Name your style: text-guided artistic style transfer

Zhi-Song Liu[1] Li-Wen Wang[2] Wan-Chi Siu[1] Vicky Kalogeiton[3]
[1]Caritas Institute of Higher Education [2]The Hong Kong Polytechnic University
[3]VISTA, LIX, Ecole Polytechnique, IP Paris

Abstract

Image style transfer has attracted widespread attention in the past years. Despite its remarkable results, it requires additional style images available as references, making it less flexible and inconvenient. Using text is the most natural way to describe the style. Text can describe implicit abstract styles, like styles of specific artists or art movements. In this work, we propose a text-driven style transfer (TxST) that leverages advanced image-text encoders to control arbitrary style transfer. We introduce a contrastive training strategy to effectively extract style descriptions from the image-text model (i.e., CLIP), which aligns stylization with the text description. To this end, we also propose a novel cross-attention module to fuse style and content features. Finally, we achieve an arbitrary artist-aware style transfer to learn and transfer specific artistic characters such as Picasso, oil painting, or a rough sketch. Extensive experiments demonstrate that our approach outperforms the state-of-the-art methods. Moreover, it can mimic the styles of one or many artists to achieve attractive results, thus highlighting a promising future direction.

1. Introduction

Image style transfer is a popular topic that aims to apply a desired painting style to an input content image. The transfer model requires the information of *"what content"* in the input image and *"which painting style"* to be used [17, 29]. Conventional style transfer methods require a content image accompanied by a style image to provide the content and style information [2, 7, 13, 24, 30]. However, people have specific aesthetic needs. Usually, finding a single style image that perfectly matches one's requirements is inconvenient or infeasible. Text or language is a natural interface to describe the preferred style. Instead of using a style image, using text to describe style preference is easier to obtain and more adjustable. Furthermore, achieving perceptually pleasing artist-aware stylization typically requires learning from collections of art, as one reference image is typically not representative enough. In this work, we learn arbitrary artist-aware image style transfer, which transfers the painting styles of any artist to the target image using texts and/or images. Most studies on universal style transfer [24,29] limit their applications using reference images as style indicators that are less creative or flexible. Text-driven style transfer has been studied [9, 17] and has shown promising results using a simple text prompt. However, these approaches require either costly data collection and labeling or online optimization for every content and style. Instead, our proposed Text-driven artistic aware Style Transfer model, TxST, overcomes these two problems and achieves better and more efficient stylization.

To obtain artist awareness, TxST explicitly explores the latent space using CLIP [25] feature representation: it maximizes the global distance amongst different artworks –and hence different artists–, while it minimizes the distance amongst the same artworks –hence same artists. Specifically, given artists' names, TxST projects feature from different artists onto the CLIP [25] space for classification.

Our contributions can be summarized as follows: (1) To achieve text-driven image style transfer, we propose to embed the task-agnostic image-text model, i.e., CLIP, into our network TxST. This enables TxST to obtain style preference from *images or text descriptions*, making the image style transfer more interactive. (2) We use a contrastive training strategy to learn art collection awareness, equivalent to a self-supervised classification. This exploits the inter-class similarity of different artists without explicit label guidance and data collection.

2. Related Work

Here, we introduce two related topics: arbitrary style transfer and text-driven style transfer.

Arbitrary style transfer. It can be split into two groups: (1) style-aware optimization [11, 15, 19, 32] and (2) universal style transfer [13, 30]. Inspired by the attention mechanism [34], a few works use it to explore statistical correlations. SANet [24] matches the content and style statistics via cross-attention. AdaAttN [21] further explores the second-order attention to preserve more content information without losing style patterns. Artflow [2]

and VAEST [23] explore the normalization flow [18] and VAE [14] to fuse style and content images. Recent approaches propose transformer-based style transfer [7] by using the vision transformer [8] for stylization.

Text-driven image style transfer. Style transfer is a subjective topic, that is, different people may have different preferences for stylization. Using style images as references may not be able to obtain sufficiently good results as texts can describe styles in a more abstract and aesthetic manner. The success of CLIP [25], VQVAE [33] and multimodality [3, 16, 22] show that text and image can be related via a shared projection space. Most recently, LDAST [9] uses the CLIP as the condition for style transfer that increases the cross-correlation between the output and text description for text-guided style transfer. Clipstyler [17] further develops this idea by using both global and patch CLIP losses to generate high-resolution stylized images. Instead, our proposed TxST does not require retraining for every style. Moreover, unlike LDAST, TxST uses Clip features from both images and texts to align with the target style and then use either image or text prompts for stylization.

3. Approach

The motivation of our proposed work is that text and image can work interchangeably in the CLIP space [25] for style transfer. In other words, text and image are co-linear in the CLIP space and hence, they can both be used as style indicators. From the aspect of style description, this co-linearity also exists between artists and their paintings, making it possible to realize artist-aware style transfer.

Overview. Inspired by this, we propose TxST. Given input content images I_c and desirable styles, artist's name t_s and corresponding painting I_s, TxST outputs the stylized image I_{cs}. At training, we use CLIP to maximize the variance of different styles by preserving the co-linearity between texts and style images. TxST computes the contrastive similarity [4] to minimize the inter-distance of the same styles. This training process is illustrated in Figure 1. Once the TxST is optimized, it collectively learns to align the specific artist's name to his/her paintings. At test time, either text (painter's name) or image (painter's work) can be input as style indicators for arbitrary style transfer.

Architecture. TxST consists of five parts: Image encoder, Image decoder, CLIP (text and image encoders), and style attention module. Following [23], structure of the encoder is the same as in VGG-19 [31], discarding the fully connected layers. The decoder is symmetric to the encoder, with gradual upsampling feature maps toward final stylized images. We use CLIP to encode the paired style texts ($t_s^i, i = 1, 2, .., C$, where C is the number of artists) and style images ($I_s^i, i = 1, 2, .., N$, where N is the number of

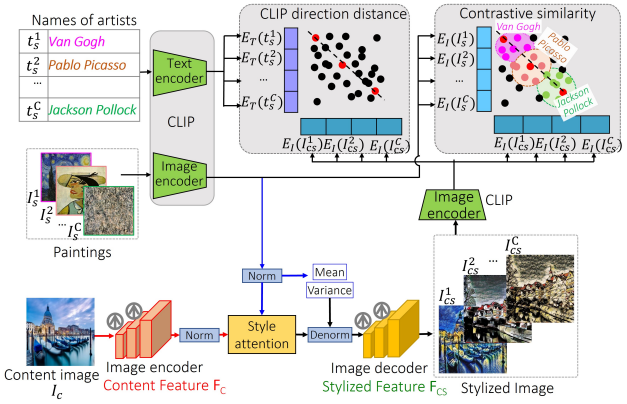

Figure 1. **Training process of TxST.** Given content images, artists' names and paintings, TxST learns stylized images so that they move towards the text/image prompts of the same artists and also away from the different artists. Then, we maximize the style's variance, as well as the similarity of the same style.

paintings), and obtain corresponding style features: $E_T(t_s^i)$ for text and $E_I(I_s^i)$ for the image. TxST uses $E_T(t_s^i)$ and $E_I(I_s^i)$ as style conditions to edit the content features. As shown in Figure 1, we normalize the style feature to obtain style mean and variance. Moreover, the style attention module P_R fuses content and style features to obtain stylized features $F_{cs} = \sigma(F_s) \times P_R(\bar{F}_s, \bar{F}_c) + \mu(F_s)$, where $\bar{F}_s, \bar{F}_c$ are normalized content and style features, $\mu(F_s)$ and $\sigma(F_s)$ are mean and variance of the style features.

• **Style attention module.** The core idea of using style attention for style transfer is to project style features onto the content ones via cross-attention [24]. Mathematically, we have:

$$
\begin{aligned}
&F_{\text{cs}} = \sigma(F_s) \cdot P_R(\bar{F}_s, \bar{F}_c) + \mu(F_s) \quad \text{, where} \\
&P_R(\bar{F}_s, \bar{F}_c) = \bar{F}_c + \text{Softmax}\left(\frac{Q[\bar{F}_c] \times K[\bar{F}_s]}{\sqrt{d}}\right) V[\bar{F}_s] \ ,
\end{aligned} \tag{1}
$$

where d is the dimension of the feature maps. Q, K and V are 1×1 convolution for query, key and value.

Loss functions. Here, we propose the losses we can use to train TxST.

• **Directional CLIP loss.** To guide the content image to follow the semantics of the target text (artist's name), we suggest directional CLIP loss L_{CLIP} from [10, 17]. This aligns the CLIP-space direction between the text-image pairs of the target t_s and outputs I_{cs}.

• **Contrastive similarity loss.** To encourage image features to be correlated with the target style and uncorrelated with other styles, we propose Contrastive similarity loss [4]. Given the i^{th} stylized image feature ν_i and N-1 other ν_j from the same batch: $\nu_1, \nu_2, \nu_{i-1}, \nu_{i+1}, ..., \nu_N$, we have:

$$L_{\text{sim}} = -\log \frac{\exp\left(S(\nu_i^I, \nu_j^I)/\tau\right)}{\sum_{\substack{j=1 \\ j\neq i}}^{N} \exp\left(S(\nu_i^I, \nu_j^I)/\tau\right)} - \log \frac{\exp\left(S(\nu_i^t, \nu_j^t)/\tau\right)}{\sum_{\substack{j=1 \\ j\neq i}}^{N} \exp\left(S(\nu_i^t, \nu_j^t)/\tau\right)}, \tag{2}$$

where τ is the temperature factor, S is the cosine similarity, ν_i^I is the stylized results using reference image I and ν_i^t is the stylized results using reference text t. Note that to distinguish from the CLIP vector $E_I(I_{cs})$, symbol ν is to represent the CLIP feature computed after Vision transformer [25] and before the final projection layer. In Equation (2), there are two loss terms: the first one computes the feature similarity between stylized results obtained by different style images (ν_i^I and ν_j^I); the second one computes the feature similarity between stylized results obtained by different style texts (ν_i^I and ν_j^t).

• **Content and style feature loss.** Following the existing style transfer methods [21, 23, 24], content and style feature losses are proposed by using a pre-trained VGG network to minimize the distance in the deep feature space. For style loss L_{style}, VGG-19 [31] is used to extract features (*relu1_2*, *relu2_2*, *relu3_4*, *relu4_1*) and compute the mean and variance differences. For content loss L_{content}, features from *relu2_2*, *relu3_4* are used. To preserve more content information, identity loss L_{ID} [21] is used for enhancement.

• **Total loss.** The final loss is $L = \lambda_{\text{CLIP}} L_{\text{CLIP}} + \lambda_{\text{sim}} L_{\text{sim}} + \lambda_{\text{style}} L_{\text{style}} + \lambda_{\text{content}} L_{\text{content}} + \lambda_{\text{ID}}\text{ID}$, where $\lambda_{\text{CLIP}}, \lambda_{\text{sim}}, \lambda_{\text{style}}, \lambda_{\text{con}}$ and λ_{ID} are coefficients to balance these loss components.

4. Experiments

4.1. Implementing Details

Datasets. We used the images from MS-COCO [20] for the image reconstruction task in the first training stage. In the second training stage, we trained TxST with MS-COCO [20] as our content set and WikiArt [1] as the style set. In the training phase, we loaded the images with size 512×512 and randomly cropped training patches of size 256×256.

Metrics. For evaluation, we propose several metrics: the VGG Score (style and content), the CLIP similarity score (style, content and F1), and the Deception Rate.

• **VGG Score.** Following VAEST [23], we estimated the VGG score using VGG features. Its style score is computed based on first and second orders (mean and variance) feature differences between reference and stylized images, while its content score is computed as the average feature loss between content and stylized images.

• **CLIP Similarity Score [17]** is computed with CLIP features. Its style score is the similarity between stylized image and {image, text} prompts: $s_{\text{style}}(s, I_{cs}) = \frac{s \cdot E_I(I_{cs})}{\|s\| \times \|E_I(I_{cs})\|}$, where $s = \{E_I(I_s), E_T(t_s)\}$ for reference image and text, respectively. Its content score is $s_{\text{content}}(I_c, I_{cs}) = \frac{E_I(I_c) \cdot E_I(I_{cs})}{\|E_I(I_c)\| \times \|E_I(I_{cs})\|}$. $F1$ score is widely used in binary classification to measure the harmonic mean of precision and recall [5].We measured the $F1$ score of the content and style scores $\{s_{\text{content}}(I_c, I_{cs}), s_{\text{style}}(s, I_{cs})\}$ as $F1 = 2 \times \frac{s_{\text{style}}(s, I_{cs}) \times s_{cont}(I_c, I_{cs})}{s_{\text{style}}(s, I_{cs}) + s_{\text{content}}(I_c, I_{cs})}$, where $s = \{I_s, t_s\}$ for image and text prompts, respectively.

• **Deception Rate.** For artist awareness, we followed AST [29] and computed the style transfer deception rate as the fraction of generated images classified by VGG-16 as the artworks of an artist for which the stylization was produced. Higher values mean closer to the artist's style.

• **Setting.** We compared against AST [29], MGAD [12], LDAST [9], CLIPstyler [17] (CLIPstyler(fast) and CLIPstyler(opti)), DALL-E-2 [26][1], VQGAN-CLIP [27][2] and Stable Diffusion [28][3]. For *text-driven style transfer*, we used the artist names as style descriptions (denoted as T_s).

4.2. Text-driven Artistic-aware Style Transfer

Quantitative Comparison. We measured the similarity to the content and artist in the CLIP feature space using the CLIP scores and the F1 score accordingly. Table 1 reports the results. We observe that CLIPstyler(fast) leads to the best content similarity score (0.736); however, the transferred images have poor artistic style performance. Results of AST have the second-best deception rate, but the similarity to the content image is only 0.538. CLIPstyler(opti), a slow but optimal version compared to CLIPstyler(fast), reaches good CLIP scores, but achieves the worst deception rate. This is expected since CLIPstyler(opti) requires dedicated training for each individual content and style image. Stable Diffusion leads to the best CLIP style score but its content score is low. Overall, our TxST provides a good balance between style and content, reaching the best performance on CLIP F1 score and on the deception rate, which demonstrates its effectiveness. We also note that CLIPstyler(opti) MGAD and VQGAN-CLIP need retraining for new artists and input images; in contrast, TxST does not need retraining for new styles.

Qualitative Comparison. Figure 2 shows the visual comparison of different methods for artist-aware style transfer. Note that AST does not require text input. For others, we used artists' names as texts to guide stylization. For reference, we also show three representative paintings in blue boxes. Our findings are summarized as: (1) our results show similar or better content preservation compared to AST and CLIPstyler. (2) TxST can faithfully mimic the signature styles of specific artists, such as the color tone and temperature patterns from *El-Greco, Van Gogh* and the distorted curves in *El-Greco* and *Van Gogh*. (3) TxST can maximize the visual differences among different artistic styles.

[1] https://openai.com/dall-e-2/
[2] https://github.com/nerdyrodent/VQGAN-CLIP
[3] https://beta.dreamstudio.ai/dream

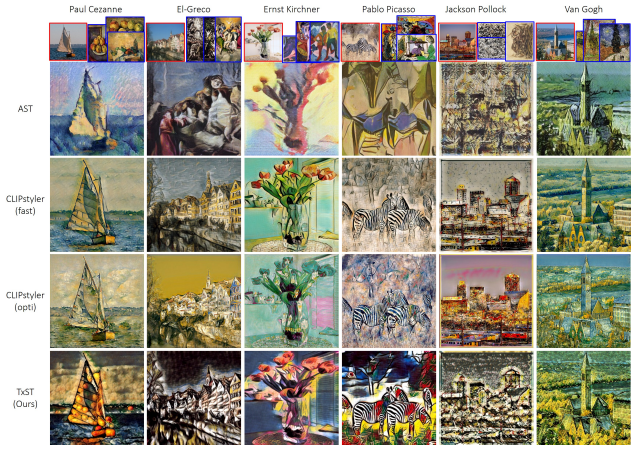

Figure 2. **Comparison among text-driven artist-aware style transfer methods.** TxST not only preserves the content details, but also successfully transfers the signature color and painting style of painters. Other methods either cannot preserve contents (AST), or learn incorrect painting styles (AST, Clipstyler).

Table 1. **Comparison with different Artistic Style Transfer methods.** (Red: best and Blue: 2^{nd} best).

Method	Clip Scores		Deception		Running time (s)
	Content↑	Style↑	F1↑	Rate↑	
AST [29]	0.538	0.269	0.359	0.664	1.3
CLIPstyler(fast) [17]	0.736	0.254	0.378	0.469	0.7
CLIPstyler(opti) [17]	0.624	0.306	0.410	0.441	220
LDAST [9]	0.669	0.270	0.385	0.435	1.6
MGAD [12]	0.397	0.203	0.269	0.339	604
VQGAN-CLIP [27]	0.557	0.230	0.326	0.682	240
DALL-E-2 [26]	0.665	0.228	0.340	0.425	34
Stable Diffusion [28]	0.542	0.332	0.412	0.702	37
Ours	**0.678**	**0.313**	**0.418**	**0.769**	**0.7**

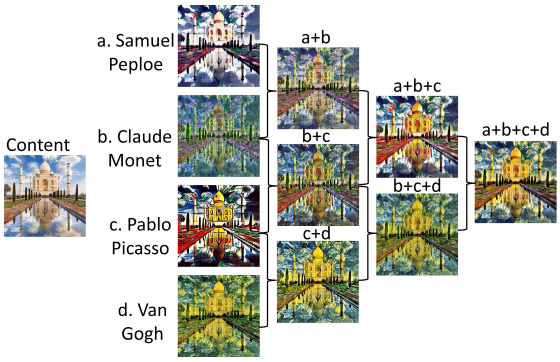

Figure 3. **Multiple style transfer.** We use four artists' names for style interpolation and observe successful multiple-style fusion.

Multiple style transfer. TxST can also achieve multiple style transfer. Unlike previous works [13, 23], it does not require multiple reference images for style fusion; instead, we can give a description that combines multiple styles. In Figure 3, we use four artists' names as input for style fusion: *Samuel Peploe, Claude Monet, Pablo Picasso, Van Gogh.*

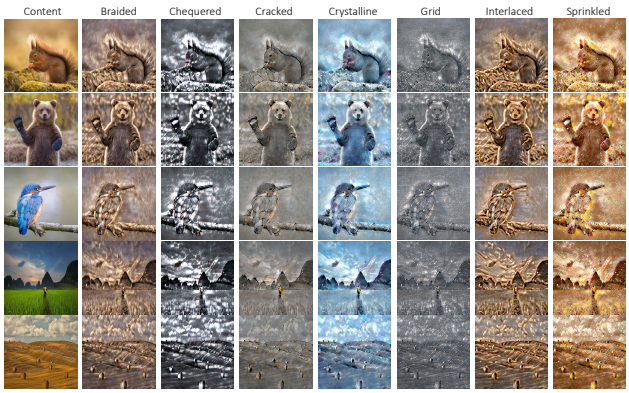

Figure 4. **Texture aware style transfer.** We train TxST on the DTD dataset [6], and then use texture language as style input for stylization. For instance, the edges around the images in column 2 (i.e., the tail of the squirrel and the feature of the bird) look like a 'braided pattern'.

We observe that the proposed TxST can combine different art styles successfully and visually pleasing.

Texture-aware style transfer. Similar to artist-aware style transfer, we train the proposed TxST using a texture dataset to learn texture-aware style transfer. DTD dataset [6] provides many texture samples, like *sprinkled*, *cracked* and so on. We use the texture languages as style input to train our model to examine if our proposed TxST can also learn texture-aware style transfer. In Figure 4, we observe that the proposed method can successfully and accurately transfer distinct texture features onto different content images.

5. Conclusion

In this paper, we proposed a text-driven approach for artist-aware style transfer, coined TxST. The CLIP-based contrastive training enables exploring the co-linearity between texts and images without requiring costly data collection and annotation. We conducted comprehensive experiments on text-driven artistic style transfer. Extensive results demonstrated that TxST achieves perceptually pleasing arbitrary stylization, revealing its ability to extract critical representations from the CLIP space and produce aesthetics close to the artists' works. Future work includes combining images, texts, and other cues to deliver a more flexible user-guided interactive style transfer framework.

Acknowledgements This work was supported by Caritas Institute of Higher Education (UGC/IDS(C)11/E01/20 and ISG200206), V. Kalogeiton's DIM RFSI grant, and the ANR APATE ANR-22-CE39-0016 project.

References

[1] K. nichol. painter by numbers, wikiart. https://www.kaggle.com/c/painter-by-numbers, 2016. 3

[2] Jie An, Siyu Huang, Yibing Song, Dejing Dou, Wei Liu, and Jiebo Luo. Artflow: Unbiased image style transfer via reversible neural flows. In *CVPR*, 2021. 1

[3] Federico Bianchi, Giuseppe Attanasio, Raphael Pisoni, Silvia Terragni, Gabriele Sarti, and Sri Lakshmi. Contrastive language-image pre-training for the italian language. *arXiv preprint arXiv:2108.08688*, 2021. 2

[4] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E. Hinton. Big self-supervised models are strong semi-supervised learners. In *NeurIPS*, 2020. 2

[5] Nancy Chinchor. Muc-4 evaluation metrics. In *Proceedings of the 4th Conference on Message Understanding*, USA, 1992. 3

[6] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi. Describing textures in the wild. In *CVPR*, 2014. 4

[7] Yingying Deng, Fan Tang, Weiming Dong, Chongyang Ma, Xingjia Pan, Lei Wang, and Changsheng Xu. Stytr 2: Image style transfer with transformers. In *CVPR*, 2022. 1, 2

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021. 2

[9] Tsu-Jui Fu, Xin Eric Wang, and William Yang Wang. Language-driven artistic style transfer. In *ECCV*, 2022. 1, 2, 3, 4

[10] Rinon Gal, Or Patashnik, Haggai Maron, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain adaptation of image generators. *arXiv preprint arXiv:2108.00946*, 2021. 2

[11] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *CVPR*, 2016. 1

[12] Nisha Huang, Fan Tang, Weiming Dong, and Changsheng Xu. Draw your art dream: Diverse digital art synthesis with multimodal guided diffusion. MM '22, page 1085–1094, 2022. 3, 4

[13] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*, 2017. 1, 4

[14] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In *arXiv preprint arXiv:1312.6114*, 2014. 2

[15] Nicholas Kolkin, Jason Salavon, and Gregory Shakhnarovich. Style transfer by relaxed optimal transport and self-similarity. In *CVPR*, 2019. 1

[16] Alexander Kuhnle and Ann Copestake. Shapeworld - a new test methodology for multimodal language understanding. *arXiv preprint arXiv:1704.04517*, 2017. 2

[17] Gihyun Kwon and Jong Chul Ye. Clipstyler: Image style transfer with a single text condition. *arXiv preprint arXiv:2112.00374*, 2021. 1, 2, 3, 4

[18] Jonas Köhler, Andreas Krämer, and Frank Noé. Smooth normalizing flows. In *NeurIPS*, 2021. 2

[19] Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Universal style transfer via feature transforms. *NeurIPS*, 2017. 1

[20] 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*, 2014. 3

[21] 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*, 2021. 1, 3

[22] Zhisong Liu, Robin Courant, and Vicky Kalogeiton. Funnynet: Audiovisual learning of funny moments in videos. In *ACCV*, 2022. 2

[23] Zhi-Song Liu, Vicky Kalogeiton, and Marie-Paule Cani. Multiple style transfer via variational autoencoder. In *ICIP*, 2021. 2, 3, 4

[24] Dae Young Park and Kwang Hee Lee. Arbitrary style transfer with style-attentional networks. In *CVPR*, 2019. 1, 2, 3

[25] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. 2021. 1, 2, 3

[26] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. 2021. 3, 4

[27] Nerdy Rodent. Vqgan-clip. https://github.com/nerdyrodent/VQGAN-CLIP, 2022. 3, 4

[28] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pages 10674–10685, 2022. 3, 4

[29] Artsiom Sanakoyeu, Dmytro Kotovenko, Sabine Lang, and Björn Ommer. A style-aware content loss for real-time hd style transfer. In *ECCV*, 2018. 1, 3, 4

[30] Artsiom Sanakoyeu, Dmytro Kotovenko, Sabine Lang, and Bjorn Ommer. A style-aware content loss for real-time hd style transfer. In *ECCV*, 2018. 1

[31] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 2, 3

[32] Dmitry Ulyanov, Vadim Lebedev, Andrea Vedaldi, and Victor Lempitsky. Texture networks: Feed-forward synthesis of textures and stylized images. 2016. 1

[33] Aaron van den Oord, Oriol Vinyals, and koray kavukcuoglu. Neural discrete representation learning. In *NeurIPS*, 2017. 2

[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 1