Hindawi
International Journal of Intelligent Systems
Volume 2023, Article ID 9601261, 10 pages
https://doi.org/10.1155/2023/9601261

WILEY | Hindawi

Research Article

Unsupervised Portrait Drawing Generation for Free Styles

Ming Liu,[1] **Jianxin Liao**,[1] **Jingyu Wang**,[1] **Qi Qi**,[1] **Haifeng Sun**,[1] **Zirui Zhuang**,[1] **and Cong Liu**[2]

[1]*National Pilot Software Engineering School, Beijing University of Posts and Telecommunications, Beijing 100876, China*
[2]*China Mobile Research Institute, Beijing, China*

Correspondence should be addressed to Zirui Zhuang; zhuangzirui@bupt.edu.cn

Received 21 March 2023; Revised 3 August 2023; Accepted 7 September 2023; Published 30 September 2023

Academic Editor: Mohammad R. Khosravi

Artistic portrait drawing (APDrawing) generation has seen progress in recent years. However, due to the naturally high scarcity and artistry, it is difficult to collect large-scale labeled and paired data and generally divide drawing styles into several specific recognized categories. Existing works suffer from the limited labeled data and naive manual division of drawing styles according to the corresponding artists. They cannot adapt to the actual situations, for example, a single artist might have multiple drawing styles and APDrawings from different artists might share similar styles. In this paper, we propose to use unlabeled and unpaired data and perform the task in an unsupervised manner. Without manual division of drawing styles, we take each portrait drawing as a unique style and introduce self-supervised feature learning to learn free styles for unlabeled portrait drawings. Besides, we devise a style bank and a decoupled cycle structure to take over two main considerations in the task: generation quality and style control. Extensive experiments show that our model is more adaptable to different style inputs than state-of-the-art methods.

1. Introduction

In recent years, studies on neural style transfer [1–3] have flourished. Researchers are no longer satisfied with a single image expression form and have begun to consider more complex and diverse image translation relations [4–16]. For example, CycleGAN [17] transfers a color photograph to a Monet painting. Neural style transfer [18] considers style transfer as a problem of texture transfer, taking the texture from a style image to a content one. Based on the same texture style modeling assumption, many other methods [19, 20] have been developed for neural style transfer.

However, artistic portrait drawing (APDrawing) generation is completely not a texture-based task in neural style transfer. Artistic portrait drawing (APDrawing) differs in styles from other portrait paintings in terms of strokes, color composition, etc. APDrawing generation is to transform a face photo with the characteristic of a human face reserved and generate a highly abstract artistic portrait drawing. With little texture information, the previous texture-based style transfer methods thereby are not suitable in the APDrawing generation task.

Two recent methods [21, 22] have been specifically proposed for the APDrawing generation task. APDrawingGAN [21] firstly constructed an APDrawing dataset, and developed a hierarchical structure and a distance transform loss. However, it requires paired data and the drawing style is thereby limited to a single one. The method [22] further improved the generation quality and increased the number of generation styles to three. It proposed the asymmetric cycle structure, including a truncation loss and a relaxed forward cycle consistency. However, it naively divided drawing styles according to the corresponding artists, which did not conform to the actual situations. As shown in Figure 1, the first artist uses parallel lines to draw shadows while the second artist often uses continuous thick lines and large dark regions, which is sometimes close to the drawing style of the third artist. The similarities or even the sameness in styles between drawings of different artists and the differences between drawings of the same artist, make it inappropriate to simply divide drawing styles by the artists.

Meanwhile, there exists no publicly available dataset with APDrawings of obviously different styles, which heavily relies on manual crawling and filtering. It is time-consuming and

Figure 1: Examples of portraits and their corresponding artists. It can be noticed that there exist different drawing styles of portraits from the same artist and rarely similar drawing styles among different artists. The observation motivates us to develop a new paradigm for generating portrait drawings with free styles.

costly to collect large-scale labeled and paired training data. In addition, compared with other art paintings, the number of APDrawings is relatively rare. The shortage of training data leads to a great need for a new unsupervised paradigm.

In this paper, we propose two realistic assumptions for the APDrawing generation task, i.e., there is only access to unlabeled and unpaired data and any manually explicit style definition should be discarded. The first assumption enables us to naturally avoid the cost and difficulty of annotating large-scale labeled and paired data. Based on the second assumption, we turn to treating each APDrawing as a single style instead of relying on manual division. Specifically, contrastive self-supervised learning is introduced to learn styles for all input APDrawings, which actually ensures low-cost acquisition of large-scale training data. It pulls close each APDrawing and its augmented images and pushes away different APDrawings. In this way, our style feature extractor can explore latent relations among input data.

Intuitively, without a clear division of drawing styles, the self-supervised style features are accompanied by irrelevant information. It is not easy to embed such unconstrained style features into our generation process, preserving the generation quality and controlling the drawing styles as expected. Accordingly, we build up a style bank for all these styles. As a set of representative styles for style groups, the style bank can also be viewed as a way to reduce the dimension of the style feature space to stabilize the training process. However, we propose a decoupled cycle structure with two streams to guarantee the generation of vivid APDrawings and the generation for free styles.

In summary, the main contributions of our work are listed as follows:

- We propose to use unlabeled and unpaired data for the APDrawing generation task, which frees us from excessive dependence on labeled data.
- Without the naively manual division of drawing styles, we treat each APDrawing as a single style and introduce contrastive self-supervised learning to learn style features for them. It enables us to generate APDrawings with free styles for different style inputs.
- We propose a style bank to update the original style features and a decoupled cycle structure, which guarantees the stability and robustness of training with a set of unsupervised style features.

2. Related Works

2.1. Deep Learning in APDrawing. With the help of deep learning techniques, great strides have been made towards more powerful and adequate artificial intelligence for many vision processing tasks. Drawing related applications, such as line drawing colorization [23] and artistic shadow creation [24], also benefit from deep learning, which can produce more creative and richer paintings with less human efforts. Zhang et al. [25] proposed a deep learning framework for user-guided line art flat filling. It included the split filling mechanism to directly estimate the result colors and influence areas of scribbles. Im2Pencil [26] translated from photos to pencil drawings by a two-branch framework that learned separate filters for outline and shading generation, respectively. It can generate pencil drawings with style control.

2.2. Image-to-Image Translation. Our method also takes the advantage of deep learning technique to image generation. Efforts on image-to-image translation usually fall into two categories: domain-level translation and instance-level translation. It was firstly proposed by [27] to use a non-parametric texture model to learn the translation function between a single training image pair. More recent methods mainly focus on the translation function between two

domains, defined by two sets of datasets. Many of them resorted to conditional generative adversarial network (cGAN) to synthesis images. Pix2Pix [28] was built on cGAN and used paired data between domains to learn the translation function. For many tasks, paired data are not available. To overcome the limitation, cycle consistency was then proposed in CycleGAN [17] and DualGAN [2], which make use of the cycle consistency constraint. This constraint enforces that the two mappings from domains A to B and from B to A, when applied consecutively to an image, revert the image back to itself. It regularized the training by reconstructing an original image from its translated image. StarGAN [29, 30] and ComboGAN [31] were then proposed to extend the image-to-image translation between two domains to multiple domains based on the cycle consistency. Another line of methods [1, 32] was not restricted in the image level but in the feature level. They assumed a shared latent space but MUNIT [32] postulated that only part of the latent space should be shared rather than a full latent space proposed in UNIT [1].

However, these methods either cannot generate images of different styles, or are not suitable for our APDrawing generation task because of the lack of ability to describe facial features in detail.

2.3. Neural Style Transfer. Neural style transfer is closely related to image-to-image translation, which aims at preserving the content of an image but transferring from the style of another image. Classic neural style transfer usually refers to the example-guided style transfer, while image-to-image translation mainly refers to the domain-based image translation. Neural style transfer was firstly proposed in image style transfer [18] to introduce a CNN to reproduce famous painting styles on natural images. It penalised differences between high-level CNN features of the generated image and the content image, and used the Gram matrix statics of features in the CNN to measure the style similarity.

Many follow-up studies [4, 33–38] were conducted to either improve or extend the method. These methods [20, 39] addressed the issue of slow optimization process by training feed-forward neural networks. Different from these methods, the method of [19] replaced the Gram-based modeling way of styles but used a Markov random field (MRF) regularizer. Combing deep convolutional neural networks (DCNNs) with MRF models-based texture synthesis can be applied to both photographic and nonphotorealistic synthesis tasks.

However, most methods model style as texture, which is not suitable for our task. With little textual information, APDrawing generation requires a high degree of abstraction and completeness of some facial details in strokes at the same time.

3. Method

3.1. Overview. Our method is proposed for the APDrawing generation task, transferring the style of an APDrawing in domain $\mathscr{A}$ to the input photo in domain $\mathscr{P}$. The APDrawings and the photos are denoted as $\{a_i\}_{i=1}^{i=N}$, where $a_i \in \mathscr{A}$, and $\{p_i\}_{i=1}^{i=M}$, where $p_i \in \mathscr{P}$, respectively. N and M are the number of APDrawings and photos in our training set.

As illustrated in Figure 2, the training process can be divided into two phases, i.e., extracting style features for unlabeled APDrawings and generating APDrawings with the desired styles. The first training phase uses unlabeled and unpaired APDrawings and introduces contrastive self-supervised learning to learn styles for them. The style bank B is built up and the style features are updated to the similarities with the style bank. The second stage uses a set of generators and discriminators. The generator G and an inverse generator F are included to generate vivid APDrawings from input photos and style features, and transform APDrawings back to input photos without edge information loss, respectively. The discriminators consist of $D_{\mathscr{A}}$ and $D_{\mathscr{P}}$ to guarantee the discrimination between the generated fake images and the real images in both domain $\mathscr{A}$ and $\mathscr{P}$.

Next, we will introduce details of our proposed method from the following aspects: (1) unsupervised style feature extraction and (2) unsupervised portrait generation.

3.2. Unsupervised Style Feature Extraction. Previous methods [21, 22] for the APDrawing generation task are either limited to one single style drawn by an artist, or a predefined division of drawing styles based on corresponding artists. In fact, it is hard to divide the drawing styles of APDrawings into several specific categories. Meanwhile, there is a lack of public APDrawing datasets with several different drawing styles, and it is quite costly to collect such a large-scale labeled one. A new benchmark is required for the task, due to the high artistry in drawing styles and the scarcity in labeled data. The new benchmark should be able to use unlabeled data and adapt to various drawing styles.

We introduce the contrastive self-supervised learning to train our style extractor for unlabeled and unpaired APDrawings. It is capable of adopting self-defined pseudo-labels as supervision and utilizing the learned style features in the next APDrawing generation phase. As a discriminative approach, contrastive self-supervised learning aims at grouping similar samples closer and separating diverse samples far from each other as shown in Figure 2. Specifically, the VGG19 network is adopted as our feature extractor, denoted as W. We pull augmented versions of the same sample close to each other while pushing away style features from different samples. The formulation of the loss function is written as follows:

$$L_1 = -\sum_{i}^{i=N} \log \frac{ES\left(a_i, a_i^+\right)}{ES\left(a_i, a_i^+\right) + \sum_{j=0}^{n^-} ES\left(a_i, a_j^-\right)},$$
$$ES\left(a_i, a_j\right) = \exp\left(\frac{\text{sim}\left(W\left(a_i\right), W\left(a_j\right)\right)}{\tau}\right), \tag{1}$$

where τ is a temperature coefficient, n^- represents the number of negative samples of a_i in a minibatch, and sim $(\cdot)$ measures the cosine similarity between two input vectors. In order to keep the style-invariance of these APDrawings, we choose some data augmentation methods, including cropping, resizing, horizon-flip, and rotation. In a minibatch, the original APDrawing, the transformed version of the original

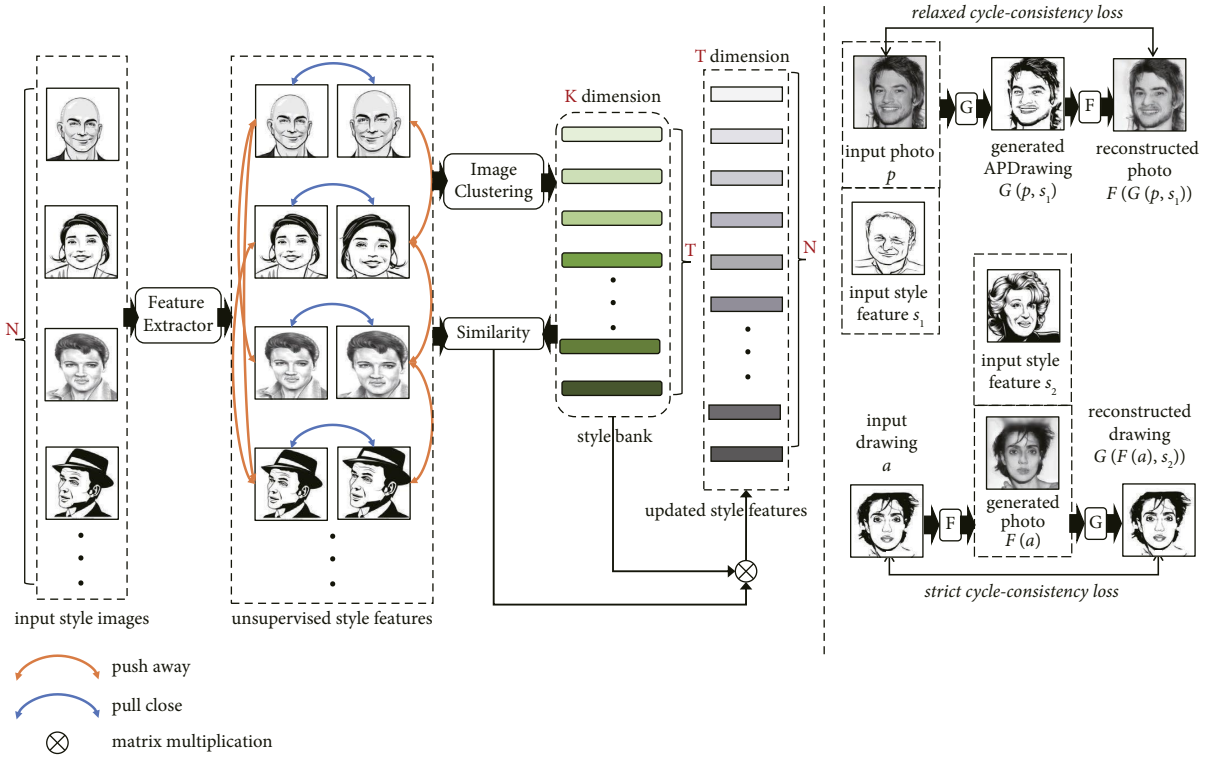


FIGURE 2: The pipeline of our proposed model for the APDrawing generation task. It consists of two training phases, i.e., (a) a style feature extraction phase and (b) a generation phase. The style feature extraction aims at learning the style feature for any given portrait drawing. The generation phase generates portrait drawings with specific styles of given style images from original input photos.

APDrawing, and the aligned version of the APDrawing and its transformed version are positive samples to each other. In this self-supervised way, our feature extractor explores the underlying data structure in these APDrawings.

Considering the high dimensionality of the feature space and the scarcity of data, the model might not be able to learn a stable and robust mapping from APDrawings to style features. We turn to building up a style bank $B = [b_1, \ldots, u_T]$, where T is the bank size. It is obtained by clustering style features of these unlabeled APDrawings using the K-means algorithm. On the one hand, it can reduce the dimension of the style feature space and stabilize the training process to obtain robust style features for APDrawings. On the other hand, the style bank can be viewed as a set of representative styles for style groups, which might benefit in alleviating the negative impact of irrelevant information of drawing styles brought by contrastive self-supervised learning. Finally, the updated style feature of a_i is computed as the cosine similarities between the style bank and the original style feature, which is written as follows:

$$s_i = W(a_i) \cdot B, \tag{2}$$

where $W(a_i)$ is a K-dimensional vector and $B \in \mathbb{R}^{K \times T}$ represents the style bank. All updated style features make up the set $\mathcal{S}$, denoted as $\mathcal{S} = \{s_i\}_{i=1}^{i=N}$.

3.3. Unsupervised Portrait Generation. With the updated style features as input, the portrait generation phase aims at transferring styles, defined by an input style image, to a face photo. During the generation process, the following two considerations need to be guaranteed: generation quality and style control. Generation quality ensures to generate a vivid portrait, preserving the facial features and less discrimination from the real ones. Style control enables to keep the drawing style unchanged, compared with the style input. The total loss function can be summarized in the following form:

$$L_2 = L_{\text{quality}} + L_{\text{style}}. \tag{3}$$

3.3.1. Generation of Vivid Portraits. There are two generators, G and F, using the architecture of autoencoder with residual blocks. The discriminator set $D_{\mathcal{A}}$ is based on PatchGAN [28]. It involves a global discriminator, without information loss in the holistic characteristics, and a set of local discriminators for the fine details in facial regions. G and $D_{\mathcal{A}}$ are used in the generation from input photos to APDrawings. F and $D_{\mathcal{P}}$ are optimized in the opposite direction. They are trained with the adversarial loss, the asymmetric cycle consistency loss, and truncation loss [22], which are formulated as follows:

$$L_{\text{quality}} = L_{\text{adv}}(G, D_{\mathscr{A}}) + L_{\text{adv}}(F, D_{\mathscr{P}}) + \lambda_1 L_{\text{relaxed-cycle}}(G, F) + \lambda_2 L_{\text{strict-cycle}}(G, F) + \lambda_3 L_{\text{trunc}}(G, F). \quad (4)$$

In other style transfer tasks, color information is quite important for the style modeling, while it is irrelevant for our task. We propose to transform an input image to a gray-scale one before sending it to the generation network. It enforces our network focusing on the line strokes and shadow usage, bringing balance for training F and $D_{\mathscr{P}}$ in the second cycle. Besides, we decouple two cycles by sending different pairs of input, including APDrawings, photos, and style features. The richness of the pair combination is more conducive to our generation.

3.3.2. Generation of Portraits for Specific Styles. The style classification network, denoted as D_s, shares the first few blocks with the global discriminator of $D_{\mathscr{A}}$. In order to achieve the purpose of style control, the style loss is formulated as follows:

$$L_{\text{style}} = \mathbb{E}_{a\in\mathscr{A}}\left[\left\|D_s(a) - \dot{s}\right\|_2\right] + \mathbb{E}_{p\in\mathscr{P}}\left[\left\|D_s(G(p, \ddot{s})) - \ddot{s}\right\|_2\right]. \quad (5)$$

For real APDrawing a, D_s outputs the predicted style feature to get close to which computed in the first style feature extraction phase, denoted as $\dot{s}$. For the generated APDrawing $G(p, \ddot{s})$, its output style feature is specified by the input style feature $\ddot{s}$. The style loss guides D_s to produce the style features close to the real feature distribution $\mathscr{S}$ and generate APDrawings close to the desired styles.

4. Experiments

4.1. Datasets. Although the training set of APDrawings in the method [22] have not been released, we have collected similar number of APDrawings to train our method. Due to the lack of public datasets with multiple styles, the APDrawings are crawled from the Internet to construct an APDrawing dataset with the size of 641, named **APDrawingCrawl**. It consists of 116 APDrawings of the artist Charles Burns, 45 APDrawings of the artist Yann Legendre, 89 APDrawings of the artist Kathryn Rathke, 233 APDrawings from vectorportral.com, and other 158 APDrawings without tagged/labeled artist/source information. 641 APDrawings and 1000 face images from CELEBA-HQ [40] form our training set. The testing set consists of 200 face photos, mainly from CELEBA-HQ.

All the training images, including APDrawings and face photos, are resized at the resolution of 512×512. For training local discriminators in $D_{\mathscr{A}}$, training images are aligned using facial landmarks and perform face-parsing from the model BiSeNet [41].

4.2. Implementation Details

4.2.1. Extraction of Style Features. The CNNs for image feature extraction consists of three Conv-BatchNorm-ReLU blocks with two 1/2-scale downsampling operations. For the transformer, the feature dimension is 256 and both encoder and decoder have 3 layers. We use the ReLU output of the 13th convolutional layer of VGG19 as the style feature. We set the initial learning rate to 0.001 for the first 10 epochs and decay it by 10 times in the next 10 and 25 epochs. The total training epoches are 30. The batch size is fixed to 16 and the bank size is 10.

4.2.2. Generation of APDrawings. We set the hyperparameter in equation (5) as $\lambda_1 = 5 - 4.5i/n$, $\lambda_2 = 5$, and $\lambda_3 = 4.5i/n$, where i and n are the current epoch and the total epochs, respectively. The learning rate is set to $1.5e-5$ for the first 100 epochs and is linearly decayed to 0 in the next 100 epochs.

4.3. User Study. To evaluate the effectiveness of our method, we conduct a user study to compare with CycleGAN [17] and the method [22]. We randomly sample 30 face photos from the test set of **APDrawingCrawl** and transform them to three different styles. There are 50 participants involved in the user study and each of them is provided with these 30 images, resulting in 1,500 votes. With a style example image, every 10 images are transformed to that style. There are 3 style images in total. All participants are given an input photo, a style example image and drawings generated by these three methods at the same time. The voting criterion is based on image integrity, image quality, style perseverance, and face characteristic similarity.

As shown in Table 1, we have similar performance with the state-of-the-art method [22], which is in line with the expectations. Our method aims at solving the problem of APDrawing generation with unlabeled and unpaired training data, preserving the input style and the content from the face photo. The method [21] requires paired data and the drawing style is thereby limited to a single one. The method [22] further increases the number of generation styles to three with more collected data with labels. However, they suffer from the limited labeled data and naive manual division of drawing styles according to the corresponding artists, which do not conform to the actual situations.

Besides, it is truly hard for unsupervised methods to obtain apparently better generation quality than supervised methods, without the restricted classification categories and abundant labeled data for each class. So we emphasize the superior of our method on higher flexibility and scalability for different style inputs, and less dependence on abundant labeled data.

4.4. Qualitative Results. We conduct qualitative model analysis on both style feature extraction phase and APDrawing generation phase to demonstrate our effectiveness.

4.4.1. Style Feature Extraction. We use K-means to divide the style images into five clusters and visualize the learned style features by t-SNE. As shown in Figure 3, our style

Table 1: User study results comparing our method with CycleGAN [17] and Yi et al. [22].

Methods	Top1 (%)	Top2 (%)	Top3 (%)
CycleGAN [17]	10.2	18.4	71.4
Yi2020 [22]	44.2	41.0	14.8
Ours	45.6	40.6	13.8

The i-th column represents the percentage of different methods that rank i-th among the three methods.

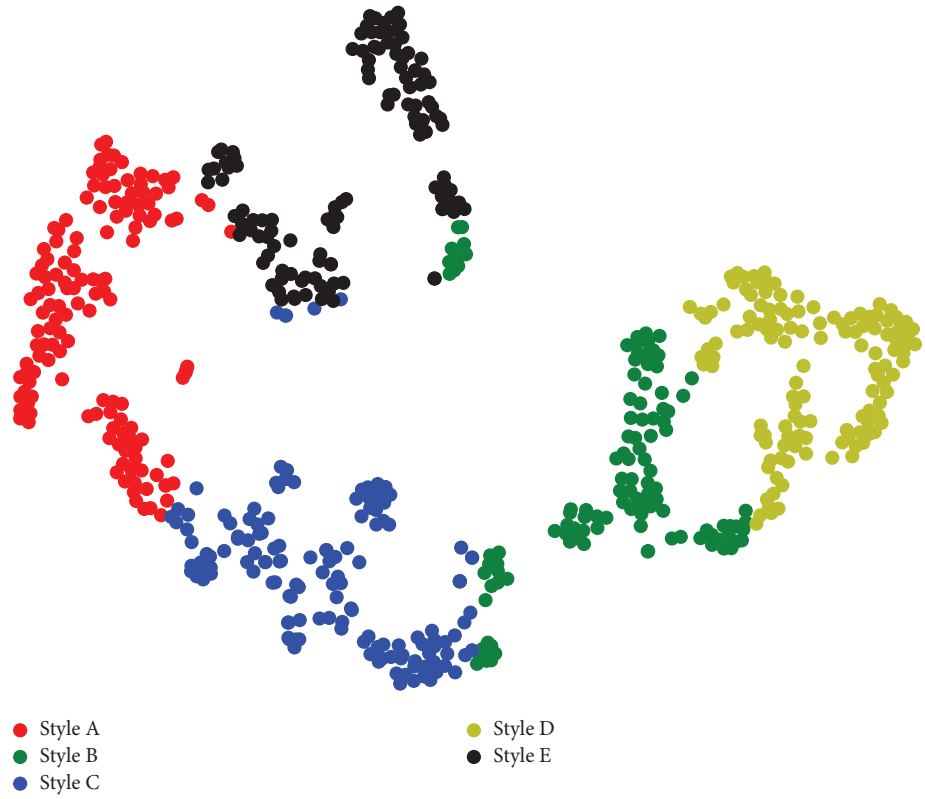


Figure 3: T-SNE visualization on collected dataset APDrawingCrawl in the style extraction phase. We split all style images into five clusters as representative styles.

extraction is able to learn well-separated features. However, there is still the problem of unclear boundaries between different clusters, which can be expected and explained. From the very beginning, we do not want to manually divide the styles and therefore the style bank is introduced. The style clustering is for generating the style bank and some unseparated samples demonstrate the inapplicability of simple divisions. Due to the high dimensionality of the feature space and the scarcity of data, it is hard for the model to learn robust style features. With the dimensionality constraints brought by the style bank, the learning process could be stabilized.

As shown in Figure 4, we show the portrait drawings nearest to the cluster centers (style bank) of all APDrawing styles. These drawings can be viewed as the prototype style images of each style in the style bank. From the line strokes and shadow usage of these APDrawings, our method has the ability to distinguish among various drawing styles and they are actually from diverse artists, i.e., Yann Legendre, vectorportral.com, Kathryn Rathke, Charles Burns, and an unknown artist.

4.4.2. APDrawing Generation. We compare our method with a method Yi et al. [22] designed for the same APDrawing task, two examples-guided neural style transfer methods, image style transfer [18] and linear style transfer [5], and a unpaired image-to-image translation method ComboGAN [31].

As shown in Figure 5, we make a comparison with Yi et al. [22] to demonstrate the limitation of relying on the manual division of styles based on artists. We list three input styles and the corresponding generation results. The first two style input images are from the same artist, Kathryn Rathke. According to the style division strategy defined in Yi et al. [22], the output transferred images of input style1 and style2 should be the same, while ones of style2 and style3 should be obviously different. However, in fact, we can easily distinguish different drawing techniques between the input style1 and the input style2. The style1 input image is distinguished by a large area of gray thick shadow on the face, while the style2 input image is not. The input images of style2 and style3 share the similar usage of dark regions. The manual style division according to the artist is apparently not

Figure 4: Visualization of portrait drawings nearest to the cluster centers (style bank) of all portrait drawings in the style extraction. These drawings can be viewed as the prototype style images of each style in the style bank. They are obviously different in drawing styles and actually from diverse artists, i.e., Yann Legendre, vectorportral.com, Kathryn Rathke, Charles Burns, and an unknown artist.

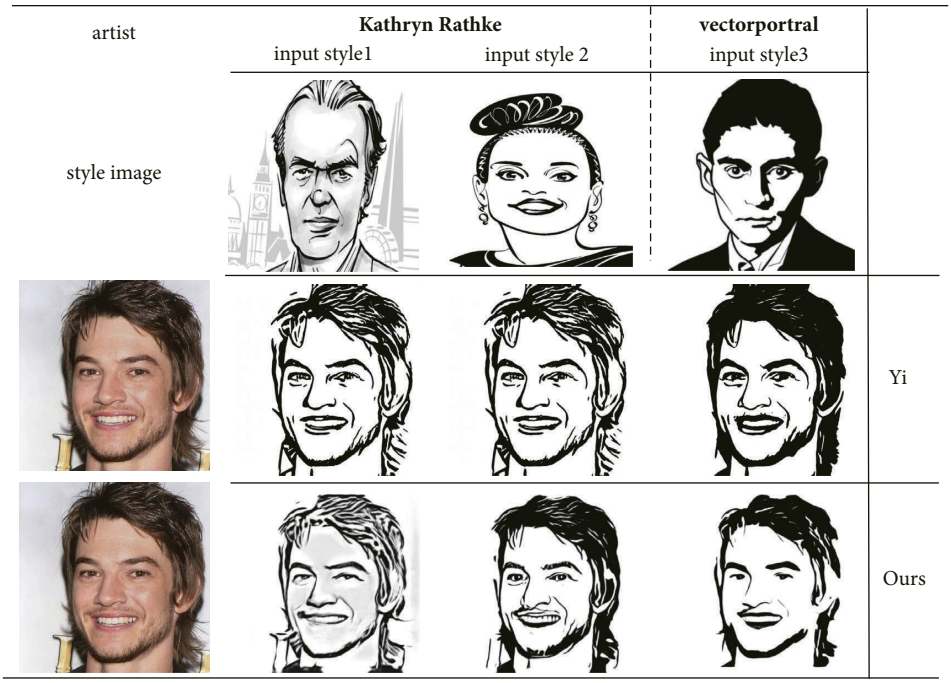


Figure 5: Comparison results with Yi et al. [22] and ours. The input style1 and input style2 are actually from the same artist Kathryn Rathke, while the input style3 is from another artist. Although input images of style1 and style2 are from the same artist, they apparently differ in drawing styles. The method [22] simply treats style1 and style2 as the same style by predefined manual division of styles, while ours use a more adaptive paradigm for free style APDrawing generation.

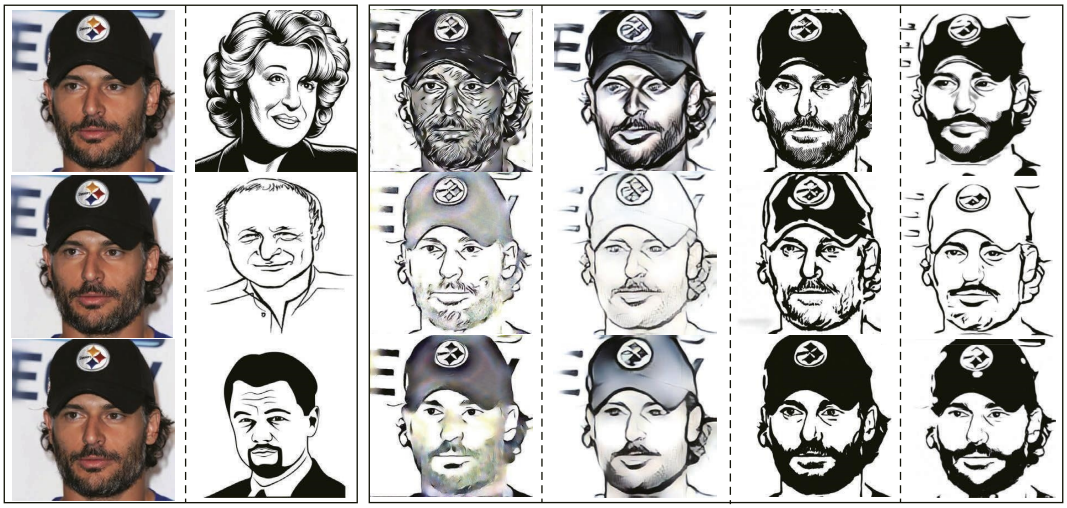
suitable in the situation, resulting in confusion in the generated results of Yi. Compared to the method of Yi et al. [22], our method is more flexible to generate APDrawings with desired drawing styles.

As shown in Figure 6, we make a comparison with two example-guided methods and a state-of-the-art one, Yi et al. [22]. It can be easily seen that these two neural transfer methods either fail to capture the differences in input styles, or cannot generate images with acceptable generation quality. Yi et al. [22] can generate drawings with three distinct styles. However, our method can actually generate drawings according to different input styles, with higher flexibility. Our first generation result uses parallel lines to draw shadows, the second result tends to use clean lines, and the third one uses large area of dark regions.

As shown in Figure 7, we make a comparison with an image-to-image translation method and a state-of-the-art one. ComboGAN fails to generate vivid APDrawings with good generation quality. For example, there exist some strips and undesired gray color on the face. Yi et al. [22] can generate discriminative results for three fixed styles. Despite the lack of label information, our model can still capture differences between different styles. The generated images are consistent in a single style while they differ from images in other styles.

4.5. Ablation Study. We conduct ablation studies as follows: (1) generation without using the style bank and (2) generation without gray-scale inputs, i.e., using the original color inputs.

As shown in Figure 8(c), only the center areas have visible traces of the generated drawings, and the hair disappears. Figure 8(e) has finer details in the cap and the hair than Figure 8(c). Without using the style bank, our method falls into the situation that the negative influence is brought in by the accompanied redundant information of style features. It might lead to the leaving of a blank outside the fixed area or other undesired situations in our generated drawings. The introduction of style bank is to eliminate the



(a) Input Image (b) Input Style (c) Gatys (d) LinearStyleTransfer (e) Yi (f) Ours

FIGURE 6: Comparisons with two state-of-the-art example-guided neural style transfer methods, i.e., image style transfer [18] and linear style transfer [5] and an APDrawing generation method [22] with multiple styles.

(a) Input (B) ComboGAN (style1, style2) (a) Yi (style1, style2, style3) (a) Ours (style1, style2, style3)

FIGURE 7: Comparisons with a general unpaired image-to-image translation method ComboGAN [31] and a method [22] specially designed for the APDrawing generation task. They can generate images with different styles or translate image into several domains.

(a) (b) (c) (d) (e)

FIGURE 8: Ablation study. (a) Input aligned face photos, (b) input styles, (c) results without building the style bank, (d) results without gray-scale inputs, and (e) our results.

interference of factors irrelevant to style and keep the network focusing on the key factors related to styles, such as line strokes. The model with the style bank guarantees the details and completeness of the drawings.

As shown in Figure 8(d), the color changes on the face lead to the extra or erroneous lines. With the color input, we can see that the results might be more sensitive to the areas with color changes, which is not desired in the APDrawing task. The introduction of using gray-scale inputs is to avoid being affected by the task-irrelevant inputs.

5. Conclusion

In this paper, we propose to perform the APDrawing generation task in an unsupervised manner, which can avoid the difficulties of collecting large-scale labeled data and the irrationality of dividing drawing styles into some specific categories. We introduce contrastive self-supervised learning to learn free styles of APDrawings by treating each as a single style. The style bank and corresponding decoupled cycle structure guarantee the generation quality and style control of the output APDrawings. Experiments show the flexibility and scalability of our method in generation of APDrawings with different styles, which is more adaptive compared to other state-of-the-art methods. In our future study, we will investigate how to improve the realism and faithfulness of the low-quality original photos, such as these with blurry texture, which would cause noisy and messy lines, failing to preserve fine details.

Data Availability

Previously reported (CELEBA-HQ) data were used to support this study and are available at https://doi.org/10.1007/978-3-030-01261-8_20. These prior studies (and datasets) are cited at relevant places within the text as references [41].

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grants 62171057, 62101064, 62201072, 62071067, and 62001054, in part by the Ministry of Education and China Mobile Joint Fund (MCM20200202), and in part by the Beijing University of Posts and Telecommunications-China Mobile Research Institute Joint Innovation Center.

References

[1] M. Y. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks," *Advances in Neural Information Processing Systems*, pp. 700–708, 2017, https://arxiv.org/abs/1703.00848.

[2] Z. Yi, H. Zhang, P. Tan, and M. Gong, "Dualgan: unsupervised dual learning for image-to-image translation," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2849–2857, Venice, Italy, October 2017.

[3] H. Wu, Z. Sun, Y. Zhang, and Q. Li, "Direction-aware neural style transfer with texture enhancement," *Neurocomputing*, vol. 370, pp. 39–55, 2019.

[4] A. J. Champandard, "Semantic style transfer and turning two-bit doodles into fine artworks," 2016, https://arxiv.org/abs/1603.01768.

[5] X. Li, S. Liu, J. Kautz, and M. H. Yang, "Learning linear transformations for fast image and video style transfer," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3809–3817, Long Beach, CA, USA, June 2019.

[6] W. Zeng, M. Zhao, Y. Gao, and Z. Zhang, "TileGAN: category-oriented attention-based high-quality tiled clothes generation from dressed person," *Neural Computing & Applications*, vol. 32, no. 23, pp. 17587–17600, 2020.

[7] S. Xu, Q. Zhu, and J. Wang, "Generative image completion with image-to-image translation," *Neural Computing & Applications*, vol. 32, no. 11, pp. 7333–7345, 2020.

[8] S. Xu, Q. Zhu, and J. Wang, "Correction to: generative image completion with image-to-image translation," *Neural Computing & Applications*, vol. 32, no. 23, Article ID 17809, 2020.

[9] S. Wu, W. Liu, Q. Wang, S. Zhang, Z. Hong, and S. Xu, "Reffacenet: reference-based face image generation from line art drawings," *Neurocomputing*, vol. 488, pp. 154–167, 2022.

[10] F. Yang, Y. Wang, L. Herranz, Y. Cheng, and M. G. Mozerov, "A novel framework for image-to-image translation and image compression," *Neurocomputing*, vol. 508, pp. 58–70, 2022.

[11] H. Zhang, Y. Sun, L. Liu, and X. Xu, "CascadeGAN: a category-supervised cascading generative adversarial network for clothes translation from the human body to tiled images," *Neurocomputing*, vol. 382, pp. 148–161, 2020.

[12] H. Dou, C. Chen, X. Hu, L. Jia, and S. Peng, "Asymmetric CycleGAN for image-to-image translations with uneven complexities," *Neurocomputing*, vol. 415, pp. 114–122, 2020.

[13] X. Nie, H. Ding, M. Qi, Y. Wang, and E. K. Wong, "Urca-gan: upsample residual channel-wise attention generative adversarial network for image-to-image translation," *Neurocomputing*, vol. 443, pp. 75–84, 2021.

[14] D. Wu, J. Gan, J. Zhou, J. Wang, and W. Gao, "Fine-grained semantic ethnic costume high-resolution image colorization with conditional GAN," *International Journal of Intelligent Systems*, vol. 37, no. 5, pp. 2952–2968, 2022.

[15] W. Zheng, L. Yan, C. Gou, and F. Wang, "Fighting fire with fire: a spatial-frequency ensemble relation network with generative adversarial learning for adversarial image classification," *International Journal of Intelligent Systems*, vol. 36, no. 5, pp. 2081–2121, 2021.

[16] Z. Chen, T. Zhu, P. Xiong, C. Wang, and W. Ren, "Privacy preservation for image data: a GAN-based method," *International Journal of Intelligent Systems*, vol. 36, no. 4, pp. 1668–1685, 2021.

[17] J. Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, pp. 2223–2232, Venice, Italy, October 2017.

[18] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2414–2423, Las Vegas, NV, USA, June 2016.

[19] C. Li and M. Wand, "Combining Markov random fields and convolutional neural networks for image synthesis," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2479–2486, Las Vegas, NV, USA, June 2016.

[20] D. Ulyanov, V. Lebedev, A. Vedaldi, and V. S. Lempitsky, "Texture networks: feed-forward synthesis of textures and stylized images," p. 4, New York City, NY, USA, June 2016.

[21] R. Yi, Y. J. Liu, Y. K. Lai, and P. L. Rosin, "ApdrawingGAN: generating artistic portrait drawings from face photos with hierarchical GANs," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 10743–10752, Long Beach, CA, USA, January 2019.

[22] R. Yi, Y. J. Liu, Y. K. Lai, and P. L. Rosin, "Unpaired portrait drawing generation via asymmetric cycle mapping," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8217–8225, Seattle, WA, USA, June 2020.

[23] M. Yuan and E. Simo-Serra, "Line art colorization with concatenated spatial attention," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3946–3950, Nashville, TN, USA, June 2021.

[24] L. Zhang, J. Jiang, Y. Ji, and C. Liu, "Smartshadow: artistic shadow drawing tool for line drawings," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5391–5400, Montreal, Canada, October 2021.

[25] L. Zhang, C. Li, E. Simo-Serra, Y. Ji, T. T. Wong, and C. Liu, "User-guided line art flat filling with split filling mechanism," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9889–9898, Nashville, TN, USA, June 2021.

[26] Y. Li, C. Fang, A. Hertzmann, E. Shechtman, and M. H. Yang, "Im2pencil: controllable pencil illustration from photographs," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1525–1534, Long Beach, CA, USA, June 2019.

[27] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin, "Image analogies," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pp. 327–340, New York, NY, USA, August 2001.

[28] P. Isola, J. Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1125–1134, Honolulu, HI, USA, July 2017.

[29] Y. Choi, M. Choi, M. Kim, J. W. Ha, S. Kim, and J. Choo, "Stargan: unified generative adversarial networks for multi-domain image-to-image translation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8789–8797, Salt Lake City, UT, USA, June 2018.

[30] Y. Choi, Y. Uh, J. Yoo, and J. W. Ha, "Stargan v2: diverse image synthesis for multiple domains," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8188–8197, Seattle, WA, USA, June 2020.

[31] A. Anoosheh, E. Agustsson, R. Timofte, and L. Van Gool, "Combogan: unrestrained scalability for image domain translation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 783–790, Salt Lake City, UT, USA, December 2018.

[32] X. Huang, M. Y. Liu, S. Belongie, and J. Kautz, "Multimodal unsupervised image-to-image translation," in *Proceedings of the European Conference on Computer Vision*, pp. 172–189, Munich, Germany, September 2018.

[33] A. Selim, M. Elgharib, and L. Doyle, "Painting style transfer for head portraits using convolutional neural networks," *ACM Transactions on Graphics*, vol. 35, no. 4, pp. 1–18, 2016.

[34] M. Lu, H. Zhao, A. Yao, F. Xu, Y. Chen, and L. Zhang, "Decoder network over lightweight reconstructed feature for fast semantic style transfer," in *Proceedings of the IEEE international conference on computer vision*, pp. 2469–2477, Venice, Italy, October 2017.

[35] Mallika, J. S. Ubhi, and A. K. Aggarwal, *Journal of Visual Communication and Image Representation*, J Vis Commun Image Represent, 2022.

[36] Y. Liu, A. Jiang, J. Pan, J. Liu, and J. Ye, "Deliberation on object-aware video style transfer network with long–short temporal and depth-consistent constraints," *Neural Computing & Applications*, vol. 33, no. 14, pp. 8845–8856, 2021.

[37] X. Chen, S. Zhang, G. Shen, Z. H. Deng, and U. Yun, "Towards unsupervised text multi-style transfer with parameter-sharing scheme," *Neurocomputing*, vol. 426, pp. 227–234, 2021.

[38] Z. Ma, J. Li, N. Wang, and X. Gao, "Semantic-related image style transfer with dual-consistency loss," *Neurocomputing*, vol. 406, pp. 135–149, 2020.

[39] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," 2016, https://arxiv.org/abs/1603.08155.

[40] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," 2017, https://arxiv.org/abs/1710.10196.

[41] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Bisenet: bilateral segmentation network for real-time semantic segmentation," in *Proceedings of the European conference on computer vision*, pp. 325–341, Munich, Germany, September 2018.