Chinese Character Style Transfer with conditional GAN

Ho Kei Cheng
Hong Kong University of Science and Technology
hkchengad@connect.ust.hk

Abstract

Image generation and style transfer have became a hot topic in research. Generative Adversarial Networks (GANs) have been used to solve a lot of image generation problems. We here present an attempt at using GANs to perform style transfer by formulating it as an image generation problem on Chinese characters. Unlike most previous works, our method does not focus on a singleton image transform. It learns how to transform an image to a unseen style by looking at font samples from the new style. Thus, no re-training is required to deal with novel fonts.

1. Introduction

The advancement of deep convolutional neural networks (DCNNs) has enabled computers to better understand images at different abstraction levels. They are able to transfer the low level features like storkes and color from an image to another, or even to generate images based on description. Eariler studies on neural style transfer tries to minimize the difference between both the style and content of the source and target images using features from DCNN. [3] [9] [1]

Recently, Generative Adversarial Networks (GANs) have also been used on style transfer and image generation problems and have achieved success. GANs include both a generator and a discriminator. By imposing an additional discriminator as a guidance to the generator, GANs tend to generate more realistic images. GANs include a generator and a discriminator having the objective function

$$\begin{aligned}\mathcal{L}_{GAN}(G,D) =& \mathbb{E}_y[\log(D(y))]+ \\ & \mathbb{E}_{x,z}[\log(1-D(G(x,z)))]\end{aligned}$$

where the generator G tries to minimize this objective while D tries to maximize it. [6]

In this paper, we propose a style transfer GAN that can be used on multiple, unseen styles and content. The network is designed to extract content information and style information separately early on and combine them at a later stage.

We focus on style transfer for Chinese fonts. Traditionally it is difficult to design new Chinese fonts as there are more than 4,000 commonly used Chinese characters and all of them have to be designed manually [8]. Our method aims to automatically generate the entire character set including thousands of characters given a few style references.

2. Related Work

2.1. Image translation

Image-to-image translation learns from paired images and attempts to establish the transformation from one image domain to another. It includes generating color images from grayscale images or converting day scene to night scene. Previous work like conditional GAN (cGAN) [5], pix2pix[2] and cycle-consistent adversarial network (CycleGAN)[12] have shown appealing results in image translation. However, all of them are fixed to learn only one transformation at a time. To perform a new transform, the network must be retrained.

2.2. Font Style Transfer

A number of previous works have also studied the character style transfer process. "From A to Z" perform style transfer on English characters using variational autoencoder (VAE) [10]. A online project zi2zi borrows idea from pix2pix, using GAN with encoder-decoder and the U-net structure in the generator to perform font generation in a latent space created by some fixed font styles. Another work AEGN also uses GAN to generate calligraphy using a standard font directly [4]. A more recent and aligned work is the EMD model, which separate the content and style representation using two distinct encoders [11].

3. Method

Our model consists of two parts: *Generator* and *Discriminator*. The generator will be feed with reference context and reference style, trying to generate the character in the said context with the given style. The disciminator then tries to distinguish generated characters with ground truths.

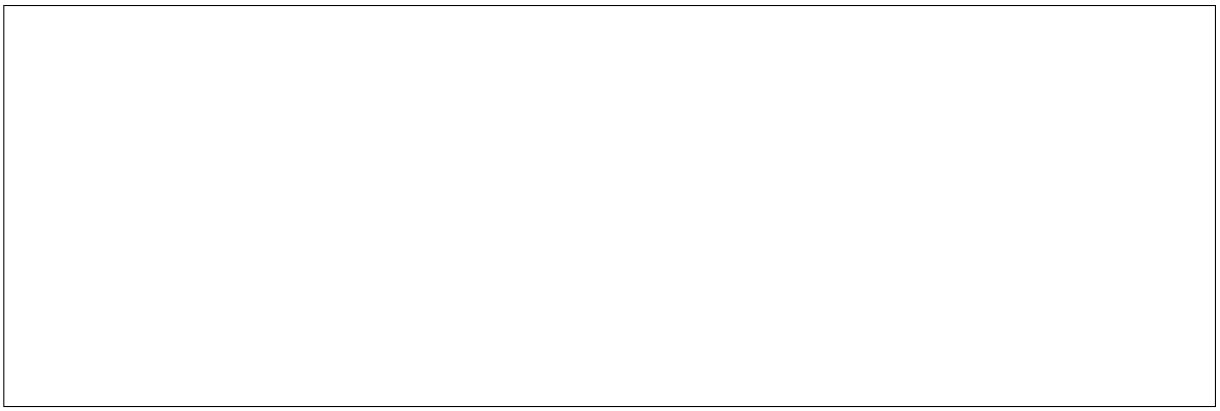

Figure 1. The network structure

3.1. Generator

The generator consists of two encoders and a decoder. The two encoders are *Style Encoder* and *Context Encoder* respectively. All the reference context will be concatenated together and fed into the context encoder. The styles will be fed one by one into the style encoder.

After all the contexts and styles have been encoded, they are concatenated together and passed to the decoder network. The output from the decoder network is the generated result.

3.1.1 Encoder

Both the *Style Encoder* and *Context Encoder* consist of layers of $3x3Convolution - BatchNorm - ReLU$. Max-poolings are used to reduce the dimension of the original $64 * 64$ input to $8 * 8$ encoded output.

3.1.2 Decoder

To enchance the quaility of generated images, transposed convolutions (also known as deconvolution) are used to upsample the encoded context and style reference, as suggested by DCGAN[6]. The decoder network is a transposed version of the encoder, except that transposed convolution with input padding of 2 and output padding of 1 has replaced max-poolings.

There are U-Net[7] like structure which connect the encoder and decoder together with skip-connection to preverse some of the lost information in max-pooling.

3.2. Discriminator

There are two small networks in the discriminator. They are the *Context Discriminator* and *Style Discriminator* respectively. The context discriminator takes the image-to-be tested with the context reference to determine its realness in terms of context. The style discriminator take the style reference instead, and compare realness in terms of style. Their results are averaged to obtain overall discriminator score.

3.3. Losses

We have used the binary cross entropy loss to update the discriminator, guiding it to output 1 for ground truth images and 0 for generated images.

For the generator, both binary cross entropy loss and L1 loss is used as suggested by pix2pix. The binary cross entropy loss guides the generator to output images that the discriminator will give high score, i.e. to fool the discriminator. The L1 loss compare the generator's output and the ground truth image.

4. Result

The network can successfully generate readable characters that have similar low-level features as ground truths. However, there are a few issues:

1. Thins stroke might be missing.
2. The generated image looks "dirty", i.e. has noises.
3. It failed to learn high level features like spacing between strokes.

5. Implementation and data

The model is implementated with PyTorch. It is trained with SGD with learning rate 0.001, momentum of 0.9 and batch size 64 for 400 epochs. The training took 15 hours on a GTX970.

31 fonts are obtained onlined. 23 of them are used in style reference training, 5 of them used as context reference and the rest act as test set. 1702 commonly used Chinese characters are picked to be used in training.

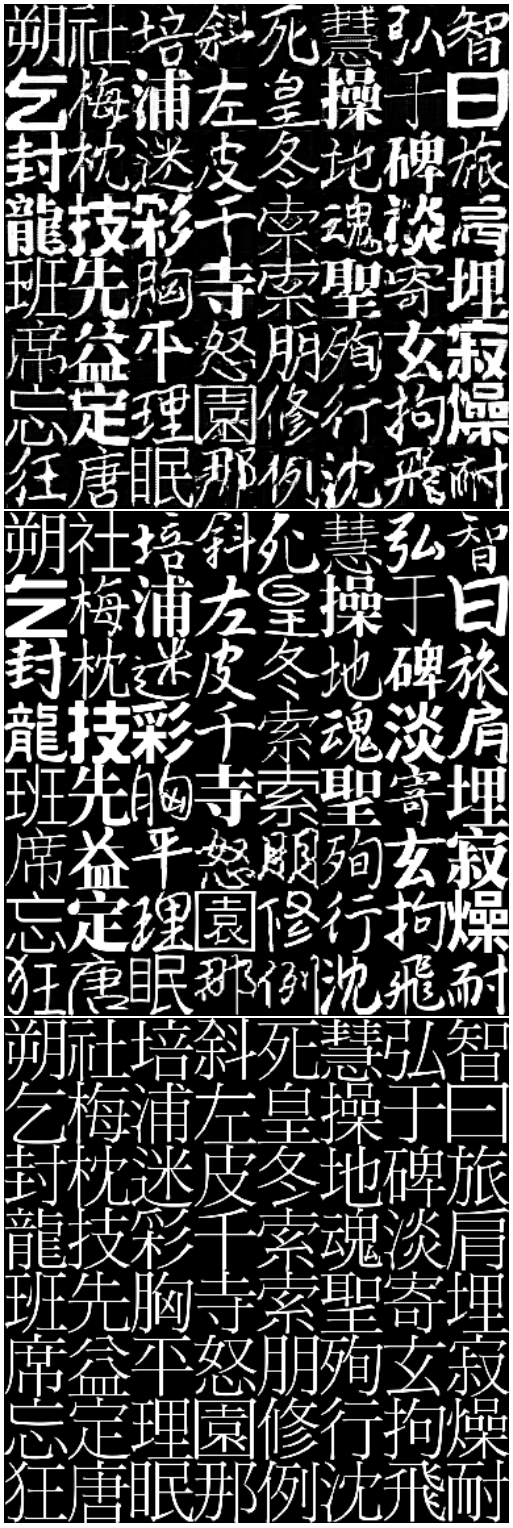

Figure 2. Generating different characters in different fonts in a single batch. Top to bottom: Generated characters, ground truth and one of the context reference

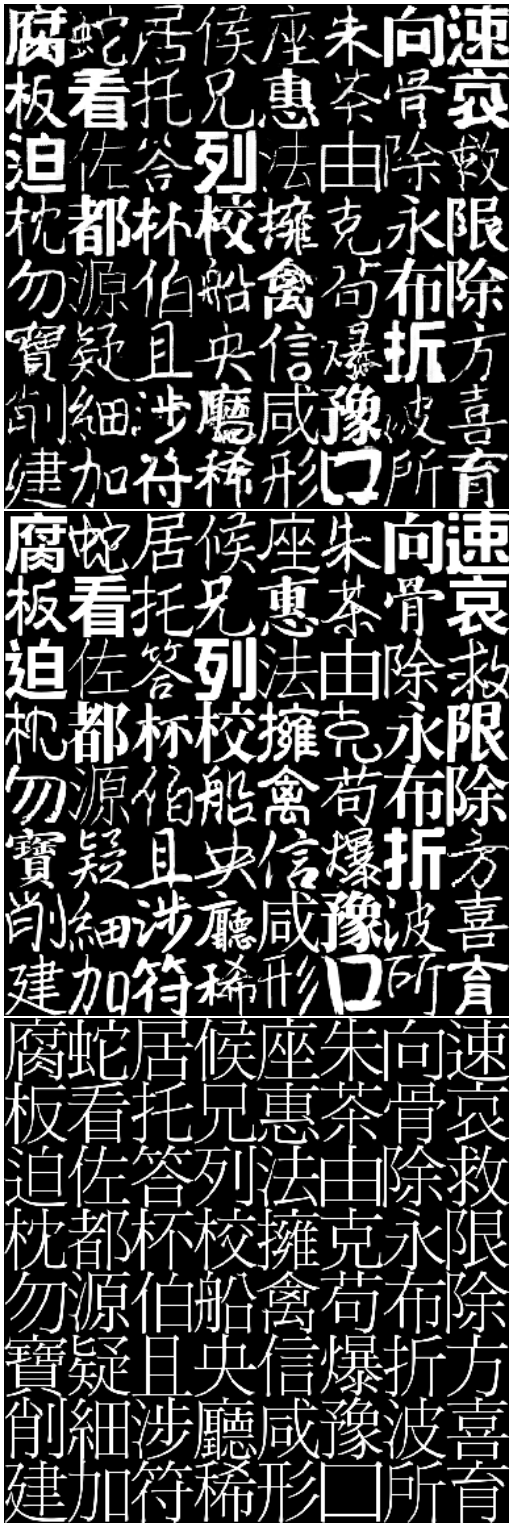

Figure 3. Another result. Top to bottom: Generated characters, ground truth and one of the context reference

Note that different fonts can have significant difference in size and position even when the same font size and origin is used. Preprocessing is done so that the bounding box of each character aligns.

References

[1] L. A. Gatys, A. S. Ecker, and M. Bethge. A neural algorithm of artistic style. *CoRR*, abs/1508.06576, 2015.

[2] P. Isola, J. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. *CoRR*, abs/1611.07004, 2016.

[3] J. Johnson, A. Alahi, and F. Li. Perceptual losses for real-time style transfer and super-resolution. *CoRR*, abs/1603.08155, 2016.

[4] P. Lyu, X. Bai, C. Yao, Z. Zhu, T. Huang, and W. Liu. Auto-encoder guided GAN for chinese calligraphy synthesis. *CoRR*, abs/1706.08789, 2017.

[5] M. Mirza and S. Osindero. Conditional generative adversarial nets. *CoRR*, abs/1411.1784, 2014.

[6] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *CoRR*, abs/1511.06434, 2015.

[7] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015.

[8] C. L. E. Section. Lexical items with eng. explanations for fundamental chin. learning in hk schools. www.edbchinese.hk.

[9] D. Ulyanov, V. Lebedev, A. Vedaldi, and V. S. Lempitsky. Texture networks: Feed-forward synthesis of textures and stylized images. *CoRR*, abs/1603.03417, 2016.

[10] P. Upchurch, N. Snavely, and K. Bala. From A to Z: supervised transfer of style and content using deep neural network generators. *CoRR*, abs/1603.02003, 2016.

[11] Y. Zhang, W. Cai, and Y. Zhang. Separating style and content for generalized style transfer. *CoRR*, abs/1711.06454, 2017.

[12] J. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *CoRR*, abs/1703.10593, 2017.