Supplementary Material: Selective Scene Text Removal

Hayato Mitani[1]
hayato.mitani@human.ait.kyushu-u.ac.jp
Akisato Kimura[2]
akisato@ieee.org
Seiichi Uchida[1]
uchida@ait.kyushu-u.ac.jp

[1] Kyushu University, Fukuoka, Japan
[2] NTT Corporation, Kanagawa, Japan

A Implementation Details

This section details the implementation of the proposed method. Batch normalization was used for each layer of the four modules. The background extraction module uses Leaky ReLU for the activation function, whereas the text extraction module and the reconstruction module use ReLU. The selective word removal module uses Leaky ReLU in the encoder layers, ReLU in the decoder layers, and Sigmoid in the final layer. The batch size was 256 for the selective word removal module and 32 for the other modules. The batch size 32 was also used in fine-tuning. Adam was used as the optimizer; the learning rate was set at 0.0001 for individual module training and 0.00001 for end-to-end fine-tuning. We used early stopping by the validation loss.

During training and testing, we assume removing a single target word $\omega \in \mathbf{\Omega}$. There are two reasons for this assumption. First, it makes the setup and evaluation of experiments very simple and straightforward. Second, repetitive use of the single target word removal is practically equivalent to the multiple target word removal. For example, if we want to remove two words *A* and *B*, we first remove *A* and then *B*.

B Architecture of U-Nets

The individual U-Net Modules have the following architecture. Background Extraction Module: $C_{3,64,1}$-$C_{2,128,2}$-$C_{2,256,2}$-$C_{2,512,2}$-R-R-R-R-$D_{2,256,2}$-$D_{2,128,2}$-$D_{2,64,2}$-$C_{3,3,1}$. Text Extraction Module: $C_{3,64,1}$-M-$C_{3,128,1}$-M-$C_{3,256,1}$-M-$C_{3,512,1}$-M-$C_{3,1024,1}$-$D_{2,512,2}$-$D_{2,256,2}$-$D_{2,128,2}$-$D_{2,64,2}$-$C_{1,4,1}$. Selective Word Removal Module: $C_{5,64,2}$-$C_{5,128,2}$-$C_{5,256,2}$-$C_{5,512,2}$-$C_{5,1024,2}$-$C_{5,2048,2}$-$D_{5,1024,2}$-$D_{5,512,2}$-$D_{5,256,2}$-$D_{5,128,2}$-$D_{5,64,2}$-$D_{5,4,2}$. Reconstruction Module: $C_{3,64,1}$-M-$C_{3,128,1}$-M-$C_{3,256,1}$-M-$C_{3,512,1}$-M-$C_{3,1024,1}$-$D_{2,512,2}$-$D_{2,256,2}$-$D_{2,128,2}$-$D_{2,64,2}$-$C_{1,3,1}$. Here, $C_{i,c,s}$ denotes a convolutional layer of a c-channel $i \times i$ filter with stride s. $D_{i,c,s}$ denotes a deconvolutional layer with the same suffixes as $C_{i,c,s}$. R is a residual layer, and M is a max-pooling layer with a 2×2 pooling window and 2-stride. Skip connections are prepared between the corresponding C and D as the standard U-Net.

Figure 9: SSTR results with different target words.

C Additional Experimental Results

Fig. 9 shows how the proposed model responds to different target words. In each of the figures (a), (b), and (c), the target words are “France,” “China,” and “Germany.” Also in figures (d), (e), and (f), the target words are “China,” “Germany,” and “India.” In (a), three target words are located close to each other. Even in such a difficult case, the proposed model could remove only the target word selectively. In (b), also the three target words are densely clustered, but only the target word was successfully removed. In (c), the three target words are located far from each other, but of course, only the target word was successfully removed. In (d), (e), and (f), it can be seen that the proposed model is successful in removal for each of the different target words as well.

Fig. 10 hows an SSTR result where the target word appears two or three times times.

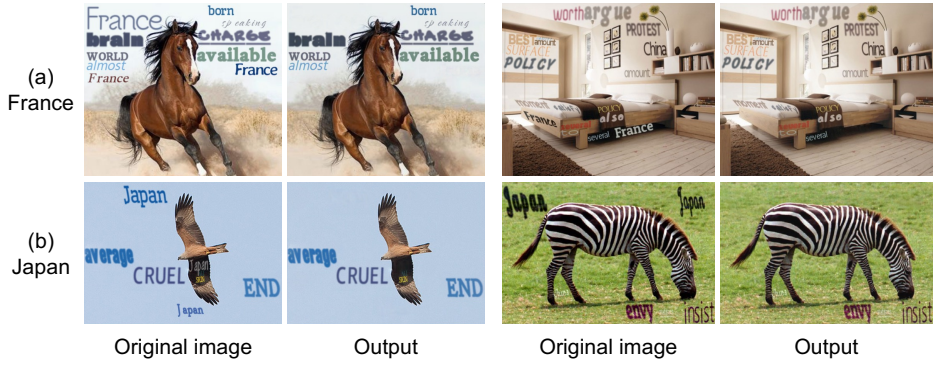

Figure 10: SSTR results on multiple appearances of the target word.

Figure 11: SSTR results to confirm the case-sensitivity. In addition to the target word "China," the non-target word "china" appears.

In the left image in (a), the target word "France" appears three times. The result shows that all "France" words are removed successfully. We just use the same trained model as the experiments in the current paper. Namely, we did not retrain our model to deal with the multiple appearances. Nevertheless, our proposed model succeeds in removing all target words.

Fig. 11 shows an SSTR result where "China" and "china" appear, and the former is the target. Again, we did not retrain our model for this result. Even though 'c' and 'C' are similar, our model *successfully* distinguishes them, and only the target word is removed. Case sensitivity is important for SSTR's higher reliability by not removing non-target words.