Few-shot style transfer over Chinese fonts using GAN

Hanxi Guo and Xiaozhou Zhou
Shanghai Jiao Tong University

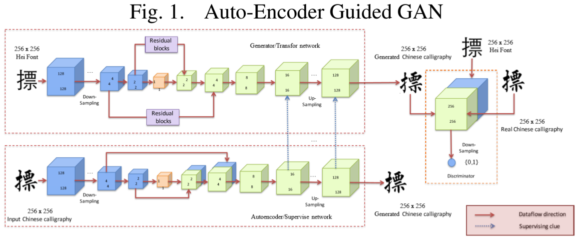
1. INTRODUCTION

We found that there are already sufficient numbers of works on English characters and handwritten numbers, however, the works on Chinese characters are much fewer. We thought the reason may be that compared with English characters and handwritten numbers, Chinese characters have much more complicated structures which are very hard for networks to learn. Additionally, we also found that the existing works are almost all based on large quantity of samples. They don't use few-shot methods. Therefore, in this report, we explain our efforts on Chinese character style transfer using generative adversarial networks.

2. BACKGROUND AND RELATED WORKS

Chinese Character Synthesis: In the past few years, there are many works on Chinese characters synthesis. In 2005, a method based on shape analysis technique and hierarchical parameterization to automatically generate novel artistically appealing Chinese calligraphy artwork from existing calligraphic artwork samples for the same character was proposed [Xu et al. 2005]. In 2017, a deep neural network based model which can generate calligraphy images from standard font images directly was proposed [Lyu et al. 2017]. The model in this paper is:

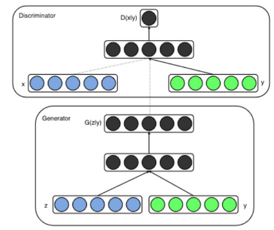
Fig. 1. Auto-Encoder Guided GAN



The whole network was trained end-to-end and they used the autoencoder to supervise the low feature of the generator in the training phase. The researchers also created a benchmark to test the existing methods.

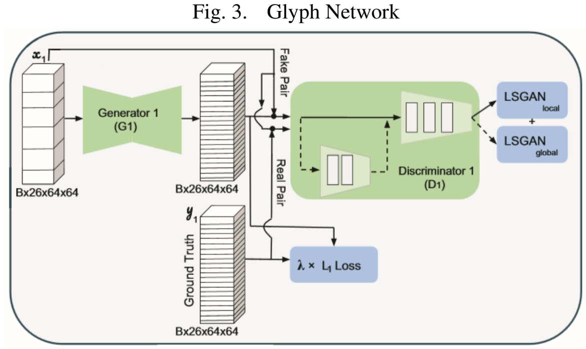
Generative Adversarial Networks: Recently, GAN was very popular in computer vision. Many researchers have developed various models. In 2014, conditional generative adversarial nets [Mirza and Osindero 2014] was proposed. conditionaql generative adversarial nets can be constructed by simply feeding the data that are wished to condition on to both the generator and discriminator. The model is:

Fig. 2. Conditional Generative Adversarial Nets



This model can generate MNIST digits conditioned on class labels and can also be used to learn a multi-modal model. In 2018, Multi-Content GAN [Azadi et al. 2018] was proposed. The researchers also did few-shot style transfer, but they just used English characters. The Glyph Network propsed in this paper inspired us a lot:

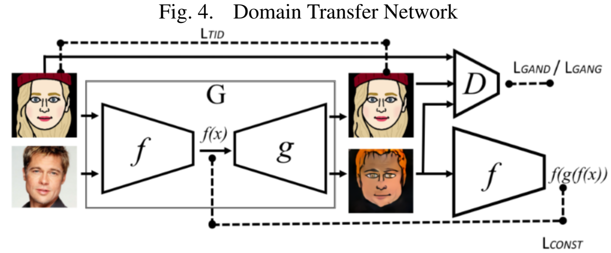
Fig. 3. Glyph Network



In this paper, researchers focused on the challenge of taking partial observations of highly-stylized text and generalizing the observations to generate unobserved glyphs in the ornamented typeface. They propose an endto-end stacked conditional GAN model considering content along channels and style along network layers. Their proposed network transfers the style of given glyphs to the contents of unseen ones, capturing highly stylized fonts found in the real-world such as those on movie posters or infographics. They based their experiments on their collected data set including 10,000 fonts with different styles and finally demonstrated effective generalization from a very small number of observed glyphs.

Style Transfer: In 2016, the Domain Transfer Network(DTN) [Taigman et al. 2016] was proposed. This network employs a compound loss function that includes a multiclass GAN loss, an f-constancy component, and a regularizing component that encourages a generative function to map the samples from a domain to themselves. The researchers applied this method to visual domains including digits and face images and demonstrate its ability to generate convincing novel images of previously unseen entities, while preserving their identity. The model is as follow:

Fig. 4. Domain Transfer Network



In 2017, researchers [Yang et al. 2017] exploit the analytics on the high regularity of the spatial distribution for text effects to guide the synthesis process. Researchers characterize the stylized patches by their normalized positions and the optimal scales to depict their style elements. As this paper expressed, the method shown first estimates these two features and derives their correlation statistically. They are then converted into soft constraints for texture transfer to accomplish adaptive multi-scale texture synthesis and to make style elemen tdistribution uniform. It allows the algorithm to produce artistic typography that fits for both local texture patterns and the global spatial distribution in the example.

3. MOTIVATION

Since we found that the works on English characters and handwritten numbers are already adequate and the results are quite well, we thought that whether we could come up with a method using the exsiting works to implement style transfer over Chinese fonts which doesn't be researched sufficiently yet. And because the style of a font is often very identical and can be shown in a few characters, we thought that it is possible to extract the style information from a small set of Chinese characters. Therefore, we wanted to implement the few-shot style transfer over Chinese fonts.

GAN is a very popular network structure recently in computer vision field and it performs well in generating pictures, thus we thought that we could use GAN to help us generate the Chinese characters when we extract the style information.

4. PROBLEM FORMULATION

In this paper, we have two given datasets. One is the standard library having a large amount of Chinese characters with various fonts styles. We use this standard library to train the encoder to distinguish different Chinese characters and get the characteristic vector which will be used to generate new characters. The other dataset is the stylization set having a new font style which is not collected by the standard library. The stylization set just have a few samples. We use the stylization set to extract the style information of this new font style. Then, we use the characteristic vector gained from the standard library and the style information to generate many other new Chinese characters having this new font style.

5. PROPOSED METHODS

5.1 Dataset Generation

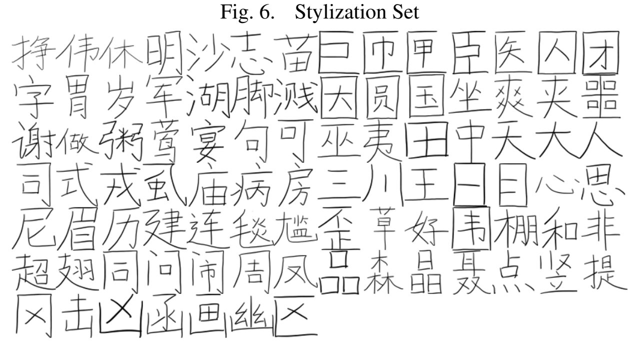
According to our problem formulation, we need to use two separate datasets. One is a standard library and the other one is a stylization set possessing a new font style. Since there are few related works about style transfer over Chinese fonts, we could hardly find two perfect datasets that fit our purpose well. Therefore, we decided to generate the two datasets by ourselves.

To generate the standard library which contains a large amount of Chinese characters with various fonts styles, we used **cv2**. We first collected about eighty different fonts files. Then we made a list of characters that we wanted to generate. After that, we started to use **cv2** to draw the Chinese character on a new blank canvas. Since we wanted to make the samples in the dataset more suitable to train the network in the next following steps, we cut the white parts that don't contain information of the generated pictures and resized all the samples to 64×64. In this way, the samples are in the same size which are good to the training process. We finally got nearly 35,000 samples for 500 Chinese characters. The samples are shown as the following example figure:

Fig. 5. Generated Samples of Standard Library

To generate the stylization set with a new font style, we decided to use handwritten Chinese characters. We first chose nearly 90 Chinese characters wich contain almost all the structures of Chinese characters. After that, we used graphics tablet to write the characters and then used the similar method that we also used to generate the standard library. Finally, we got the stylization set, shown as follow:

Fig. 6. Stylization Set



5.2 Image synthesis

In general, we divide the contents of a character into two independent parts, glyph information and style information. The glyph, including information like the relative position of different strokes and components in local field, determines what the character is. The style information determines how all the components are organized, for example, the line width and curvature of a stroke. If we can disentangle glyph information from style information, it would be easy for us to swap and reconstruct new images.

To obtain glyph information, we train an encoder along with a weak classifier. The encoder encodes the picture into latent space, and the classifier tries to distinguish among this latent space. In order to raise classification accuracy, the encoder will learn to extract glyph features and make abstraction. In our standard library, every category has images of the same character in different styles. After the encoder and the classifier are trained and well-tuned, we then can extract the glyph information among all the classes and find their clustering points. By doing so, we can obtain characteristic vector for every class, representing the critical features of one category.

After the glyph information is obtained, we also need to get style information. To accomplish this, we train a generative adversarial

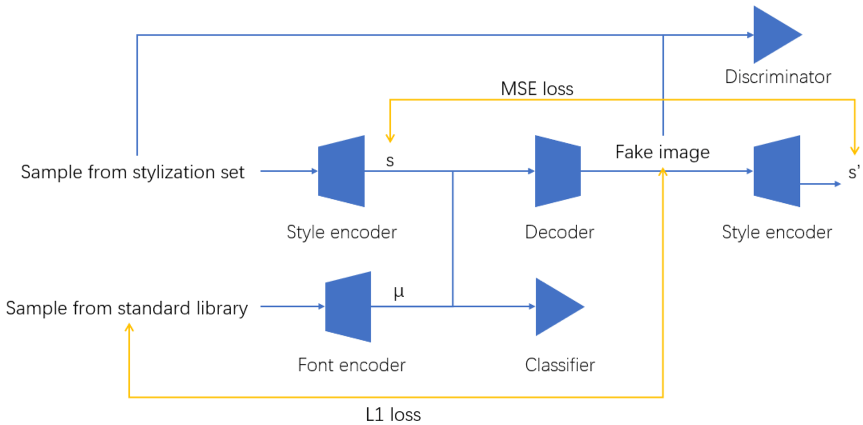


Fig. 7. Architecture of the network

Fig. 8. (a)

Fig. 9. (b)

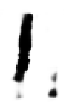


Fig. 10. (c)

net, which will first encode the picture from stylization set into a style vector denoted as s, then feed this style vector along with glyph vector into the generator to synthesize a new image. The new image is then fed to the same style encoder, and the output is denoted as s'. We hope that the distance of s and s' to be close, so we carry out mean square error loss on them. Meanwhile, the generated image cannot look too different from the image that contributes the glyph vector, so we also carry out L1 loss on the two images. Apart from all these, a discriminator will try to judge if the generated image looks like real handwritten images or not. The task for the generator is trying to fool the discriminator and let the discriminator make wrong judgements.

6. EXPERIMENTS

In our experiment, the first big problem we met is that, the samples for every category are not enough. For the 3872 characters, we only have 12 images for every one of them. We carried out two turns of data augmentation, but still, the problem remains. So we decided to use only a small subset of the library for glyph information. We pick out 50 characters and every class has around 70 samples. The classification accuracy among this picture reaches 96%. We encoded all the picture into vectors and found that, the mean square errors within one class vary from 0.802 to 1.118, suggesting that the glyph vector for different classes of the same character was close to each other.

In training the generative model, we update the weight of the generator and discriminator by turn. After around 200 epoches, the discriminator converged, but the loss of generator still changed drastically. We stopped training and checked our model. The output of the generator is shown as above, where (a) is sampled from standard library for glyph information, (b) is the an instance in stylization set and (c) is the output of the generator.

The other outputs are similar cases that has blurred and discontinuous sketches, suggesting that our solution fails. After simple analysis, we think that the main reason is that we failed in extracting style information with all these limited stylization samples. On the other hand, the domain gap between our handwritten stylization set and print-body standard library is too large, yet we didn't carry out domain generalization or adaption method to fill in the gap. The result might be better if we make those efforts.

7. CONCLUSION

Style transfer for Chinese characters is very difficult, especially when there are only limited samples. But this is of great significance and can be applied in many situations. We will continue trying to solve this big problem.

REFERENCES

Samaneh Azadi, Matthew Fisher, Vladimir Kim, Zhaowen Wang, Eli Shechtman, and Trevor Darrell. 2018. Multi-content gan for few-shot font style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 11. 13.

Pengyuan Lyu, Xiang Bai, Cong Yao, Zhen Zhu, Tengteng Huang, and Wenyu Liu. 2017. Auto-Encoder Guided GAN for Chinese Calligraphy Synthesis. In *Document Analysis and Recognition (ICDAR), 2017 14th IAPR International Conference on*, Vol. 1. IEEE, 1095–1100.

Mehdi Mirza and Simon Osindero. 2014. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784* (2014).

Yaniv Taigman, Adam Polyak, and Lior Wolf. 2016. Unsupervised cross-domain image generation. *arXiv preprint arXiv:1611.02200* (2016).

Songhua Xu, Francis CM Lau, William K Cheung, and Yunhe Pan. 2005. Automatic generation of artistic Chinese calligraphy. *IEEE Intelligent Systems* 20, 3 (2005), 32–39.

Shuai Yang, Jiaying Liu, Zhouhui Lian, and Zongming Guo. 2017. Awesome typography: Statistics-based text effects transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 7464–7473.