

IM-Animation: An Implicit Motion Representation for Identity-decoupled Character Animation

Supplementary Material

1. Implementation Details

In the following section, we outline the training strategies for IM-Animation, designed to achieve efficient convergence with limited resources. Unlike existing implicit video-driven methods, our approach consists of three stages. In Stage 1, we train the motion encoder, fine-tuning the pretrained encoder and quantizer from TiTok [10] using self-reconstructed joint maps for supervision. Stage 2 involves joint training of the motion encoder and the retargeting module, incorporating data augmentation techniques and constructing action-consistent but identity-inconsistent paired datasets. Finally, in Stage 3, we inject motion and expression control signals into the DiT model for end-to-end training, while maintaining supervision from the joint decoder. This structured approach aims to enhance the model’s performance in complex scenarios and ensure effective learning of motion representations.

Stage 1: Motion encoder training stage. As mentioned in the main text, our subsequent experiments are based on the pretrained checkpoints provided by TiTok.

TiTok’s original design is based on image reconstruction and generation tasks, and the features extracted by its encoder tend to capture more detailed image semantics than desired. During the motion representation phase, we aim to minimize identity detail leakage to the downstream DiT model. To achieve this, we have meticulously designed a decoder that transforms the compressed tokens, of which we mentioned there are 32 in the main text, into a joint map. In this process, we restructure the motion decoder. We align the number of mask tokens with the quantity of joint supervision. Specifically, we select the 20 body joints from DW-Pose [9] along with all hand joints as supervision points. After obtaining the corresponding joint tokens, we employ a series of convolutional layers to upsample the mask tokens to the scale of the ground truth joint map.

$$Loss_{motion} = \frac{1}{T} \sum_{t=1}^T (H_{motion,t} - H_{gt,t})^2 \quad (1)$$

Here, T represents the total number of frames in the video sequence, $H_{motion,t}$ is the heatmap generated by the model for the t -th frame, and $H_{gt,t}$ is the corresponding ground truth heatmap. At this stage, we do not design data augmentation or similar self-supervised retargeting training constructs like X-UniMotion [5]. In other words, this stage is more akin to training a keypoint prediction process.

Stage 2: Retargeting training stage. In the imple-

mentation of the retargeting module, we first compress the source image into latent space using the VAE from Wan2.2 [7] before performing patchification. In our design, the number of mask tokens is aligned with the number of patches from the source image. This approach allows us to overcome the limitations of the patch grid while facilitating channel concatenation during the subsequent process of controlling condition injection.

In this phase, unlike the first stage where we directly regress the heatmap, we employ random data augmentation to enable the model to learn retargeting across different IDs and at larger scales. After applying random color transformations, we randomly crop or scale the original video, and then supervise the model using the original video. The loss function of this stage is defined as follow,

$$L_{retarget} = \frac{1}{T} \sum_{t=1}^T (H_{retarget,t} - H_{gt,t})^2 \quad (2)$$

where T denotes the total number of time steps, $H_{retarget,t}$ is the generated retargeting heatmap, and $H_{gt,t}$ is the ground truth heatmap at time step t . By averaging the squared differences between the predicted heatmap and the corresponding ground truth heatmap at each time step, $L_{retarget}$ aims to minimize the disparity between the generated and true heatmaps, thereby improving the model’s performance in the retargeting task.

Stage 3: End-to-End training stage. In the final stage, we conduct end-to-end training. In this part, we use additional synthetic data to enhance the data scale. Our method for injecting control signals has been thoroughly described in the main text. To avoid disrupting the pre-trained capabilities of DiT, we reuse the weights from the original checkpoint for the embedding part corresponding to the channel-wise concatenation, while initializing the region part to zero. The original Wan2.2 5B model contains 30 DiT blocks. We insert an expression block after every 6 DiT blocks, employing a skip connection technique to prevent disrupting the original capabilities of the DiT. At this stage, we also employ intermediate supervision for training. For the diffusion loss function,

$$L_{DiT} = \mathbb{E}_{t, x_0^{1:N}, y, \epsilon} \left\| \epsilon - \epsilon_{\theta}(\sqrt{\alpha_t}x_0^{1:N} + \sqrt{1 - \alpha_t}\epsilon, t, y) \right\|^2, \quad (3)$$

where t is sampled from the range $[1, T]$ (denoting the denoising steps), ϵ represents the random noise, y denotes the text prompts, and $x_0^{1:N}$ refers to the video data comprising N frames.



Figure 1. Samples of synthesized data . The first row is the reference video and the second row is the target video.



Figure 2. Samples of UE data . The first row is the reference video and the second row is the target video.



(a) GT Joint Map



(b) Predicted Heatmap

Figure 3. Visualization of Joint Heatmap.

We also employ the aforementioned intermediate supervision to accelerate convergence. The loss function for the entire Stage Three is defined as

$$L_{total} = L_{DiT} + \alpha \cdot L_{retarget} \quad (4)$$

During our training process, α is set to 10.

2. Dataset Details

We utilize Kling [4] to synthesize data that maintains consistent motion across diverse identities. Throughout the synthesis process, we engineered data featuring a range of spatial characteristics and body types, followed by a meticulous manual selection of the outputs. Sample outputs of the generated data are presented in Figure 1.

For the UE data, similar to the approach taken in Re-CameraMaster [1], we synthesized data from various camera positions and an array of scenes. This data synthesis strategy enhances our model’s ability to adapt to different viewpoints and environmental variations, thereby improving both the quality and diversity of the generated character animations.

Table 1. Motion Encoder Comparison .

Method	PSNR* \uparrow	SSIM \uparrow	LPIPS \downarrow	FVD \downarrow	FID \downarrow
Wan + VAE motion encoder	17.14	0.76	0.37	94.44	653.30
full model	22.87	0.91	0.24	51.19	270.42

Table 2. Retargeting Module Comparison .

Method	PSNR* \uparrow	SSIM \uparrow	LPIPS \downarrow	FVD \downarrow	FID \downarrow
Wan + SA retargeting module	19.89	0.82	0.26	68.99	477.63
full model	22.87	0.91	0.24	51.19	270.42

3. Visualization of Joint Map

To validate the effectiveness of our intermediate supervision, we present visual results of the joint decoder outputs following the redirection process, as illustrated in Figure 3. The heatmaps generated by our decoder demonstrate a strong correspondence with the ground truth, indicating a high level of accuracy in the predictions. This alignment not only underscores the robustness of our model, but also highlights the efficacy of the intermediate supervision strategy in enhancing the learning process.

The visual comparison reveals that the decoder successfully captures the intricate details of the target output, suggesting that our approach effectively mitigates discrepancies that typically arise during the generation process. Such results provide compelling evidence that our method contributes to improved performance in generating high-fidelity character animations.

4. More Ablation Study

For the motion encoder, we conduct more fine-grained ablation experiments. We compare our motion encoder with the results of directly inputting latent variables compressed by VAE into the downstream model for retargeting. This comparison allows us to more clearly assess the advantages of the motion encoder in terms of generation quality and performance.

As shown in the Figure 4 and Table 1, directly using VAE for encoding results in background information being directly encoded into the motion tokens, leading to a decline in generation quality. In contrast, our method effectively prevents the leakage of semantic information from the motion tokens.

In addition, we conduct a fine-grained validation of the effectiveness of the mask token-based retargeting module. In fact, we experiment with the retargeting implementation proposed by X-UniMotion, which utilizes self-attention to combine two sources of tokens. Although this implementation is not open-sourced, we develop a similar version and conduct comparative experiments. As shown in the table, we find that in some cases, this type of retargeting leads to an increased probability of implicit representation control failure, resulting in generated videos that tend to maintain



Figure 4. Ablation Study.

static motions which can be found in Figure 4.

5. More Visualization results.

Here, as shown in the Figure 5, we provide some generated results of IM-Animation on the TikTok dataset.

Additionally, as shown in Figure 6, we present more qualitative results here compared to Animate-X [6], MimicMotion [11], AnimateAnyone [3], UniAnimate-DiT [8], Champ [12] and Wan-Animate [2], highlighting that our model better preserves certain details compared to those in the main text. We also demonstrate that our model performs well in cases where explicit driven retargeting fails. In Figure 7, we also present the generation quality on the synthetic UE dataset. In the Figure 8, we also present additional visualization results.

References

- [1] Jianhong Bai, Menghan Xia, Xiao Fu, Xintao Wang, Lianrui Mu, Jinwen Cao, Zuoqiu Liu, Haoji Hu, Xiang Bai, Pengfei Wan, et al. Recammaster: Camera-controlled generative rendering from a single video. *arXiv preprint arXiv:2503.11647*, 2025. 2
- [2] Gang Cheng, Xin Gao, Li Hu, Siqi Hu, Mingyang Huang, Chaonan Ji, Ju Li, Dechao Meng, Jinwei Qi, Penchong Qiao, et al. Wan-animate: Unified character animation and replacement with holistic replication. *arXiv preprint arXiv:2509.14055*, 2025. 3
- [3] Li Hu. Animate anyone: Consistent and controllable image-to-video synthesis for character animation. In *Proceedings of*

the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 8153–8163, 2024. [3](#)

- [4] Kwai. Keling, 2025. [2](#)
- [5] Guoxian Song, Hongyi Xu, Xiaochen Zhao, You Xie, Tianpei Gu, Zenan Li, Chenxu Zhang, and Linjie Luo. X-unimotion: Animating human images with expressive, unified and identity-agnostic motion latents. *arXiv preprint arXiv:2508.09383*, 2025. [1](#)
- [6] Shuai Tan, Biao Gong, Xiang Wang, Shiwei Zhang, Dandan Zheng, Ruobing Zheng, Kecheng Zheng, Jingdong Chen, and Ming Yang. Animate-x: Universal character image animation with enhanced motion representation. *arXiv preprint arXiv:2410.10306*, 2024. [3](#)
- [7] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025. [1](#)
- [8] Xiang Wang, Shiwei Zhang, Longxiang Tang, Yingya Zhang, Changxin Gao, Yuehuan Wang, and Nong Sang. Unianimate-dit: Human image animation with large-scale video diffusion transformer. *arXiv preprint arXiv:2504.11289*, 2025. [3](#)
- [9] Zhendong Yang, Ailing Zeng, Chun Yuan, and Yu Li. Effective whole-body pose estimation with two-stages distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4210–4220, 2023. [1](#)
- [10] Qihang Yu, Mark Weber, Xueqing Deng, Xiaohui Shen, Daniel Cremers, and Liang-Chieh Chen. An image is worth 32 tokens for reconstruction and generation. *arxiv: 2406.07550*, 2024. [1](#)
- [11] Yang Zhang, Jiayi Gu, Li-Wen Wang, Han Wang, Junqi Cheng, Yuefeng Zhu, and Fangyuan Zou. Mimicmotion: High-quality human motion video generation with confidence-aware pose guidance. *arXiv preprint arXiv:2406.19680*, 2024. [3](#)
- [12] Shenhao Zhu, Junming Leo Chen, Zuozhuo Dai, Zilong Dong, Yinghui Xu, Xun Cao, Yao Yao, Hao Zhu, and Siyu Zhu. Champ: Controllable and consistent human image animation with 3d parametric guidance. In *European Conference on Computer Vision*, pages 145–162. Springer, 2024. [3](#)



Figure 5. More visualization results on TikTok dataset.

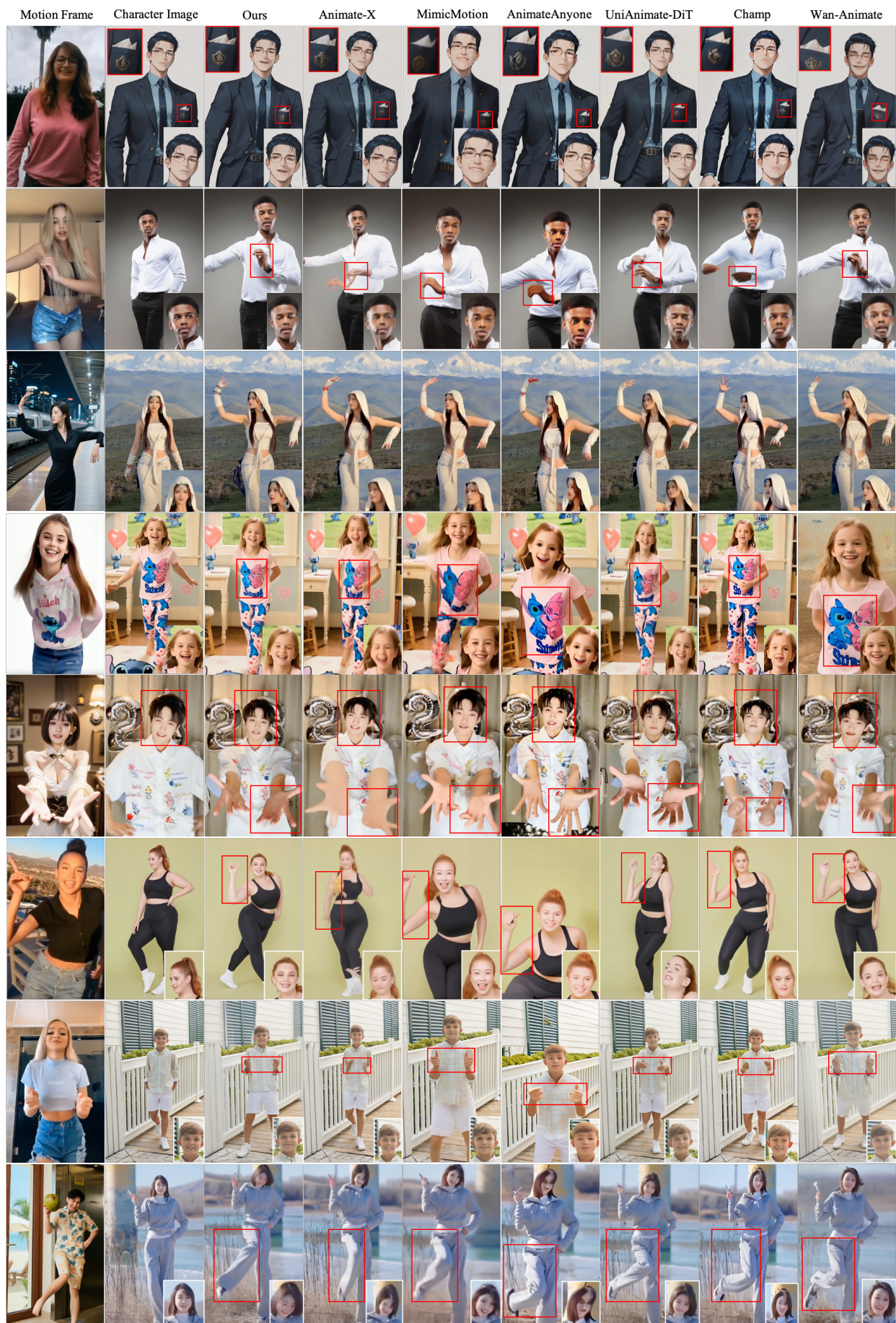


Figure 6. More visualization results of Comparson Results.

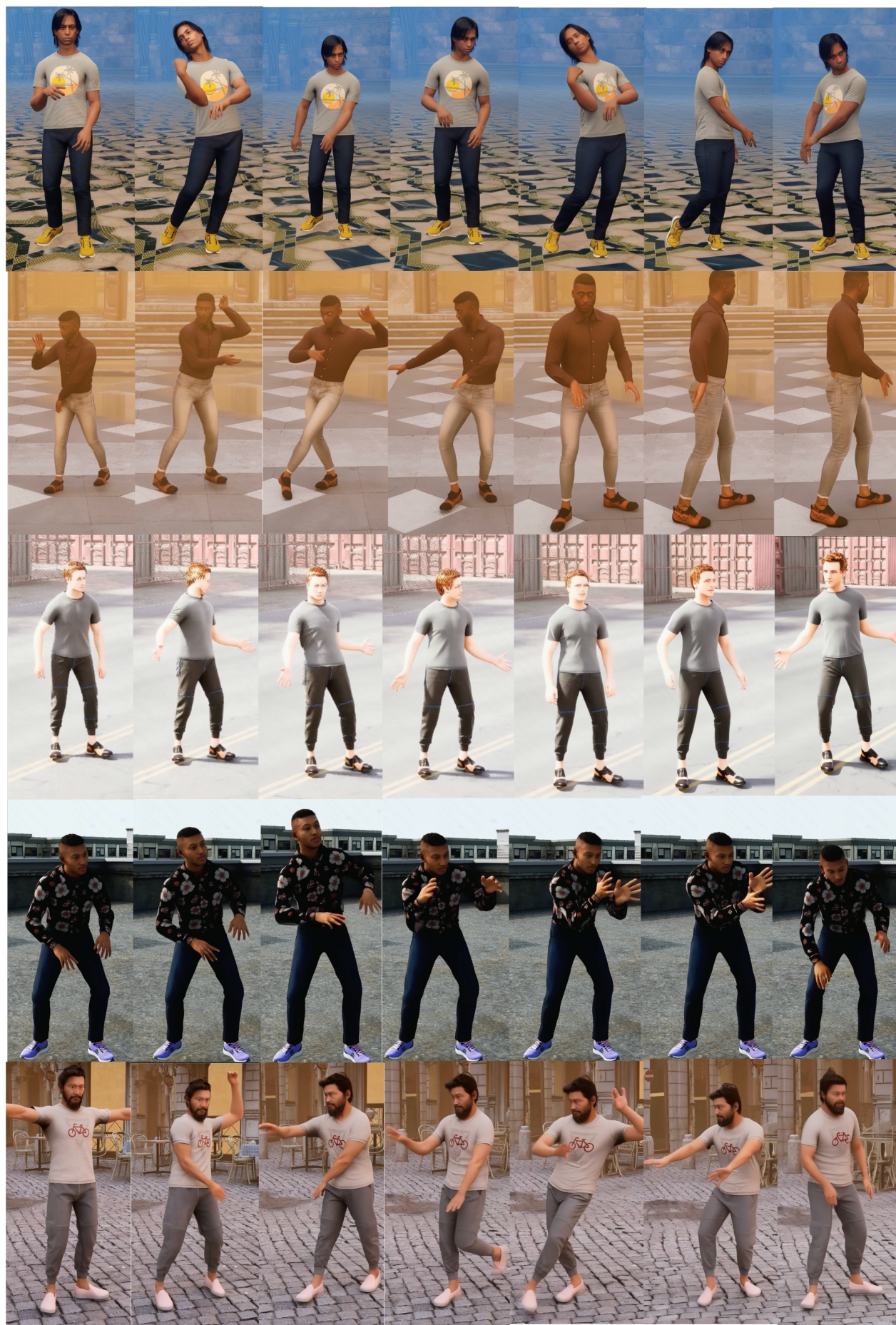


Figure 7. More visualization results on UE dataset.



Figure 8. More visualization results on in the wild data