Diversified Arbitrary Style Transfer via Deep Feature Perturbation

Zhizhong Wang, Lei Zhao*, Haibo Chen, Lihong Qiu,
Qihang Mo, Sihuan Lin, Wei Xing, Dongming Lu

College of Computer Science and Technology, Zhejiang University
{endywon, cszhl, feng123, zjusheldon, moqihang, linsh, wxing, ldm}@zju.edu.cn

Abstract

Image style transfer is an underdetermined problem, where a large number of solutions can satisfy the same constraint (the content and style). Although there have been some efforts to improve the diversity of style transfer by introducing an alternative diversity loss, they have restricted generalization, limited diversity and poor scalability. In this paper, we tackle these limitations and propose a simple yet effective method for diversified arbitrary style transfer. The key idea of our method is an operation called deep feature perturbation (DFP), which uses an orthogonal random noise matrix to perturb the deep image feature maps while keeping the original style information unchanged. Our DFP operation can be easily integrated into many existing WCT (whitening and coloring transform)-based methods, and empower them to generate diverse results for arbitrary styles. Experimental results demonstrate that this learning-free and universal method can greatly increase the diversity while maintaining the quality of stylization.

1. Introduction

Style transfer, or to repaint an existing image with the style of another, is considered as a challenging but interesting task in both academia and industry. Recently, the pioneering works of Gatys *et al*. [7, 6, 8] have proved that the correlations (*i.e*., Gram matrix) between feature maps extracted from a pre-trained deep convolutional neural network (DCNN) can represent the style of an image well. Since then, significant efforts have been made to improve in many aspects including efficiency [29, 12, 16], quality [15, 31, 21, 10], generality [2, 5, 11, 19, 26, 22], user control [1, 9] and photorealism [23, 20, 32], etc. However, despite the remarkable success, these methods often neglect an important aspect, *i.e*., the diversity, since many of the applications (*e.g*., art creation and creative design) are required to satisfy the preferences of different users.

In terms of diversity, one common explanation is that, image style transfer is an underdetermined problem, where a large number of solutions can satisfy the same content and style, just like the results generated by different methods could all be visually pleasing and perceptually correct. However, the lack of meaningful variations in vanilla style transfer mechanism [8, 12, 29] hampers the emergence of diversity, as the optimization-based methods often converge to the similar local optimum, while the feed-forward networks only produce fixed outputs for the fixed inputs.

Although challenging and meaningful, unfortunately, this problem has barely received enough attention and there are only a few efforts to solve it. For instance, based on the feed-forward networks, Li *et al*. [18] introduced a diversity loss that penalized the feature similarities of different samples in a mini-batch. Ulyanov *et al*. [30] minimized the Kullback-Leibler divergence between the generated distribution and a quasi-uniform distribution on the Julesz ensemble [13, 35]. Although their methods could generate diverse texture samples or stylized images to a certain extent, they still suffer from three main limitations. (1) Restricted generalization. Once trained, their feed-forward network is tied to a specific style, which cannot be generalized to other styles. (2) Limited diversity. Since their diversity is learned by penalizing the variations in mini-batches of a finite dataset and the weight of diversity loss should be set to a small value, the degree of diversity is limited. (3) Poor scalability. Extending their approaches to other methods requires the intractable modifications to training strategies and network structures, which might be useful for some learning-based methods like [11], but not suitable for recent learning-free methods [19, 26, 20] as these methods transfer arbitrary styles in a style-agnostic manner.

Facing the aforementioned challenges, we rethink the problem of diversity and an important insight we will use is that a Gram matrix [8], which is widely used as the style representation of an image, can correspond to an infinite number of different feature maps, and the images recon-

* Corresponding author

structed from these feature maps are the diverse results we are looking for. Obviously, the problem of diversity has now been transformed into the problem of how to obtain the different feature maps with the same Gram matrix. Inspired by the work of Li *et al*. [19] which decomposes the Gram matrices and separates the matching of them by whitening and coloring transforms (WCTs), we propose a simple yet effective method, *i.e*., deep feature perturbation (DFP), to achieve diversified arbitrary style transfer. Our diversity is obtained by using an orthogonal noise matrix to perturb the image feature maps extracted from a DCNN while keeping the original style information unchanged. That is to say, although the perturbed feature maps are different from each other, they all have the same Gram matrix. For ease of understanding, we regard Gram matrix as the style representation, and define that different feature maps with the same Gram matrix share the same style-specific feature space.

In this work, our DFP is based on the framework of WCT [19], so it can be easily incorporated into many WCT-based methods [19, 26, 20] and empower them to generate diverse results without any extra learning process. Note that this learning-free process is fundamentally different from the aforementioned diversified methods that require learning with pre-defined styles. Therefore, our method is able to achieve diversified arbitrary style transfer.

The main contributions of this work are threefold:

• We propose to use deep feature perturbation, *i.e*., perturbing the deep image feature maps by an orthogonal noise matrix while keeping the original style information unchanged, to achieve diversified arbitrary style transfer.

• Our method can be easily incorporated into existing WCT-based methods [19, 26, 20] which are used for different style transfer tasks, *e.g*., artistic style transfer, semantic-level style transfer and photo-realistic style transfer.

• Theoretical analysis proves the capability of the proposed method in generating diversity, and the experimental results demonstrate that our method can greatly increase the diversity while maintaining the quality of stylization.

2. Related Work

Gram-based Methods. Gatys *et al*. [7, 6, 8] first proposed an algorithm for arbitrary style transfer and texture synthesis based on matching the correlations (*i.e*., Gram matrix) between deep feature maps extracted from a pre-trained DCNN within an iterative optimization framework, but one major drawback is the inefficiency. To address this, Johnson *et al*. [12] and Ulyanov *et al*. [29, 30] directly trained feed-forward generative networks for fast style transfer, but these methods need to retrain the network every time for a new style, which is inflexible. For this limitation, some methods [5, 33, 2, 18, 25] were proposed to incorporate multiple styles into one single network, but they are still limited in a fixed number of pre-defined styles. More recently, Huang and Belongie [11] further allowed arbitrary style transfer in one single feed-forward network.

WCT-based Methods. Recently, Li *et al*. [19] have proposed to exploit a series of feature transforms to achieve fast arbitrary style transfer in a style learning-free manner. They reformulated the task of style transfer as an image reconstruction process, with the feature maps of the content image being *whitened* at intermediate layers with regard to their style statistics (*i.e*., Gram matrix), and then *colored* to exhibit the same statistical characteristics of the style image. This method is essentially a Gram-based method, but it splits the Gram matrices by matrix decomposition, and separates the matching of them by whitening and coloring transforms (WCTs), thus providing an opportunity for our deep feature perturbation. Furthermore, Sheng *et al*. [26] combined it with style swap [3] for higher quality semantic-level style transfer. Li *et al*. [20] and Yoo *et al*. [32] developed this to fast photo-realistic style transfer. More recently, Li *et al*. [17] derived the form of transformation matrix theoretically and directly learned it with a feed-forward network. Lu *et al*. [22] derived a closed-form solution by treating it as the optimal transport problem. In our work, taking the most representative ones [19, 26, 20] as examples, the proposed method can be easily integrated into the learning-free WCT process and empower these methods to generate diverse results, which will be shown in Section 5.

Diversified Methods. Our method is closely related to [18] and [30]. Li *et al*. [18] introduced a diversity loss to allow the feed-forward networks to generate diverse outputs. It explicitly measures the variations in visual appearances between the generated results, and penalizes them in a mini-batch. Ulyanov *et al*. [30] proposed a new formulation that allowed to train generative networks which sampled the Julesz ensemble [13, 35]. Specifically, the diversity term of its learning objective is similar to that of Li *et al*. [18], which quantifies the lack of diversity in the batch by mutually comparing the generated images. Although these methods could generate diverse outputs to a certain extent, they still suffer from the restricted generalization, limited diversity and poor scalability, as we have introduced in Section 1.

The proposed method is based on WCT [19], and can be easily integrated into WCT-based methods to empower them to generate diverse results. Unlike the previous diversified methods [18, 30] that need to train an independent network for every style, our diversity is learning-free and suitable for arbitrary styles. Moreover, without extra constraints, our method can generate an infinite number of solutions with satisfactory quality as well as distinct diversity.

3. Style-Specific Feature Space

Defining the style of an image is a quite tricky problem, and so far no unified conclusion has been reached.

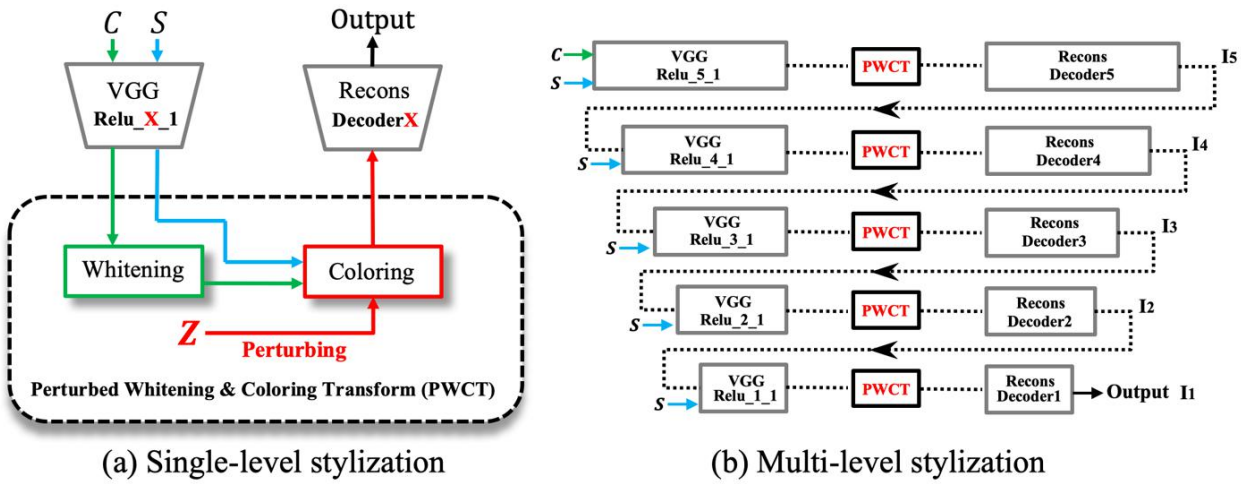

Figure 1. Our diversified arbitrary style transfer pipeline. (a) We add an orthogonal noise matrix $\mathbf{Z}$ to perturb the whitening and coloring transform (WCT). Like [19], the VGG and DecoderX are first trained for image reconstruction and then fixed for style transfer. C and S denote the content image and style image, respectively. (b) Our perturbed whitening and coloring transform (PWCT) can be applied in every level of the multi-level stylization framework of [19].

Informally, a style can be regarded as a family of visual attributes, such as color, brush strokes and line drawing, etc. Recently, Gatys *et al*. [7, 6, 8] have proposed a new style representation (Gram matrix) for artistic images. In their works, the style of an image is represented by the correlations between deep feature maps extracted from a pre-trained DCNN. Given an image $\vec{x}$ as input, the vectorized feature map extracted from a certain layer (we only take one layer as an example) of the VGG model [28] is denoted as $F \in \mathbb{R}^{C\times HW}$, where H, W are the height and width of the original feature map, C is the number of channels. The style of the image $\vec{x}$ can be represented as follows:

$$G_{ij} = \sum_k F_{ik}F_{jk} = FF^T \in \mathbb{R}^{C\times C}, \tag{1}$$

where F_{ik} and F_{jk} are the activations of the i^{th} and j^{th} filter at position k, F^T is the transpose matrix of F.

It is obvious that, for a definite Gram matrix $\mathcal{G}$, there could be a large number of feature maps corresponding to it. Let $\mathcal{F}_l$ denote the vectorized feature map of an image in layer l. $\mathcal{F}_l$ is perceived as the style $\mathcal{G}$ in layer l if its Gram matrix matches $\mathcal{G}$. Formally, given the loss function:

$$\mathcal{L}_{\mathcal{G}}(\mathcal{F}_l) = ||\mathcal{F}_l\mathcal{F}_l^T - \mathcal{G}||, \tag{2}$$

we define the feature maps that satisfy the following constraint belong to the same style-specific feature space of $\mathcal{G}$.

$$\mathcal{S}_{\mathcal{G}} = \{\mathcal{F}_l \in \mathbb{F} : \mathcal{L}_{\mathcal{G}}(\mathcal{F}_l) = 0\}, \tag{3}$$

where $\mathbb{F}$ is a set of feature maps. Features belonging to the same $\mathcal{S}$ are perceptually equivalent in style characteristics.

In particular, sometimes we do not need their Gram matrices to be exactly equal, and then we can get the relaxed constraint,

$$\mathcal{S}^{\epsilon}_{\mathcal{G}} = \{\mathcal{F}_l \in \mathbb{F} : \mathcal{L}_{\mathcal{G}}(\mathcal{F}_l) \leq \epsilon\}, \tag{4}$$

in which the feature maps are approximately equivalent in style characteristics.

In this work, our deep feature perturbation can easily achieve the first constraint (Eq. (3)), while the methods [18, 30] only satisfy the second constraint (Eq. (4)). That is to say, the Gram matrices of the diverse perturbed feature maps obtained by our method can be completely equal.

4. Deep Feature Perturbation

Our deep feature perturbation (DFP) is based on the work of Li *et al*. [19] and incorporated into its whitening and coloring transform (WCT) process to help generate diverse stylized results. The pipeline of our method is shown in Fig. 1, where the diversified style transfer is mainly achieved by the perturbed whitening and coloring transform (PWCT), which consists of two steps, *i.e*., whitening transform and perturbed coloring transform.

Whitening Transform. Given a pair of content image I_c and style image I_s, we first extract their vectorized VGG feature maps $F_c = \Phi(I_c) \in \mathbb{R}^{C\times H_cW_c}$ and $F_s = \Phi(I_s) \in \mathbb{R}^{C\times H_sW_s}$ at a certain layer Φ (*e.g*., $Relu\_3\_1$), where H_c, W_c (H_s, W_s) are the height and width of the content (style) feature, and C is the number of channels. We first center F_c by subtracting its mean vector m_c. Then the whitening transform (Eq. (5)) is used to transform F_c to $\hat{F}_c$, in which the feature maps are uncorrelated from each other (*i.e*., $\hat{F}_c\hat{F}_c^T = I$).

$$\hat{F}_c = E_cD_c^{-\frac{1}{2}}E_c^TF_c, \tag{5}$$

Table 1. Quantitative comparisons between single-level perturbation and multi-level perturbation in terms of run-time, tested on images of size 512×512 and a 6GB Nvidia 980Ti GPU.

Fig. 2	Li *et al*. [19]	I5	I4	I3	I2	I1	I5+I4	I5+I1	I3+I2+I1	I5+I4+I3+I2+I1
Time/sec	3.01	**3.53**	3.51	3.04	3.03	3.02	4.14	3.54	3.05	4.15
Fig. 3	Li *et al*. [20]	-	I4	I3	I2	I1	I4+I3	I4+I1	I2+I1	I4+I3+I2+I1
Time/sec	0.29	-	**0.32**	0.31	0.30	0.29	0.33	0.32	0.30	0.34

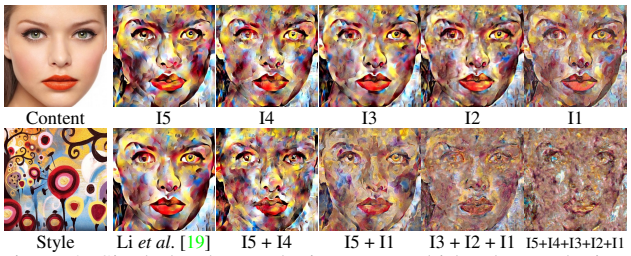

Figure 2. Single-level perturbation vs. Multi-level perturbation. Our DFP is integrated into method [19]. The top row shows the results obtained by only perturbing a single-level stylization in Fig. 1(b). The bottom row shows the results obtained by perturbing stylizations in multiple levels.

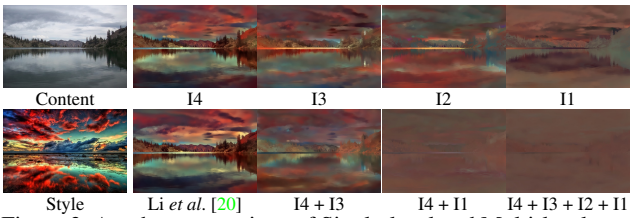

Figure 3. Another comparison of Single-level and Multi-level perturbation. Our DFP is integrated into method [20]. This method only uses four-level stylizations. The top row shows the results obtained by only perturbing a single-level stylization. The bottom row shows the results obtained by perturbing stylizations in multiple levels.

where D_c and E_c are obtained by the singular value decomposition (SVD) of the Gram matrix $F_c F_c^T \in \mathbb{R}^{C\times C}$ (Eq. (1)), *i.e.*, $F_c F_c^T = E_c D_c E_c^T$. D_c is the diagonal matrix of the eigenvalues, and E_c is the corresponding orthogonal matrix of eigenvectors.

Perturbed Coloring Transform. We first center F_s by subtracting its mean vector m_s. The coloring transform used in [19] is essentially the inverse of the whitening step, *i.e.*, using Eq. (6) to transform $\hat{F}_c$ so that we can obtain $\hat{F_{cs}}$ which satisfies the same Gram matrix of F_s (*i.e.*, $\hat{F_{cs}}\hat{F_{cs}}^T = F_s F_s^T$).

$$\hat{F_{cs}} = E_s D_s^{\frac{1}{2}} E_s^T \hat{F}_c, \tag{6}$$

where D_s and E_s are obtained by the SVD of the Gram matrix $F_s F_s^T \in \mathbb{R}^{C\times C}$, *i.e.*, $F_s F_s^T = E_s D_s E_s^T$. D_s is the diagonal matrix of the eigenvalues, and E_s is the corresponding orthogonal matrix of eigenvectors.

The goal of coloring transform is to make the Gram matrix of $\hat{F_{cs}}$ the same as that of F_s. According to our analysis in Section 3, these two feature maps share the same style-specific feature space. In theory, $\hat{F_{cs}}$ should have a large number of possibilities, but Eq. (6) only produces one of them. In order to traverse these solutions as much as possible, we propose to use deep feature perturbation.

The key idea of our deep feature perturbation is incorporating an orthogonal noise matrix into Eq. (6) to perturb the feature $\hat{F_{cs}}$ while preserving its Gram matrix. Obviously, there are three places to insert the noise matrix, *i.e.*, between $D_s^{\frac{1}{2}}$ and E_s^T, between E_s^T and $\hat{F}_c$, and on the right side of $\hat{F}_c$ (since $E_s^T E_s = I$ and $\hat{F}_c \hat{F}_c^T = I$). We eventually insert the orthogonal noise matrix between $D_s^{\frac{1}{2}}$ and E_s^T as this may consume the least computation and run-time (we will discuss this in Section 5.2).

We first obtain a random noise matrix N (*e.g.*, sampled from the standard normal distribution, we will discuss it in Section 5.2) according to the shape of $D_s^{\frac{1}{2}}$ and E_s^T. Assume that the shape of $D_s^{\frac{1}{2}}$ is $(C-k)\times(C-k)$, where k is the number of small singular values (*e.g.*, less than 10^{-5}, Li *et al*. [19] suggest removing these small singular values to obtain higher quality results), and the shape of E_s^T is $(C-k)\times C$, then the shape of N is $(C-k)\times(C-k)$. To obtain orthogonal noise matrix, we apply the SVD to decompose N, *i.e.*, $N = E_n D_n V_n^T$, and directly use the orthogonal matrix $\mathbf{Z} = E_n \in \mathbb{R}^{(C-k)\times(C-k)}$. Finally, we insert $\mathbf{Z}$ between $D_s^{\frac{1}{2}}$ and E_s^T of Eq. (6). Our new perturbed coloring transform is formulated as follows:

$$\hat{F_{csn}} = E_s D_s^{\frac{1}{2}} \mathbf{Z} E_s^T \hat{F}_c, \tag{7}$$

since $\mathbf{Z}\mathbf{Z}^T = I$, we can deduce as follows:

$$\begin{aligned}\hat{F_{csn}}\hat{F_{csn}}^T &= (E_s D_s^{\frac{1}{2}} \mathbf{Z} E_s^T \hat{F}_c)(\hat{F}_c^T E_s \mathbf{Z}^{\mathbf{T}} D_s^{\frac{1}{2}} E_s^T) \\ &= E_s D_s^{\frac{1}{2}} (\mathbf{Z} E_s^T \hat{F}_c \hat{F}_c^T E_s \mathbf{Z}^{\mathbf{T}}) D_s^{\frac{1}{2}} E_s^T \\ &= E_s D_s E_s^T = F_s F_s^T\end{aligned}$$

In our later experiments, we find that only using our perturbed coloring transform may reduce the quality of stylization. This may be because $\hat{F_{cs}}$ (Eq. (6)) contains not only

Figure 4. Trade-off between diversity and quality by varying diversity hyperparameter λ in method [19] (+ our DFP).

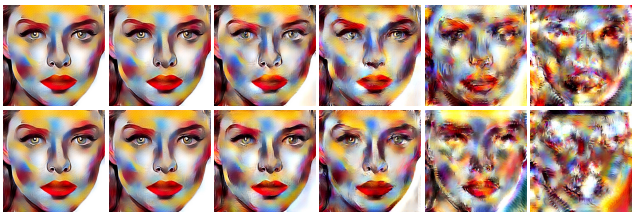

Figure 5. Trade-off between diversity and quality by varying diversity hyperparameter λ in method [26] (+ our DFP).

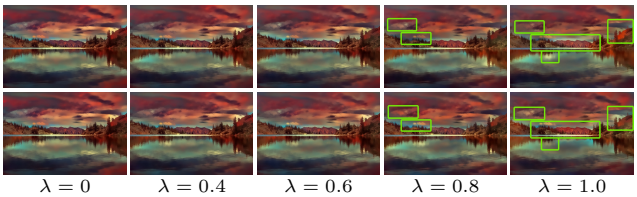

Figure 6. Trade-off between diversity and quality by varying diversity hyperparameter λ in method [20] (+ our DFP).

style information (Gram matrix) from $E_s D_s^{\frac{1}{2}} E_s^T$, but also some *content information from* $\hat{F}_c$ (Eq. (5)). Although our feature perturbation (Eq. (7)) can keep the style information unchanged, the content information may be affected by the noise matrix, which is manifested as a decline in quality. Fortunately, in WCT-based methods [19, 26, 20], the content information in $\hat{F}_c$ is *not the determinant* of the content in the final result, as in these methods $\hat{F}_{cs}$ is mainly served as the style feature, and blended with the content feature F_c to balance the style and content (similar to our Eq. (9)). In order to increase the diversity while maintaining the original quality, we introduce a diversity hyperparameter λ to provide user controls on the trade-off between them.

$$\hat{F_{csn}}' = \lambda \hat{F_{csn}} + (1-\lambda)\hat{F_{cs}}. \tag{8}$$

Then, we re-center the $\hat{F_{csn}}'$ with the mean vector m_s of the style, *i.e.*, $\hat{F_{csn}}' = \hat{F_{csn}}' + m_s$. At last, we blend $\hat{F_{csn}}'$ with the content feature F_c before feeding it to the decoder.

$$\hat{F_{csn}}' = \alpha \hat{F_{csn}}' + (1-\alpha)F_c, \tag{9}$$

where the hyperparameter α serves as the weight for users to control the stylization strength, like [19].

Multi-level Stylization. We follow the multi-level coarse-to-fine stylization used in [19], but replace their WCTs with our PWCTs, as shown in Fig. 1 (b). In fact, we do not need to add noise to every level. We will discuss this in Section 5.2.

Discussions. As a matter of fact, optimizing the diversity loss of [18, 30] can be viewed as a sub-optimal approximation of our method, as analyzed in Section 3. But since the diversity loss is only optimized on mini-batches of a finite dataset and the weight should be set to a small value (otherwise it will seriously reduce the quality), the degree of diversity is limited. By contrast, the different orthogonal noise matrices can be innumerable and diverse, so there could be endless possibilities with distinct diversity for the results of our approach. Moreover, our method is learning-free and can be effective for arbitrary styles, while the diversity loss of [18, 30] needs to be optimized every time for every style.

5. Experimental Results

5.1. Implementation Details

We incorporate our deep feature perturbation into three existing WCT-based methods which are used for different style transfer tasks, *i.e.*, [19] for artistic style transfer, [26] for semantic-level style transfer and [20] for photo-realistic style transfer. Except for replacing the WCTs with our PWCTs, we do not modify anything else, such as pre-trained models, pre-processing or post-processing operations, etc. If not specifically stated, in all experiments, the stylization weight α of our diversified version is consistent with the original version, and the random noise matrix N is sampled from the standard normal distribution. We fine-tune the diversity hyperparameter λ to make our quality similar to previous works, *i.e.*, 0.6 for [19], 0.5 for [26] and 1 for [20]. We will discuss these settings in the following sections. Our code is available at: https://github.com/EndyWon/Deep-Feature-Perturbation.

5.2. Ablation Study

Single-level Perturbation versus Multi-level Perturbation. We study the effects of single-level perturbation and multi-level perturbation on two WCT-based methods [19, 20], since they both use the multi-level stylization (while the method [26] only uses a single-level stylization). To perturb only specific levels, we set the diversity hyperparameter λ of the selected levels to default values (*i.e.*, 0.6 for [19] and 1 for [20]), and the other levels to 0. As shown in the top row of Fig. 2, when we perturb separately from the deepest level (I5) to the shallowest level (I1), the quality decreases accordingly. This phenomenon exists in the top row of Fig. 3 as well. We analyze the reason may be that the deeper level stylizes more low-frequency coarse

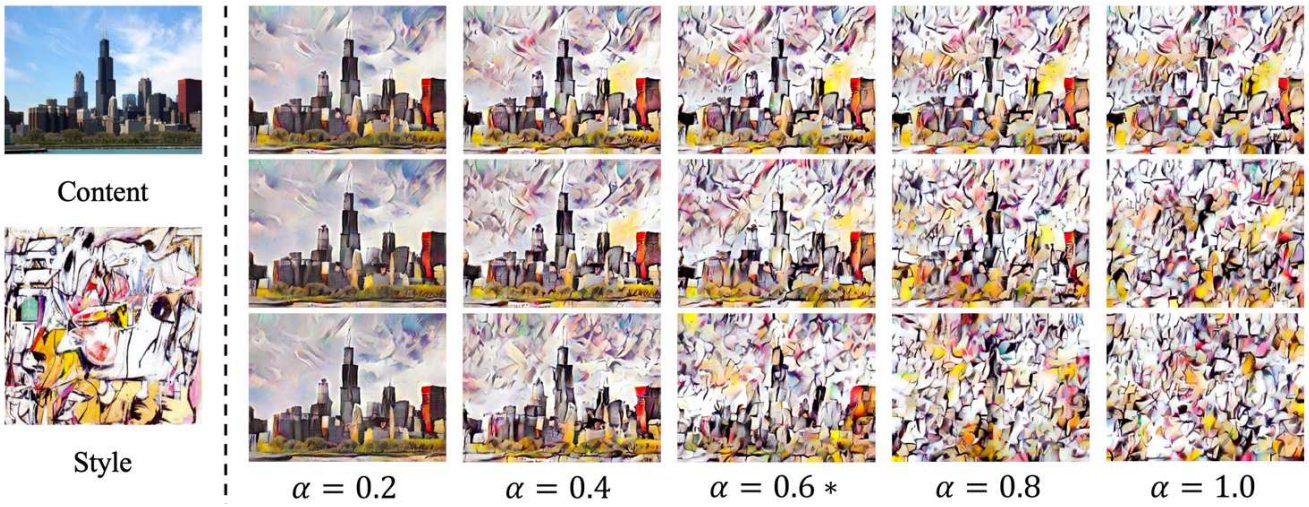

Figure 7. Relation between diversity and stylization strength. Each column (except for the first one) shows the results obtained by different α values (stylization strength). The top row shows the results of the original method [19]. The middle row shows the results obtained by setting $\lambda = 0.6$ (the default diversity strength) for our diversified version of [19]. The bottom row shows the results obtained by increasing the value of λ to 1 for our diversified version of [19]. $\alpha = 0.6$ is the default stylization setting of [19].

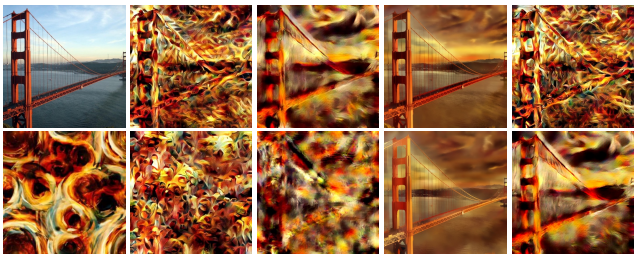

Figure 8. Relation between orthogonal noise matrix and generated result. The first column shows the input content (top) and style (bottom) images. The second to fourth columns show the results obtained by using the orthogonal noise matrix (top) and original random noise matrix (bottom) to perturb the methods [19, 26, 20], respectively. The last column shows the results obtained by varying the sampling distribution of the orthogonal noise matrix for methods [19] (top) and [26] (bottom).

characteristics while the shallower level stylizes more high-frequency fine characteristics, so adding noise into the shallower levels will affect the pixel performance of the final results. Perturbing at the deepest level can achieve comparable stylization quality as the original methods (see I5 in Fig. 2 and I4 in Fig. 3). On the other hand, multi-level perturbation introduces noise into multiple levels, as shown in the bottom rows of Fig. 2 and Fig. 3. We can see that introducing too much noise is unnecessary and will reduce the quality of stylization. We also compare the run-time in Table 1. Note that for method [20], we only consider the stylization time. Compared with the original methods (column 2), the incremental run-time decreases when we perturb the shallower levels. Nevertheless, the deepest-level perturbation only increases a very small amount of time (in **bold**).

Trade-off between Diversity and Quality. In Eq. (8), we introduce a diversity hyperparameter λ to provide user controls on the trade-off between diversity and quality. Different methods may require different λ values. In this part, we demonstrate the impact of different λ values on methods [19, 26, 20] while keeping their default stylization settings. For method [19] and [20], we only perturb the deepest level as suggested in the former sections. For method [26], we perturb its bottleneck layer as it only uses a single-level stylization. The results are shown in Fig. 4, 5 and 6. As we can see, the degree of diversity rises with the increase of λ values, but for method [19] and [26] (Fig. 4 and 5), the quality is obviously reduced when large λ values are applied. However, this problem does not arise in method [20] (Fig. 6), it may be because this method [20] contains a smoothing step to remove noticeable artifacts and it suppresses the emergence of diversity to some extent, which will also be verified by the quantitative comparisons in later Section 5.3. For trade-offs, we finally adopt 0.6, 0.5 and 1 for the default λ values of [19], [26] and [20], respectively.

Relation between Diversity and Stylization Strength. The diversity is also related to the stylization strength. Taking method [19] as an example, Fig. 7 demonstrates the relation between these two aspects. Comparing the top two rows, we can observe that for our default diversity setting ($\lambda = 0.6$), it works well for the situations where the stylization strength $\alpha \leq 0.6$, but destroys the content structure for those with larger α values. We set a larger diversity strength ($\lambda = 1$) in the bottom row, and we can observe that it still works fine for those with lower stylization strength (*e.g.*, $\alpha \leq 0.4$). That is to say, we can set a larger diversity strength for a smaller stylization strength. In fact, as

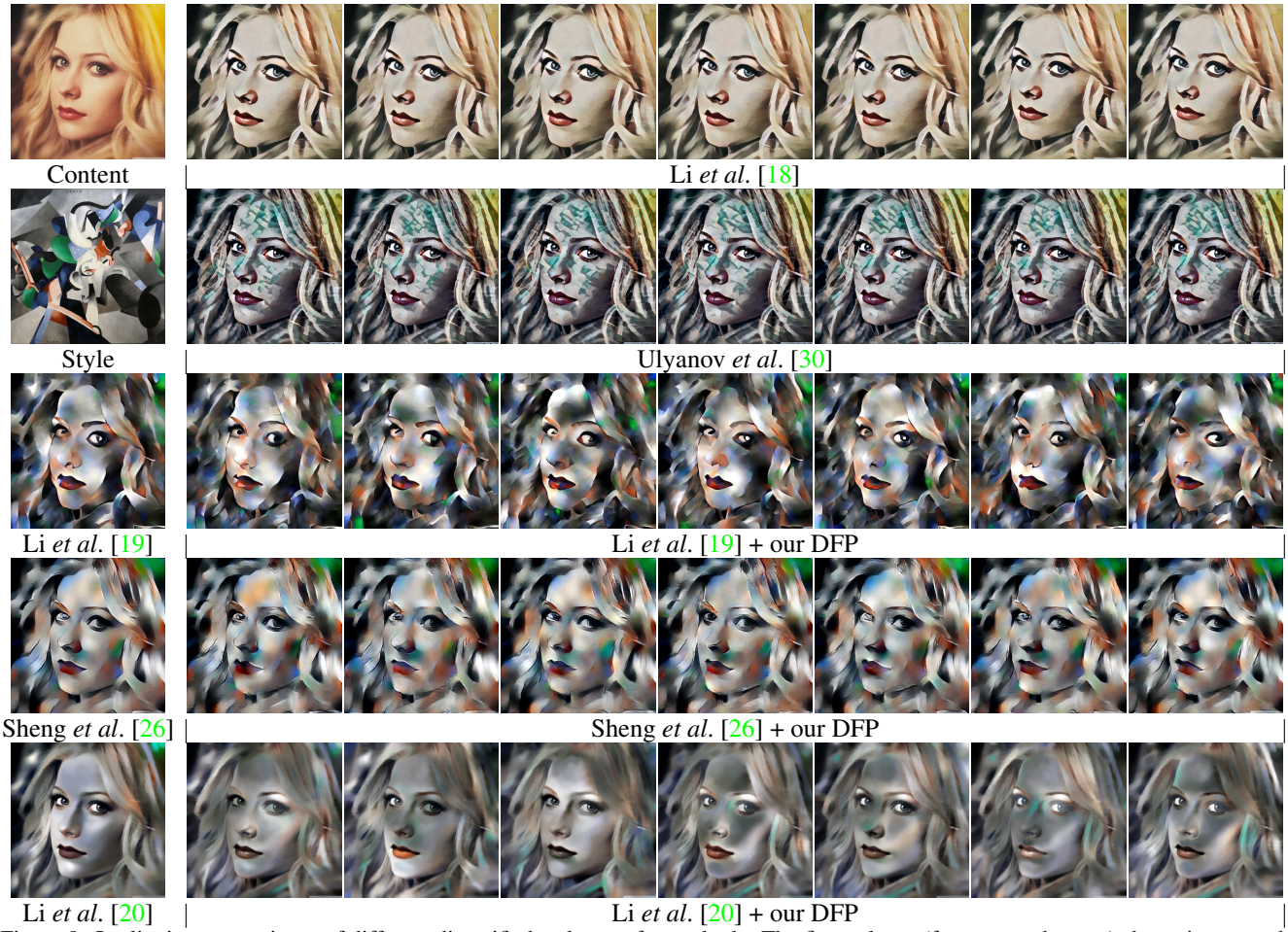

Figure 9. Qualitative comparisons of different diversified style transfer methods. The first column (from top to bottom) shows inputs and original outputs of [19, 26, 20]. The other columns (from top to bottom) show diverse outputs of [18, 30] and [19, 26, 20] (+ our DFP).

we have analyzed in Section 4, our diversity may affect the content information from $\hat{F}_c$ (Eq. (5)), so the content structure will be overwhelmed by the style patterns when the value of λ is too high, as validated in the last two columns. Therefore, the tradeoff between stylization strength (α) and diversity strength (λ) should be considered. Nevertheless, in practice, users only need to first determine the optimal stylization strength α (usually the default one) for different methods, and then adjust the appropriate λ values to keep the quality. Besides, in each method, our results have verified that the constant λ value can work *stably* on different content and style inputs.

Locations to Insert the Orthogonal Noise Matrix. In Section 4, we have mentioned three places to insert the orthogonal noise matrix in Eq. (6), *i.e.*, between $D_s^{\frac{1}{2}}$ and E_s^T, between E_s^T and $\hat{F}_c$, and on the right side of $\hat{F}_c$. We conduct the same experiments for each of them and find that there is no difference in qualitative comparisons. But in quantitative comparisons, *e.g.*, run-time and computation requirements, there are some differences. This is mainly due to the different computation of matrix multiplication caused by the different size of noise matrix. As we have analyzed in Section 4, when we insert the orthogonal noise matrix $\mathbf{Z}$ between $D_s^{\frac{1}{2}}$ and E_s^T, the size of $\mathbf{Z}$ is only $(C-k)\times(C-k)$, where C is the number of channels and k is the number of small singular values in $D_s^{\frac{1}{2}}$. For the other two cases, since the shapes of E_s^T and $\hat{F}_c$ are $(C-k)\times C$ and $C\times H_cW_c$, respectively (where H_c, W_c are the height and width of the content feature), the size of $\mathbf{Z}$ should be $C\times C$ if we insert it between E_s^T and $\hat{F}_c$, and $H_cW_c\times H_cW_c$ if we insert it on the right side of $\hat{F}_c$. Generally, for the deepest level, $C-k<C<H_cW_c$, so we eventually insert $\mathbf{Z}$ between $D_s^{\frac{1}{2}}$ and E_s^T as this may consume the least computation and run-time.

Relation between Orthogonal Noise Matrix and Generated Result. To verify the importance and necessity of the orthogonal noise matrix $\mathbf{Z}$ in our DFP, we compare it with the original random noise matrix N, and also discuss the influence of its sampling distribution. The results are

Table 2. Quantitative comparisons of different methods. We measure diversity using average Pixel distance and LPIPS distance [34].

Method	Pixel Distance	LPIPS Distance
Li *et al*. [18]	0.080	0.175
Ulyanov *et al*. [30]	0.077	0.163
Li *et al*. [19]	0.000	0.000
Sheng *et al*. [26]	0.000	0.000
Li *et al*. [20]	0.000	0.000
Li *et al*. [19] + our DFP	**0.162**	**0.431**
Sheng *et al*. [26] + our DFP	**0.102**	**0.264**
Li *et al*. [20] + our DFP	**0.091**	**0.203**

shown in Fig. 8, as we can see, using the original random noise matrix produces low quality results (see column 2 to 4 in bottom row). The results obtained by [19] and [26] are just like combinations of texture and noise, which drown out the content information. Compared with the former two, [20] can maintain the content information as much as possible even with the original random noise perturbation. This may be because it consists of two steps, and the second step removes noticeable artifacts to maintain the structure of the content image. But as the result shows, the quality is still significantly reduced. Similar to the former experiments, we also adjust the values of α and λ for original random noise perturbation, but the poor generation effect still cannot be alleviated. To explore the influence of sampling distribution of orthogonal noise matrix, we use uniform distribution instead of the standard normal distribution for method [19] (see the last column in top row), and vary the mean and standard deviation of normal distribution for method [26] (see the last column in bottom row). As we can see, the generated images do not show a significant difference from the default ones, which indicates that the key factor affecting the result is the orthogonality of noise $\mathbf{Z}$, rather than its sampling distribution.

5.3. Comparisons

In this section, we incorporate our DFP into methods [19, 26, 20] and compare them with other diversified style transfer methods [18, 30] from both qualitative and quantitative aspects. For methods [18] and [30], we run the author-released codes or pre-trained models with the default configurations. For our methods, we use the default settings as described in Section 5.1.

Qualitative Comparisons. We show qualitative comparison results in Fig. 9. We observe that [18] and [30] only produce subtle diversity (*e.g*., slight changes in the faces), which does not contain any meaningful variation. By contrast, for the methods with our DFP, the results show a distinct diversity (*e.g*., the faces, the hairs, the backgrounds, and even the eyes). Compared with the original outputs, the results obtained by incorporating our DFP are almost without quality degradation.

Quantitative Comparisons. We compute the average distance of sample pairs in pixel space and deep feature space to measure the diversity, respectively. For each method, we use 6 content images and 6 style images to get 36 different combinations, and for each combination, we obtain 20 outputs. There are totally 6840 pairs (each pair has the same content and style) of outputs generated by each method, we compute the average distance between them.

In pixel space, we directly compute the average pixel distance in RGB channels, which can be formulated as follows:

$$d_{pixel}(\vec{x}_1, \vec{x}_2) = \frac{||\vec{x}_1 - \vec{x}_2||_1}{W \times H \times 255 \times 3}, \tag{10}$$

where $\vec{x}_1$ and $\vec{x}_2$ denote the two images to compute the pixel distance. W and H are their width and height (they should have the same resolution).

In deep feature space, we use the LPIPS (Learned Perceptual Image Patch Similarity) metric proposed by Zhang *et al*. [34]. It computes distance in AlexNet [14] feature space ($conv1\_5$, pre-trained on Imagenet [24]), with linear weights to better match human perceptual judgments.

As shown in Table 2, [18] and [30] produce low diversity scores in both Pixel and LPIPS distance. Without our DFP, the original methods [19, 26, 20] cannot generate diverse results. By incorporating DFP, these methods show great diversity improvement. Note that since the method [26] (+ our DFP) is still restricted by some semantic constraints when transferring styles, and method [20] (+ our DFP) contains a smoothing step to remove detailed effects, their diversity scores are lower than those of method [19] (+ our DFP).

6. Conclusion

In this work, we introduce deep feature perturbation (DFP) into the whitening and coloring transform (WCT) to achieve diversified arbitrary style transfer. By incorporating our method, many existing WCT-based methods can be empowered to generate diverse results. Experimental results demonstrate that our approach can greatly increase the diversity while maintaining the quality of stylization. At this stage, we only explore the WCT-based methods, but this learning-free and universal paradigm may inspire a series of more ingenious and effective works in the future. Besides, WCT has also been widely used in many other fields, such as *image-to-image translation* [4], *GANs* [27], etc. Therefore, we believe our method may also provide a good inspiration for these research fields.

Acknowledgments. We sincerely thank the anonymous reviewers for helping us to improve this paper. This work was supported in part by the Zhejiang science and technology program (No: 2019C03137), and Zhejiang Fund Project (No: LGF18F020006, LY19F020049).

References

[1] Alex J Champandard. Semantic style transfer and turning two-bit doodles into fine artworks. *arXiv preprint arXiv:1603.01768*, 2016. 1

[2] Dongdong Chen, Lu Yuan, Jing Liao, Nenghai Yu, and Gang Hua. Stylebank: An explicit representation for neural image style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1897–1906, 2017. 1, 2

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

[4] Wonwoong Cho, Sungha Choi, David Keetae Park, Inkyu Shin, and Jaegul Choo. Image-to-image translation via group-wise deep whitening-and-coloring transformation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10639–10647, 2019. 8

[5] Vincent Dumoulin, Jonathon Shlens, and Manjunath Kudlur. A learned representation for artistic style. In *International Conference on Learning Representations (ICLR)*, 2017. 1, 2

[6] Leon Gatys, Alexander S Ecker, and Matthias Bethge. Texture synthesis using convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 262–270, 2015. 1, 2, 3

[7] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*, 2015. 1, 2, 3

[8] 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 (CVPR)*, pages 2414–2423, 2016. 1, 2, 3

[9] Leon A Gatys, Alexander S Ecker, Matthias Bethge, Aaron Hertzmann, and Eli Shechtman. Controlling perceptual factors in neural style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3985–3993, 2017. 1

[10] Shuyang Gu, Congliang Chen, Jing Liao, and Lu Yuan. Arbitrary style transfer with deep feature reshuffle. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8222–8231, 2018. 1

[11] 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 (ICCV)*, pages 1501–1510, 2017. 1, 2

[12] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 694–711. Springer, 2016. 1, 2

[13] Bela Julesz. Textons, the elements of texture perception, and their interactions. *Nature*, 290(5802):91, 1981. 1, 2

[14] Alex Krizhevsky. One weird trick for parallelizing convolutional neural networks. *arXiv preprint arXiv:1404.5997*, 2014. 8

[15] 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 (CVPR)*, pages 2479–2486, 2016. 1

[16] Chuan Li and Michael Wand. Precomputed real-time texture synthesis with markovian generative adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 702–716. Springer, 2016. 1

[17] Xueting Li, Sifei Liu, Jan Kautz, and Ming-Hsuan Yang. Learning linear transformations for fast arbitrary style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[18] Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Diversified texture synthesis with feed-forward networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 2, 3, 5, 7, 8

[19] Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Universal style transfer via feature transforms. In *Advances in Neural Information Processing Systems (NIPS)*, pages 386–396, 2017. 1, 2, 3, 4, 5, 6, 7, 8

[20] Yijun Li, Ming-Yu Liu, Xueting Li, Ming-Hsuan Yang, and Jan Kautz. A closed-form solution to photorealistic image stylization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 453–468, 2018. 1, 2, 4, 5, 6, 7, 8

[21] Jing Liao, Yuan Yao, Lu Yuan, Gang Hua, and Sing Bing Kang. Visual attribute transfer through deep image analogy. *ACM Transactions on Graphics (TOG)*, 2017. 1

[22] 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 International Conference on Computer Vision (ICCV)*, pages 5952–5961, 2019. 1, 2

[23] Fujun Luan, Sylvain Paris, Eli Shechtman, and Kavita Bala. Deep photo style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4990–4998, 2017. 1

[24] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. 8

[25] Falong Shen, Shuicheng Yan, and Gang Zeng. Neural style transfer via meta networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8061–8069, 2018. 2

[26] 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 (CVPR)*, pages 8242–8250, 2018. 1, 2, 5, 6, 7, 8

[27] Aliaksandr Siarohin, Enver Sangineto, and Nicu Sebe. Whitening and coloring batch transform for gans. In *International Conference on Learning Representations (ICLR)*, 2019. 8

[28] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 3

[29] Dmitry Ulyanov, Vadim Lebedev, Andrea Vedaldi, and Victor S Lempitsky. Texture networks: Feed-forward synthesis of textures and stylized images. In *International Conference on Machine Learning (ICML)*, pages 1349–1357, 2016. 1, 2

[30] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6924–6932, 2017. 1, 2, 3, 5, 7, 8

[31] Xin Wang, Geoffrey Oxholm, Da Zhang, and Yuan-Fang Wang. Multimodal transfer: A hierarchical deep convolutional neural network for fast artistic style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5239–5247, 2017. 1

[32] Jaejun Yoo, Youngjung Uh, Sanghyuk Chun, Byeongkyu Kang, and Jung-Woo Ha. Photorealistic style transfer via wavelet transforms. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 9036–9045, 2019. 1, 2

[33] Hang Zhang and Kristin Dana. Multi-style generative network for real-time transfer. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 349–365. Springer, 2018. 2

[34] 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 (CVPR)*, pages 586–595, 2018. 8

[35] Song Chun Zhu, Xiu Wen Liu, and Ying Nian Wu. Exploring texture ensembles by efficient markov chain monte carlo-toward a "trichromacy" theory of texture. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 22(6):554–569, 2000. 1, 2