electronics

MDPI

Article

Visual Attention Adversarial Networks for Chinese Font Translation

Te Li [1,2], Fang Yang [1,2,*] and Yao Song [1,2]

[1] Computer Science and Technology, School of Cyberspace Security and Computer, Hebei University, Baoding 071000, China
[2] Hebei Machine Vision Engineering Reasearch Center, Hebei University, Baoding 071000, China
* Correspondence: yangfang@hbu.edu.cn

Abstract: Currently, many Chinese font translation models adopt the method of dividing character components to improve the quality of generated font images. However, character components require a large amount of manual annotation to decompose characters and determine the composition of each character as input for training. In this paper, we establish a Chinese font translation model based on generative adversarial network without decomposition. First, we improve the method of image enhancement for Chinese character images. It helps the model learning structure information of Chinese character strokes to generate font images with complete and accurate strokes. Second, we propose a visual attention adversarial network. By using visual attention block, the network catches global and local features for constructing details of characters. Experiments demonstrate our method generates high-quality Chinese character images with great style diversity including calligraphy characters.

Keywords: Chinese font generation; generative adversarial network; style translation; visual attention

check for updates

Citation: Li, T.; Yang, F.; Song, Y. Visual Attention Adversarial Networks for Chinese Font Translation. *Electronics* **2023**, *12*, 1388. https://doi.org/10.3390/electronics12061388

Academic Editor: Silvia Liberata Ullo

Received: 18 February 2023
Revised: 10 March 2023
Accepted: 12 March 2023
Published: 14 March 2023

1. Introduction

Chinese characters, with a total of over 90,000, is one of the oldest and most complex writing systems in the world. The various writing styles of Chinese characters have formed a unique traditional art in China—Chinese calligraphy. In the past, due to the large number and complex structure of Chinese characters, establishing a complete style font library is an extremely difficult task. Using convolutional neural networks (CNNs) to establish a style translation model is an effective method to simplify the font generation task.

With the proposal of a generative adversarial network (GAN) [1], a lot of image style translation models based on GAN have been established. In 2017, Isola et al. [2] introduced the image-to-image translation model pix2pix, which trains by paired input images and target images. Based on pix2pix, Rewrite [3] and Zi2zi [4] were introduced for Chinese font translation. In 2017, the unpaired image-to-image translation model CycleGAN [5] was introduced, which reduced the requirement of paired training in pix2pix. In 2018, a non-paired font style translation model [6] based on CycleGAN was established. Subsequently, Chinese font style translation models based on adversarial generative networks were also divided into two categories: paired and unpaired training.

However, due to the complexity of the Chinese character structure, the generated Chinese character images usually have incomplete strokes, blurred strokes and other phenomena.

Decomposing Chinese characters into components is an effective method to generate high-quality character images. Huang et al. introduced RD-GAN [7] which adopted 576 radicals and adds BLSTM on the network for encoding and decoding features. Song et al. [8] decomposed components of each Chinese character and specifies 371 components. CalliGAN proposed by Wu et al. [9] built 517 components, using a recurrent neural network (RNN) to encoder these components and generate Chinese calligraphy characters. Zhang et al. [10] labelled the stroke order and position of Chinese characters as the input to generate and

recognize Chinese characters using RNN. Lian et al. [11] manually marked stroke points and orders of 27,533 characters in the Kaiti style as the input, and set 339 fine-grained categories of strokes as a reference. Tang et al. [12] added a monotone attention layer to the model based on the stroke points and orders using bidirectional RNN. Although these models can improve the quality of generated character images, the method of decomposing characters into components is a complex task and greatly affected by prior knowledge.

In this paper, we propose a Chinese font style translation model without character decomposition. Using better image pre-processing and attention mechanisms to improve the networks; the model can better learn the structure features and stroke details of character images. The main contributions of our method are summarized as follows:

- We propose a visual attention adversarial network based on generative adversarial networks, the network catches global and local features for constructing details of character images. Our method achieves good results in Chinese style translation tasks.
- We improved the image enhancement method, which encourages the network to learn the structural information of Chinese character strokes, then makes it more suitable for font image generation.

2. Related Work

Font image translation is essentially a task of image-to-image translation and an attention mechanism is effective in assisting in the production of higher-quality translations. We discuss these related work in the following.

2.1. Image-to-Image Translation

In 2017, pix2pix was proposed. It is an image-to-image translation model based on CGAN [13] which made the output controllable by adding conditional information. Pix2pix used Unet [14] as the backbone of generative network, and proposed PatchGAN as the backbone of its discriminator. With an input image, the model generates a corresponding output image. This has been used for tasks such as image colourization, image super-resolution, and style transfer. CycleGAN improves the quality of generated images by using cyclic training. It not only transfers the style from A to B, but also from B to A. Therefore, CycleGAN does not need paired training data unlike pix2pix. CycleGAN style transfer has a wide range of applications in art, fashion, interior design, advertising, and many other fields.

Compared with other image-to-image translation tasks, Chinese font style translation is more difficult. Chinese font has a large number of characters and each character has a different structure and strokes, causing it difficult to extract the features. In addition to performing style translation, the translation model must ensure the preservation of the original content features of Chinese characters in each input image. By adding an attention mechanism, our proposed method improves the ability of obtaining structural information from font images.

2.2. Attention Mechanism

In 2014, Google DeepMind published "Recurrent Models of Visual Attention" [15], the attention mechanism has been widely used since then. In 2017, the Google machine translation team published "Attention is All You Need" [16] which achieved good results. However, these methods were originally designed for natural language processing which means they are suitable for one-dimensional data. Visual attention methods on a two-dimensional scale will perform better in image tasks. There are representative methods such as the gMLP proposed by Liu et al. [17] and the VAN proposed by Guo et al. [18].

Compared with similar size vision transformers (ViTs) and convolutional neural networks (CNNs), a visual attention network (VAN) is better in various tasks. There is a novel linear attention named large kernel attention (LKA) in VAN. Apart from adopting a self-attention mechanism [19,20], another well-known method to build a relationship between different parts is using large kernel convolution [21,22] which needs numerous

parameters and huge computational cost. Therefore, LKA was proposed to overcome these shortcomings. It decomposes a large kernel convolution into three components: a spatial local convolution (depth-wise convolution), a spatial long-range convolution (depth-wise dilation convolution), and a channel convolution (1×1 convolution). LKA combines the advantages of large kernel convolution and self-attention.

3. Methods

3.1. Overview

We proposed an end-to-end adversarial network based on pix2pix, the paired training model structure is shown in Figure 1. Before training on the network, we use image enhancement to improve the robustness of the model. As for a generative network, we proposed an encoder–decoder network based on linknet [23] to replace Unet as the backbone. Each encoder layer has a following attention block based on VAN and these attention blocks comprise the visual attention module. The output of the generator and corresponding target images are sent to a discriminator to discriminate if the image is real or fake. In the following sections, we introduce the image enhancement method first, then we discuss the generative and discriminative networks. The loss function is discussed in the last section.

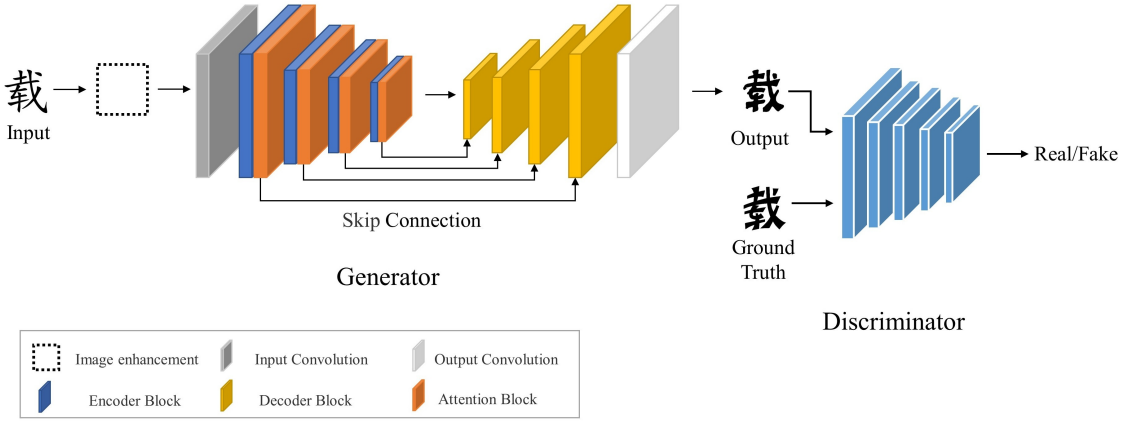

Figure 1. Overview of the proposed method. The generator translates input images of font Kaiti to images of target font. The discriminator differentiates the output of the generator and ground truth, the black dotted bordered rectangle is the image enhancement. The blue block is the encoder block, the yellow block is the decoder block, the orange block is the attention block.

3.2. Image Enhancement

Image enhancement is a process of improving the information content of an image. There are many methods of image enhancement for image translation task such as rotation, scaling, cropping, etc. As for Chinese character images, in order to obtain structural information, we improve upon the method of random cropping as image enhancement.

The steps of random cropping are as follows. First, the offset ratio values Rx and Ry are obtained from random number in the range of $-MR$ to MR. MR means the maximum ratio of the offset, in case the value is too large and valid information is lost, as shown in Figure 2A.

$$Rx = random(-MR, MR), \tag{1}$$

$$Ry = random(-MR, MR). \tag{2}$$

Second, we obtain four values which are the left, upper, right, and lower pixel coordinate (xl, yu, xr, yl) according to Rx, Ry and image size W, H.

$$xl = -Rx * W, \tag{3}$$

$$xr = (1 - Rx) * W, \tag{4}$$

$$yu = -Ry * H, \tag{5}$$

$$yl = (1 - Ry) * H. \tag{6}$$

At last, according to the pixel coordinates, we obtain the cropped image

$$Image = Crop(xl, yu, xr, yl). \tag{7}$$

The parts exceeding the original image are automatically filled with the background colour, as shown in Figure 2B.

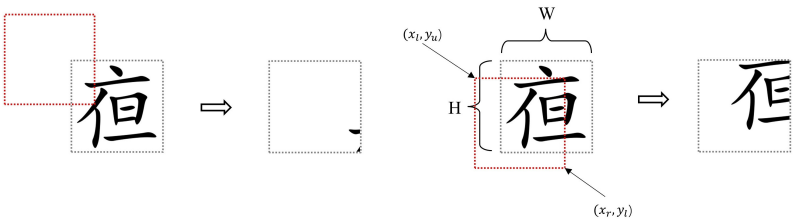

Figure 2. (**A**) The cropped image which lost the most valid information. (**B**) The method of random cropping with pixel coordinates.

We apply the same operations of random cropping at the input image and its paired target image while training. Each random cropping of a Chinese font image results in a different outcome, which makes the training process much more difficult. Therefore, we proposed a parameter named the image pre-processing percentage (IPP). IPP means the probability of each image to be processed. In this way, the model will receive some original images as the input. It can balance the model's rate of convergence and its generalization ability.

3.3. Generative Network

In order to generate Chinese font images of better quality, we selected a backbone based on linknet [23] instead of Unet. Linknet, initially applied in the field of semantic segmentation, uses ResNet18 as its encoder and incorporates skip connections between the encoder and decoder parts to retain information lost during encoding. To catch more global and local features, we added a visual attention module in the encoder and eliminate operations of downsampling before encoding and upsampling after decoding.

The architecture of the generative network is shown in Table 1. We load the input as a greyscale image so the input channel is 1. The input and output convolution blocks both use ReflectionPad of size one. The encoder–decoder has four layers. All the encoder layers use the same convolution kernel size 3×3, and an activation function ReLu.

Table 1. The architecture of the generative network.

Layer	Encoder	Decoder
Input	$256 \times 256 \times 1$	$256 \times 256 \times 1$
Inc/Outc	$256 \times 256 \times 32$	$256 \times 256 \times 32$
$L1$	$128 \times 128 \times 64$	$128 \times 128 \times 64$
$L2$	$64 \times 64 \times 128$	$64 \times 64 \times 128$
$L3$	$32 \times 32 \times 256$	$32 \times 32 \times 256$
$L4$	$16 \times 16 \times 512$	$16 \times 16 \times 512$

A stage of the encoder block is shown in Figure 3. It has two resblocks, each resblock consists of two convolution layers, two batch normalization layers and a skip-connection. The second convolution layer of the first block has a stride size of two for downsampling, while the others have a stride size of one.

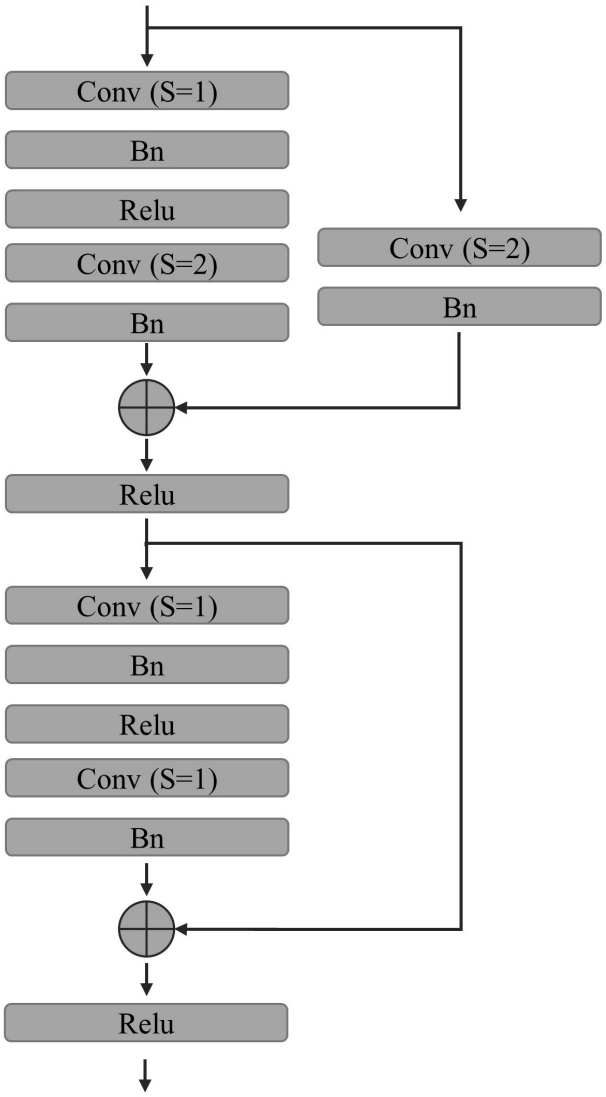

Figure 3. A stage of the encoder block. S denotes stride.

A stage of the attention block is shown in Figure 4. There are some convolution layers with kernel size 1×1 to capture the relationship in channel dimensions. The linear attention, named LKA, consists of a 5×5 depth-wise convolution, a 5×5 depth-wise dilation convolution with a dilation rate of three, and a point-wise convolution. It reduces the computation and parameter count by decomposing a 13×13 large kernel convolution. The block uses batch normalization before convolution and an activation function ReLu.

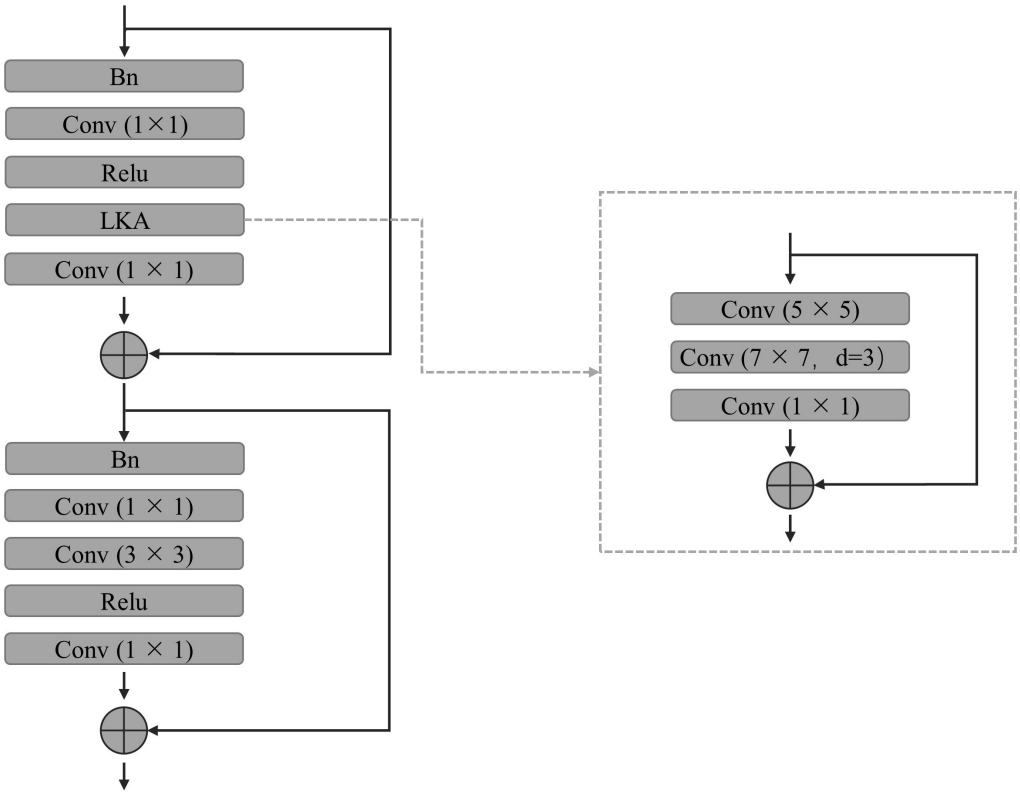

Figure 4. A stage of attention block.

3.4. Discriminative Network

Low frequency is captured by $L1$ loss, so the discriminator focuses on high frequencies. The discriminative network is trained to translate a full image into N × N patches and classify each patch as real or fake. It improves the coherent and visually consistency of the generated image.

To address the issue of generator and discriminator training imbalances, we incorporated spectral normalization [24] to enhance the stability of training. Spectral normalization not only ensures the Lipschitz constraint condition but also avoids matrix structure destruction in WGAN [25]. The Lipschitz constraint is as follows:

$$\frac{\| f(x) - f(x') \| 2}{\| x - x' \| 2} \leq M, \tag{8}$$

where M is a constant, this constraint limits the maximum function gradient to be less than or equal to M, which reduces the convergence rate of the discriminator.

3.5. Loss Function

The loss function of our model consists of an adversarial loss and $L1$ loss. $L1$ loss measures the absolute difference between the true and predicted pixel values. It is used to minimize blurring of the generated character image. $L1$ loss is shown as:

$$\mathcal{L}_{L1} = \mathbb{E}_{x,y,\hat{y}}[\| y - G(x,\hat{y}) \|_1]. \tag{9}$$

The adversarial loss is used to obtain information of high-frequency. It is used to make the generated image look as real as possible. The adversarial loss is shown as:

$$\mathcal{L}_{GAN}(G,D) = \mathbb{E}_{x,y}[logD(x,y)] + \mathbb{E}_{x,\hat{y}}[log(1 - D(x,G(x,\hat{y}))]. \tag{10}$$

The goal of adversarial loss is to minimize G and maximize D. The loss function of the proposed methods is shown as:

$$\mathcal{L} = arg \min_{G} \max_{D} \mathcal{L}_{GAN}(G, D) + \lambda \mathcal{L}_{L1}(G)], \tag{11}$$

where λ is 100.

4. Experiments

4.1. Dataset

As there is no unified training data for Chinese font style translation tasks, our test fonts are all downloaded from font websites. We selected 6763 Chinese characters (GB2312 official standard) as the target font dataset and 20,901 Chinese characters (GB18130 official standard) from font Kaiti as the input reference dataset. For training and validation, we selected the input reference Chinese characters which had the same code with the characters of target. For testing or use, the entire input reference dataset is available. The image size was 256×256.

4.2. Experiment Setup

We adopted normal initialization with a standard deviation of 0.02 and used Adam as the optimizer with $\beta_1 = 0.5$ and $\beta_2 = 0.999$. The epoch was set to 200, batch size was 1, and the initial learning rate for the generator network was 2×10^{-3} from epoch 1 to 100, gradually decreasing to zero from epoch 101 to 200. For the parameters of image enhancement, we set the maximum ratio of offset $MR = 0.5$ and image pre-processing percentage $IPP = 80\%$. The learning rate for the discriminator network was fixed at 2×10^{-7}. We implemented the proposed model using PyTorch and trained it on an NVIDIA GeForce RTX 3080 GPU.

In this paper, we use the $L1$ loss, structural similarity index (SSIM) and root-mean-squared error (RMSE) to measure the similarity between the ground truth and generated images.

4.3. Parameter Comparison of Image Enhancement

In order to determine the experimental parameters and enhance the effectiveness of image enhancement, we conducted 11 sets of experiments on IPP using Unet and Linknet as generators with increments of 10% from 0 to 100% using font SST. We also conducted another experiment using font HYDYTJ to prove stability of the IPP on different fonts.

The results are shown in Table 2. The bold number indicates the best. According to the experimental results, 80% is the optimal value of IPP for one or both fonts SST and HYDYTJ, as well as for models Linknet and Unet. This proves that using the best parameter of image enhancement has good applicability across different target styles of Chinese font images and different models for Chinese font generation. Therefore, the proposed method in this paper uses image enhancement with 80% set as the IPP.

Table 2. Evaluations of image enhancement with different IPPs.

IPP	Linknet SST *L*1 Loss	Linknet SST SSIM	Linknet SST RMSE	Unet SST *L*1 Loss	Unet SST SSIM	Unet SST RMSE	Unet HYDYTJ *L*1 Loss	Unet HYDYTJ SSIM	Unet HYDYTJ RMSE
0%	0.1624	0.6457	0.3881	0.1742	0.6184	0.3980	0.1421	0.6524	0.3623
10%	0.1591	0.6529	0.3840	0.1721	0.6260	0.3962	0.1383	0.6624	0.3578
20%	0.1611	0.6499	0.3863	0.1693	0.6324	0.3932	0.1369	0.6684	0.3575
30%	0.1604	0.6517	0.3856	0.1691	0.6342	0.3937	0.1361	0.6666	0.3550
40%	0.1651	0.6465	0.3912	0.1691	0.6353	0.3939	0.1348	0.6698	0.3535
50%	0.1622	0.6496	0.3874	0.1678	0.6385	0.3929	0.1318	0.6754	0.3500
60%	0.1585	0.6557	0.3821	0.1670	0.6410	0.3027	0.1302	0.6782	0.3482
70%	0.1589	0.6571	0.3822	0.1649	0.6460	0.3970	0.1290	0.6812	0.3468
80%	**0.1561**	0.6638	**0.3792**	**0.1568**	**0.6655**	**0.3806**	**0.1281**	**0.6830**	**0.3456**
90%	0.1581	**0.6645**	0.3813	0.1602	0.6559	0.3851	0.1298	0.6811	0.3480
100%	0.1771	0.6459	0.4048	0.1720	0.6566	0.4008	0.1389	0.6682	0.3600

4.4. Ablation Study

In order to demonstrate the effectiveness of the modifications in the model on improving results of Chinese character images, we conducted ablation experiments on these modified parts, including image enhancement (IE), an attention block (attn) and the change in generator backbone. We used font HYDYTJ as the target style.

The results of generated images are shown in Figure 5. The generated Chinese character images of the proposed method without image enhancement have some problems with stroke structure. Some strokes are missing and some strokes have abnormal deformations. The fine details of the edges of the character images are processed well. The results of the proposed method without the attention block have complete stroke structures, but some details are not well processed. Unet and Resnet are backbones of pix2pix and the proposed method replaces the backbone network with linknet. Results of models based on Unet and Resnet have problems with stroke structure and edge detail processing. The proposed method delivers the best results for both stroke structure and fine details.

Quantitative evaluations are shown in Table 3. The bold number indicates the best. According to the evaluation results, Resnet performs the worst on the three loss functions. Except for the proposed method, all other methods on the ablation study show a worsening of the loss.

The evaluation results and a comparison of the generated images above indicate that each component we modified is effective.

Table 3. Evaluations of the ablation study.

Loss	*L*1 Loss	RMSE	SSIM
Unet	0.1318	0.3500	0.6754
Resnet	0.1391	0.3617	0.6668
Proposed *w/o* IE	0.1333	0.3532	0.6720
Proposed *w/o* attn	0.1285	0.3469	0.6834
Proposed	**0.1231**	**0.3383**	**0.6920**

Figure 5. The ablation experiment of our model. Red rectangles mark some images with incomplete strokes or fuzzy details.

4.5. Comparison with Existing Methods

In this part, we compare our method with pix2pix and zi2zi for Chinese font generation. Chinese font style generation is a complex task because of the large number of characters. Therefore, models using paired data for training, such as pix2pix and zi2zi, generate the character images well with structures and details. All the models as well as ours trained

with the same input and target dataset with corresponding configurations. We selected two fonts, HYDYTJ and HYXJTJ, as the target styles and font Kaiti as the source dataset for generation.

4.5.1. Qualitative Evaluation

Pix2pix is a classic image style transfer model. However, its performance on Chinese font image generation is not satisfactory. Each dataset of Chinese font contains thousands of different characters. Although the input and target images are one-to-one correspondence and have the same content information of characters, it is still too difficult for pix2pix to preserve the complete content information of Chinese characters, which makes it a challenging task for pix2pix to generate high-quality results. According to the generated images of Chinese characters, the style and content of the generated characters are easy to be distinguished, but the strokes of characters are incomplete and image details, such as edges, are very blurred.

As a professional model in font style translation, zi2zi generate Chinese font images with well-defined edges and distinct style. Zi2zi utilizes category embedding to train on multiple fonts. By expanding the dataset, it can better capture and distinguish content information. However, zi2zi still uses Unet which is the original generator architecture of pix2pix so the generated images lose many detailed features, resulting in unsatisfactory results.

Comparatively, by improving the generator, discriminator and image enhancement, our model has the best image restoration effect in terms of style among them. Our model also preserves complete content information of the Chinese characters. The results are shown in Figure 6.

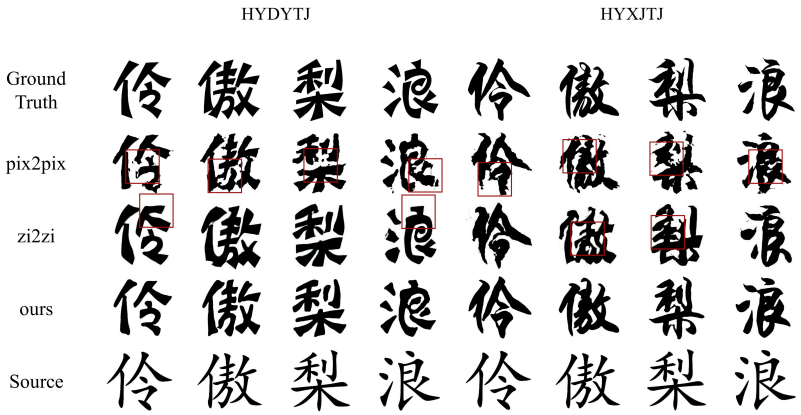

Figure 6. The ablation experiment of our model. Pix2pix generated images with blurred edges and zi2zi with deformed strokes as marked by red rectangles.

4.5.2. Quantitative Evaluation

Evaluation results are shown in Table 4. Bold numbers represent the best results. It is different from our previous human perception of the generated images, compared to zi2zi, the evaluation results of pix2pix are better. For Chinese font generation tasks, having a complete stroke structure of the character and accurately processed details are more important than overall similarity. Therefore, pix2pix tends to perform better in terms of the evaluation metrics. Our evaluation results were significantly better compared to the other models.

Table 4. Quantitative comparison of different methods.

Model	HYDYTJ			HYXJTJ		
	***L*1 Loss**	**SSIM**	**RMSE**	***L*1 Loss**	**SSIM**	**RMSE**
pix2pix	0.1421	0.6524	0.3623	0.1356	0.6894	0.3595
zi2zi	0.1719	0.6183	0.3979	0.1664	0.6481	0.3948
ours	**0.1231**	**0.6920**	**0.3383**	**0.1322**	**0.6931**	**0.3532**

4.6. Model Analysis

We evaluated the improvement of generated Chinese character images' quality with the style of font HYDYTJ while training as the epoch increased using RMSE, *L*1 loss and SSIM. The RMSE, *L*1 loss and SSIM of the validation set for each epoch is shown in Figure 7. All loss stabilize around the 200th epoch. Therefore, we trained the model for 200 epochs. For fonts with greater style variations, increasing the epoch appropriately can improve the quality of the generated images.

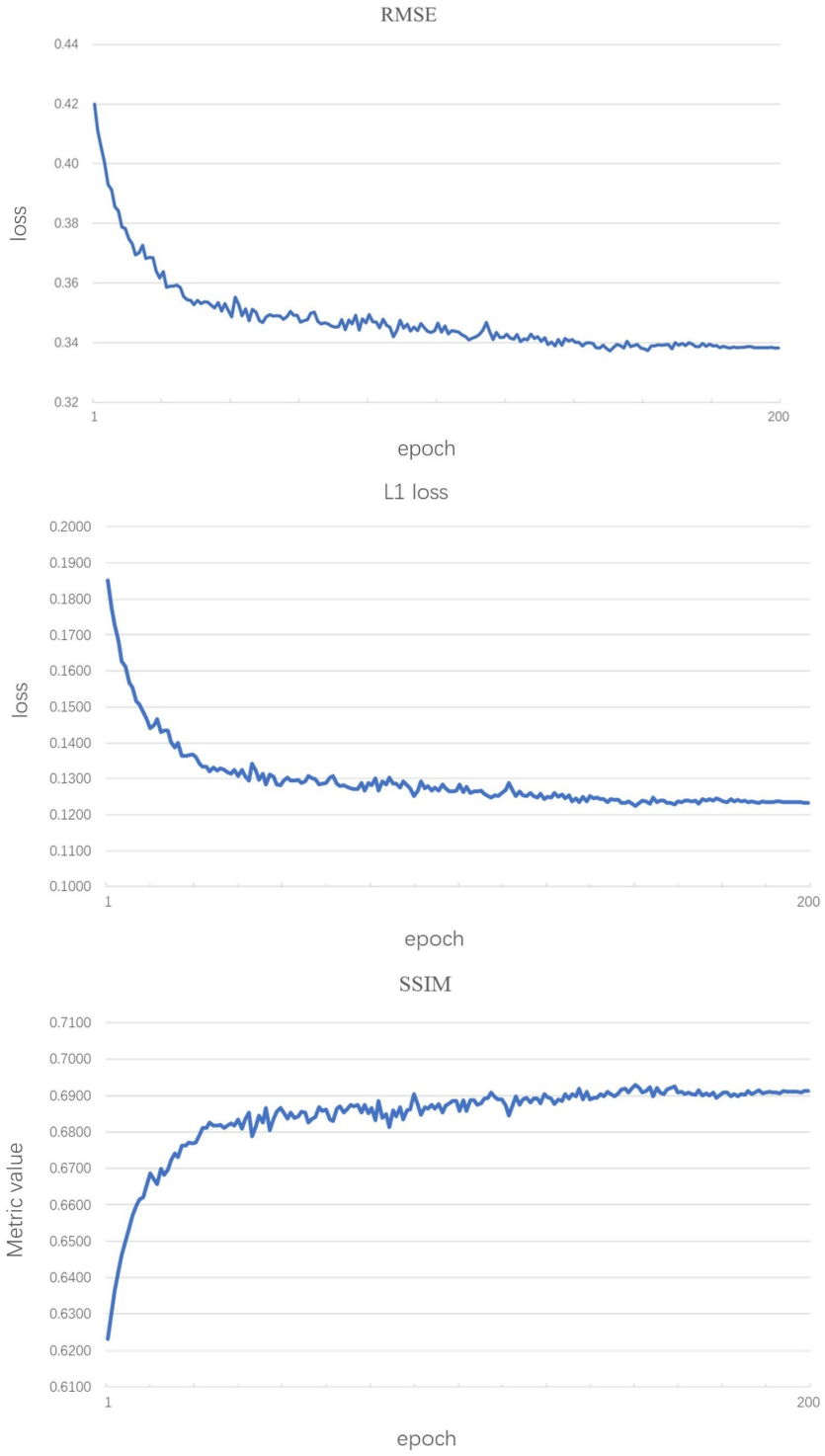

Figure 7. The curve of the RMSE, *L*1 loss and SSIM as the epoch increased.

Our model also produced high-quality outputs in Chinese calligraphy style translation. As shown in Figure 8, the proposed model is trained on several font styles with significant differences. It performed well in restoring the details of the calligraphy font.

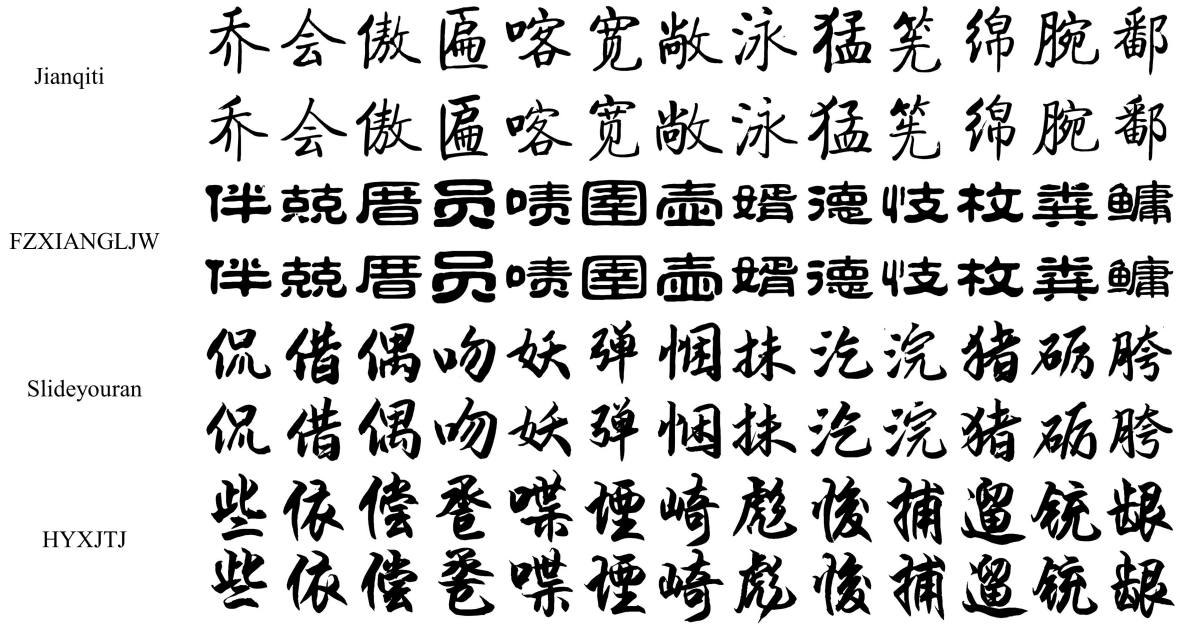

Figure 8. Comparisons of generated images with different calligraphy font styles. The first row shows the generated images, and the second row shows the ground truth.

We also conducted experiments on Chinese calligraphy with style between the script style and cursive script style, as shown in Figure 9. The target style was font YZGCZHZ. Cursive script style has characters executed swiftly and with flowing strokes. The strokes are often abbreviated and simplified, with many characters having fewer visible strokes than their printed or formal counterparts. The resulting characters can be difficult to read for those unfamiliar with the style. Therefore, translation tasks from the source font Kaiti to target font with the cursive script style cannot be performed using our model. As shown in Figure 10, the difference between the two font styles is so significant that it is hard to catch both style and content features.

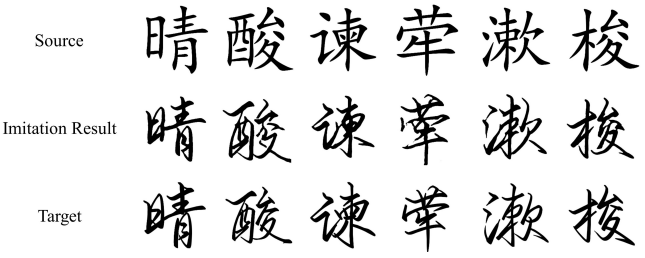

Figure 9. Generated result of Chinese calligraphy images with style between the script style and cursive script style.

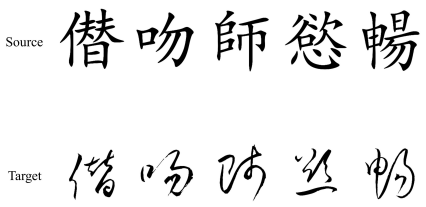

Figure 10. Chinese calligraphy images with cursive script style and the their corresponding source images.

5. Discussion

In this paper, we propose a method of image enhancement and experimentally validate it. It helps the network to reduce the image noise in generated images and makes the stroke structure of character images complete and accurate. Our experiments demonstrated that this method has good performance on images of different style fonts and various network architectures so it has a wide range of applications. However, this method aims to enhance the content and structural information of Chinese character images and it is suitable for situations where the number of input character images is small. For models trained on hundreds of fonts, the effect may not be significant. Future work might use this method for the style translation of some historical Chinese calligraphy images, which only have a small number of images available.

In addition, we improved the generative network. The generative network catches global and local features for constructing details of character images. Our generator network is portable and can be added to other Chinese style translation tasks, such as multi-style training, if necessary.

In this paper, all the Chinese fonts selected have complete strokes, including regular and running scripts. Although the strokes are complete, the stylistic differences between the different fonts are significant. Future work will involve generating Chinese character images with a cursive script style. In this paper, the source font selected was Kaiti with regular script. To generate Chinese character images with cursive script, future work needs to modify both the source and target images.

6. Conclusions

In this paper, we proposed a model of visual attention adversarial networks for Chinese font style translation. We improve upon the method of image enhancement for Chinese characters for the learning structural information of Chinese character strokes. We also enhanced the generative network by modifying the backbone with reference to linknet. To catch global and local features for constructing character details, we added the visual attention block. We conducted experiments on Chinese font datasets with significant style differences. Our model demonstrated an impressive performance in preserving content information while also performing style translation. We conducted an ablation study and a comparative experiment with the existing methods. Experiments demonstrate our method generates Chinese characters of high-quality and great style diversity. Future work includes the restoration of Chinese characters in ancient books and the style translation of ancient Chinese chirography.

Author Contributions: Writing—original draft preparation, T.L.; writing—review and editing, F.Y.; supervision, Y.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported and funded by the Science and Technology Project of Hebei Education Department (No. ZD2019131).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Nets. In Proceedings of the Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014.
2. Isola, P.; Zhu, J.-Y.; Zhou, T.; Efros, A.A. Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1125–1134.
3. Tian, Y. Rewrite: Neural Style Transfer for Chinese Fonts. Available online: https://github.com/kaonashi-tyc/Rewrite/(accessed on 2 June 2022).

4. Tian, Y. zi2zi: Master Chinese Calligraphy with Conditional Adversarial Networks. Available online: https://github.com/kaonashi-tyc/zi2zi/ (accessed on 13 October 2022).
5. Zhu, J.-Y.; Park, T.; Isola, P.; Efros, A.A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2223–2232.
6. Chang, B.; Zhang, Q.; Pan, S.; Meng, L. Generating handwritten chinese characters using cyclegan. In Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Tahoe, NV, USA, 12–15 March 2018; pp. 199–207
7. Huang, Y.; He, M.; Jin, L.; Wang, Y. Rd-gan: Few/zero-shot chinese character style transfer via radical decomposition and rendering. In Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; pp. 156–172.
8. Park, S.; Chun, S.; Cha, J.; Lee, B.; Shim, H. Few-shot font generation with localized style representations and factorization. In Proceedings of the AAAI Conference on Artificial Intelligence, Online, 2–9 February 2021; pp. 2393–2402.
9. Wu, S.-J.; Yang, C.-Y.; Hsu, J.Y. Calligan: Style and structure-aware chinese calligraphy character generator. *arXiv* **2020**, arXiv:2005.12500.
10. Zhang, X.-Y.; Yin, F.; Zhang, Y.-M.; Liu, C.-L.; Bengio, Y. Drawing and recognizing chinese characters with recurrent neural network. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 849–862. [CrossRef] [PubMed]
11. Lian, Z.; Zhao, B.; Chen, X.; Xiao, J. EasyFont: A style learning-based system to easily build your large-scale handwriting fonts. *ACM Trans. Graph. (TOG)* **2018**, *38*, 1–18. [CrossRef]
12. Tang, S.; Xia, Z.; Lian, Z.; Tang, Y.; Xiao, J. FontRNN: Generating Large-scale Chinese Fonts via Recurrent Neural Network. *Comput. Graph. Forum* **2019**, *38*, 567–577. [CrossRef]
13. Mirza, M.; Osindero, S. Conditional generative adversarial nets. *arXiv* **2014**, arXiv:1411.1784 .
14. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015: 18th International Conference, Munich, Germany, 5–9 October 2015; pp. 234–241.
15. Mnih, V.; Heess, N.; Graves, A. Recurrent models of visual attention. In Proceedings of the Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; Volume 27.
16. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; Volume 30.
17. Liu, H.; Dai, Z.; So, D.; Le, Q.V. Pay attention to mlps. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 9204–9215.
18. Guo, M.-H.; Lu, C.-Z.; Liu, Z.-N.; Cheng, M.-M.; Hu, S.-M. Visual attention network. *arXiv* **2022**, arXiv:2202.09741.
19. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
20. Zhang, H.; Goodfellow, I.; Metaxas, D.; Odena, A. Self-attention generative adversarial networks. In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019; pp.7354–7363.
21. Woo, S.; Park, J.; Lee, J.-Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
22. Hu, J.; Shen, L.; Albanie, S.; Sun, G.; Vedaldi, A. Gather-excite: Exploiting feature context in convolutional neural networks. In Proceedings of the Neural Information Processing Systems, Montreal, QC, Canada, 3–8 December 2018; Volume 31.
23. Chaurasia, A.; Culurciello, E. Linknet: Exploiting encoder representations for efficient semantic segmentation. In Proceedings of the 2017 IEEE Visual Communications and Image Processing (VCIP), St. Petersburg, FL, USA, 10–13 December 2017; pp. 1–4.
24. Miyato, T.; Kataoka, T.; Koyama, M.; Yoshida, Y. Spectral normalization for generative adversarial networks. *arXiv* **2018**, arXiv:1802.05957.
25. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein generative adversarial networks. In Proceedings of the International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 214–223.