Supplementary Material of Semantic Component Decomposition for Face Attribute Manipulation

Ying-Cong Chen[1] Xiaohui Shen[4] Zhe Lin[3] Xin Lu[3] I-Ming Pao[3] Jiaya Jia[1,2]
[1]The Chinese University of Hong Kong [2]Tencent Youtu Lab [3]Adobe Research [4]ByteDance AI Lab
{ycchen,leojia}@cse.cuhk.edu.hk shenxiaohui@gmail.com {zlin, xinl, pao}@adobe.com

1. Implementation Details

The architecture of our model is shown in Fig. 1. The design details are shown below. Our code will be released later.

Encoder We use the convolutional part of VGG-19 network as our encoder. The weights are pretrained with ImageNet [2]. Following [3, 1], the ReLU3_1, ReLU4_1 and ReLU5_1 are used to form our feature space $\phi(\cdot)$.

Painter Network As shown in Fig. 1(a), the Painter Network takes input of VGG feature of the image $\phi(I_{\mathcal{S}^-})$, and produces k Painter vectors $\mathcal{P}_i$, $i = 1, 2, \cdots k$. After the global average pooling layer, the network produces a $512 \times k$ vector. This vector is sliced to k painter vectors whose dimension is 512.

Region Network Fig. 1(b) shows the architecture of the Painter Network, which takes $\phi(I_{\mathcal{S}^-})$ as input, then produces k attention maps. We use 3 convolutional layers to produce a k channel feature map, then pass it to the SoftMax2D layer. As indicated in Section 3.2 of the paper, this encodes each location to a soft one-hot vector and makes the k channels towards non-overlapping activation regions. The k channels are then sliced to k attention maps.

Fusion Network The Fusion Networks shown in Fig. 1(d) takes the concatenation result of painter vector $\mathcal{P}_i$ (spatially repeated to $\mathbb{R}^{h\times w\times 512}$) and $\mathcal{M}_i$ to estimate the corresponding i^{th} component $\mathcal{V}_{\mathcal{S}_i}$. The attribute vector $\mathcal{V}_{\mathcal{S}}$ is then estimated by summing up all components.

Decoder As shown in Fig. 1(c), the decoder is trained to convert the edited feature $\phi(I_{\mathcal{S}^+})$ to the image space, which is formulated as

$$\mathcal{L} = \mathbb{E}(||\phi(D(\phi(I_{\mathcal{S}^+})) - \phi(I_{\mathcal{S}^+})||^2), \quad (1)$$

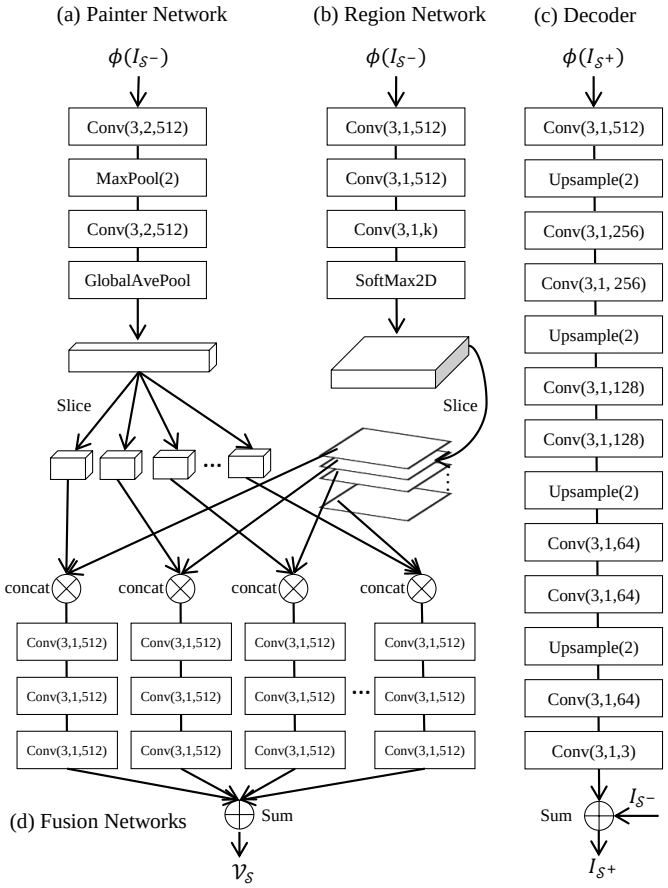


Figure 1. The architecture of our decoder. (a-d) shows the architecture of Painter Network, Region Network, Decoder and Fusion Networks respectively. Conv(x,y,z) denotes a convolutional layer whose kernel size is x, stride is y and channel number is z. Note that all convolutional layers are followed by ReLU activation layer. MaxPool(2) denotes maxpooling with stride 2. Upsample(2) denotes 2x nearest neighbor upsampling. SoftMax2D means passing each location of the feature map to a SoftMax function.

where D denotes the decoder, and $\mathbb{E}$ denotes averaging among the training set. Note that we add the original image $I_{\mathcal{S}^-}$ to the last layer. In this way, the decoder only needs

to learn the edited part rather then the whole image, which simplifies the learning task.

2. Face Edit Interface

The attached video illustrates how interactive face manipulation works. Users can manipulate the attribute of the face image in different ways, i.e., manipulating the global edit strength, adjusting the strength of each component, or drawing on $\mathcal{M}_i$ to adjust the edited region.

3. Details of the Quantitative Evaluation

Fig. 2 shows a snapshot of the A/B test. Subjects are presented with one original image, and two edited one with a randomized order. Images are presented in one row so that subjects can do careful comparison.

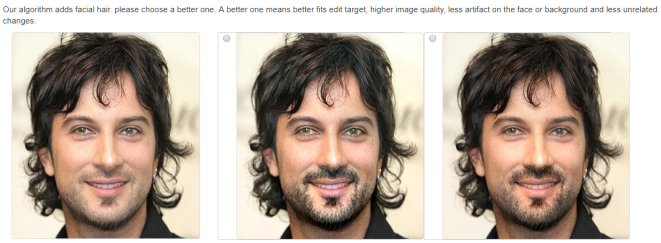


Figure 2. Our algorithm adds facial hair. please choose a better one. A better one means better fits edit target, higher image quality, less artifact on the face or background and less unrelated changes.

4. More results

We have also presented more results in the following pages, including more cases comparing to the state-of-the-art and more attribute edits with our approach.

References

[1] Y.-C. Chen, H. Lin, M. Shu, R. Li, X. Tao, Y. Ye, X. Shen, and J. Jia. Facelet-bank for fast portrait manipulation. In *CVPR*, 2018. 1

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 1

[3] P. Upchurch, J. Gardner, K. Bala, R. Pless, N. Snavely, and K. Weinberger. Deep feature interpolation for image content changes. In *CVPR*, 2017. 1

Original Ours Facelet DFI CycleGAN ResGAN

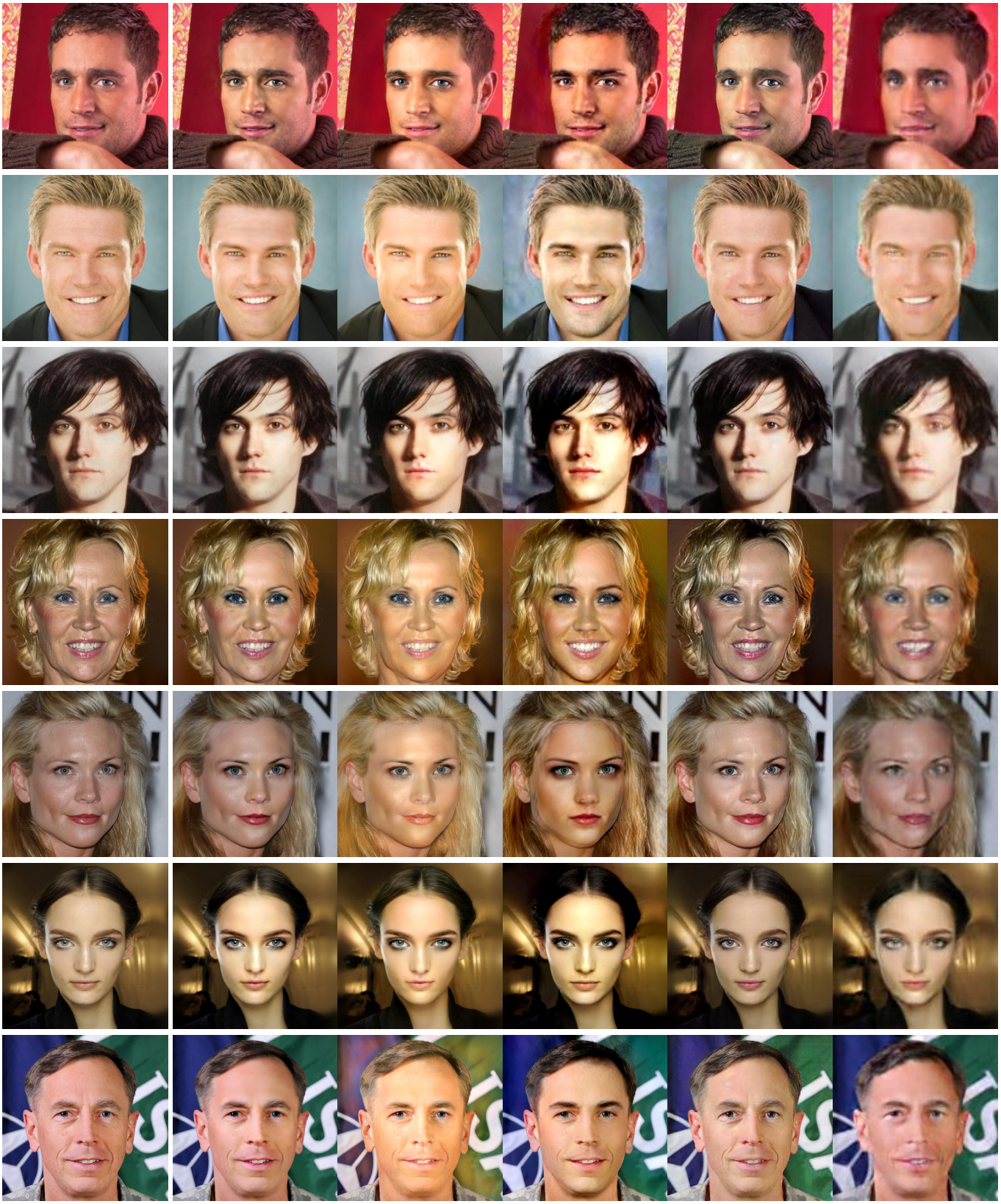
Figure 3. Comparison with state-of-the-art approaches of the "younger" attribute. **Please zoom in to see more details.**

Original Ours Facelet DFI CycleGAN ResGAN

Figure 4. Comparison with state-of-the-art approaches of the "older" attribute. **Please zoom in to see more details.**

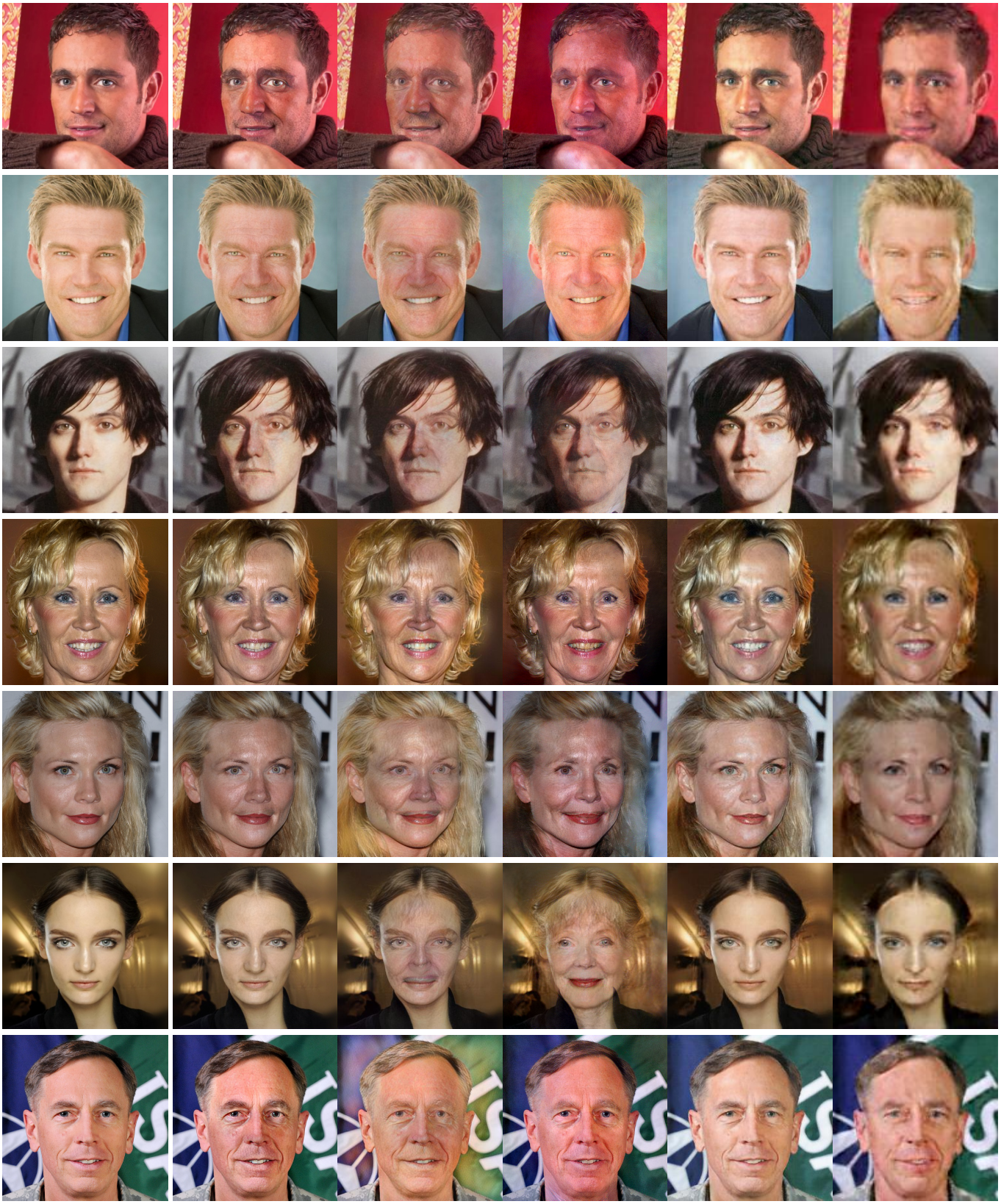
Original Ours Facelet DFI CycleGAN ResGAN

Figure 5. Comparison with state-of-the-art approaches of the "facehair" attribute. **Please zoom in to see more details.**



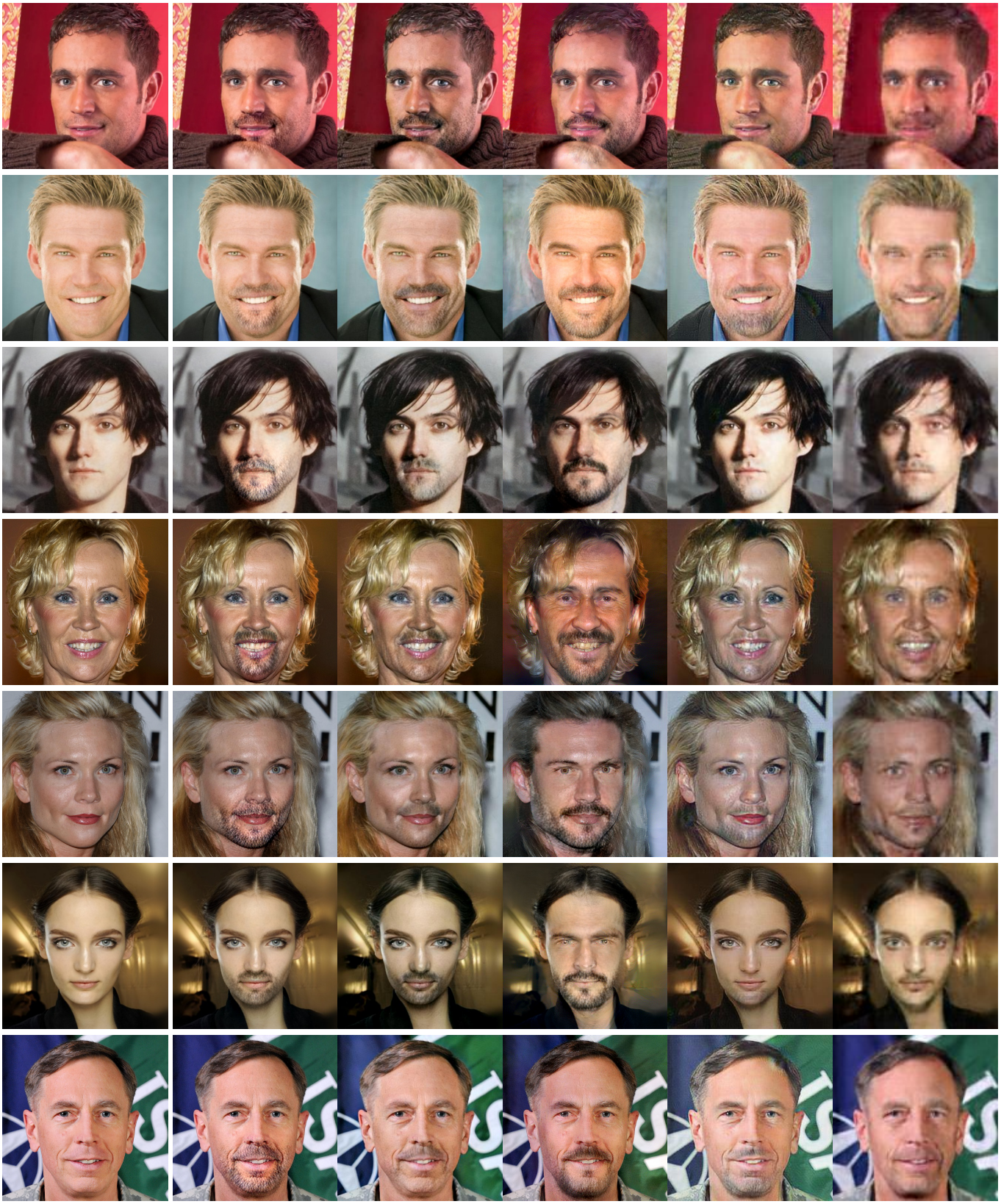
Original Ours Facelet DFI CycleGAN ResGAN

Figure 6. Comparison with state-of-the-art approaches of the "younger" attribute. **Please zoom in to see more details.**



Original Ours Facelet DFI CycleGAN ResGAN

Figure 7. Comparison with state-of-the-art approaches of the "older" attribute. **Please zoom in to see more details.**



Original Ours Facelet DFI CycleGAN ResGAN

Figure 8. Comparison with state-of-the-art approaches of the "facehair" attribute. **Please zoom in to see more details.**

Figure 9. Other attribute manipulation results.**Please zoom in to see more details.**

Original | Bushy Eyebrows | Eyewear | Attractive | Smiling | Pale Skin

Figure 10. Other attribute manipulation results.**Please zoom in to see more details.**

Figure 11. Other attribute manipulation results.**Please zoom in to see more details.**