Supplementary Document for
"PISE: Person Image Synthesis and Editing with Decoupled GAN"

In this document, we provide the following supplementary contents:

- Implementation details.
- Comparison with state-of-the-art methods.
- Texture transfer results.
- Texture interpolation results.
- Region editing results.
- Failure cases and analysis.
- Results on images in the wild.

1. Implementation Details.

In our parsing generator and image generator, we use several blocks shown in Figure 1. The detailed architecture of our model is shown in Figure 2. Our method is implemented in PyTorch. We train and test our code using a GeForce RTX 2080Ti with 11 GB memory. The weights in the loss function ($\lambda_{p\ell}$,λ_c, λ_ℓ, λ_p, λ_s, λ_a) are set to (100.0, 30.0, 5.0, 0.5, 200.0, 2.0). We train our network in stages. The parsing generator is first trained to generate a human parsing map aligned with the target pose. Then, we train our model without spatial-aware normalization to constrain the generated image features and the real image features in the same domain. Finally, we train our full model in an end-to-end manner. We adopt the Adam optimizer [1] to train our network with the learning rate 0.0001 for the generator and 0.00001 for the discriminator.

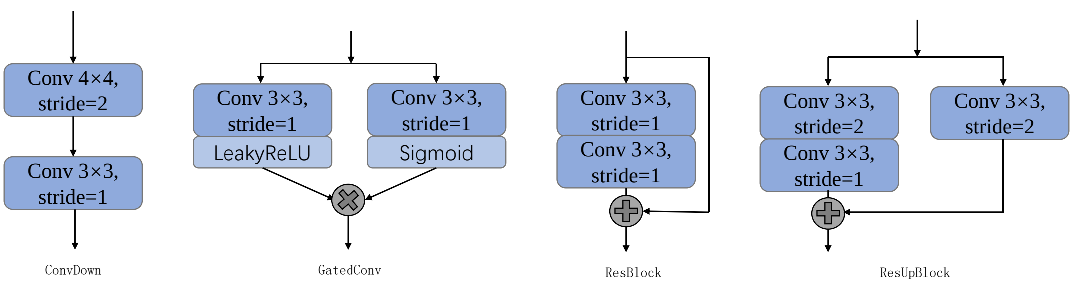

Figure 1: Several blocks used in our model.

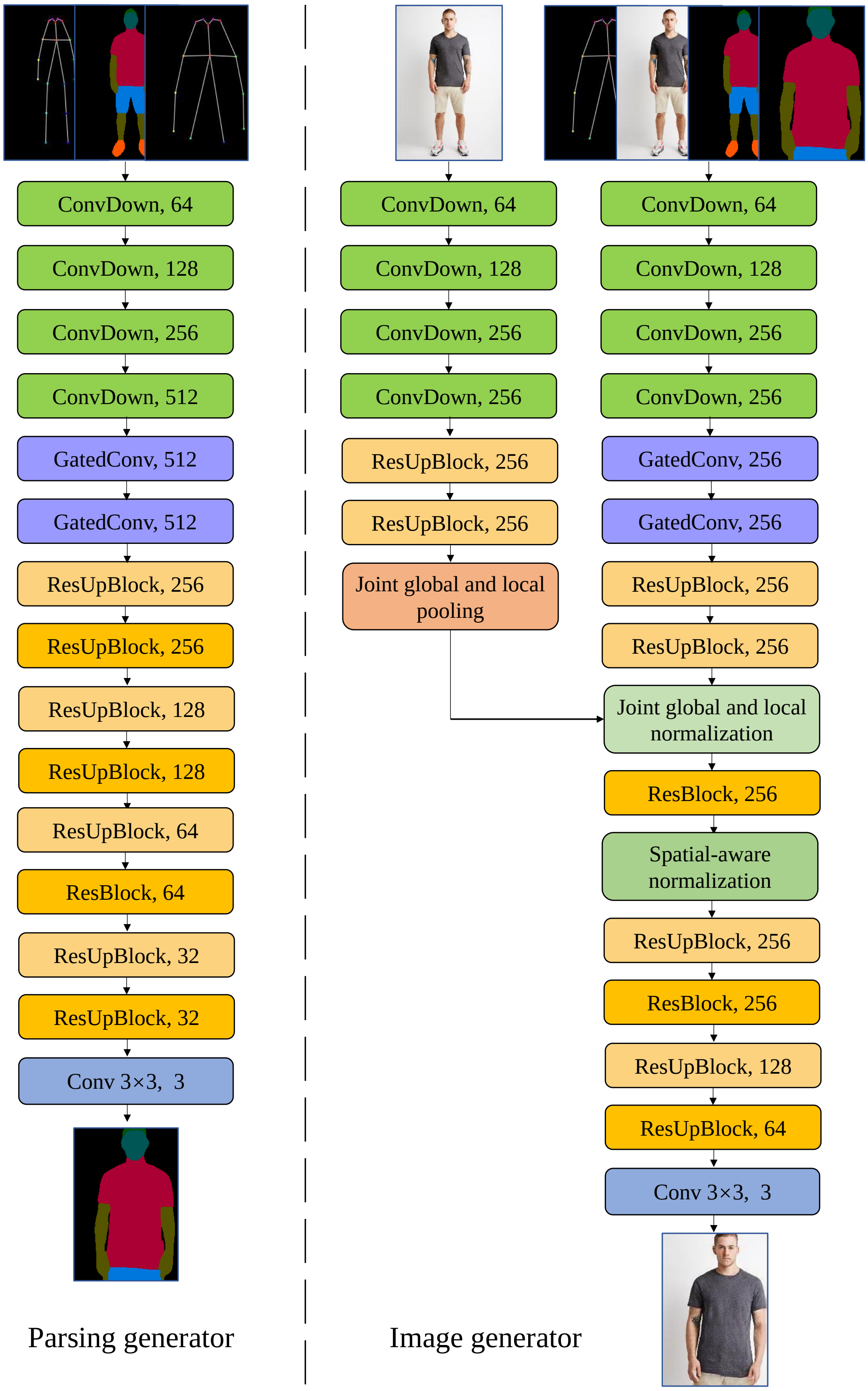

Figure 2: The detailed architecture of our model.

2. Comparison with State-of-the-art Methods.

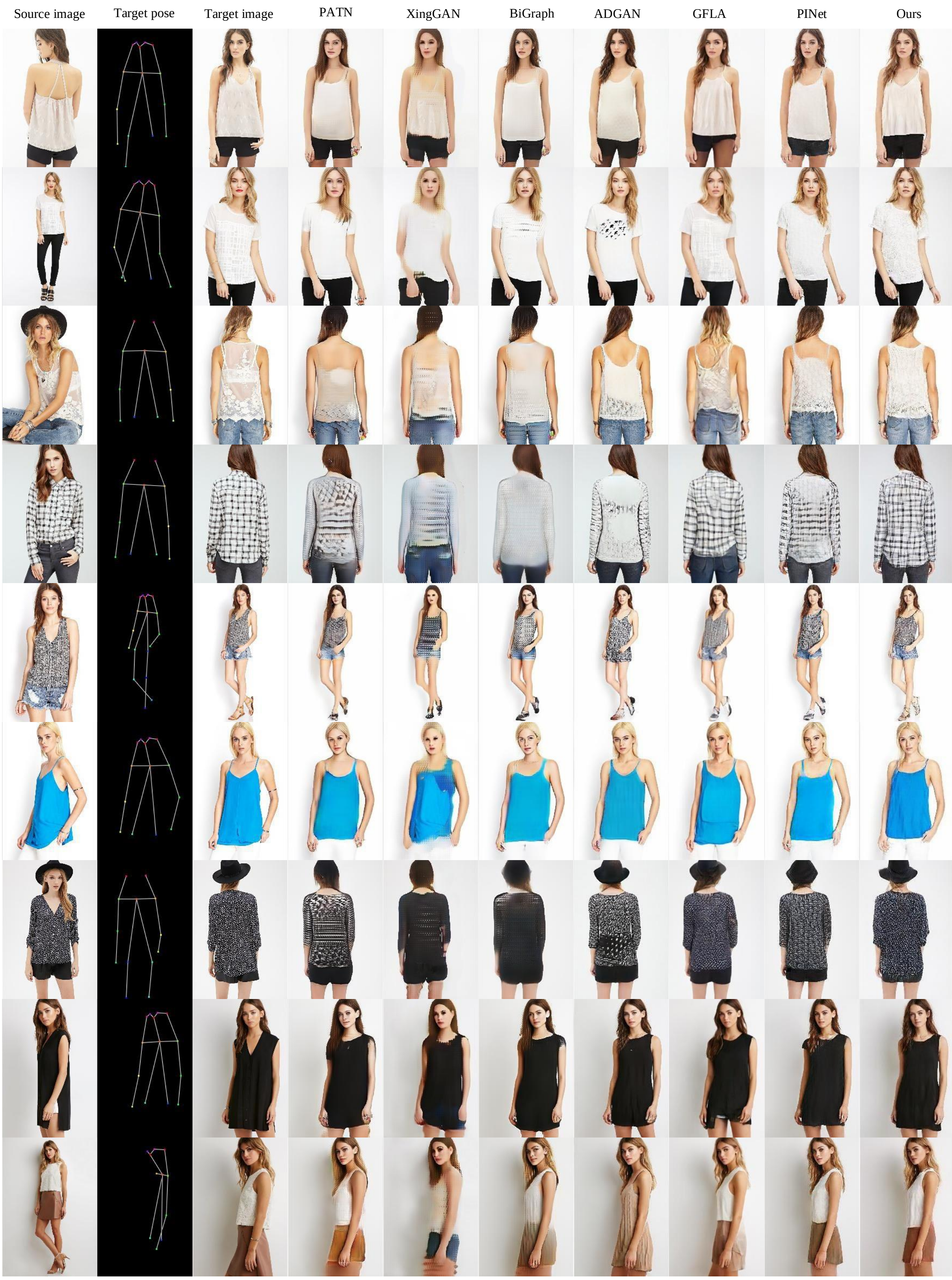

Figure 3: Comparison with PATN [7], XingGAN [5], BiGraph [4], ADGAN [2], GFLA [3] and PINet [6].

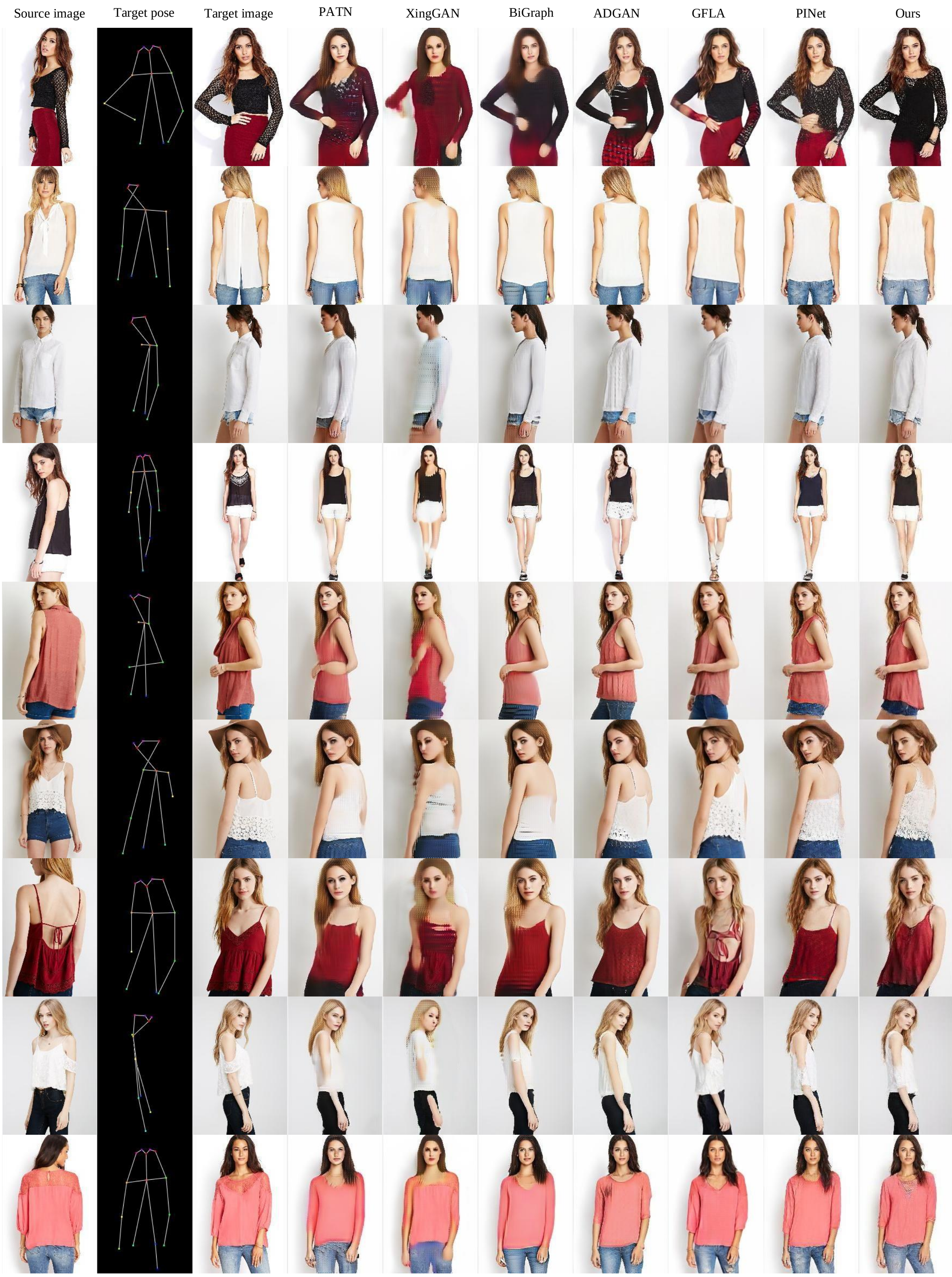

Figure 4: Comparison with PATN [7], XingGAN [5], BiGraph [4], ADGAN [2], GFLA [3] and PINet [6].

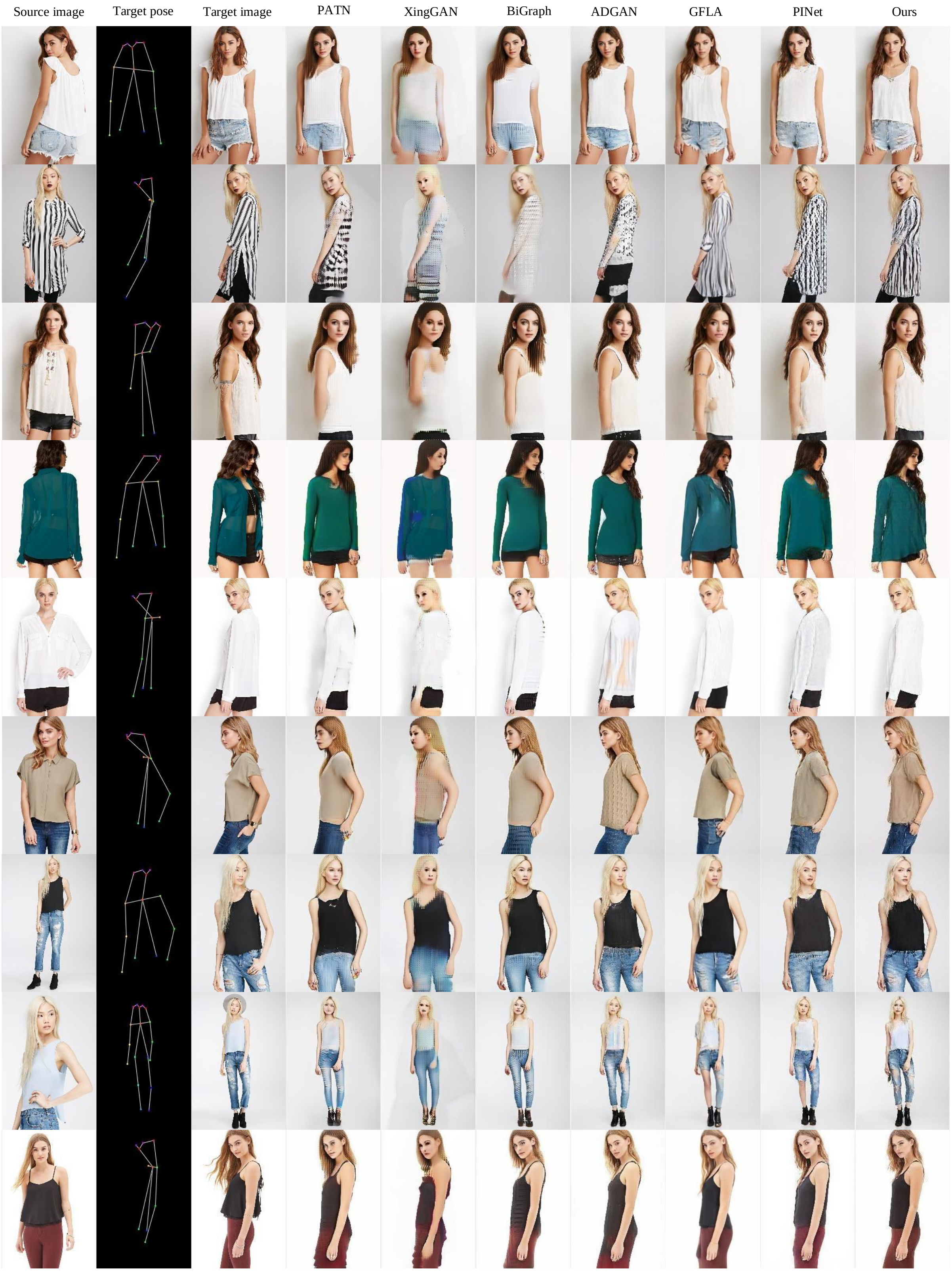

Figure 5: Comparison with PATN [7], XingGAN [5], BiGraph [4], ADGAN [2], GFLA [3] and PINet [6].

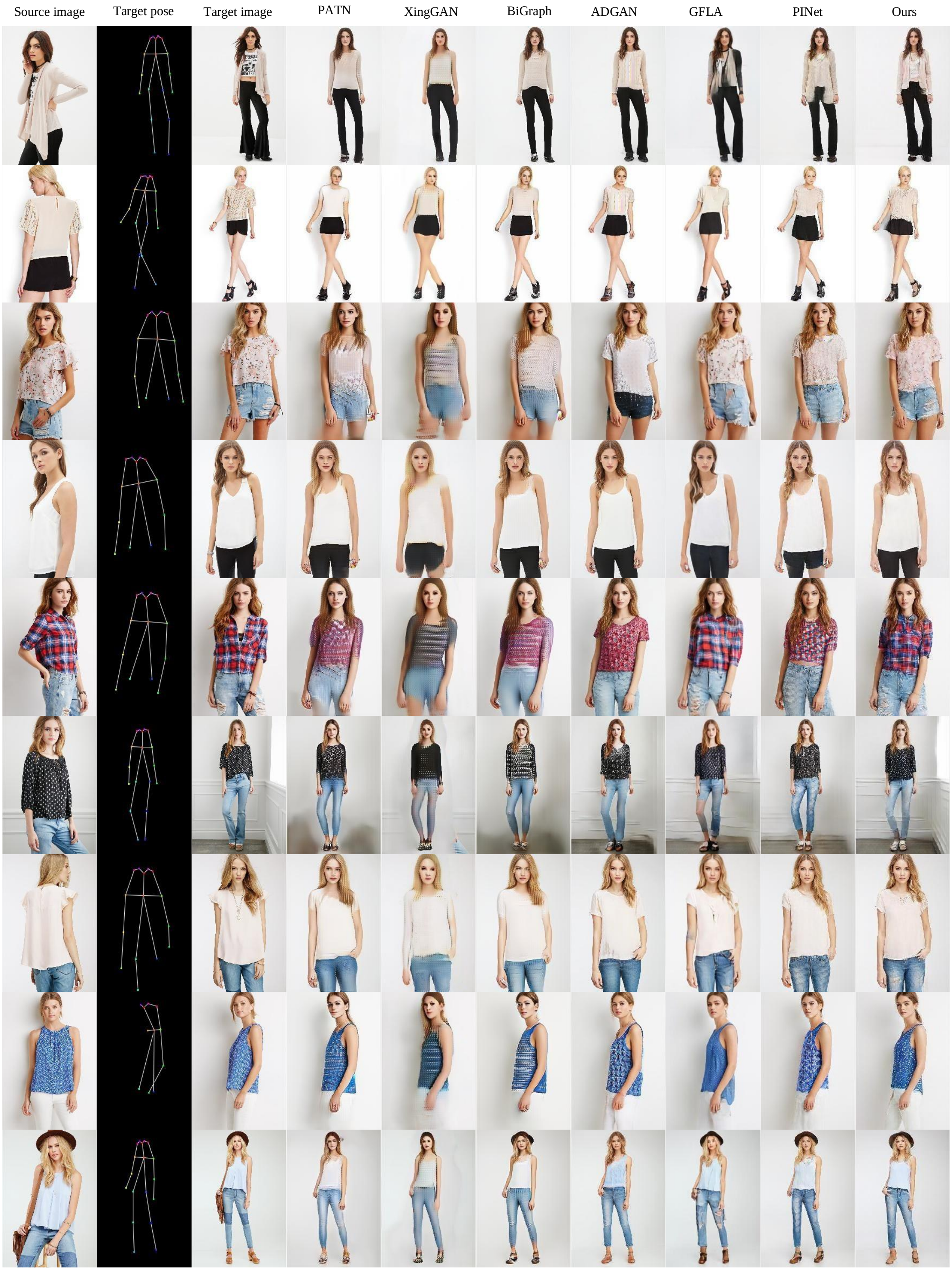

Figure 6: Comparison with PATN [7], XingGAN [5], BiGraph [4], ADGAN [2], GFLA [3] and PINet [6].

Figure 7: Comparison with PATN [7], XingGAN [5], BiGraph [4], ADGAN [2], GFLA [3] and PINet [6].

3. Texture Transfer Results.

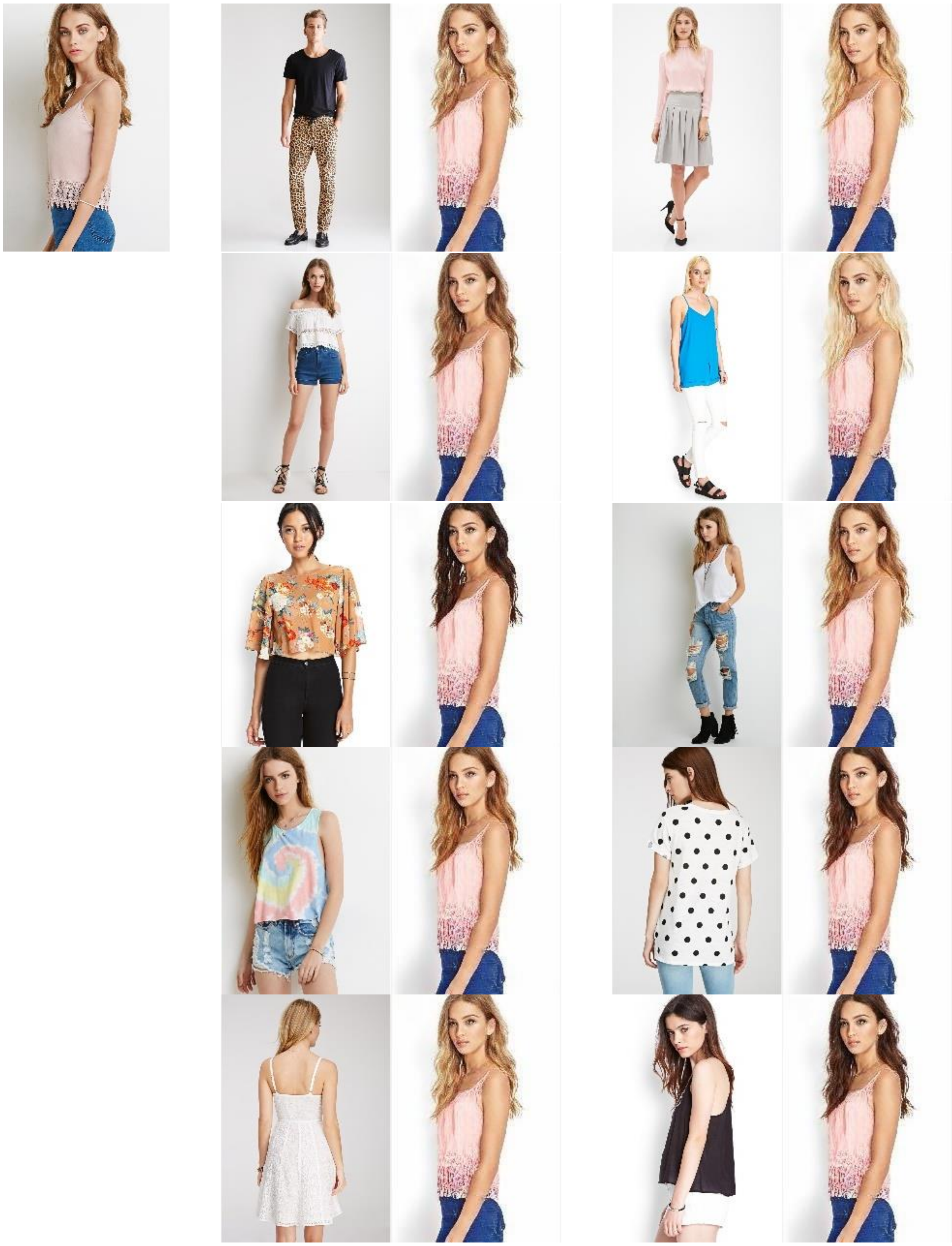

Figure 8: Texture transfer results of the hair.

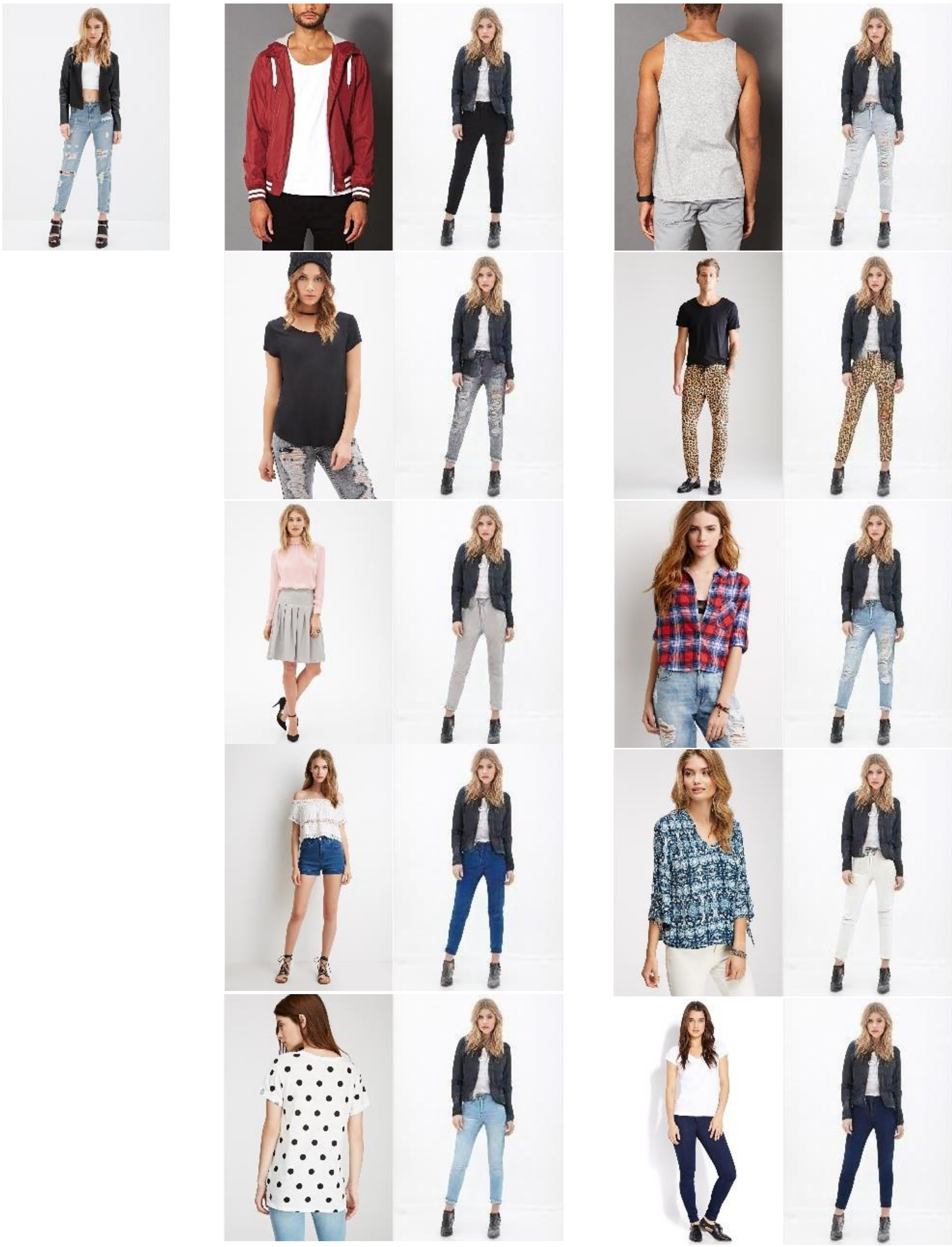

Figure 9: Texture transfer results of the pants.

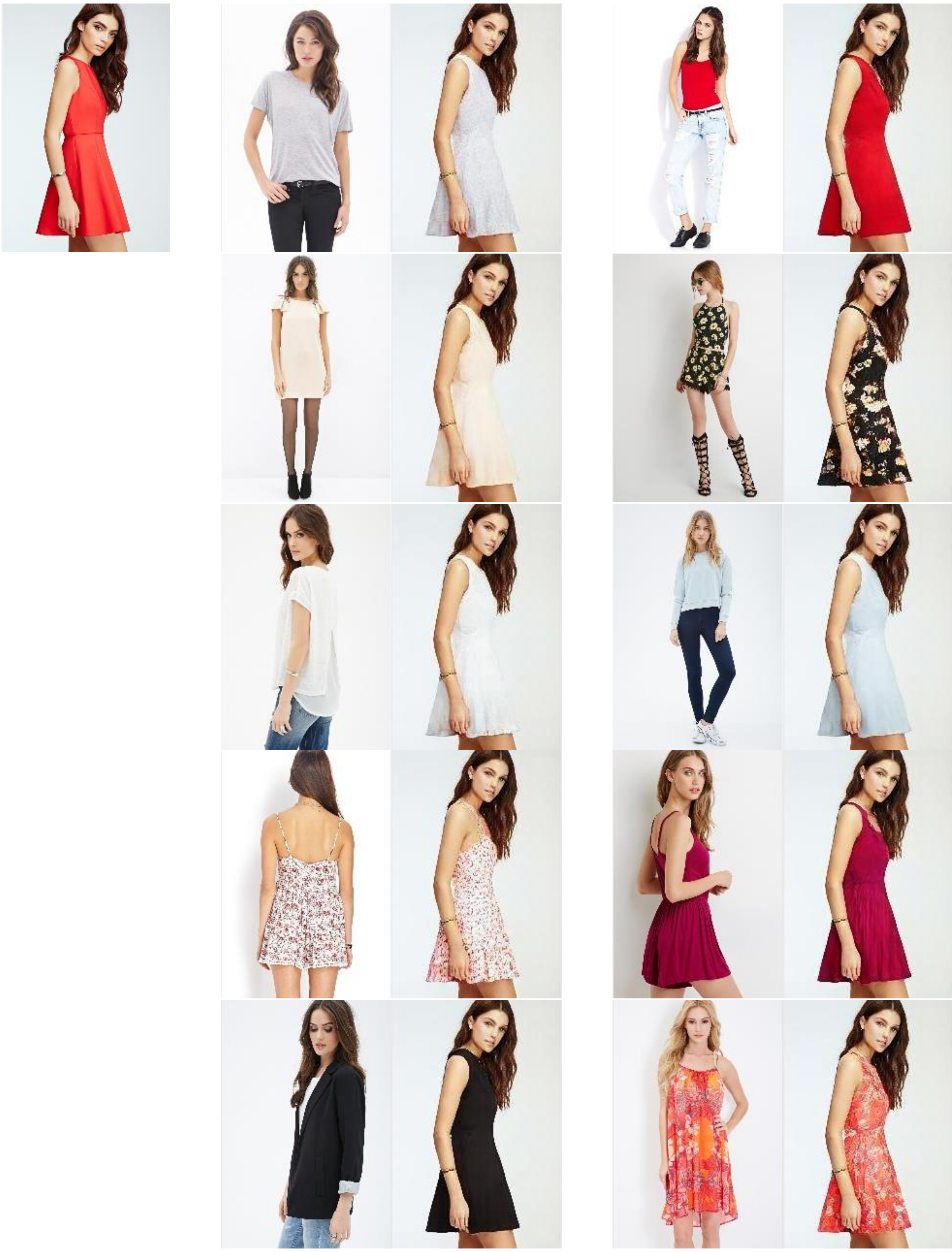

Figure 10: Texture transfer results of the dress.

4. Texture Interpolation Results.

Figure 11: Texture interpolation results of the upper clothes.

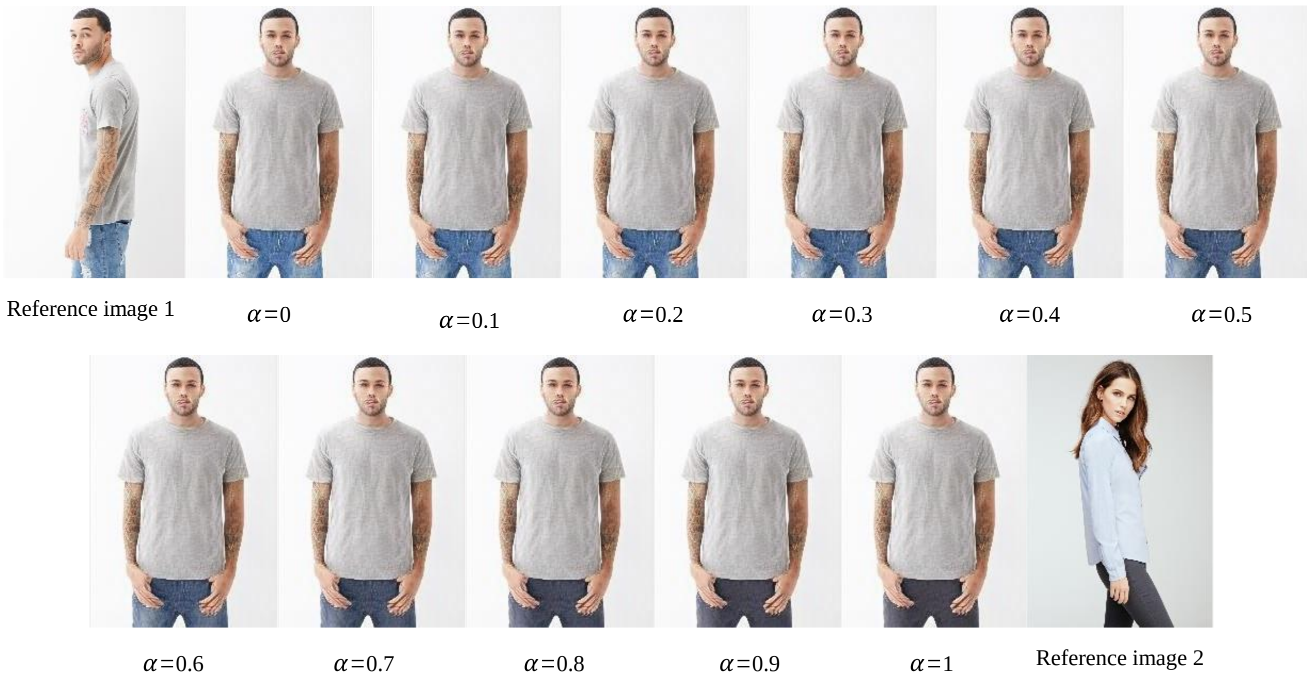

Figure 12: Texture interpolation results of the pants.

Figure 13: Texture interpolation results of the hair.

Figure 14: Texture interpolation results of the hair, the upper clothes and the pants.

5. Region Editing Results.

Figure 15: Region editing results.

6. Failure Cases and Analysis.

Although our method generates natural images with detailed textures in most cases, it cannot cope well with the styles and textures that deviate extremely from the training set distribution. The neural network predicts the outputs by the interpolation operator in the manifold built on the training data. Therefore, it is difficult to predict reasonable results for some challenging patterns, especially with large pose changes. Figure 16 shows some failure cases of our method. For specific pattern of clothing, our model fails to predict the correct pattern, and cannot preserve spatial context relationships. Besides, small parsing errors will not affect the performance of our method much as demonstrated by various results in the paper with imprecise parsing results.

Source image | Target image | Generated image | Source image | Target image | Generated image

Figure 16: Failure cases of our method.

6. Results on Images in the Wild.

We test our method on images in the wild. Figure 17 shows two cases of our method. Our model generates promising results with good texture consistency.

Source image | Ours | Target pose | Source image | Ours | Target pose

Figure 17: Failure cases of our method.

References

[1] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Int. Conf. Learn. Represent.*, 2015. 1

[2] Yifang Men, Yiming Mao, Yuning Jiang, Wei-Ying Ma, and Zhouhui Lian. Controllable person image synthesis with attribute-decomposed gan. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 5083–5092, 2020. 3, 4, 5, 6, 7

[3] Yurui Ren, Ge Li, Shan Liu, and Thomas H Li. Deep spatial transformation for pose-guided person image generation and animation. *IEEE Trans. Image Process.*, 29:8622–8635, 2020. 3, 4, 5, 6, 7

[4] Hao Tang, Song Bai, Philip HS Torr, and Nicu Sebe. Bipartite graph reasoning gans for person image generation. In *Brit. Mach. Vis. Conf.*, 2020. 3, 4, 5, 6, 7

[5] Hao Tang, Song Bai, Li Zhang, Philip HS Torr, and Nicu Sebe. Xinggan for person image generation. In *Eur. Conf. Comput. Vis.*, 2020. 3, 4, 5, 6, 7

[6] XingziandLi Kun Zhang, JinsongandLiu. Human Pose Transfer by Adaptive Hierarchical Deformation. 3, 4, 5, 6, 7

[7] Zhen Zhu, Tengteng Huang, Baoguang Shi, Miao Yu, Bofei Wang, and Xiang Bai. Progressive pose attention transfer for person image generation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2342–2351, 2019. 3, 4, 5, 6, 7