Supplementary Materials
ClothFormer: Taming Video Virtual Try-on in All Module

A. Implementation Details

A.1. Model Architectures

GMM(TPS-based warping module). To infer an anti-occlusion target clothes $\hat{C}_t$, we predict a TPS-based warped result by GMM, which is composed of two feature extractors and a regression network in GMM. First, we calculate a correlation matrix from two extracted features, and then predict TPS parameter θ by the regression network. There are 6 convolational layers in each feature extractor, and the regression network is composed of 4 convolution layers and a full-connection layer. The details of the GMM network are shown in Fig. 9.

AFWM(Appearance-flow-based warping module). Similar to [2], we set a dual pyramid network to extract features of (A_t, D_t, P_t) and $\hat{C}_t$ in different scales (c_N, b_N). For each scale, we adopt a Flow Network(FN) to estimate the dense appearance flow f_n. All FNs are cascaded to predict the finest flow f_N. In detail, the inputs of $n-th$ FN are (c_n, b_n) and f_{n-1}, and the output is f_n.

Encoder and Decoder in MPDT Generator. As described in the main text, there are three identical frame-level encoders and one frame-level decoder. There are 4 convolution layers with two times downsampling in encoder, and 4 convolution layers and 2 upsampling layers in decoder. The details of encoder and decoder are shown in Fig. 10.

A.2. Training Details

Optimization. We use Adam [5] as optimizer with $\beta_1 = 0.5$, $\beta_2 = 0.999$, a fixed learning rate of 0.0002 and all with a batch size of 16.

Hardware. All the codes are implemented by deep learning toolkit PyTorch and 8 NVIDIA V100 GPUs are used in our experiments. The training takes around 2 days for TPS-based warping module in Frame-level Anti-occlusion Warp Module, and around 3 days for Appearance-flow warping module in Frame-level Anti-occlusion Warp Module, and around 4 days for MPDT Generator.

Loss detail in Anti-occlusion Warp Module. The full loss $L_t^{TPS\text{-warp}}$ for TPS-based warping mudule are written as Eq. (1), we set the hyper-parameter λ_t^{sdc} to 0.04 and the second-order difference constraint L_t^{sdc} detailed as:

$$L_t^{sdc} = \sum_{p\in P} |\|pp_0\|_2 - \|pp_1\|_2| + |\|pp_2\|_2 - \|pp_3\|_2| + \|S(pp_0 - pp_1)\| + \|S(pp_2 - pp_3)\| \tag{13}$$

where symbol p denotes a certain sampled TPS control point and p_0, p_1, p_2 and p_3 are top, bottom, left and right point of p, respectively. The $S(p, pi)$ is the slope between p and p_i [7].

The full loss $L_t^{flow-\text{warp}}$ for Appearance-flow-based warping mudule are written as Eq. (2), we set the hyper-parameter λ_t^{sec} to 20 and the second-order smooth constraint L_t^{sec} detailed as:

$$L_t^{sec} = \sum_{i=1}^{N} \sum_{t} \sum_{\pi\in N_t} P(f_i^{t-\pi} + f_i^{t+\pi} - 2f_i^t) \tag{14}$$

Where P denotes the generalized charbonnier loss function [6], f_i^t is the t^{th} point on the appearance-flow maps of i^{th} scale. N_t indicates the set of horizontal, vertical, and both diagonal neighborhoods around the t^{th} point [2].

Matrix X in Appearance-flow Tracking Module. We use ridge regression [3] to track appearance-flow to obtain temporally smooth warped clothing sequences as shown in Eq. (3), the feature matrix X detail as:

$$X = \begin{bmatrix} x_{t-N}^{(1)} & x_{t-N+1}^{(1)} & \cdots & x_{t-1}^{(1)} \\ y_{t-N}^{(1)} & y_{t-N+1}^{(1)} & \cdots & y_{t-1}^{(1)} \\ x_{t-N}^{(2)} & x_{t-N+1}^{(2)} & \cdots & x_{t-1}^{(2)} \\ y_{t-N}^{(2)} & y_{t-N+1}^{(2)} & \cdots & y_{t-1}^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ x_{t-N}^{(W\times H)} & x_{t-N+1}^{(W\times H)} & \cdots & x_{t-1}^{(W\times H)} \\ y_{t-N}^{(W\times H)} & y_{t-N+1}^{(W\times H)} & \cdots & y_{t-1}^{(W\times H)} \end{bmatrix}$$

Here, x_m^n and y_m^n (for m = 1...N; n = 1...$W \times H$) are the directly estimated horizontal and vertical coordinate values for n_th point in appearance-flow at frame m, and we set N=3 for high computational efficiency. After Eq. (3) we reshape $\hat{f}_t^{1D}$ back to $\hat{f}_t$ with original spatial size $W \times H$.
Loss detail in MPDT Generator. The Full loss $L_{\text{try-on}}$ for

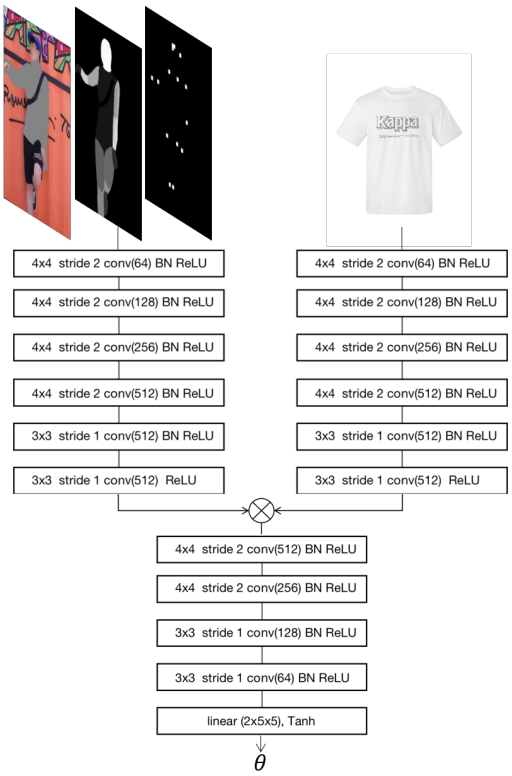

Figure 9. GMM network

MPDT generator are written as Eq. (12), we set the hyper-parameter $\lambda_1 = \lambda_3 = 1, \lambda_4 = 0.01, \lambda_2 = 10$. And $L_{l1}^{clothes}$ is the L1 loss in clothing regions and denoted as:

$$L_{l1}^{clothes} = \frac{\left\| M_{C_1}^T \odot (I_1^T - \tilde{I}_1^T) \right\|}{\left\| M_{C_1}^T \right\|} \tag{15}$$

where $\odot$ indicates element-wise multiplication.

The perceptual loss function L_{perc} is denoted as:

$$L_{\text{perc}} = \sum_{i=1}^{N_l} \frac{1}{Ne_i} [\left\| F^{(i)}(I_1^T) - F^{(i)}(\tilde{I}_1^T) \right\|] \tag{16}$$

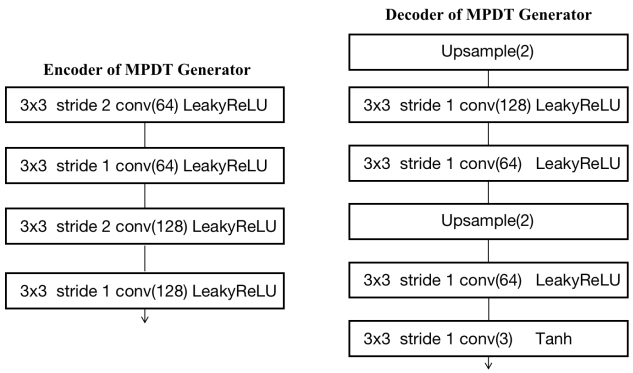

Figure 10. Encoder and Decoder architecture of MPDT Generator

where N_l is the number of features extracted from different layers of the VGG network F [4], and $F^{(i)}$ and R_i are the activation and the number of elements in the i^{th} layer of F, respectively.

The adversarial loss L_{TPGAN} is denoted as:

$$L_{\text{TPGAN}} = -E_{z \backsim p_{\tilde{I}_1^T}}(z)[D(z)] \tag{17}$$

And the optimization function for the T-PatchGAN discriminator is shown as follows:

$$\begin{aligned} L_{\text{TPGAN}} = & E_{x \backsim p_{I_1^T}}(x)[RELU(1 - D(x))] \\ & + E_{z \backsim p_{\tilde{I}_1^T}}(z)[RELU(1 + D(z))] \end{aligned} \tag{18}$$

B. Additional Experiments

B.1. Comparison with different warping methods

To demonstrate TPS-based warping can handle partial occlusions but lead to the misalignment between the warped clothes and the appearance-flow-based methods predicts more accurate deformations but are very sensitive to occlusions, we conduct a comparison experiment as shown in Fig. 11. Obviously, the shape of the clothes warped by TPS-based warping method is different from the shape of the clothes worn on the reference person. On the contrary, the shape of the clothes warped by the appearance-flow-based is better but the pixel-squeeze phenomenon appears around the doll. Compared to the baseline method, our two-stage anti-occlusion warping method warped a both accurate and anti-occlusion results.

B.2. Baseline methods without fusing background

As shown in Fig. 14, the baseline methods without fusing background fail to reconstruct complex background when training and testing on our collected wild virtual try-on dataset.

B.3. Qualitative Results

We provide additional qualitative results to demonstrate our model's capability of generating a temporally smooth and photo-realistic video. Fig. 15 show the qualitative comparison of the baselines in our wild virtual try-on dataset. Fig. 16 and Fig. 17 show additional results of ClothFormer in our dataset and the VVT dataset, respectively.

C. Failure Cases and Limitations

As shown in the forth column of Fig. 13, similar to existing parser-based methods, when parsing results are inaccurate (the grape pattern on the clothes is predicted as occlusion), ClothFormer generates visually terrible try-on images with noticeable artifacts in the inaccurate region. However, as shown in the sixth column of Fig. 13, ClothFormer can produce a realistic and natural video result when

we fix the parsing results manually, which shows it can be a valuable future direction to generate try-on videos by developing a parser-free video virtual try-on method. As shown in Fig. 12, ClothFormer generates a redundant sleeve when short-sleeve clothes to long-sleeve clothes, which might be solved by predicting a human-parsing maps of a person wearing the target clothes to guide the try-on video synthesis like ACGPN [7] and VITON-HD [1] does, but we need to predict a temporally smooth parsing sequence to avoid flickering results in time dimension.

One of the limitations of our model is that ClothFormer is not able to generate clothes with textured patterns on the back, instead, ClothFormer learns a pure color on the back due to most clothes are pure color on the back in both VVT dataset and our dataset. The root cause of the above problem is that only the front of the target clothes are available, we believe that ClothFormer has the capability to cover the case when training with both front of the target clothes and back of the target clothes as input.

References

[1] Seunghwan Choi, Sunghyun Park, Minsoo Lee, and Jaegul Choo. Viton-hd: High-resolution virtual try-on via misalignment-aware normalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14131–14140, 2021. 3

[2] Yuying Ge, Yibing Song, Ruimao Zhang, Chongjian Ge, Wei Liu, and Ping Luo. Parser-free virtual try-on via distilling appearance flows. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8485–8493, 2021. 1

[3] Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970. 1

[4] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016. 2

[5] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 1

[6] Deqing Sun, Stefan Roth, and Michael J Black. A quantitative analysis of current practices in optical flow estimation and the principles behind them. *International Journal of Computer Vision*, 106(2):115–137, 2014. 1

[7] Han Yang, Ruimao Zhang, Xiaobao Guo, Wei Liu, Wangmeng Zuo, and Ping Luo. Towards photo-realistic virtual try-on by adaptively generating-preserving image content. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7850–7859, 2020. 1, 3

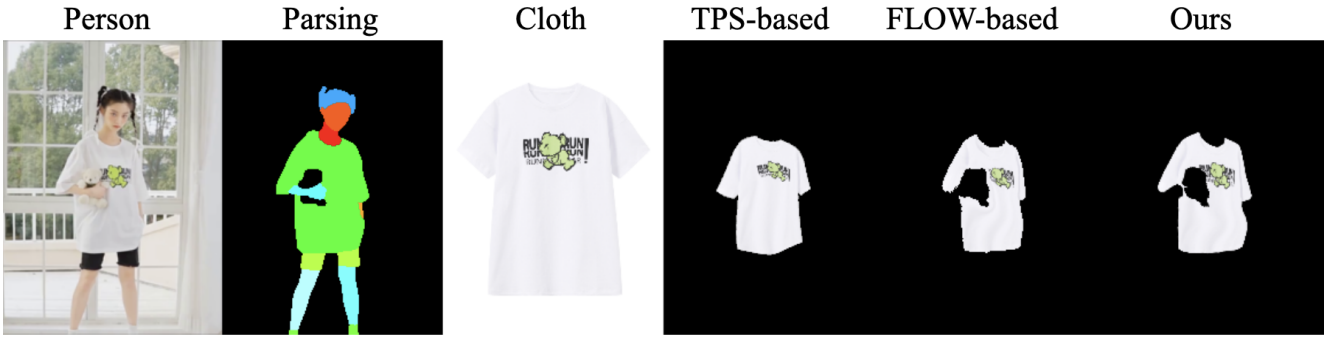

Figure 11. Qualitative comparison of warping methods.


Figure 12. Failure cases of redundant sleeve.

Figure 13. Failure cases of inaccurate parsing results.

Figure 14. Qualitative comparison of the baseline methods w/o fusing background.

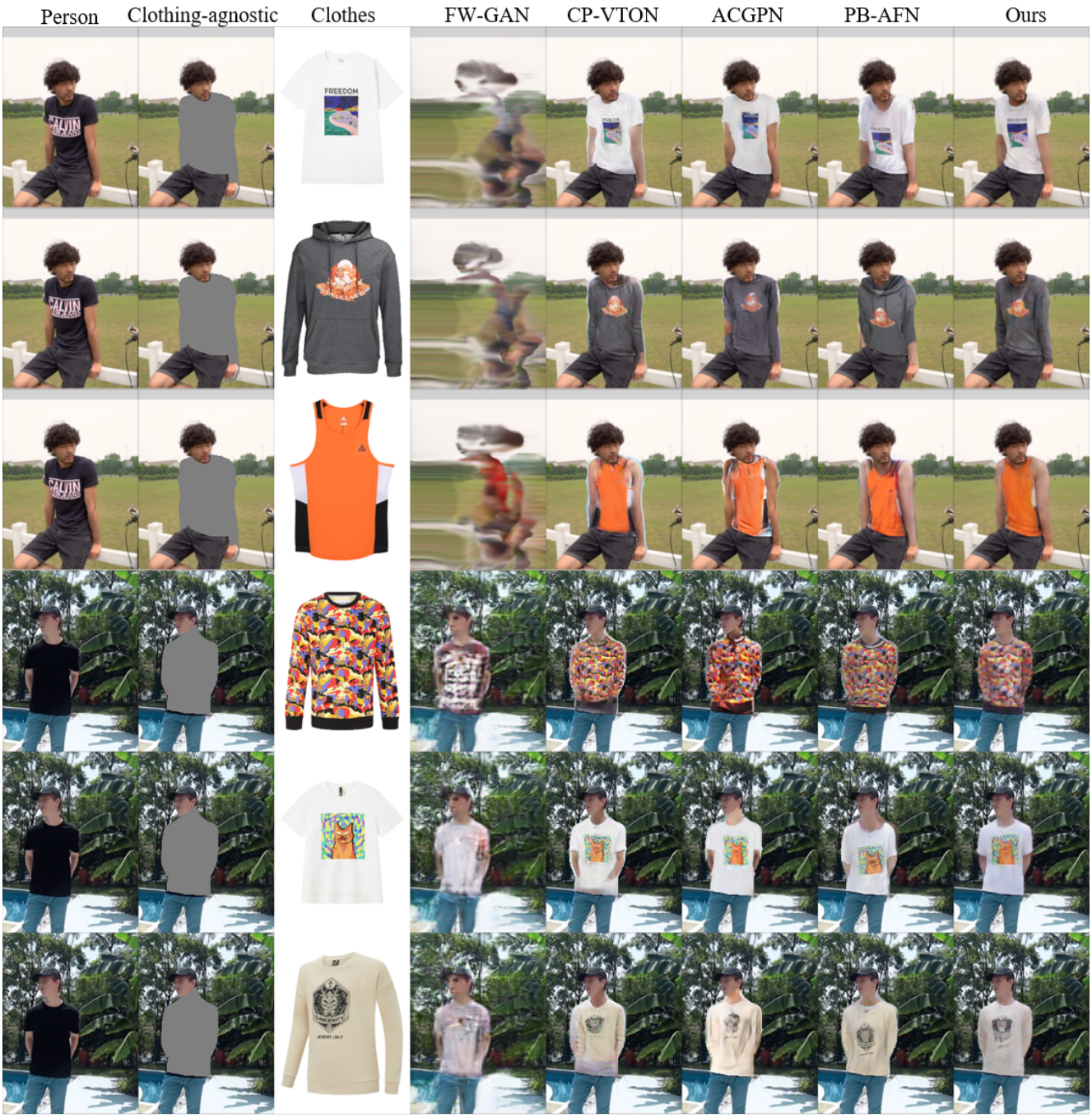

Figure 15. Qualitative comparison of the baseline methods.

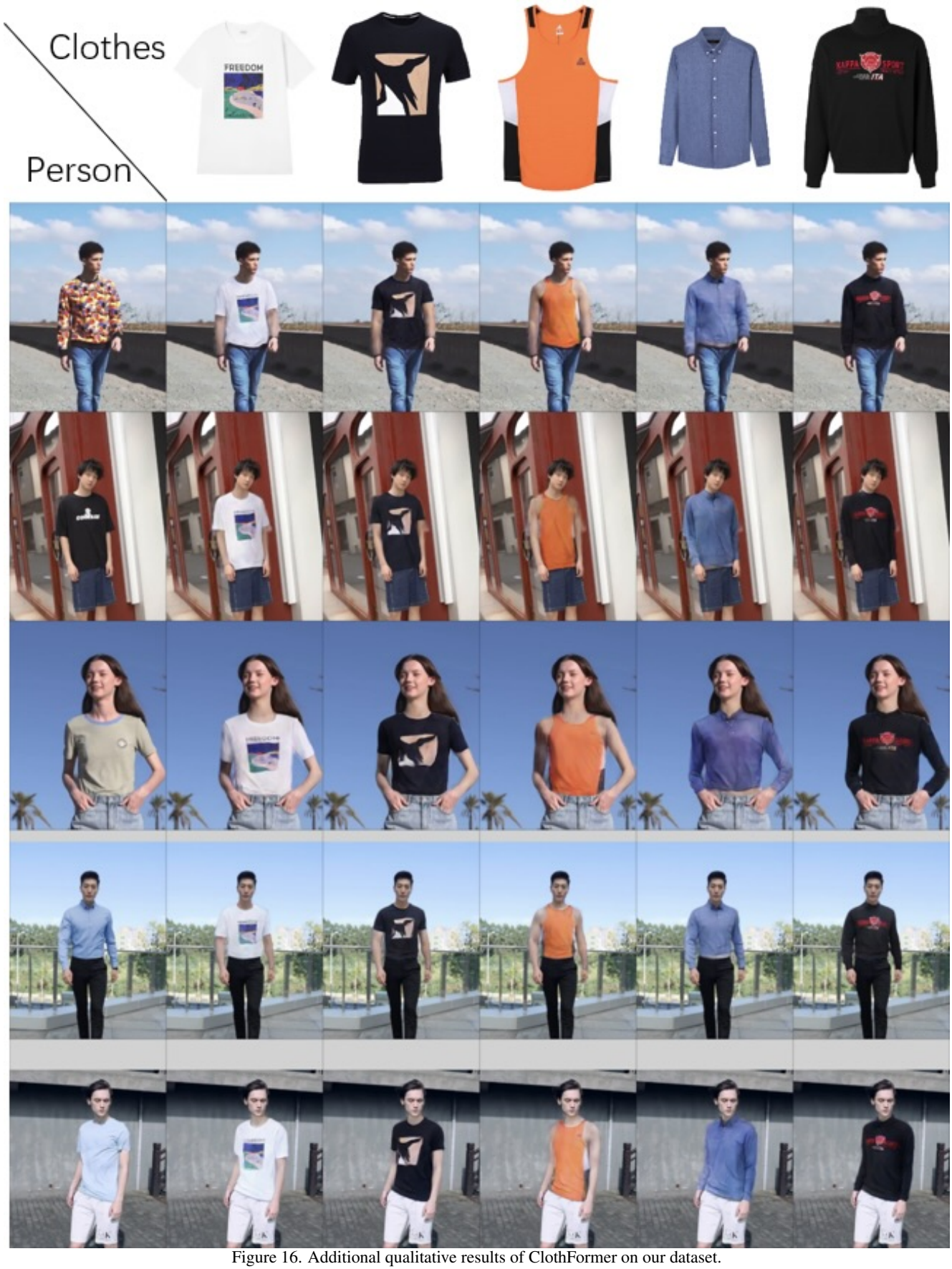

Figure 16. Additional qualitative results of ClothFormer on our dataset.

Figure 17. Additional qualitative results of ClothFormer on the VVT dataset.