7. Supplementary material

7.1. Effect parameter details

In the experiment, we assumed the most common three text effects: *fill*, *border*, and *shadow*. Fig 7 illustrates an example of those effects. We also assume the ordering of the effects is fixed to shadow $\rightarrow$ fill $\rightarrow$ border.

7.2. Illustration of alpha generation

We illustrate the process of alpha generation based on pre-rendered alpha maps in Fig 8.

7.3. Text style transfer examples

As shown in Fig.6, the proposed model can use the text style of another text on the rendering step. Here, we show more detailed examples of text style transfer, as Fig.7. Five different styles are transferred to a text (e.g., "FREE" and "BACON") in an input image. Note again that the proposed model can transfer not just font style but also effect style.

7.4. Text image generator details

We use a text image generator, which is modified from SynthText, and record the exact rendering parameter used in the text image generator to supervise the training of our model. Here, we introduce our text image generator details, in terms of the type of the background images, text placement rule, and sampling rule of the rendering parameter.

The background images originally used in SynthText are insufficient for the robust performance on the display media; we therefore add background images from single-color background data, Book cover dataset, BAM dataset, and FMD dataset. For those five types of background images, we render texts and place texts in background images.

The rule of the text placement procedure is different in the type of the background images, respectively. For the background images from SynthText, the text placement procedure follows SynthText; we apply over-segmentation to generate candidate regions to place a text. Note that we disable text rotation for the function of SynthText. For BAM and Book cover data, we utilize a saliency map to generate candidate locations. For Book cover data, we also generate candidate locations by applying the existing OCR model. Unlike other background images, BAM and Book cover images often include texts, then we apply text inpainting to erase texts in advance. We randomly generate text locations for single-color flat backgrounds and FMD images. On FMD, we crop material regions and use those regions as backgrounds. We generate 10,000 text rendered images for the background images, respectively. Then we exclude images that have too-small candidate regions for locating texts, and finally obtained a total of 42,285 images.

After generating candidate regions, we set rendering parameters. Font categories are randomly sampled from predefined categories. The effect parameters and colors are randomly sampled from a predefined range. Unlike SynthText, we implement our data generator using the Skia graphics library[1], which can handle both font and effects without raster artifacts.

7.5. Architecture details

We show the detailed configurations for the parser models in Table 3. Our encoder model, i.e. backbone model, is based on an hour-glass model. We add some convolution layers to the outputs' head to enlarge the receptive fields because texts tend to be large in the display media. There are branches for predicting text rendering parameters: the OCR branch, the alpha branch, the font branch, the effect visibility brach, and the effect parameter branch. The OCR branch is further split into the word detector, the character detector, and the character classifier. For extracting text colors, we predict pixel-wise alpha maps to decompose an image. We obtain font information for each text by a classification model. To parse text effects, we predict both effects visibility and effects parameters. We consider visibility prediction as a binary classification problem and shadow parameter estimation as a regression problem. We predict discretized parameters by a classification model for border effects because we use pre-rendered alpha maps for them. We quantize the border parameter into five bins in our experiments.

In Table3, the third "kernel and stride" column indicate the kernel size and the stride in the convolution layer if the layer has those configurations. The three numerical values in both columns of input size and output size represent tensors' size for channel, height, and width. We represent the intermediate representation in inputs and outputs by B1-B5 for the backbone model, W1-W3 for the word detector, C1-C3 for the character detector, R1-R3 for the character classifier, A1-A6 for the alpha model, F1 for the font model, V1 for the effects visibility model, and P1 for the effects parameter model.

[1] https://skia.org/

(a) Composited (b) Fill

(c) Border (d) Shadow

Figure 7: Effect examples.

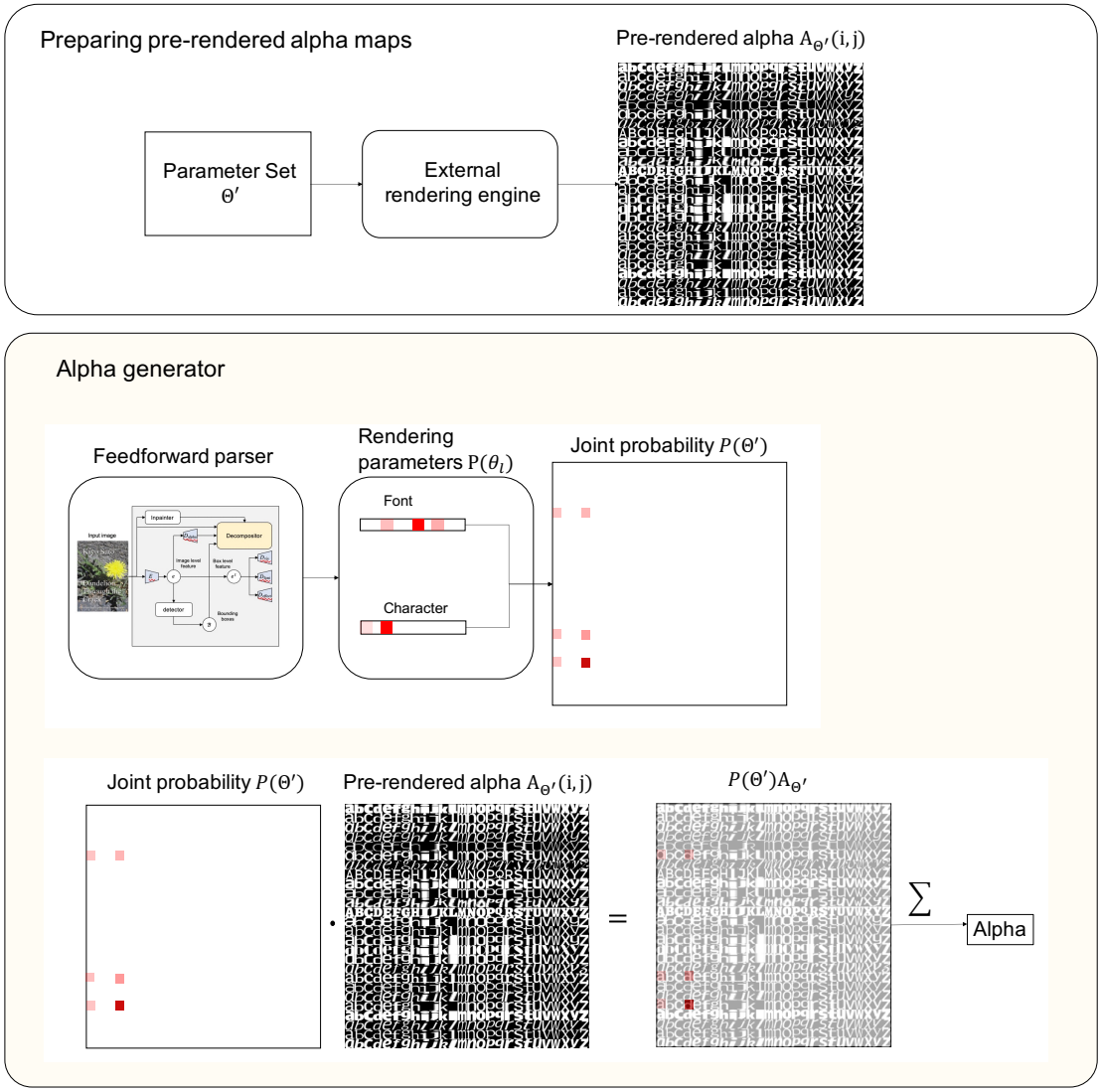

Figure 8: Alpha generator using pre-rendered alpha maps. We illustrate the case where we have 26 characters and fonts.

Figure 9: Examples of text style transfer in an external renderer. Since we transfer style information in parameter space, we produce no pixel artifacts such as blur on texts.

Table 3: Architecture details.

Model	Layers	Kernel, Stride	Input	Input size	Output	Output Size
Backbone	Hour Glass Net	-	x	$3 \times \mathrm{H} \times \mathrm{W}$	B1	$256 \times \mathrm{H}/4 \times \mathrm{W}/4$
	CONV + BN + RELU	(3×3), (2×2)	B1	$256 \times \mathrm{H}/4 \times \mathrm{W}/4$	B2	$128 \times \mathrm{H}/8 \times \mathrm{W}/8$
	CONV + BN + RELU	(3×3), (2×2)	B2	$128 \times \mathrm{H}/8 \times \mathrm{W}/8$	B3	$128 \times \mathrm{H}/16 \times \mathrm{W}/16$
	CONV + BN + RELU	(3×3), (2×2)	B3	$128 \times \mathrm{H}/16 \times \mathrm{W}/16$	B4	$128 \times \mathrm{H}/32 \times \mathrm{W}/32$
	Upsampling	-	B4	$128 \times \mathrm{H}/32 \times \mathrm{W}/32$	B5	$128 \times \mathrm{H}/4 \times \mathrm{W}/4$
	CONV + BN + RELU	(3×3), (1×1)	B1 and B5	$384 \times \mathrm{H}/4 \times \mathrm{W}/4$	$e(x)$	$256 \times \mathrm{H}/4 \times \mathrm{W}/4$
Word detector	CONV + BN + RELU	(3×3), (1×1)	$e(x)$	$256 \times \mathrm{H}/4 \times \mathrm{W}/4$	W1	$128 \times \mathrm{H}/4 \times \mathrm{W}/4$
	CONV + BN + RELU	(3×3), (1×1)	W1	$128 \times \mathrm{H}/4 \times \mathrm{W}/4$	W2	$32 \times \mathrm{H}/4 \times \mathrm{W}/4$
	CONV + BN + RELU	(3×3), (1×1)	W1	$128 \times \mathrm{H}/4 \times \mathrm{W}/4$	W3	$32 \times \mathrm{H}/4 \times \mathrm{W}/4$
	CONV	(1×1), (1×1)	W2	$128 \times \mathrm{H}/4 \times \mathrm{W}/4$	Text foreground map	$2 \times \mathrm{H}/4 \times \mathrm{W}/4$
	CONV + RELU	(1×1), (1×1)	W3	$128 \times \mathrm{H}/4 \times \mathrm{W}/4$	Text geometry map	$32 \times \mathrm{H}/4 \times \mathrm{W}/4$
Char detector	CONV + BN + RELU	(3×3), (1×1)	$e(x)$	$256 \times \mathrm{H}/4 \times \mathrm{W}/4$	C1	$128 \times \mathrm{H}/4 \times \mathrm{W}/4$
	CONV + BN + RELU	(3×3), (1×1)	C1	$128 \times \mathrm{H}/4 \times \mathrm{W}/4$	C2	$32 \times \mathrm{H}/4 \times \mathrm{W}/4$
	CONV + BN + RELU	(3×3), (1×1)	C1	$128 \times \mathrm{H}/4 \times \mathrm{W}/4$	C3	$32 \times \mathrm{H}/4 \times \mathrm{W}/4$
	CONV	(1×1), (1×1)	C2	$128 \times \mathrm{H}/4 \times \mathrm{W}/4$	Char foreground map	$32 \times \mathrm{H}/4 \times \mathrm{W}/4$
	CONV + RELU	(1×1), (1×1)	C3	$128 \times \mathrm{H}/4 \times \mathrm{W}/4$	Char geometry map	$32 \times \mathrm{H}/4 \times \mathrm{W}/4$
Char recognizer	CONV + BN + RELU	(3×3), (1×1)	$e(x)$	$256 \times \mathrm{H}/4 \times \mathrm{W}/4$	R1	$128 \times \mathrm{H}/4 \times \mathrm{W}/4$
	CONV + BN + RELU	(3×3), (1×1)	R1	$128 \times \mathrm{H}/4 \times \mathrm{W}/4$	R2	$32 \times \mathrm{H}/4 \times \mathrm{W}/4$
	CONV + BN + RELU	(3×3), (1×1)	R2	$128 \times \mathrm{H}/4 \times \mathrm{W}/4$	R3	$32 \times \mathrm{H}/4 \times \mathrm{W}/4$
	CONV	(1×1), (1×1)	R3	$94 \times \mathrm{H}/4 \times \mathrm{W}/4$	Char recognition map	$32 \times \mathrm{H}/4 \times \mathrm{W}/4$
Alpha	CONV + BN + RELU	(3×3), (1×1)	$e(x)$	$256 \times \mathrm{H}/4 \times \mathrm{W}/4$	A1	$128 \times \mathrm{H}/4 \times \mathrm{W}/4$
	CONV + BN + RELU	(3×3), (1×1)	A1	$128 \times \mathrm{H}/4 \times \mathrm{W}/4$	A2	$32 \times \mathrm{H}/4 \times \mathrm{W}/4$
	Upsampling	-	A2	$32 \times \mathrm{H}/4 \times \mathrm{W}/4$	A3	$32 \times \mathrm{H} \times \mathrm{W}$
	CONV + BN + RELU	(3×3), (1×1)	x	$3 \times \mathrm{H} \times \mathrm{W}$	A4	$32 \times \mathrm{H} \times \mathrm{W}$
	CONV + BN + RELU	(3×3), (1×1)	A3 and A4	$64 \times \mathrm{H} \times \mathrm{W}$	A5	$32 \times \mathrm{H} \times \mathrm{W}$
	CONV + BN + RELU	(3×3), (1×1)	A5	$32 \times \mathrm{H} \times \mathrm{W}$	A6	$32 \times \mathrm{H} \times \mathrm{W}$
	CONV + Sigmoid	(3×3), (1×1)	A6	$32 \times \mathrm{H} \times \mathrm{W}$	Alpha for decomposition	$3 \times \mathrm{H} \times \mathrm{W}$
Font	CONV + BN + RELU	(3×3), (1×1)	$e_b^t(x)$	$256 \times 1 \times 1$	F1	$128 \times 1 \times 1$
	CONV	(1×1), (1×1)	F1	$128 \times 1 \times 1$	Font categories	$100 \times 1 \times 1$
Effects visiblity	CONV + BN + RELU	(3×3), (1×1)	$e_b^t(x)$	$256 \times 1 \times 1$	V1	$128 \times 1 \times 1$
	CONV	(1×1), (1×1)	V1	$128 \times 1 \times 1$	Shadow visibility	$2 \times 1 \times 1$
	CONV	(1×1), (1×1)	V1	$128 \times 1 \times 1$	Border visibility	$2 \times 1 \times 1$
Effects params	CONV + BN + RELU	(3×3), (1×1)	$e_b^t(x)$	$256 \times 1 \times 1$	P1	$128 \times 1 \times 1$
	CONV + Tanh	(1×1), (1×1)	P1	$128 \times 1 \times 1$	Shadow offset	$2 \times 1 \times 1$
	CONV + Sigmoid	(1×1), (1×1)	P1	$128 \times 1 \times 1$	Shadow blur	$1 \times 1 \times 1$
	CONV	(1×1), (1×1)	P1	$128 \times 1 \times 1$	Border weights	$5 \times 1 \times 1$