Components Regulated Generation of Handwritten Chinese Text-lines in Arbitrary Length

Shuo Li[1,2], Xiyan Liu[1,2], Gaofeng Meng*[1,2,3],Shiming Xiang[1,2], Chunhong Pan[1]
[1] National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences
[2] School of Artificial Intelligence, University of Chinese Academy of Sciences
[3] CAS Centre for Artificial Intelligence and Robotics, HK Institute of Science and Innovation
{lishuo2020}@ia.ac.cn,{xiyan.liu, gfmeng, smxiang, chpan}@nlpr.ia.ac.cn

***Abstract*—Generating readable images of handwritten Chinese text-lines is very challenging due to complicated topological structures in Chinese. To address this problem, we propose a components regulated model named HCT-GAN to generate the entire lines of Chinese handwriting from text-line labels. Specifically, HCT-GAN is designed as a CGAN-based architecture that additionally integrates a Chinese text encoder (CTE), a sequence recognition module(SRM), and a spatial perception module (SPM). Compared with the one-hot embedding, CTE learns the latent content representation by reusing the structure and component embedding shared among the Chinese characters. SRM provides sequence-level constraints to the generated images. SPM can adaptively constrain the spatial correlation between the generated components, which facilitates the modeling of characters with complicated topological structures. Benefiting from such artful modeling, our model suffices to generate images of handwritten Chinese text-lines in arbitrary length. Extensive experimental results demonstrate that our model achieves state-of-the-art performance in handwritten Chinese lines generation.**

I. Introduction

With the development of deep learning, the automatic generation of the Chinese character has increasingly attracted attention among the research community [1]–[17]. Nevertheless, the generation of handwritten Chinese text-lines still remains underexplored. Compared with the alphabetic system of writing such as English, Chinese has a huge amount of ideographic characters(e.g., as many as 70244 in GB180102005 standard) with complicated shapes. Prior work only focuses on the above challenge of generating single characters. But the generation of handwritten Chinese text-lines has the following additional challenges. (i) handwritten Chinese lines contain line-level features, such as the adhesions between adjacent characters. (ii) There are subtle differences in the same characters under the influence of neighbors. Currently, the images of handwritten Chinese lines with an arbitrary number of characters are acquired by stacking single-character images. The method typically lacks a realistic-looking handwriting style for failing to solve the above two critical challenges. To mimic the natural writing process, our work aims to alleviate the problem by directly generating images of handwritten Chinese text-lines with arbitrary lengths(see Figure 1).

*Corresponding author.

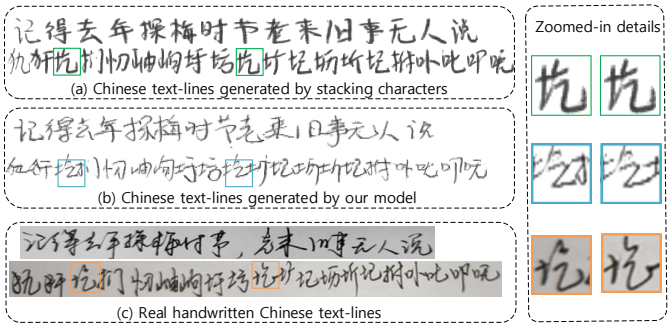

Figure 1. Examples of the text-line images. For real handwritten lines, the same characters are slightly different under the influence of neighbors, and there may be an adhesion between neighbors. The enlarged handwriting details are shown on the right column.

Recent Chinese character generation are treated as an image-to-image translation problem via learning to map the source-style to a target-style [12]–[17], but only single character images can be generated. Handwriting text generation (HTG) [18] is originally proposed for the system of alphabetic writings in which it has the nature of modeling the relationship between adjacent characters. The current alphabetic system of writing can not be used to generate higher-quality handwritten Chinese text-lines. The reason is the large vocabulary with complicated shapes not only, but the reuse mechanism of components in Chinese. The mechanism will bring the problem of similar character interference. To acquire realistic-looking handwriting, we propose a model named HCT-GAN that can generate handwritten Chinese lines with specified content and various styles in an end-to-end manner.

Unlike prior work [12]–[17], we only take text-line labels as the network input, instead of the images of reference characters. Due to the large vocabulary in Chinese, one-hot embedding may lead to an explosive growth of model parameters. To address this issue, an components regulated Chinese text encoder (CTE) is introduced to replace one-hot embedding. By reusing the structure and component embedding shared among the Chinese characters, CTE enables each character to be transformed to a combination of embedding vectors. Compared with the one-hot embedding, CTE is a more informative encoding method. Benefiting from CTE,

our model can generate Chinese text-lines containing unseen characters via combining embedding vectors. For alphabetic writing, it is a breeze to create a make-up word(e.g., sleeping and boring may be sloring). In the Internet community, people express their feelings by creating make-up words, which can attract more attention in social media. However, make-up Chinese glyphs can only rely on manual drawing, which increases the burden of creators. Like unseen characters, nonsense make-up glyphs also can be automatically generated.

The alphabetic model of writing generally uses a 1-D character recognition network to induce legibility. Considering large ideographic characters, we no longer treat a Chinese line as a 1-D character sequence, but as a 2-D ordered combination of components. Thus, we propose a sequence recognition module(SRM) to predict 1-D component sequences. To capture the spatial correlation between components, a spatial perception module (SPM) is introduced. SPM performs 2-D predictions to guide the generator to adaptively learn the spatial correlation between the internal components of lines, which facilitates the modeling of characters with complicated shapes.

In addition, augmenting data in a generative fashion may potentially boost handwritten text recognition. We conduct experiments to demonstrate that data augmentation using HCT-GAN is better than only warping the training images. Extensive qualitative and quantitative experiments are performed on challenging datasets, demonstrating HCT-GAN achieves the state of the arts. Due to nefarious uses of forgery handwriting technology, our proposed model does not aim to the specific writing styles.

To sum up, the contributions of this work are threefold:

- We propose a generalized Chinese text generation network, which can generate images of text in arbitrary length from text-line labels. This work is the first one directly generating images of handwritten Chinese lines.
- We introduce a Chinese text encoder(CTE) and spatial perception module (SPM). The former enables our model to generate text-lines containing unseen characters and nonsense glyphs through a component-structure-based coding method. The latter imposes content constraints at the fine-grained level through 2-D prediction.
- Finally, we improve text recognition performance by 4.37%, using the HWDB2.2 [19] dataset extended with generated lines compared to using only affine augmentation. Our study also discover that using nonsense glyphs to extend the HWDB1.1 [19] dataset is better than equivalent generated character.

II. Related Work

Chinese character generation. Generally speaking, existing Chinese character generation methods can be classified into two categories: component assembling-based methods and deep learning-based methods. The assembling-based method regards a character as a combination of strokes or components, which first extract strokes from character samples, and then some strokes are selected and assembled into unseen characters by reusing parts of a character [20]–[23].

After the generative adversarial networks(GANs) [24] was proposed, its derivative version [25]–[27] was widely adopted for style transfe [28]–[33]. Several attempts have been recently made to model font synthesis as an image-to-image translation problem [1]–[12], [34], [35], which transforms the image style while preserving the content consistency.

Handwritten text generation. Since the high inter-class variability of text styles from writer to writer and intra-class variability of same writer styles [36], the handwritten text generation is challenging.

At present, handwritten text generation mainly focuses on alphabetic writing. Alonso et al. [37] proposed an offline handwritten text generation model for fixed-size word images. ScrabbleGAN [38] used a fully-convolutional handwritten text generation model, which produces word images conditioned on a letter string and applies a character recognizer to constrain the text content. For handwritten Chinese text generation, the existing text generation model can not generate readable content. In contrast, our method is applicable for images of handwritten Chinese lines with arbitrary length.

III. Methodology

Chinese is a highly structured ideograph, completely different from alphabetic writing. Our approach exploits the structured properties to decompose the Chinese text. Meanwhile, two component-level recognition networks encourage high-quality images.

A. Overview

In this paper, we take the sampled latent priors from Gaussian distribution as style vectors and then select the text labels as inputs to generate handwritten Chinese text-lines with specified contents and diverse styles. Figure 2 shows an overview of the proposed model, mainly consisting of five modules: a Chinese text encoder(CTE), a generator G, a discriminator D, a sequence recognition module(SRM), and a spatial perception module(SPM). Given a Chinese text, it is queried in the dictionary CS to obtain corresponding component and structure indexes. Then, the indexes pass through CTE to obtain content representation e. Later, e concatenated by noise z_1 is fed into G to generate images of handwritten Chinese lines(i.e., fake). At the same time, the indexes are provided for SPM and SRM(not pictured in Figure 2) as labels.

B. Chinese text encoder(CTE)

Previous alphabetic text generative models [37], [38]usually cast each category as a one-hot embedding. Those methods work well for some alphabetic writing but fail to solve the Chinese text generation with complicated shapes. Due to the large vocabulary in Chinese, one-hot embedding may lead to an explosive growth of model parameters. We propose the dictionary CS to handle this problem.

Since the same components appear repeatedly in various characters, the category of components and structures is much smaller than characters. Exploiting the property, we build a dictionary CS to decompose each character of text-lines into

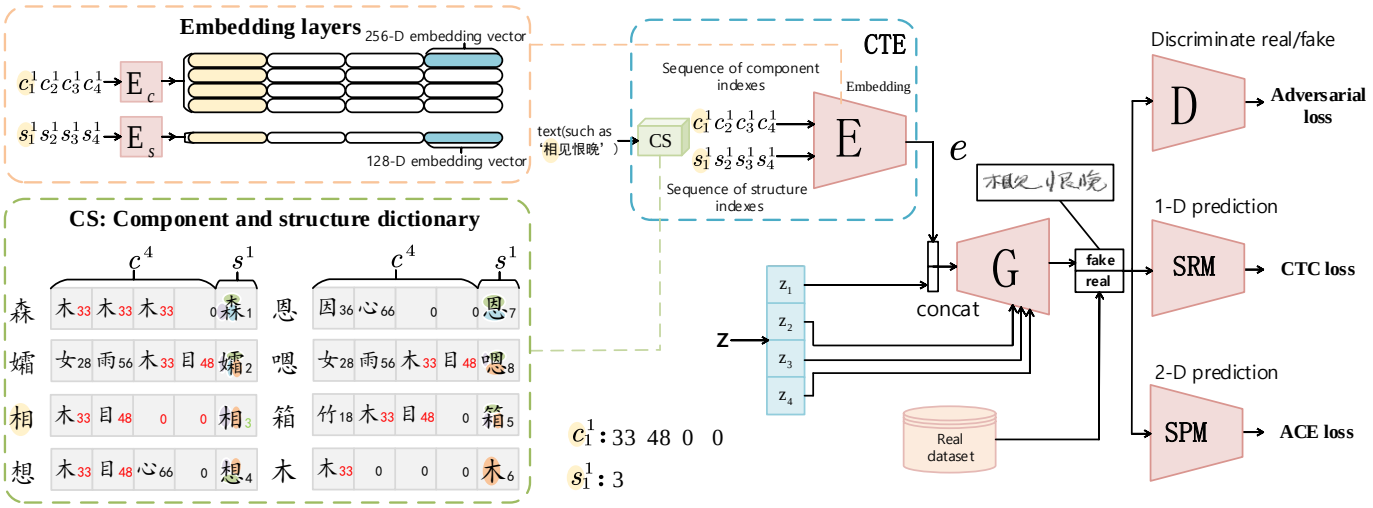

Figure 2. Left: Illustrates of Chinese text encoder(CTE). The same indexes share the same embedding. c_1^4 and s_1^1 shows the details of the encoding method. Right: Overview of the proposed method at training. Given a Chinese text(e.g., the four characters), it was queried in the dictionary CS to obtain corresponding component and structure indexes. Then, the sequences pass through embedding layers E to obtain content representation e. Later, e concatenated by noise z_1 was fed into the generator G to generate images of handwritten Chinese lines(i.e., fake). The multi-scale nature of the fake is controlled by additional noise vectors,z_2, z_3 and z_4 fed to G each layer through Conditional Batch Normalization (CBN) layers [39]. Fake and real are transmitted to the discriminator D, the spatial perception module(SPM), and the sequence recognition module(SRM), which respectively correspond to adversarial loss, CTC loss, and ACE loss.

separate component sequence indexes c^4 and structure index s^1. Taking c_1^4 as an example, we convert the variable-length component sequence to a length of four c_1^4 by padding. Figure 2(left) shows a few examples. We define 1179 components and 31 structures(e.g.,single-element, left-right, top-bottom, left-middle-right and top-middle-bottom) separated from the Chinese characters in GB2312 standard. Given a text with length l, we use the dictionary to obtain its component sequence indexes $(c_1^4, c_2^4, ..., c_l^4)$ and structure sequence indexes $(s_1^1, s_2^1, ..., s_l^1)$. Where l is length of the text.

Chinese text encoder(CTE) learns the embedding vector of each index, and the same index shares the same embedding vector. Each character corresponds to a fix-dimension(i.e.,$4 \times 256D + 128D$) combination of embedding vector, and the latent text-line representation e is the splicing of the combination in the width direction. Figure 2(left) shows some details. e related to text-length is used for generating the text-line images in arbitrary length, which helps to mimic the writing process that the components are only connected with surrounding components, instead of using a recurrent network to learn the coupling fixed-length embedding vector [13]. As a component-structure-based coding method, CTE enables our model to generate text-lines containing unseen characters and nonsense glyphs(more details can be found in supplementary material).

C. SPM and SRM

The alphabetic text generation model generally uses a 1-D character recognition network to evaluate the content of the generated images. Due to the huge number of Chinese characters with complicated shapes, only using a 1-D character recognizer to predict each category is not enough to obtain high-quality images. The reuse mechanism of Chinese components also brings the problems of similar Chinese character interference and information redundancy. Finer-grained supervision leads to clearer text, so we no longer regard images of Chinese lines as a 1-D character sequence, but as a 2-D ordered combination of components.

Sequence recognition module(SRM) is a 1-D component sequence recognition network with CTC loss [40] to induce basic legibility. SRM consists of a feature extractor based on a convolutional neural network, followed by a max-pooling on the vertical dimension and a full connection layer. Using SRM alone is not enough to generate realistic-looking handwritten Chinese lines. The reason is that the SRM is essentially a 1-D sequence prediction system, which is difficult to capture the spatial correlation of components of the text. A similar situation is that the performance of CRNN [41] based system is insufficient in the irregular text recognition.

Meanwhile, a spatial perception module(SPM) is introduced to guide the generator to capture the spatial correlation between the internal components of text lines(e.g., finer strokes and offset of the writing process). SPM is inspired by inexact supervision where the training data are given with only coarse-grained labels. We treat it as a non-sequence 2-D component prediction problem with text-level annotations only. SPM is composed of a full convolution network with an attention module, followed by a full connection layer trained on ACE [42] loss. Via implicitly constraining the component in the 2-D feature map, SPM guides to generate the corresponding component at the appropriate location.

Note that most recognition models use a recurrent network, typically bidirectional LSTM [41], [43], which may predict characters based on linguistic context instead of clear character shapes. In contrast, SRM and SPM only use local visual

features for character recognition and therefore provide better optimization direction for generating texts.

D. Generator G and Discriminator D

G is inspired by SAGAN [26], but differs in architecture and receives the variable-length tensor as input. Some common module are used, such as self-attention mechanism [44] to refine local area image quality, spectral normalization [45], hinge loss function [46] to stabilizes the training and full convolution network(FCN) [47] [36], [38]. To cope with variable-length images, D is also an FCN structure and performs global average pooling. The pooling layer aggregates scores from the variable-length feature map into the final output.

E. Loss functions

We implement the hinge version of the adversarial loss from Geometric GAN [46].

$$\begin{aligned} L_G &= -\mathbb{E}_{\boldsymbol{z}\sim p_z, \boldsymbol{e}\sim p_{text}}[D(G(\boldsymbol{z},\boldsymbol{e}))] \\ L_D &= +\mathbb{E}_{(\boldsymbol{x})\sim p_{\text{data}}}[\max(0, 1-D(\boldsymbol{x}))] \\ &\quad +\mathbb{E}_{\boldsymbol{z}\sim p_z, \boldsymbol{e}\sim p_{text}}[\max(0, 1+D(G(\boldsymbol{z},\boldsymbol{e})))] \end{aligned} \tag{1}$$

SRM use the CTC loss:

$$L_{SRM} = +\mathbb{E}_{\boldsymbol{z}\sim p_z, \boldsymbol{e}\sim p_{text}}[\text{CTC}(\boldsymbol{e}, SRM(G(\boldsymbol{z},\boldsymbol{e}))] \tag{2}$$

SPM use the ACE loss:

$$L_{SPM} = +\mathbb{E}_{\boldsymbol{z}\sim p_z, \boldsymbol{e}\sim p_{text}}[\text{ACE}(\boldsymbol{e}, SPM(G(\boldsymbol{z},\boldsymbol{e})))] \tag{3}$$

Here, p_{data} denotes the distribution of real handwritten Chinese text image, p_z is a prior distribution on input noise z and p_{text} refers to a prior distribution of the text.

We adopt the loss terms balance rule [37] to obtain blance coefficient α and β. The total loss is:

$$L = L_G + \alpha L_{SRM} + \beta L_{SPM} \tag{4}$$

IV. Experiments

A. Datasets

The offline handwritten Chinese database, CASIA-HWDB [19] is a widely used database for handwritten Chinese recognition, containing single characters and handwritten text lines. Single character samples are divided into three databases: HWDB1.0 1.2(Including 7,185 classes Chinese characters and 171 classes English letters, numbers, and symbols). Handwritten text lines are also divided into three databases: HWDB2.0 2.2(Its character classes are contained in HWDB1.0 1.2).

The datasets HWDB1.0-Train and HWDB2.0 2.1-Train are added to HWDB1.1-Train and HWDB2.2-Train respectively for enlarging the training set size to promote the generation of characters/lines. The datasets HWDB1.0 1.1-Test and HWDB2.0 2.2-Test are used for inspecting performance.

B. Evaluation metrics

We follow the same quantitative evaluation measures as previously handwritten text generation methods [37], [38], [48]. We compare real handwritten images with generated results using these measures: (1)Fréchet Inception Distance (FID) is widely used and calculates the distance between the real and generated images; (2)the Geometric Score (GS), which compares the topology between the real and generated manifolds. For the above two indicators, lower is better. We evaluate FID/GS on a sampled set(HWDB1.1: with HWDB1.0 test set for FID and 7k samples for GS, HWDB2.2: with 20k samples for FID and 5k samples for GS), considering computational costs. FID/GS was computed on sampled images using 32×32W images.

For promoting handwritten Chinese text recognition, we evaluate the performance with accuracy(ACC), character error rate(CER), and edit-distance(ED). We use ACC to measure character recognition accuracy. CER and ED are used to measure lines recognition performance. CER is the number of misreads in the test set. The ED is calculated as the minimum edit distance between the predicted and true text. The font style transferring use different evaluation metrics and so are not directly comparable with our work.

C. Chinese handwriting generation results.

We report the experimental results on HWDB1.1 and HWDB2.2 respectively.

1) Ablation study: We conduct an ablation study by removing key modules: the Chinese text encoder(CTE), the sequence recognition module(SRM), and the spatial perception module(SPM). Without the CTE, the model produces blurry results in both isolated characters and handwritten lines images. Appling only the SRM or SPM, generated samples leads to an improvement in readability, but the character strokes are still not clear compared with HCT-GAN.

Compared with SRM, SPM is more critical for generating realistic-looking lines with spatial features(up-down offsetting between characters) and refined strokes, while it is not obvious in single character images. Without the SRM, The character in the generated line image is too tight, which shows that SRM is indispensable. We attempt to replace SRM with a character recognition network, which is not able to obtain better the quality of the images generated. Compared with the character recognizer, SRM effectively improves the image quality and does not increase the workload(the components sequence labels come from the dictionary CS). Meanwhile, SRM decreases the training overhead(Component class: 1,179, character class: 7,318). FID and GS are reported in Table I. Figure 3 shows some images generated by all versions.

2) Comparison to ScrabbleGAN: We train the HCT-GAN on two datasets (HWDB1.1 and HWDB2.2). Figure 4 represent results trained by ScrabbleGAN [38] alongside results of our method on the same characters/lines images. It is obvious from the figure that our network produces much clearer images, whether for isolated characters or variable-length lines.

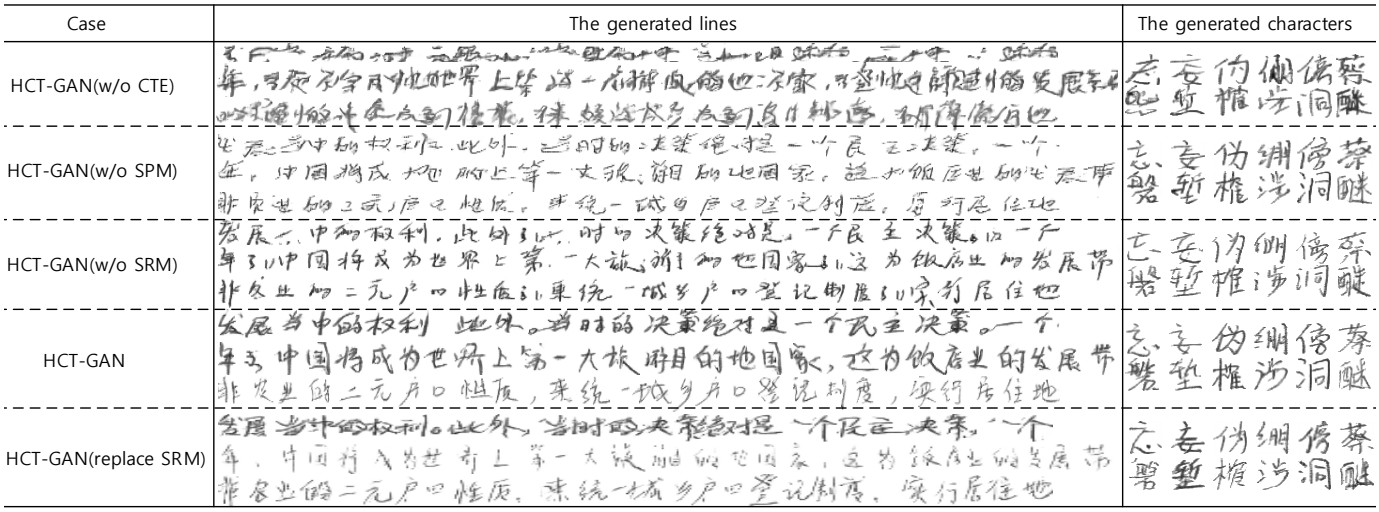

Figure 3. Ablation study on HWDB2.2(right) and HWDB1.1(left). HCT-GAN(replace SRM): Replacing SRM with character recognition network.

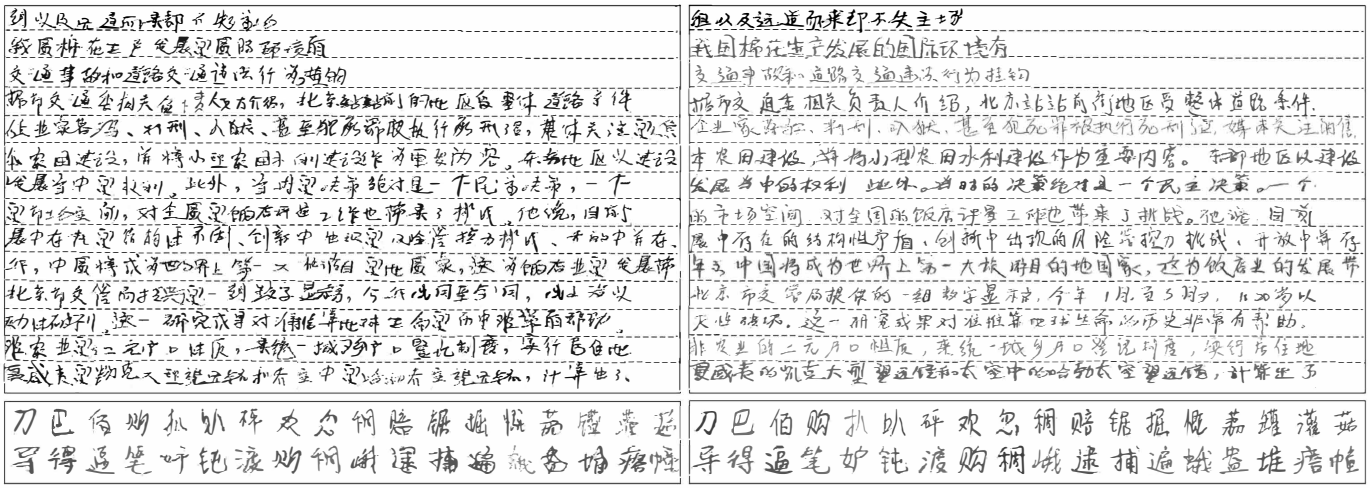

Figure 4. Comparing our results(right side) to those from ScrabbleGAN [38](left side) trained on the CASIA-HWDB [19] dataset. Top frame: the lines generated, bottom frame: the characters generated.

Table I
ABLATION STUDY. HCT-GAN(REPLACE SRM): REPLACING SRM WITH CHARACTER RECOGNITION NETWORK.

Dataset	Model	FID ↓	GS ↓
HWDB1.1	HCT-GAN(w/o CTE)	21.08	1.30×10^{-2}
	HCT-GAN(w/o SPM)	15.47	5.82×10^{-4}
	HCT-GAN(w/o SRM)	16.23	6.20×10^{-3}
	HCT-GAN	**15.14**	$\mathbf{4.10 \times 10^{-4}}$
	HCT-GAN(replace SRM)	18.62	8.30×10^{-3}
HWDB2.2	HCT-GAN(w/o CTE)	22.76	1.50×10^{-1}
	HCT-GAN(w/o SPM)	20.42	4.86×10^{-3}
	HCT-GAN(w/o SRM)	18.61	4.10×10^{-3}
	HCT-GAN	**17.82**	$\mathbf{3.33 \times 10^{-3}}$
	HCT-GAN(replace SRM)	20.69	4.00×10^{-3}

When the character strokes become complicated, the details of the top-left begin to blur. On the contrary, our network produces images(top-right) that are still clear. Directly generating variable-length handwritten lines is more challenging. The bottom-left text has lost readability, and the bottom-right text still has realistic-looking handwriting.

Table II
FID AND GS SCORES IN COMPARISON TO SCRABBLEGAN.

Dataset	Model	FID ↓	GS ↓
HWDB1.1	ScrabbleGAN	22.83	3.49×10^{-2}
	HCT-GAN	**15.14**	$\mathbf{4.10 \times 10^{-4}}$
HWDB2.2	ScrabbleGAN	32.04	1.39×10^{-1}
	HCT-GAN	**17.81**	$\mathbf{3.33 \times 10^{-3}}$

3) Diverse handwriting style: We can generate different handwriting styles by changing the noise vector z. Figure 5 shows examples of randomly selected characters/lines generated in different handwriting styles.

Figure 5. Diverse handwriting styles in the character-level(left) and line-level(right).

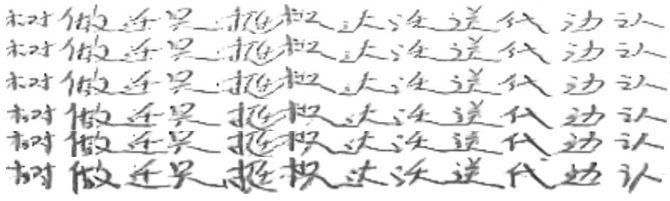

Figure 6. An interpolation between two different styles of handwriting generated by HCT-GAN.

4) Interpolation between different style: We are able to capture low dimensional manifold in high dimensional space by interpolating between two the random noises. Figure 6 shows the interpolation of the different handwriting styles between the random noises.

5) Generating unseen text containing similar characters: Each character in the Figure 7(left) is not in the training set, which indicates that HCT-GAN can generate handwritten lines consisting of unseen characters. Figure 7(right) shows the generated unseen lines containing similar Chinese characters.

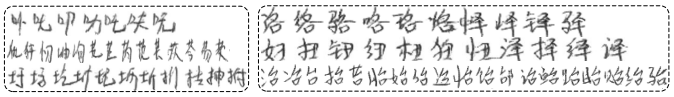

Figure 7. Generating unseen text. No character in the left lines appear in the training set. The right lines come from the similar Chinese characters.

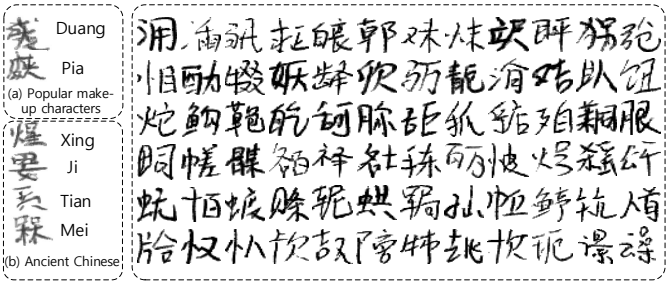

Figure 8. Generated nonsense glyphs by our model(right frame). (a) Generation of newly make-up fashionable Chinese characters from the Internet; (b) Generation of rarely used ancient Chinese characters with new interpretation from the Internet.

Table III
IMPACT ON THE TEXT RECOGNITION PERFORMANCE IN TERMS OF EDIT DISTANCE (ED),CHARACTER ERROR RATE (CER) ON THE TEST SET.

Set	Aug	HCT-GAN	CER[%] ↓	ED ↓
HWDB2.2	×	×	31.21 ± 0.13	8.18
	✓	×	11.13 ± 0.11	2.94
	×	20k	23.17 ± 0.21	6.01
	×	40k	20.12 ± 0.15	5.30
	✓	40k	**6.76 ± 0.17**	**1.76**

Table IV
IMPACT ON THE PERFORMANCE IN TERMS OF ACCURACY(ACC).

Set	Aug	HCT-GAN	ACC[%] ↑
HWDB1.1	×	×	85.65 ± 0.05
	✓	×	91.49 ± 0.22
	×	200k	86.44 ± 0.18
	×	400k	88.74 ± 0.31
	✓	400k	**92.27 ± 0.19**
	×	100k(nonsense)	**90.90 ± 0.08**
	✓	100k(nonsense)	**94.50 ± 0.22**

6) Generating nonsense glyphs: Figure 8(right) shows nonsense glyphs formed by the random combination of components and structures. Some fashionable make-up glyphs are shown in Figure 8(left).

D. Improving recognition performance

We use the code provided by [41] as our handwritten Chinese text recognition(HCTR) framework. We prove that HCTR performance can be improved by simply appending the generated image to the training set. Note that HCT-GAN only use the training set.

Table III, IV compare HCTR results on the CICAS-HWDB dataset. The second column('Aug') indicates usage of random affine augmentation. The third column ('HCT-GAN') indicates whether synthetic images were added to the original train set, and how many. As shown in the Table, using the HCT-GAN generated samples further improve the recognition performance compared to only using affine augmentation. Moreover, we find that applying nonsense glyphs as negative samples is more conducive to improvement in performance than equivalent character augmentation.

V. CONCLUSION

We have presented a components regulated model capable of directly generating images of handwriting Chinese lines with arbitrary length. In our work, we design a Chinese text encoder(CTE) suitable for Chinese text image generation and propose the spatial perception module(SPM). Experimental results show that the proposed method generates high-quality images of handwritten Chinese lines.

Our work still has some limitations,e.g., for characters with many strokes and close coupling, the generated images tend to get blurred. In fact, the same phenomenon is common in the GAN-based model. We point out this generally is not a big problem, since these characters are rarely used in modern Chinese. We will further study this problem in future work.

ACKNOWLEDGEMENTS

This research was supported in part by the National Key Research and Development Program of China under Grant No. 2020AAA0109702, and the National Natural Science Foundation of China under Grants 61976208, and the InnoHK project.

REFERENCES

[1] S. Azadi, M. Fisher, V. G. Kim, Z. Wang, E. Shechtman, and T. Darrell, "Multi-content gan for few-shot font style transfer," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7564–7573.

[2] B. Chang, Q. Zhang, S. Pan, and L. Meng, "Generating handwritten chinese characters using cyclegan," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2018, pp. 199–207.

[3] H. Jiang, G. Yang, K. Huang, and R. Zhang, "W-net: one-shot arbitrary-style chinese character generation with deep neural networks," in *International Conference on Neural Information Processing*. Springer, 2018, pp. 483–493.

[4] Y. Jiang, Z. Lian, Y. Tang, and J. Xiao, "Dcfont: an end-to-end deep chinese font generation system," in *SIGGRAPH Asia 2017 Technical Briefs*, 2017, pp. 1–4.

[5] X. Liu, G. Meng, S. Xiang, and C. Pan, "Fontgan: A unified generative framework for chinese character stylization and de-stylization," *arXiv preprint arXiv:1910.12604*, 2019.

[6] P. Lyu, X. Bai, C. Yao, Z. Zhu, T. Huang, and W. Liu, "Auto-encoder guided gan for chinese calligraphy synthesis," in *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, vol. 1. IEEE, 2017, pp. 1095–1100.

[7] D. Sun, Q. Zhang, and J. Yang, "Pyramid embedded generative adversarial network for automated font generation," in *2018 24th International Conference on Pattern Recognition (ICPR)*. IEEE, 2018, pp. 976–981.

[8] Y. Xie, X. Chen, L. Sun, and Y. Lu, "Dg-font: Deformable generative networks for unsupervised font generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 5130–5140.

[9] S. Yang, J. Liu, W. Wang, and Z. Guo, "Tet-gan: Text effects transfer via stylization and destylization," in *AAAI*, 2019.

[10] Y. Zhang, Y. Zhang, and W. Cai, "Separating style and content for generalized style transfer," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2018.

[11] Z. Zheng and F. Zhang, "Coconditional autoencoding adversarial networks for chinese font feature learning," 2018.

[12] C. Wen, J. Chang, Y. Zhang, S. Chen, Y. Wang, M. Han, and Q. Tian, "Handwritten chinese font generation with collaborative stroke refinement," 2019.

[13] S.-J. Wu, C.-Y. Yang, and J. Y. jen Hsu, "Calligan: Style and structure-aware chinese calligraphy character generator," 2020.

[14] Y. Huang, M. He, L. Jin, and Y. Wang, "Rd-gan: Few/zero-shot chinese character style transfer via radical decomposition and rendering," in *Computer Vision – ECCV 2020*, 2020.

[15] D. Sun, T. Ren, C. Li, H. Su, and J. Zhu, "Learning to write stylized chinese characters by reading a handful of examples," 2017.

[16] J. Cha, S. Chun, G. Lee, B. Lee, S. Kim, and H. Lee, "Few-shot compositional font generation with dual memory," 2020.

[17] S. Park, S. Chun, J. Cha, B. Lee, and H. Shim, "Multiple heads are better than one: Few-shot font generation with multiple localized experts," 2021.

[18] A. Graves, "Generating sequences with recurrent neural networks," *arXiv preprint arXiv:1308.0850*, 2013.

[19] C.-L. Liu, F. Yin, D.-H. Wang, and Q.-F. Wang, "Casia online and offline chinese handwriting databases," in *2011 International Conference on Document Analysis and Recognition*. IEEE, 2011, pp. 37–41.

[20] S. Xu, F. Lau, W. Cheung, and Y. Pan, "Automatic generation of artistic chinese calligraphy," *IEEE Intelligent Systems*, vol. 20, no. 3, pp. 32–39, 2005.

[21] S. Xu, H. Jiang, T. Jin, F. C. Lau, and Y. Pan, "Automatic generation of chinese calligraphic writings with style imitation," *IEEE Intelligent Systems*, vol. 24, no. 02, pp. 44–53, 2009.

[22] Z. Lian and J. Xiao, "Automatic shape morphing for chinese characters," in *SIGGRAPH Asia 2012 Technical Briefs*, 2012, pp. 1–4.

[23] A. Zong and Y. Zhu, "Strokebank: Automating personalized chinese handwriting generation," in *Twenty-Sixth IAAI Conference*, 2014.

[24] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," 2014.

[25] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014.

[26] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, "Self-attention generative adversarial networks," 2019.

[27] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," 2019.

[28] D. Chen, L. Yuan, J. Liao, N. Yu, and G. Hua, "Stylebank: An explicit representation for neural image style transfer," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1897–1906.

[29] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European conference on computer vision*. Springer, 2016, pp. 694–711.

[30] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.

[31] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[32] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8789–8797.

[33] Y. Choi, Y. Uh, J. Yoo, and J.-W. Ha, "Stargan v2: Diverse image synthesis for multiple domains," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8188–8197.

[34] Rewrite, "https://github.com/kaonashi-tyc/rewrite."

[35] Zi2zi, "https://github.com/kaonashi-tyc/zi2zi."

[36] P. Krishnan, R. Kovvuri, G. Pang, B. Vassilev, and T. Hassner, "Textstylebrush: Transfer of text aesthetics from a single example," *arXiv preprint arXiv:2106.08385*, 2021.

[37] E. Alonso, B. Moysset, and R. Messina, "Adversarial generation of handwritten text images conditioned on sequences," in *2019 International Conference on Document Analysis and Recognition (ICDAR)*. IEEE, 2019, pp. 481–486.

[38] S. Fogel, H. Averbuch-Elor, S. Cohen, S. Mazor, and R. Litman, "Scrabblegan: Semi-supervised varying length handwritten text generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 4324–4333.

[39] M. Shell, H. Simpson, J. Kirk, and M. Scott, "Bare demo of ieeetran. cls for ieee conferences," 2015.

[40] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.

[41] B. Shi, X. Bai, and C. Yao, "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, 2016.

[42] Z. Xie, Y. Huang, Y. Zhu, L. Jin, Y. Liu, and L. Xie, "Aggregation cross-entropy for sequence recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6538–6547.

[43] P. He, W. Huang, Y. Qiao, C. C. Loy, and X. Tang, "Reading scene text in deep convolutional sequences," in *Thirtieth AAAI conference on artificial intelligence*, 2016.

[44] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[45] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, "Spectral normalization for generative adversarial networks," *arXiv preprint arXiv:1802.05957*, 2018.

[46] J. H. Lim and J. C. Ye, "Geometric gan," *arXiv preprint arXiv:1705.02894*, 2017.

[47] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.
[48] B. Davis, C. Tensmeyer, B. Price, C. Wigington, B. Morse, and R. Jain, "Text and style conditioned gan for generation of offline handwriting lines," *arXiv preprint arXiv:2009.00678*, 2020.