PAPER *Special Section on Information and Communication Technology to Support Hyperconnectivity*

Multi-Style Shape Matching GAN for Text Images

Honghui YUAN[†a)], ***Nonmember and*** **Keiji YANAI**[†b)], ***Member***

SUMMARY Deep learning techniques are used to transform the style of images and produce diverse images. In the text style transformation field, many previous studies attempted to generate stylized text using deep learning networks. However, to achieve multiple style transformations for text images, the methods proposed in previous studies require learning multiple networks or cannot be guided by style images. Thus, in this study we focused on multistyle transformation of text images using style images to guide the generation of results. We propose a multiple-style transformation network for text style transfer, which we refer to as the Multi-Style Shape Matching GAN (Multi-Style SMGAN). The proposed method generates multiple styles of text images using a single model by training the model only once, and allows users to control the text style according to style images. The proposed method implements conditions to the network such that all styles can be distinguished effectively in the network, and the generation of each styled text can be controlled according to these conditions. The proposed network is optimized such that the conditional information can be transmitted effectively throughout the network. The proposed method was evaluated experimentally on a large number of text images, and the results show that the trained model can generate multiple-style text in realtime according to the style image. In addition, the results of a user survey study indicate that the proposed method produces higher quality results compared to existing methods.

key words: *font translation, style transfer, text images, multistyle, GAN*

1. Introduction

Artistic text is frequently used in a variety of fields and applications, especially advertising. However, manually designing realistic artistic letters is frequently time-consuming, and different text elements, e.g., Chinese or Japanese characters, often require unique designs. Neural style transfer techniques have shown impressive results in deep learning. For example, for text style transfer, recent studies have completed the transition from the text font domain to the text style domain. In addition, using style images or text descriptions to achieve text image style transformation has achieved significant results. Although using text descriptions to stylize text images is extremely popular, utilizing style images for the text style transformation task is meaningful and necessary due to the limitations associated with vague text descriptions and specific user requirements. However, when attempting text style transfer with style images, multiple style transfer often requires learning many models. Training numerous models is time-consuming and inconvenient. Although the use of one model could realize arbitrary style transformation in ordinary images, this has not been achieved for text images due to the special structure characteristics of the font. Therefore, designing a multi-style network for text images is necessary and can be used as a basis model for allowing arbitrary style transformation of text images. Even though many studies have investigated text style transfer, generating multiple styles of text images in a single model using style images remains an unsolved problem. Furthermore, when using style images to guide text style transformation, the network must learn the content and texture features of a large number of style images and it must match those style features to text features, which increases the complexity of the task significantly. Therefore, when using style images as desired styles, text style transformation has not been able to achieve multiple styles or arbitrary style transformation in a single model.

Therefore, to solve this problem, we propose the Multi-Style SMGAN transfer network for text images to investigate the possibility of achieving multiple style transformations using a single model with style images. A recent study demonstrated that Shape MatchingGAN [1] can change the style of text using a single style image, which makes it possible to transform text images into multiple styles. Thus, in the current study, we utilized Shape MatchingGAN [1] as the base network and optimized it appropriately to realize multiple style transformations. Here, the main concept is to control the generation of each styled text image by adding conditions to the network and modifying the network to ensure that the features of each style image can be learned efficiently. Specifically, we employ a pair of mask images and style images as the input to the network. The mask images are used as a condition to control the network to learn various style features. In addition, to allow mask images to control the style generation more effectively, we supplemented the network with SPatially-Adaptive DE-normalization (SPADE) residual blocks (ResBlk) [2], which allows the network to retain information about the mask images more effectively. The proposed method was evaluated experimentally, and the results demonstrate that our method is superior in terms of generating multiple-style text images compared to existing image style transformation methods, and the generation of styled text images achieves the expected results. In addition, we utilized a large amount of text to verify the effectiveness of our multistyle network.

In addition, although the proposed network can achieve

Manuscript received June 1, 2023.
Manuscript revised November 15, 2023.
Manuscript publicized December 27, 2023.

[†] The authors are with the Department of Informatics, The University of Electro-Communications, Chofu-shi, 182–8585 Japan.
a) E-mail: yuan-h@mm.inf.uec.ac.jp
b) E-mail: yanai@cs.uec.ac.jp

DOI: 10.1587/transele.2023IHP0010

multistyle transformation successfully, the number of styles that can be transformed is limited by the shape characteristics of the style. Compared to SPADE [2], the semantic region-adaptive normalization (SEAN) [3] network can better capture the global information of the image and generate more natural images. Thus, we further optimized the proposed method to determine whether using the SEAN [3] network rather than SPADE [2] ResBlks would improve the quality of the generated images or if arbitrary style transformation can be achieved. We applied the SEAN network to the proposed Multi-Style SMGAN and optimized the network appropriately. The experimental results indicate that relative to achieving multiple style transformations, more styles can be transformed in a single network using the SEAN network, compared to the method that uses the SPADE ResBlk technique. In addition, we found that the quality of the results is approximately the same as the base Shape MatchingGAN method (the results of the base method were obtained by training multiple models). Note that this paper is based on our previous conference paper [4].

Our primary contributions are summarized as follows:

1. We propose the Multi-Style SMGAN transfer network for text. The proposed method can learn multiple styles in a single model according to style images.
2. We control the generation of various styles in a simple manner. Specifically, we utilize mask images of the corresponding style to control the generation of text images.
3. Experimental results prove that the proposed network can generate effective multistyle images and is superior in terms of image quality.

2. Related Work

2.1 Image-to-Image Translation

The image-to-image translation task involves transferring image content from one domain to another domain. In the following, we introduce several representative examples of image-to-image translation methods. Pix2Pix [5] realizes image generation from simple sketches or masks using paired trained data and CycleGAN [6] performs transformation between two domains with unpaired data based on the generative adversarial network (GAN). The BicycleGAN [7] method implements improvements to Pix2pix [5] by adding a variational autoencoder (VAE) and enabling multiple style changes. The UNIT [8] method assumes that different data spaces share a potential space, which enables unsupervised transformations between different domains. MUNIT [9], which realizes unsupervised multistyle transfer, is an extension of UNIT [8]. In MUNIT, the content space of the image is assumed to be shared and the style space is assumed to be independent. In addition, StarGAN [10] is a multiple-domain version of CycleGAN [6], that enables multidomain conversion using a single generator and a single discriminator. To achieve the transformation to multiple domains, StarGAN [10] adds control information about the domain selection, similar to the conditional GAN [11] format. In the network structure, the discriminator needs to learn to identify whether the sample is real and determine which domain the real image comes from. These existing methods have demonstrated good results in image-to-image translation based on the GAN network [12]. Similar to these studies, the goal of this study was to develop a GAN-based method to achieve the translation of text images from the text domain to the style domain.

The SPADE [2] method allows users to create a composite image from a simple semantic image drawn by the user. The user can also select the style of the image to be synthesized, which makes it possible to obtain a wide variety of synthetic results. SEAN [3] improved SPADE [2] by implementing a new normalization module. With this method, it is possible to create spatially distinct normalization parameters for each semantic region using style input images; thus, individual control of each region of a semantic segmentation image was realized. These two methods realize the translation from semantic images to natural images, and this concept was also utilized in the current study to control the generated images effectively.

2.2 Style Transfer

Similar to the image-to-image translation task, image style transformation involves changing the color and other features of an image while retaining the original content of the image unchanged. In neural image style transfer [13], a pretrained CNN is used to separate the content and style of the image, which can generate highquality images in any style. In addition, real-time style transfer and super-resolution [14] uses the perceptual loss function to train a feedforward network for image style transformation and this method has achieved highresolution style transformation. AdaIN [15] employs a new normalization layer, where the mean and variance of the style image are used as radiometric parameters. As a result, arbitrary style transformations can be achieved. The swapping autoencoder [16] employs a texture swapping technique to achieve better results in texture performance. StyleGAN [17] employs a new generator architecture that can control the high-level attributes of the generated image, e.g., hairstyle and freckles. In addition, StyleGAN [17] can generate highresolution images, e.g., like 1024 × 1024 pixels. StyTr^2 [18] implements a visual transformer model, which is a transformer-based network for the image style transformation tasks. Content-Aware Positional Encoding (CAPE) was proposed to achieve better style transformation than previous methods. StyleFormer [19] deployes a transformer model in the encoder-decoder union. Here, by applying style and content features to the transformer, better results can be obtained in terms of preserving the content architecture and style patterns. These previous studies achieved satisfactory results for multistyle transformations of ordinary images. The goal of the current study is similar in that we attempt to realize multistyle transformations using style images. However, our target object is text images which is a more difficult style transformation task.

Recently, the CLIP [20] text-to-image matching model demonstrated excellent performance using text to guide image generation and this model has been used in the style transformation task. In addition, CLIPStyler [21] employs a pretrained CLIP model and a simple CNN network to transfer the given text description's semantic style to a target content image without requiring style images. StyleCLIP [22] can manipulate images with text descriptions as a condition through the CLIP model's text and image matching capability and StyleGAN's image generation capability. Differing from these methods, the proposed method utilizes style images (rather than text descriptions) to guide the generation of results.

2.3 Text Font Style Transfer

Unlike the style transformation of ordinary images, the style transformation of text must ensure the readability of the font while reflecting the style features; thus, style transformation of text images is a relatively difficult task. Through style transfer and style removal, the TET-GAN method [23] enables the network to learn to decompose and recombine the content and style features of style text images, which is useful for text image style transformation. In addition, FETGAN [24] employs an adaptive instance-normalized font style migration for few-shot learning, which solves the problem of converting existing fonts to the new style while keeping the text unchanged with only a few new style font samples. GlyphGAN [25] utilizes the GAN framework to create new fonts with a consistent style across all 26 letters in the English alphabet. In other words, this method can generate an infinite variety of fonts based on existing fonts. SwapText [26] can swap text content in scene images while preserving the original font, color, size, and background of the image. MC-GAN [27] enabled us to change letters to a particular style based on a few sample letters in that style. The Shape MatchingGAN [1] can transform text styles using a single style image and can control different degrees of style. The intelligent text style transfer method [28] generates decorated text style images by separating, transferring, and recombining decorative and basic text effects, and CLIPFont [29] achieves zero-shot text font style transformation with text description control using the CLIP model. Although these methods have demonstrated good results for text style transformation, including the design of new fonts, achieving multiple styles of text style transformation using a single network guided by style images has not yet been demonstrated.

3. Proposed Method

In this section, we first introduce the basic Shape MatchingGAN [1] network, and then we describe the proposed method in detail.

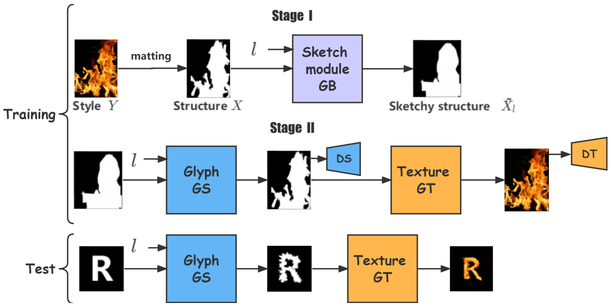

Fig. 1 Structure of Shape MatchingGAN [1].

3.1 Base Methods

The proposed method employs the Shape MatchingGAN [1] as the basic network. The structure of the Shape MatchingGAN is shown in Fig. 1. As can be seen, the network structure is mainly divided into two stages. In Stage I, the sketch module GB is employed to change the structure of style images X into different degrees of deformation via parameter $l(\in [0,1])$. In addition, the sketch module uses text images for training; thus, when style images are input to the network during the training process in Stage II, the edges of the output $\tilde{X}_l$ obtained from the sketch module appear more like the edges of the text. There are two modules in Stage II, i.e., the glyph module (GS, DS) and the texture module (GT, DT). Here, the glyph module enables the results $\tilde{X}_l$ of the different degrees of deformation obtained in Stage I back into the original style image shape to learn the structure features, and the texture module learns the texture features of the style image.

3.2 Proposed method Multi-Style SMGAN

When utilizing style images for multiple style transformation tasks, e.g., MUNIT [9], StyleGAN [17], and AdaIN [15], the corresponding networks employ a large number of style images for training to achieve multiple style transformations. However, it is difficult for text images to utilize such a large number of style images to achieve multiple style transformations. One reason for this difficulty is that acquiring a large number of varied style images is difficult. Another reason is that, for the multiple style transformation of text task, the network needs to learn both the structural features of the text images and the structural and textual features of the various style images. In addition, the network must adapt those features in a proper manner to generate natural text images, which greatly increases the burden on the network. Thus, in this study, we utilized Shape MatchingGAN [1], which only requires a single style image for text style transformation, as the basic network structure to achieve the multistyle text transformation task. This avoids the need to collect a large number of style images and reduces the learning burden of the multistyle network.

Our main goal is to enable the network to learn multiple

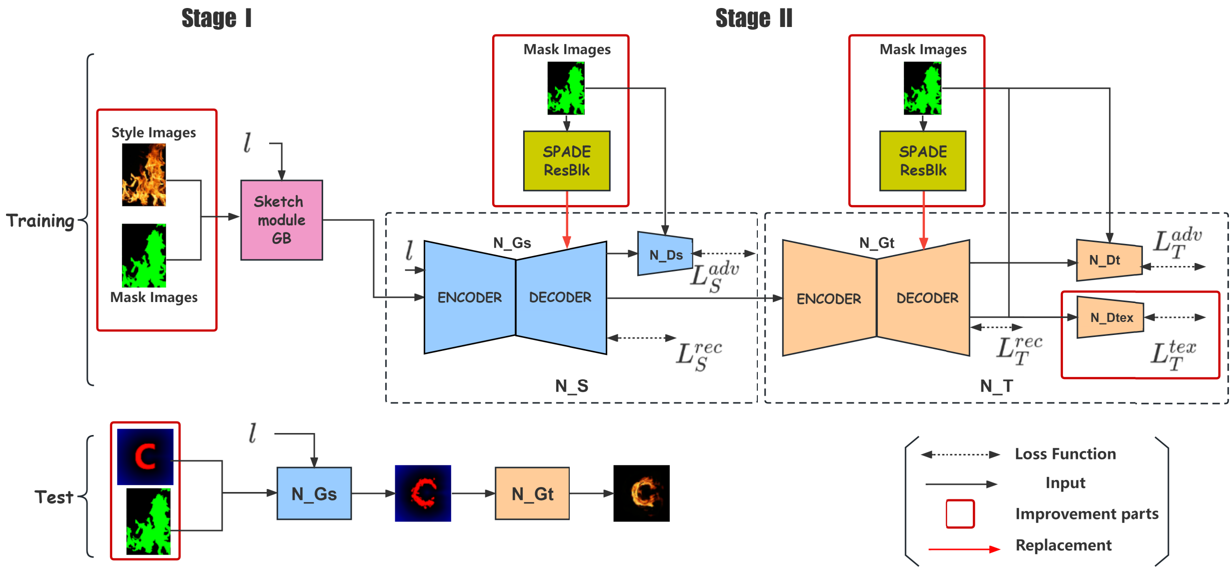

Fig. 2 Architecture of the proposed Multi-Style SMGAN method. We implement semantic mask images as conditional input to guide the generation of various styles. The SPADE ResBlk [2] is utilized to extract features from the mask images and the extracted features enable the network to learn the features of each style effectively. The parts surrounded by red lines represent our improvements to the Shape MatchingGAN [1] model.

styles effectively using a single style image for each style and enable users to control the style of the text images during the generation stage. Thus, the users can employ the trained model to achieve realtime multistyle transformation for text images. Here, the key questions are determining how to learn multiple styles in a single network effectively and what technique to use to control the generation of styles.

Note that SPADE [2] can generate realistic images using only a semantic mask image. The traditional batch normalization (BN) layer can easily lose the spatial information of the mask image; thus, SPADE [2] implements the spatially-adaptive normalization layer to address this issue. The core of modification lies in the calculation of the affine transformation parameters γ and β. In BN, the calculation of γ and β is realized via network training. In contrast, with the spatially-adaptive normalization technique, γ and β are obtained from the mask image through the network. Thus, inspired by these studies, we employ semantic mask images to control the different styles of the generated text images.

As shown in Fig. 2, the structure of the proposed method Multi-Style SMGAN is divided into two stages and comprises three networks based on the Shape MatchingGAN [1]. The three networks include the module GB, the structure network N_S, which contains generator N_Gs and discriminator N_Ds, and the texture network N_T comprising the generator N_Gt and two discriminators N_Dt and N_Dtex. In this study, we improved the basic Shape MatchingGAN [1] by adding semantic mask images as a condition to the network input in Stage I and Stage II. In addition, the normalization layer is utilized by SPADE ResBlk [2] rather than BN in the N_Gs and N_Gt networks in Stage II. We also implemented the discriminator N_Dtex to improve the quality of the generated images. In the testing phase, we employ the semantic mask images to select the style of the output text images.

3.2.1 Conditional Input

When using a single model to perform multistyle transformation, images of various styles must be input and the network needs to learn multiple style features effectively. However, Shape MatchingGAN [1] is designed to learn only a single style feature in the training process; thus, if multiple style images are input simultaneously, the network cannot identify the various styles. Conditional GAN [11] adds conditions to the network, that are used to control the generation of the image. Thus, in the proposed method, in reference to the Conditional GAN [11], we extract mask images for each style image and input the mask images as conditions into our network to guide the generation process. In addition, to allow the mask vector to control both the structure and texture of the style images, we add the mask images as a condition in all three networks of the proposed model. Here, the sketch module input is the initial input of the entire network; thus, we input the mask images in pairs with the style images. Therefore, the first convolution layer of the sketch module accepts 6-channel inputs that are the concatenation of style images and the corresponding mask images. We also utilize these mask images in the decoder parts of the network, i.e., N_Gs and N_Gt.

3.2.2 Multistyle Training

If we only add mask images to the model, the network will not learn the information about each mask image effectively, which can easily result in the integration of various styles in a single image. Note that this is discussed relative to our ablation experiments in Sect. 4.1.3.

The mask images that guide image generation in SPADE [2] have many different colored labels to distinguish the different parts of the image. Here, the different parts of the generated image labeled by the mask do not merge, and the different colored labels can generate their part of the image separately. The SPADE method implements a new SPADE layer that can effectively prevent the information about mask images from being washed out in the network, thereby making it possible for the network to learn the mask image information effectively. Then, in reference to the SPADE method, we extract the mask images in different colors for different styles to enable the network to separate each style. We improve the decoder parts of the network GS and GT in Shape MatchingGAN [1]. Specifically, we replace the typical normalization layer in the decoder parts of GS and GT with the SPADE ResBlk. In addition, the semantic masks of the style images are utilized as the input to the SPADE ResBlk.

By adding the mask images as conditions and implementing the modification to decoder parts, we can learn multiple styles effectively in a single model and control the generation of various style images using the mask images of the style.

3.2.3 Improving the Quality of the Generated Images

The discriminator in Shape MatchingGAN [1] is sufficient when the model only needs to learn a single style, however, it does not function as expected when learning multiple styles, especially for the texture of styles. Thus, the proposed method implements an additional discriminator in the texture network N_T to satisfy the quality requirements of multiple styles. Specifically, we add another PatchGAN discriminator N_Dtex to the network N_T in reference to the PatchGAN structure in the swapping autoencoder [16].

3.2.4 Loss Function

Shape MatchingGAN [1] uses text images to train the sketch module, and we employ the pretrained model directly. Thus we do not introduce the loss functions for the sketch module. We primarily focus on the loss functions of the networks N_S and N_T. For N_S network, in each of the loss functions, x_i represents the structural sketch of each style image obtained after the binary transformation process. Here, y_i represents a raw style image and i indicates which style image is used. We use a mask image $mask_i$ as guide information to reconstruct the features of the different style images. In this process, N is the total number of styles used.

The network N_S uses both reconstruction and adversarial losses. In the reconstruction loss, $\widetilde{x}_{li}$ is the result of each style structure image with different degrees of deformation obtained from GB's network, and l $(\in [0,1])$ represents the parameter that controls the degree of deformation. The reconstruction loss $\mathcal{L}_S^{rec}$ restores the original structure of the different degrees of the images for each style. In the adversarial loss $\mathcal{L}_S^{adv}$, we add the mask images to the generator and discriminator, which is similar to the conditional GAN method.

$$\mathcal{L}_S^{rec} = \sum_{i=1}^{N} \mathbb{E}_{x,l,mask}\left[\|G_s(\widetilde{x}_{li}|(l,mask_i)) - x_i\|_1\right], \qquad (1)$$

$$\begin{aligned}\mathcal{L}_S^{adv} &= \sum_{i=1}^{N} \mathbb{E}_{x,mask}\left[\log D_s(x_i|mask_i)\right] \\ &+ \sum_{i=1}^{N} \mathbb{E}_{x,l,mask}\left[\log(1 - D_s(G_s(\widetilde{x}_{li}|(l,mask_i))))\right]\end{aligned} \qquad (2)$$

The overall N_S losses are expressed as follows.

$$\mathcal{L}_{N\_S} = \min_{G_s}\max_{D_s} \lambda_S^{adv}\mathcal{L}_S^{adv} + \lambda_S^{rec}\mathcal{L}_S^{rec} \qquad (3)$$

The main task of the N_T network is to assign texture features to the structural images obtained in N_S. The discriminator used in the swapping autoencoder [16] can help the network learn texture features effectively; thus, we implement a new texture loss function $\mathcal{L}_T^{tex}$ to the N_T network. Here, $\mathcal{L}_T^{tex}$ is the loss function of the cooccurrence patch discriminator used in the swapping autoencoder [16]. Thus, N_T employs the reconstruction losses $\mathcal{L}_T^{rec}$, conditional adversarial losses $\mathcal{L}_T^{adv}$, and texture loss $\mathcal{L}_T^{tex}$. We also implement $mask_i$ as conditional information in each loss function. $\mathcal{L}_T^{rec}$ and $\mathcal{L}_T^{adv}$ are expressed as follows.

$$\mathcal{L}_T^{rec} = \sum_{i=1}^{N} \mathbb{E}_{x,y,mask}\left[\|G_t(x_i|mask_i) - y_i\|_1\right], \qquad (4)$$

$$\begin{aligned}\mathcal{L}_T^{adv} &= \sum_{i=1}^{N} \mathbb{E}_{x,mask,y}\left[\log D_t((x_i|mask_i), y_i)\right] \\ &+ \sum_{i=1}^{N} \mathbb{E}_{x,mask}\left[\log(1 - D_t(G_t(x_i|mask_i)))\right]\end{aligned} \qquad (5)$$

The overall N_T losses are expressed as follows.

$$\mathcal{L}_{N\_T} = \min_{G_t}\max_{D_t} \lambda_T^{adv}\mathcal{L}_T^{adv} + \lambda_T^{rec}\mathcal{L}_T^{rec} + \lambda_T^{tex}\mathcal{L}_T^{tex} \qquad (6)$$

3.3 Multi-Style SMGAN with SEAN

The Multi-Style SMGAN achieves our goal of multistyle text transformation. However, in the basic method Shape MatchingGAN, when dealing with styles that lack distinct nonsmooth shape features, the outcomes frequently fall short of expectations. Therefore, our model needs styles that exhibit unique and sharp shape features for training. Limited by the number of styles that fulfill our requirements, Multi-Style SMGAN can transform up to four styles simultaneously in a single model. When the number of styles exceeds four, maintaining distinct shape characteristics and avoiding nonsmooth edges in all style images becomes challenging. Consequently, styles that share similar shape features or lack

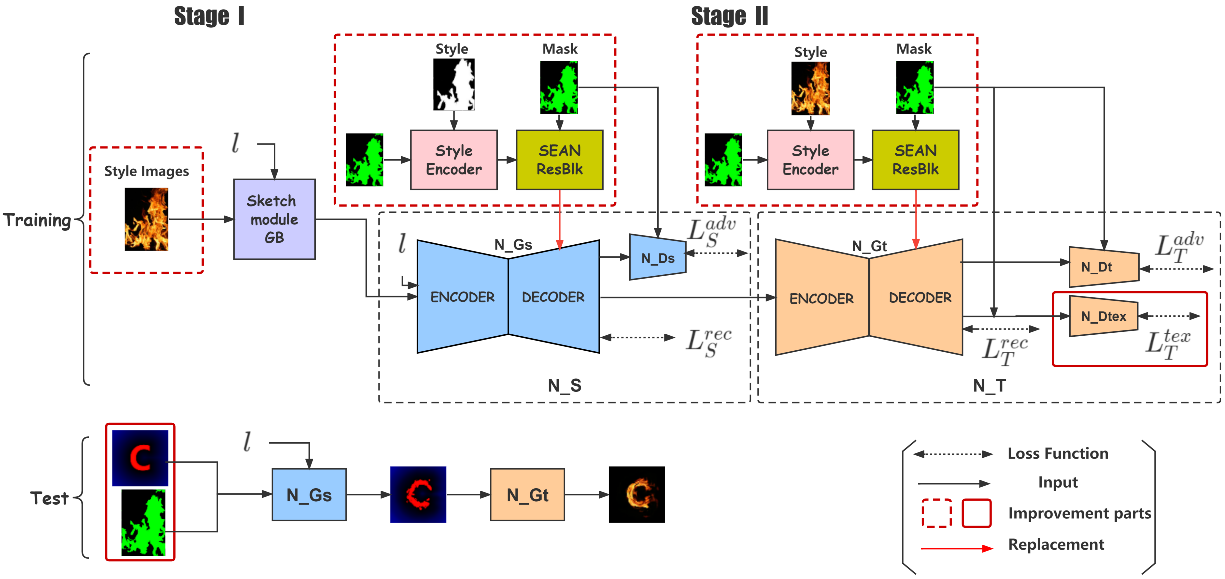

Fig. 3 The network structure of the proposed Multi-Style SMGAN method with a SEAN network. Modifications made to the proposed method that uses the SPADE network are indicated by the red dotted line. The red solid line identifies modifications to Shape MatchingGAN [1].

sharp shape features often lead to the loss of style characteristics, or style fusion in one single image. Thus, to address this issue, we employ the SEAN network rather than SPADE ResBlks to achieve more style transformation. Compared to SPADE [2], SEAN [3] provides a higher level of detail control. Thus, based on the Multi-Style SMGAN method, we implemented an improvement to achieve higher quality results and transfer more styles. Here, we primarily modified the two decoder parts of the Multi-Style SMGAN. Specifically, The SEAN [3] style encoder was added to the network, and the SPADE ResBlk was replaced by the SEAN ResBlk. In addition, the operation to add the mask images to the input side of Stage I was eliminated because the mask image information is used more effectively in the SEAN network. Here the first convolution layer of the sketch module is only 3-channel inputs of the style images, similar to the basic method. The input of the style encoder and the SEAN ResBlk is the same as that of SEAN [3], i.e., style images and semantic mask images. The structure of the proposed Multi-Style SMGAN with SEAN network is shown in Fig. 3.

4. Experiments

4.1 Multi-Style SMGAN

The basic structure of Multi-Style SMGAN is essentially the same as Shape MatchingGAN [1]. The primary modifications include replacing the three original normalization layers with SPADE ResBlks [2] in the decoders of the N_S and N_T networks, as well as adding a new discriminator to the N_T network. Note that a large number of additional SPADE ResBlks [2] increases the calculation costs and learning time; thus, it is reasonable to minimize the number of the SPADE ResBlks [2] with the assurance of image quality. Therefore, we only replaced three regularization layers. The structure of the new discriminator is designed in reference to the swapping autoencoder [16].

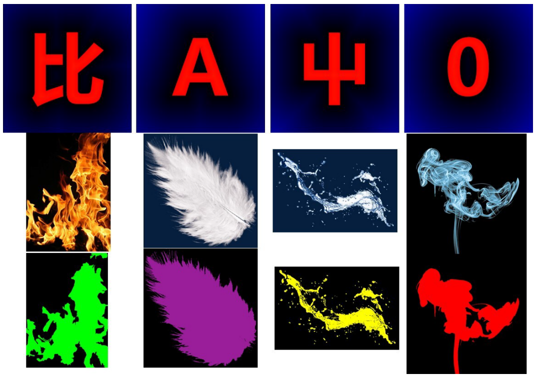

Fig. 4 Example images from the dataset. The first row is the text images, style images are in the second row, and corresponding style masks are in the last row.

4.1.1 Dataset

In our experiments, we used 129 text images provided by Shape MatchingGAN's authors and four style images. Note that we froze the Sketch model; thus, the text images were only used for the testing phase. We also added the corresponding colored mask images for the four style images. Here, the text images were transformed in Shape MatchingGAN [1] using the distance function. Figure 4 shows example images from the dataset.

4.1.2 Network Training

For the sketch module GB, we used the existing model data

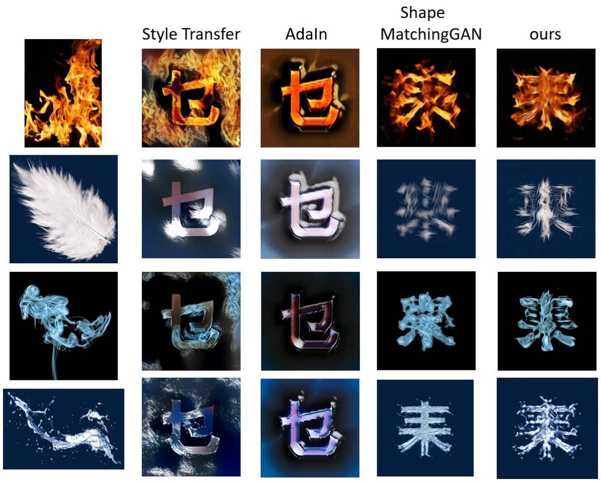

Fig. 5 Comparison between the results of the proposed method and three baseline methods. It is evident that AdaIN and Style Transfer are not very effective in terms of text style transformation. The proposed method and Shape MatchingGAN [1] achieved good text style transfer results.

of Shape MatchingGAN [1], and we primarily train on the improved parts of the network, e.g., style structure network and style texture network. In the training process, we input the style images and the corresponding mask images into the network in pairs. In the testing stage, we input the selected text image and style mask image to generate the corresponding style text image.

4.1.3 Experimental Results

The methods most similar to the proposed method are multistyle transfer methods; however, the MUNIT [9] and other similar methods require a large number of style images to perform style transformation. As a multidomain transformation method, StarGAN [10] also requires a large number of images for each of the target domains to learn. Thus, since only a single style image is required for the proposed method, it is difficult to directly compare our method with these existing methods. Therefore, we selected arbitrary style transformations for the comparison test. Note that the AdaIN [15] and neural style transfer [13] methods have performed well in achieving arbitrary style transformations. Consequently, we compared the proposed method with these methods and the baseline Shape MatchingGAN [1] qualitatively.

In AdaIN [15] and style transfer [13], we use the pretrained model to run experiments. Compared to the Shape MatchingGAN [1], the proposed method only needs to train a single model to achieve four transformations. In contrast, Shape MatchingGAN [1] trains four models separately to perform the same number of style transformations. The four styles of text images obtained from our model simultaneously were compared with the results generated in the Shape MatchingGAN [1] using the separately trained network, and with the results of the other two methods, e.g., AdaIN and style transfer. A comparison of the results is shown in Fig. 5.

The results obtained by the AdaIN [15] and Style Transfer [13] methods were poor because the style features aimed at text were not learned. The proposed method and Shape MatchingGAN are specific to the text; thus, these methods demonstrated better text style transformation performance. In a comparison with the basic Shape MatchingGAN [1] method, the multiple style results generated by a single model were as good as or better than the results obtained from Shape MatchingGAN [1], which was trained separately. As can be seen, the proposed method achieved multiple styles of text transformations using a single model and the results exhibit stylistic features in both structure and texture. Overall, the proposed method obtained excellent results for English letters, Arabic numerals, and Chinese characters. We found that the acquired text is readable, and multistyle transformation was realized. The results obtained by the proposed model are shown in Fig. 6. Note that using our trained model, the users can quickly get the stylized text image without waiting.

We also performed an ablation study of the proposed network, and the results are shown in Fig. 7. When only adding mask images to the network for input, the network cannot distinguish individual styles effectively, thereby resulting in merged styles, and in the absence of a texture discriminator, the style features were not learned adequately. The results of the full model demonstrate that the proposed method is effective in terms of learning multiple styles and produces good results.

4.1.4 User Study

We conducted a user study on Amazon Mechanical Turk on all four styles. Here, the test set included three types of images, i.e., four style images, 25 text images, and 50 transformed text images. Using the test set, we conducted a user study with 42 users. The users were shown the style image, the original text image, and the two transformed images. Note that one of the two transformed images was obtained by the proposed method, and the other image was obtained by the Shape MatchingGAN [1] method. Here, the users were asked to select the better result image relative to the following question: *“Which of the following not only ensures the readability of the text but also shows the style characteristics of the style image well ?”*

We counted the user votes for each style, and the results are shown in Fig. 10. As can be seen, the proposed method outperformed the baseline method regarding the three styles, and it was identical regarding the other one style, “fire style.”

4.2 Multi-Style SMGAN with SEAN

Here, the experimental environment was the same as that used in the SPADE experiment. In this experiment, based on the four styles, we increased the number of input style images to six (maple and ketchup were increased). The results are shown in Fig. 8. The added style images are shown on the rightmost side of Fig. 8. In terms of results obtained by the network utilizing SEAN, we found that the network realized six style transformations while maintaining the readability

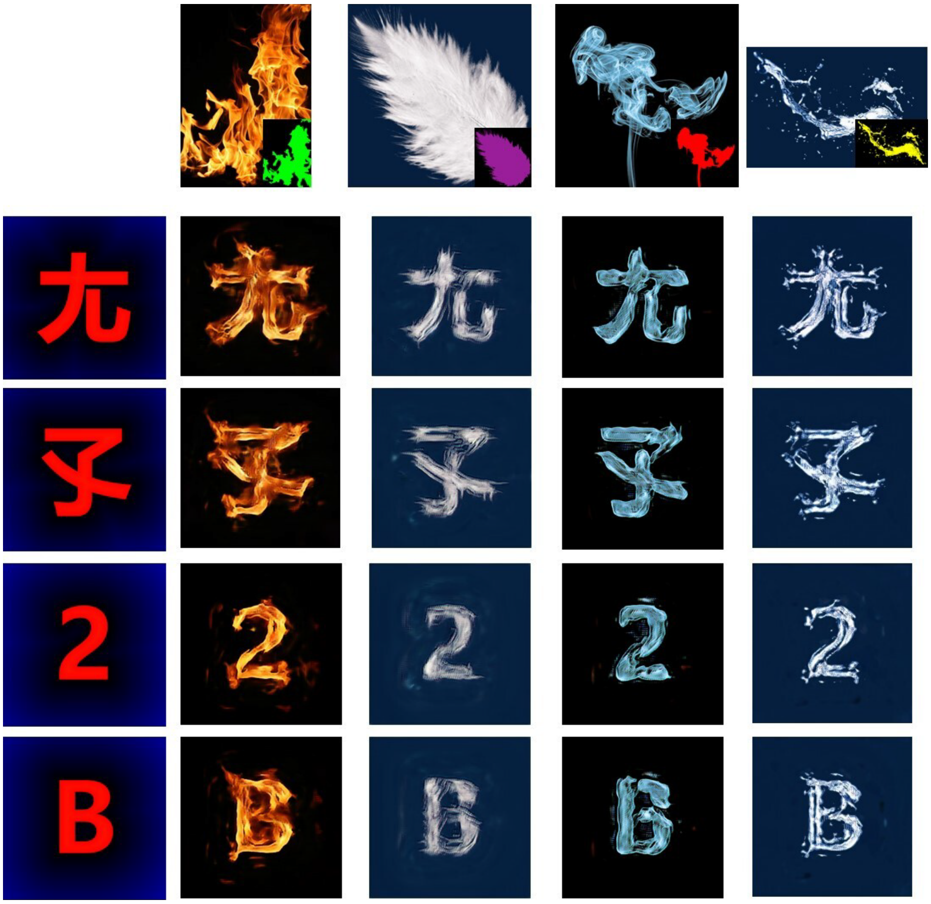

Fig. 6 Results obtained by the proposed method. The leftmost column shows the input text images. The top row shows the style images and the corresponding mask images. The results demonstrate that the proposed method implemented multistyle text transformation successfully.

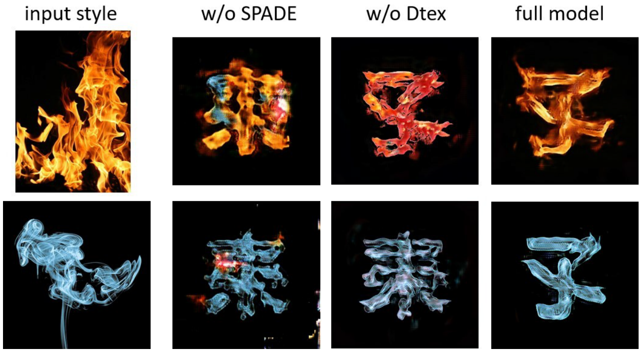

Fig. 7 Results of the ablation study.

of the characters. Compared to using the SPADE network, the SEAN network method increased the number of styles that can be transformed and achieved the same quality as the SPADE network method. Figure 9 compares the results of these two methods. As can be seen, the results obtained by the SPADE network method are clearer in terms of style feature performance, and the SEAN method exhibits better font maintenance. Overall, these two networks both achieved satisfactory results in style transformation.

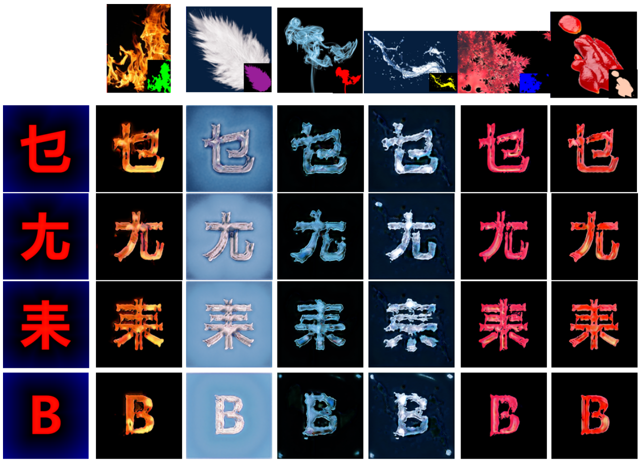

Fig. 8 Results obtained by the proposed method using the SEAN network.

We utilized the positive effects of the SEAN network for style feature extraction to mitigate the negative influence of the basic network when there are no unique shape features of the styles. Our approach enabled us to incorporate six different styles successfully, including the smooth-shaped "ketchup" style. Nevertheless, our network continues to

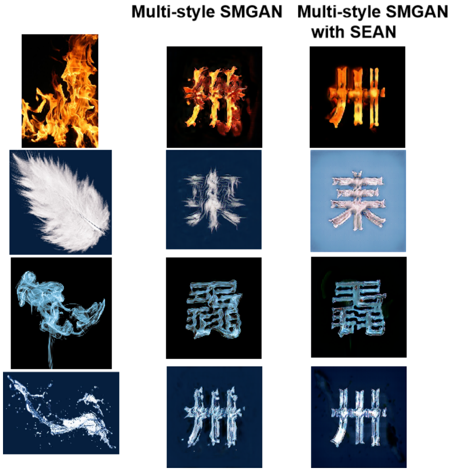

Fig. 9 Comparison of the results obtained using the Multi-Style SMGAN and Multi-Style SMGAN with SEAN methods.

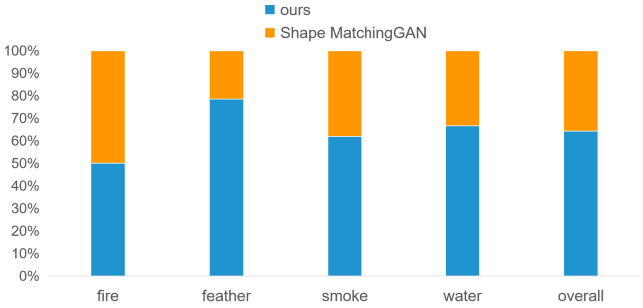

Fig. 10 Results of the user study.

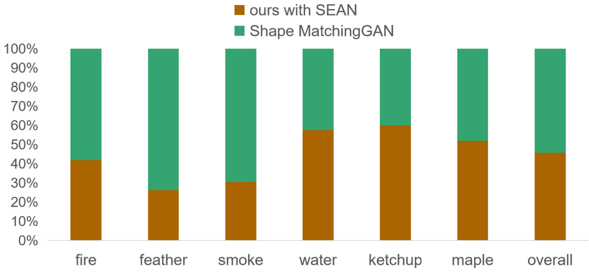

Fig. 11 Results of the user study with the SEAN network.

suffer from the adverse effects of the basic method, causing a loss of style features as more styles are introduced.

In addition, a user evaluation was conducted with 50 people using Amazon Mechanical Turk. Here, the questions and approach used in the experiment were the same as the SPADE method and we counted the user votes. As shown in Fig. 11, compared with the base Shape MatchingGAN method, we observe that the water, maple, and ketchup styles outperformed the base method, and the overall results are nearly the same. Furthermore, we conducted a user evaluation for our two proposed methods, following the conditions mentioned earlier. As depicted in Fig. 12, the overall performance across four common styles remained nearly identical. In Fig. 13, we present examples of user-preferred and non-preferred results for the basic method and our two proposed methods.

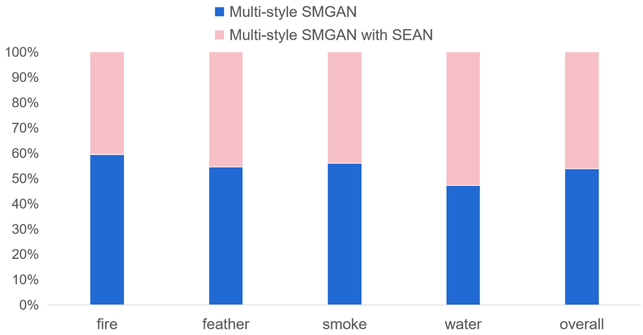

Fig. 12 Results of the user study with the two proposed methods.

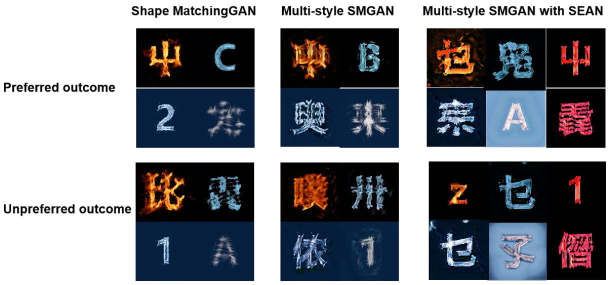

Fig. 13 Some sample of results that users have preferred and not preferred.

5. Conclusions

This paper has proposed the Multi-Style SMGAN transfer network for text images. By implementing masks for style images and reforming the network structure, the proposed method realizes the multiple style transformation task for text images in a single model using style images. Thus, multiple trained models are not required for each different style, and it is possible to achieve realtime style transformation using our trained models. In addition, the proposed method can control the generation of various styles of text images in the generation stage according to the style images. The experimental results demonstrate that the proposed method achieves a satisfactory effect in terms of generating multiple style text images.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Numbers, 22H00540, 22H00548, and 22K19808.

References

[1] S. Yang, Z. Wang, Z. Wang, N. Xu, J. Liu, and Z. Guo, “Controllable artistic text style transfer via shape-matching GAN,” Proc. IEEE Computer Vision and Pattern Recognition, pp.4442–4451, 2019.

[2] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu, “Semantic image synthesis with spatially-adaptive normalization,” Proc. IEEE Computer Vision and Pattern Recognition, pp.2337–2346, 2019.

[3] P. Zhu, R. Abdal, Y. Qin, and P. Wonka, "SEAN: Image synthesis with semantic region-adaptive normalization," Proc. IEEE Computer Vision and Pattern Recognition, pp.5104–5113, 2020.
[4] H. Yuan and K. Yanai, "Multi-style transfer generative adversarial network for text images," Proc. IEEE International Conference on Multimedia Information Processing and Retrieval (MIPR), pp.63–69, IEEE, 2021. https://ieeexplore.ieee.org/abstract/document/9565534.
[5] P. Isola, J.-Y. Zhu, T. Zhou, and A.A. Efros, "Image-to-image translation with conditional adversarial networks," Proc. IEEE Computer Vision and Pattern Recognition, pp.1125–1134, 2017.
[6] J.-Y. Zhu, T. Park, P. Isola, and A.A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," Proc. IEEE International Conference on Computer Vision, pp.2223–2232, 2017.
[7] J.Y. Zhu, R. Zhang, D. Pathak, T. Darrell, A.A. Efros, O. Wang, and E. Shechtman, "Toward multimodal image-to-image translation," Advances in Neural Information Processing Systems, pp.465–476, 2017.
[8] M.Y. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks," Advances in Neural Information Processing Systems, pp.700–708, 2017.
[9] X. Huang, M.Y. Liu, S. Belongie, and J. Kautz, "Multimodal unsupervised image-to-image translation," Proc. European Conference on Computer Vision, pp.172–189, 2018.
[10] 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," Proc. IEEE Computer Vision and Pattern Recognition, pp.8789–8797, 2018.
[11] M. Mirza and S. Osindero, "Conditional generative adversarial nets," arXiv preprint arXiv:1411.1784, 2014.
[12] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," Advances in Neural Information Processing Systems, pp.2672–2680, 2014.
[13] L.A. Gatys, A.S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," Proc. IEEE Computer Vision and Pattern Recognition, pp.2414–2423, 2016.
[14] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," Proc. European Conference on Computer Vision, pp.694–711, 2016.
[15] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," Proc. IEEE Computer Vision and Pattern Recognition, pp.1501–1510, 2017.
[16] T. Park, J.Y. Zhu, O. Wang, J. Lu, E. Shechtman, A.A. Efros, and R. Zhang, "Swapping autoencoder for deep image manipulation," arXiv preprint arXiv:2007.00653, 2020.
[17] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," Proc. IEEE Computer Vision and Pattern Recognition, pp.4401–4410, 2019.
[18] Y. Deng, F. Tang, X. Pan, W. Dong, C. Ma, and C. Xu, "StyTr^2: Unbiased image style transfer with transformers," arXiv preprint arXiv:2105.14576, 2021.
[19] X. Wu, Z. Hu, L. Sheng, and D. Xu, "Styleformer: Real-time arbitrary style transfer via parametric style composition," Proc. IEEE/CVF International Conference on Computer Vision, pp.14618–14627, 2021.
[20] A. Radford, J.W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., "Learning transferable visual models from natural language supervision," Proc. International Conference on Machine Learning, pp.8748–8763, 2021.
[21] G. Kwon and J.C. Ye, "Clipstyler: Image style transfer with a single text condition," Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.18062–18071, 2022.
[22] O. Patashnik, Z. Wu, E. Shechtman, D. Cohen-Or, and D. Lischinski, "StyleClip: Text-driven manipulation of stylegan imagery," Proc. IEEE/CVF International Conference on Computer Vision, pp.2085–2094, 2021.
[23] S. Yang, J. Liu, W. Wang, and Z. Guo, "Tet-gan: Text effects transfer via stylization and destylization," Proc. AAAI Conference on Artificial Intelligence, vol.33, no.01, pp.1238–1245, 2019.
[24] W. Li, Y. He, Y. Qi, Z. Li, and Y. Tang, "FET-GAN: Font and effect transfer via k-shot adaptive instance normalization.," Proc. AAAI Conference on Artificial Intelligence, vol.34, no.02, pp.1717–1724, 2020.
[25] H. Hayashi, K. Abe, and S. Uchida, "GlyphGAN: Style-consistent font generation based on generative adversarial networks," Knowledge-Based Systems, vol.186, p.104927, 2019.
[26] Q. Yang, J. Huang, and W. Lin, "Swaptext: Image based texts transfer in scenes," Proc. IEEE Computer Vision and Pattern Recognition, pp.14700–14709, 2020.
[27] S. Azadi, M. Fisher, V. Kim, Z. Wang, E. Shechtman, and T. Darrell, "Multi-content gan for few-shot font style transfer," Proc. IEEE Computer Vision and Pattern Recognition, pp.7564–7573, 2018.
[28] W. Wang, J. Liu, S. Yang, and Z. Guo, "Typography with Decor: Intelligent text style transfer," Proc. IEEE Computer Vision and Pattern Recognition, pp.5889–5897, 2019.
[29] Y. Song and Y. Zhang, "CLIPFont: Text guided vector wordart generation," Proc. British Machine Vision Conference, 2022.

Honghui Yuan received M.E. degrees from the Department of Informatics, the University of Electro-Communications Tokyo, Japan, in 2021. He is now a doctoral course student at the Department of Informatics, the University of Electro-Communications, Tokyo, Japan. He is working on font style transfer.

Keiji Yanai is a professor at Department of Informatics, the University of Electro-Communications, Tokyo, Japan. He received B.Eng., M.Eng. and D.Eng. degrees from the University of Tokyo in 1995, 1997 and 2003, respectively. From 1997 to 2006 he was a research associate and until 2015 he was an associate professor in the Department of Computer Science, the University of Electro-Communications, Tokyo. From November, 2003 to September, 2004, he was a visiting scholar at the Department of Computer Science, University of Arizona, USA. His recent research interests include computer vision and deep learning.