SCIENCE CHINA
Information Sciences

• **Supplementary File** •

Few-shot Font Style Transfer with Multiple Style Encoders

Kejun ZHANG[1], Rui ZHANG[1], Yonglin WU[1], Yifei LI[2],
Yonggen LING[3], Bolin WANG[1], Lingyun SUN[1] & Yingming LI[4*]

[1]*State Key Lab of CAD&CG, Zhejiang University, Hangzhou* 310027, *China;*
[2]*School of software Technology, Zhejiang University, Hangzhou* 310027, *China;*
[3]*Robotics X, Tencent, Shenzhen* 518054, China;
[4]*College of Information Science & Electronic Engineering, Zhejiang University, Hangzhou* 310027, *China.*

Appendix A Network structure

The details of the whole network structure are shown in Table A1.

Appendix B Experiments

Appendix B.1 Data preparation and experimental settings

We use public 243 fonts (each font has 1125 characters) and divide them into images with known styles (training set) and novel styles (validation set). As shown in Figure B1, during each time of the training, we first randomly sample K target images $I_{i_k,j}$ with known styles and known contents. Then we construct K style reference sets $R_{S_{i_k}}$ and one content reference set R_{C_j} by randomly sampling r images of the corresponding style/content. The input are $R_{S_{i_k}}$ and R_{C_j}, and the target are $I_{i_k,j}$. When testing for one specific style transfer, we input three same style reference sets $R_{S_{i_1}}$ and one content reference set R_{C_j}, which has the same input and output as EMD, to get a new character $\hat{I}_{i_1,j}$. Therefore, we can compare our testing results with those of EMD in the same condition.

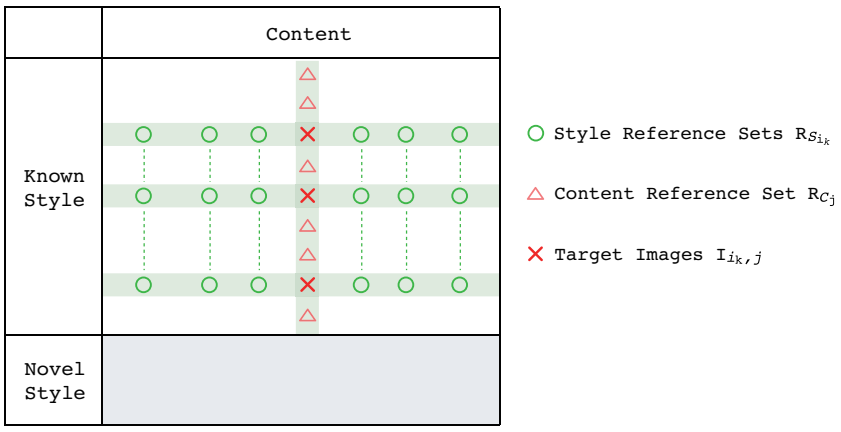

Figure B1 The illustration of data partition and the selection of target images, style reference sets and content reference set

All experiments are run on a server equipped with Ubuntu 16.04 system, two Intel(R) Xeon(R) Gold 5218 CPUs @ 2.3.0 GHz, and an NVIDIA GTX 2080 Ti GPU. The proposed method is implemented on Tensorflow 2.30. We use the pygame package to transform True-Type fonts into images. During training, our model is trained for 200 epochs to generate images with 80×80 resolution. The learning rate of the Adam optimizer is 0.0002. The batch size is set to 50 in the multi-style network.

* Corresponding author (email: yingming@zju.edu.cn)
† Authors Kejun ZHANG and Rui ZHANG have the same contribution to this work.

	Content Encoder: Conv			Style Encoder: Conv		
Inputs	80×80×10			80×80×10		
Layer	Kernel	Stride	Additional Layer	Kernel	Stride	Additional Layer
1	5×5×64	1×1	Batchnorm+LeakyReLU(0.2)	5×5×64	1×1	Batchnorm+LeakyReLU(0.2)
2	3×3×128	2×2	Batchnorm+LeakyReLU(0.2)	3×3×128	2×2	Batchnorm+LeakyReLU(0.2)
3	3×3×256	2×2	Batchnorm+LeakyReLU(0.2)	3×3×256	2×2	Batchnorm+LeakyReLU(0.2)
4	3×3×512	2×2	Batchnorm+LeakyReLU(0.2)	3×3×512	2×2	Batchnorm+LeakyReLU(0.2)
5	3×3×512	2×2	Batchnorm+LeakyReLU(0.2)	3×3×512	2×2	Batchnorm+LeakyReLU(0.2)
6	3×3×512	2×2	Batchnorm+LeakyReLU(0.2)	3×3×512	2×2	Batchnorm+LeakyReLU(0.2)
7	3×3×512	2×2	Batchnorm+LeakyReLU(0.2)	3×3×512	2×2	Batchnorm+LeakyReLU(0.2)
8	3×3×512	2×2	Batchnorm+LeakyReLU(0.2)	3×3×512	2×2	Batchnorm+LeakyReLU(0.2)
Output	1×1×512			1×1×512		

Mixer:BiLinear	
Inputs	1×1×512, 1×1×512
weight	512×512×512
Outputs	1×1×512

	Decoder: Deconv			Discriminator: Conv		
Inputs	1×1×512			80×80×10, 80×80×10, 80×80×1		
Layer	Kernel	Stride	Additional Layer	Kernel	Stride	Additional Layer
1	3×3×512	2×2	Batchnorm+ReLU	5×5×64	2×2	Batchnorm+LeakyReLU(0.2)
2	3×3×512	2×2	Batchnorm+ReLU	5×5×128	2×2	Batchnorm+LeakyReLU(0.2)
3	3×3×512	2×2	Batchnorm+ReLU	5×5×256	2×2	Batchnorm+LeakyReLU(0.2)
4	3×3×512	2×2	Batchnorm+ReLU	3×3×1	1×1	
5	3×3×256	2×2	Batchnorm+ReLU			
6	3×3×128	2×2	Batchnorm+ReLU			
7	3×3×64	2×2	Batchnorm+ReLU			
8	5×5×1	1×1	——			
Output	80×80×1			10×10×1		

Table A1 Network structure of different modules in MS-EMD.

Appendix B.2 Evaluation metrics

To evaluate our method, three commonly used evaluation metrics are employed: L1 loss (Least Absolute Deviations loss), SSIM (Structural SIMilarity) and FID (Frechet Inception Distance). L1 Loss Function is used to calculate the distance which is the sum of all the absolute differences between the true value and the predicted value. SSIM [7] calculates the image similarity by luminance, contrast and structure. FID [3] captures the similarity of generated images to real ones by comparing the distribution of generated images with the distribution of real images,

$$L_1 = |f(x) - Y|$$

$$SSIM(I_t, I_g) = [l(I_t, I_g)]^{\alpha}[c(I_t, I_g)]^{\beta}[s(I_t, I_g)]^{\gamma}$$

where l, c and s are the luminance, contrast and structure respectively, and α, β, γ are positive constants that indicate the relative importance of each component.

$$FID = ||\mu_r - \mu_g||^2 + T_r(\Sigma_r + \Sigma_g - 2(\Sigma_r\Sigma_g)^{1/2})$$

where μ_r, μ_g and Σ_r, Σ_g are the respective means and covariance matrices of true images and generated images.

A higher value of SSIM is better, while for L1 loss and FID, the lower the better.

Appendix B.3 Ablation study

To analyze the impacts of the multi-style encoder module and the discriminator module, we conduct a series of ablation studies. We remove or change the proposed modules and then compare the results with those of EMD. At last, we compare the generation for novel content with EMD.

Appendix B.3.1 *Multi-style encoder module*

We adopt 2, 3 and 4 style encoders for detailed comparison and denote the corresponding methods as MS-2, MS-3 and MS-4, respectively. In this part, we do not employ the discriminator. As shown in Table B1, combining EMD with the multi-style module has a remarkable promotion on all the metrics on known styles. For novel styles, the Multi-style encoder also has a promotion, especially on FID. In Figure B2, the qualitative results also show the same conclusion. Though MS-4 obtains better values in Table B1 in most cases, for the novel style generation testing its performance is similar to that of MS-3. Since in this work novel style generation and fusion are the main focus, MS-3 is selected for the effectiveness and reduced computation costs.

	known style			novel style		
	L1 loss↓	SSIM↑	FID↓	L1 loss↓	SSIM↑	FID↓
EMD	0.0598	0.8046	23.0840	0.1258	0.6203	32.3935
MS-2	0.0337	0.8897	15.2095	0.1243	0.6286	27.6735
MS-3	0.0253	0.9186	12.0797	0.1244	0.6298	**26.2178**
MS-4	**0.0208**	**0.9340**	**10.9472**	**0.1241**	**0.6313**	26.6044

Table B1 Quantitative results for ablation studies of multi-style encoder module. x in MS-x means the number of style encoders.

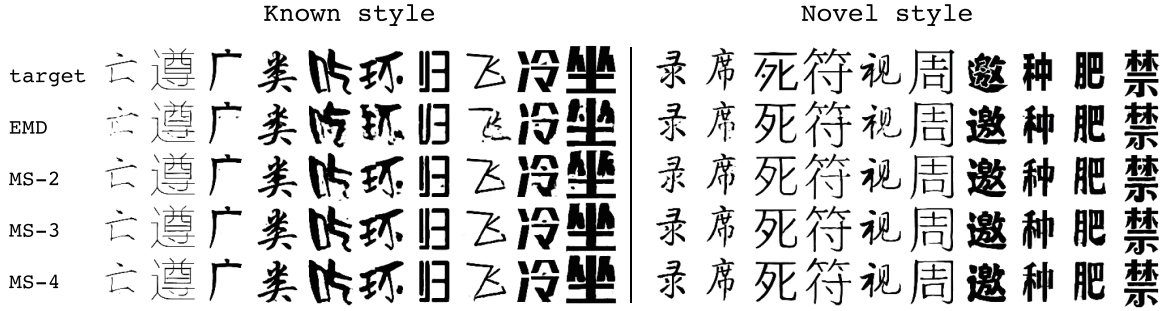

Figure B2 Qualitative results for ablation studies of multi-style encoder module. x in MS-x means the number of style encoders.

Appendix B.3.2 *Discriminator module*

Compared with the original EMD, the discriminator module is proposed to conduct the generation process and get more similar images as targets. In this part, we test the impact of the discriminator module through an EMD+Dis setup. We set the weight of L_{adv} as 1 and try different weights of L_1. We test different weights by dichotomy. As shown in Table B2, the discriminator with the weight of 420 has the best results for the metrics, especially for FID. Bigger or smaller weights make the results worse. For unknown styles, the results of FID also have a great promotion, and the values of other metrics are better than the original EMD. In Figure B3, the qualitative results show that the local details of "EMD+Dis" are more smooth and have fewer blurs.

	known style			novel style		
	L1 loss↓	SSIM↑	FID↓	L1 loss↓	SSIM↑	FID↓
EMD	0.0598	0.8046	23.0840	0.1258	0.6203	32.3935
EMD+Dis (w=400)	0.0614	0.8008	12.4298	0.1265	0.6185	18.6554
EMD+Dis (w=420)	**0.0591**	**0.8084**	**11.4372**	**0.1249**	**0.6227**	**18.2333**
EMD+Dis (w=450)	0.0595	0.8071	12.1836	0.1255	0.6216	18.4557
EMD+Dis (w=500)	0.0603	0.8047	13.0437	0.1258	0.6208	19.5331
EMD+Dis (w=600)	0.0595	0.8073	14.0394	0.1251	0.6220	20.5439

Table B2 Quantitative results for ablation studies of the discriminator module. w means the weights of L_1.

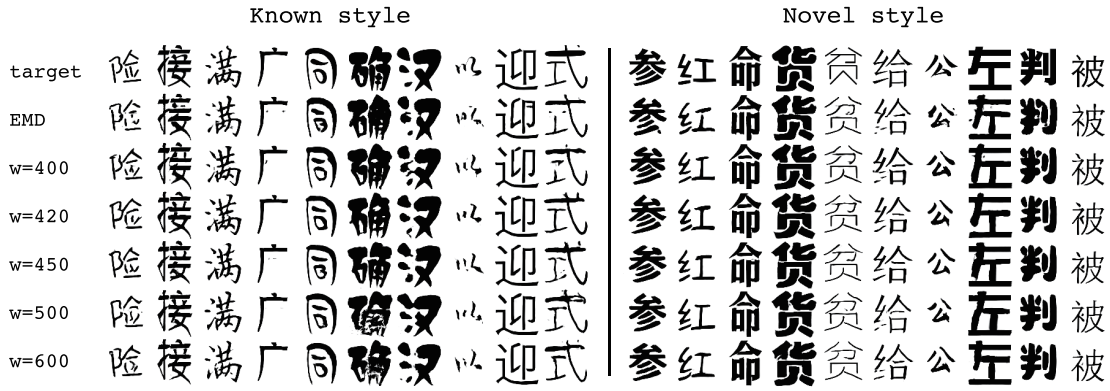

Figure B3 Qualitative results for ablation studies of the discriminator module. w means the weights of L_1.

Appendix B.3.3 *Multi-style encoder and discriminator*

In this part, the two modules above are incorporated into EMD to verify that they can work better together. As shown in Table B3, our MS-EMD is better than the original EMD or methods in that we only use multi-style or discriminator for all three metrics. This proves that the combination of the above two modules can work better than one. The results also show that the multi-style encoder has a significant impact on L1 loss and SSIM, while the discriminator remarkably influences FID. In Figure B4, the qualitative results show the influence of these two parts.

	known style			novel style		
	L1 loss↓	SSIM↑	FID↓	L1 loss↓	SSIM↑	FID↓
EMD	0.0598	0.8046	23.0840	0.1258	0.6203	32.3935
MS-3	0.0253	0.9186	12.0797	0.1244	0.6298	26.2178
EMD+Dis(w=420)	0.0591	0.8084	11.4372	0.1249	0.6227	18.2333
MS-EMD	**0.0246**	**0.9221**	**5.6239**	**0.1243**	**0.6305**	**16.6098**

Table B3 Quantitative results for ablation studies of multi-style and discriminator.

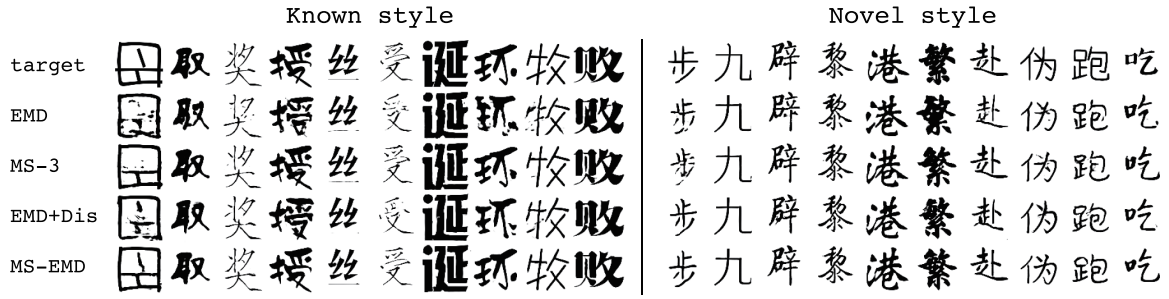

Figure B4 Qualitative results for ablation studies of multi-style and discriminator.

Appendix B.3.4 *Generation for novel content*

Although our task is style transfer, we still generate characters for novel content and compare them with EMD as shown in Table B4 and Figure B5. MS-EMD's results have fewer blurs than EMD's. For some failure cases, we can see that the characters are both wrong. Actually, both methods have the limitation of generation for some novel contents. In general, MS-EMD obtains comparable results as EMD.

	L1 loss↓	SSIM↑	FID↓
EMD	0.1119	0.6617	40.0141
MS-EMD	0.1173	0.6529	28.7113

Table B4 Quantitative results for novel content.

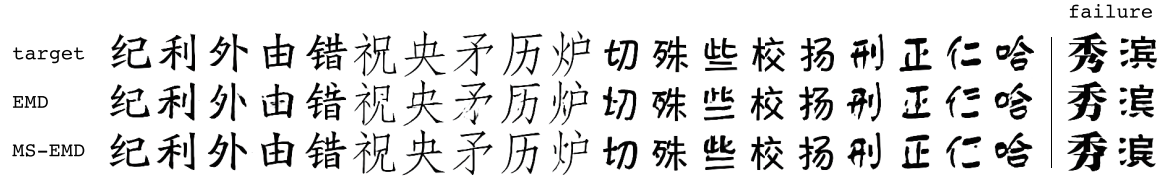

Figure B5 Qualitative results for novel content. Failure means failure cases.

Appendix B.4 Experimental results

In this subsection, we make qualitative and quantitative comparisons between existing methods and our method MS-EMD. The main tasks are generating characters with novel styles and font fusion. The latter task includes font fusion of different styles and weights. We select the following methods for comparison in this paper:

Zi2zi [5]: zi2zi is a representative work for Chinese font generation. It has a structure of GAN and is an extension of pix2pix [4] model. zi2zi can not achieve few-shot font generation, so we omit it on this task. However, it can learn many fonts at the same time and can get a fusion of different fonts by inputting a decimal as the label. During the experiment, we use Source Han Serif [1], a widely-used open source font, as its source font for the generation. We choose certain 17 fonts instead of 182 fonts that our model used as its target fonts because zi2zi can not generate so many fonts at the same time.

EMD [6]: EMD is proposed to generate Chinese fonts by separating style and content. EMD consists of a content encoder, a style encoder, a mixer and a decoder. By interpolating features extracted by style encoder, it can generate fonts with fused style. We use the same dataset as our MS-EMD for training and testing.

AGIS-Net [2]: AGIS-Net is proposed to generate artistic glyph images. AGIS-Net has two encoders, two decoders and three discriminators. Similar to EMD, by interpolating features extracted by style encoder, it also can generate fonts with fused style. We use the average skeleton of many fonts as the inputs of content encoder, and other data used are the same as our MS-EMD.

For comparison, we only input one style reference each time, which is fed into the three style encoders. Then we calculate the average of the latent vectors of the three encoders and feed it into the next layer. This can be regarded as a way of ensemble learning which makes the results better.

Appendix B.4.1 *Generation for known style*

We test three models (EMD, AGIS-Net, MS-EMD) to generate images of known styles from $r = 10$ reference images. This task has influence on font fusion. The quantitative results (Table B5) and qualitative results (Figure B6) show that MS-EMD gets the best results. The last two failure cases show that these three methods can not generate great characters for some strange styles.

	L1 loss↓	SSIM↑	FID↓
EMD	0.0598	0.8046	23.0840
AGIS-Net	0.0734	0.7847	13.9097
MS-EMD	**0.0246**	**0.9221**	**5.6239**

Table B5 Quantitative results for known style.

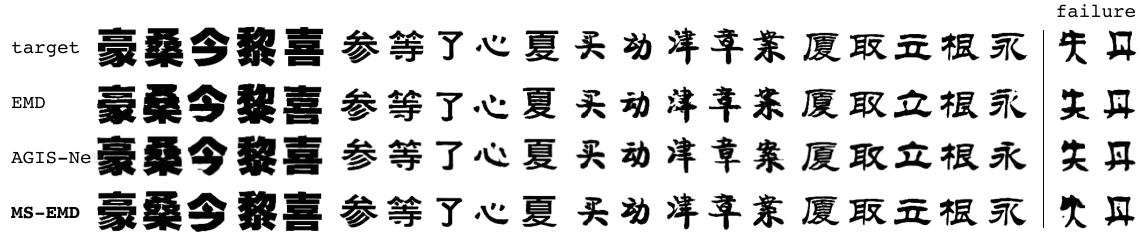

Figure B6 Qualitative results for known style. Failure means failure cases.

Appendix B.4.2 *Generation for novel style*

We test three models (EMD, AGIS-Net, MS-EMD) to generate images of novel styles from $r = 10$ reference images. As shown in Table B6, MS-EMD obtains the best quantitative results for L1 loss, SSIM and FID. It is noted that L1 loss and SSIM focus on pixel-level between generated images and targets, while ignoring some important features for human perception, such as structure, smoothness, aesthetics, etc. FID can represent human perception to some degree, and the proposed method obtains big promotion on this measure.

For qualitative comparison, we choose different characters with several different kinds of styles, as shown in Figure B7, EMD can generate characters similar to the target but the details are awful. The most basic stroke *horizontal (Heng)* and *vertical (Shu)* should be straight in most cases, but EMD's results usually have jitters on strokes. This flaw may only slightly influence the quantitative value but it is awful for human perception and is unacceptable. Besides, there are too many blurs and some lack of pixels which make the font very coarse. If not considering the required targets, some results are pretty good, but it is a big

	L1 loss↓	SSIM↑	FID↓
EMD	0.1258	0.6203	32.3935
AGIS-Net	0.1320	0.6091	20.0100
MS-EMD	**0.1243**	**0.6305**	**16.6098**

Table B6 Quantitative results for novel style.

problem that the styles of generated characters are not similar to targets. MS-EMD generates characters with better details and more similar styles. For the failure cases in Figure B7, we can see that all three methods generate wrong styles. Although MS-EMD can generate great characters from 10 references in most cases, it is still hard to generate some strange styles.

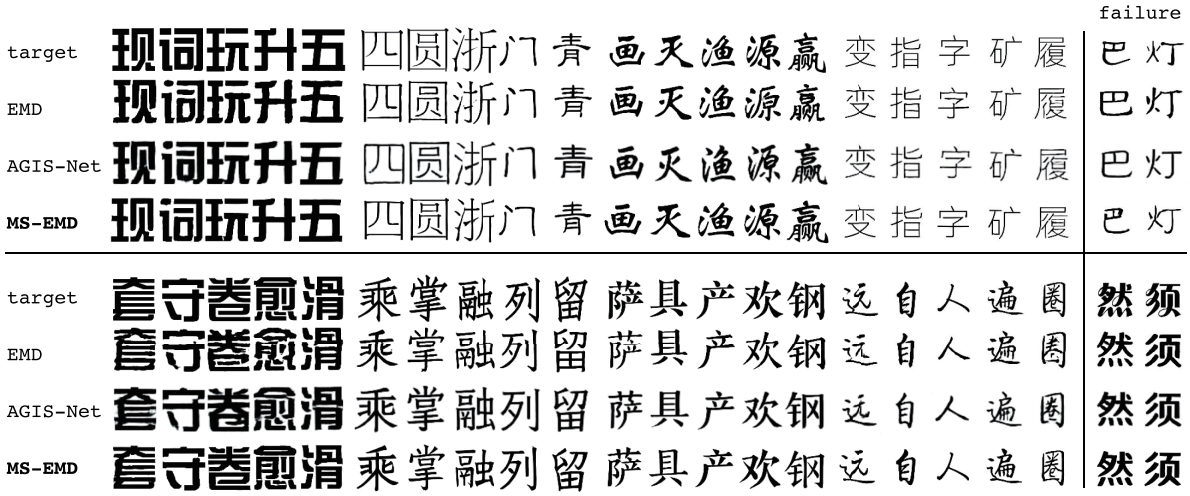

Figure B7 Qualitative results for novel style. Failure means failure cases.

Appendix B.4.3 *Font fusion*

The fusion of different fonts is a special style transfer task and there are no ground truths for fused characters. Several qualitative comparisons are performed between zi2zi, AGIS, EMD and MS-EMD.

First, we test the two-style fusion performance with several representative fonts. Since Heiti, Kaiti and Songti are the three most frequently used fonts, we choose these three to test the ability of MS-EMD. Besides, OPPO Sans and STLiti are also employed to show the superiority of MS-EMD. As shown in Figure B8, zi2zi's results always have a clear boundary along with the fusion of varying weights between two font styles, which means zi2zi is unstable to generate fused characters. For AGIS's fusion, the middle characters are light and strange. EMD can generate fused characters with reasonable style but the details are awful, i.e., there are too many blurs. MS-EMD generates characters with a fused style clearly and correctly in most cases. However, there are still a few flaws in some characters, this may need further processing. After generating the fused characters with different styles, designers can easily create a new font by using these characters as drafts.

Further, we also try to fuse characters with more than two styles. As shown in Figure B9, the most classical character *yong* is selected as the testing object. Meanwhile, the most frequently used fonts are adopted: Heiti, Kaiti and Songti. If we choose the point on the edges of the triangle, the results will be the fusion of two corresponding styles. If we choose the point in the triangle, the fusion of the three styles would be generated. Three midcourt lines are selected for the presentation. By the font fusion of three styles, designers can try more combinations and design new fonts more conveniently.

For generating fonts with middle weights between two known weights of a same glyph family, we choose the Tencent Sans as the testing object. Tencent Sans has two different weights to generate fonts with new weights. To make the characters transform smoothly, we carefully calculate the weights and select the characters with evenly divided weights. As shown in Figure B10, MS-EMD obtains the best results. Zi2zi's middle characters have a sudden change between light font and bold font. AGIS's middle characters are hazy. EMD's results have many noisy points. MS-EMD's results are acceptable for most cases. However, there are still a few flaws in some characters, which may need further processing. With this fusion process, the generated characters can be packaged in new fonts more conveniently.

References

1 Adobe Fonts. adobe-fonts/source-han-serif, March 2021.
2 Yue Gao, Yuan Guo, Zhouhui Lian, Yingmin Tang, and Jianguo Xiao. Artistic glyph image synthesis via one-stage few-shot learning. *ACM Transactions on Graphics*, 38(6):185:1–185:12, November 2019.
3 Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. *arXiv:1706.08500 [cs, stat]*, 2018.
4 Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. *arXiv:1611.07004 [cs]*, November 2016.
5 Yuchen Tian. kaonashi-tyc/zi2zi: Learning Chinese Character style with conditional GAN, September 2017.

6 Yexun Zhang, Ya Zhang, and Wenbin Cai. Separating Style and Content for Generalized Style Transfer. pages 8447–8455, 2018.
7 Zhou Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

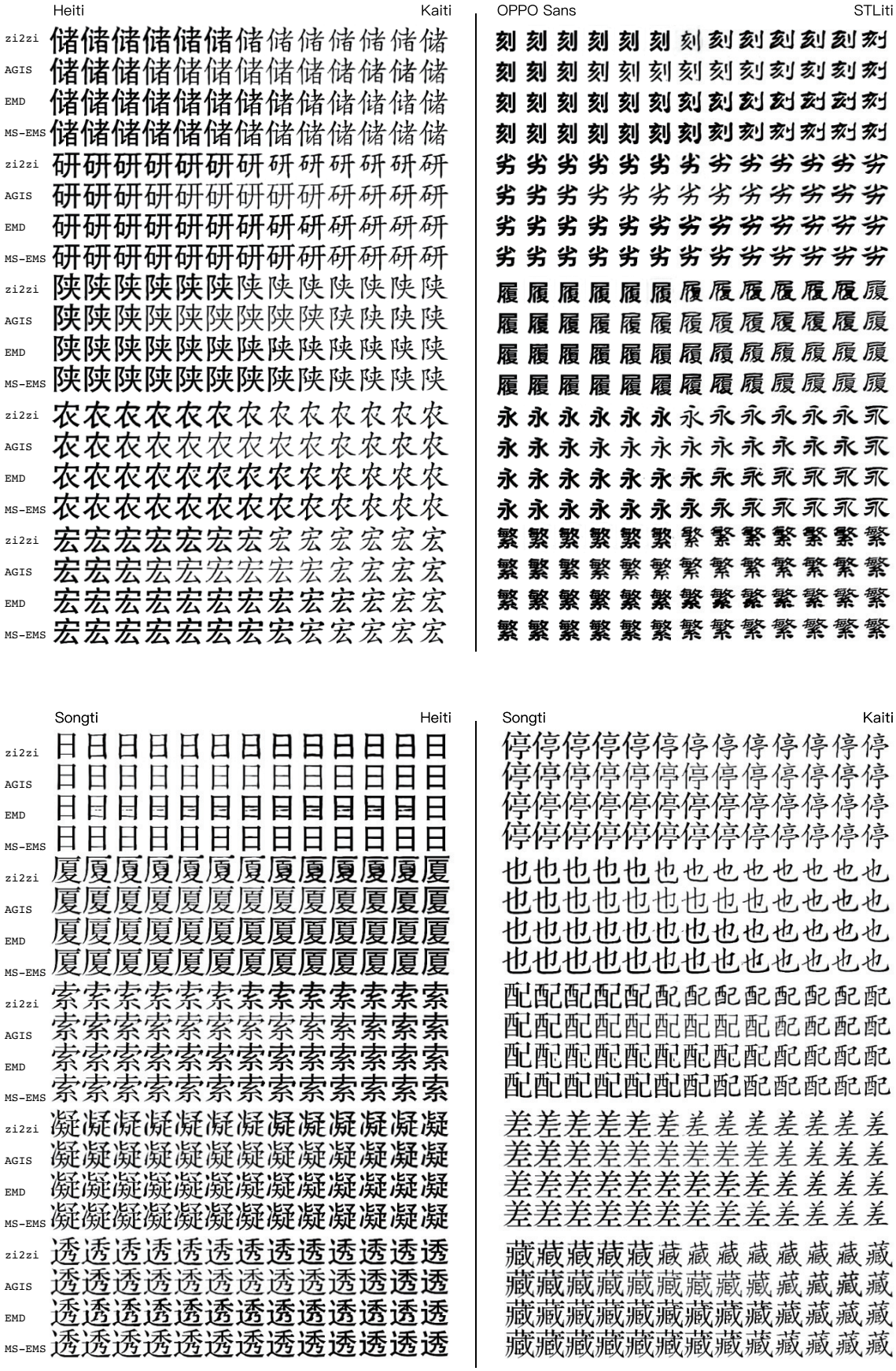

Figure B8 Font fusion of two styles by four methods: zi2zi, EMD, AGIS and our MS-EMD.

Figure B9 Image generation for fused styles of three fonts: Heiti, Kaiti and Songti.

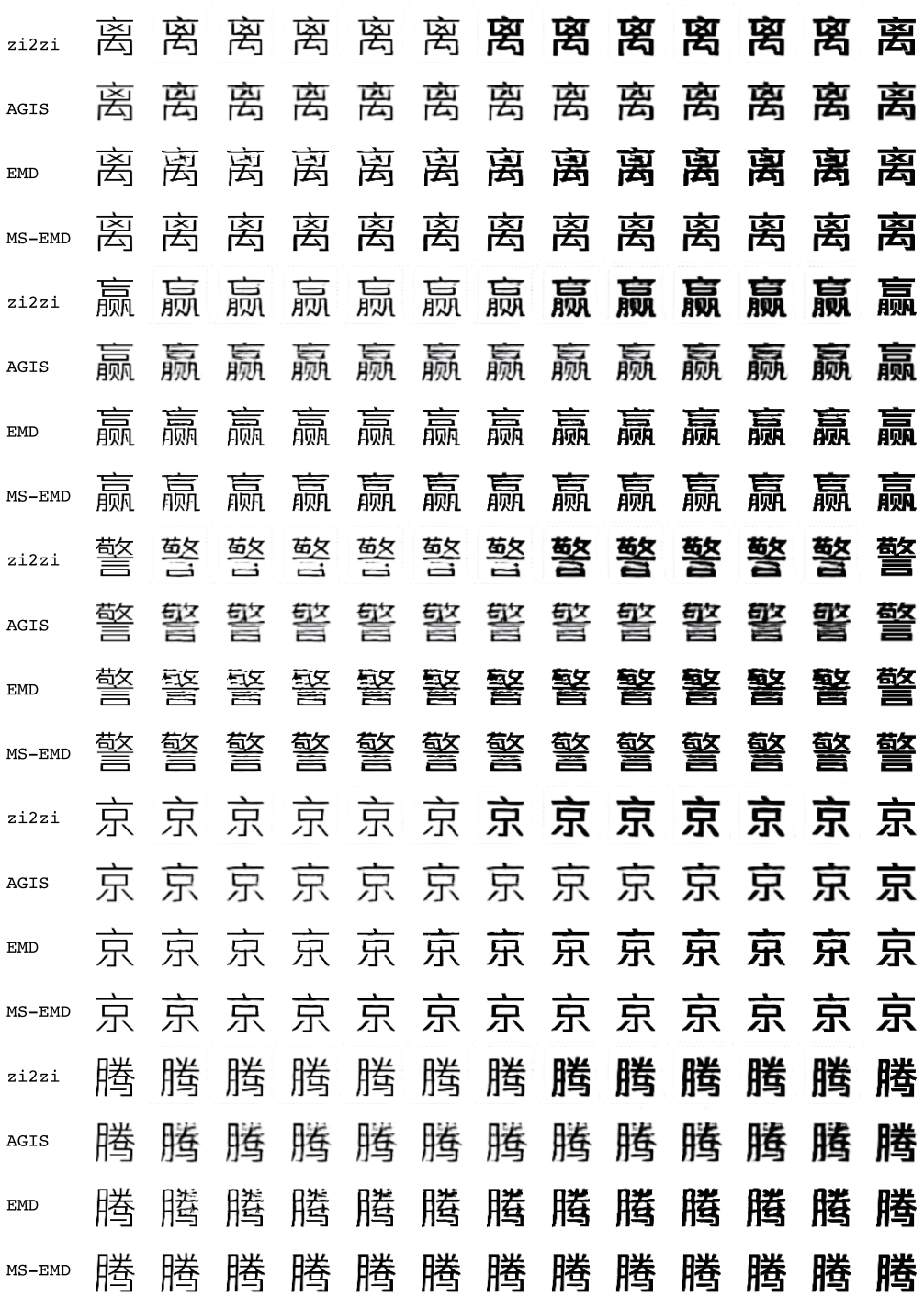

Figure B10 Font fusion of two weights by four methods: zi2zi, EMD, AGIS and our MS-EMD.