Pacific Graphics (2021)
M. Okabe, S. Lee, B. Wuensche, and S. Zollmann (Editors)

# GANST: Gradient-aware Arbitrary Neural Style Transfer

Haichao Zhu

Tencent America, USA

**Abstract**

*Artistic style transfer synthesizes a stylized image with content from a target image and style from an art image. The latest neural style transfer leverages texture distributions as style information, and applies the style to content images afterwards. These methods are promising; however, they could introduce semantic content loss into synthesized results inevitably with the disregarded gradient information of input images. To tackle this problem, we propose a novel gradient-aware technique, called* GANST. *First,* GANST *decomposes input images to intermediate steerable representation that capture gradient information at multiple scales based on a Steerable Pyramid Neural Network (SPNN). With the extracted information,* GANST *preserves semantic content by integrating a novel loss representation of local gradients to AdaIN architecture, which we call Steerable Style Transfer Network (SSTN). Experimental results on various images demonstrate that our proposed* GANST *outperforms the state-of-the-art methods in producing results with concrete style reflected and detailed content preserved.*

**CCS Concepts**

• ***Computing methodologies*** → ***Neural networks;***

## 1. Introduction

Artistic style transfer, creating an artistic stylized image for content of a target image and style of an example art image, is widely used in a broad range of domains [WWH06,RDB16]. Conventional texture-modeling based techniques first extract the texture information from source style images and then transfer the extracted information to target content images while preserving the semantic content. However, these methods only cover low-level image features (e.g., pixel values) without considering the perceptual and semantic information, leading to quality degradation in semantic regions of content images; hence, balancing style transfer effect and content preserving remains an open research question.

Recently, neural network based artistic style transfer [JYF*19], has emerged as one of the most effective techniques to synthesize a stylized image since the pioneering work [GEB15]. Such techniques are based on an implicit assumption that style information could be represented as texture distributions using Gram matrices [LWLH17]. Gram matrices have been proven to be highly effective to incorporate high-level semantic information in style transfer. Consequently, neural style transfer based applications are getting increasingly prevalent such as Prisma because it lowers the barrier to create an artistic image.

However, prior work usually fails to preserve the semantic content of input images, *e.g.,* the synthesized image could become vague when the size of the content image is much smaller than that of the style image. To capture content information, local gradients of input images must be considered in the style transfer. Yet this *gradient-aware* style transfer for *a*rbitrary styles problem has been largely overlooked.

On one hand, without considering local gradients, *e.g.,* the texture orientation, of input images may lead to undesired artifacts into the synthesized images. Figure 1 shows an example in which the neural style transfer [HB17] produces diffident visual results after we resize or rotate the input images. The two images in the first row give the content image and style image respectively. To better illustrate the consequences of disregarded local gradients, in this example the style image only contains vertical and horizontal lines. After scaling down the size of the content image, the results become vaguer because large structures become small, as presented in the images in the second row. The third row shows scaling down of the style image, which introduces various strokes into the output images. Last, if we rotate the style image, the directions of edges in the synthesized images are changed (e.g., the edges of the glasses frame are different from horizontal). This motivates that capturing the overlooked gradients is necessary. In [WSZL19], Wu et al. propose to address the direction of content image in style transfer; however, it cannot handle directions at multiple scales and cannot be applied to arbitrary style transfer.

On the other hand, many of existing feed-forward neural style transfer techniques are restricted to a fixed set of styles and they do not work well for unseen styles. To tackle this problem, arbitrary style transfer is proposed. [JYF*17] proposes an arbitrary-style-per-model framework (ASPM) to achieve arbitrary style transfer



DOI: 10.2312/pg.20211399

**Figure 1:** *The synthesized results (using [HB17]) are visually different when we resize or rotate the input images. First row: the input content and style images with sizes* $694 \times 694$ *and* $1000 \times 1000$ *respectively. Second row: Synthesized results by scaling down the content image by the factors of* 1, 2, 4 *and* 8 *respectively. Third row: Synthesized results by scaling down the style image by the factors of* 1, 2, 4 *and* 8 *respectively. Last row: Synthesized results by clockwise rotating the content image by* $0^\circ$, $30^\circ$, $45^\circ$ *and* $60^\circ$ *respectively and then rotating them back for visualization.*

using only one neural network (i.e., they do not have to retrain the neural network for unknown styles). Other representative work include adaptive instance normalization (AdaIN) [HB17], whitening and coloring transforms (WCT) [LFY*17], Avatar-Net [SLSW18], attention-aware multi-stroke (AAMS) [YRX*19] and error transition network (ETNet) [SWZ*19]. Unfortunately, none of these methods consider the local gradient problem.

In this paper, we propose a novel Gradient-Aware Neural Style Transfer technique, called GANST, to incorporate multi-scale local gradients of images into *arbitrary* style transfer. Such gradients characterize the semantic contents. To be specific, we propose a Steerable Pyramid Neural Network (SPNN) to decompose an image into multiple scales and orientations to get multi-scale gradients accurately, and we preserve the semantic content by minimizing a novel loss of local gradients while transferring arbitrary styles in training a Steerable Style Transfer Network (SSTN), which follows the AdaIN architecture. Experimental results demonstrate GANST's ability to efficiently generate artistic images with concrete style reflected and detailed content preserved, which we show is out-of-reach for state-of-the-art methods [HB17, LFY*17, SLSW18, YRX*19, SWZ*19]. In summary, this work makes the following contributions:

- Compared to existing work, GANST synthesizes an artistic image with an *arbitrary* style, while preserving *semantic contents*. No specific styles are baked into the framework.
- We adapt steerable pyramids to construct an SPNN that decomposes an image to get multi-scale gradients.
- We propose a novel gradient loss to capture the semantic contents in training the style transfer network.

## 2. Related Work

**Texture Modeling.** Texture modeling has attracted extensive research attention over decades. Most of the texture modeling research work focuses on either filtering-based texture representations or statistical modeling based texture representations. The first one decomposes an image with manually designed filters, such as wavelets [Mal89], and steerable filters [FA91]. The statistical modeling based methods describe textures as probability distributions on random fields, such as Markov Random Fields [CJ83]. After that, the research focus changed to invariant feature representations. This gave a rise to the development of local invariant descriptors, such as Scale Invariant Feature Transform (SIFT) [Low04]. These invariant features have been dominating computer vision area for many years until the success of image classification using deep learning [KSH12] in 2012. Different to previous methods, deep learning based methods [CMV15] seek to learn good feature representations from images directly rather than design features manually.

**Neural Style Transfer.** Neural style transfer utilizes a neural network to transfer styles. It has been received extensive attention after the work of Gatys et al. [GEB15]. Their main idea is to align texture distributions captured in a CNN using Gram matrices to trade off between style effect and content preserving. As they do not consider the local image gradients, the stylized images may not be plausible. In [JAFF16], Johnson et al. propose to use a feed-forward neural network to replace the gradient descent step to speed up reconstructing images. However, their method cannot achieve arbitrary style transfer because the networks are tied to a fix set of styles. More recently, Arbitrary-Style-Per-Model (ASPM) [JYF*17] has been proposed to transfer arbitrary styles using only one network. Huang et al. [HB17] introduce the Adaptive Instance Normalization (AdaIN) to align feature distributions using the mean and variance of features. In [LFY*17], Li et al. propose to use whitening and coloring transforms (WCT) to align feature distributions. Song et al. [SWZ*19] introduce an iterative error-correction mechanism to improve arbitrary style transfer effect. The stylizing effects can be further improved by attention mechanism [YRX*19].

Unfortunately, all these work fails to give high quality results when we change the scales or orientations of the input images because they disregard the local image gradients. [WSZL19] proposes a direction-aware style transfer to get gradient information using local direction field, but it is fixed to a set of styles with a single scale only. To the best of our knowledge, *none* of the above work tackles *m*ulti-scale local gradients of input images for *a*rbitrary style transfer, and this motivates the proposed GANST technique in this paper.

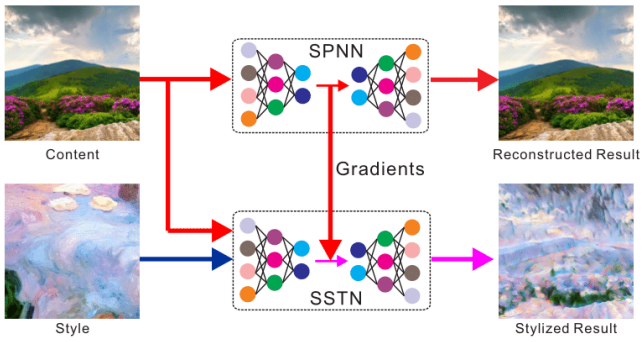


**Figure 2:** *System Overview. Our method contains two neural networks: SPNN and SSTN. The SPNN decomposes an image into multiple scales and orientations to get local gradients. The SSTN transfers styles between images to minimize the loss based on the gradients extracted from SPNN.*

## 3. Proposed Approach

### 3.1. System Overview

GANST contains two components that work in concert to make arbitrary style transfer preserving as much image content as possible by leveraging the multi-scale local gradients. Figure 2 shows (A) multi-scale gradients extraction based on Steerable Pyramid Neural Network (SPNN), discussed in Section 3.2 and (B) Steerable Style Transfer Network (SSTN) with gradient loss based on AdaIN [HB17] architecture, discussed in Section 3.3. This approach is based on the insights that (1) a steerable pyramid can decompose an image into multiple scales and multiple orientations to get multi-scale gradients; and (2) neural style transfer can preserve semantic contents by minimizing the losses in local gradients, which characterize the contents of an image.

### 3.2. Steerable Pyramid Neural Network

Multi-scale gradients can be used for characterizing the semantic contents in the style transfer process. However, extracting such information is challenging because the extracted gradients should be accurate and local to the image structures.

To tackle this problem, GANST resorts to learning steerable pyramid filter kernels using deep convolutional neural networks. A steerable pyramid [SF95] has been proven to be highly effective in texture synthesis. It decomposes an image into multiple scales and multiple orientations to get gradients, based on which it can reconstruct the image. Inspired by this decomposition and reconstruction procedure, the proposed network SPNN, as shown in Figure 3, adopts the encoder-decoder framework. The encoder decomposes input images into intermediate representations (i.e., filter responses), and the decoder reconstructs images using these representations from the encoder. To make the learned intermediate representations capture gradient information, the filter kernels of convolutional layers are required to be steerable at multiple scales.

The encoder is composed of three convolutional layers. At each convolutional layer, there are two kinds of filter kernels: low frequency filters and steerable filters. The first group of filters are used

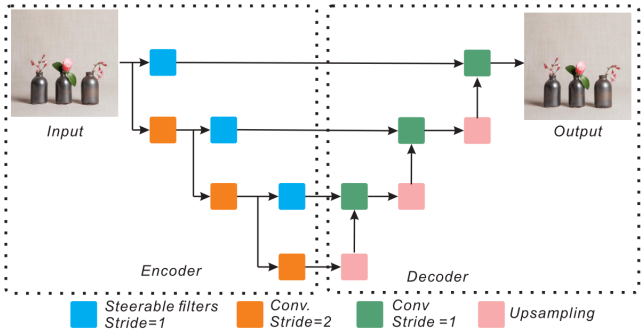


**Figure 3:** *The architecture of the SPNN. The encoder contains three levels of convolution. At each level, the input is convoluted with a group of steerable filters and a group of low frequency filters. The decoder almost mirrors the encoder with all convolutional layers are followed with nearest upsampling.*

to capture low frequencies in images. The convolution of these filters is performed by the stride of two to downscale the input for the next level convolution. The steerable filters are designed to capture orientation information in images by the stride of one at each scale.

We use the formulation by Freeman et al. [SF95] to describe our learning strategy. In this formulation, a filter $\Psi$ is steerable if it is a combination of angular coefficient functions $\kappa_q$:

$$\rho_\theta \Psi(x) = \sum_{q=1}^{Q} \kappa_q(\theta) \psi_q(x) \tag{1}$$

for angles $\theta \in (-\pi, \pi]$. Here $\rho_\theta$ is the rotation operator, which rotates a function or a coordinate vector counterclockwise by angle $\theta$. Then the response of orientation can be synthesized from basis responses:

$$(f * \rho_\theta)\Psi(x) = \sum_{q=1}^{Q} \kappa_q(\theta)(f * \psi_q)(x)) \tag{2}$$

The rotation operation can be constructed by multiplication with a complex exponential:

$$\rho_\theta \psi_q(x) = e^{-iq\theta} \psi_q(x) \tag{3}$$

Let $(r, \phi)$ be the polar coordinate of $x = (x_1, x_2)$, $\tau(r)$ be a radial function and $k \in \mathbb{Z}$ be the angular frequency, then $\psi_q(x)$ can be written as:

$$\psi_k(r, \phi) = \tau(r) e^{ik\phi} \tag{4}$$

and Gaussian functions are used for the radial parts as $\tau(r) = \exp\left(-(r-\mu)^2 / 2\sigma^2\right)$. Finally, the learned filters are represented as linear combinations of the elementary filters:

$$\hat{\Psi} = \sum_{k=0}^{K} w_k \psi_k(x) \tag{5}$$

The decoder almost mirrors the encoder except that all convolutional layers will be followed with nearest upsampling. To reconstruct the images accurately, we measure the loss between original image $I$ and the output of decoder $\hat{I}$ using squared error:

$$\mathcal{L} = \frac{1}{2} \left\| I - \hat{I} \right\|^2 \tag{6}$$

By optimizing Equation 6, the encoder and decoder are trained jointly to get the intermediate filter kernels.

The above decomposition from an input image to intermediate steerable representations captures the local gradients at multiple scales, which maintain the semantic content of an image. The extracted multi-scale gradients using these filters will be integrated into the loss function in Section 3.3.

### 3.3. Steerable Style Transfer Network with Gradient Loss

To adapt neural style transfer to transfer arbitrary style while preserving as much content as possible, we (1) formulate a novel loss function based on the extracted gradients in Section 3.2 to represent the content preservation, and (2) incorporate such loss function into our steerable style transfer network (SSTN) based on the AdaIN [HB17] architecture. The insight behind is semantic content can be preserved by minimizing the losses on local gradients. Particularly, the overall loss $\mathcal{L}$ with the proposed loss $\mathcal{L}_g$ is formulated as follows:

$$\mathcal{L} = \alpha\mathcal{L}_c + \beta\mathcal{L}_s + \gamma\mathcal{L}_g \tag{7}$$

where $\mathcal{L}_c$ and $\mathcal{L}_s$ are the original content and style loss respectively, and $\mathcal{L}_g$ is the sum of gradient loss over three layers of SPNN:

$$\mathcal{L}_g = \sum_l w_l \mathcal{L}_g^l \tag{8}$$

where $w_l$ is the weight and $\mathcal{L}_g^l$ is defined by the sum of the squared error between the output of steerable filters:

$$\mathcal{L}_g = \frac{1}{2}\sum_{i=1}^{N_l}\sum_{j=1}^{M_l}\left(S_c^l(i,j) - S_t^l(i,j)\right)^2 \tag{9}$$

where $S_c^l(i,j)$ and $S_t^l(i,j)$ are the feature maps of the steerable filters for the content and synthesized images respectively.

By training the AdaIN layer in the network with this modified loss formulation, GANST is able to transfer arbitrary style while minimizing the semantic content losses between the content images and the results.

## 4. Experiments

Our proposed GANST is successful in addressing gradient-aware arbitrary style transfer challenge. In this section we train it by using two datasets, including MS-COCO [LMB*14] and WikiArt [Nic16], and evaluate in on various images.

### 4.1. Implementation Details

To extract multi-scale local gradients, we train the proposed SPNN using MS-COCO dataset, which contains roughly 80,000 images. We implement SPNN by TensorFlow on a workstation with a Intel i7 8700K CPU and an NVIDIA GeForce GTX 1080ti GPU. During training, we use the adam optimizer [KB14] and a batch size of 256 to optimize Eq. 6. The training loss reduces rapidly after the first few epochs and converges in around 100 epochs, as shown in Figure 5 (A).

As for the style transfer network, we use MS-COCO as content images and WikiArt as style images. WikiArt contains roughly 80,000 art images. During training, we use the adam optimizer and a batch size of 16 image pairs to optimize Eq. 7. All the weights, including $w_i$, $\alpha$, $\beta$ and $\gamma$ are set to 1. Figure 5 (B) gives the training loss of Eq. 7. All these three losses reduce rapidly after a few epochs and converge after 150 epochs.

### 4.2. Qualitative Evaluation

**Comparison against Prior Work.** To demonstrate the effectiveness of our method, we compare GANST with prior work [HB17, LFY*17, SLSW18, YRX*19, SWZ*19] with arbitrary styles. We evaluate various content images including portraits, animals, landscapes with distinctive styles and report them in Figure 4. The results of competitors are produced using their default settings. AdaIN transfers styles by aligning the mean and variance of feature maps without considering local information. The contents of input images, *e.g.,* the grass textures, are not well preserved. Besides, AdaIN cannot fully capture style information, *e.g.,* the style strokes, in the portrait examples, are not fully presented in the results. Similarly, WCT also introduces undesired artifacts to the results, *e.g.,* the face contours are destroyed. Avatar-Net uses domain adaptation and AAMS uses an attention-based method to transfer styles. Although concrete styles are presented, the details of content images are poorly preserved. ETNet uses both progressive strategy and error-correction to improve style effect; however, large distortion are also introduced, *e.g.,* the human eyes. In contrast, GANST produces results with concrete styles reflected and detailed content preserved.

**Ablation Study with GANST Variants.** To better evaluate the impact of our new loss component, and the extracted gradients over different configuration layers of SPNN in isolation, we conduct an ablation study against downgraded versions of GANST. Please note that AdaIN is the downgraded version of GANST without gradient loss, which has already been discussed in Section 4.2. Additionally, Figure 6 shows the results when we train the SSTN with different extracted gradients over different configuration layers of SPNN. The first column shows the input content and style pairs. The three results in each row are generated with the loss over the third layer, the third and the second layers, and all the layers of SPNN respectively. The SSTN can capture the gradient changes from coarse-to-fine granularity through the third layer to the first layers of SPNN, resulting in different output images.

### 4.3. Quantitative Evaluation

**User Study.** Artistic style transfer is a highly subjective task; hence, we conduct a cross-subjects user study with twenty participants to investigate whether GANST synthesizes stylized images with higher quality than others or not. This user study is conducted on Upwork [Upw20]. We hired 20 participants to evaluate the results. Ten professional participants were working in image processing and computer graphics with experiences in style transfer. Five of the remaining non-professional participants were working in photography or painting field, while the others had no related background, showing a good mix of different levels of expertise and different tastes of arts.


**Figure 4:** *We compare our method with [HB17, LFY*17, SLSW18, YRX*19, SWZ*19] on various content images including portraits, animals and landscapes with distinctive styles.*

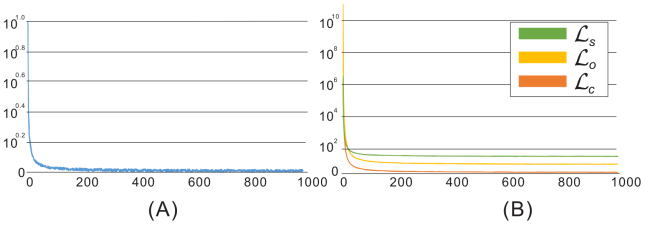


**Figure 5:** *(A) Training loss of SPNN. (B) Training Loss of SSTN.*

To prepare for the study, we collect 20 content and 20 style images from the dataset aforementioned and synthesize 400 images for each of the following six methods: GANST, AdaIN, WCT, Avatar-Net, AAMS and ETNet. These input content and style images are selected by five experienced artists in the participants instead of randomness. In the study session, we show each participant the original 20 content and 20 style images, and the $400 \times 6$ stylized images generated using the above six methods. Then each participant is required to: (1) choose one image that has better transfer results in terms of style effect; (2) choose one image that better preserve the content characters and (3) choose one image that is preferred to be shared on social network.

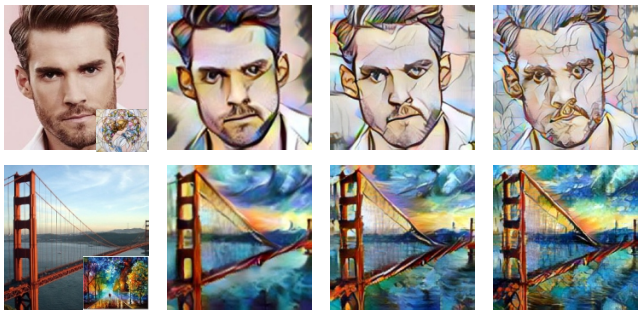

**Figure 6:** *The impact of different layers of SPNN on Style Transfer. First column: input content and style images pairs. Second to fourth columns: results generated with the network trained with loss over the third layer, the third and the second layers, and all the layers of SPNN respectively.*

The results are shown in Table 1, in which the three columns "S. E.", "C. P." and "Pr." represent Stylization Effect, Content Preserving, and Preference, respectively. For each column, we report both of the vote percentages from professional participants and non-professional ones. We can see that GANST has the most votes for stylization, content preservation, and preference for non-

**Table 1:** *User Study. We report vote percentages of non-professional / professional participants in each column.*

| | S. E. | C. P. | Pr. |
|---|---|---|---|
| AdaIN | 0.145 / 0.119 | 0.193 / 0.199 | 0.175 / 0.160 |
| WCT | 0.164 / 0.138 | 0.140 / 0.119 | 0.092 / 0.101 |
| Avatar-Net | 0.117 / 0.146 | 0.148 / 0.150 | 0.100 / 0.097 |
| AAMS | 0.171 / 0.180 | 0.172 / 0.179 | 0.190 / 0.180 |
| ETNet | 0.188 / **0.210** | 0.146 / 0.145 | 0.173 / 0.195 |
| GANST | **0.215** / 0.207 | **0.201 / 0.208** | **0.270 / 0.267** |

**Table 2:** *Execution Time (Second)*

| | $256 \times 256$ | $512 \times 512$ | $1024 \times 1024$ |
|---|---|---|---|
| AdaIN | 0.015 | 0.043 | 0.108 |
| WCT | 0.152 | 0.210 | 0.519 |
| Avatar-Net | 0.127 | 0.287 | 0.578 |
| AAMS | 0.149 | 0.324 | 1.530 |
| ETNet | 8.740 | 31.25 | 80.35 |
| GANST | 0.033 | 0.083 | 0.178 |

professional users. In professional group, ETNet receives the most votes for stylization, while GANST receives most votes in content preservation and preference.

**Performance Evaluation.** To evaluate the performance of the proposed GANST, we compare the running time against our competitors on a workstation equipped with a Intel i7 8700K CPU and a NVIDIA GeForce GTX 1080 Ti GPU. Table 2 shows the statistics. The results are obtained via averaging over $1,000$ transfers. Though our method uses the same network as AdaIN, our method achieves the second because of different implementation to the original AdaIN. We believe that our method can be speeded up by a better implementation.

## 5. Conclusion

This paper extends the neural style transfer techniques to arbitrary styles with semantic contents preserved. The key essence of our approach GANST is that: (a) we extract the local gradients at multiple scales based on a novel Steerable Pyramid Neural Network (SPNN); and (b) we formulate the extracted information into Gradient Loss to train an arbitrary style transfer network. Our evaluation on various images demonstrates that GANST can efficiently produce results with concrete styles reflected and detailed content preserved when compared with existing state-of-the-art methods.

## References


[CJ83] CROSS G. R., JAIN A. K.: Markov random field texture models. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 1 (1983), 25–39. 2

[CMV15] CIMPOI M., MAJI S., VEDALDI A.: Deep filter banks for texture recognition and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2015), pp. 3828–3836. 2

[FA91] FREEMAN W. T., ADELSON E. H.: The design and use of steerable filters. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 9 (1991), 891–906. 2

[GEB15] GATYS L. A., ECKER A. S., BETHGE M.: A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576* (2015). 1, 2

[HB17] HUANG X., BELONGIE S.: Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision* (2017), pp. 1501–1510. 1, 2, 3, 4, 5

[JAFF16] JOHNSON J., ALAHI A., FEI-FEI L.: Perceptual losses for real-time style transfer and super-resolution. In *Proceedings of the European Conference on Computer Vision (ECCV)* (2016), Springer, pp. 694–711. 2

[JYF*17] JING Y., YANG Y., FENG Z., YE J., YU Y., SONG M.: Neural style transfer: A review. *arXiv preprint arXiv:1705.04058* (2017). 1, 2

[JYF*19] JING Y., YANG Y., FENG Z., YE J., YU Y., SONG M.: Neural style transfer: A review. *IEEE transactions on visualization and computer graphics* (2019). 1

[KB14] KINGMA D. P., BA J.: Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014). 4

[KSH12] KRIZHEVSKY A., SUTSKEVER I., HINTON G. E.: Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (2012), pp. 1097–1105. 2

[LFY*17] LI Y., FANG C., YANG J., WANG Z., LU X., YANG M.-H.: Universal style transfer via feature transforms. In *Advances in neural information processing systems* (2017), pp. 386–396. 2, 4, 5

[LMB*14] LIN T.-Y., MAIRE M., BELONGIE S., HAYS J., PERONA P., RAMANAN D., DOLLÁR P., ZITNICK C. L.: Microsoft coco: Common objects in context. In *European conference on computer vision* (2014), Springer, pp. 740–755. 4

[Low04] LOWE D. G.: Distinctive image features from scale-invariant keypoints. *International journal of computer vision 60*, 2 (2004), 91–110. 2

[LWLH17] LI Y., WANG N., LIU J., HOU X.: Demystifying neural style transfer. *arXiv preprint arXiv:1701.01036* (2017). 1

[Mal89] MALLAT S. G.: A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 7 (1989), 674–693. 2

[Nic16] NICHOL K.: Painter by numbers, wikiart. https://www.kaggle.com/c/painter-by-numbers, 2016. 4

[RDB16] RUDER M., DOSOVITSKIY A., BROX T.: Artistic style transfer for videos. In *German Conference on Pattern Recognition* (2016), Springer, pp. 26–36. 1

[SF95] SIMONCELLI E. P., FREEMAN W. T.: The steerable pyramid: A flexible architecture for multi-scale derivative computation. In *Proceedings., International Conference on Image Processing* (1995), vol. 3, IEEE, pp. 444–447. 3

[SLSW18] SHENG L., LIN Z., SHAO J., WANG X.: Avatar-net: Multi-scale zero-shot style transfer by feature decoration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 8242–8250. 2, 4, 5

[SWZ*19] SONG C., WU Z., ZHOU Y., GONG M., HUANG H.: Etnet: Error transition network for arbitrary style transfer. *arXiv preprint arXiv:1910.12056* (2019). 2, 4, 5

[Upw20] UPWORK: Upwork. https://www.upwork.com, 2020. 4

[WSZL19] WU H., SUN Z., ZHANG Y., LI Q.: Direction-aware neural style transfer with texture enhancement. *Neurocomputing 370* (2019), 39–55. 1, 2

[WWH06] WANG G., WONG T.-T., HENG P.-A.: Deringing cartoons by image analogies. *ACM Transactions on Graphics (TOG) 25*, 4 (2006), 1360–1379. 1

[YRX*19] YAO Y., REN J., XIE X., LIU W., LIU Y.-J., WANG J.: Attention-aware multi-stroke style transfer. *arXiv preprint arXiv:1901.05127* (2019). 2, 4, 5