CS230

Unpaired Traditional Chinese Calligraphy Generator from Chinese Characters Using Cycle GAN

Ta-Hsuan Chao
Department of Electrical Engineering
Stanford University
thchao@stanford.edu

Meng-Jin Lin
Department of Electrical Engineering
Stanford University
mm1130@stanford.edu

Abstract

Chinese calligraphy is a tradition art form of handwriting, and our goal of this final project is to generate calligraphy written by four different calligraphers from typed version of Chinese characters in the kaite font-family. However, due to the rareness of Chinese calligraphy characters by different calligraphers, we would have to use unpaired characters to do the training. In this final project, we successfully demonstrated that we can create Chinese characters according to their calligrapher writing style by Cycle GAN. We showed our generation results as well as strengths and weaknesses of our model in this project.

1 Introduction

Chinese calligraphy is a traditional art form of handwriting, and it has rich history and ancient beauty. What's special about Chinese calligraphy is that it is unique to every calligraphist, and different works from different people possess beauty in its own style. In this final project, our motivation is to generate traditional Chinese characters, as in fig 1 written in the traditional calligraphy style given the specific typed Chinese characters.

However, this task is very challenging because calligraphy differ by writing style, brushes, stroke, and touch, so generation of calligraphy is subjective and diverse. But most importantly, calligraphy is unique to each writer, and we cannot guarantee that every writer would write every word in Chinese. This result is that paired data of specific Chinese characters to specific calligraphers are very hard to find and generate. Therefore, generation using paired data, such as the Generative Adversarial Net (GAN) [1], would not work in this case. Therefore, we propose to use the Cycle GAN [2] and test the method on various works of famous calligraphers and see if we can generate Chinese Calligraphy which represent different calligraphers.

This project is very interesting since we are using the most advanced technologies nowadays to generate the traditional beauty from what everyone can achieve easily, namely, typed characters in specific fonts, is a combination of technology and culture. We aim to create artistic calligraphy characters.

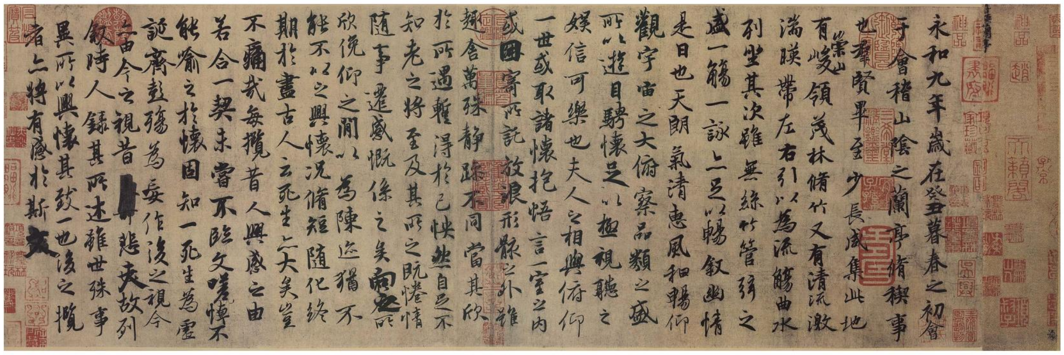

Figure 1: Sample Calligraphy sample Lantin Xu by Wang Xizhi

2 Related Work

In terms of the most relevant topic to our research problem, we found that CalliGan [3] was proposed as an improved model for generating Chinese calligraphy characters from plain style, using encoder-decoder-based image translation network with two supporting branches to control styles and structures. It has 4 image-based losses which are adversarial, pixel-wise, constancy and category. The drawback of this paper is that the font of the input character image is fixed and the performance is still unknown if input fonts are changed to another font. They also did not try other GAN training methods such as WGAN-GP or SN-GAN. Therefore, we will investigate other kinds of GAN training methods further.

Besides, we also surveyed for related projects using CycleGAN [2] which is our intended method. [4] proposed another DenseNet-5 CycleGAN model for generating handwritten styled Chinese characters. They found using DenseNet as part of the CycleGAN generator can improve the generation quality of the original CycleGAN which uses ResNet.

[5] on the other hand, proposed an end-to-end multi-content GAN to take a subset of stylized images of specific categories (such as font glyphs) and predict the whole set of stylistically similar images. They specifically designed the model for the font generation problem to predict the set of letters from A to Z for in-the-wild fonts with a few observed letters. Their model contains GlyphNet and OrnaNet to handle glyph generation and texture transfer.

[6] proposes a style transfer model based on meta learning and CycleGAN [2] to generate fonts automatically, focusing on pop (point of purchase) font style, which aims to stimulate consumption and activate the atmosphere of the store and contains more decorative elements. They added meta-loss computing unit to original CycleGAN to improve the model performance.

The publications above are related work that we found helpful for our research problem. We are planning to dive into further to investigating potential problems and space of improvement for those research.

3 Dataset

In this project, we used the Chinese Calligraphy Styles by Calligraphers Dataset on Kaggle which has 5000 - 7000 jpg images in each subset of 20 famous Chinese calligraphers along with their corresponding calligraphy writing. For this project, we picked four of them and trained their corresponding generation models. The four calligraphers of which we trained on are Zhao-Meng Fu, Yen-Zen Qin, Song-Hui Zong, Liu-Gong Quan. An example would be Fig 2b, which is the sample of calligraphy characters written by Liu-Gong Xuan.

As to the typed fonts for the training, we generated our own pictures of Chinese fonts. The generator style we used is the Chinese Characters Generator from Kaggle, which consists of 16 types of fonts in which we can generate. We then used the code in [7] to generate image files of Chinese typed characters, and the font we used is kaite, which is the fangzheng_jeite.TFF font family in the Chinese Characters Generator from Kaggle. An example would be Fig 2a, which is the sample of typed characters with the kaite font-family.

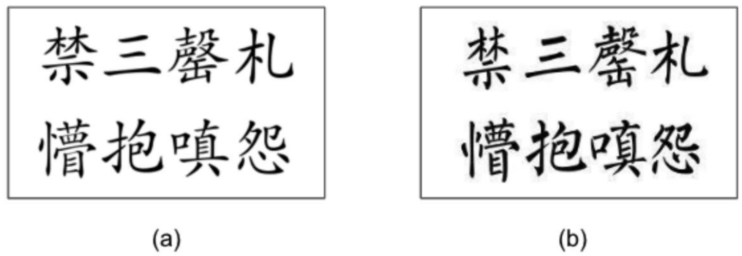

Figure 2: Sample figure of dataset. (a) Sample of typed characters (Kaite). (b) Sample of calligraphy characters written by Liu-Gong Xuan

4 Method

The method we use in this final project is the Cycle GAN [2]. In the milestone, we have already proven that the Cycle GAN is capable of generating Chinese Calligraphy, and thus we decided to use the Cyele GAN for our project and test it on larger datasets and on different calligraphers. The reason we choose to use Cycle GAN is that it is very hard to find paired data of handwriting and calligraphy. In the implementation of the method, we referred to online open-source implementations of the Cycle GAN. The flow chart of Cycle GAN is depicted in figure 3a. Figure 3b is an indication of how we modify the two styles from the original Cycle GAN implementation and generate calligraphy characters. The X here is typed characters, and Y is calligraphy characters written by different calligraphers. We trained two generators G and F to transform between typed characters and calligraphy characters, and we use G as our final generator for results.

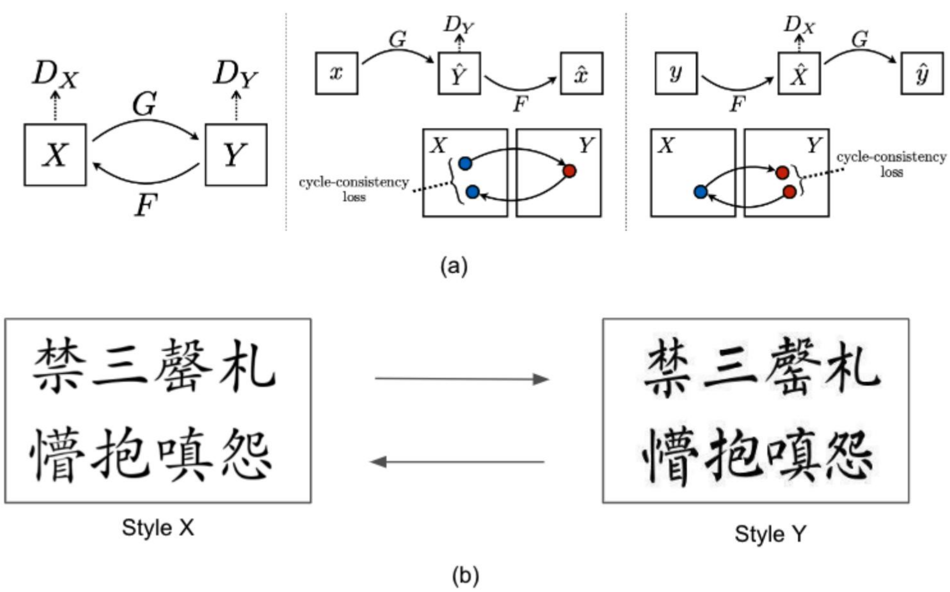

Figure 3: (a) Flow chart of Cycle GAN (b) Indication that in our project, typed characters represent style X, and calligraphy characters represent style Y.

5 Experiments and Results

For the experiment, the input images in style X consist of 400 128*128 typed Chinese character images (written in DFKai-SB, and kaite for short). The images in style Y consist of 400 128*128 images of Chinese characters written by a certain calligrapher. For this project, we trained on 4 different calligraphers, Zhao-Meng Fu, Yen-Zen Qin, Song-Hui Zong, and Liu-Gong Quan, which represents different calligraphy styles for comparison. All of the images are grayscale. The number of CPUs are 4 according to AWS, batch size is set to 4, learning rate is 0.0002, and number of epochs is set to 200.

The results are shown in Fig 4. Fig 4a is the typed version of the first eight Chinese characters of LanTin Xu by Wang Xizhi in the kaite font-family. Our results are presented in Fig 4d, which reads the input in Fig 4a, and generate the corresponding calligraphy type. The corresponding calligraphers are in 4b, and the calligraphy samples are in 4c. By this result, we can see that the results of our generation of our model clearly represents the Chinese letters, and it is clear that the styles represent that of the corresponding calligrapher. Thus, we would claim that our model can successfully transform the Chinese from the original font-family (kaite) style into the corresponding calligrapher style. This is an example of creating traditional artwork from modern technology.

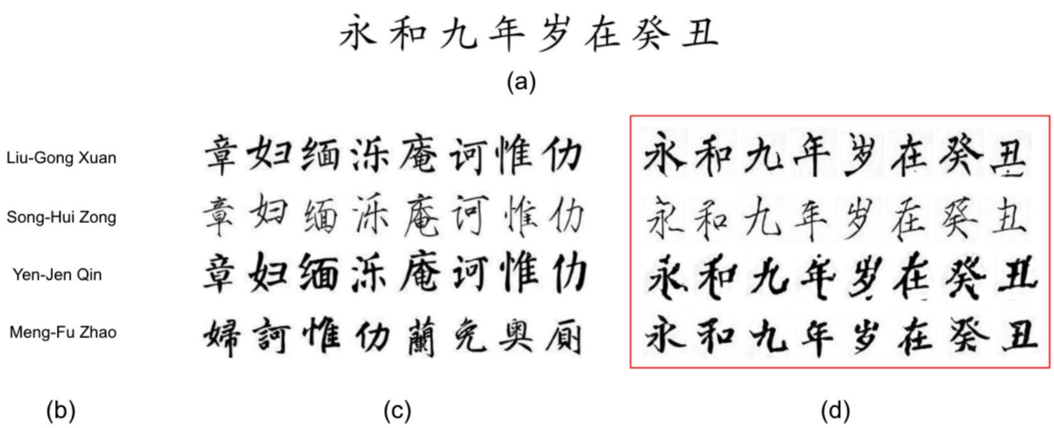

Figure 4: (a) Typed version of the first eight Chinese characters of LanTin Xu by Wang Xizhi in the kaite font-family. (b) Names of the Calligraphers. (c) Sample of written calligraphy by the calligraphers on the left. (d) Generation of calligraphy by our model.

6 Discussion

However, not all characters are generated successfully. Fig 5 demonstrates two successfully generated Chinese characters, and two Chinese characters that are unsuccessful.

We can see that the two successful characters, je and mu, are more complicated and contain more strokes. Thus, the creation is clear and represent the corresponding calligrapher very much. However, zhi and yi, contain very little strokes (zhi only has three strokes, and yi only have one). The formation of the two characters are very simple, and many blanks are in the original typed version.

As we can see in the circled spot in Fig 5c, there is a lot of white pixel in zhi, and so it might confuse the model that there should be something there, thus the generation of calligraphy isn't successful. This phenomenon is more clear in yi, in which some of the generation failed completely, and we cannot even recognize that this is the character yi.

7 Conclusion

In this paper, we successfully demonstrated that we can generate Chinese characters from typed characters. We also proved that the Cycle GAN method can transform styles of even the written language. Although this method has its limitations, most of the letters generated are recognizable and can represent the original character, and thus we can claim that this demonstration is a success. This is surely a creation of traditional beauty with modern technology.

8 Group Member Contributions

Each group member contributed evenly, and would assist others with their parts. Ta-Hsuan Chao mainly created the training and testing datasets, designed the Cycle GAN training algorithm along

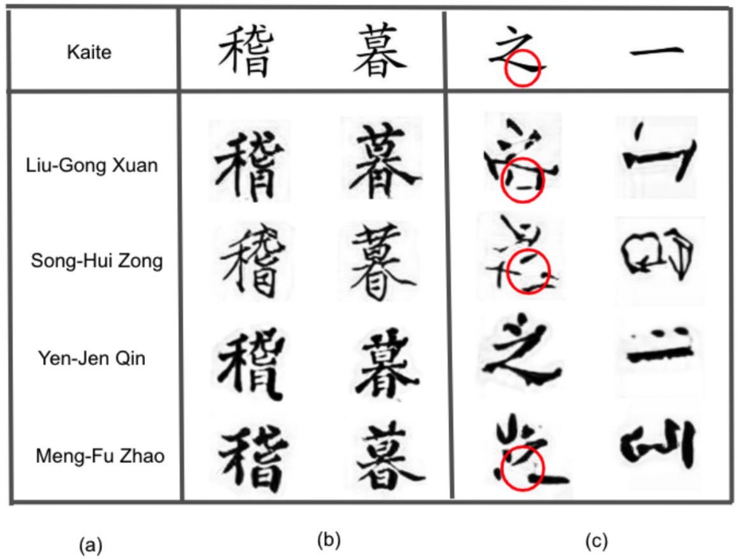

Figure 5: (a) Name of calligraphers. (b) Sample of successful generation (je and mu). (c) Sample of unsuccessful generation (zhi and yi).

with the corresponding inputs and outputs, wrote parts 4, 5, 6 parts of the paper, and created figures for the paper and PowerPoint. Men-Jin Lin mainly did training and inferencing on four calligraphers using Cycle GAN, wrote parts 1, 2, 3 parts of the paper, and created the figures for the paper and PowerPoint.

References

[1] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, 35(1):53–65, 2018.

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

[3] Shan-Jean Wu, Chih-Yuan Yang, and Jane Yung-jen Hsu. Calligan: Style and structure-aware chinese calligraphy character generator. *arXiv preprint arXiv:2005.12500*, 2020.

[4] Bo Chang, Qiong Zhang, Shenyi Pan, and Lili Meng. Generating handwritten chinese characters using cyclegan. *CoRR*, abs/1801.08624, 2018.

[5] Samaneh Azadi, Matthew Fisher, Vladimir G. Kim, Zhaowen Wang, Eli Shechtman, and Trevor Darrell. Multi-content GAN for few-shot font style transfer. *CoRR*, abs/1712.00516, 2017.

[6] Yong Yu. Few shot POP chinese font style transfer using CycleGAN. *Journal of Physics: Conference Series*, 2171(1):012031, jan 2022.

[7] Bo Chang, Qiong Zhang, Shenyi Pan, and Lili Meng. Generating handwritten chinese characters using cyclegan. In *2018 IEEE winter conference on applications of computer vision (WACV)*, pages 199–207. IEEE, 2018.