SpectralCLIP: Preventing Artifacts in Text-Guided Style Transfer from a Spectral Perspective

Zipeng Xu[1]* Songlong Xing[1]* Enver Sangineto[2] Nicu Sebe[1]
[1]University of Trento, Italy [2]University of Modena and Reggio Emilia, Italy
{zipeng.xu, songlong.xing, niculae.sebe}@unitn.it enver.sangineto@unimore.it

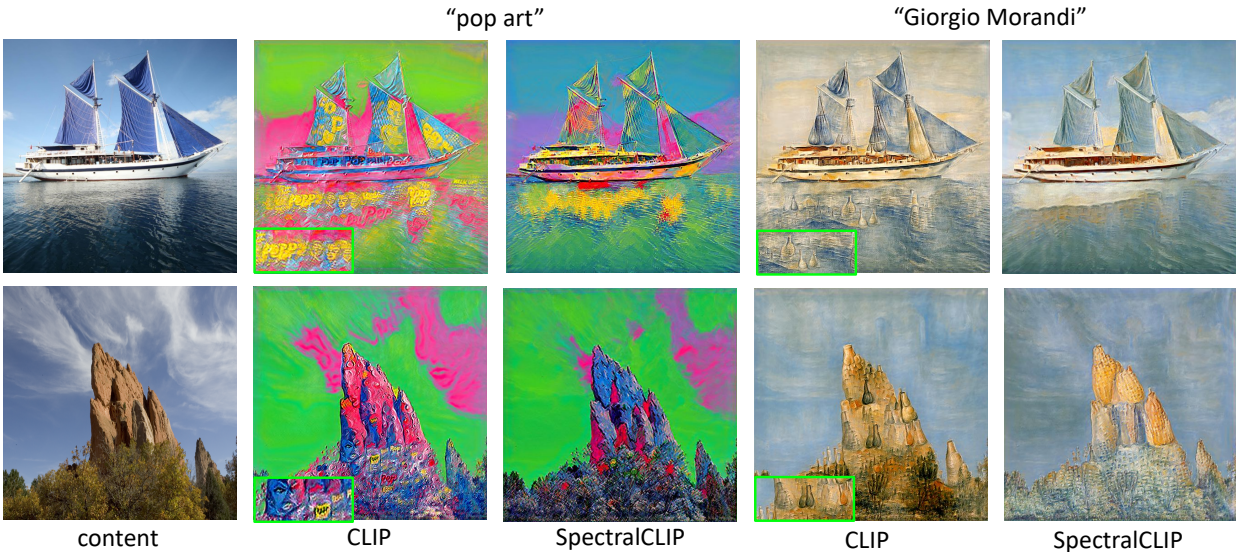

Figure 1. Text-guided style transfer results using either a CLIP-based image-text similarity or SpectralCLIP. While using the standard CLIP leads to the generation of undesirable artifacts (written words and unrelated visual entities as *highlighted*), the proposed SpectralCLIP prevents the artifacts while maintaining other style features.

Abstract

*Owing to the power of vision-language foundation models, e.g., CLIP, the area of image synthesis has seen recent important advances. Particularly, for style transfer, CLIP enables transferring more general and abstract styles without collecting the style images in advance, as the style can be efficiently described with natural language, and the result is optimized by minimizing the CLIP similarity between the text description and the stylized image. However, directly using CLIP to guide style transfer leads to undesirable artifacts (mainly written words and unrelated visual entities) spread over the image. In this paper, we propose SpectralCLIP, which is based on a spectral representation of the CLIP embedding sequence, where most of the common artifacts occupy specific frequencies. By masking the band including these frequencies, we can condition the generation process to adhere to the target style properties (*e.g.*, color, texture, paint stroke, etc.) while excluding the generation of larger-scale structures corresponding to the artifacts. Experimental results show that SpectralCLIP prevents the generation of artifacts effectively in quantitative and qualitative terms, without impairing the stylisation quality. We also apply SpectralCLIP to text-conditioned image generation and show that it prevents written words in the generated images. Our code is available at https://github.com/zipengxuc/SpectralCLIP.*

1. Introduction

Style transfer is about transforming the overall appearance of a given *content image* to adhere to a specific style while preserving its content. Starting from the pioneering paper of Gatys *et al*. [14], this task has attracted a grow-

*The two authors contributed equally to this paper.

ing interest in the scientific community because of its large application interest (*e.g.*, in the e-commerce or the entertainment industry, etc.). While most of the methods proposed so far extract the style information from a *reference image* [2, 3, 5, 8, 9, 14, 20, 27, 29, 30, 33, 41, 43, 44, 48] or a set of reference images [23, 38, 49], very recently, with the emergence of vision-language foundation models, a few approaches have started investigating the use of a *textual description* of the target style [13, 24, 47]. The main idea is to describe the target style with a natural language sentence (*e.g.*, "pop art") which is used to condition the content image transformation. The main advantage of this approach is that natural language sentences can describe more general and more abstract style characteristics that can hardly be extracted from a single reference image. Moreover, this way, it is possible to indirectly exploit the knowledge contained in the vision-language foundation model, which is usually pre-trained using hundreds of millions of image-text pairs.

Kwon *et al*. proposed CLIPstyler [24], which utilises the power of CLIP for text-guided style transfer for arbitrary images, demonstrating a broader range of styles and higher transfer quality than previous work based on reference images. However, as pointed out in [24], this method tends to generate images with over-specific artifacts. To distinguish, we define two types of artifacts: textual and visual artifacts. Visual artifacts are over-specific entities drawn on the generated image. In the example of Fig. 1, when the style is "pop art", CLIPstyler adds red lips and faces onto the image. Textual artifacts are written words, typically from the textual prompt describing the desired style, that appear on the generated image in an unwanted manner. Examples can also be seen in Fig. 1, where the word "pop" is spread over the generated image when CLIPstyler transfers the image with the style of "pop art". The presence of this type of artifact is largely due to the entanglement of visual concepts and written texts inherent in CLIP [31]. This entanglement issue in CLIP has been shown to be problematic and prevalent in a variety of CLIP application scenarios, including zero-shot classification and text-guided generation. The study and alleviation of effects resulting from such undesirable entanglement is a significant direction that has attracted increasing research attention [15, 25, 31].

In this paper, we propose to prevent artifact generation in text-guided style transfer using a spectral approach. Spectral analysis has been used to analyze temporal or spatial variations of signals [16]. Recently, researchers in the field of natural language processing (NLP) have proven its effectiveness in capturing linguistic information at different granular levels [32, 40]. Tamkin *et al*. [40] show that the sequence of textual tokens input to a Transformer network [42] contains structures at different scales: *e.g.*, the word scale, the sentence scale, the document scale, etc. These scales correspond to different frequencies in the changes of the values of the neurons' activations and can be isolated using the coefficients obtained by applying a DCT to the neuron activation sequence [40]. Intuitively, analogously to a sequence of linguistic tokens containing a hierarchy of semantic levels (*i.e*. from word level up to the document level), the constituent patches of an image also contain information at different levels. Similarly to [40], we sort the frequency components into several continuous bands. After analysing the patterns of artifacts present in CLIPstyler-generated stylised images, we find that these artifacts are highly related to certain frequency bands, and that by masking out these frequency components, we can remove the artifacts effectively without hurting the quality of stylised images. Hence, we propose SpectralCLIP to mask out those frequency bands, which implements a spectral filtering layer on top of the last layer of the CLIP vision encoder. We conduct experiments that verify the following points: (i) we experiment with many types of styles and find SpectralCLIP can effectively reduce both visual and textual artifacts while maintaining the target style well; (ii) we conduct a user study of 30 participants to compare the visual quality in terms of the overall style and the artifact-free performance of generated images, and find that our generated images are preferred by ***55.28%*** and ***74.44%*** of the users in terms of overall quality and artifact-free performance, respectively (Sec. 4.2); and (iii) we also leverage the 'learn-to-spell' CLIP (the CLIP subspace that focuses on written texts in the image) [31] to quantitatively validate that SpectralCLIP efficiently reduces textual artifacts (Sec. 4.1) as the score w.r.t. written texts is notably reduced. In addition, we employ SpectralCLIP for text-guided image generation (Sec. 4.4) and show it effectively prevents written words on the generated images.

To conclude, the contributions of this paper are:

- We propose SpectralCLIP to prevent both textual and visual artifacts in CLIP-guided style transfer. The effectiveness of SpectralCLIP has been verified on multiple styles through qualitative results, quantitative results, and a user study.

- SpectralCLIP is the first work to use spectral filtering in vision-language models. Other than solving the artifacts issues in CLIP-guided image style transfer, it also gives a new perspective on the disentangling of written texts and visual concepts in the CLIP space.

- To emphasize the generality of SpectralCLIP, we show that it can reduce the textual artifact generation also when used in a non style-transfer task and jointly with a completely different generator based on VQGAN.

2. Related Work

Reference Image-Based Style Transfer. After a few initial works on style transfer [11, 19], Gatys et al. [14] propose to use the Gram matrix of a convolutional network to represent the target style extracted from a single reference image. Following this paradigm, multiple aspects have been successively explored, ranging from *e.g.*, arbitrary style transfer [17, 20, 33], diversified style transfer [41, 44], attention mechanisms to fuse style and content [29, 33, 48], reducing artifacts [2, 5, 30, 43], increasing the content persistence [26, 45], and many others. However, it is tricky to *generalize* the description of an abstract style (*e.g.*, "pop art" or "warm and calm") from a single reference image. For this reason, a line of work is based on using a (large) *set* of reference images of the target style [4, 23, 38, 49]. In contrast, humans can usually understand a style using one or a few words, thanks to their knowledge and the relation they learned between words and visual appearance. Massively trained vision-language models like CLIP [36] now make it possible to emulate this human process, and text-guided style transfer approaches can avoid collecting a dataset for each target style, with a simple textual query.

CLIP-Guided Image Synthesis. The CLIP space has been largely used for image synthesis (*e.g.*, image generation [7, 35, 39] and image manipulation [1, 6, 34, 46]). However, when using CLIP for a text-guided style transfer task, a challenging aspect is to preserve the content while changing the style, since the style textual description usually does not contain any reference to this content. To solve this problem, Gal *et al*. [13] propose a directional loss and fine-tune a StyleGAN [21, 22] pre-trained using images of a specific domain. CLIPstyler [24] extends this approach to an open-domain scenario and uses multiple patches. However, the multi-patch directional loss leads to the generation of textual artifacts and "over-specification", where the latter refers to over-specific visual artifacts which locally remind of the textual description of the style [24] (Sec. 1). Most of the experiments shown in this paper are based on a CLIPstyler baseline, and we show that using SpectralCLIP for the directional loss computation, we can largely alleviate both the visual and textual artifact problem. Finally, Materzynska *et al*. [31] analyse the text-image entanglement problem in the CLIP space, and learn orthogonal projections ("forget-to-spell" and "learn-to-spell") of this space to disentangle the two modalities. We empirically show that using the "forget-to-spell" projection on a CLIPstyler baseline, we can indeed reduce the textual artifacts. In contrast, the proposed SpectralCLIP can reduce the generation of both the visual and the textual artifacts.

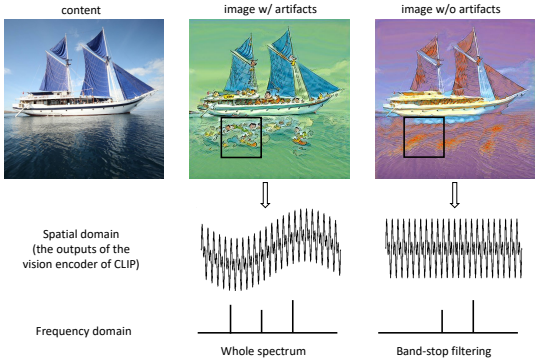

Figure 2. An illustration of SpectralCLIP. To transfer a "cartoon" style to the leftmost content image, CLIPstyler generates many cartoon-like artifacts, spreading over the whole image (central figure). The corresponding spectral representation is a composition of frequencies with different periods. Removing the frequencies corresponding to the artifact scales (SpectralCLIP) prevents the generation of these unwanted artifacts (right figure).

3. Method

SpectralCLIP is based on computing a text-image similarity using a frequency filter of the CLIP representations. In this section, we first describe how this filtering is obtained (Sec. 3.1), and then we show how it can be plugged into existing text-based generative approaches (Sec. 3.2), and finally how the band filters are selected (Sec. 3.3).

3.1. Spectral based filtering

Given an image I, we use the CLIP vision encoder $E_v(\cdot)$ to represent I with a grid of $k \times k$ vectors which either are extracted from the last convolutional layer of a ResNet [18] or correspond to the final embeddings of a Vision Transformer [10]. Our method is independent of the specific encoder architecture and can be applied to both types of networks. In the experiments of this paper, following [24], we used a ViT-B/32 [10], pre-trained by the authors of CLIP and then frozen. Since a ViT-based encoder also includes a class token, we get $n = 1+k^2$ vectors, which we flatten into a sequence: $V = E_v(I)$, where $V = \{\boldsymbol{v}_0, ..., \boldsymbol{v}_i, ..., \boldsymbol{v}_{n-1}\}$. Despite each $\boldsymbol{v}_i \in \mathbb{R}^d$ being a d-dimensional vector, following [40], a spectral representation of V can be obtained separately considering each dimension j ($0 \leq j \leq d-1$), which, in our case, corresponds to the j-th channel of the CLIP embedding. For a given j, if $x_i^{(j)} = \boldsymbol{v}_i[j]$ is the j-th component of $\boldsymbol{v}_i$, the corresponding (scalar-valued) sequence $X^{(j)} = \{x_0^{(j)}, ..., x_{n-1}^{(j)}\}$ can be represented in the frequency domain using, *e.g.*, the DCT-II variant of DCT:

$$f_m^{(j)} = \sum_{i=0}^{n-1} x_i^{(j)} \cos\left[\frac{\pi m}{n}(i+1/2)\right], \qquad (1)$$

where $m = 0, ..., n-1$, and $f_m^{(j)}$ is the coefficient of the m-th frequency and represents the contribution of the cor-

responding cosine wave in the "signal" discretely sampled by the sequence $X^{(j)}$. Different frequencies describe different cosine waves, and each frequency period (i.e., the number of elements of $X^{(j)}$ it takes to complete a full cycle) corresponds to the scale of the change in the activation of the j-th neuron. Tamkin *et al*. [40] observe that, in the natural language processing domain, these scale changes usually correspond to different structures contained in the input document: *e.g*., the single word structure corresponds to the highest frequencies, while medium-level frequencies correspond to sentences, etc. Analogously, since, in our domain, artifacts are relatively large-scale visual structures appearing repeatedly throughout the image (Sec. 1), we separate those frequencies most likely corresponding to the artifacts from the other frequencies containing useful style information (texture, color, etc.). To do so, we use a *band-stop* filter (see Fig. 2), inspired by periodic noise removal techniques in spectral-based image processing [16].

Concretely, we stack all the frequencies of all the d channels in a single $d \times n$ matrix F, where the j-th row F_j contains the n DCT coefficients in Eq. (1). Then, for each target style, we define a binary filter $\boldsymbol{b} \in \{0,1\}^n$, which contains zero elements only in specific bands (see Sec. 3.3 for details). We use $\boldsymbol{b}$ to zero out those columns in F which should be filtered:

$$S = F \odot M[\boldsymbol{b}], \tag{2}$$

where $\odot$ is the Hadamard product, and $M[\boldsymbol{b}]$ is a $d \times n$ matrix in which all the elements are ones except those corresponding to the columns in $\boldsymbol{b}$, which are zeros.

S is the spectral representation of V, in which frequencies $\boldsymbol{b}$ are *ignored*. Note that, differently from [40], where the spectrum of each neuron is individually filtered, in our case $\boldsymbol{b}$ is uniformly used for all the d dimensions. This is because an artifact is a complex visual structure, most likely simultaneously involving different dimensions of the CLIP space. S is finally back-projected into the original CLIP space using the inverse DCT (IDCT), obtaining $\hat{V} = \{\hat{\boldsymbol{v}}_0, ..., \hat{\boldsymbol{v}}_{n-1}\}$. $\hat{V}$ is a representation of I which can be used jointly with different metrics (*e.g*., the Euclidean metric or a cosine similarity, etc.) to compute a CLIP-based similarity between images or between images and text *which is not influenced by the frequencies in* $\boldsymbol{b}$. This way, we can condition the generation process using a textual sentence (Sec. 3.2) while simultaneously ignoring those frequencies corresponding to the artifact generation.

3.2. Computing an image-text similarity

In this section, we show how the proposed spectral-based filtering of an image representation (Sec. 3.1) can be used to condition a generative process and plugged into existing text-conditioned generative frameworks with negligible modifications of the original approaches. In our experiments, we use both CLIPstyler [24] (a state-of-the-art text-guided style transfer method, see Secs. 1 and 2) and the VQGAN+CLIP method [7] adopted in [31].

VQGAN+CLIP [7] is a text-to-image generation approach based on VQGAN discrete latent codes [12]. The latter are randomly sampled and then optimized using the cosine similarity between the CLIP embedding of the generated image ($\boldsymbol{z}_v$) and the CLIP embedding of a textual prompt ($\boldsymbol{z}_t$). The only thing we need to change to use SpectralCLIP in this framework is the image representation. To do so, we use $\boldsymbol{z}_v = \hat{\boldsymbol{v}}_0$, where $\hat{\boldsymbol{v}}_0$ is the representation of the class token in $\hat{V}$ (Sec. 3.1). Note that $\hat{\boldsymbol{v}}_0 \neq \boldsymbol{v}_0$ because the frequencies in $\boldsymbol{b}$ have been removed. Other possible choices can be, *e.g*., using an average pooling of $\hat{V}$ or a linear projection of the concatenation of all the elements of $\hat{V}$ into a vector of the same dimensions as $\boldsymbol{z}_t$. Following [7] we use the class token which is a simple and effective solution.

CLIPstyler [24] is based on a U-Net generator [37] which, given a content image I_c as input, generates a style-transferred image I_s. To condition the generation process on a style textual description s (where s is a natural language sentence), the embedding of s, obtained using the textual CLIP encoder, is compared with the textual embedding of a fixed sentence ("Photo"). The difference between these two textual embeddings should have the same direction of the difference between the visual embedding of I_c and I_s. This *directional CLIP loss*, initially proposed in [13], is further developed in CLIPstyler by introducing patch-level comparisons. We adopt exactly the same framework, and the only necessary change to use SpectralCLIP in CLIPstyler is to replace the standard CLIP visual embedding of an image (or an image patch) with our filtered representation. Since in CLIPstyler an image/image patch is represented using the class token, we analogously use the class token extracted from $\hat{V}$ (i.e., $\boldsymbol{z}_v = \hat{\boldsymbol{v}}_0$).

band	Frequency index	Period (tokens)
b_1	0-1	25-∞
b_2	2-3	7-25
b_3	4-7	4-7
b_4	8-15	2-4
b_5	16-49	1-2

Table 1. Correspondences between the bands, frequency indexes and period (tokens). The corresponding periods are approximate numbers of tokens that are needed to complete a cosine wave cycle.

3.3. Band selection

So far we have assumed that we can associate a frequency filter $\boldsymbol{b}$ to a given textual description of a style (in CLIPstyler) or to a textual prompt (in VQGAN+CLIP). However, since selecting the best $\boldsymbol{b} \in \{0,1\}^n$ would be

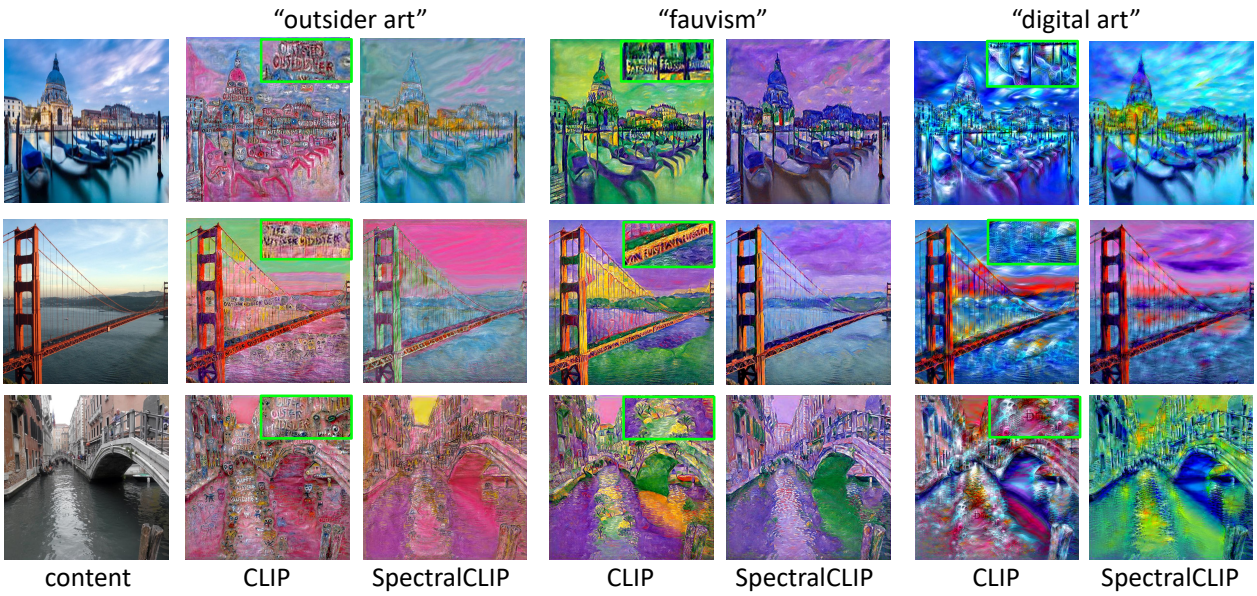

Figure 3. Style transfer results using either the CLIP or SpectralCLIP to condition the generation. SpectralCLIP effectively prevents artifact generation while achieving similar style features, e.g., color, paint stroke (artifacts are *highlighted*).

intractable, we adopt a simpler solution, inspired by [40], where the whole spectrum of frequencies $(0, ..., n-1)$ is split in fixed bands, each being a contiguous interval of frequencies. Note that each index m of the DCT has a frequency of $2m$ (see Eq. (1)), and that given a sequence of length N, the period is $N/2m$ (it takes $N/2m$ tokens to complete a cycle). For example, in our case where ViT-B/32 is used as the vision encoder, the sequence length is $N = (224/32)^2 + 1 = 50$. Therefore, index 1 of the DCT corresponds to the period of 25, and index 5 to the period of 5, etc. Specifically, we define the following 5 bands: $b_1 = [0, 1]$, $b_2 = [2, 3]$, $b_3 = [4, 7]$, $b_4 = [8, 15]$, $b_5 = [16, 49]$. The correspondence between these frequency bands and periods is presented in Tab. 1. In ViT-B/32, the input image is first resized to the resolution of 224^2, and then divided into 7×7 image patches of 32^2. In this sense, it can be estimated that b_1 relates to artifacts that roughly span more than 3 lines of patches, b_2 to those spanning 2 lines, and $b_3 - b_5$ to small artifacts spanning within 1 line.

Another problem is that, given a style description, the artifact condition is unpredictable. Specifically, it is difficult to judge if the stylized image contains artifacts or not, as well as to detect the artifact appearance. Nevertheless, through experiments on various styles, we find the artifacts are usually at three scales. Therefore, we propose a simple yet effective method based on empirical studies. Through experimenting on multiple band combinations, we find three filtering strategies ($c_1 = \{b_1, b_2, b_4\}$, $c_2 = \{b_1, b_2\}$, $c_3 = \{b_1\}$) that are effective for preventing the artifacts at the corresponding three scales, respectively. For instance, using c_1, the associated filter $\boldsymbol{b}$ contains ones in the intervals b_1, b_2, b_4, and it is used in Eq. (2) to zero out the corresponding bands. We use visual inspection to select the band combination that leads to the best result, then it is used in all image stylisation conditioned on s. This selection step is done *only once per given style* s using a single image content. More details are provided in Appendix A.

4. Experiments

4.1. Style Transfer Results

In this section, we evaluate SpectralCLIP in a text-guided style transfer task. For a fair comparison with CLIPstyler [24], we use its same network, loss functions, training protocols, hyperparameters, etc., changing *only* the basic image-text similarity as described in Sec. 3.2.

Qualitative results. In Fig. 1 and Fig. 3, we show multiple text-guided style transfer results generated using either the standard CLIP space (i.e., the original CLIPstyler method) or our SpectralCLIP. These images show that CLIPstyler frequently generates visual and textual artifacts. By contrast, the results generated using SpectralCLIP do not have the issues of both visual artifacts and textual artifacts while the styles are presented well. Take our results of “outsider art” (Fig. 3) for example, the style shown in the images is similar to the style in the results of CLIPstyler, but with artifacts excluded. More qualitative results are provided in Appendix E. Additionally, results of non-artistic concrete styles (*e.g.*, fire) are also provided in Appendix F.

Quantitative results. Quantitatively evaluating a style-transfer approach is difficult because of the lack of a uni-

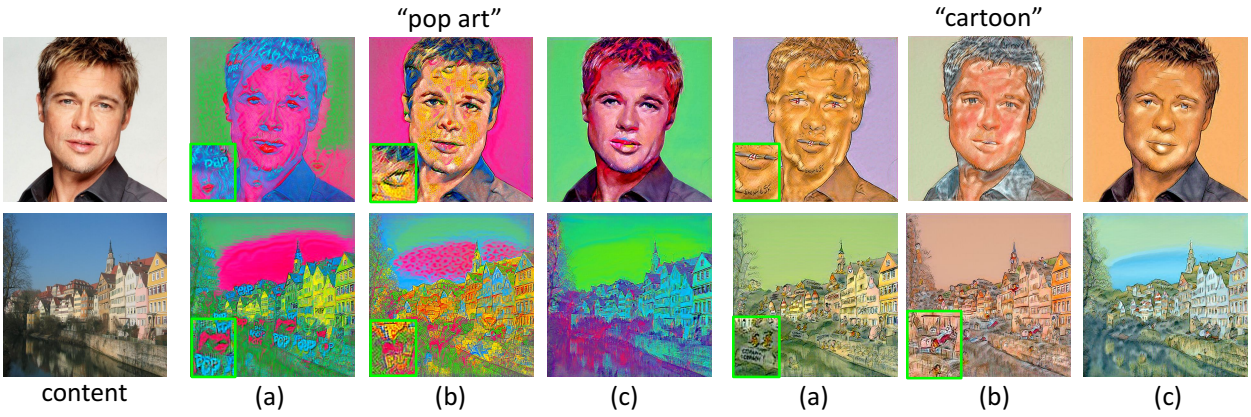

Figure 4. Comparisons among text-guided style transfer results, generated using CLIPstyler with (a) CLIP, (b) forget-to-spell CLIP [31], and (c) SpectralCLIP, respectively (artifacts are *highlighted*).

versally accepted metric that can assess the reflection of a target style in a generated image. On the other hand, using a cosine similarity between the CLIP-based representations of the target style and the generated image may favour those methods (such as CLIPstyler) which use the same similarity in the optimization stage. To partially solve this problem, we use an additional metric, based on the “learn-to-spell” projection of the CLIP space proposed (and trained) in [31] (Sec. 2). The idea behind this metric is that since part of the artifacts have a textual nature (i.e., strings drawn on the generated images, Secs. 1 and 2), then a learn-to-spell based similarity between a generated image and the corresponding textual description of the style should be *higher* for those images containing *more* textual artifacts. We provide more discussion of this metric in Appendix D.

Concretely, we sample 100 images from the COCO [28] val-set and use them as content images. Then, for each style, we generate the style transfer results using either SpectralCLIP or CLIPstyler. Finally, we use both the original CLIP space and learn-to-spell CLIP projection [31] to evaluate the similarity between the textual style description and the generated images. Tab. 2 shows that using SpectralCLIP, the learn-to-spell CLIP score is significantly reduced, indicating that SpectralCLIP effectively prevents textual artifact generation. On the other hand, the CLIP similarity is also reduced; however, as aforementioned, this metric is biased towards CLIPstyler, where the whole, non-filtered CLIP image representation is used for optimization.

4.2. Comparison with Forget-to-Spell CLIP

In this section, we compare SpectralCLIP with “forget-to-spell” CLIP [31], which is a learned subspace of CLIP semantic space that alleviates the text-image entanglement problem (Sec. 1). Specifically, we again use CLIPstyler as the baseline and we replace its (standard) CLIP space with the forget-to-spell projection proposed and trained in [31]. Hence, we compare three methods: (a) CLIPstyler with CLIP (i.e., the original CLIPstyler), (b) CLIPstyler with the forget-to-spell CLIP, and (c) CLIPstyler with SpectralCLIP.

		CLIPstyler w. CLIP	CLIPstyler w. SpectralCLIP
cartoon	*CLIP*	0.269 ±0.014	0.256 ±0.014
	CLIP-Spell	0.482 ±0.056	0.441 ±0.066
pop art	*CLIP*	0.315 ±0.023	0.287 ±0.018
	CLIP-Spell	0.419 ±0.137	0.353 ±0.121
visionary art	*CLIP*	0.322 ±0.016	0.278 ±0.018
	CLIP-Spell	0.527 ±0.086	0.397 ±0.057
outsider art	*CLIP*	0.314 ±0.018	0.255 ±0.019
	CLIP-Spell	0.571 ±0.108	0.360 ±0.095

Table 2. Average cosine similarity between the stylized images and the textual description of the style, measured both on the original CLIP space (↑) and on the learn-to-spell CLIP (in short referred to as CLIP-Spell) (↓).

(%)	w. CLIP	w. forget-to-spell	w. SpectralCLIP
Overall	34.44	10.28	**55.28**
Artifact-Free	19.45	6.11	**74.44**

Table 3. User preference of the three style transfer methods with respect to the overall quality of the generated images (↑) and the presence of artifacts (↑).

Qualitative results. From the qualitative results shown in Fig. 4, we draw three conclusions: 1) the images generated by the original CLIPstyler contain both visual and textual artifacts; 2) using the forget-to-spell CLIP alleviates the textual artifact issue, but it still generates visual artifacts, which makes the results unlike human created artworks; and 3) similar to the analysis in Sec. 4.1, using SpectralCLIP, no visual nor textual artifacts have been generated, improving the overall quality of the results.

User Study. We further compare the three methods through

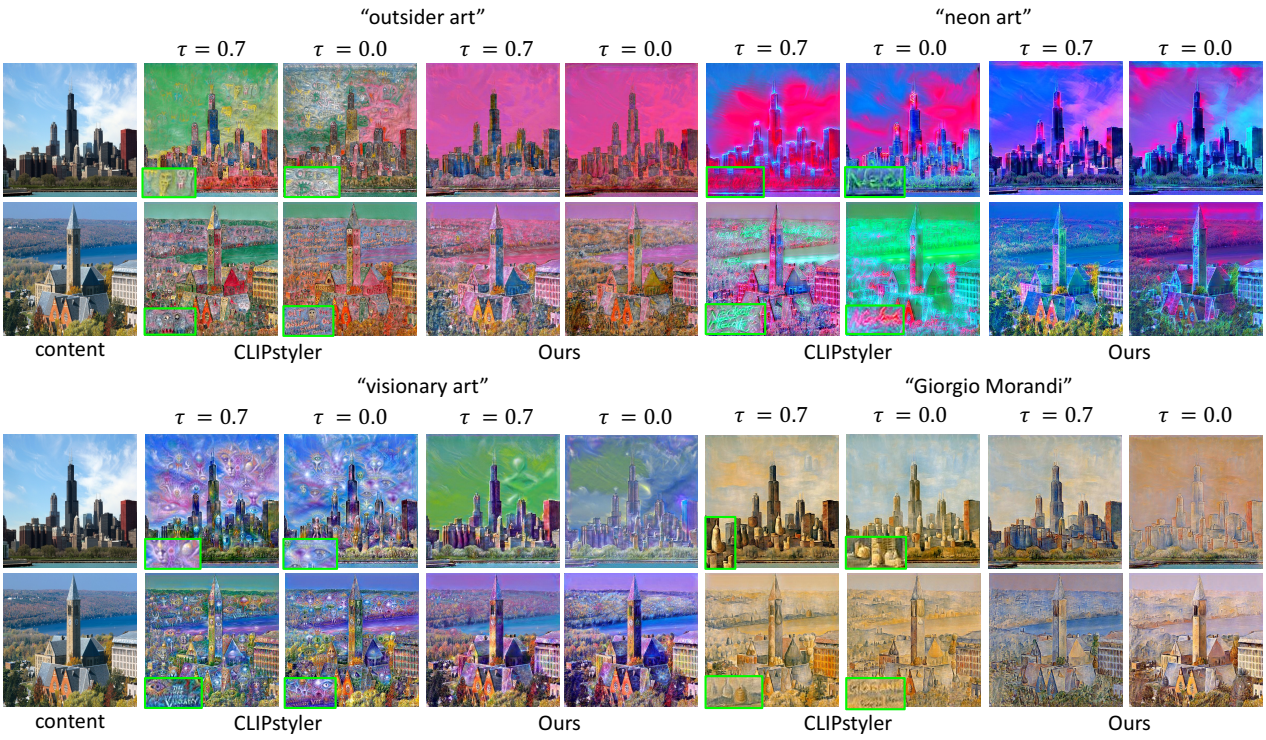

Figure 5. The effect of the patch-rejection threshold τ in the CLIPstyler patch-loss (artifacts are *highlighted*, zoom in to see details).

a user study. Specifically, we use 10 styles jointly with two different tasks which respectively analyse: 1) the overall quality of the generated images, asking the users to assess whether the stylized results are consistent with the target style, the content is well preserved, and no inharmonious artifacts are generated; and 2) the specific artifact issue, asking the users to assess the generated images regarding the possible presence of visual/textual artifacts. We randomly sampled 100 content images from the COCO val-set and then created a questionnaire with 24 questions. We recruited 30 users, who were asked to select one out of the three methods for each content image. The user preference results, as reported in Tab. 3, show that SpectralCLIP gets the best scores on both tasks. Specifically, SpectralCLIP achieves a significantly higher preference score (77.44%) in the artifact-free evaluation, indicating the effectiveness of our proposal in preventing artifacts. More details about the user study are provided in Appendix B.

4.3. Hyperparameter Study

Threshold in the patch loss. CLIPstyler uses a patch-rejection threshold τ in its patch loss to alleviate the over-specification problem (Sec. 2). The value of this threshold is a (manually selected) hyperparameter, which is fixed to $\tau = 0.7$ in [24]. In Fig. 5 we compare the use of this threshold ($\tau = 0.7$) with a non-thresholding variant ($\tau = 0.0$). The results show that the original CLIPstyler is heavily influenced by the thresholding, since its removal ($\tau = 0.0$) leads to the generation of many more artifacts, independently of the target style. By contrast, SpectralCLIP is much less sensitive to this thresholding, since the results using $\tau = 0.0$ are consistent with the images generated with $\tau = 0.7$, showing that our method does not rely on this thresholding step to avoid over-specification.

Band selection. We study the effect of the band selection (Sec. 3.3) using the style "visionary art" (Fig. 6), jointly with five different filters: (i) masking bands 1, 2 and 4 (corresponding to c_1 in Sec. 3.3); (ii) masking bands 1, 2 and 5; (iii) masking bands 1 and 2 (c_2 in Sec. 3.3); (iv) masking bands 1 (c_3 in Sec. 3.3); and (v) only masking the lowest frequency (i.e., the frequency index $m = 0$). Fig. 6 shows that the visual appearance change caused by higher frequencies tends to be more local, as can be observed by comparing (i) with (ii), and (ii) with (iii). For instance, comparing (i) with (ii), the former filter leads to greater visual appearance changes within a larger region. Moreover, the differences between (ii) and (iii) are marginal, indicating that band 5 (which includes the highest frequencies) is related to changes in smaller areas. Furthermore, lower frequencies result in the generation of larger artifacts, as shown by the comparison between (iii) and (iv), and between (iv) and (v). For example, not masking band 2 leads to the generation

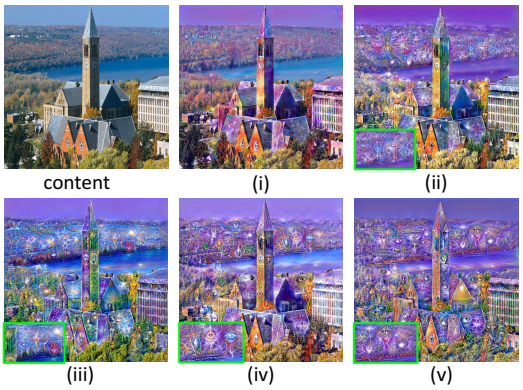

Figure 6. The effects of different band-stop filters (artifacts are *highlighted*).

of larger and more obvious artifacts on the eyes. A similar phenomenon can be observed when we additionally do not mask the frequency with index 1 (see (iv) and (v)). These results confirm our assumption (Sec. 1) that most artifacts are visual structures with a specific period, whose generation can be prevented by adopting the proposed spectral representation and the corresponding frequency filters.

4.4. Text-to-Image Generation

To test the generality of SpectralCLIP, we consider a different task, adopting the text-to-image generation framework used in [31] to evaluate the word-image disentanglement of forget-to-spell and learn-to-spell (Secs. 1 and 4.2). Specifically, following [31], we use VQGAN+CLIP [7] and we replace its text-image similarity computed on the original CLIP space with SpectralCLIP. Fig. 7 compares the results obtained with VQGAN+CLIP and VQGAN+SpectralCLIP, and confirms the observations of Materzynska *et al*. [31], who highlight that VQGAN+CLIP frequently generates inappropriate textual strings (textual artifacts) mixed with visual content. By contrast, this problem is largely alleviated with SpectralCLIP. Meanwhile, the generated image content in VQGAN+SpectralCLIP is still consistent with the given text prompt (except for the nonsense text input "irmin"). In all the VQGAN+SpectralCLIP results shown in Fig. 7, we use the same filtering strategy (masking only band 4, i.e., b_4, see Sec. 3.3). Note that the scale of a textual/visual artifact depends on the CLIP encoder input, which is the full image in the case of VQGAN, and this results in a shorter period with respect to CLIPstyler.

Conclusion. Despite the wide success of vision-language foundation models like CLIP in different vision-language tasks, directly using CLIP for style transfer suffers from the generation of visual and textual artifacts. To resolve this problem, we propose SpectralCLIP, which transforms

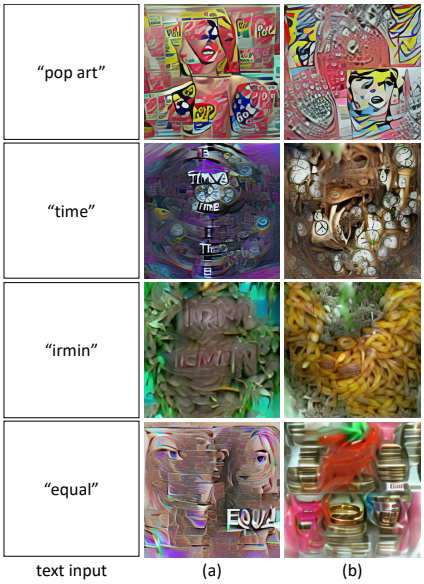

Figure 7. Text-to-image generation results using (a) VQGAN+CLIP, and (b) VQGAN+SpectralCLIP.

the CLIP embedding sequence into the frequency domain and filters those frequencies whose period corresponds to the artifact scales. Experimental style transfer results show that SpectralCLIP significantly mitigates artifact generation, thus improving the realistic degree and the quality of the generated images.

Limitations. Despite the promising results of SpectralCLIP, there are still some limitations. Firstly, we empirically analyse the artifact patterns present in a range of artistic styles, and mask out certain bands using one of the three general filters. The reason why a certain target style tends to produce different scales of artifacts is still unclear. This may require a deeper understanding of how CLIP captures these artistic concepts when it was pre-trained. Secondly, this work defines three general band combinations that effectively produce cleaner stylised images. A more promising alternative for future work is to automatically select frequency bands that cater to a target style. Recently, in the language domain, Müller-Eberstein *et al*. [32] promote [40] and develop learnable filters rather than handcrafted ones, offering an intriguing direction to follow. To this end, for image style transfer, a widely recognised metric to measure the presence of artifacts is still missing.

Acknowledgment. This work was supported by the MUR PNRR project FAIR (PE00000013) funded by the NextGenerationEU, by the PRIN project CREATIVE (Prot. 2020ZSL9F9), and by the EU Horizon project ELIAS (No. 101120237).

References

[1] David Bau, Alex Andonian, Audrey Cui, YeonHwan Park, Ali Jahanian, Aude Oliva, and Antonio Torralba. Paint by word. *arXiv preprint arXiv:2103.10951*, 2021. 3

[2] Haibo Chen, Zhizhong Wang, Huiming Zhang, Zhiwen Zuo, Ailin Li, Wei Xing, Dongming Lu, et al. Artistic style transfer with internal-external learning and contrastive learning. *Advances in Neural Information Processing Systems*, 34:26561–26573, 2021. 2, 3

[3] Haibo Chen, Lei Zhao, Zhizhong Wang, Huiming Zhang, Zhiwen Zuo, Ailin Li, Wei Xing, and Dongming Lu. Dualast: Dual style-learning networks for artistic style transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 872–881, June 2021. 2

[4] Haibo Chen, Lei Zhao, Zhizhong Wang, Huiming Zhang, Zhiwen Zuo, Ailin Li, Wei Xing, and Dongming Lu. Dualast: Dual style-learning networks for artistic style transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 872–881, 2021. 3

[5] Jiaxin Cheng, Ayush Jaiswal, Yue Wu, Pradeep Natarajan, and Prem Natarajan. Style-aware normalized loss for improving arbitrary style transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 134–143, 2021. 2, 3

[6] Myungsub Choi. Referring object manipulation of natural images with conditional classifier-free guidance. In *European Conference on Computer Vision*, pages 627–643. Springer, 2022. 3

[7] Katherine Crowson, Stella Biderman, Daniel Kornis, Dashiell Stander, Eric Hallahan, Louis Castricato, and Edward Raff. Vqgan-clip: Open domain image generation and editing with natural language guidance. In *European Conference on Computer Vision*, pages 88–105. Springer, 2022. 3, 4, 8

[8] Yingying Deng, Fan Tang, Weiming Dong, Chongyang Ma, Xingjia Pan, Lei Wang, and Changsheng Xu. Stytr2: Image style transfer with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11326–11336, June 2022. 2

[9] Yingying Deng, Fan Tang, Weiming Dong, Chongyang Ma, Xingjia Pan, Lei Wang, and Changsheng Xu. Stytr2: Image style transfer with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11326–11336, 2022. 2

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 3

[11] Alexei A Efros and William T Freeman. Image quilting for texture synthesis and transfer. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 341–346, 2001. 3

[12] Patrick Esser, Robin Rombach, and Björn Ommer. Taming transformers for high-resolution image synthesis. In *CVPR*, 2021. 4

[13] Rinon Gal, Or Patashnik, Haggai Maron, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain adaptation of image generators. *ACM Transactions on Graphics (TOG)*, 41(4):1–13, 2022. 2, 3, 4

[14] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2414–2423, 2016. 1, 2, 3

[15] Gabriel Goh, Nick Cammarata, Chelsea Voss, Shan Carter, Michael Petrov, Ludwig Schubert, Alec Radford, and Chris Olah. Multimodal neurons in artificial neural networks. *Distill*, 2021. 2

[16] R.C. Gonzalez and R.E. Woods. *Digital Image Processing, fourth edition*. Pearson, 2018. 2, 4

[17] 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*, pages 8222–8231, 2018. 3

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 3

[19] Aaron Hertzmann, Charles E Jacobs, Nuria Oliver, Brian Curless, and David H Salesin. Image analogies. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 327–340, 2001. 3

[20] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE international conference on computer vision*, pages 1501–1510, 2017. 2, 3

[21] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019. 3

[22] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8110–8119, 2020. 3

[23] Dmytro Kotovenko, Artsiom Sanakoyeu, Sabine Lang, and Bjorn Ommer. Content and style disentanglement for artistic style transfer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4422–4431, 2019. 2, 3, 11

[24] Gihyun Kwon and Jong Chul Ye. Clipstyler: Image style transfer with a single text condition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18062–18071, 2022. 2, 3, 4, 5, 7

[25] Yoann Lemesle, Masataka Sawayama, Guillermo Valle Pérez, Maxime Adolphe, Hélène Sauzéon, and Pierre-Yves Oudeyer. Language-biased image classification: evaluation based on semantic representations. In *ICLR*, 2022. 2

[26] Xueting Li, Sifei Liu, Jan Kautz, and Ming-Hsuan Yang. Learning linear transformations for fast image and video style transfer. In *Proceedings of the IEEE/CVF Conference*

on Computer Vision and Pattern Recognition, pages 3809–3817, 2019. 3

[27] Tianwei Lin, Zhuoqi Ma, Fu Li, Dongliang He, Xin Li, Errui Ding, Nannan Wang, Jie Li, and Xinbo Gao. Drafting and revision: Laplacian pyramid network for fast high-quality artistic style transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5141–5150, June 2021. 2

[28] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 6

[29] Songhua Liu, Tianwei Lin, Dongliang He, Fu Li, Meiling Wang, Xin Li, Zhengxing Sun, Qian Li, and Errui Ding. Adaattn: Revisit attention mechanism in arbitrary neural style transfer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6649–6658, 2021. 2, 3

[30] Yahui Liu, Enver Sangineto, Yajing Chen, Linchao Bao, Haoxian Zhang, Nicu Sebe, Bruno Lepri, Wei Wang, and Marco De Nadai. Smoothing the disentangled latent style space for unsupervised image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2, 3

[31] Joanna Materzynska, Antonio Torralba, and David Bau. Disentangling visual and written concepts in CLIP. In *CVPR*, 2022. 2, 3, 4, 6, 8, 12

[32] Max Müller-Eberstein, Rob van der Goot, and Barbara Plank. Spectral probing. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7730–7741, Abu Dhabi, United Arab Emirates, Dec. 2022. Association for Computational Linguistics. 2, 8

[33] Dae Young Park and Kwang Hee Lee. Arbitrary style transfer with style-attentional networks. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5880–5888, 2019. 2, 3

[34] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2085–2094, October 2021. 3

[35] Justin NM Pinkney and Chuan Li. clip2latent: Text driven sampling of a pre-trained stylegan using denoising diffusion and clip. *arXiv preprint arXiv:2210.02347*, 2022. 3

[36] 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. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 3

[37] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention - MICCAI*, 2015. 4

[38] Artsiom Sanakoyeu, Dmytro Kotovenko, Sabine Lang, and Björn Ommer. A style-aware content loss for real-time hd style transfer. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 698–714, 10 2018. 2, 3, 11

[39] Peter Schaldenbrand, Zhixuan Liu, and Jean Oh. Styleclipdraw: Coupling content and style in text-to-drawing translation. *arXiv preprint arXiv:2202.12362*, 2022. 3

[40] Alex Tamkin, Dan Jurafsky, and Noah D. Goodman. Language through a prism: A spectral approach for multiscale language representations. In *NeurIPS*, 2020. 2, 3, 4, 5, 8

[41] 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*, pages 6924–6932, 2017. 2, 3

[42] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 2

[43] Zhizhong Wang, Zhanjie Zhang, Lei Zhao, Zhiwen Zuo, Ailin Li, Wei Xing, and Dongming Lu. Aesust: Towards aesthetic-enhanced universal style transfer. In *Proceedings of the 30th ACM International Conference on Multimedia (ACM MM)*, 2022. 2, 3

[44] Zhizhong Wang, Lei Zhao, Haibo Chen, Lihong Qiu, Qihang Mo, Sihuan Lin, Wei Xing, and Dongming Lu. Diversified arbitrary style transfer via deep feature perturbation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7789–7798, 2020. 2, 3

[45] Zijie Wu, Zhen Zhu, Junping Du, and Xiang Bai. Ccpl: Contrastive coherence preserving loss for versatile style transfer. In *European Conference on Computer Vision*, pages 189–206. Springer, 2022. 3

[46] Zipeng Xu, Tianwei Lin, Hao Tang, Fu Li, Dongliang He, Nicu Sebe, Radu Timofte, Luc Van Gool, and Errui Ding. Predict, prevent, and evaluate: Disentangled text-driven image manipulation empowered by pre-trained vision-language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18229–18238, 2022. 3

[47] Zipeng Xu, Enver Sangineto, and Nicu Sebe. Stylerdalle: Language-guided style transfer using a vector-quantized tokenizer of a large-scale generative model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7601–7611, October 2023. 2

[48] Yuan Yao, Jianqiang Ren, Xuansong Xie, Weidong Liu, Yong-Jin Liu, and Jun Wang. Attention-aware multi-stroke style transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1467–1475, 2019. 2, 3

[49] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017. 2, 3, 11