南京理工大学
NANJING UNIVERSITY OF SCIENCE & TECHNOLOGY

Few-shot Font Generation

南京理工大学
NANJING UNIVERSITY OF SCIENCE & TECHNOLOGY

DG-Font: Deformable Generative Networks for Unsupervised Font Generation

Yangchen Xie, Xinyuan Chen, Li Sun, Yue Lu

CVPR 2021

Overview

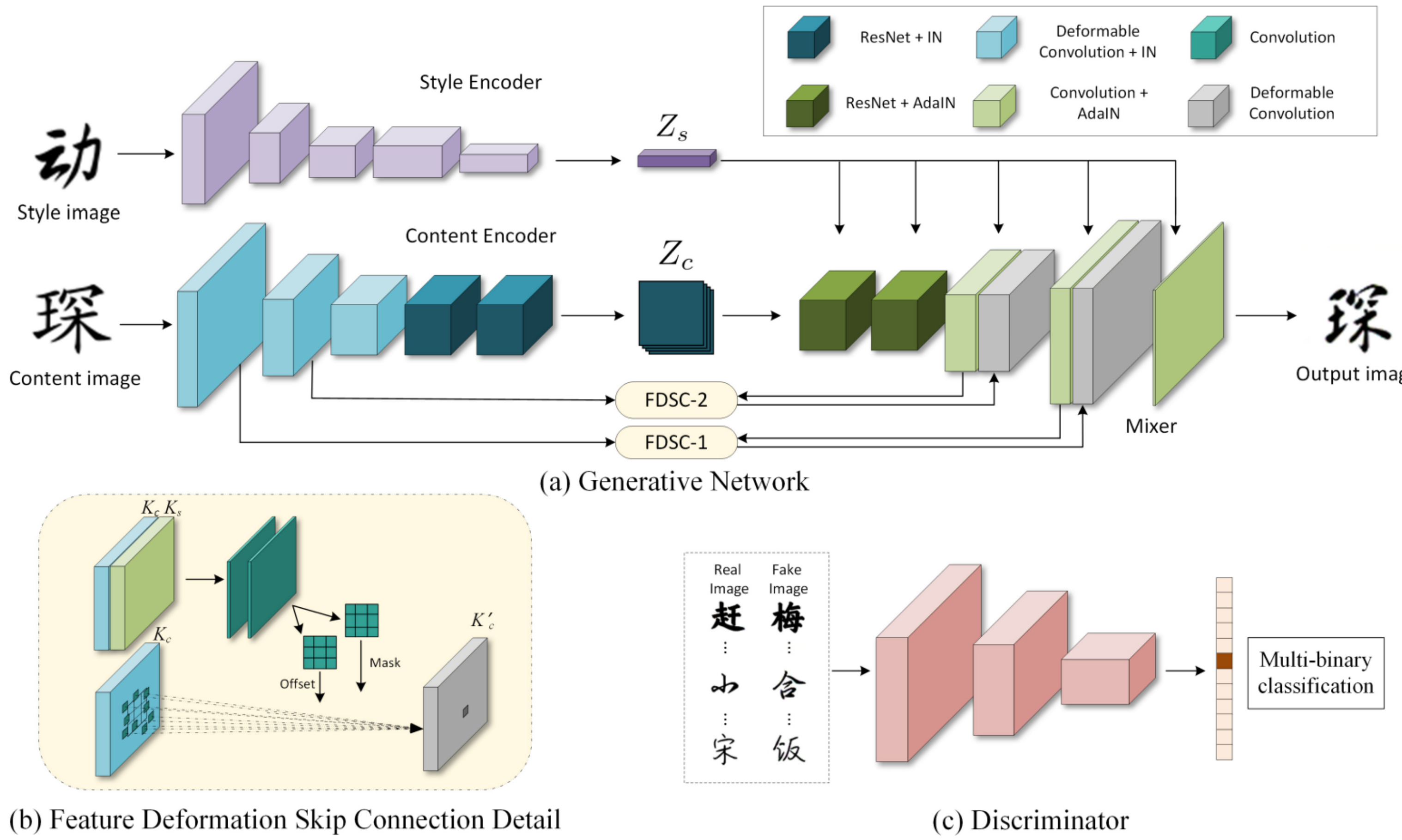


Figure 2. **Overview of the proposed method**. a) Overview of our generative network. The Style/content encoder maps style/content image to style/content representation Z_s/Z_c. FDSC-1 and FDSC-2 have the same architecture and apply transformation convolution to the low-level feature from the content encoder and inject the results into the mixer. The mixer generates the output image. b) A detailed illustration of the FDSC module. c) The discriminator output a binary vector, where each element indicates a binary classification to distinguish between generated and real images.

Feature Deformation Skip Connection(FDSC)

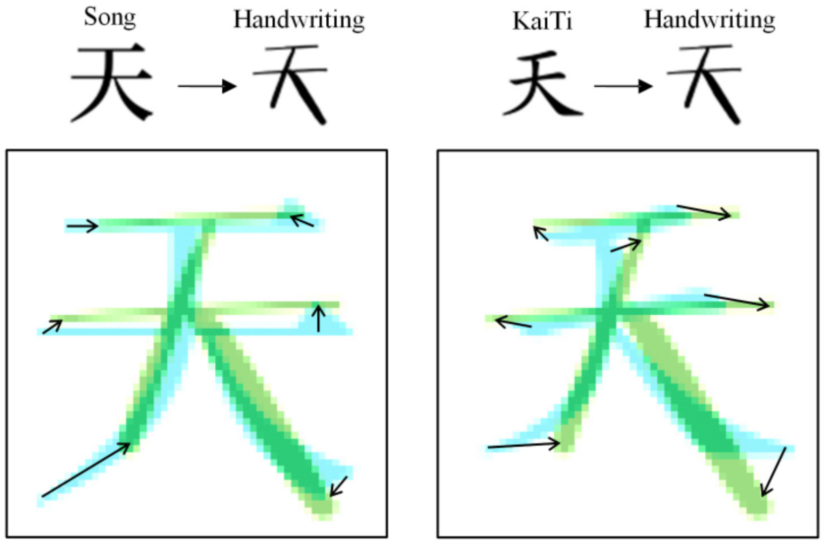
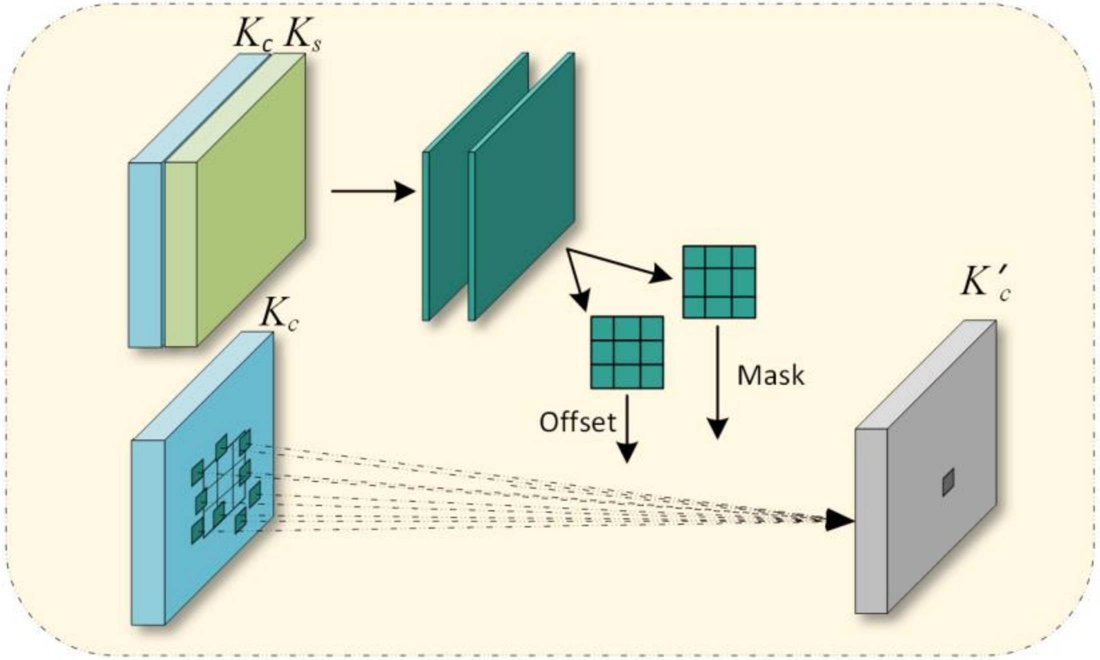


Figure 3. **The geometric deformation of two fonts for a character**. We employ the character "Tian" to compare a handwritten style with the fonts of Kaiti and Song. There is a correspondence for each stroke between two fonts.



$$\Theta = f_\theta(K_s, K_c)$$

where, $\Theta = \{\Delta p_k, \Delta m_k \mid k = 1, \cdots, |\mathcal{R}|\}$ refers to the offsets and mask of the convolution kernel, $\mathcal{R} = \{(-1,-1), (-1,0), \cdots, (0,1), (1,1)\}$ indicates a regular grid of a 3×3 kernel.

$$K'_c = f_{DC}(K_c, \Theta)$$

$$K'_c(p) = \sum_{k=1}^{\mathcal{R}} w(p_k) \cdot x(p + p_k + \Delta p_k) \cdot \Delta m_k$$

Loss Function

Adversarial loss:

$$\mathcal{L}_{adv} = \max_{D_s} \min_{G} \mathbb{E}_{I_s \in P_s, I_c \in P_c} [\log D_s(I_s) + \log(1 - D_s(G(I_s, I_c)))], \quad (4)$$

where $D_s(\cdot)$ denotes the logit from the corresponding style of discriminator's output.

Image Reconstruction loss:

$$\mathcal{L}_{img} = \mathbb{E}_{I_c \in P_c} \left\| I_c - G(I_c, I_c) \right\|_1 . \quad (6)$$

Content consistent loss:

$$\mathcal{L}_{cnt} = \mathbb{E}_{I_s \in P_s, I_c \in P_c} \left\| Z_c - f_c(G(I_s, I_c)) \right\|_1 . \quad (5)$$

Deformation offset normalization:

$$\mathcal{L}_{offset} = \frac{1}{|\mathcal{R}|} \left\| \Delta p \right\|_1 , \quad (7)$$

where Δp denotes offsets of the deformable convolution kernel, $|\mathcal{R}|$ denotes the number of the convolution kernel.

Overall Objective loss: $\mathcal{L} = \mathcal{L}_{adv} + \lambda_{img}\mathcal{L}_{img} + \lambda_{cnt}\mathcal{L}_{cnt} + \lambda_{offset}\mathcal{L}_{offset}, \quad (8)$

Dataset

Total

- 410 字体
- 每个字体包含 990 个汉字

Train

- 400 字体
- 每个字体 800 个汉字

Test

- Seen：400*190
- Unseen：10*990

Resolution：80×80

Source Font：宋体

Reference Images：随机 10 个训练集中字符

Experiments

Methods	one-to-many	training	L1 loss	RMSE	SSIM	LPIPS	FID
			Seen fonts				
EMD [58]	✓	paired	0.0538	0.1955	0.7676	0.1036	89.65
Zi2zi [61]	✓	paired	**0.0521**	0.1802	0.7789	0.1065	142.23
Cycle-GAN [59]	×	unpaired	0.0863	0.2555	0.6392	0.1825	175.24
GANimorph [20]	×	unpaired	0.0563	**0.1759**	**0.7808**	0.1403	72.89
FUNIT [33]	✓	unpaired	0.0807	0.2510	0.6669	0.1216	53.77
Ours	✓	unpaired	0.0562	0.1994	0.7580	**0.0814**	**46.15**
			Unseen fonts				
EMD [58]	✓	paired	0.0430	0.1755	0.7849	0.1255	82.53
FUNIT [33]	✓	unpaired	0.0588	0.2089	0.7417	0.1125	59.98
Ours	✓	unpaired	**0.0414**	**0.1709**	**0.7982**	**0.0867**	**50.29**

Table 1. **Quantitative evaluation on the whole dataset**. We evaluate the methods on seen and unseen font sets. The bold number indicates the best.

Experiments

(a) Easy cases (*i.e.*, non-cursive writing).

(b) Challenging cases (*i.e.*, cursive writing).

Figure 4. Comparisons to the stat-of-art methods for font generation.

CF-Font: Content Fusion for Few-shot Font Generation

Chi Wang, Min Zho, Tiezheng Ge, Yuning Jiang, Hujun Bao, Weiwei Xu

CVPR 2023

Framework

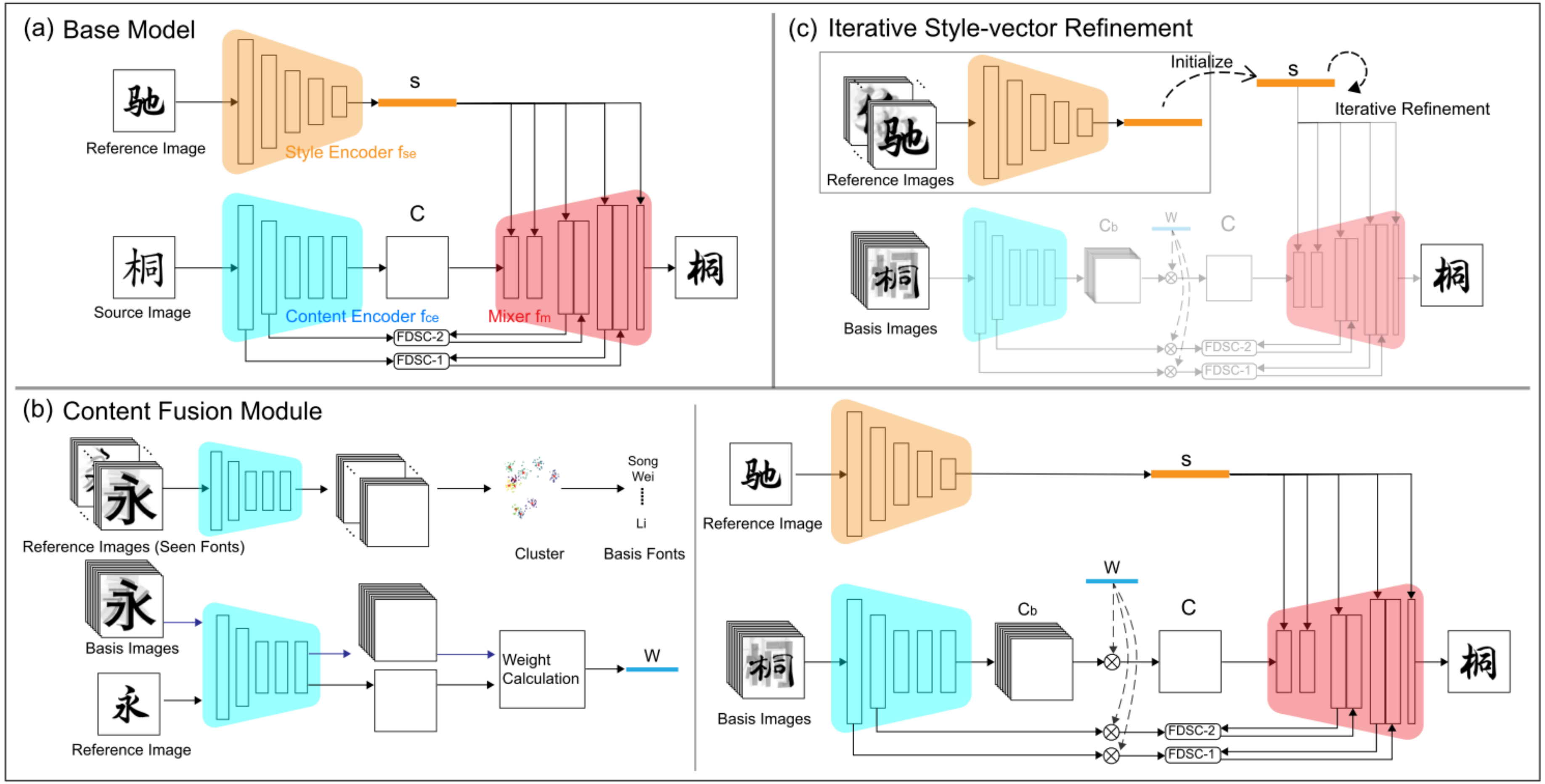


Figure 2. The framework of our model. (a) We first train the DGN [42] and use PCL to enhance the supervision of character skeletons. (b) After the model converges, content features of all training fonts are clustered and basis fonts are selected according to cluster centers. The original content encoder is replaced by CFM, and original content features are changed to fused features of basis fonts. Then we continue to train the model so that it adapts to fused content features. (c) In inference, we utilize ISR to polish the style of a font. The extracted mean style vector is treated as the only trainable variable to be fine-tuned for a few iterations.

Content Fusion Module

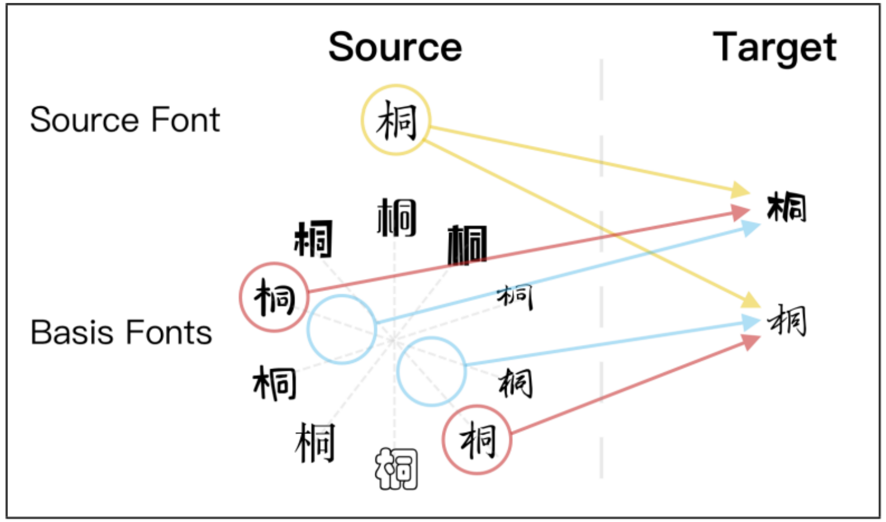


Figure 3. Visualization of content fusion. The yellow and red arrows are denoted for content features from the commonly used source font *Song* [20, 31, 42, 47] and the nearest font of the target respectively. The blue arrow represents the interpolation of content features of basis fonts to approximate the target.

Basis selection

$$\begin{aligned} \boldsymbol{C}_i &= f_{ce}(\boldsymbol{I}_i), \\ \boldsymbol{d}_i &= (d_{i1}, d_{i2}, ..., d_{iN}), \quad d_{ij} = \|\boldsymbol{C}_i - \boldsymbol{C}_j\|_1, \\ \boldsymbol{e}_i &= \sigma(\boldsymbol{d}_i), \\ \mathcal{B} &= \textbf{Cluster}(M, \{\boldsymbol{e}_1, \boldsymbol{e}_2, ..., \boldsymbol{e}_N\}), \end{aligned} \tag{1}$$

[1]

where $\sigma(\cdot)$ is the softmax operation, d_{ij} is the L1 distance between two fonts.

Weight calculation

$$\begin{aligned} \boldsymbol{d}'_t &= (d_{t1}, d_{t2}, ..., d_{tM}), \quad d_{tm} = \|\boldsymbol{C}_t - \boldsymbol{C}_m\|_1, \\ \boldsymbol{w}_t &= \sigma(-\boldsymbol{d}'_t/\tau), \end{aligned} \tag{2}$$

where τ is the temperature of the softmax operation.

Content fusion

$$\boldsymbol{C}'_t = \textstyle\sum_{m \in \mathcal{B}} w_{tm} \cdot \boldsymbol{C}_m. \tag{3}$$

Projected Character Loss(PCL)

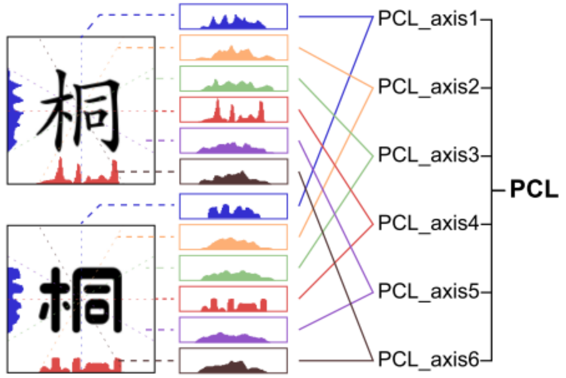


Figure 4. Illustration of PCL. We project the binary characters into multi-direction 1D spaces (distinguished by color) and calculate normalized histograms for each. It is obvious that for the different fonts with the same character, the projected distributions vary along with the skeletons and are less sensitive to textures or colors.

$$\mathcal{L}_p(\boldsymbol{Y}, \hat{\boldsymbol{Y}}) = \frac{1}{P} \sum_{p=1}^{P} \mathcal{L}_{1d}(\phi_p(\boldsymbol{Y}), \phi_p(\hat{\boldsymbol{Y}})), \qquad (4)$$

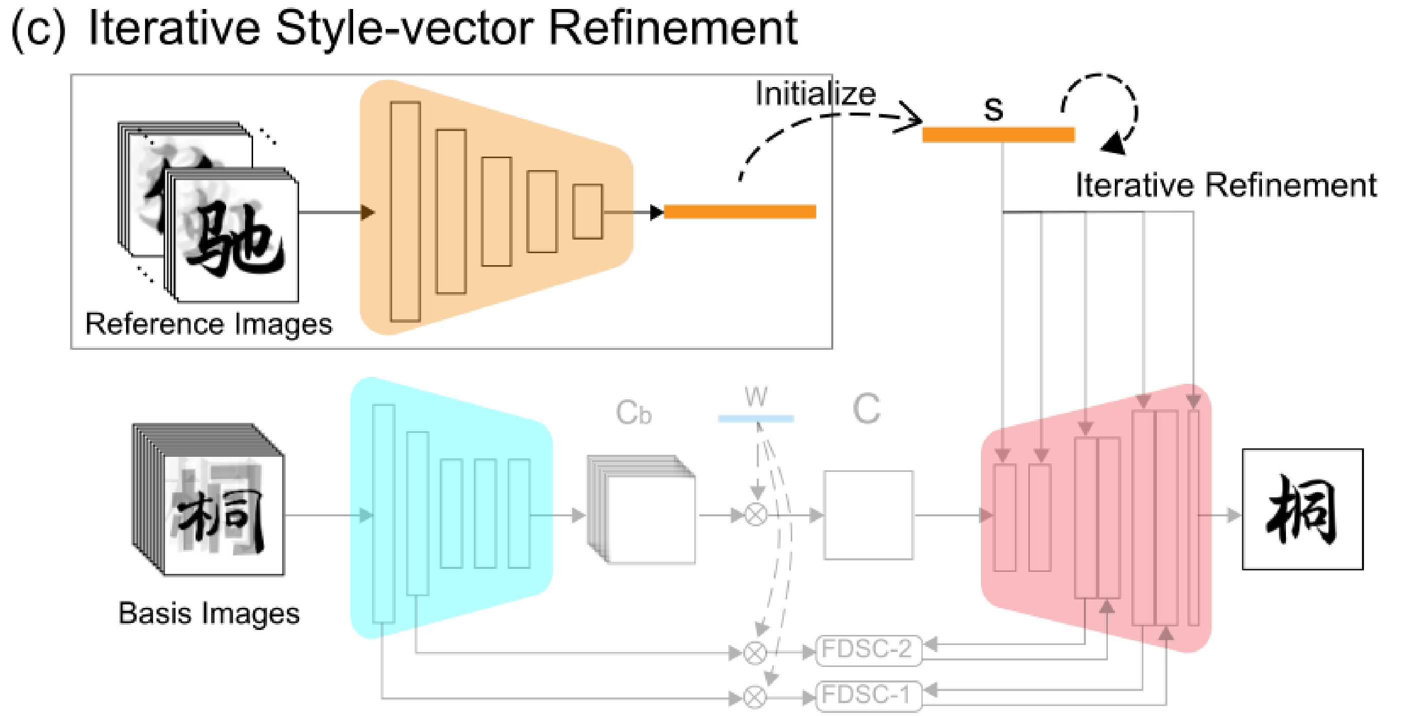
where $\boldsymbol{Y}$ and $\hat{\boldsymbol{Y}}$ represent the generated and ground-truth image respectively, P is the number of projections, and $\phi_p(\cdot)$ denotes a projection function with the p-th direction.

$$\mathcal{L}_{pc-wdl}(\boldsymbol{Y}, \hat{\boldsymbol{Y}}) = \frac{1}{P} \sum_{p=1}^{P} \left\| \frac{\Lambda(\phi_p(\boldsymbol{Y}))}{\sum \phi_p(\boldsymbol{Y})} - \frac{\Lambda(\phi_p(\hat{\boldsymbol{Y}}))}{\sum \phi_p(\hat{\boldsymbol{Y}})} \right\|$$

$$\mathcal{L}_{pc-kl}(\boldsymbol{Y}, \hat{\boldsymbol{Y}}) = \frac{1}{P} \sum_{p=1}^{P} \mathbf{KL}\left(\frac{\phi_p(\boldsymbol{Y})}{\sum \phi_p(\boldsymbol{Y})} - \frac{\phi_p(\hat{\boldsymbol{Y}})}{\sum \phi_p(\hat{\boldsymbol{Y}})}\right),$$

$$\mathcal{L} = \mathcal{L}_{adv} + \lambda_{img}(\mathcal{L}_{img} + \lambda_{pcl}\mathcal{L}_{pcl}) + \lambda_{cnt}\mathcal{L}_{cnt} + \lambda_{offset}\mathcal{L}_{offset},$$

Iterative Style-vector Refinement



$$s_t' = \frac{1}{Q}\sum_{q=1}^{Q} f_{se}(\boldsymbol{I}_t^q), \tag{7}$$

where $\boldsymbol{I}_t^q$ is an image of character q of font t, and Q denotes the reference character number.

Dataset

Total

- 300 字体
- 每个字体包含 6446 个汉字

Train

- 240 字体
- 每个字体 800 个汉字

Test

- Seen：229*5646
- Unseen：60*5646

Resolution：80×80

Source Font：11 个字体

Reference Images：随机 16 个训练集中字符

Experiments

Table 1. Comparison with state-of-the-art methods on seen/unseen fonts. Bold and underlined numbers denote the best and the second best respectively. The numbers in the last row represent our improvement over the second-best scores.

Methods	Seen Fonts					Unseen Fonts					User Study %
	L1↓	RMSE↓	SSIM ↑	LPIPS↓	FID↓	L1↓	RMSE↓	SSIM ↑	LPIPS↓	FID↓	
FUNIT	0.08591	0.2529	0.6661	0.1169	**11.66**	0.09377	0.2686	0.6432	0.1427	28.10	11.74
LF-Font	0.08098	0.2435	0.6829	0.1226	27.73	0.09037	0.2620	0.6534	0.1448	38.46	13.01
MX-Font	0.07470	0.2319	0.7038	0.1034	18.75	0.08171	0.2468	0.6830	0.1193	27.91	10.86
Fs-Font	0.08214	0.2519	0.6657	0.1502	45.33	0.08917	0.2657	0.6467	0.1647	55.21	12.03
CG-GAN	0.07977	0.2409	0.6883	0.1117	23.93	0.08639	0.2549	0.6690	0.1303	37.22	16.67
DG-Font	0.06251	0.2105	0.7437	0.0846	17.10	0.07841	0.2442	0.6853	0.1198	27.98	14.11
CF-Font	**0.05997**	**0.2053**	**0.7538**	**0.0836**	13.13	**0.07394**	**0.2354**	**0.7007**	**0.1182**	**26.51**	**21.58**
	(4.1%)	(2.5%)	(1.4%)	(1.1%)	(-)	(5.7%)	(3.6%)	(2.3%)	(0.92%)	(5.0%)	(29.5%)

Experiments

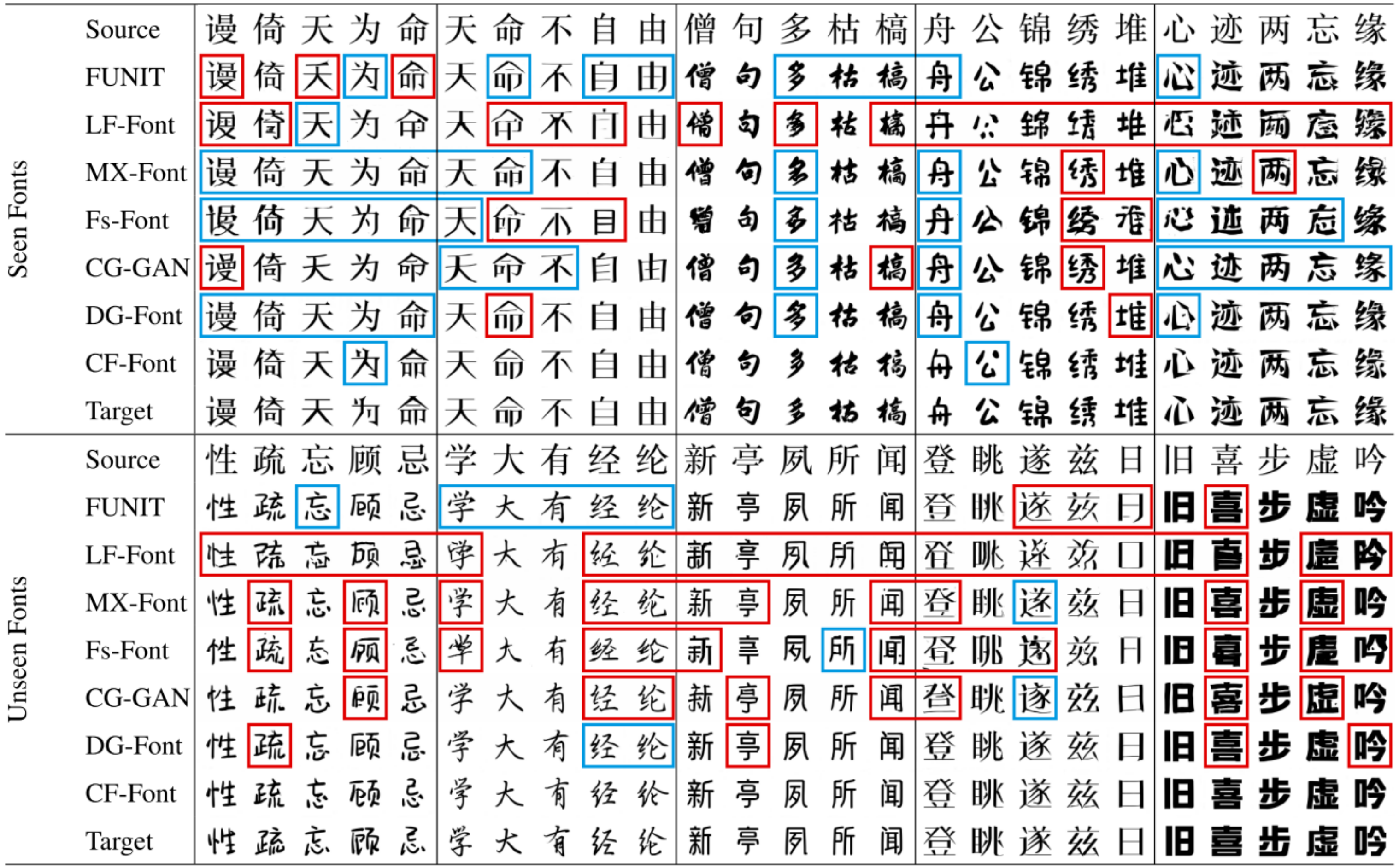


Figure 6. Qualitative comparison with state-of-the-art methods on Chinese poems. As mentioned earlier, we use multiple source fonts and pick the best results for these comparison methods for fairness. Here we just plot font *Song* as an example of source fonts for convenience. We mark erroneous skeletons with red boxes and other mismatch styles, such as stroke style, joined-up style, and body frame [26], with blue boxes.

Thanks for watching!

2023-06-28