Smooth-Swap: A Simple Enhancement for Face-Swapping with Smoothness

Jiseob Kim\textsuperscript{1,2}, Jihoon Lee\textsuperscript{2}, Byoung-Tak Zhang\textsuperscript{1}
\textsuperscript{1}Seoul National University, \textsuperscript{2}Kakao Brain
jkim@bi.snu.ac.kr, jihoonlee.in@gmail.com, btzhang@bi.snu.ac.kr

Contents

A Architecture Details
\hspace{1em} A.1 Identity Embedding Model: $f_{\text{emb}}$ \hspace{1em} 1
\hspace{1em} A.2 Swap-Image Generator: $f_{\text{gen}}$ \hspace{1em} 1
\hspace{1em} A.3 Discriminator: $f_{\text{dis}}$ \hspace{1em} 2

B More Image Samples from Smooth-Swap \hspace{1em} 2

C Extreme Cases and Limitations \hspace{1em} 2

List of Figures

A1 Detailed Architecture \hspace{1em} 3
A2 Failure Cases \hspace{1em} 3
A3 More FaceForensics++ Results \hspace{1em} 4
A4 More FFHQ Results \hspace{1em} 5
A5 Face-Swapping on Metfaces \hspace{1em} 6
A6 More Wild-image Results \hspace{1em} 7

A. Architecture Details

A.1. Identity Embedding Model: $f_{\text{emb}}$

Our identity embedding model is based on ResNet-50 [3], where we use a different head for contrastive learning as seen in Fig. A1. Note the UnitNorm in the final layer makes $z_{\text{src}}$ to be unit-length ($\|z_{\text{src}}\| = 1$). The network involves total 32.3M parameters.

A.2. Swap-Image Generator: $f_{\text{gen}}$

Our generator architecture is mostly the same as NCSN++ [8] except for the following three differences (as described in the main manuscript, Sec. 4.2): 1) we use half as many channels, 2) we use the identity embedding instead of the time embedding, and 3) we add an input-to-output skip connection. Fig. A1 (a) shows the detailed structure with dimensional information. The network involves total 9.8M parameters.

Up/Down Sampling & Skip-Connections

Note in each of the outer block containing multiple ResBlocks, the first ResBlock handles upsampling or downsampling (except for the ResBlock x5, where the second ResBlock handles upsampling). There are 13 skip connections in total ($13 = 3 \times 4 + 1$; $+1$ is the input-to-output skip), where the input to each of the ResBlock in the encoder part (before the Attention Block) is handed over to the decoder part (after the Attention Block). On the decoder side, the first three ResBlocks of each outer block get the skip-connections (except for the ResBlock x5, where the second through the fourth get the skip-connections).

Details on the ResBlocks of the Generator

We describe some essential details of the ResBlocks of the generator here. The complete information can be found in [8].

The overall structure of ResBlock is not much different from the conventional design [3]. However, as shown in Fig. A1 (b), a structure for conditioning on the identity embedding vector $z_{\text{src}}$ is added (similar to [1]). The conditioning is done by 1) projecting $z_{\text{src}}$ onto a $c_{\text{out}}$-dimensional vector, 2) spatially broadcasting the result, and 3) adding it to the intermediate output of the original path ($c_{\text{out}}$ is the number of output channels of the current block). When upsampling or downsampling is used, the optional components (denoted by yellow and dash-dotted outline) are also computed.

Throughput and FLOPs at Inference Time

Smooth-Swap generator has much higher FLOPs (in MACs) than HifiFace [9] (214.47G to 102.39G). However, it shows far better throughput (42.96 fps) than others (SimSwap [2]: 31.17, HifiFace [9]: 25.29; FaceShifter [6]: 22.34)\footnote{Test settings and values of other models are adopted from [9]}. We speculate this is due to the simple and homogeneous architecture, which is advantageous for speed-up with GPU computing.
A.3. Discriminator: $f_{\text{dis}}$

We use the same discriminator as the one used in StyleGAN2 [5]. The network involves total 28.9M parameters.

B. More Image Samples from Smooth-Swap

We show extended sets of swapped-image samples from our Smooth-Swap model. The following three figures, Fig. A3, A4, and A6 present the results of the same experiments as Fig. 4, 5, and 6 in the main manuscript, but with different source and target pairs. Fig. A5 shows the results for out-of-distribution cases, where oil paintings (Metfaces dataset [4]) are used for swapping. Although the model is never trained on such images, the results are of decent quality, reflecting the characteristics of the source and the target with shape change.

C. Extreme Cases and Limitations

We note that Smooth-Swap can fail when a target image involves occlusion or an extreme pose as shown in Fig. A2. However, we believe each case can be handled by post-processing (e.g., HEAR-Net of [6]) and supplying more extreme-pose examples for training.

References

Figure A1. Detailed architecture of our Smooth-Swap model; both the identity embedder and the generator are shown. The intermediate feature-map dimensions are written in the order of (channels × height × width). ‘ResBlock x4’ in (a) denotes that there are four residual sub-blocks connected sequentially; the structure of the sub-block (i.e., ResBlock) is detailed in (b). Note the multi-line text inside blocks should be read from the bottom (e.g., Linear(512) to BatchNorm to UnitNorm for the last embedder block). See Sec. A.2.

Figure A2. Some failure cases of Smooth-Swap. See Sec. C
Figure A3. Comparison of the face-swapping results of various models on the FaceForensics++ dataset [7] (extension of Fig. 4 in the main manuscript)
Figure A4. More results of Smooth-Swap on the FFHQ test split (extension of Fig. 5 in the main manuscript). Active change of identity is observed. However, in some cases where the source and the target have largely different face shapes (e.g., a child in the rightmost column in the lower-right block), artifacts are noticed. In real-world applications, such cases can be avoided by choosing the swapping pairs from a similar age range.
Figure A5. Results of Smooth-Swap across the FFHQ test split and Metfaces [4]. Even though the model is not trained on the oil paintings of Metfaces, it can still produce swapped images with a decent quality.
Figure A6. More face swapping results of Smooth-Swap on wild images (extension of Fig. 6 in the main manuscript).