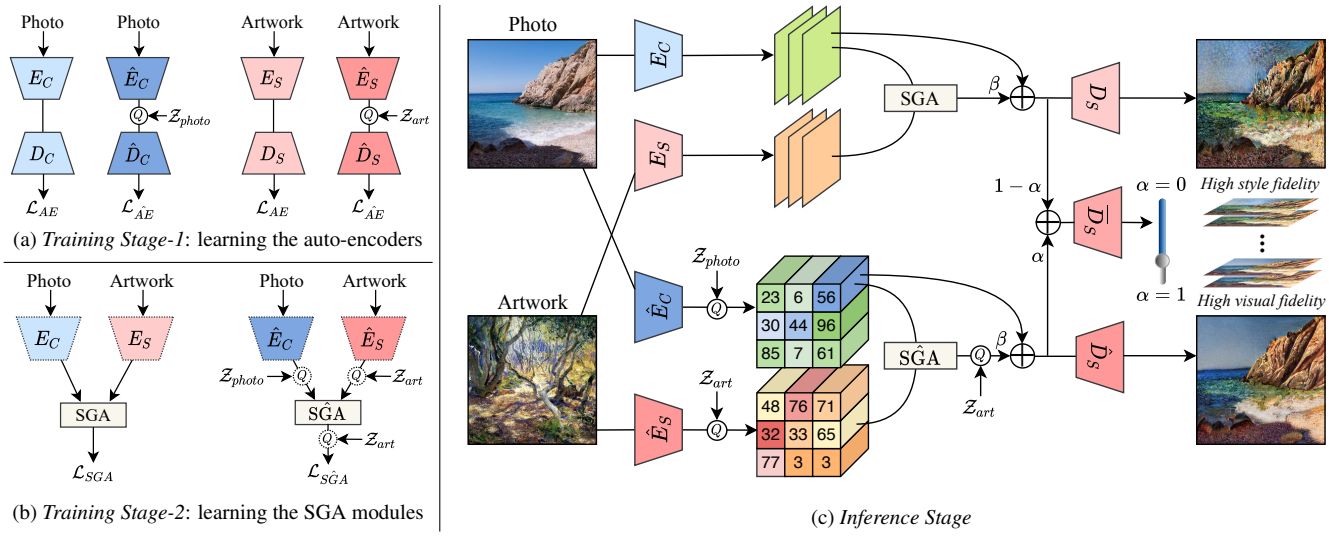
4. Our Approach

4.1. Framework Overview

In this work, we propose a novel framework dubbed Quantizing Artistic Style Transfer (QuantArt) to enhance the visual fidelity of generations in image style transfer. As illustrated in Fig. 3, QuantArt adopts four auto-encoders to extract the continuous/quantized features of photo/artwork images respectively, two codebooks to store the cluster centers of photo and artwork distributions, and two SGA modules to transfer the styles of feature representations. The training of QuantArt consists of two stages. In the first training stage (see Fig. 3(a)), we learn the auto-encoders and the codebooks by reconstructing the photo and artwork images, respectively. In the second training stage (see Fig. 3(b)), we train the SGA modules based on the extracted feature representations. In the inference phase, as illustrated in Fig. 3(c), users can easily trade off the style and visual fidelity of generations by adjusting the discretization level $\alpha \in [0, 1]$ between the SGA outputs. In the following, we discuss more details of QuantArt.

4.2. Learning Auto-Encoders and Codebooks

We first extract the features of the content image c and the style reference s with two convolutional encoders E_C and E_S, then decode the features back into images c_{rec} and s_{rec} with two convolutional decoders D_C and D_S, respec-



(a) *Training Stage-1*: learning the auto-encoders

(b) *Training Stage-2*: learning the SGA modules

(c) *Inference Stage*

Figure 3. The training and inference pipelines of QuantArt. **(a)** The first training stage, where we learn the auto-encoders and codebooks for photo and artwork images, respectively. Ⓠ denotes the vector quantization operator in Eq. 5. **(b)** The second training stage, where we learn the SGA-based style transfer modules. The dashed lines denote the parameters of encoders and the codebooks are frozen in this stage. **(c)** In the inference phase, one can trade off the content, style and visual fidelities by simply adjusting the parameters $\alpha, \beta \in [0, 1]$.

tively as

$$c_{rec} = D_C(E_C(c)), \quad s_{rec} = D_S(E_S(s)). \tag{1}$$

This is distinct from the general neural style transfer methods, which use the VGG [48] or ResNet [18] network pre-trained on natural image datasets (*e.g.*, ImageNet [7]) as the image encoder to extract features of both content and style images. To optimize encoder E_C and decoder D_C, the reconstruction loss is

$$\mathcal{L}_{AE}(E_C, D_C) = ||c_{rec} - c|| + \mathcal{L}_{adv}(E_C, D_C, \mathbb{D}_C), \tag{2}$$

where $\mathcal{L}_{adv}$ is the adversarial training loss and $\mathbb{D}_C$ is the corresponding discriminator network,

$$\mathcal{L}_{adv}(E_C, D_C, \mathbb{D}_C) = \log \mathbb{D}_C(c) + \log(1 - \mathbb{D}_C(c_{rec})). \tag{3}$$

E_S and D_S are also optimized by the reconstruction loss $\mathcal{L}_{AE}(E_S, D_S)$ as formulated in Eq. 2.

Next, we build two codebooks $\mathcal{Z}_{photo}, \mathcal{Z}_{art} \in \mathbb{R}^{N \times d}$ to model the distributions of the photo dataset and artwork dataset, where N is the number of entries in the codebook and d is the dimension of each entry. To enable a better representation performance of the quantized features, we use two extra encoders to extract the features, respectively as

$$z_c = \hat{E}_C(c), \quad z_s = \hat{E}_S(s). \tag{4}$$

We then apply vector quantization [52] to the latent features to get the quantized features $\hat{z}_c$ and $\hat{z}_s$, where the vector quantization operator $Q_{\mathcal{Z}}(z)$ is formulated as

$$Q_{\mathcal{Z}}(z) := \underset{\mathbf{z}_k \in \mathcal{Z}}{\arg\min} ||z - \mathbf{z}_k||, \tag{5}$$

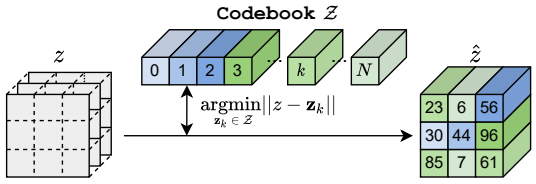


Figure 4. The vector quantization operator $Q_{\mathcal{Z}}(z)$ in Eq. 5.

where $\mathbf{z}_k$ is the k-th entry in the codebook $\mathcal{Z}$. As illustrated in Fig. 4, z is replaced with the nearest neighbour entry in the codebook $\mathcal{Z}$ via $Q_{\mathcal{Z}}(z)$. The quantized features $\hat{z}_c$ and $\hat{z}_s$ are collected from codebooks $\mathcal{Z}_{photo}$ and $\mathcal{Z}_{art}$ as

$$\hat{z}_c = Q_{\mathcal{Z}_{photo}}(z_c), \quad \hat{z}_s = Q_{\mathcal{Z}_{art}}(z_s), \tag{6}$$

The quantized features $\hat{z}_c$ and $\hat{z}_s$ are decoded into images $\hat{c}_{rec}$ and $\hat{s}_{rec}$ via the decoders $\hat{D}_C$ and $\hat{D}_S$, as

$$\hat{c}_{rec} = \hat{D}_C(\hat{z}_c), \quad \hat{s}_{rec} = \hat{D}_S(\hat{z}_s). \tag{7}$$

By following [52], we optimize the codebook $\mathcal{Z}_{photo}$ jointly with the reconstruction loss in Eq. 2. The reconstruction loss for encoder $\hat{E}_C$, decoder $\hat{D}_C$, and codebook $\mathcal{Z}_{photo}$ is

$$\mathcal{L}_{\hat{AE}}(\hat{E}_C, \hat{D}_C, \mathcal{Z}_{photo}) = \mathcal{L}_{AE}(\hat{E}_C, \hat{D}_C) + ||\mathtt{sg}[z_c] - \hat{z}_c|| + ||\mathtt{sg}[\hat{z}_c] - z_c||, \tag{8}$$

where $\mathtt{sg}[\cdot]$ indicates the stop gradient operator. The second term in Eq. 8 optimizes the codebook $\mathcal{Z}_{photo}$, while, the third term in Eq. 8 forces the latent feature z_c to be close to the nearest neighbor entry in $\mathcal{Z}_{photo}$. $\hat{E}_S$, $\hat{D}_S$, and $\mathcal{Z}_{art}$ are

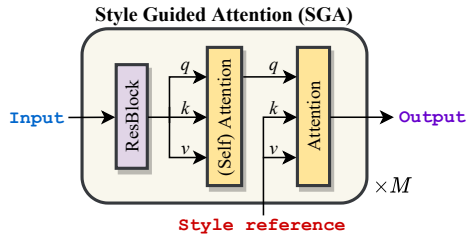


Figure 5. The SGA module proposed for feature style transfer.

also optimized by the loss function $\mathcal{L}_{\hat{AE}}(\hat{E}_S, \hat{D}_S, \mathcal{Z}_{art})$ as formulated in Eq. 8.

After training the Stage-1 models, we can compute the continuous features and quantized features for the content and style images. In the following Section 4.3, we discuss how to perform feature-level style transfer based on the extracted features.

4.3. Style Transfer with Style Guided Attention

To perform effective style transfer for both the continuous and quantized features, we propose a feature-level style transfer module dubbed style-guided attention (SGA). Fig. 5 shows an illustration of the SGA module. The module takes the content feature $z_c \in \mathbb{R}^d$ and style reference feature $z_s \in \mathbb{R}^d$ as inputs, then outputs the stylized feature vector $z_y \in \mathbb{R}^d$. It consists of three blocks including a ResBlock used in ResNet-18 [18] and two attention blocks [53] each with a residual connection. The attention block accepts the query q, key k, and value v as inputs,

$$\text{Attn}(q, k, v) = \text{softmax}(f_q(q)f_k(k))f_v(v) + q, \quad (9)$$

where f_q, f_k, f_v are the embedding layers. The first attention block is a self-attention block that all its input heads q, k, v accept the transformed content feature $\tilde{z}_c = \text{ResBlock}(z_c)$. The second attention block takes $\tilde{z}_c$ as q, and the style reference feature z_s as k and v. The output of the SGA module is formulated as

$$z_y = \text{SGA}(z_c, z_s) = \text{Attn}(\text{Attn}(\tilde{z}_c, \tilde{z}_c, \tilde{z}_c), z_s, z_s). \quad (10)$$

Compared with the existing attention-based style transfer modules [39, 42], our SGA module adopts an additional self-attention block to model the global information of the quantized codes. Our attention block also enables a residual connection between the input and output to better preserve the content details. In practice, we repeat the SGA module for M times to achieve a better style transfer performance.

The objective function for $\text{SGA}(z_c, z_s)$ includes the content loss $\mathcal{L}_{content}$, style loss $\mathcal{L}_{style}$, and adversarial training loss $\mathcal{L}_{featadv}$

$$\mathcal{L}_{SGA} = \mathcal{L}_{content} + \mathcal{L}_{style} + \mathcal{L}_{featadv} \quad (11)$$

where

$$\mathcal{L}_{content} = ||z_y - z_c||, \quad (12)$$

$$\mathcal{L}_{style} = ||\mu(z_y) - \mu(z_s)|| + ||\sigma(z_y) - \sigma(z_s)||, \quad (13)$$

$$\mathcal{L}_{featadv} = \log \mathbb{D}_{SGA}(z_s) + \log(1 - \mathbb{D}_{SGA}(z_y)). \quad (14)$$

$\mu(\cdot)$ and $\sigma(\cdot)$ denote the channel-wise mean and standard deviation of feature maps, respectively. Note that since the encoders and decoders are not optimized in Stage-2, the content and style losses can be directly computed on the features without using an extra pre-trained network to extract features [13, 26, 28]. The adversarial loss $\mathcal{L}_{featadv}$ is also computed on the features, and it forces the SGA output to be more close to the distribution of style reference features.

We adopt another SGA module to transfer the quantized features $\hat{z}_c$ and $\hat{z}_s$, as

$$\hat{z}_y = Q_{\mathcal{Z}_{art}}(\hat{\text{SGA}}(\hat{z}_c, \hat{z}_s)). \quad (15)$$

$\hat{\text{SGA}}(\cdot,\cdot)$ and $\text{SGA}(\cdot,\cdot)$ have the same network architecture but different parameters. The output of the SGA module is further quantized by the art codebook $Q_{\mathcal{Z}_{art}}$ to ensure that the output is on the latent space of the decoder $\hat{D}_S$. The optimization objective of $\hat{\text{SGA}}$ follows Eq. 11 with an additional codebook loss, as

$$\mathcal{L}_{\hat{SGA}} = \mathcal{L}_{SGA} + ||\text{sg}[\hat{z}_y] - z_y]|| \quad (16)$$

4.4. Inference

Fig. 3(c) illustrates the inference procedure of QuantArt framework. We first extract the continuous and quantized features of input photo and artwork images using the corresponding encoders. Then, the features are transformed to the stylized continuous feature z_y and stylized quantized feature $\hat{z}_y$ by using the corresponding SGA modules. To trade off between the content, style reference, and visual fidelity, we have

$$z_{test} = \oplus_\alpha(\oplus_\beta(\hat{z}_y, \hat{z}_c), \oplus_\beta(z_y, z_c)), \quad (17)$$

where z_c and $\hat{z}_c$ are the content features, and $\oplus$ is the weighted sum operator as $\oplus_p(a, b) = pa + (1-p)b$. z_{test} is decoded into a stylized image with a fused decoder $\bar{D}_S = \oplus_\alpha(\hat{D}_S, D_S)$. β controls the style fidelity, and a larger α results in higher visual fidelity. In practice, a good trade-off of the fidelity terms depends on both the input images and individual users' preferences. QuantArt(α, β) provides a simple and easy-to-understand handle grip for users to adjust the style transfer results.

Extension to more style transfer tasks. In Section 4.3, we take the artistic image style transfer task as an example to discuss the learning of SGA modules. In addition to artistic style transfer, QuantArt can be applied to other image style transfer tasks such as artwork-to-artwork style transfer and photorealistic style transfer, via training the SGA modules on the basis of the learned auto-encoders and codebooks discussed in Section 4.2.

5. Experiments

5.1. Experiment Settings

Dataset. For photo-artwork, photo-photo, and artwork-artwork style transfer tasks, we use the MS-COCO [38] as the photo dataset and the WikiArt [41] as the artwork dataset, by following the existing artistic/photorealistic style transfer methods [1, 26, 36, 39]. For face-to-artwork style transfer, we use the FFHQ [30] as the photo dataset and the MetFaces [29] as the artwork dataset. For artwork-to-photo transfer, we let the 12k images with the "genre" tag of "landscape" in the WikiArt [41] be the artwork dataset, and the LandscapesHQ [49] be the photo dataset, to ensure a semantic alignment between artistic and realistic domains [50]. All input images are resized to 256×256 pixels.
Network architecture and training. The encoder/decoder of QuantArt consists of four blocks, where each block contains two ResBlocks [18] and a downsampling/upsampling layer. The intermediate feature has a spatial size of 16×16 and a feature dimension of 256. The codebook has $N = 1024$ entries and an entry dimension of $d = 256$. The style transfer model consists of six SGA modules. We implement QuantArt on the PyTorch framework [43]. For both two training stages, we use an Adam optimizer [31] with a batch size of 32, a learning rate of 4.5×10^{-6}, and a training epoch of 50. The loss weights of $\mathcal{L}_{AE}$, $\mathcal{L}_{codebook}$, $\mathcal{L}_{content}$, $\mathcal{L}_{style}$, and $\mathcal{L}_{adv}$ are set to 1, 1, 1, 10, and 0.8, respectively.

5.2. Multi-Fidelity Image Style Transfer

Our QuantArt(α, β) method produces diverse stylization results to meet the preferences of content, style, and visual fidelities for different users. Taking the Photo-to-Artwork style transfer task as an example, Fig. 6 shows a trilinear interpolation of the three fidelity terms by uniformly sampling parameter combinations (α, β). When the style fidelity $\beta = 0$, the model simply reconstructs the input content image. With $\beta = 1$ and visual fidelity $\alpha \to 1$, the generated images look more vivid but lose some texture details in the style reference. Note that neither $\alpha = 0$ or $\alpha = 1$ performs the best for this example, revealing the necessity of flexibly adjusting the fidelity terms in practical application.

5.3. Comparing with State-of-the-Art Methods

For a comprehensive evaluation, we qualitatively and quantitatively compare our QuantArt framework with the state-of-the-art algorithms on tasks including artistic style transfer (*i.e.*, Photo-to-Artwork, Artwork-to-Artwork) and photo-realistic style transfer (*i.e.*, Photo-to-Photo).
Methods for comparison. For artistic style transfer, we compare three lines of methods: 1) The feature statistics-based methods, including AdaIN [26], WCT [35], LinearWCT [33], DSTN [23], ArtFlow [1], EFDM [64], and CAST [65]; 2) The patch swapping methods, including StyleSwap [6] and AvatarNet [47]; 3) The attention-based methods, including SANet [42], AdaAttn [39], and StyTR2 [8]. For photorealistic style transfer, we compare PhotoWCT [36], WCT^2 [60], LinearWCT [33], and DSTN [23].

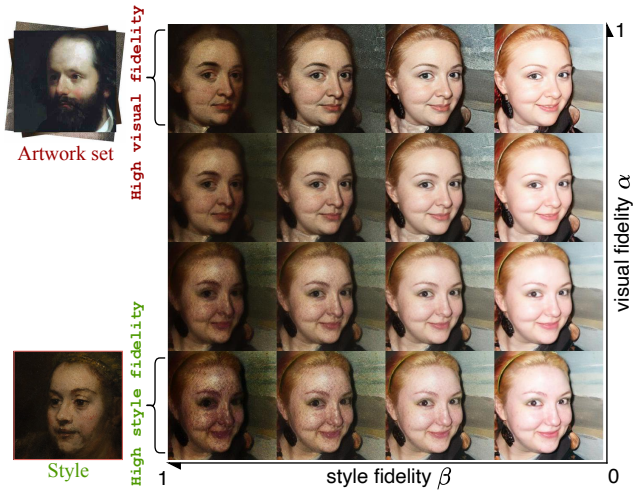


Figure 6. QuantArt achieves smooth interpolation among content, style, and visual fidelities by adjusting the parameters α and β.

Qualitative results. As shown in Fig. 7, on all three tasks, QuantArt with $\alpha = 0$ faithfully transfers the texture from input style images, while QuantArt with $\alpha = 1$ paints realistic textures thanks to the learned style codebook. For the Photo-to-Photo example, the content image does not have clearly visible textures, and it is challenging for other methods to hallucinate texture details when the scene in the style reference is lit up.
Quantitative results. We further quantitatively compare the performances of the state-of-the-art methods on the Photo-to-Artwork task. We randomly sample 10K images from the MS-COCO [38] and WikiArt [41] datasets, respectively, forming a total of 10K pairs of evaluation images. We use the LPIPS loss [62] to measure the content fidelity between the content and generated images. We use the style loss with a pretrained VGG-19 network [8, 26] to measure the style fidelity between the style and generated images. We compute the FID metric [21] between the style dataset and all the generated images to measure the visual fidelity of algorithms. We also adopt a recently proposed metric ArtFID = $(1+\text{LPIPS}) \cdot (1+\text{FID})$ [56] to evaluate the overall stylization performance. For a fair comparison, we compare our methods QuantArt(0,1) and QuantArt(1,1) with the previous methods. QuantArt(0,0) serves as a baseline to show the image reconstruction performance. As shown in Table 1, both our methods QuantArt(0,1) and QuantArt(1,1) achieve competitive performances compared with the baseline methods on all metrics. Specifically, QuantArt(1,1) performs better than other methods for FID and ArtFID, indicating that it can significantly enhance the visual fidelity

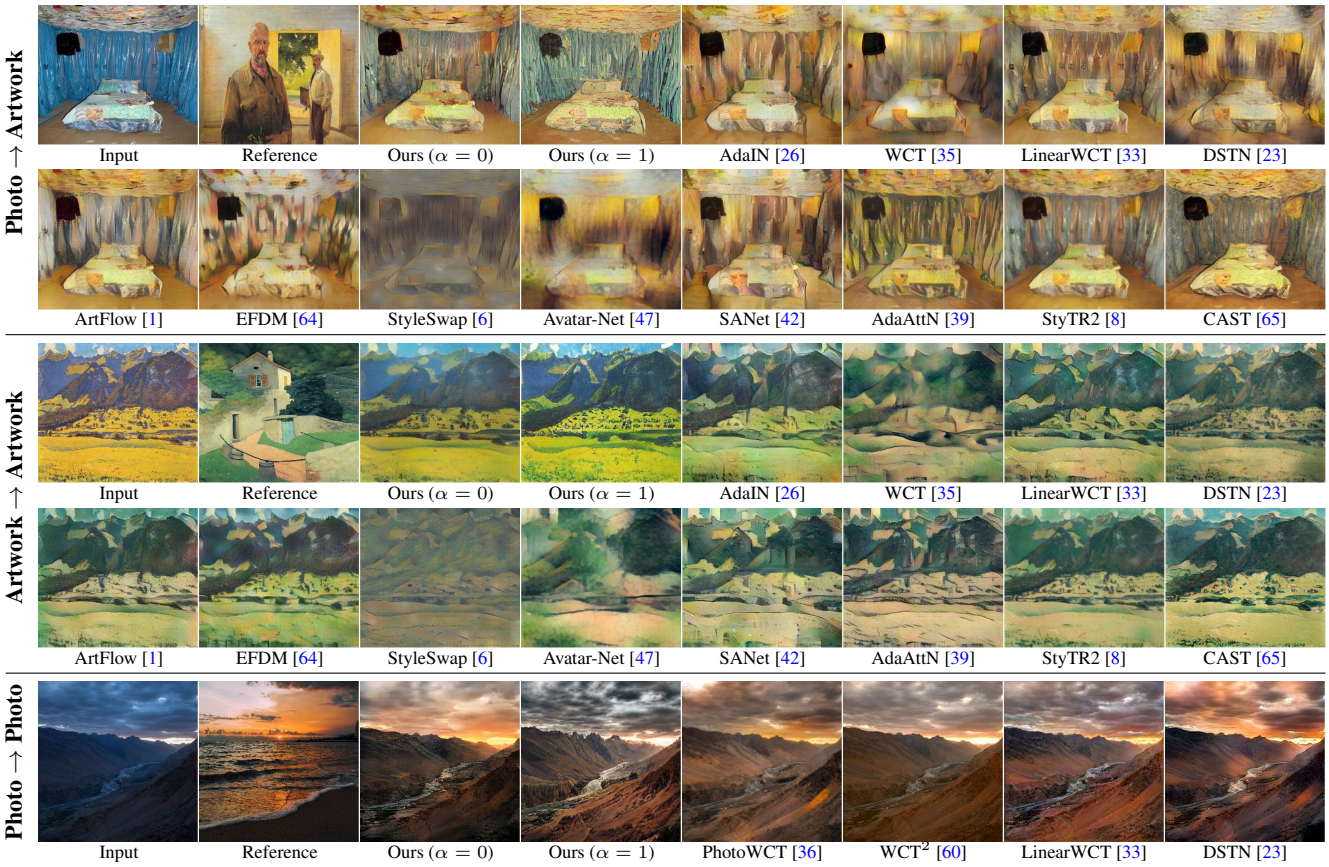


Figure 7. Comparisons of the state-of-the-art methods for artistic image style transfer, *i.e.*, Photo-to-Artwork and Artwork-to-Artwork, and photorealistic image style transfer, *i.e.*, Photo-to-Photo. *Zoom in to view the details.*

Table 1. Quantitative comparison of the universal style transfer methods. The **best** and second best results are highlighted, respectively.

Metric ↓	AdaIN	WCT	LinearWCT	ArtFlow	EFDM	StyleSwap	AvatarNet	SANet	AdaAttN	StyTR2	Ours (α, β)		
											$(0,0)$	$(0,1)$	$(1,1)$
LPIPS ↓	0.681	0.695	0.657	0.603	0.652	0.607	0.706	0.686	0.633	**0.514**	*0.159*	0.565	0.581
Gram loss ($\times 10^3$) ↓	0.163	0.282	0.172	**0.101**	0.402	1.357	0.718	0.120	0.246	0.386	-	0.395	0.864
FID ↓	36.618	65.193	48.156	28.899	51.070	74.168	58.178	27.080	25.894	30.893	-	25.590	**17.787**
ArtFID ↓	63.240	112.229	81.452	47.936	86.007	120.803	100.964	47.356	43.910	48.284	-	41.623	**29.695**

Table 2. Inference time (seconds) of a 256×256 image with a NVIDIA Tesla V100 GPU.

ArtFlow	StyleSwap	AvatarNet	SANet	AdaAttN	StyTR2	Ours
0.068	0.043	0.124	**0.007**	0.050	0.055	0.045

of style transfer results. QuantArt(0,1) performs better than QuantArt(1,1) on the Gram loss, empirically demonstrating the trade-off between style and visual fidelities. In Table 2, the inference time of our method is comparable with most of the benchmark universal style transfer methods.

5.4. Human Evaluation

Since image style transfer is a highly subjective task, we further examine our model with human evaluation.

User study on method performance. We perform a user study to subjectively evaluate the performance of different methods on three image style transfer tasks. On Photo-to-Artwork and Artwork-to-Artwork, we compare AvatarNet [47], ArtFlow [1], StyTr2 [8], and our QuantArt(1,1). On Photo-to-Photo, we compare PhotoWCT [36], WCT^2 [60], LinearWCT [33], and QuantArt(1,1). In each example, we show the input image, style reference, and style transfer results of the four comparison methods. The user study participants are asked to select one of the four stylizations to their preference. We show 12 examples to every participant, where each style transfer task has 4 examples. We have collected effective responses from a total of 59 participants. As shown in Fig. 8, our method significantly outperforms the prior arts on all three tasks with an average preference ratio of 43.5%, 51.9%, and 40.2%, respectively. The results also indicate that the visual fidelity is an important evaluation metrics for human to assess the style transfer algorithms.

Confusion test on visual fidelity. We further perform a novel user study, dubbed artistic style transfer confusion

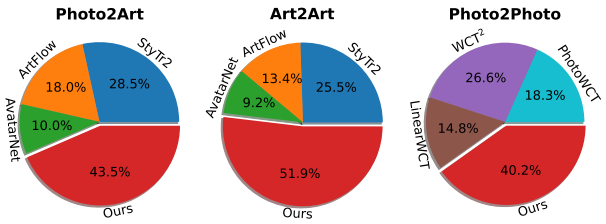


Figure 8. User study results on three image style transfer tasks. We show the percentages of methods preferred by the participants.

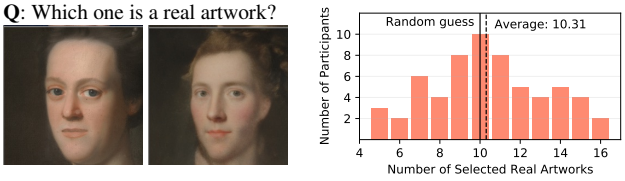


Figure 9. **(Left)** An example of our Artistic Style Transfer Confusion Test. Only 40.6% participants successfully distinguished the real artwork in this example. The answer can be found in our supplementary material. **(Right)** The statistical results with a total of 61 participants, where each participant is asked 20 questions.

test, to subjectively evaluate the visual fidelity of the stylized images. The left part of Fig. 9 shows an example question of the test. Within each question of the test, we ask the participant to select the real artwork from a pair of *i)* a real artwork image s and *ii)* a photo stylized according to the reference of s. The test is more challenging compared with the previous tests designed for image style transfer [65], since the fake image should follow the artistic style of the paired real image. We generate a total of 50 images using QuantArt(1,1). The content images are randomly selected from the FFHQ [30], LandscapeHQ [49], or MS-COCO [38] dataset. The style images are randomly selected from the WikiArt [41] or MetFaces [29] dataset. To avoid image contents that would lead to unfair comparison, such as the hairstyles of middle ages in the reference image or the modern articles in the content image, we manually filter 36 examples from the 50 examples. Each participant is requested to answer 20 questions, where they can choose either one of the two presented images, or skip the question if she/he feels difficult to make a choice. We have collected effective responses from 61 participants.

We plot the histogram of participants over the number of correct selections in the right part of Fig. 9. The distribution of the participants roughly follow the normal distribution with the mean number of correct selections 10.31, close to the random guess result (10). It indicates that our method can generate highly realistic artistic images which are difficult for human to identify from the real ones. More examples of this artistic confusion test can be found in the supplementary material.

Table 3. An ablation study of the design choices of QuantArt.

	LPIPS ↓	Gram ↓	FID ↓
Complete model, $\alpha = 1$, code size 16×16	0.581	0.864	**17.787**
Remove Quantization in SGA	0.545	0.993	18.757
Remove all Quantizations (*i.e.*, $\alpha = 0$)	0.565	0.395	25.590
Code size 8×8	0.656	1.078	27.741
Code size 32×32	0.513	0.798	28.663
Remove self-attention in SGA	0.584	0.883	19.025
Remove ResBlock in SGA	0.608	1.841	27.177
Replace SGA with StyTR2 decoder layer [8]	0.539	0.875	26.299
Shared encoders	0.588	0.788	30.806
Shared encoders and decoders	0.593	0.891	35.830

5.5. Ablation Study

As shown in Table 3, we conduct the following ablation studies to justify the design choices of the proposed framework. 1) *The effectiveness of vector quantization.* By removing the quantization operators in SGA or removing all the quantization operators, the Gram loss decreases while the FID increases. It is aligned with our motivation that vector quantization trades-off the style fidelity with visual fidelity. 2) *Code patch size.* Each code in QuantArt corresponds to a 16×16 patch of the feature maps. Either increasing or decreasing the results in an increase of FID, indicating that the code size of 16×16 is a good choice for image style transfer. 3) *Components in SGA.* Removing either the self-attention or ResBlock in SGA results in a worse FID. Removing the ResBlock additionally results in a significant increase of Gram loss. Replacing SGA with the decoder layer of StyTR2 [8] leads to a worse FID. 4) *Shared auto-encoders.* QuantArt adopts separate auto-enocoders to extract the continuous and quantized feature representations, respectively (see Fig. 3). The FID drastically increases when we share the encoders or auto-encoders for continuous and quantized representations. Therefore, in this work we use separate auto-encoders for feature extraction.

6. Conclusion

In this paper, we study the problem of enhancing the visual fidelity of image style transfer. We have proposed a quantizing style transfer algorithm to make the latent feature of the generated artwork closer to the real distribution. However, the algorithm may result in a generation that displays fewer style details of the reference image. To address this limitation, we have further presented a style transfer framework called QuantArt, which consists of a continuous branch and a quantized branch, to allow users to arbitrarily control the content preservation, style similarity, and visual fidelity of generated artworks. The experiments on Photo-to-Artwork, Artwork-to-Artwork, and Photo-to-Photo transfer settings have shown that our method achieves higher visual fidelity along with comparable content and style fidelities compared with the state-of-the-art methods.

Acknoledgements. This work was supported by NIH grant 5U54CA225088-03. The computations in this paper were run on the FASRC Cannon cluster supported by Harvard.

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 *CVPR*, pages 862–871, 2021. 1, 6, 7

[2] Jie An, Haoyi Xiong, Jun Huan, and Jiebo Luo. Ultrafast photorealistic style transfer via neural architecture search. In *AAAI*, volume 34, pages 10443–10450, 2020. 2

[3] Hsin-Yu Chang, Zhixiang Wang, and Yung-Yu Chuang. Domain-specific mappings for generative adversarial style transfer. In *ECCV*, pages 573–589, 2020. 1

[4] Dongdong Chen, Lu Yuan, Jing Liao, Nenghai Yu, and Gang Hua. Stylebank: An explicit representation for neural image style transfer. In *CVPR*, pages 1897–1906, 2017. 1

[5] Haibo Chen, Lei Zhao, Huiming Zhang, Zhizhong Wang, Zhiwen Zuo, Ailin Li, Wei Xing, and Dongming Lu. Diverse image style transfer via invertible cross-space mapping. In *ICCV*, 2021. 1, 2

[6] Tian Qi Chen and Mark Schmidt. Fast patch-based style transfer of arbitrary style. *arXiv preprint arXiv:1612.04337*, 2016. 1, 2, 6, 7

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009. 4

[8] Yingying Deng, Fan Tang, Weiming Dong, Chongyang Ma, Xingjia Pan, Lei Wang, and Changsheng Xu. Stytr2: Image style transfer with transformers. In *CVPR*, pages 11326–11336, 2022. 1, 6, 7, 8

[9] Vincent Dumoulin, Jonathon Shlens, and Manjunath Kudlur. A learned representation for artistic style. *arXiv preprint arXiv:1610.07629*, 2016. 2

[10] Vincent Dumoulin, Jonathon Shlens, and Manjunath Kudlur. A learned representation for artistic style. In *ICLR*, 2017. 2

[11] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *CVPR*, pages 12873–12883, 2021. 2, 3

[12] Oriel Frigo, Neus Sabater, Julie Delon, and Pierre Hellier. Split and match: example-based adaptive patch sampling for unsupervised style transfer. In *CVPR*, 2016. 2

[13] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *CVPR*, pages 2414–2423, 2016. 1, 2, 3, 5

[14] Leon A Gatys, Alexander S Ecker, Matthias Bethge, Aaron Hertzmann, and Eli Shechtman. Controlling perceptual factors in neural style transfer. In *CVPR*, pages 3985–3993, 2017. 1, 2

[15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 1, 3

[16] Shuyang Gu, Congliang Chen, Jing Liao, and Lu Yuan. Arbitrary style transfer with deep feature reshuffle. In *CVPR*, pages 8222–8231, 2018. 2

[17] Lanqing Guo, Chong Wang, Wenhan Yang, Siyu Huang, Yufei Wang, Hanspeter Pfister, and Bihan Wen. Shadowdiffusion: When degradation prior meets diffusion model for shadow removal. *arXiv preprint arXiv:2212.04711*, 2022. 3

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 4, 5, 6

[19] David J Heeger and James R Bergen. Pyramid-based texture analysis/synthesis. In *Annual conference on computer graphics and interactive techniques*, pages 229–238, 1995. 2

[20] Aaron Hertzmann. Painterly rendering with curved brush strokes of multiple sizes. In *SIGGRAPH*, 1998. 2

[21] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *NeurIPS*, 30, 2017. 6

[22] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 33:6840–6851, 2020. 3

[23] Kibeom Hong, Seogkyu Jeon, Huan Yang, Jianlong Fu, and Hyeran Byun. Domain-aware universal style transfer. In *ICCV*, pages 14609–14617, 2021. 2, 6, 7

[24] Siyu Huang, Haoyi Xiong, Zhi-Qi Cheng, Qingzhong Wang, Xingran Zhou, Bihan Wen, Jun Huan, and Dejing Dou. Generating person images with appearance-aware pose stylizer. In *IJCAI*, 2020. 3

[25] Siyu Huang, Haoyi Xiong, Tianyang Wang, Bihan Wen, Qingzhong Wang, Zeyu Chen, Jun Huan, and Dejing Dou. Parameter-free style projection for arbitrary image style transfer. In *ICASSP*, pages 2070–2074, 2022. 2

[26] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *CVPR*, pages 1501–1510, 2017. 1, 2, 3, 5, 6, 7

[27] Yongcheng Jing, Yezhou Yang, Zunlei Feng, Jingwen Ye, Yizhou Yu, and Mingli Song. Neural style transfer: A review. *IEEE transactions on visualization and computer graphics*, 2019. 2

[28] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, pages 694–711, 2016. 3, 5

[29] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. *NeurIPS*, 33:12104–12114, 2020. 6, 8

[30] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, pages 4401–4410, 2019. 3, 6, 8

[31] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[32] Nicholas Kolkin, Jason Salavon, and Gregory Shakhnarovich. Style transfer by relaxed optimal transport and self-similarity. In *CVPR*, pages 10051–10060, 2019. 3

[33] Xueting Li, Sifei Liu, Jan Kautz, and Ming-Hsuan Yang. Learning linear transformations for fast image and video style transfer. In *CVPR*, pages 3809–3817, 2019. 1, 6, 7

[34] Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Diversified texture synthesis with feed-forward networks. In *CVPR*, pages 3920–3928, 2017. 2, 3

[35] Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Universal style transfer via feature transforms. In *NIPS*, pages 386–396, 2017. 1, 2, 6, 7

[36] Yijun Li, Ming-Yu Liu, Xueting Li, Ming-Hsuan Yang, and Jan Kautz. A closed-form solution to photorealistic image stylization. In *ECCV*, pages 453–468, 2018. 2, 6, 7

[37] Jing Liao, Yuan Yao, Lu Yuan, Gang Hua, and Sing Bing Kang. Visual attribute transfer through deep image analogy. *arXiv preprint arXiv:1705.01088*, 2017. 2

[38] 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 *ECCV*, pages 740–755, 2014. 6, 8

[39] 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 *ICCV*, pages 6649–6658, 2021. 1, 3, 5, 6, 7

[40] Fujun Luan, Sylvain Paris, Eli Shechtman, and Kavita Bala. Deep photo style transfer. In *CVPR*, pages 4990–4998, 2017. 1, 2

[41] K Nichol. Painter by numbers, wikiart. https://www.kaggle.com/c/painter-by-numbers, 2016. 6, 8

[42] Dae Young Park and Kwang Hee Lee. Arbitrary style transfer with style-attentional networks. In *CVPR*, 2019. 3, 5, 6, 7

[43] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. volume 32, 2019. 6

[44] Javier Portilla and Eero P Simoncelli. A parametric texture model based on joint statistics of complex wavelet coefficients. *IJCV*, 40(1):49–70, 2000. 2

[45] Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. In *NeurIPS*, volume 32, 2019. 2, 3

[46] Eric Risser, Pierre Wilmot, and Connelly Barnes. Stable and controllable neural texture synthesis and style transfer using histogram losses. *arXiv preprint arXiv:1701.08893*, 2017. 2

[47] Lu Sheng, Ziyi Lin, Jing Shao, and Xiaogang Wang. Avatar-net: Multi-scale zero-shot style transfer by feature decoration. In *CVPR*, pages 8242–8250, 2018. 1, 2, 3, 6, 7

[48] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 4

[49] Ivan Skorokhodov, Grigorii Sotnikov, and Mohamed Elhoseiny. Aligning latent and image spaces to connect the unconnectable. In *ICCV*, pages 14144–14153, 2021. 6, 8

[50] Matteo Tomei, Marcella Cornia, Lorenzo Baraldi, and Rita Cucchiara. Art2real: Unfolding the reality of artworks via semantically-aware image-to-image translation. In *CVPR*, pages 5849–5859, 2019. 6

[51] D. Ulyanov, A. Vedaldi, and V. Lempitsky. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In *CVPR*, pages 6924–6932, 2017. 2

[52] Aaron Van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *NeurIPS*, volume 30, 2017. 2, 3, 4

[53] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 30, 2017. 5

[54] Zhizhong Wang, Lei Zhao, Haibo Chen, Zhiwen Zuo, Ailin Li, Wei Xing, and Dongming Lu. Divswapper: Towards diversified patch-based arbitrary style transfer. In *IJCAI*, pages 4980–4987, 2022. 2

[55] Holger Winnemöller, Sven C. Olsen, and Bruce Gooch. Real-time video abstraction. *ACMTOG*, 25(3):1221–1226, 2006. 2

[56] Matthias Wright and Björn Ommer. Artfid: Quantitative evaluation of neural style transfer. In *GCPR*, pages 560–576, 2022. 6

[57] Xide Xia, Tianfan Xue, Wei-sheng Lai, Zheng Sun, Abby Chang, Brian Kulis, and Jiawen Chen. Real-time localized photorealistic video style transfer. In *WACV*, pages 1089–1098, 2021. 2

[58] Xide Xia, Meng Zhang, Tianfan Xue, Zheng Sun, Hui Fang, Brian Kulis, and Jiawen Chen. Joint bilateral learning for real-time universal photorealistic style transfer. In *ECCV*, pages 327–342, 2020. 2

[59] Wenju Xu, Chengjiang Long, Ruisheng Wang, and Guanghui Wang. Drb-gan: A dynamic resblock generative adversarial network for artistic style transfer. In *ICCV*, pages 6383–6392, 2021. 1

[60] Jaejun Yoo, Youngjung Uh, Sanghyuk Chun, Byeongkyu Kang, and Jung-Woo Ha. Photorealistic style transfer via wavelet transforms. In *ICCV*, pages 9036–9045, 2019. 1, 2, 6, 7

[61] Jiahui Yu, Xin Li, Jing Yu Koh, Han Zhang, Ruoming Pang, James Qin, Alexander Ku, Yuanzhong Xu, Jason Baldridge, and Yonghui Wu. Vector-quantized image modeling with improved vqgan. *arXiv preprint arXiv:2110.04627*, 2021. 2

[62] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, pages 586–595, 2018. 3, 6

[63] Yabin Zhang, Minghan Li, Ruihuang Li, Kui Jia, and Lei Zhang. Exact feature distribution matching for arbitrary style transfer and domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8035–8045, 2022. 1, 2

[64] Yabin Zhang, Minghan Li, Ruihuang Li, Kui Jia, and Lei Zhang. Exact feature distribution matching for arbitrary style transfer and domain generalization. In *CVPR*, pages 8035–8045, 2022. 6, 7

[65] 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. 6, 7, 8

[66] Xingran Zhou, Siyu Huang, Bin Li, Yingming Li, Jiachen Li, and Zhongfei Zhang. Text guided person image synthesis. In *CVPR*, pages 3663–3672, 2019. 3