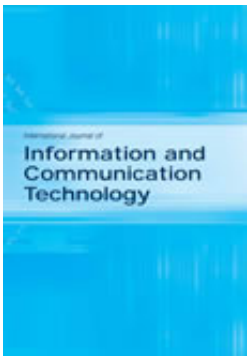


International Journal of Information and Communication Technology

ISSN online: 1741-8070 - ISSN print: 1466-6642
https://www.inderscience.com/ijict

Chinese character style transfer based on improved StarGAN v2 network

Ruohao Wang, Yilihamu Yaermaimaiti

DOI: 10.1504/IJICT.2024.10063507

Article History:

Received:	19 January 2024
Last revised:	22 February 2024
Accepted:	24 February 2024
Published online:	12 May 2024

Chinese character style transfer based on improved StarGAN v2 network

Ruohao Wang and Yilihamu Yaermaimaiti*

School of Electrical Engineering,
Xinjiang University,
Urumqi, Xinjiang 830047, China
Email: 1403346085@qq.com
Email: nf193ukoh781@126.com
*Corresponding author

Abstract: Chinese character generation has attracted a lot of attention due to its wide range of applications. Mainstream methods for generating Chinese character fonts are mainly based on generative adversarial networks, however, the structure of Chinese characters is more complex than other fonts and the problem of font structure change and style loss occurs when generating complex fonts and the mainstream methods require paired datasets, which is difficult and time-consuming to collect paired datasets. This paper proposes Trans-StarGAN v2 network for the above problems, which is based on StarGAN v2, introduces the Transformer structure for spatial feature extraction and channel feature extraction, which improves the feature extraction and generation ability of the network, and secondly, introduces the perceptual loss to strengthen the model training process. The experimental results show that compared with other Chinese character generation networks, the proposed network can generate multiple styles of fonts at the same time, improve the quality of the generated characters, preserve the structure of the fonts and make the style more complete in the face of complex fonts, and improve the FID and LPIPS indexes of the generated Chinese character content.

Keywords: transformer; generative adversarial network; GAN; style migration; StarGAN v2.

Reference to this paper should be made as follows: Wang, R. and Yaermaimaiti, Y. (2024) 'Chinese character style transfer based on improved StarGAN v2 network', *Int. J. Information and Communication Technology*, Vol. 24, No. 6, pp.71–91.

Biographical notes: Ruohao Wang graduated with a Bachelor's degree from Zhengzhou University of Aeronautics and is currently pursuing a graduate degree at Xinjiang University, with a research focus on pattern recognition and intelligent systems.

Yilihamu Yaermaimaiti is a Professor at Xinjiang University, specializes in areas including pattern recognition and intelligent control, modern power electronics technology, and artificial intelligence.

1 Introduction

Chinese is currently the most widely spoken language in this world. Chinese characters are widely utilised throughout the Chinese country and have been instrumental in the dissemination of Chinese culture, serving as the language's carrier. Chinese characters have evolved over thousands of years, mirroring the growth of the Chinese country. From oracle bone inscriptions to small seal writing, clerical script, regular script, and cursive script. Human communication has increasingly transitioned from traditional letters to voice and video in recent years due to the popularity of social media applications and the rise of short films, which has resulted in the phenomenon known as 'forgetting how to write characters'. Chinese characters remain a subject of interest in the age of digital information, leading to an increasing amount of research being done on them.

At the moment, most research focuses on the identification and categorisation of Chinese characters (Shi et al., 2016; Busta et al., 2017; Ul-Hasan et al., 2015; Yin et al., 2017), with comparatively less attention paid to character generation, particularly when considering multi-style Chinese character production. Common methods of generating Chinese characters fall into two categories: traditional methods and those based on deep learning. Liu et al. (2012) and others initially decomposed Chinese characters into various components such as strokes and radicals, and then recombined these to generate the target font. The introduction of generative adversarial networks (GANs) (Goodfellow et al., 2020) in 2014 received widespread attention and was also employed for the task of Chinese character generation. GANs have derived many variants including CGAN (Mirza and Osindero, 2014), WGAN (Arjovsky et al., 2017), BIGGAN (Brock et al., 2018), CYCLEGAN (Zhu et al., 2017), etc. CGAN can control the classes of the generated data but requires labelling of the dataset. WGAN solves the problem of pattern collapse for training with the disadvantage that it is prone to gradient vanishing. BIGGAN generates high-resolution images but is slow to train because of the large model. CYCLEGAN can be trained using unpaired datasets with the disadvantage that geometric transformations of the image are not obvious. Isola et al. (2017) designed a style transfer model, pix2pix, based on conditional GANs, providing a general framework for image-to-image style transfer. This model, utilising a U-Net architecture generator, enhanced the detail in generated images.

With the development of the style migration model, the task of generating Chinese characters is viewed as a transition between different styles of images. Gao and Wu (2020) performed Chinese character style transformation by training multiple GANs to be used in combination through three steps, namely skeleton extraction, skeleton transformation, and stroke rendering. This approach focuses on preserving the overall glyph content of the Chinese characters so that the resulting Chinese character shapes do not change too much, but it requires training three networks and spending a lot of time tuning each network to achieve the best results. In 2017, Tian (2017) proposed the Chinese character generation model Zi2Zi based on pix2pix. This model introduced category embedding based on the multi-class classification loss in the ACGAN model, allowing for the simultaneous modelling of numerous typefaces. This improved the discriminator in line with the expected results. With Zi2Zi, a source font could be translated into multiple styles of fonts simultaneously. However, both pix2pix and Zi2Zi

required large datasets of paired Chinese characters, necessitating the corresponding target style fonts for the input characters. The variability in styles even when the same person writes the same character poses challenges in dataset collection, making the use of unpaired datasets particularly important in Chinese character generation tasks. Li and colleagues computed the structural loss between the output and input images by using graphical matching techniques after extracting important points of Chinese character structures using the SSD object detection algorithm. A new font style model, OFM-CycleGAN, was proposed by Zhang (2019) and colleagues and is based on an enhanced feature matching technique. To tackle the challenge of generating fonts from unpaired datasets, Chang et al. (2018) proposed the HCCG-Cycle GAN network, defining the task of learning the mapping from existing printed fonts to target handwritten font styles. Although this method resolved the issue of paired datasets, it could only achieve the transfer of a single attribute. When multiple style attributes are required, training multiple models becomes necessary. In order to deal with the changes in the structure of Chinese characters in the Chinese character style migration, CS-GAN (Xiao et al., 2021) is proposed. CS-GAN introduces distribution transformations, reparameterisation techniques, and sampling features to ensure the effective transformation of high-dimensional features and low-dimensional features. TH-GAN (Cai et al., 2019) targets historical Chinese character recognition. For blurry, low-quality Chinese character images, the network focuses on the edges of the text and the font structure information to generate a target image. Choi et al. (2018), building on StarGAN, introduced the StarGAN v2 (Choi et al., 2020) model, addressing the problem of multi-domain translation in the image-to-image process. Subsequently, based on the StarGanV2 model many scholars made different improvements and applications (Li and Gu, 2023; Holmes et al., 2023; Ko et al., 2023; Ning, 2022; Wang et al., 2020). In 2017, Zeng et al. (2021) introduced a diversity regulariser in the StarGAN network, resulting in better diversity of the generated Chinese characters. In 2021, pan fused style-attentional net into jump-connected U-Net as a generator for GAN (Zeng and Pan, 2022), which effectively integrates local style patterns based on the semantic spatial distribution of content images while preserving feature information of different sizes. In 2022, Ning and others added a self-attention mechanism to the StarGAN V2 network to make the generated font strokes clearer and higher quality.

The introduction of the transformer (Vaswani et al., 2017) in 2017, primarily as a deep learning model for sequence-to-sequence problems, marked a significant advancement in natural language processing (NLP) tasks. In recent years, the transformer architecture has also been applied in the field of computer vision (CV) and beyond. Jiang et al. (2021) proposed TransGAN, which employs the transformer structure to construct a GAN model. Due to its use of memory-efficient generators that incrementally increase the resolution of feature maps, it achieved impressive results. Lee and others introduced VITGAN. VITGAN (Lee et al., 2021) is built upon the vision transformer (Dosovitskiy et al., 2020), exhibiting competitive picture recognition performance with reduced vision-specific inductive bias requirements. VITGAN was the first to prove that the transformer architecture could match the effectiveness of traditional convolutional GANs, challenging the dominance of convolutional neural networks (CNNs) in CV. Subsequently, the research on the use of transformer in combination with GAN has become more and more in-depth, and the more famous literature includes (Zhang et al., 2022; Lin et al., 2018; Jiang et al., 2021; Luo et al., 2021; Li et al.,. 2021, 2022, 2023;

Metsis et al., 2022; Dalmaz et al., 2022), and the research on the combination of transformer and GAN is also one of the mainstream directions in the future.

This paper is based on the StarGAN V2 model for multi-style font generation research, the study found that: the original model in the generation of the target style fonts will change the font structure will lead to the emergence of the wrong fonts, and in the font outline of the style inconsistency, so that the output fonts are not the fonts that we need, in view of these problems, we proposes a font style transfer network structure,Trans-StarGanV2, which incorporates the Transformer structure in the generator. This integration aims to address the limitations of convolutions in capturing long-range pixel relationships. The experiments were conducted on a handwritten dataset containing three styles: Founder Shuti, regular script, and clerical script. The results demonstrate that the network generates fonts with more standardised structures, preserving the integrity of strokes across different target styles and offering richer detail in style. Overall, the fonts generated by this network are of higher quality, and the network's performance is commendably robust.

The primary contributions of this paper are as follows:

1 The design of two distinct transformer-based feature extraction modules, the CTransformer feature extraction module and the STransformer feature extraction module, is introduced. Additionally, a CTransformer V2 feature fusion module that integrates adaptive instance normalisation (AdaIN) is presented. These developments significantly enhance the model's capability to collect features and improve the precision of the generated image styles.

2 The introduction of style-aware loss in font generation tasks represents a significant advancement. Compared to standard pixel-level loss, this perceptual loss is more appropriate for evaluating the quality of images produced by GANs because it is computed using feature representations from deep neural networks.

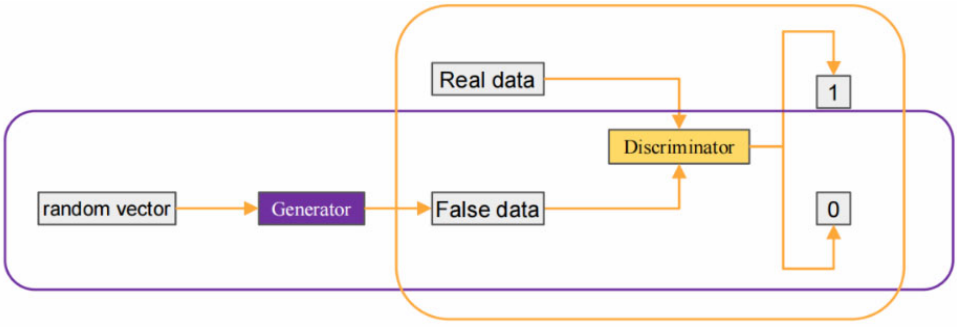
3 A new font generation model, Trans-StarGAN V2, is proposed. This model produces fonts with clearer structures, richer detail in style, and overall higher quality compared to those outputted by the original model.

2 Related works

2.1 GAN inversion

One kind of generative deep learning model is the GAN. A GAN is based on the adversarial training of two deep neural network models: a generator and a discriminator. The structure of a GAN, involves the Generator receiving a random noise vector (usually following a uniform or Gaussian distribution) as input and mapping it to the output space through a series of transformations. The goal of the generator is to produce data that mimics the statistical properties of real data, thereby deceiving the discriminator and generating high-quality fake samples. The discriminator, a binary classifier, receives either real or generated data as input and attempts to distinguish their sources. Maximising the discriminator's capacity to accurately distinguish between produced and actual data is its goal, the basic structure of a GAN as shown in Figure 1.

Figure 1 Network architecture of GAN (see online version for colours)



The StarGAN network, which this paper is based on, was proposed in 2018 by Choi et al. to address the transformation issues of multi-domain images. It has a structure similar to the original GAN, built on a generator and a discriminator. An improved version, StarGAN v2, was released to address the restricted diversity of generated images and the scalability across numerous domains in the original StarGAN. The StarGAN v2 model has a mapping network and a style encoder in addition to the generator and discriminator. In order to create the necessary style images, the style encoder extracts the target style's style vectors, which are then integrated into the generator via adaptive normalisation. The discriminator is designed as a multi-branch classifier, performing binary classification in each branch. The overall loss function of StarGAN v2 is as shown in equation (1):

$$\min_{G,F,E} \max_{D} \mathcal{L}_{adv} + \lambda_{sty}\mathcal{L}_{sty} - \lambda_{ds}\mathcal{L}_{ds} + \lambda_{cyc}\mathcal{L}_{cyc} + \lambda_{per}\mathcal{L}_{per} \tag{1}$$

In this context, $\mathcal{L}_{adv}$ represents the adversarial loss function in equation (2), and $\tilde{s} = F_{\tilde{y}}(z)$:

$$\mathcal{L}_{adv} = \mathbb{E}_{x,y}\left[\log D_y(\mathbf{x})\right] + \mathbb{E}_{\mathbf{x},\tilde{y},\mathbf{z}}\left[\log\left(1 - D_{\tilde{y}}(G(\mathbf{x},\tilde{\mathbf{s}}))\right)\right] \tag{2}$$

$\mathcal{L}_{sty}$ is the style reconstruction loss in equation (3):

$$\mathcal{L}_{sty} = \mathbb{E}_{\mathbf{x},\tilde{y},\mathbf{z}}\left[\left\| \tilde{\mathbf{s}} - E_{\tilde{y}}(G(\mathbf{x},\tilde{\mathbf{s}}))\right\|_1\right] \tag{3}$$

$\mathcal{L}_{ds}$ is the style diversity loss in equation (4):

$$\mathcal{L}_{ds} = \mathbb{E}_{\mathbf{x},\tilde{y},\mathbf{z}_1,\mathbf{z}_2}\left[\left\| G(\mathbf{x},\tilde{\mathbf{s}}_1) - G(\mathbf{x},\tilde{\mathbf{s}}_2)\right\|_1\right] \tag{4}$$

$\mathcal{L}_{cyc}$ is the cycle consistency loss in equation (5):

$$\mathcal{L}_{cyc} = \mathbb{E}_{\mathbf{x},y,\tilde{y},\mathbf{z}}\left[\left\| \mathbf{x} - G(G(\mathbf{x},\tilde{\mathbf{s}}),\hat{\mathbf{s}})\right\|_1\right] \tag{5}$$

In addition to these losses, this paper introduces a new style-aware loss, utilising the pretrained VGG19 model to extract and compute the feature representations of generated and real images. This strategy significantly minimises the style discrepancy between the target style photos and the model-generated images by integrating the difference calculation into the training's overall loss function. This improves the network's overall performance. The perceptual loss is calculated using the feature maps obtained from the

first five convolutional layers of VGG19, denoted as conv1, conv2, conv3, conv4 and conv5, formula is shown in equation (6), where E represents the feature maps extracted by each convolutional layer of the pretrained VGG network, and F and G are the output and original images, respectively:

$$\mathcal{L}_{per} = \frac{1}{N}\sum_{i=1}^{N}\left\|\varphi_i\left(I_{out}\right)-\varphi_i\left(I_{gt}\right)\right\|_1 \tag{6}$$

3 Methods

3.1 Overall structure

The network presented in this paper is primarily designed based on the StarGAN network architecture. It comprises four main modules: a transformer-based generator module, a discriminator module, a mapping network module, and a StyleEncoder module for style encoding.

The generator module in this paper features a U-shaped structure and is primarily composed of three parts: the channel-based transformer (CBT) feature extraction module, the spatial-based transformer (SBT) feature extraction module, and the channel-based transformerV2 (CBTV2) feature fusion module. The use of both channel-based and SBTs enables the simultaneous extraction of features in both channel and spatial dimensions, resulting in more realistic model-generated outcomes that closely align with the desired results. The CBT module employs self-attention in the channel dimension to extract features, while also utilising up-sampling and down-sampling modules to respectively increase and decrease spatial resolution. After channel dimension feature extraction, the SBT module extracts spatial dimension features. These are then upscaled and combined using the CBTV2's AdaIN, incorporating style vectors extracted from the target font by the StyleEncoder module to generate the required style images. By enabling adversarial training between the discriminator and generator, the discriminator evaluates the veracity of the images produced by the generator, improving the generation network's overall performance. The structure of Trans-StarGAN v2 is shown in Figure 2.

3.2 CBT feature extraction module

Currently, the generators in GANs predominantly employ CNNs for feature extraction. However, the transformer, as a novel neural network architecture, enables feature extraction that is not solely reliant on CNNs. The transformer model addresses limitations such as the finite receptive field of CNNs and its unique self-attention mechanism allows the model to gather information not just from adjacent areas but from any position. Consequently, this paper proposes a transformer-based feature extraction module, named the CBT feature extraction module. Each CBT module consists of a LayerNorm layer; a multi-Dconv head transposed attention (MDTA) layer, and a gated-Dconv feed-forward network (GDFN) layer. The LayerNorm layer, or normalisation layer, normalises vectors in each dimension, thereby maintaining feature scale uniformity and enhancing the model's overall generalisation capability. The use of LayerNorm in transformers also improves the model's ability to process high-resolution images. The MDTA layer calculates attention not in the pixel dimension but in the channel dimension, eliminating

the need for interactive computations in the pixel dimension and instead calculating covariance in the feature channel dimension to extract feature maps. This is initially achieved through 1×1 convolution for cross-channel pixel aggregation, and then using 3×3 depthwise convolutions for local contextual channel-level aggregation. The structure of CBT is shown in Figure 3. This structure offers two main advantages.

The first advantage is the introduction of depthwise convolution, which emphasises local context before computing feature covariance to generate global attention maps. The second is the calculation of cross-channel covariance to produce attention maps that implicitly encode global context. The specific computation process is detailed in equations (7), (8), and (9).

$$K = W_d^K W_p^K X \tag{7}$$

$$Q = W_d^Q W_p^Q X \tag{8}$$

$$V = W_d^V W_p^V X \tag{9}$$

In the formula, X represents the input: W_d is a 3×3 depthwise convolution; W_p is a 1×1 point convolution; Subsequently, the projections Q and K undergo a reshape operation and then a dot product to generate a $\mathbb{R}^{\hat{C}\times\hat{C}}$ size channel attention map. The subsequent calculations are similar to those in a conventional transformer, as detailed in equation (10):

$$\hat{X} = W_P \hat{V} \cdot Soft\max\left(\frac{\hat{K}\cdot\hat{Q}}{\alpha}\right) \tag{10}$$

In the equation, $\hat{Q}$, $\hat{K}$ and $\hat{V}$ represent the results of Q, K, V after the reshape operation; α is a learnable scaling parameter, GDFN layer is a gated forward network based on local content 0fusion, emphasising spatial context. It first uses 1×1 convolutions for dimensionality increase, followed by 3×3 convolutions for feature extraction, and then gated with the GELU activation function. The GDFN's gating mechanism decides when and how complementary features should be passed down, enabling higher levels in the network hierarchy to concentrate on certain finer image characteristics and generate output of superior quality. Equation (11) provides an illustration of the computation process.

$$\hat{X} = W_P^0\left(\phi\left(W_d^1 W_P^1(X)\right)\odot W_d^2 W_P^2(X)\right) \tag{11}$$

In the formula: $\odot$ represents element-wise multiplication; ϕ denotes the GELU nonlinear function; W_d^1 and W_d^2 respectively represent the first and second linear projection layers after 3×3 depthwise convolution; W_P^1 and W_P^2 represent the first and second linear projection layers after 1×1 expansion convolution; W_P^0 represents a 1×1 dimension-reducing convolution.

Figure 2 Network architecture of Trans-StarGAN V2 (see online version for colours)

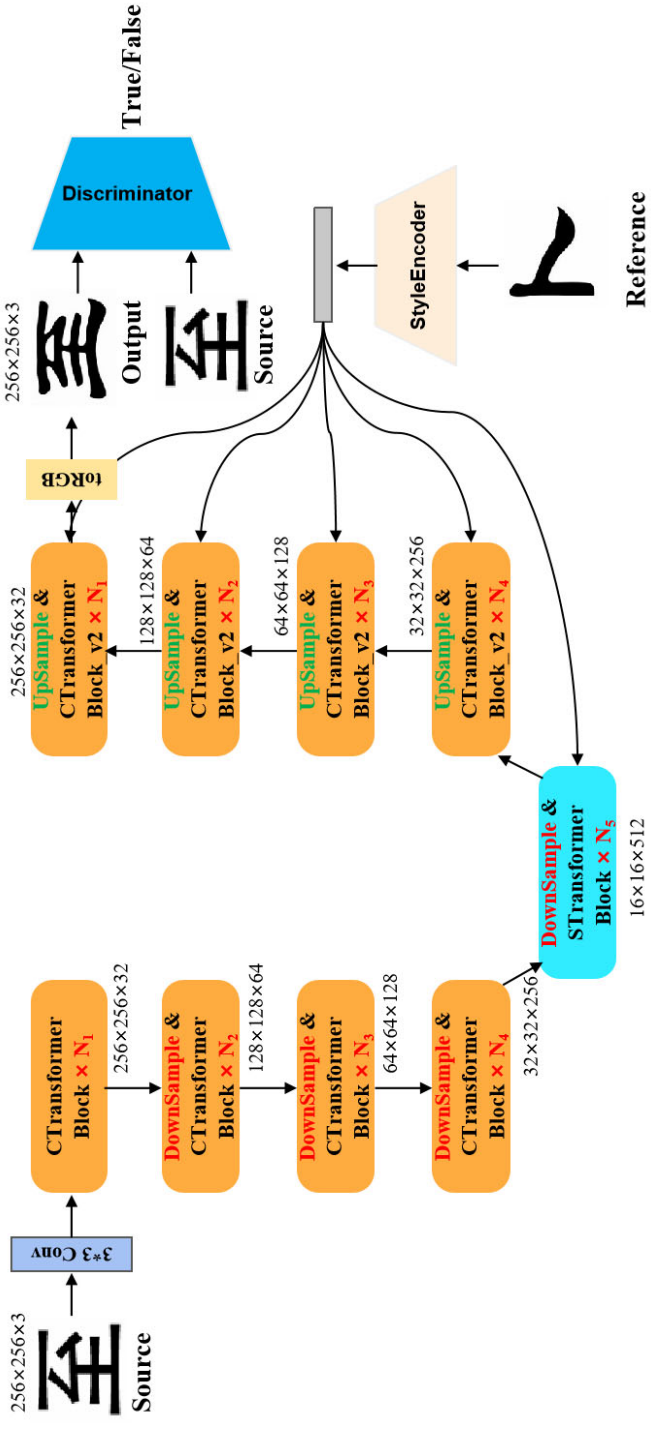


Figure 3 Network architecture of CBT (see online version for colours)

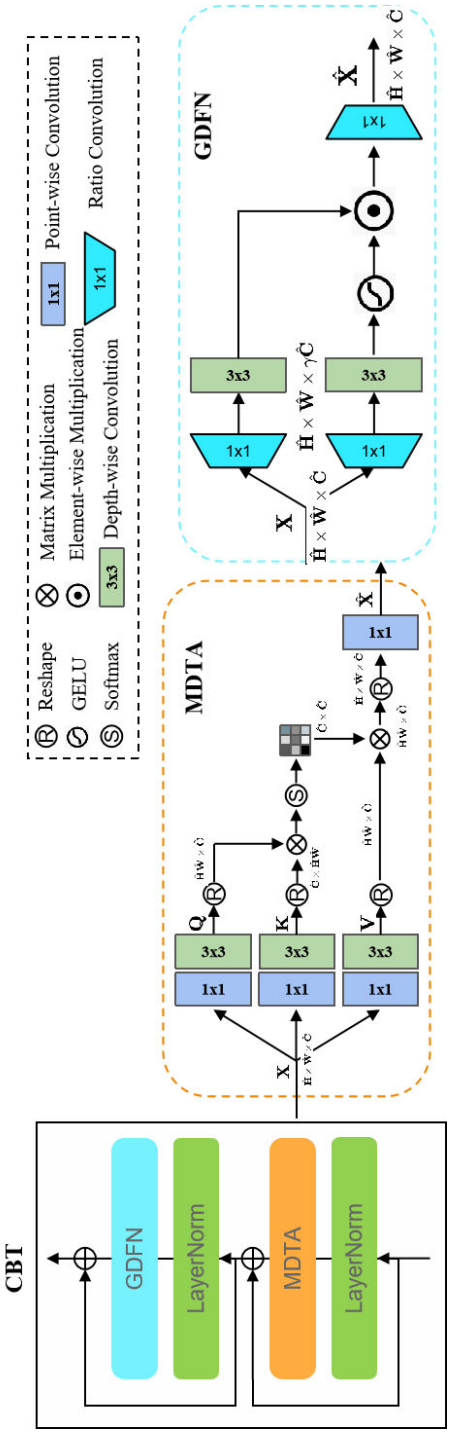


Figure 4 Network architecture of SBT (see online version for colours)

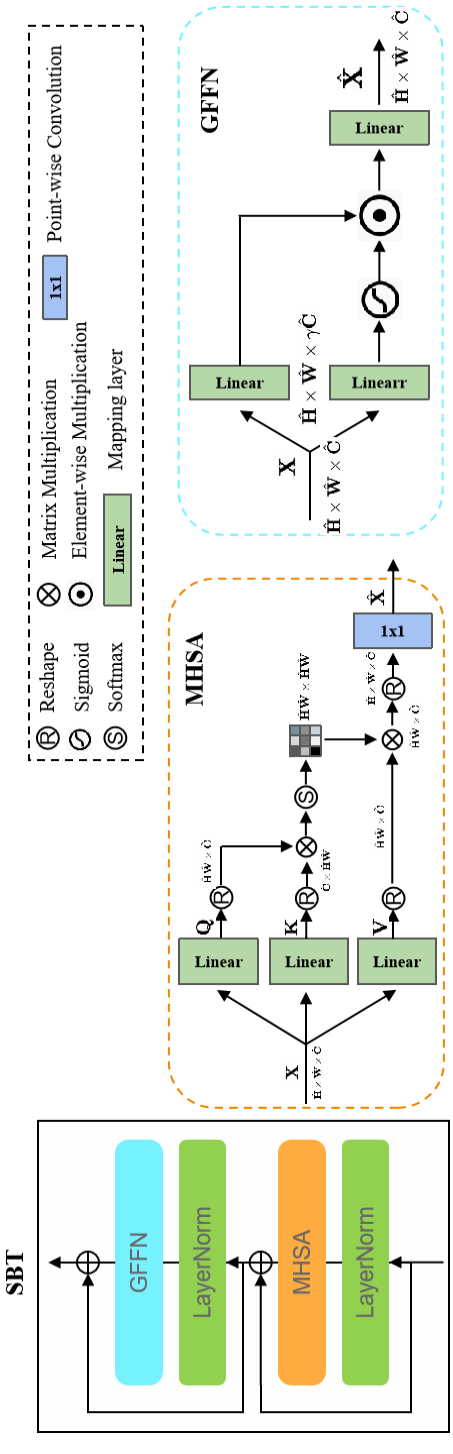
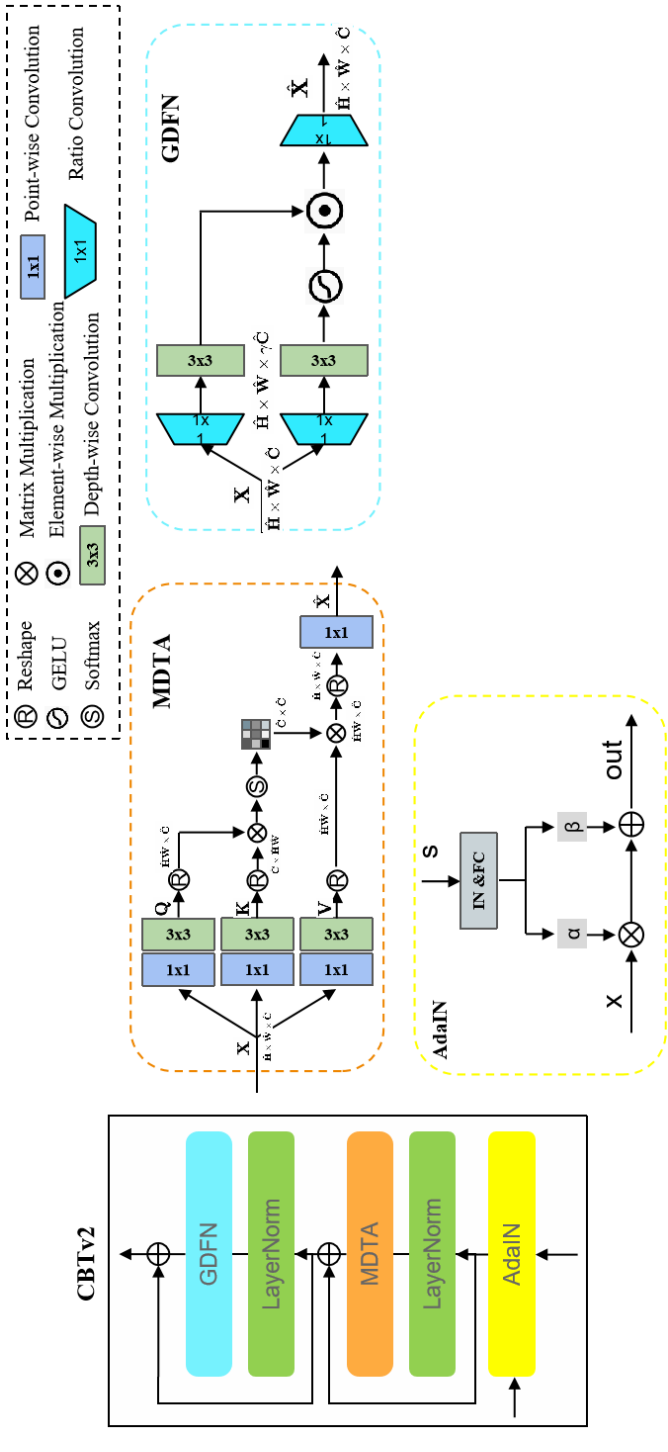


Figure 5 Network architecture of CBTV2 (see online version for colours)



3.3 SBT feature extraction module

Relying solely on the CBT feature extraction module for channel dimension feature extraction is insufficient. Therefore, the SBT feature extraction module was designed to facilitate global interaction of spatial features after the CBT module, optimising the final generated images. Since the SBT module is connected after the CBT, and undergoes multiple downsampling processes, the features are significantly reduced in the spatial dimension, focusing primarily on the channel level. This setup adapts well to the space-based self-attention operation. The SBT module is an enhancement of the conventional Transformer module's FFN layer. The structure of SBT is shown in Figure 4.

The specific structure of the SBT module is illustrated as follows: in the SBT module, the multi-head self-attention (MHSA) mechanism replaces the MDTA layer of the CBT, and a gated feed-forward network (GFFN) substitutes the GDFN in CBT. Through the CBT feature extraction and downsampling, the features are reduced by a factor of 16 in the spatial dimension, resulting in a size of only 16×16, while the channel dimension expands to 512. After processing through the CBT feature extraction module, comprehensive global interaction has already been achieved in the channel dimension, eliminating the need for self-attention on this level. Instead, self-attention is applied in the spatial dimension to compensate for the limitations in global interaction of 1×1 and 3×3 convolution operations, thereby enhancing the global representational capability of the extracted features.

$$Q = W_l^Q X \tag{12}$$

$$K = W_l^K X \tag{13}$$

$$V = W_l^V X \tag{14}$$

In the formula: X represents the input; W denotes a fully connected layer, subsequently, the projections Q and K are reshaped, facilitating their dot product interaction to generate a spatial attention map of size $\mathbb{R}^{\hat{H}\hat{W}\times\hat{H}\hat{W}}$. The subsequent calculations proceed as in a conventional transformer, as specifically illustrated in equation (15):

$$\hat{\mathbf{X}} = W_l \text{Soft}\max\left(\frac{\hat{\mathbf{Q}}\cdot\hat{\mathbf{k}}}{\alpha}\right)\cdot\hat{\mathbf{v}} \tag{15}$$

In the formula: $\hat{\mathbf{Q}}, \hat{\mathbf{k}}$ and $\hat{\mathbf{V}}$ are respectively the reshaped results of Q, K and V obtained after the reshape operation; α is a learnable scaling parameter. With one layer turned on by the GELU nonlinear function, the GFFN layer is still intended to be the element-wise product of two linear projection layers. The proposed GFFN also emphasises spatial context based on local content blending (similar to the MDTA module). The difference is that the GFFN performs linear mapping on channels, highlighting the fusion of channel features. The gating mechanism, similar to that in the MDTA, controls which complementary features should be forwarded. This allows subsequent layers in the network hierarchy to focus on finer image attributes, thereby generating higher quality output. The computation process is as described in equation (16):

$$\hat{\mathbf{X}} = W_l^0\left(\phi\left(W_l^1(\mathbf{X})\right) \odot W_l^2(\mathbf{X})\right) \tag{16}$$

$\odot$ represents element-wise multiplication; ϕ denotes the GELU nonlinear function; W_l^1 indicates the linear mapping of the first linear projection layer; W_l^2 refers to the linear mapping of the second linear projection layer; W_l^0 signifies the fusion linear mapping.

3.4 CBTV2 feature fusion module

The CBTV2 feature extraction module primarily differs from the CBT module in the inclusion of the feature fusion layer AdaIN. The AdaIN layer has two inputs: a content input *X* and a style input *S*. It synchronises the content features' mean and variation with the style features' mean and variance. The structure of CBTV2 is shown in Figure 5.

CBTV2 is utilised in the generator's upsampling process, where it combines style vectors through the AdaIN layer to generate the target font required. The structure is illustrated as shown, and the formula for AdaIN is presented in equation (17), where *s* represents the style vector, and *x* denotes the input content.

$$\text{AdaIN}(x, s) = \sigma(s)\left(\frac{x - \mu(x)}{\sigma(x)}\right) + \mu(x) \tag{17}$$

4 Experiment

We first go over the specifics of our implementation, and then we compare our approach with existing approaches on our dataset both numerically and qualitatively. Numerous ablation studies show how effective our module is.

4.1 Implementation dataset

In this experiment, we collected and produced a dataset with three different styles of fonts: Founder Shuti, clerical script, and regular script. These three fonts are the fonts that are used more often in Chinese characters, and they are the fonts that are most similar to the fonts that people write in their daily lives, so the difficulty of collecting the dataset is greatly reduced, because there are more than 80,000 Chinese characters in total, but there are only 20,000 fonts that are used in daily life, and the rest of 60,000 fonts are only found in the history books and the ancient books, so we did not include them in the dataset. We chose 10,000 commonly used fonts among the 20,000 commonly used characters. These three fonts were treated as three distinct classes within the model, with the dataset divided into training and validation sets. Each font's training set contained 10,000 images of different font characters, while the validation set for each font included 2,000 images of different font characters. The dataset underwent pre-processing, where each font image was resized to 256×256 pixels. Additionally, parts of the images with large blank spaces were cropped to maximise the font coverage ratio in the image. A partial sample of this experimental dataset is shown in Figure 6.

Figure 6 Sample data sets for this experiment (see online version for colours)

4.2 Experimental parameterisation and evaluation indicators

The experimental environment for this paper was set up using PyCharm based on the Pytorch framework. The code was written in Python 3.6.7 under the Windows operating system and utilised an Nvidia A4000 GPU for training, with a memory size of 16 GB. During the training experiments, the Adam optimiser was chosen to optimise the network, and $\beta_1 = 0$, $\beta_2 = 0.99$, with the training batch size set to 4; the number of iterations set at 100,000. The initial learning rate was set to 1e-4, we set $\lambda_{sty} = 1$, $\lambda_{ds} = 1$, $\lambda_{cyc} = 1$ and $\lambda_{reg} = 1$. The learning rates for G, D, E, and F were set to 10^{-4}.

Three different metrics were used to evaluate the font style generation accuracy of the network model in this paper:

1 Fréchet inception distance (FID), which measures the distance between two multivariate normal distributions.

2 Learned perceptual image patch similarity (LPIPS), a trained perceptual image patch similarity metric that quantifies the differences between two images using a more human-perceptible manner.

4.3 Experimental results

To thoroughly validate the efficacy of Trans-StarGanV2 in Chinese character generation, we conducted a comprehensive comparison with other font generation models in terms of objective measurement metrics as well as human subjective visual effects. All comparisons were based on the same dataset to ensure a fair premise.

Firstly, we compared the generated font images produced by the generators of Trans-StarGAN V2 network and StarGAN V2 network at different iteration steps. We observed that the font images generated by Trans-StarGAN V2 network exhibited higher quality and faster speed compared to those generated by the original network. At 1,000 iterations of both networks' generators, it was noted that the font images generated by Trans-StarGAN V2 displayed a complete font structure, whereas those generated by the original StarGAN V2 network appeared incomplete. Subsequently, at 2,000 iterations, it was observed that the background of the images generated by Trans-StarGAN V2 network turned white, consistent with the background colour of the target font, and the generated font images began to adopt the style of the target font. In contrast, font images generated by StarGAN V2 network still had a grey background, and the font structure remained incomplete until 5,000 iterations were reached to generate the correct background. Experimental results indicate that, regardless of the iteration count, the

image quality and generation speed of Trans-StarGAN V2 network are superior to those of StarGAN V2 network.

Upon examining the data from both the Trans-StarGAN V2 and StarGAN V2 models during the task of font style transfer, we have observed a noteworthy distinction. The original model, during its training process, tends to alter the structural integrity of Chinese characters, whereas the Trans-StarGAN V2 model preserves the original structure of these characters. Specifically, in the training process of the StarGAN V2 model, the character '背' undergoes a transformation where the component '月' changes to '目', thereby altering the meaning of the character '背' from its original sense. This misrepresentation of the character's meaning represents a significant error in the model's learning process.

Conversely, the Trans-StarGAN V2 model, owing to its transformer architecture, adeptly maintains the overall structure of the fonts. This is particularly evident when the model handles more complex characters like '陆'. It successfully retains the entire structure of the character, demonstrating its superior ability to preserve the linguistic integrity of the text during the style transfer process. This difference is exemplified in Figure 7 and Figure 8.

Figure 7 The results of the StarGAN V2 model during the iterations process

Figure 8 The results of the Trans-StarGAN V2 model during the iterations process

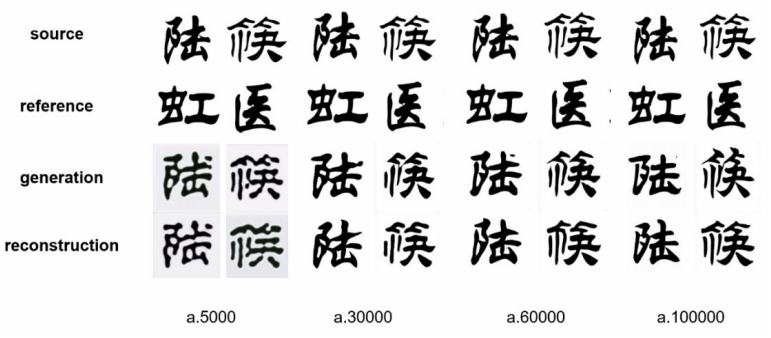
Secondly, to verify the preservation capability of the Trans-StarGAN V2 network regarding Chinese character font structures, we selected two characters with comparatively intricate typographic architectures: '陆' (Lu) and '筷' (Kuai). Subsequently, we documented the variations occurring in these two complex Chinese characters as the iteration count increased. We opted for four iteration counts as anchor points, specifically: 5,000, 30,000, 60,000, and 100,000. The resultant output consists of the source font image in the first row, the target font style image in the second row, the model-generated target style image in the third row, and the font image reconstructed by the network to maintain cyclical consistency in the fourth row. Transitioning the character '陆' from regular script to clerical script and '筷' from clerical script to regular script, we corroborated the Trans-StarGAN V2 network's ability to robustly preserve Chinese character font structures across various stylistic transformations.

Through experimentation, it was observed that the Trans-StarGAN V2 network consistently preserves the structural integrity of input fonts at any given iteration step. The preservation of Chinese character structure stands as a pivotal stage in the font generation process, as any alteration in structure may impact the intended semantics of the characters. Experimental findings indicate that at the iteration count of 5,000, the

generated images begin to exhibit traces of the target font style, albeit subtly. By the time the iteration count reaches 30,000, a substantial portion of the font closely aligns with the target style. Optimal results are achieved at 100,000 iterations, where the generated font displays superior clarity and stylistic similarity compared to that at 60,000 iterations.

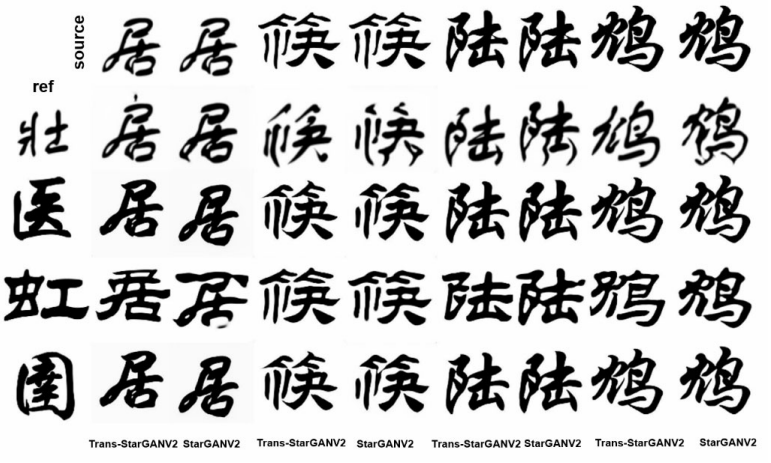
As the number of iterations increases, the generated results are shown in Figure 10.

Figure 9 Results under different iterations



In Figure 10, we compared the output results of the Trans-StarGAN v2 network and the StarGAN v2 network in simultaneously transforming three different styles of fonts: Founder Shuti, clerical script, and regular script, as illustrated in Figure 10. The first row represents the input font, while the first column denotes the target font.

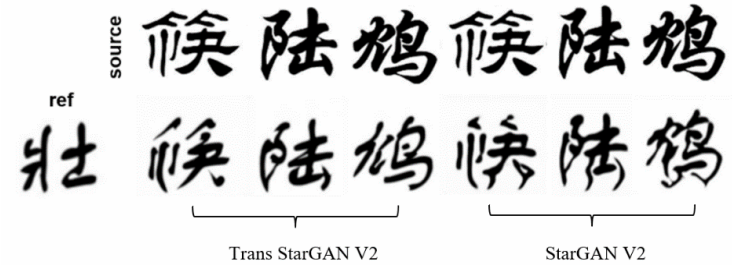
Figure 10 Example of Trans-StarGAN v2 and StarGAN v2 font generation results



In Figure 11, it is evident that compared to the original StarGAN v2 network, the Trans-StarGAN v2 network preserves the style and structure of the target font more accurately at the character level, particularly for complex characters. For instance, focusing on the structure of the character '陆', when transitioning from regular script to Fangzheng Shuti, the Trans-StarGAN v2 network transforms the bottom strokes of the character '陆' into the style of Fangzheng Shuti while retaining the original font's structure. Conversely, the output generated by the StarGAN v2 network for the character

‘陆’ does not achieve the target style and introduces distortion at the bottom of the character, resulting in a change in the font structure. Similarly, for the characters ‘筷’ and ‘鸠’, the results generated by the StarGAN v2 network exhibit noticeable distortion, while those generated by the Trans-StarGAN v2 network not only conform to the target style but also present a more aesthetically pleasing font structure overall.

Figure 11 The generated results for the characters ‘陆’, ‘筷’, and ‘鸠’



Above is our analysis of the generated images. Next, we will further analyse the performance of the model by calculating the FID metric. Upon examination of the FID metrics of the two models, it becomes evident that our model exhibits the most substantial enhancement in the transition from Founder Shuti to clerical, manifesting a notable reduction of seven points in the metrics. From the result images, it’s also clear that our model significantly outperforms the original StarGAN v2 model in terms of effect. Other style transformations also show considerable improvements in FID metrics Tables 1 and 2 present the font style transfer FID metrics for the Trans-StarGAN v2 model and StarGAN v2, respectively.

Table 1 FID values for trans-StarGAN V2 models on the dataset

S \ *T*	*Founder Shuti*	*Regular*	*Clerical*
Founder Shuti	Null	44.821	35.032
Regular	18.712	Null	34.502
Clerical	24.123	30.215	Null

Table 2 FID values for StarGAN v2 models on the dataset

S \ *T*	*Founder Shuti*	*Regular*	*Clerical*
Founder Shuti	Null	48.234	42.124
Regular	25.242	Null	38.425
Clerical	28.324	33.564	Null

Table 3 presents the metrics for various font generation models trained on our dataset, indicating that our model outperforms others in these metrics. Compared to models like zi2zi, HCCG-CycleGAN, and StyleGAN, our model achieves distinct advantages in both FID and LPIPS evaluation metrics. Our method, unlike zi2zi and HCCG-CycleGAN, only requires constructing a target style dataset for immediate training, imparting the

desired style features onto the input original images. The use of the transformer structure has demonstrated superior performance in aspects of detailed extraction and fusion of font styles. Experimental evidence shows that the CBT and SBT modules effectively explore the feature relationships in spatial channels, significantly enhancing the fusion of font style generation across various evaluation metrics.

Table 3 FID values and LPIPS values for Trans-StarGAN V2 models on the dataset

Models	*FID*	*LPIPS*
Zi2Zi	91.35	0.1163
HCCG-CycleGAN	81.35	0.0824
StyleGAN	59.34	0.0624
CycleGAN	94.34	0.1453
StarGAN V2	58.54	0.0682
Ours	48.23	0.0321

4.4 Ablation study

To validate the efficacy of the proposed network model, we conducted ablation experiments on the model. Initially, our model combines the use of transformer and convolution. To verify the effectiveness of the transformer structure, three sets of ablation experiments were performed. These involved replacing the CBT and SBT modules with the original model's ResBLK, and substituting the CBTV2 with the original model's AdaINResBLK. The effectiveness of these modifications was then assessed using FID and LPIPS metrics. The results of the ablation experiments are shown in Table 4.

Table 4 Generator layers ablation experiment results

Models	*FID*	*LPIPS*
CBT ONLY	66.34	0.0635
SBT ONLY	89.25	0.1025
CBTV2 ONLY	91.78	0.1123
CBT+SBT	56.64	0.0501
SBT+CBTV2	61.25	0.0542
CBT+CBTV2	55.14	0.0421
Ours	48.23	0.0321

The experimental results revealed that when feature extraction was confined to either channels or spatial dimensions, the FID metrics increased by 30–40. This indicates a significant deviation of the generated images from the target style, demonstrating that the Transformer structure, in contrast to convolution, is not limited to a small receptive field. Instead, it extracts features from a global perspective, which proves to be more advantageous than convolution. The use of CBTV2 and AdaINResBLK for upsampling and style fusion showed a relatively minor improvement. This is attributed to the use of AdaIN for combining the target style by calculating the mean and variance for alignment. This choice of AdaIN as the style transfer module is justified by its effectiveness in terms of speed and flexibility.

5 Conclusions

Based on the multi-style transfer model StarGAN v2, this paper proposed the Trans-StarGAN v2, a novel multi-style transfer model, by incorporating a transformer structure. The model initially utilises an improved Transformer feature extraction module to extract channel and spatial features, followed by an upsampling module integrated with AdaIN to fuse the target style onto the image. Subsequently, the discriminator evaluates the images generated by the generator to facilitate adversarial training, enhancing the capabilities of both the generator and the discriminator. This model has been trained on a font dataset comprising three styles: clerical script, Shu style, and semi-cursive script. Through intuitive comparison of the generated Chinese character images and evaluation based on the FID and LPIPS metrics, it has been found that Trans-StarGAN v2 surpasses StarGAN v2 in generative ability, producing higher quality Chinese character images. Compared to other partial font generation models, it achieves better results in both overall and detailed aspects of the font. However, the parameters in the feature extraction process are crucial factors affecting network performance. Therefore, future research will focus on exploring the impact of parameters on model efficacy and optimising the transformer structure for subsequent device deployment.

Acknowledgements

The authors want to acknowledge the financial support from the National Natural Science Foundation of China (Project No: 62362063, 61866037).

References

Arjovsky, M., Chintala, S. and Bottou, L. (2017) 'Wasserstein generative adversarial networks', in *International Conference on Machine Learning*, PMLR, July, pp.214–223.

Brock, A., Donahue, J. and Simonyan, K. (2018) *Large Scale GAN Training for High Fidelity Natural Image Synthesis*, arxiv preprint arxiv:1809.11096.

Busta, M., Neumann, L. and Matas, J. (2017) 'Deep textspotter: an end-to-end trainable scene text localization and recognition framework', in *Proceedings of the IEEE International Conference on Computer Vision*, pp.2204–2212.

Cai, J., Peng, L., Tang, Y. et al. (2019) 'TH-GAN: generative adversarial network based transfer learning for historical Chinese character recognition', *2019 International Conference on Document Analysis and Recognition (ICDAR)*, IEEE, pp.178–183.

Chang, B., Zhang, Q., Pan, S. et al. (2018) 'Generating handwritten Chinese characters using cyclegan', *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, IEEE, pp.199–207.

Choi, Y., Choi, M., Kim, M., Ha, J-W., Kim, S. and Choo, J. (2018) 'StarGAN: unified generative adversarial networks for multi-domain image-to-image translation', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.8789–8797.

Choi, Y., Uh, Y.J., Yoo, J. and Ha, J-W. (2020) 'StarGAN v2: diverse image synthesis for multiple domains', *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.8188–8197.

Dalmaz, O., Yurt, M. and Çukur, T. (2022) 'ResViT: residual vision transformers for multimodal medical image synthesis', *IEEE Transactions on Medical Imaging*, Vol. 41, No. 10, pp.2598–2614.

Dosovitskiy, A., Beyer, L., Kolesnikov, A. et al. (2020) *An Image is Worth 16x16 Words. Transformers for Image Recognition at scale*, arXiv preprint arXiv:2010.11929.

Gao, Y. and Wu, J. (2020) 'GAN-based unpaired Chinese character image translation via skeleton transformation and stroke rendering', *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34, No. 1, pp.646–653, https://doi.org/10.1609/aaai.v34i01.5405.

Goodfellow, I., Pouget-Abadie, J., Mirza, M. et al. (2020) 'Generative adversarial nets', *Communications of the ACM*, Vol. 63, No. 11, pp.139–144.

Holmes, K., Sharma, P. and Fernandes, S. (2023) 'Facial skin disease prediction using StarGAN v2 and transfer learning', *Intelligent Decision Technologies*, Vol. 17, No. 1, pp.55–66.

Isola, P., Zhu, J-Y., Zhou, T. and Efros, A.A. (2017) 'Image-to-image translation with conditional adversarial networks', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.1125–1134.

Jiang, Y., Chang, S. and Wang, Z. (2021) 'Two pure transformers can make one strong GAN, and that can scale up', *Advances in Neural Information Processing Systems*, Vol. 34, pp.14745–14758.

Jiang, Y., Chang, S. and Wang, Z. (2021) *Transgan: Two Transformers Can Make One Strong GAN*, arXiv preprint arXiv:2102.07074, Vol. 1, No. 3.

Ko, K., Yeom, T. and Lee, M. (2023) 'SuperstarGAN: generative adversarial networks for image-to-image translation in large-scale domains', *Neural Networks*, Vol. 162, pp.330–339, ISSN: 0893-6080, https://doi.org/10.1016/j.neunet.2023.02.042.

Lee, K., Chang, H., Jiang, L. et al. (2021) *Vitgan: Training GANS with Vision Transformers*, arXiv preprint arXiv:2107.04589.

Li, R. and Gu, J. (2023) 'OMGD-StarGAN: improvements to boost StarGAN v2 performance', *Evolving Systems*, https://doi.org/10.1007/s12530-023-09521-0.

Li, X., Metsis, V., Wang, H. et al. (2022) 'TTS-GAN: a transformer-based time-series generative adversarial network', *International Conference on Artificial Intelligence in Medicine*, Springer International Publishing, Cham, pp.133–143.

Li, Y., Han, N., Qin, Y., Zhang, J. and Su, J. (2023) 'Trans-cGAN: transformer-Unet-based generative adversarial networks for cross-modality magnetic resonance image synthesis', *Statistics and Computing*, Vol. 33, No. 5, p.113.

Li, Y., Peng, X., Zhang, J. et al. (2021) 'DCT-GAN: dilated convolutional transformer-based GAN for time series anomaly detection', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 35, No. 4, pp.3632–3644.

Lin, C.H., Yumer, E., Wang, O. et al. (2018) 'ST-GAN: spatial transformer generative adversarial networks for image compositing', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.9455–9464.

Liu, P., Xu, S. and Lin, S. (2012) 'Automatic generation of personalized Chinese handwriting characters', *2012 Fourth International Conference on Digital Home*, Guangzhou, China, pp.109–116, DOI: 10.1109/ICDH.2012.77.

Luo, Y., Wang, Y., Zu, C. et al. (2021) '3D transformer-GAN for high-quality PET reconstruction', *Medical Image Computing and Computer Assisted Intervention – MICCAI 2021: 24th International Conference*, Strasbourg, France, 27 September–1 October, Proceedings, Part VI 24, Springer International Publishing, pp.276–285.

Metsis, V., Wang, H. and Ngu, A.H.H. (2022) 'TTS-GAN: a transformer-based time-series generative adversarial network', in *International Conference on Artificial Intelligence in Medicine*, Springer International Publishing, Cham, June, pp.133–143.

Mirza, M. and Osindero, S. (2014) *Conditional Generative Adversarial Nets*, arxiv preprint arxiv:1411.1784.

Ning, F. (2022) 'Multi-style migration of Chinese characters based on self-attention mechanism and StarGAN v2', *2022 3rd International Conference on Computer Vision, Image and Deep Learning & International Conference on Computer Engineering and Applications (CVIDL & ICCEA)*, Changchun, China, pp.357–361, DOI: 10.1109/CVIDLICCEA56201.2022.9825460.

Shi, B., Bai, X. and Yao, C. (2016) 'An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 39, No. 11, pp.2298–2304.

Tian, Y. (2017) *Master Chinese Calligraphy with Conditional Adversarial Network* [EB/OL] [online] https://github.com/kaonashi-tyc.

Ul-Hasan, A., Shafaity, F. and Liwicki, M. (2015) 'Curriculum learning for printed text line recognition of ligature-based scripts', *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, IEEE, pp.1001–1005.

Vaswani, A., Shazeer, N. and Parmar, N. (2017) 'Attention is all you need', *Advances in Neural Information Processing Systems*, p.30.

Wang, Y., Wang, Z. and Jiang, M. (2020) 'Stargan-based camera style transfer for person retrieval', *International Journal of Information and Communication Technology*, Vol. 16, No. 1, pp.1–16.

Xiao, Y., Lei, W., Lu, L. et al. (2021) 'CS-GAN: cross-structure generative adversarial networks for Chinese calligraphy translation', *Knowledge-Based Systems*, Vol. 2021, No. 229, p.107334.

Yin, F., Wu, Y.C., Zhang, X.Y. et al. (2017) *Scene Text Recognition with Sliding Convolutional Character Models*, arXiv preprint arXiv:1709.01727.

Zeng, J., Chen, Q. and Wang, M. (2021) 'Diversity regularized stargan for multi-style fonts generation of Chinese characters', *Journal of Physics: Conference Series*, Vol. 1880, No. 1, p.012017, IOP Publishing.

Zeng, S. and Pan, Z. (2022) 'An unsupervised font style transfer model based on generative adversarial networks', *Multimedia Tools and Applications*, Vol. 81, No. 4, pp.5305–5324.

Zhang, B., Gu, S., Zhang, B. et al. (2022) 'Styleswin: transformer-based GAN for high-resolution image generation', *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp.11304–11314.

Zhang, Y. (2019) *Generating Handwritten Chinese Character with GANs*, East China Normal University, Shanghai.

Zhu, J.Y., Park, T., Isola, P. and Efros, A.A. (2017) 'Unpaired image-to-image translation using cycle-consistent adversarial networks', in *Proceedings of the IEEE International Conference on Computer Vision*, pp.2223–2232.