Font Style Transfer Using Neural Style Transfer and Unsupervised Cross-domain Transfer

Atsushi Narusawa, Wataru Shimoda, and Keiji Yanai

Department of Informatics, The University of Electro-Communications, Tokyo
1-5-1 Chofugaoka, Chofu-shi, Tokyo 182-8585 JAPAN
{narusawa-a,shimoda-k,yanai}@mm.inf.uec.ac.jp

Abstract. In this paper, we study about font generation and conversion. The previous methods dealt with characters as ones made of strokes. On the contrary, we extract features, which are equivalent to the strokes, from font images and texture or pattern images using deep learning, and transform the design pattern of font images. We expect that generation of original font such as hand written characters will be generated automatically by the proposed approach. In the experiments, we have created unique datasets such as a ketchup character image dataset and improve image generation quality and readability of character by combining neural style transfer with unsupervised cross-domain learning.

1 Introduction

Recently, in the research field of character recognition, various tasks come to be studied according to the progress of deep learning. For example, scene character recognition, analysis of ancient document and image captioning are widely and actively studied at present. On the other hand, an image generation task using generative adversarial network has drawn a lot of attention. In the field of character recognition, image generation is also considered as a method for applications such as shape change of character font and generation of new fonts, which are beneficial in the languages having a large number of characters such as Chinese and Japanese. Several researchers have already been studying about font generation and font transformation using deep learning methods. In this paper, we also focus on transformation of character fonts and tackle a novel paradigm to transfer unique features obtained from images, which have specific patterns or designs to font images. Our work is different from traditional character recognition, which aims to simply recognize character of images, and our work is classified as kinds of "character engineering".

Generating new fonts for Japanese and Chinese requires large cost because they have over thousand kinds of character. It is needed to generate new Japanese fonts automatically using image generation techniques from several design patterns. However, problems still remain in adaptation of generative adversarial network to character images such as improvement of readability and training with small dataset. Especially, the problem is widely known that we need a

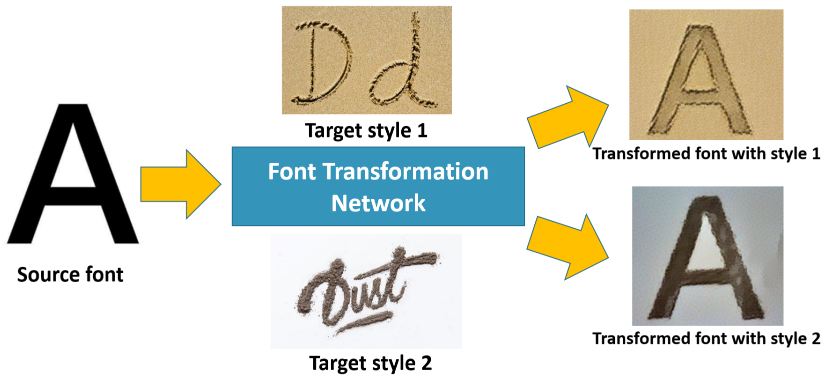

Fig. 1. Font transformation from a source font to the fonts stylized with the target styles.

variety large scale dataset for high quality image generation tasks using deep learning. However, preparing the dataset for learning transferring patterns from same characters is difficult. To solve the problems, we focus on learning transferring patterns from images which do not have correspondence of character. That is, it is unsupervised character transformation where target domain patterns are not limited to characters but to an assembly of stoke patterns.

Our objective in this paper is to train a font transformation network by unsupervised domain transfer which needs no corresponding image pairs over two domains as shown in Figure 1. In this figure, the source font, "A" was transformed into two kinds of fonts with two kinds of the target sand styles.

2 Related works

2.1 Existing font image generation methods

We have two kinds of fonts: vector fonts and bitmap fonts. In general, regarding vector fonts generation, a character is considered as combination of set of stroke such as curve and upward brush-stroke and several components such as unit of radical. Zong et al. and Miyazaki et al. [15, 9] decomposed characters into strokes and assigned them to corresponding transferred strokes. Lin et al. and Songhua et al. [8, 12] generated Kanji by combining strokes and radicals.

Previous approaches make skeletons using vector information in advance for decomposing characters into strokes and generate fonts automatically by estimating and extracting strokes for the reconstruction of Kanji. Preparing description of stroke for every Kanji requires large annotation cost. On the contrary, we can extract strokes automatically using deep learning from font images and train networks for the correspondence of transformation automatically. On the other hand, generation of artistic character image such as ornamental writing including typography [11] has drawn attention. Methods for transferring texture and patterns images to fonts are needed. In this work we try to train deep neural transformation networks by learning the correspondence between strokes of character image for transferring texture and patterns.

2.2 Font image generation using deep neural network

“Rewrite”[1] and “Zi2Zi”[2] are notable works in character image generation using deep learning. “Rewrite” is a project to modify Neural Style Transfer [3] for adapting it to the font image generation. Neural Style Transfer is a method to mix two types of images. There are several reports [2] to use neural style transfer for the font image generation. “Zi2Zi” is a project based on Pix2Pix [10] to transform an image to another-domain image using generative adversarial network for character image. They used an encoder network for representing features of an image and a decoder network for reconstruction from the features. They concatenated both networks as a transformation network for learning the correspondence between pair of images. In this work, we extend the project “Zi2Zi” by adapting cross domain learning without pair of images.

3 Method

3.1 Overview of the proposed method

We propose a method to combine Fast Style Transfer [4, 1] with unsupervised cross-domain learning using generative adversarial network [5, 14, 6, 13]. The overall architecture of the proposed transformation network is shown in Figure 2.

The networks for cross-domain learning consists of two transformation networks 3 and a discriminator networks 4. In cross-domain learning, we use two independent networks to convert the domain both directions. We define two transformation networks as G, F and two discriminator networks as D_x, D_y, respectively. Note that the transformation network, F, carries out inverse transformation of G. The transformation networks are trained with cycle loss, L_{cycle}, which enforces that any image applied with F and G goes back to the original image, i.e. $x = G(F(x))$. In addition, the transformation network is also trained with Adversarial Loss, $L_{adversarial}$, which enforces that the images transformed with G belong to the domain X and the images transformed with F belong to the domain Y. By optimizing these loss functions, all the networks are trained, and we obtain the transformation network which transform source fonts to the fonts stylized with the target style.

To improve readability in cross domain learning, we focus on losses of style transfer: content loss, $L_{content}$, and style Loss, L_{style}. In the standard style transfer, we extract features from per-trained model of VGG16 and the content features are extracted from one layer and style features are extracted from four

[1] https://github.com/kaonashi-tyc/Rewrite
[2] https://kaonashi-tyc.github.io/2017/04/06/zi2zi.html

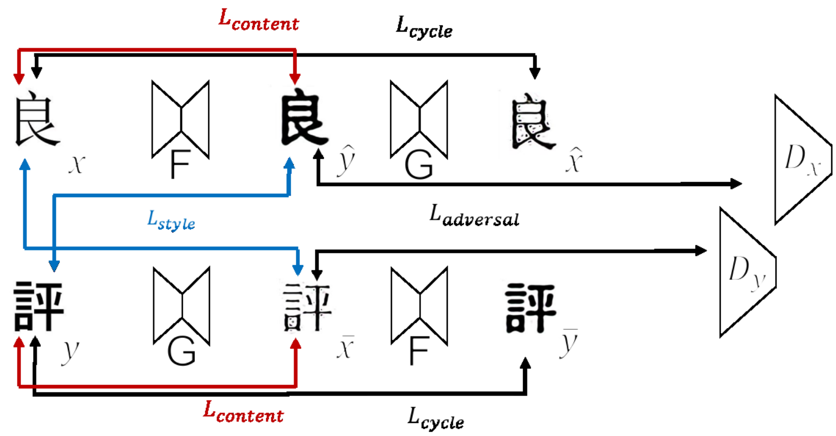

Fig. 2. The overall architecture of CycleGAN with neural style transfer.

Table 1. Configuration of the weight of loss.

Style Weight	3.00E+05
Content Weight	1.50E+00
Adversarial Weight	4.50E+08
Cycle Weight	3.50E+12

layers as shown in Figure 5. In the proposed method, we combine four kinds of loss functions, and the transformation network is trained by optimizing an Equation 1. Each of the weights for the loss functions is defined by grid search, and the values are indicated in Table 1.

$$L_{total} = \alpha L_{style} + \beta L_{content} + \gamma L_{adversarial} + \delta L_{cycle} \tag{1}$$

4 Experiments

4.1 Datasets

In this paper we transform font images to three kinds of texture pattern image dataset ketchup character, sand character and rope character images as shown in Figure 6. The number of samples of each of the fonts are shown in Table 2. The ketchup character image dataset consists of many character images, while the sand character dataset includes several English words and handwritten arts images. The rope pattern dataset does not include any character images so that the dataset is constructed by cropping rope art images by hand manual operation. Therefore, we select 16 images from each dataset and locate them in a square to obtain enough style features. We prepare the 500 arranged images respectively.

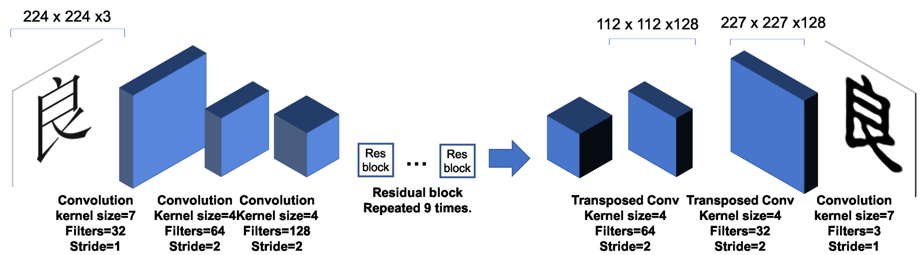

Fig. 3. The detail of encoder-decoder network.

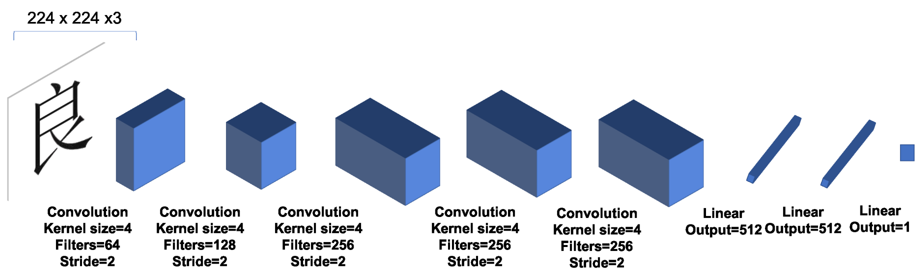

Fig. 4. The detail of discriminator.

4.2 Example of image generation

We show the image generation results of neural style transfer, CycleGAN and the proposed methods in Figure 7. The results include only one direction so that a purpose of the purposed method is to transfer texture to font images. Note that in this case the networks are trained with larger weights value of content loss than indicated in Table.1. From the results, we observe that character shape of generated images become clearer than CycleGAN. Especially, in case of the sand character dataset, the results of neural style transfer (style loss + content loss) does not keep readability, while the proposed method keeps the readability. In addition, though there is a trade off between style loss and content loss, adversarial loss and cycle loss complement readability and transferring texture in several cases.

Table 2. Texture character and pattern dataset

dataset	number
ketchup character	445
sand character	483
rope pattern	796

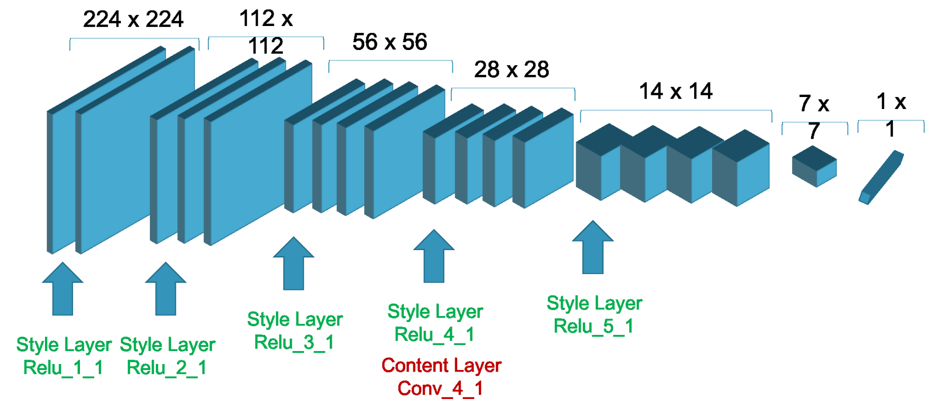

Fig. 5. Choice of layer for style features and content features.

Fig. 6. Examples of texture character and pattern dataset (From the left we show the examples of ketchup character, sand character and rope pattern images)

4.3 Objective evaluation and subjective evaluation

In this subsection, we explore combination of the losses to show effect of style loss and content loss. According to Figure 8, we can confirm that the readability is improved by content loss and adversarial loss leads natural texture of generated images visually.

Furthermore, we compared mean of style loss to evaluate transferring texture of the transformation networks. We pick up a style image from each dataset and compute style loss for every generated image. We show mean of style losses in Table 3. Results of neural style transfer should show best performance so that optimize this style loss to be small during image generation. The proposed method aims to close the mean of style loss to the performance of neural style transfer. In the ketchup character image dataset, the combination of style loss, cycle loss and content loss achieved best performance. If adversarial loss is added to the combination, the mean of loss become worse than the CycleGAN. On the contrary, in the sand character image dataset there are little difference between

Fig. 7. From the left, we demonstrate the results of the neural style transfer, CycleGAN and the proposed method.

neural style transfer and CycleGAN. In the proposed method results style loss does not work well, while adversarial loss show improvements. Finally, in the case of rope pattern image dataset, the tendency is similar to the case of ketchup dataset and all losses contribute the performance.

From the experimental results, combination of style loss, content loss and cycle loss or combination of all the losses is the better choice. Figure 9 shows the results of both simple and complicated characters with the model trained with all the losses. Compared with both, the results of the simpler samples were better in the quality, and the complicated characters were harder to transform, especially for the rope style.

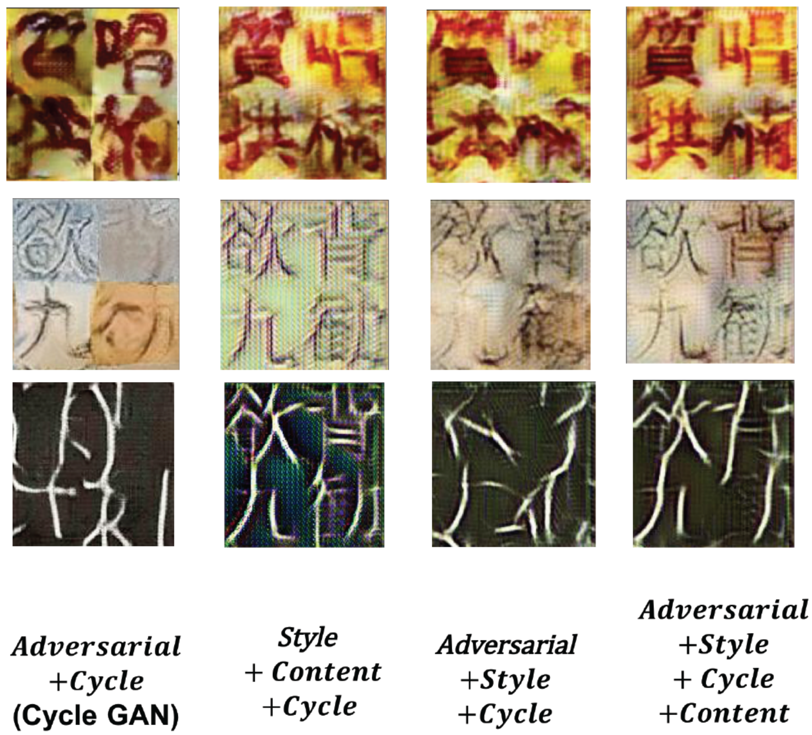

Fig. 8. The comparison of combination of the loss.

Table 3. The mean of the style loss.

	ketchup character	sand character	rope pattern
Style + Content (Neural Style)	5.17E+06	2.77E+06	2.03E+07
Adversarial + Cycle (CycleGAN)	6.03E+06	3.11E+06	2.74E+07
Style + Cycle + Content	5.66E+06	3.36E+06	2.17E+07
Adversarial + Style + Cycle	5.98E+06	2.69E+06	2.06E+07
Adversarial + Style + Cycle + Content	6.00E+06	2.71E+06	1.99E+07

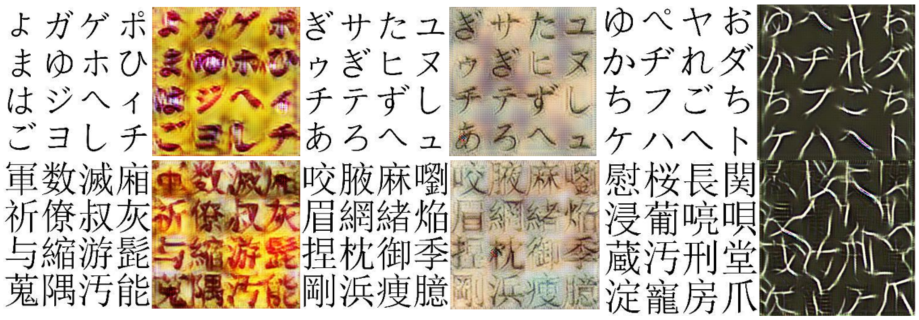

Fig. 9. (upper row) the results of simple characters, "Hiragana". (lower row) the results of complicated characters, "Kanji". For both, we used all the losses (style, content, cycle, and adversarial losses).

5 Conclusions

In this paper, we proposed a method to combine neural style network with CycleGAN by adding style loss and content loss to the CycleGAN model. We optimize four types of loss adversarial loss, cycle loss, style loss and content loss in the proposed method. These losses compete with each other, and we have explored the combination of the loss. We observed that the effective combinations differed in each dataset. The finding in the experiments over all the dataset is that content loss keeps original image character structure. The combination of content loss and style loss sometimes did not work, though there were also cases that show improvement by adding cycle loss. We consider that there are difference between Gram matrices obtained from the font images include many straight line and distortion in the shape. We also considered that the difference is relaxed by cycle loss and adversarial loss so that the losses sometimes cause distortion in straight line parts and it has potential to generate co-occurrence in the different domain data.

Currently, it is difficult to estimate patch-level correspondence on complicated patterns such as Kanji letters between source and target images. In general, the readability will not be changed if the shape of characters is changed but the structure is not changed. As future works, we consider to perturb the shape of input image to make it easy to find correspondence between sources and targets. In addition, we plan to introduce a patch-based approach [7] which used not Gram matrix which forced global consistency but only patch-based correspondences.

Acknowledgments: We would like to express great thanks to Prof. Seichi Uchida, Kyushu University, for the insightful and helpful comments. This work was supported by JSPS KAKENHI Grant Number 15H05915, 17H01745, 17H05972, 17H06026 and 17H06100.

References

1. D. Ulyanov, V. Lebedev, A.V., Lempitsky, V.: Texture networks: Feed-forward synthesis of textures and stylized images. In: arXiv:1603.03417v1 (2016)
2. Gantugs, A., Iwana, B.K., Narusawa, A., Yanai, K., Uchida, S.: Neural font style transfer. In: International Conference on Document Analysis and Recognition (2017)
3. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: Proc. of IEEE Computer Vision and Pattern Recognition (2016)
4. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: Proc.of European Conference on Computer Vision (2016)
5. Jun-Yan, Z., Taesung, P., Phillip, I., Alexei, E.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proc.of IEEE International Conference on Computer Vision (2017)
6. Kim, T., Cha, M., Kim, M., Lee, J.K., Kim, J.: Learning to discover cross-domain relations with generative adversarial networks. In: International Conference on Machine Learning (2017), http://arxiv.org/abs/1703.05192
7. Li, C., Wand, M.: Combining markov random fields and convolutional neural networks for image synthesis. In: Proc. of IEEE Computer Vision and Pattern Recognition (2016)
8. Lin, J., Hong, C., Chang, R., Wang, Y., Lin, S., Ho, J.: Complete font generation of chinese characters in personal handwriting style. In: Computing and Communications Conference (IPCCC), 2015 IEEE 34th International Performance. pp. 1–5. IEEE (2015)
9. Miyazaki, T., Tsuchiya, T., Sugaya, Y., Omachi, S., Iwamura, M., Uchida, S., Kise, K.: Automatic generation of typographic font from a small font subset. In: arXiv preprint arXiv:1701.05703 (2017)
10. Phillip, I., Jun-Yan, Z., Tinghui, Z., Alexei, E.: Image-to-image translation with conditional adversarial networks. In: Proc. of IEEE Computer Vision and Pattern Recognition (2017)
11. Shuai, Y., Jiaying, L., Zhouhui, L., Zongming, G.: Awesome typography: Statistics-based text effects transfer. In: Proc. of IEEE Computer Vision and Pattern Recognition (2016)
12. Songhua, X., Francis C.M., L., Kwok-Wai, C., Yunhe, P.: Automatic generation of artistic chinese calligraphy. In: IEEE Intelligent Systems (2005)
13. Taigman, Y., Polyak, A., Wolf, L.: Unsupervised cross-domain image generation. In: International Conference on Learning Representation (2017)
14. Yi, Z., Zhang, H., Tan, P., Gong, M.: Dualgan: Unsupervised dual learning for image-to-image translation. In: Proc.of IEEE International Conference on Computer Vision. pp. 2849–2857 (2017)
15. Zong, A., Zhu, Y.: Strokebank: Automating personalized chinese handwriting generation. In: Advancement of Artificial Intelligence. pp. 3024–3030 (2014)