The Visual Computer (2021) 37:2895–2906
https://doi.org/10.1007/s00371-021-02218-y

ORIGINAL ARTICLE

# Generative character inpainting guided by structural information

Haolong Li[1] · Zizheng Zhong[1] · Wei Guan[1] · Chenghao Du[1] · Yu Yang[3] · Yuxiang Wei[1] · Chen Ye[1,2]

Accepted: 16 June 2021 / Published online: 30 June 2021

**Abstract**
Character inpainting is an attractive and challenging task, especially for Chinese calligraphy characters with complex structures and styles. The diversity of Chinese calligraphy styles has created its unique artistic beauty, but specific style features will lead to obvious differences in style and stroke details while recovering. Most current methods are restricted to recover specific characters already present in the training set and retrain the model when recovering characters of new styles. Moreover, these methods are based on edge recovery, which requires the location of the masked area. In this paper, we propose a novel structure-guided generative framework guided by prototype character, which can not only adapt to multiple style fonts but also overall recover glyph structure and strokes without masked information by inferring the style representation. In this case, our method can recover new style characters, which is the first attempt for character inpainting without parameter retraining. Experimental results demonstrate that our method has generalizability and superiority in most application scenarios, compared with several state-of-the-art character inpainting methods.

**Keywords** Character inpainting · Generative adversarial networks · Structural information · Style encoder

## 1 Introduction

Cultural relics are precious cultural heritages inherited in the development of human historical life, reflecting the value of historical research, art and scientific archaeology. Some cultural relics are inherited in the form of words, and the recorded words play a vital role in the inheritance and development of culture. However, as time goes on, the cultural relics are eroded by oxidation, light and weathering, which makes the characters lose their original shape and contain erosion and noise. Taking rubbings of Chinese calligraphy as an example, the Chinese characters on many inscriptions have been corroded and lost their artistic and cultural value. Therefore, character inpainting is essential in realizing the value of ancient literature and serving as important ingredients of historical methodology.

Moreover, character inpainting is an important branch of pattern recognition, but it remains under-explored compared with character style transfer, which is mainly reflected in the hieroglyphs represented by Chinese characters. Chinese characters are one of the most widespread and long-used languages, which are incorporated into many other Asian languages, such as Japanese and Korean. The long history of Chinese characters has caused the erosion to be more frequent and serious. Besides, unlike traditional phonological languages, such as English and Latin, which have a limited number of characters, Chinese has a much larger dictionary with thousands of characters. To put it in practical terms, there are 27,533 unique Chinese characters in the official GB18030 charset, with daily used ones up to 3000 [1,2]. The complex structure and variable style of Chinese characters have also exacerbated the challenge of Chinese character inpainting and hindered development. Thus, the expanding demand for recovering characters of special styles emerges as the times require.

The early restoration of Chinese characters is based on stroke feature extraction [3,4]. Strokes are decomposed from undamaged images to construct a stroke dataset. Then, the features of the strokes are extracted from the eroded area. Through matching algorithms, similar strokes are selected from the stroke set to recover eroded Chinese characters.

✉ Chen Ye
yechen@tongji.edu.cn

[1] Department of Computer Science and Technology, Tongji University, Shanghai, China

[2] The Key Laboratory of Embedded System and Service Computing, Ministry of Education, Tongji University, Shanghai, China

[3] Department of Software Engineering, Tongji University, Shanghai, China

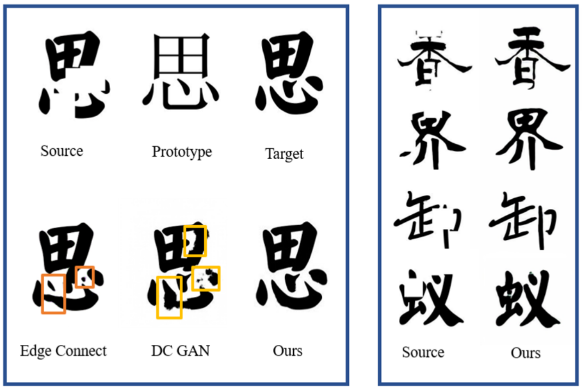

**Fig. 1** In the left part, we show the comparison chart between our method and other methods. As two of the advanced methods only based on the eroded image, EdgeConnect generates wrong strokes as providing error margin information, while DCGAN performs poorly on the font with diverse styles as well. Our method additionally combines prototype character images to achieve the desired effect. In the right part, we show the results of recovering characters of diverse styles

Due to the limitation of the stroke set, this method lies in multiple matching, while partial repair of the stroke can't keep the overall style of the character.

Differently, with the emergence of the Generative Adversarial Network [5–7] and its various applications in image inpainting tasks, most current methods achieve the overall image generation, making the inpainting character more natural. It is able to take context relevance into consideration by using learned data [8]. However, compared with other images, character images have poorer context relevance due to the differences and separation between strokes. The method based on edge detection cannot obtain enough contextual information in masked area, especially the center area of erosion. The loss of information leads to style differences in the repaired area and even wrong strokes as shown in Fig. 1. Besides, the information of the eroded area is required for these methods before recovering. The requirement of the masked information results in large amounts of time consumption. A lot of human resources need to be put into the marking of masked information, which greatly increases the workload of data input. The labelling standard turns out to be vague due to the differences of marking personnel, leading to the reduced accuracy of masked information. Moreover, these methods only focus on standardized Chinese characters or handwritten characters [9–11], which cannot be applied to actual cultural relic recovery. They poorly specialize the stylized characters, especially Chinese calligraphy characters, which embodies historical research, art and scientific archaeological value. In other fields, some successful examples guiding by structure information provide reference for us as well [12–14].

In this paper, we creatively propose a novel network that recovers characters for diverse font styles, which omits the input of masked information and automatically recovers images without retraining. We believe that the information of each character, which consists of both structure and style, is indivisible between the eroded part and the rest. Therefore, a style encoding mapping network and a pretrained recognition network are used to infer the style and content representation. In this case, our method based on overall inpainting can automatically recover new style character images without parameter retraining. We also introduce the concept of the prototype character information. Different from one-hot encoding, the prototype character information is not a code string of characters, but an intact image of regular script with more complete content information than the eroded ones. This image, which can be in discretionary styles, is considered to provide the structure information of masked area. The prototype images are also the input with the eroded character image to make up for the context information. Combining with the code extracted to provide style information, the network outputs a character image which is highly consistent with the target image. Based on the idea of introducing character prototypes, we discussed the feasibility of controlling local invariances in style transfer by merging local images with source images.

To sum up, our major contributions can be concluded as:

- We propose a novel framework for character inpainting without parameter retraining. The model can automatically adapt to new style character images by inferring the style and content representation.
- To guarantee the recovering consistency of style between the masked part and the rest, we innovatively provide a more effective recovery idea based on overall inpainting instead of edge recovery.
- We introduce intact character images of discretionary styles as prototypes to provide structural guiding information for masked area. The method also provides references for controlling local invariance in style transfer.

## 2 Related work

### 2.1 Generative adversarial networks

The generative adversarial network [5] is widely used in computer vision tasks such as image inpainting, image generation, etc. It is an effective model which consists of the discriminator and the generator to generate targets by the adversarial process. Deep convolutional GAN [15] replaces the multilayer perceptron by convolutional neural networks without the pooling in GAN. Cycle-consistent GAN [16] provides a way to learn the image translation without paired examples by training mirrored discriminators and generators between input and output images.

### 2.2 Image inpainting

Image inpainting, which means filling missing pixels of an eroded image, is a challenging problem in computer vision. Traditional approaches [17] typically solve the problem by matching and copying patches into holes from low resolution to high resolution, which are ineffective for recovering regions with complex structures such as faces. With the rapid development of generative adversarial networks [5], most recent studies [18–20] formulate image inpainting as a conditional image generation problem with both global and local discriminators contributing to adversarial losses. The global discriminator evaluates whether the generated image is coherent as a whole, while the local discriminator tries to enforce the local consistency within the generated region.

### 2.3 Chinese character inpainting

Compared with image inpainting, Chinese character inpainting is a new field to some extent. Famous calligraphers are required to copy the eroded characters in traditional methods. In recent years, inpainting has adapted to use the GAN instead of traditional methods to get better results. CGAN [9] has been used to remove the grids in the images to recover the Chinese characters. GAN with self-attention mechanism [21] is also used in Chinese character inpainting to acquire the recovery of the images and improve the effects to some extent. However, most inpainting networks need to provide location information of the masked part, and they cannot be used in the recovery of cultural relics at present. In our proposed method, the masked information is not necessary for the network, while prototype character is substituted for providing structure features.

## 3 Proposed method

In the field of image and character inpainting, GAN has been widely used as baseline network. Our model follows traditional methods and further expands the whole architecture. We believe that each character is constructed by structure and style together and the character inpainting task should aim to generate the masked content with same structure and style. Therefore, we additionally introduce a style encoder mapping network $E$ and an auxiliary recognition network $S$ to maintain the structure and style separately. We also encourage the style encoder $E$ to infer the proper style representation while training generator $G$ and discriminator $D$. In this case, the style encoder $E$ can extract style features of character images that do not appear in the training set and help the generator $G$ recover images without parameter retraining.

We propose our method in a unified architecture to generate the plausible content given masked input character $c_c$. Figure 2 shows the architecture. Due to the context-independent structure of Chinese characters, the lost information of the masked area cannot be obtained during the repair process. We decided to introduce a complete character image of discretionary styles but the same content as the prototype character $c_p$ to supplement the missing structural features. Suppose that the definition of $c_v$ is the concatenated vector of $c_c$ and $c_p$, while $X$ is the set of characters with the same font styles of $c_c$ in our task. Our goal can be formulated by recovering the $c_v$ under the constraint of the style code extracted by a character $c_s$ in $X$ through $E$.

In the following subsections, we first describe our network architecture and then its training objectives. We also discuss the expansion of our method in other fields.

### 3.1 Network architecture

Our model consists of the following four parts generator $G$, discriminator $D$, style encoder $E$ and structure extractor $S$. Compared with DCGAN, one of our baseline networks, we adjust the generator $G$ and introduce style encoder $E$ in order to extract character styles unobserved by the model. Moreover, we add a pretrained Chinese character recognition network as an auxiliary network to our model called structure extractor $S$ to encourage the generator $G$ to recover characters with structural integrity. The four parts work together to ensure the style and structure accuracy of the characters we recover.

**Generator G** The generator $G$ is made up of identity residual block [22] to maintain the high-level feature because of the importance of skip connection in feature preservation including the character skeleton. More specifically, we use four downsampling blocks, four intermediate blocks, and four upsampling blocks in our generator $G$ in Table 1 to generate the target characters $G(c_v, s)$ with the concatenated vector $c_v$ and the style code $s$. The style code $s$ from the style encoder $E$ represents the style of the target character and is injected into $G$ through adaptive instance normalization (AdaIN) [23], which is trainable rather than computing constant mean and variance. By this way, the $G$ can generate characters with new styles that may not be observed during training.

**Style encoder E** The style encoder mapping network $E$ in Table 2 takes as input a corresponding character $c_s$ under target font. Six residual blocks with ReLU are used to guarantee that our model can extract the proper style code $s = E(c_s)$ of $c_s$ and infer the style representation. This allows G to synthe-

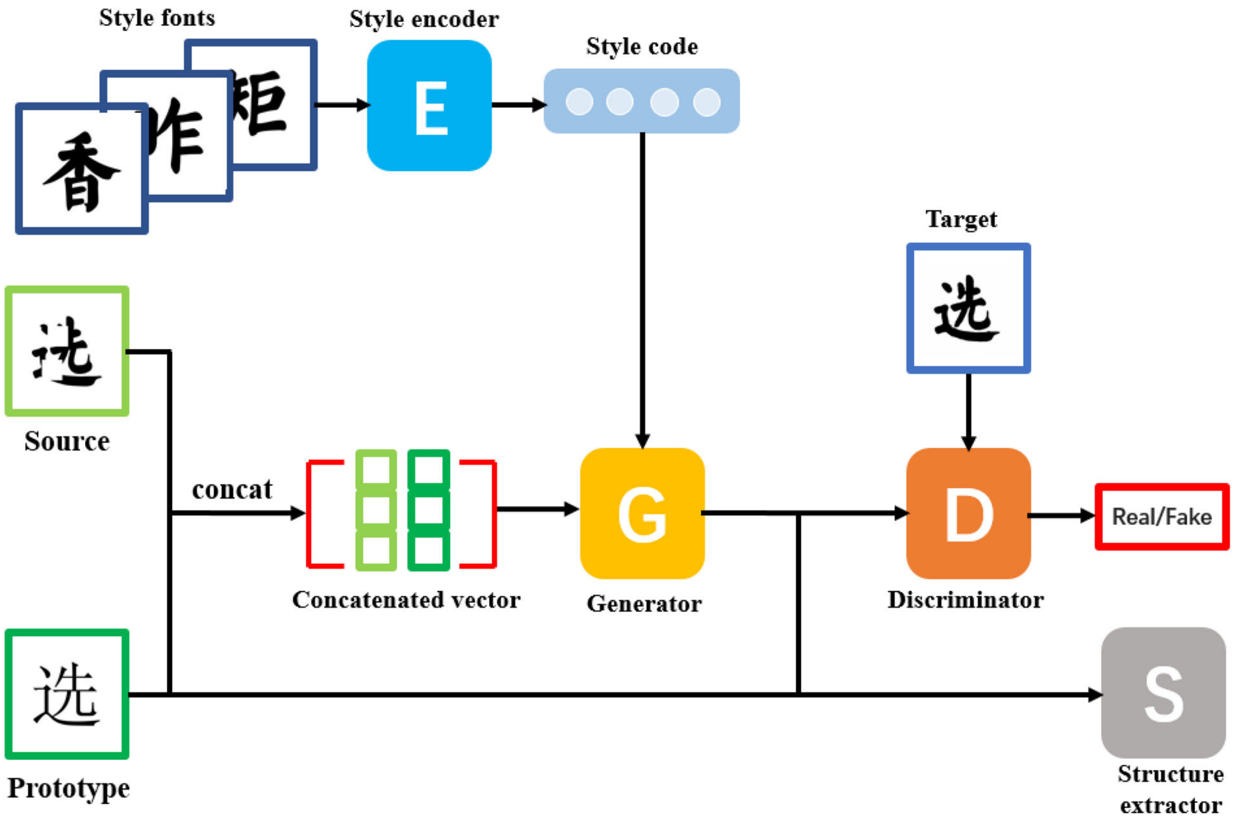

**Fig. 2** Our proposed framework mainly consists of four subnets, including a generator $G$, a discriminator $D$, a style encoding mapping network E and a pretrained structure extractor network S. The source character and prototype character are concatenated as the input of G. S extracts the style code and G combines the features to conduct the generation. The generation is labelled in D and sent to S with prototype character to extract the embedding difference. The embedding difference enforces the generator G to recover the characters with proper structure features

**Table 1** Generator architecture

| Layer | Resample | Norm | Output shape |
|---|---|---|---|
| Vector $x$ | – | – | 256×256×6 |
| Conv1×1 | – | – | 256×256×64 |
| ResBlk | AvgPool | IN | 128×128×128 |
| ResBlk | AvgPool | IN | 64×64×256 |
| ResBlk | AvgPool | IN | 32×32×512 |
| ResBlk | AvgPool | IN | 16×16×512 |
| ResBlk | – | IN | 16×16×512 |
| ResBlk | – | IN | 16×16×512 |
| ResBlk | – | AdaIN | 16×16×512 |
| ResBlk | – | AdaIN | 16×16×512 |
| ResBlk | Upsample | AdaIN | 32×32×512 |
| ResBlk | Upsample | AdaIN | 64×64×256 |
| ResBlk | Upsample | AdaIN | 128×128×128 |
| ResBlk | Upsample | AdaIN | 256×256×64 |
| Conv1 × 1 | – | – | 256×256×3 |

size an output character reflecting the style $s$ of a reference character $c_s$.

**Discriminator D** To distinguish between fake characters produced by the generator $G$ and target characters including the character and its style code, our discriminator $D$ adopts a similar network architecture to style encoder $E$.

**Structure extractor S**The character recognition network called structure extractor $S$ is used in our architecture to extract the skeleton of each character. The structure extractor in Table 3 consists of the ResNet [22] and a multilayer perceptron. We trained $S$ as an auxiliary network to keep the consistency of the generated font structure and the original. The character recognition network is trained with 3754 classes on CASIA-HWDB1.1 dataset [24]. The accuracy rate exceeds 0.96.

### 3.2 Training objectives

We describe our training objectives in terms of loss functions. For better recovery of character details and styles, we incorporate style reconstruction loss [25], cycle consistency loss [16], content loss [26], encoding diversity loss, structure loss and adversarial loss to train our model.

**Table 2** Style encoder and discriminator architectures

| Layer | Resample | Norm | Output shape |
|---|---|---|---|
| Character $x$ | – | – | 256 × 256 × 3 |
| Conv1 × 1 | – | – | 256 × 256 × 64 |
| ResBlk | AvgPool | – | 128 × 128 × 128 |
| ResBlk | AvgPool | – | 64 × 64 × 256 |
| ResBlk | AvgPool | – | 32 × 32 × 512 |
| ResBlk | AvgPool | – | 16 × 16 × 512 |
| ResBlk | AvgPool | – | 8 × 8 × 512 |
| ResBlk | AvgPool | – | 4 × 4 × 512 |
| LReLU | – | – | 4 × 4 × 512 |
| Conv4 × 4 | – | – | 1 × 1 × 512 |
| LReLU | – | – | 1 × 1 × 512 |
| Reshape | – | – | 512 |

**Table 3** Structure extractor architecture

| Layer | Norm | Activation | Output shape |
|---|---|---|---|
| Image x | – | – | 256 × 256 × 3 |
| IL | – | – | 128 × 128 × 1 |
| Conv | BN | ReLU | 63 × 63 × 32 |
| Conv | BN | ReLU | 61 × 61 × 32 |
| MaxPool | – | – | 30 × 30 × 32 |
| Conv | BN | ReLU | 28 × 28 × 64 |
| Conv | BN | ReLU | 26 × 26 × 64 |
| MaxPool | – | – | 13 × 13 × 64 |
| Conv | BN | ReLU | 11 × 11 × 128 |
| MaxPool | – | – | 5 × 5 × 128 |
| Conv | BN | ReLU | 3 × 3 × 128 |
| AvgPool | – | – | 1 × 1 × 128 |
| Flatten | – | – | 128 |
| Dense | – | – | 512 |
| Conv | BN | ReLU | 512 |
| Dense | – | – | 3755 |

**Adversarial loss** Given the masked character $c_c$, its prototype character $c_p$ and target character $c_t$, we randomly select a character $c_s \in X$ on the same font style of $c_c$ and concatenate $c_c$ and $c_p$ into $c_v$. The adversarial loss is defined by

$$\begin{aligned}\mathcal{L}_{\text{adv}} = {} & \mathbb{E}_{c_p}[\log D(c_p)] + \mathbb{E}_{c_t}[\log D(c_t)] \\ & + \mathbb{E}_{c_v,s_s}[\log(1 - D(G(c_v, s_s)))],\end{aligned} \tag{1}$$

where $D(c)$ is used to determine whether the character $c$ is real or fake. The style encoder $E$ learns to generate the target style code $s_t = E_(c_t)$, prototype style code $s_p = E_(c_p)$ and same font style code $s_s = E_(c_s)$ corresponding to $c_t$, $c_p$ and $c_s$. $G$ takes the concatenated vector $c_v$ and style code $s_s$ as input to generate an output character $G(c_v, s_s)$ which is expected to recover the character $c_c$.

**Content loss** To encourage the generator $G$ to generate more realistic images to pass the examination of the discriminator $D$, we employ a content loss

$$\mathcal{L}_{\text{con}} = \mathbb{E}_{c_v,c_t,s_s}[\| c_t - G(c_v, s_s) \|_1], \tag{2}$$

which enforces $G(c_v, s_s)$ to be near the ground-truth output $c_t$ in plane space sense.

**Encoding diversity loss** To enforce the style encoder $E$ to extract the correct style code $s_s$ from $c_s$ and infer the new style representation, we refer to the diversity sensitive loss [25] and modify it as the encoding diversity loss

$$\mathcal{L}_{\text{enc}} = \mathbb{E}_{c_c,s_s,s_p}[\| G(c_c, s_p) - G(c_c, s_s) \|_1]. \tag{3}$$

Maximizing the regularization term forces $G$ to explore the character image space and discover meaningful style features to recover diverse character images. The encoding diversity loss guarantees that the style encoder $E$ can extract appropriate style codes from different sources, even style codes that have never appeared in the training set.

**Style reconstruction loss** In order to further ensure the style encoder $E$ to extract an accurate style code $s_s$ from $c_s$ and the generator $G$ to utilize this style code, we employ the style reconstruction loss

$$\mathcal{L}_{\text{sty}} = \mathbb{E}_{c_v,c_s}[\| s_s - E(G(c_v, s_s)) \|_1]. \tag{4}$$

At test time, our style encoder $E$ allows $G$ to recover the character $c_c$, reflecting the style of a reference character.

**Cycle consistency loss** To encourage the generator $G$ to maintain the original content structure of character $c_p$ properly, we employ the cycle consistency loss

$$\mathcal{L}_{\text{cyc}} = \mathbb{E}_{c_v,c_p,s_s,s_p}[\| c_p - G(\dot{c_v}, s_p) \|_1], \tag{5}$$

where $\dot{c_v}$ is the concatenated vector of $G(c_v, s_s)$ and $c_p$. By reconstructing $\dot{c_v}$ with prototype style code $s_p$, G learns to maintain the content structure while changing the style code.

**Structure loss** Considering the difference among character of multiple fonts in terms of the space structure, we additionally introduce the structure loss

$$\mathcal{L}_{\text{str}} = \mathbb{E}_{c_v,c_p,s_s}[(S(G(c_v, s_s)) - S(c_p))^2], \tag{6}$$

where $S(c_x)$ denotes the recognition result of the character $c_x$ by the pretrained structure extractor $S$, expressed in the form of embedding. It encourages the generator $G$ to recover the original space structure via the recognition network.

**Full objectives** Combining all the loss functions together, our full objective can be expressed as the following equation

$$\min_{G,E}\max_{D} \quad \mathcal{L}_{\text{adv}} + \lambda_{\text{con}}\mathcal{L}_{\text{con}} - \lambda_{\text{enc}}\mathcal{L}_{\text{enc}} \\ + \lambda_{\text{sty}}\mathcal{L}_{\text{sty}} + \lambda_{\text{cyc}}\mathcal{L}_{\text{cyc}} + \lambda_{\text{str}}\mathcal{L}_{\text{str}}, \tag{7}$$

where $\lambda_{\text{con}}$, $\lambda_{\text{enc}}$, $\lambda_{\text{sty}}$, $\lambda_{\text{cyc}}$ and $\lambda_{\text{str}}$ are hyperparameters we provide to control their balance.

### 3.3 Discussion

We specifically discussed whether our method can be applied to the field of character style transfer. Compared with character inpainting, style transfer focuses more on global changes rather than local changes. Based on overall inpainting, we propose to combine the characters to be repaired with the prototype characters to achieve our goal. If the prototype character is regarded as the source character, we can consider providing a partial image of the target character to encourage the generator to generate characters with high similarity in this part. In this way, we can control local invariance and perform the style transfer in a specific direction to improve its effect. Our method provides a reference for style transfer in specific situations.

## 4 Experiments

### 4.1 Experimental settings

In this section, we present a new Chinese calligraphy dataset of various styles and conduct several experiments to show the effectiveness of our model. We compare the proposed model with leading baselines on our specific dataset and analyze the subjective quality and objective evaluation on the evaluation metrics.

**Dataset** We create our own dataset since there is no existing dataset containing a large number of Chinese calligraphy images. To obtain these images, we collect 2 calligraphy fonts: HYYanKaiW font [27] and HanyiSentyJournal [28] font, which are stored in TrueType format and converted them into independent high-quality images whose resolution is 256 × 256. Each font contains more than 10000 Chinese characters. We split each font into three parts: 1000 for training, 200 for evaluation, and 50 for generating the visualization results which model never seen. We process Songti font as our prototype font in the same way. For training, we generate several number of 50 × 50 square masked at random locations for each images to simulate damage to the characters.

**Baselines** We use traditional method [29], CGAN, DCGAN, CycleGAN, Self-attention [21] and EdgeConnect [30] as our baselines. It should be observed that inpainting methods usually work in two ways, one is concerned little about the location of the masked, such as DCGAN, CycleGAN and our model, while the others strongly require the location of masked area, especially EdgeConnect, which relies on edge detection. As a result, we choose the above three networks as our baselines and follow the guidance of the authors to train these models on our dataset.

**Evaluation metrics** To measure the quality of recovering, we calculate the mean absolute error (MAE) and peak signal-to-noise ratio (PSNR) as evaluation indicators. In addition, considering that Chinese character images have higher requirements for structure, we use the structural similarity index (SSIM) [31] to evaluate the consistency of the font structure.

**Training Details** We set $\lambda_{\text{con}} = 10$, $\lambda_{\text{enc}} = 2$, $\lambda_{\text{sty}} = 1$, $\lambda_{\text{cyc}=1}$ and $\lambda_{\text{str}} = 10$ for our models. All training images are resized and cropped to 256 × 256. We train the model with a batch size of 8, and the training takes 20 minutes for each epoch on our dataset. The training time to achieve an acceptable result is 46 hours, and for each epoch, we spend 30 seconds on testing our model. All the experiments are conducted with Intel Core Processor CPU and 16GB NVIDIA Tesla V100 GPU.

### 4.2 Qualitative comparisons

We train those competing methods with (the Chinese calligraphy character set) and randomly select several masked characters that are representative of content and glyphs. Figures 3 and 4 show the results of our model and baselines corresponding to the two distinctive fonts. All images are shown at the same resolution (256× 256) and directly output by the models without post-processing.

As shown in Fig. 3, CGAN and DCGAN just learn to recover part of the character structure of the occluded area and generate missing content with irrelevant repair noise and distortion. CycleGAN obviously produces the wrong content of masked areas. Self-attention and EdgeConnect generate more ideal results, but it still loses the character structure and style of the center masked area because of the edge detection. Our results show that our method generates characters of higher quality while preserving the full calligraphy style and structural features.

As shown in Fig. 4, the loss of strokes on CGAN and DCGAN becomes more pronounced. CycleGAN attempts to recover the masked area but is still far away from acceptable results. Traditional method repairs wrong character structure. Because of the distinctive font which widens the distance between strokes and causes, it is harder to get enough patching information from the edge, self-attention and EdgeConnect generates results with loss of the center strokes. Our

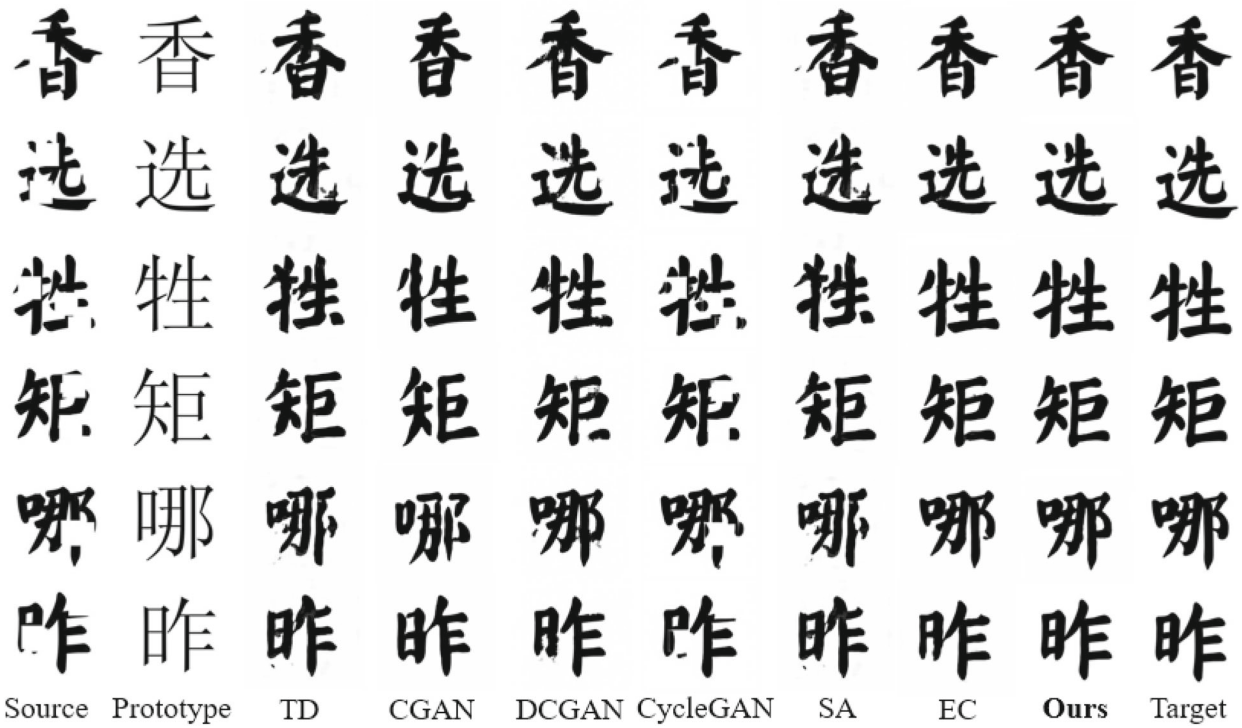

**Fig. 3** Qualitative comparison results on recovering HYYanKaiW font characters. The first and second columns are the source and prototype characters. The prototype characters we choose are Songti font characters, which is widely used in Chinese periodicals or magazines. The following six columns are the results of our baselines and the TD, SA, EC means traditional, self-attention and EdgeConnect method. The ninth column is generated by our network, which is highly similar to the target characters in the sixth column. The results show our model generate characters of better visual quality than our baselines

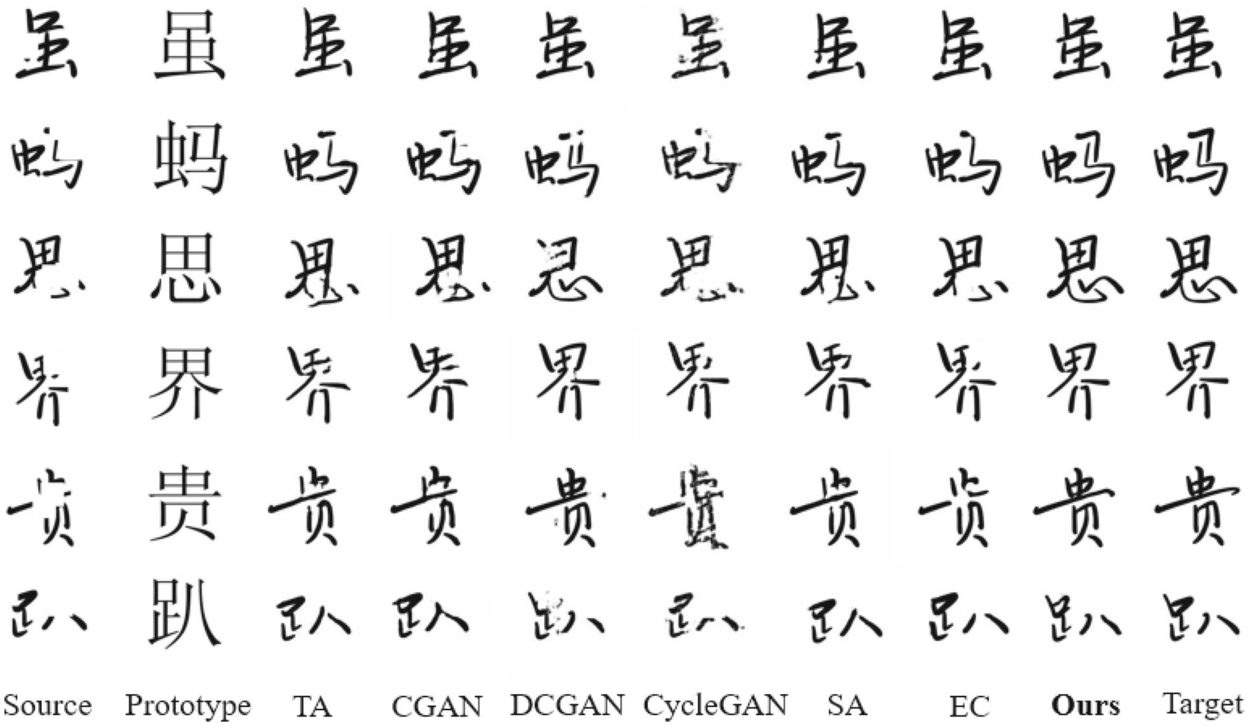

**Fig. 4** Qualitative comparison results on recovering HanyiSentyJournal font characters. The arrangement is the same as Fig. 3. Compared with baselines, our model can recover character images more completely and accurately

**Table 4** Quantitative comparisons on the dataset of two fonts. We define characters written in HYYanKaiW font as Data-1 and characters written in HanyiSentyJournal font as Data-2. Our model is obviously superior to all the baselines

| Method | MAE | | PSNR | | SSIM | |
|---|---|---|---|---|---|---|
| | Data-1 | Data-2 | Data-1 | Data-2 | Data-1 | Data-2 |
| TD | 7.92 | 9.02 | 15.59 | 15.83 | 0.92 | 0.93 |
| CGAN | 11.25 | 10.82 | 13.32 | 10.22 | 0.82 | 0.87 |
| DCGAN | 8.11 | 9.42 | 14.15 | 13.23 | 0.85 | 0.88 |
| CycleGAN | 13.26 | 12.87 | 13.86 | 14.38 | 0.83 | 0.85 |
| SA | 7.62 | 8.60 | 16.81 | 15.70 | 0.91 | 0.87 |
| EC | 7.06 | 5.36 | 17.23 | 18.74 | 0.93 | 0.96 |
| **Ours** | **2.35** | **3.17** | **26.40** | **23.72** | **0.97** | **0.98** |

model is trained to generate the ideal outputs with better consistency compared with the three state-of-the-art models.

## 4.3 Quantitative comparisons

We also investigate the effectiveness of the baselines and our proposed model on three evaluation metrics and provide a quantitative comparison in Table 4. Each method generates the output character corresponding to the same input character.

As shown in Table 4, our model is superior to all the baselines and achieves better or comparable results by a margin of the quantitative analysis of the output Chinese calligraphy characters. Compared to the baselines, our model produces decent results with the best MAE, PSNR and SSIM on the two datasets.

## 4.4 Combination with prototype characters

We evaluate our method under different combinations of concatenated vectors, which can effectively prove the superiority of our combination and the variability of the prototype characters. As shown in Fig. 5, the combination between the erosion characters and prototype characters is obviously better than the others. Besides, the replacement of the style of the prototype characters does not significantly affect the repair results. Table 5 also proves this conclusion.

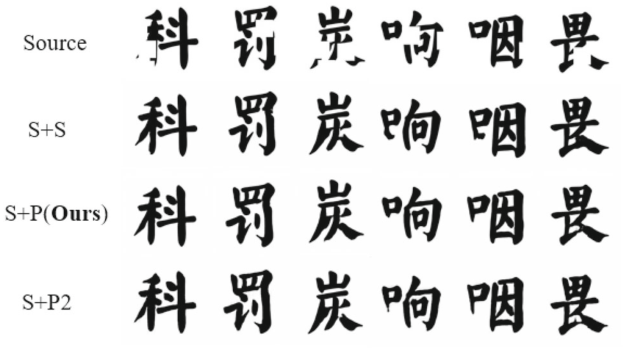

**Fig. 5** Experiment results of different combinations. The S + S in the second row means the vector which is combined by source character and source character, which shows obvious repair error on the structure. The S + P means that the vector is combined by source character and prototype character as our method. The S + P2 in the fourth row means that we replace the Songti font with HanyiSentyJournal font as the prototype character to form the vector

**Table 5** Quantitative comparisons of different combinations. The S+S stands for the vector which is combined by source character and source character. The S+P stands for the vector which is combined by source character and prototype character as our method. The S+P2 means that we replace the Songti font with HanyiSentyJournal font as the prototype character to form the vector

| Combination method | MAE | PSNR | SSIM |
|---|---|---|---|
| S + S | 4.5255 | 20.7322 | 0.9431 |
| S + P(ours) | **2.3511** | 26.4051 | **0.9729** |
| S + P2 | 2.3652 | **26.5512** | 0.9685 |

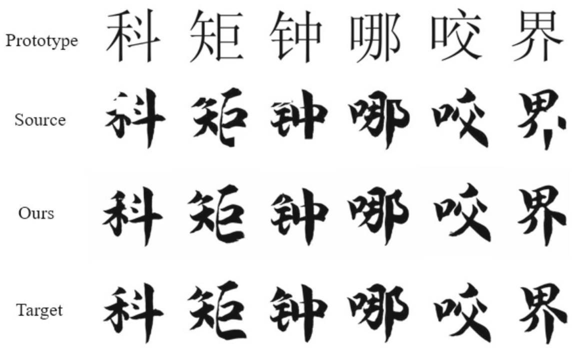

**Fig. 6** Experiment on characters of new style. The results show that our method manages to recover stylized characters that did not appear in the training set

## 4.5 Ability to recover characters of new style

As highlighted before, our model is widely adaptable and can recover characters of new style without parameter retraining. We save the previous model in the comparison experiment and change the characters of new style, HYShangWeiHeFeng font, which never appear in the training set, as the input to the model [32]. As shown in Fig. 6, the results prove that our model exhibits considerable effect on the new style characters without retraining.

## 4.6 Ablation study

We further conduct two ablation experiments for our model to prove the effectiveness of our network architecture and loss functions.

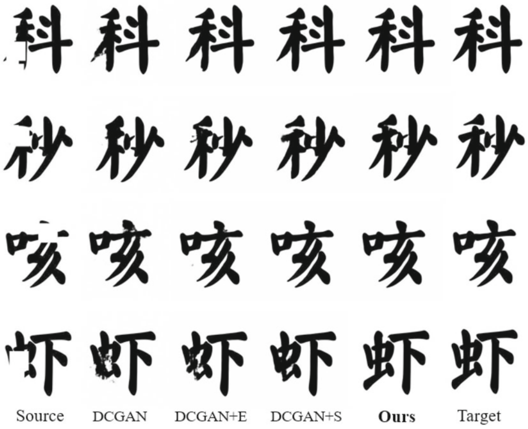

**Fig. 7** Architecture ablation experiment results on recovering HYYanKaiW font characters. E means Style Encoder and S means Structure Extractor. The best inpainting results can only be generated when both networks are in effect

**Architecture ablation experiment** In order to guarantee that the style encoder mapping network and the auxiliary recognition network play important roles in our network architecture, we specifically investigate different combinations of networks with our baseline DCGAN and train them under the same conditions. Figure 7 and Table 6 show the qualitative and quantitative comparisons between methods. As shown in Fig.7, the DC+E generates the inpainting character with significant artifacts and absence of some strokes. The DC+S basically recovers the structure of glyph, but there are still some differences between the results and the targets in style features such as the thickness of a stroke and so on.

**Loss ablation experiment** We do further experiments on loss functions. We remove the influence of the specific loss function by setting its multiplier to zero, and the results are shown in Table 7. It is evident from the table that each loss function is necessary and makes a contribution to recover the eroded character images. All loss functions work together to make our network recovering Chinese character images with high confidence.

## 4.7 Structure wandering

To prove the veracity that we extract the structure of the Chinese calligraphy character through our recognition network model, we extract the feature vector of the highest dimension from the recognition network and label them with original characters in the two-dimensional space as illustrated in Fig. 8. It is obvious that the Euclidean distance of characters with similar glyphs in the two-dimensional space is closer, which indicates the effectiveness of our recognition network.

**Table 6** Quantitative comparisons on recovering HYYanKaiW font characters. E represents Style Encoder while S represents Structure Extractor

| Method | MAE | PSNR | SSIM |
|---|---|---|---|
| DCGAN | 7.0678 | 17.239761 | 0.9283 |
| DCGAN+E | 5.0026 | 20.1667 | 0.9391 |
| DCGAN+S | 5.4556 | 18.6915 | 0.9377 |
| **Ours** | **2.3511** | **26.4051** | **0.9729** |

**Table 7** Loss ablation experiment on HYYanKaiW font. The result shows that each loss function is necessary

| Method | MAE | PSNR | SSIM |
|---|---|---|---|
| Ours-ConLoss | 4.7372 | 20.4001 | 0.9405 |
| Ours-CycleLoss | 4.7774 | 20.4352 | 0.9404 |
| Ours-AdvLoss | 4.7982 | 20.4331 | 0.9401 |
| Ours-StyleLoss | 4.6148 | 20.6518 | 0.9429 |
| **Ours** | **2.3511** | **26.4051** | **0.9729** |

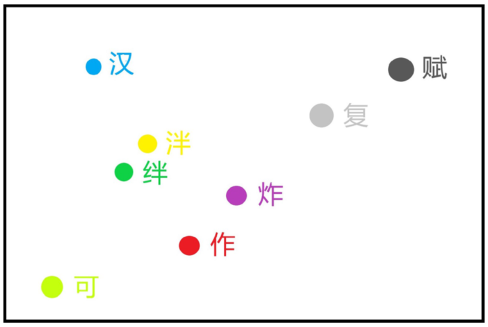

**Fig. 8** Structure feature of the highest dimension labeled in the two-dimensional space

## 4.8 Universality and limitations

**Random masked area experiment** To expand the repairable range of our model and show the universality, we modify source images by making the missing area which is larger or not rectangular. Figure 9 shows the results of this experiment. It is obvious that our model can still achieve an acceptable effect Whether we change the shape of our masked area or expand it. However, when the masked area is too large, our effects will be reduced to a certain extent as shown.

**Real-eroded Chinese characters experiment** We further test our model with real-eroded Chinese characters images. We select images segmented from the rubbing from stone inscriptions as the input to our trained model. As shown in Fig. 10, our method basically repairs most eroded areas on real Chinese character images with irregular eroded regions, but it will fail sometimes due to the wrong distinction between noise and strokes.

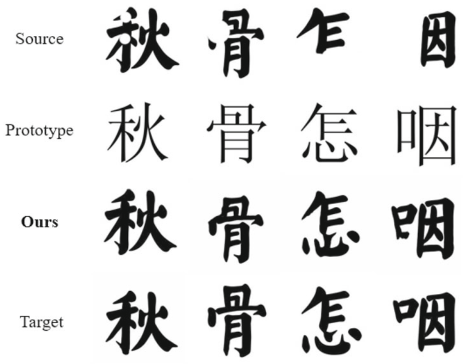

**Fig. 9** Results of sources which are eroded by irregular shapes. The first two columns are eroded by circle noises, while the last two have whole components missed

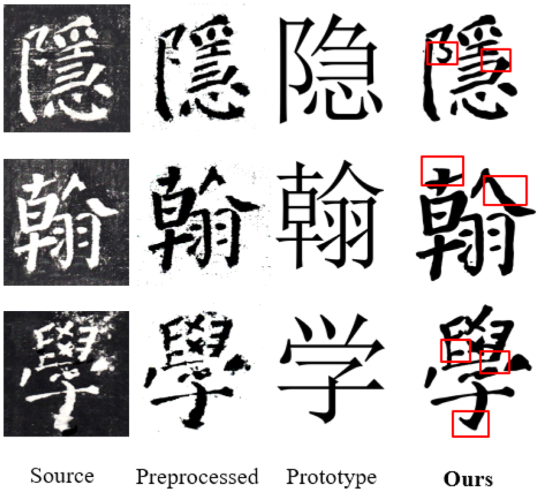

**Fig. 10** Results on real Chinese character images with irregular eroded regions. The second column shows the preprocessed figures whose noises are removed. Red rectangles have highlighted the repaired stokes

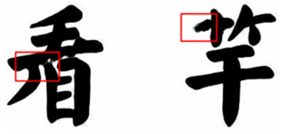

**Fig. 11** Failure examples with stroke collapse marked by red rectangles

**Limitation analysis** We also find that when the density of strokes in the masked area is too high or there is no obvious structural difference between the content to be repaired and the target one, our generated image will appear to be distorted or lost in Fig. 11. We discuss and analyze the reasons for this situation. We think that the thickness of the strokes due to its style causes our discriminator and structure extractor to misjudge the authenticity of the generated image. It cannot be avoided under the current network structure. However, this situation is not common and we will try to overcome it in subsequent experiments.

## 5 Conclusion

In this paper, we propose a novel style-preserving architecture that is capable of inpainting character images without parameter retraining. We ensure the inpainting consistency of style between the masked part and the rest by providing a more effective recovering idea based on overall inpainting instead of edge recovery. Intact character images of discretionary styles as prototypes are introduced to provide structural guiding information for masked area. Comparative experiments with other state-of-the-art methods in image inpainting prove our superiority with respect to both generalizability.

## Declarations

**Conflict of interest** We declare that we have no conflict of interest with other people or organizations.

## References

1. 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. **38**(1), 1–18 (2018)
2. Lian, Z., Zhao, B., Xiao, J.: Automatic generation of large-scale handwriting fonts via style learning. In: SIGGRAPH Asia 2016 Technical Briefs, pp. 1–4. (2016)
3. Wang, X., Liang, X., Hu, J., Sun, L.: Stroke-based Chinese character completion. In: 2012 Eighth International Conference on Signal Image Technology and Internet Based Systems, pp. 281–288. IEEE (2012)
4. Xu, S., Jin, T., Jiang, H., Lau, F.C.M.: Automatic generation of personal Chinese handwriting by capturing the characteristics of personal handwriting. In: Twenty-First IAAI Conference. Citeseer (2009)
5. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., et al.: Generative adversarial networks. arXiv preprint arXiv:1406.2661 (2014)
6. Ye, C., Guan, W.: A review of application of generative adversarial networks. J. Tongji Univ. Nat. Sci. **4**(48), 591–601 (2020)
7. Liu, M.-Y., Huang, X., Mallya, A., Karras, T., Aila, T., Lehtinen, J., Kautz, J.: Few-shot unsupervised image-to-image translation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2019)
8. Li, J., Song, G., Zhang, M.: Occluded offline handwritten Chinese character recognition using deep convolutional generative adversarial network and improved GoogLeNet. Neural Computing and Applications (2020)
9. Zhong, Z., Yin, F., Zhang, X-Y., and Liu, C-L.: Handwritten Chinese character blind inpainting with conditional generative adversarial nets. In: *2017 4th IAPR Asian Conference on Pattern Recognition (ACPR)*, pp. 804–809. IEEE (2017)
10. Mao, X., Li, Q., Xie, H., Lau, R.Y.K., Wang, Z., Paul Smolley, S.: Least squares generative adversarial networks. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (2017)
11. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C.: Improved training of wasserstein gans. Adv. Neural Inf. Process. Syst. **30**, 5767–5777 (2017)

12. Gao, Y., Jiangqin, W.: Gan-based unpaired Chinese character image translation via skeleton transformation and stroke rendering. Proc. AAAI Conf. Artif. Intell. **34**, 646–653 (2020)
13. Zhang, Y., Zhang, Y., Cai, W.: A unified framework for generalizable style transfer: Style and content separation. IEEE Transactions on Image Processing (2020)
14. Jiang, Y., Lian, Z., Tang, Y., Xiao, J.: Scfont: structure-guided Chinese font generation via deep stacked networks. Proc. AAAI Conf. Artif. Intell. **33**, 4015–4022 (2019)
15. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
16. 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, pp. 2223–2232 (2017)
17. Ballester, C., Bertalmio, M., Caselles, V., Sapiro, G., Verdera, J.: Filling-in by joint interpolation of vector fields and gray levels. IEEE Trans. Image Process. Publ. IEEE Signal Process. Soc. **10**, 1200–1211 (2001)
18. Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., Huang, T.S.: Generative image inpainting with contextual attention. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR), pp. 5505–5514 (2018)
19. Wei, Y., Liu, S.: Domain-based structure-aware image inpainting. Signal Image Video Process. **10**(5), 911–919 (2016)
20. Li, S., Zhao, M.: Image inpainting with salient structure completion and texture propagation. Pattern Recognit. Lett. **32**(9), 1256–1266 (2011)
21. Song, G., Li, J., Wang, Z.: Occluded offline handwritten Chinese character inpainting via generative adversarial network and self-attention mechanism. Neurocomputing **415**, 146–156 (2020)
22. He, K., Zhang, X., Ren, S., Sun, J.: Identity mappings in deep residual networks. In: European Conference on Computer Vision(ECCV), pp. 630–645. Springer International Publishing, Cham (2016)
23. Huang, X., Belongie, S.: Arbitrary style transfer in real-time with adaptive instance normalization. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV) (2017)
24. Liu, C.-L., Yin, F., Wang, D.-H., Wang, Q.-F.: Casia online and offline Chinese handwriting databases. In: Proceedings of the International Conference on Document Analysis and Recognition, ICDAR, pp. 37–41 (2011)
25. Choi, Y., Choi, M., Kim, M., Ha, J.-W., Kim, S., Choo, J.: Stargan: unified generative adversarial networks for multi-domain image-to-image translation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR), pp. 8789–8797. (2018)
26. 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(CVPR), pp. 1125–1134. (2017)
27. Hanyi fonts:hyyankaiw font. http://www.hanyi.com.cn/productdetail.php?id=2616&type=0
28. Hanyi fonts:hanyisentyjournal. http://www.hanyi.com.cn/productdetail.php?id=3406&type=0
29. Sun, J., Yuan, L., Jia, J., Shum, H.-Y.: Image completion with structure propagation. In: ACM SIGGRAPH 2005 Papers, pp. 861–868. (2005)
30. Nazeri, K., Ng, E., Joseph, T., Qureshi, F., Ebrahimi, M.: Edgeconnect: structure guided image inpainting using edge prediction. In: 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), pp. 3265–3274. IEEE (2019)
31. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004)
32. Hanyi fonts:hyshangweihefeng font. http://www.hanyi.com.cn/productdetail.php?id=7109&type=0

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Haolong Li** is a BEng student of Tongji University. His research interests include style transfer, image inpainting and pattern recognition.

**Zizheng Zhong** is an undergraduate student of Tongji University. His research interests include image processing and federated learning.

**Wei Guan** is a master student of Tongji University. He received his Bachelor's degree in computer technology from Tongji University. His research interests include image generation and video understanding.

**Chenghao Du** is an undergraduate student majoring in information security of Tongji University. His research interests include computer vision and font generation.

**Yuxiang Wei** is a junior student majoring in computer science and technology at Tongji University. His research interests mainly lie in the intersection between software engineering and artificial intelligence, including testing deep learning systems and deep learning compilers, improving the robustness of deep learning models, etc.

**Yu Yang** is an undergraduate student of Tongji University. His research interests include the style translation and molecule generation. He is also interested in knowledge extraction and text mining.

**Chen Ye** is a researcher in the College of Electronic and Information Engineering at Tongji University. He received his B.S in Automation from Tongji University, and his PhD in computer science from Tongji University. His research interests include machine learning, image processing, big data analysis and its application in the field of industrial intelligence.