Supplementary Material for Few-Shot Human Motion Transfer by Personalized Geometry and Texture Modeling

Zhichao Huang[1], Xintong Han[2], Jia Xu[2], Tong Zhang[1]
[1]The Hong Kong University of Science and Technology [2]Huya Inc
zhuangbx@connect.ust.hk, {hanxintong, xujia}@huya.com, tongzhang@ust.hk

In this supplementary material, we first describe the detailed network architecture of our geometry generator and texture generator. Meanwhile, the choices of hyper-parameters are provided. We present extra experimental results, including a user study and more visual comparisons. We also perform an extra experiment on large number of source images to demonstrate that our method can synthesize realistic human as a single person model. A supplementary video that contains visual results of our method and the compared baselines is available at https://youtu.be/ZJ15X-sdKSU.

1. Details of Experiments

1.1. Network Architecture

Notations for the architecture

- Conv(a, b, c, d): 2d-convolution operation with input channel a, output channel b, kernel size $c \times c$ and stride d;
- CRN(a, b, c, d): Conv(a, b, c, d)-relu-normalization sequence;
- ResBlk(a): residual block used in ResNet with input and output channel a;
- Upsample(a): bilinear interpolation to upsample the input feature to a times of its original spatial resolution.

The padding strategy for convolution is "SAME". We use instance normalization since our batch size is limited by the memory of GPU and batch normalization does not work well when the batch size is small. The O_i's and I_i's denote the position where we merge the feature of source images and target pose as shown in Figure 2(a) in our main paper. The output at O_i will be merged with attention and sent into the corresponding I_i.

Image Context Encoder E_I: CRN(3, 32, 7, 1) $\rightarrow$ CRN(32, 64, 3, 2) $\xrightarrow{O_1}$ CRN(64, 128, 3, 2) $\xrightarrow{O_2}$ CRN(128, 128, 3, 2) $\xrightarrow{O_3}$ CRN(128, 128, 3, 2) $\xrightarrow{O_4}$

Pose Attention Encoder E_W: CRN(26, 32, 7, 1) $\rightarrow$ CRN(32, 64, 3, 2) $\xrightarrow{O_1}$ CRN(64, 128, 3, 2) $\xrightarrow{O_2}$ CRN(128, 128, 3, 2) $\xrightarrow{O_3}$ CRN(128, 128, 3, 2) $\xrightarrow{O_4}$

Target Pose Encoder E_P: CRN(26, 32, 7, 1) $\rightarrow$ CRN(32, 64, 3, 2) $\xrightarrow{O_1}$ CRN(64, 128, 3, 2) $\xrightarrow{O_2}$ CRN(128, 128, 3, 2) $\xrightarrow{O_3}$ CRN(128, 128, 3, 2) $\xrightarrow{O_4}$ 10$\times$Resblk(128)

Mask Branch of Decoder D_G: $\xrightarrow{I_4}$ Upsample(2) $\rightarrow$ CRN(256, 128, 5, 1) $\xrightarrow{I_3}$ Upsample(2) $\rightarrow$ CRN(256, 128, 5, 1) $\xrightarrow{I_2}$ Upsample(2) $\rightarrow$ CRN(256, 64, 5, 1) $\xrightarrow{I_1}$ Upsample(2) $\rightarrow$ CRN(128, 32, 5, 1) $\rightarrow$ Conv(32, 25, 7, 1)

Coordinate Branch of Decoder D_G: $\xrightarrow{I_4}$ Upsample(2) $\rightarrow$ CRN(256, 128, 5, 1) $\xrightarrow{I_3}$ Upsample(2) $\rightarrow$ CRN(256, 128, 5, 1) $\xrightarrow{I_2}$ Upsample(2) $\rightarrow$ CRN(256, 64, 5, 1) $\xrightarrow{I_1}$ Upsample(2) $\rightarrow$ CRN(128, 32, 5, 1) $\rightarrow$ Conv(32, 48, 7, 1)

Texture Encoder E_T: CRN(3, 64, 7, 1) $\rightarrow$ CRN(64, 128, 3, 2) $\rightarrow$ CRN(128, 256, 3, 2) $\rightarrow$ CRN(256, 512, 3, 2)

Texture Encoder D_T: 6$\times$Resblk(512) $\rightarrow$ Upsample(2) $\rightarrow$ CRN(512, 256, 3, 1) $\rightarrow$ Upsample(2) $\rightarrow$ CRN(256, 128, 3, 1) $\rightarrow$ Upsample(2) $\rightarrow$ CRN(128, 64, 3, 1) $\rightarrow$ Conv(64, 3, 7, 1)

1.2. Hyperparamters

For λ's mentioned in the Equation (19), we empirically set $\lambda_I = 1, \lambda_M = 1, \lambda_{RT} = 1, \lambda_{RC} = 20, \lambda_{RM} = 0.8$. The λ's mentioned in the Equation (21) are the same.

We use Adam optimizer with $(\beta_1, \beta_2) = (0.5, 0.999)$ at all initialization, multi-video training and fine-tuning stages. For initialization and multi-video training stages, learning rate starts at 0.0002. The initialization stage lasts for 10 epochs and the learning rate decays half at the 5th epoch. We train the multi-video stage for 15 epochs with learning decaying half at the 5th and the 10th epoch. At the test time, we randomly select 20 images from one video as the source images. The number of fine-tuning steps is 40 for geometry generator and 300 for the texture embedding and the background. We set the learning rate to be 0.0002 for the geometry generator and 0.005 for the texture embedding and the background. We train the generators on two GPUs

with their memories fully utilized. We set batch size to be 16 for initialization of geometry generator, 8 for initialization of texture generator and 10 for the multi-video training stage. We fine-tune the model on one GPU and the batch size is 6.

2. More Experiments

We include more visual results in the supplementary video.

2.1. Sufficient Number of Source Images

As the number of source images grows, our method is able to generate realistic human motion as single person model. We fine-tune our model on all available images of the source person. The number of frames for each source person is about 4,000 and slightly varies across different source persons. We obtain the best **FReID 2.33** and **pose error 6.34**. The visual results are shown in Figure 4 and Figure 5. We observe that high quality images are effectively generated with fine texture details and accurate target pose. For example, realistic textures on the T-shirt and shoes are well preserved in the third example of Figure 5. Interestingly, in the second example of Figure 5, shadows can be accurately generated as the fine-tuning process enables the geometry generator to model the shadow as one part of the human.

Comparison with EDN [2]. We compare our method with EQN when the number of source images is sufficient. We remove the face generator from EDN and train individual model for each person using all images of the source person. In the task of motion transfer, we achieve **FReID 5.21** and **pose error 9.45**, which is inferior to our method. Figure 1 shows more visual comparisons between our method and EDN. Our proposed method achieves higher synthesis quality than EDN.

2.2. Generating HD Images

Our method is able to generate HD images using generators trained on low resolution images. The resolution of the neural rendering mainly relies on the resolution of the texture map. As the high resolution UV map is almost continuous, we can simply up-sample the UV map with bilinear interpolation. And the high resolution texture can be obtained by directly fine-tuning the texture map on HD images. Figure 2 shows some examples of the generating HD images on 512×512 resolution with few or sufficient number of source images.

2.3. More Qualitative Comparison

Figure 3 includes more synthesized images of human motion transfer of different methods in the few-shot setting. We give more detailed analysis of each competing method as follows.

Posewarp [1]. Posewarp segments each body part by the predefined masks and uses affine transformation to transfer body parts from the source pose to the target pose. However, affine transformation only produces coarse body parts in the target pose. The generative network at later stage has limited power and cannot recover all the details of the source person. Besides, the predefined mask cannot generalize to different persons with different shapes. Therefore, Posewarp may lose some parts of the body (*e.g.* head regions in Figure 3). The resulting artifacts are consistent with the what [6, 4] report in their experiments.

MonkeyNet [5]. MonkeyNet does not use 2D pose keypoints extracted by off-the-shelf models. Instead, it uses the keypoints detected by itself and moves these keypoints to generate motion for the source person. However, these keypoints are necessarily aligned with the pose and moving these keypoints does not change the pose of the source person. It only adds strange distortion to the source image while makes the pose stay the same. Similar observations are also made in the experimental comparison in [6].

LWG [4]. While LWG can produce realistic images for persons with regular shapes (the 5th-8th rows of Figure 3), it is not able to generate persons with complicated shapes such as dress and long hair (the 1st-4th rows of Figure 3). This is because LWG is built on top of SMPL models predicted by HMR [3], and SMPL is a skinned 3D human model that does not model human's clothes and hair. Furthermore, HMR might be inaccurate when estimating the target pose especially when the background is cluttered or there is motion blur, making the generated results temporally discontinuous (see supplementary video for temporal comparison).

FewShotV2V [6]. FewShotV2V only outlines the human in the target pose with plenty of artifacts. Although FewShotV2V uses SPADE module to generate different weights for different person, the module requires a lot of training subjects (1,500 videos as described in the paper) so that it can learn how to generate weights for different persons. And the 50 videos we use are not enough for FewShotV2V to learn good generalization. Our method, in contrast, decouples the generation of geometry and texture and it can generalize to a new person with only a small number of training videos.

Compared with these methods, our method obtains the most visual appealing results without presenting the aforementioned artifacts.

2.4. Texture Map

We show more examples of texture map in Figure 6 when different texture fine-tuning strategies are employed. Compare with the texture generator, direct merging source textures leaves a large proportion of the texture unfilled (Black part of Figure 6(a)). Directly fine-tuning the texture map

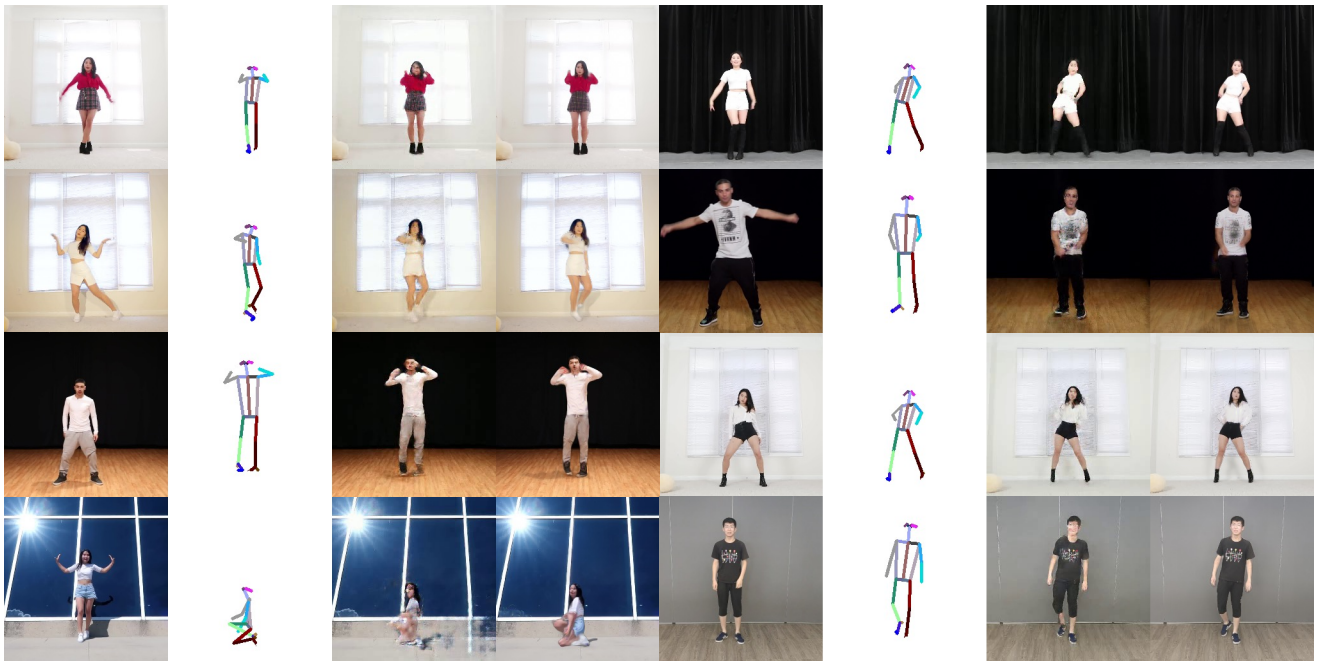
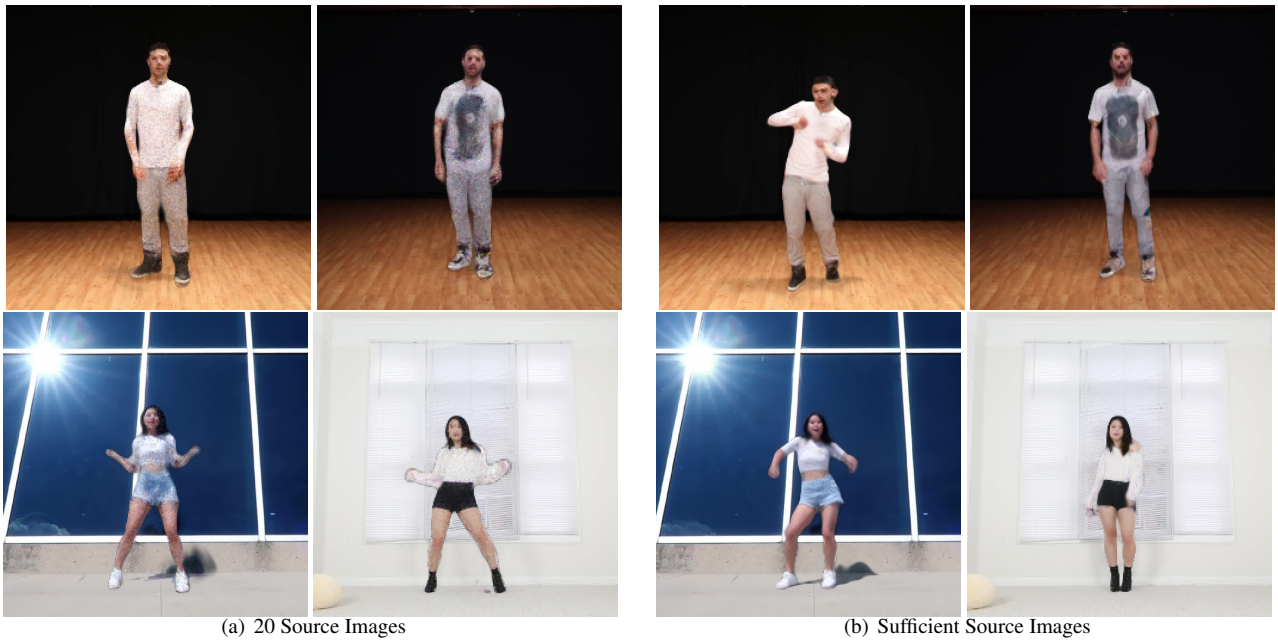


Figure 1. Visual comparison between our method and EDN [2] with sufficient number of source images. Our method generates more realistic human.



(a) 20 Source Images (b) Sufficient Source Images

Figure 2. Examples of 512×512 HD images by up-sampling the UV maps and fine-tune the texture on high resolution source images.

adds noises to the texture. If it is rendered to the geometry, the synthesized human will be noisy and unnatural. Fine-tuning the embedding gives the most realistic texture with fewer artifacts and less noise. Comparison of generated videos with different texture maps can be found in the supplementary video file.

References

[1] Guha Balakrishnan, Amy Zhao, Adrian V Dalca, Fredo Durand, and John Guttag. Synthesizing images of humans in

unseen poses. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8340–8348, 2018. 2

[2] Caroline Chan, Shiry Ginosar, Tinghui Zhou, and Alexei A Efros. Everybody dance now. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5933–5942, 2019. 2, 3

[3] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7122–7131, 2018. 2

[4] Wen Liu, Zhixin Piao, Jie Min, Wenhan Luo, Lin Ma, and Shenghua Gao. Liquid warping gan: A unified framework for human motion imitation, appearance transfer and novel view synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5904–5913, 2019. 2

[5] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. Animating arbitrary objects via deep motion transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2377–2386, 2019. 2

[6] Ting-Chun Wang, Ming-Yu Liu, Andrew Tao, Guilin Liu, Bryan Catanzaro, and Jan Kautz. Few-shot video-to-video synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, pages 5013–5024, 2019. 2

Figure 3. More visual comparison between our method and other competing methods. Supplementary video contains the corresponding video results.

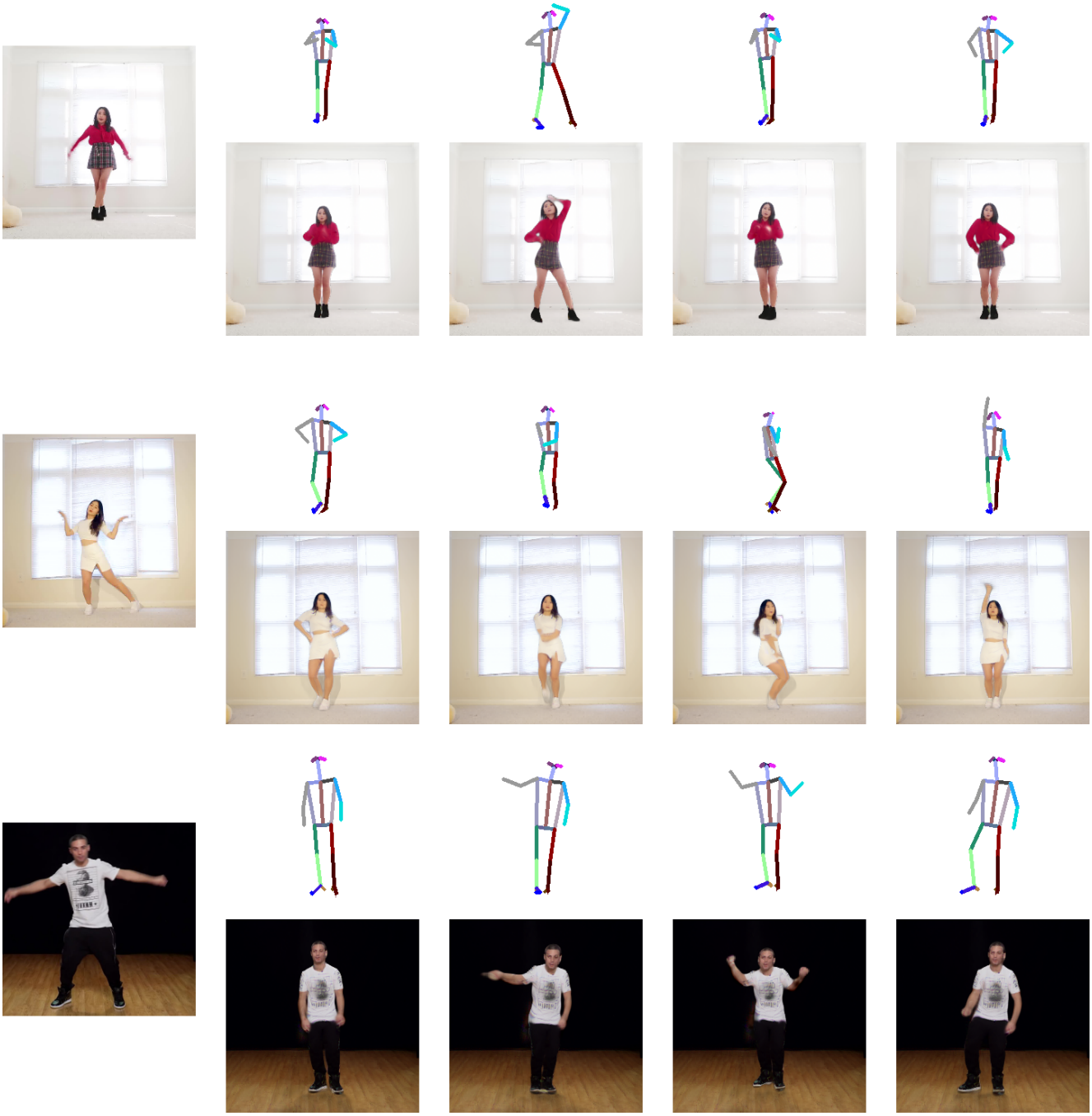


Figure 4. The synthesized motion of our method with sufficient number of source images. The first column represents source person and the other columns are the synthesized human in the corresponding pose. Video results are included in the supplementary video.

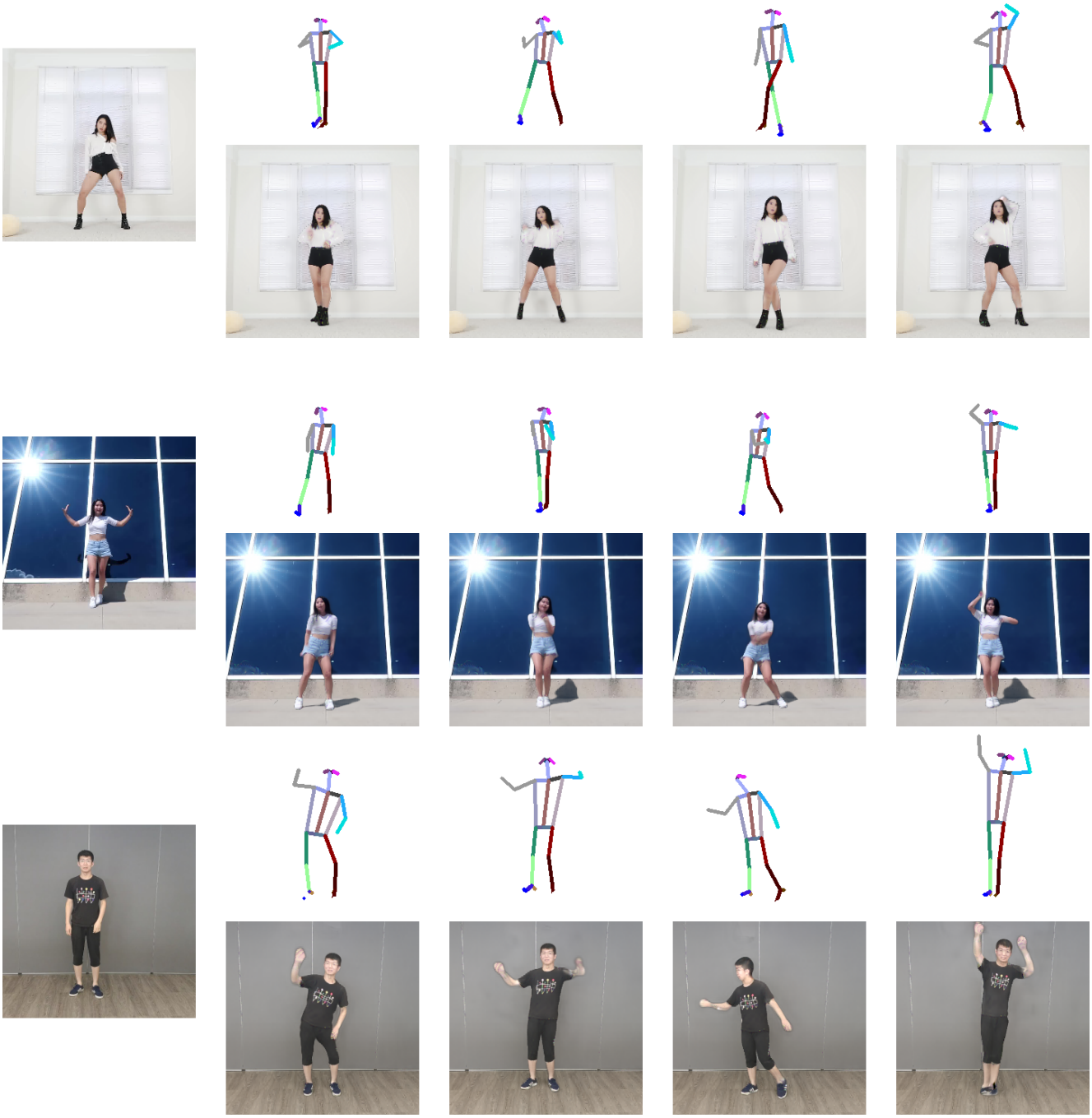


Figure 5. The synthesized motion of our method with sufficient number of source images. The first column represents source person and the other columns are the synthesized human in the corresponding pose. Video results are included in the supplementary video.

(a) Direct Merge

(b) No Fine-tune

(c) Fine-tune the Texture Map

(d) Fine-tune the Embedding

Figure 6. Texture maps of different texture map fine-tuning strategies. Corresponding video results can be found in the supplementary video.