# Controllable Human Image Generation with Personalized Multi-Garments

Supplementary Material

## A. Implementation Details

### A.1. Training and Inference

We train our decomposition module on 68,296 pairs of a human image and a single reference garment image at 512×384 resolutions with a fixed learning rate of 1e-5 using Adam optimizer [4]. We train for 140K iterations with a total batch size of 32 using 4 H100 GPUs.

For the composition module, we train on 54K pairs of a human image and multiple reference garment images at 768×576 resolution with a fixed learning rate of 1e-5 and Adam optimizer. We train for 115K iterations with a total batch size of 48 using 8 H100 GPUs.

During the inference, we generate images using the DDPM [3] sampler with 50 denoising steps. We apply classifier-free guidance (CFG) [2] with the text conditioning $\mathbf{c}$ and garment image conditioning $\mathbf{g}$ as follows:

$$\hat{\boldsymbol{\epsilon}}_\theta(\mathbf{x}_t; \mathbf{g}, \mathbf{c}, t) = w \cdot (\boldsymbol{\epsilon}_\theta(\mathbf{x}_t; \mathbf{g}, \mathbf{c}, t) - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t; t)) + \boldsymbol{\epsilon}_\theta(\mathbf{x}_t; t),$$

where $\boldsymbol{\epsilon}_\theta(\mathbf{x}_t; \mathbf{c}, \mathbf{g}, t)$ denotes noise prediction output with text and garment image conditions, and $\boldsymbol{\epsilon}_\theta(\mathbf{x}_t; t)$ denotes the unconditional noise prediction output. We use a guidance scale of $w = 2.0$ for sampling.

### A.2. Single reference Paired Dataset

To train the decomposition network, we collect pairs of a human image and a single reference garment image from VITON-HD, DressCode, and LAION-Fashion datasets. Specifically, we gather 11,647 upper garments and human images from the training dataset on VITON-HD. We also collect 13,563 upper garments, 7,151 lower garments, 27,677 dresses paired with human images from DressCode. For LAION-Fashion dataset, since it consists of single reference pairs without categorical information, we use CLIP [6] model to classify the garment image. We define 19 different garment category texts and match the garment image with the category text of the highest similarity score, resulting in 5,675 bags and 1,599 shoes, 826 scarf, and 159 hats in the training data. We provide examples of collected single reference garment and human image pairs in Fig. 1.

### A.3. Dual-Condition Classifier-free Guidance

Since we have dual conditions of text condition $\mathbf{c}$ and garment image condition $\mathbf{g}$, one can apply classifier-free guidance for two conditions following [1]. Formally:

$$\begin{aligned}\hat{\boldsymbol{\epsilon}}_\theta(\mathbf{x}_t; \mathbf{g}, \mathbf{c}, t) = & w_c \cdot (\boldsymbol{\epsilon}_\theta(\mathbf{x}_t; \mathbf{g}, \mathbf{c}, t) - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t; \mathbf{g}, t)) \\ & + w_g \cdot (\boldsymbol{\epsilon}_\theta(\mathbf{x}_t; \mathbf{g}, t) - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t; t)) \\ & + \boldsymbol{\epsilon}_\theta(\mathbf{x}_t; t),\end{aligned}$$

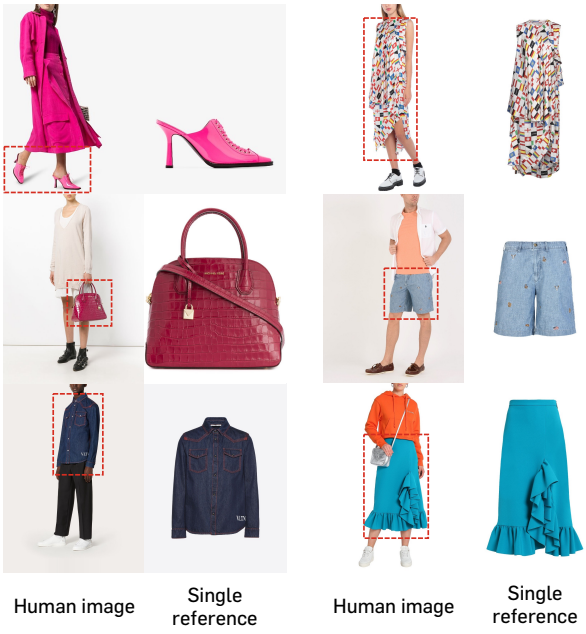


Figure 1. **Examples of training data for decomposition module.** We collect pairs of a human image and a single reference garment image from public datasets including VITON-HD, DressCode, and LAION-Fashion. It consists of various garments in different categories, *e.g.*, shirts, pants, shoes and bags *etc*.

where $w_c > 0$ and $w_g > 0$ denotes a guidance scale for text conditioning and garment image conditioning, respectively. Increasing $w_g$ encourages generated images to more similar to the reference garment images, and increasing $w_c$ guides the generated images to better align with the given text prompt. While we adopt $w_g = 2.0$ and $w_c = 2.0$ for all experiments, users can adjust the guidance values to customize the generated images according to their preferences.

## B. Synthetic Dataset Construction

In this section, we provide a detailed explanation of the data curation process with visualizations.

### B.1. Filtering Strategy

As illustrated in **??**, we apply filtering on our synthetic paired data based on the image similarity between the segmented and generated garments. Among several possible metrics, we try LPIPS, CLIP score, and DreamSim, and empirically find that DreamSim aligns the most with human perception. As shown in Fig. 2, DreamSim can measure the similarity aligned with human perception and filters out undesirable samples while CLIP and LPIPS struggle. For example, LPIPS determines that similar garments


Figure 2. **Examples of pairs filtered out by different similarity metrics.** We present examples of generated garment images and their corresponding human images that were excluded based on various image similarity metrics. Using LPIPS, garments with complicated patterns are filtered out, and using CLIP score, inner layer garments are filtered out even when they are considered identical in human perception. In contrast, DreamSim captures the distance between images in a way aligned with human perception, filtering out undesirable pairs.

do not resemble each other, even if garment pairs look identical to humans, especially when they contain intricate patterns or stripes. Also, CLIP fails to identify the same garments, mainly when garments are inner layers under jackets, whereas DresmSim captures similarity in a way aligned with human perception, filtering out the undesirable pairs.

We adopt DreamSim for measuring the distance between segmented garments and generated garments. We visualize human images and generated garment images based on the image distance value in Fig. 3. With the distance value $d \geq 0.6$, we observe that the generated garment is inconsistent with the garment on the human image, and with $0.4 \leq d < 0.6$, fine details are not fully preserved. On the other hand, with $d < 0.4$, generated garments closely resemble the actual garments.

## B.2. Synthetic Dataset Examples

We provide visualizations of the synthetic paired dataset generated by our decomposition network in Fig. 4. The synthetic dataset contains high-quality pairs of a human image and *multiple* reference garments. The decomposition network can generate product garment images on different categories, even with challenging garments such as one-shoulder sweaters (Third-row in Fig. 4).

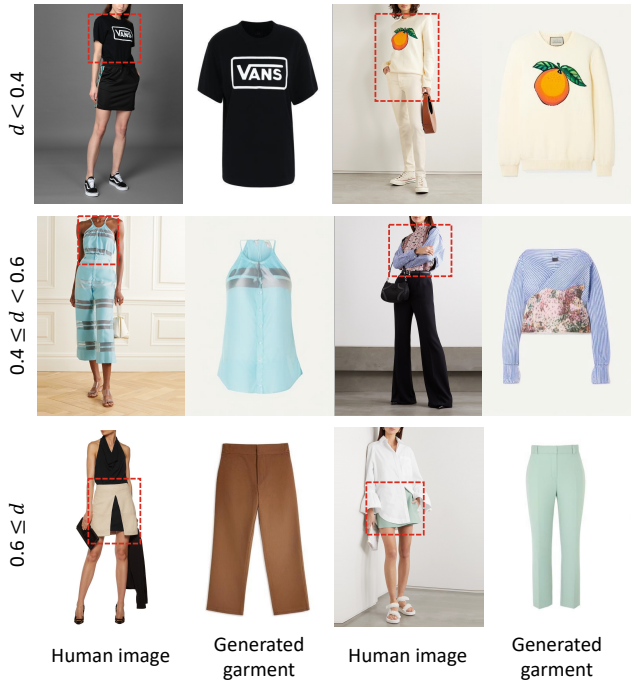


Figure 3. **Examples of generated garment images with different image distance values.** We provide examples of generated garment images and corresponding human images, varying the distance values measured by DreamSim. With the distance value $d \geq 0.4$, generated garments are inconsistent with the actual garment, while for $d < 0.4$, the generated garments closely resemble the actual garment.

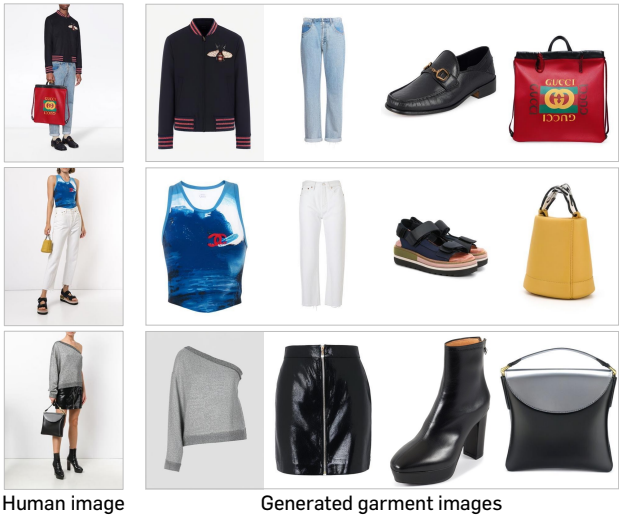


Figure 4. **Examples of our synthetic paired data.** We visualize our synthetic pairs of a human image and multiple garment images. Our decomposition module generates high-quality garment images in product view on different categories including shirts, pants, shoes and bags.

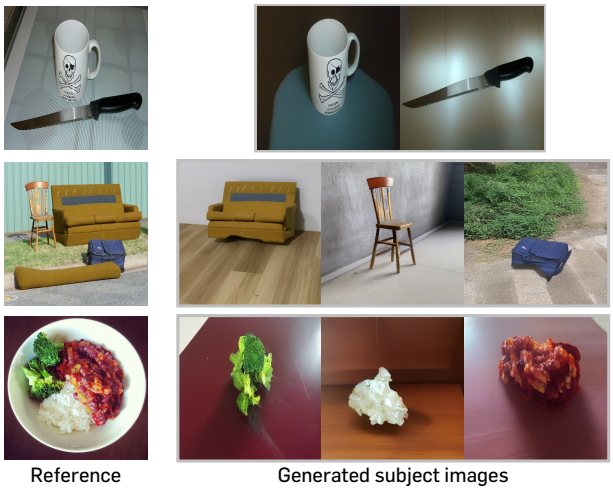


Figure 5. **Examples of synthetic paired data generated by the decomposition module trained on MVImgNet [9].** We show the potential extension of our decomposition module to the general domain. Given an image containing common objects such as cups, chairs, and broccoli, the decomposition module generates each object in a different view, constructing paired data. Reference images are obtained from COCO [5].

## C. Applications of Decomposition module

In this section, we explore the potential applications of our decomposition module, including applying it on the general domain and using it as a multi-view image generator.

### C.1. Synthetic Paired Data on General Domain

Recent work [8] demonstrates remarkable performance in diverse image generation tasks by leveraging large-scale paired data, underscoring the importance of paired datasets in image generation. We have demonstrated our decomposition module's capability to generate high-quality paired data in the fashion domain, and we further explore its potential for applicability to the general domain. Specifically, we train the decomposition network on MVImgNet [9] dataset, which contains large-scale object images in multi-view from 238 classes. As shown in Fig. 5, the network decomposes each object in different views from reference images, demonstrating its potential for broader applications and inspiring future research.

### C.2. Multi-view Image Generator

We show that the decomposition network can be used as a multi-view image generator. By utilizing the decomposition network with segmented single-subject images, one can generate different views of the reference subject images while faithfully preserving their identity. In Fig. 6, we present multi-view images generated by the decomposition module using subject images obtained from DreamBooth [7]. These multi-view images can be utilized for various applications, such as data augmentation.

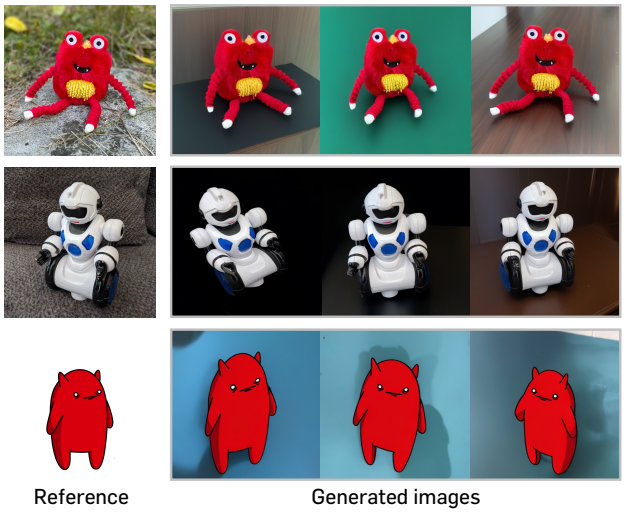


Figure 6. **Examples of generated subjects in multi-view by the decomposition module trained on MVImgNet.** The decomposition module can serve as a multi-view generator for single-subject images. Subject images are from DreamBooth [7].

## D. Additional Qualitative Results

We provide more visualizations of human images generated by BootComp. We show more qualitative comparisons of BootComp with baselines in Fig. 8. We also showcase additional human images with multiple reference garments generated by BootComp in Fig. 9 and more visualizations of application results, including controllable generation, stylization, and personalized generation in Fig. 10.

## E. Limitations

While BootComp is capable of generating human images with various categories of garments, it sometimes struggles to place hats on humans naturally. This arises from the limited number of hat images in the training data. One can address this by scaling up the paired data simply using our data generation pipeline. Also, BootComp fails to preserve tiny details such as letters, which is attributed to the limitations of the backbone model, SDXL. This can be relieved by replacing backbone to other diffusion models trained with better VAE encoders with larger number of channels.

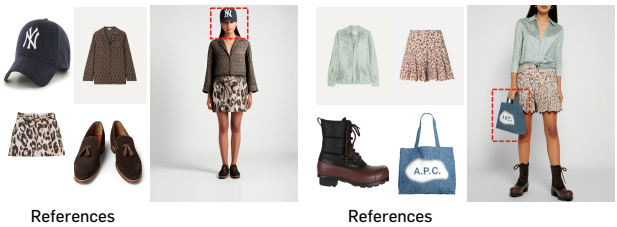


Figure 7. **Limitations of BootComp.** BootComp struggles on naturally dressing hats and preserving tiny details like letters.

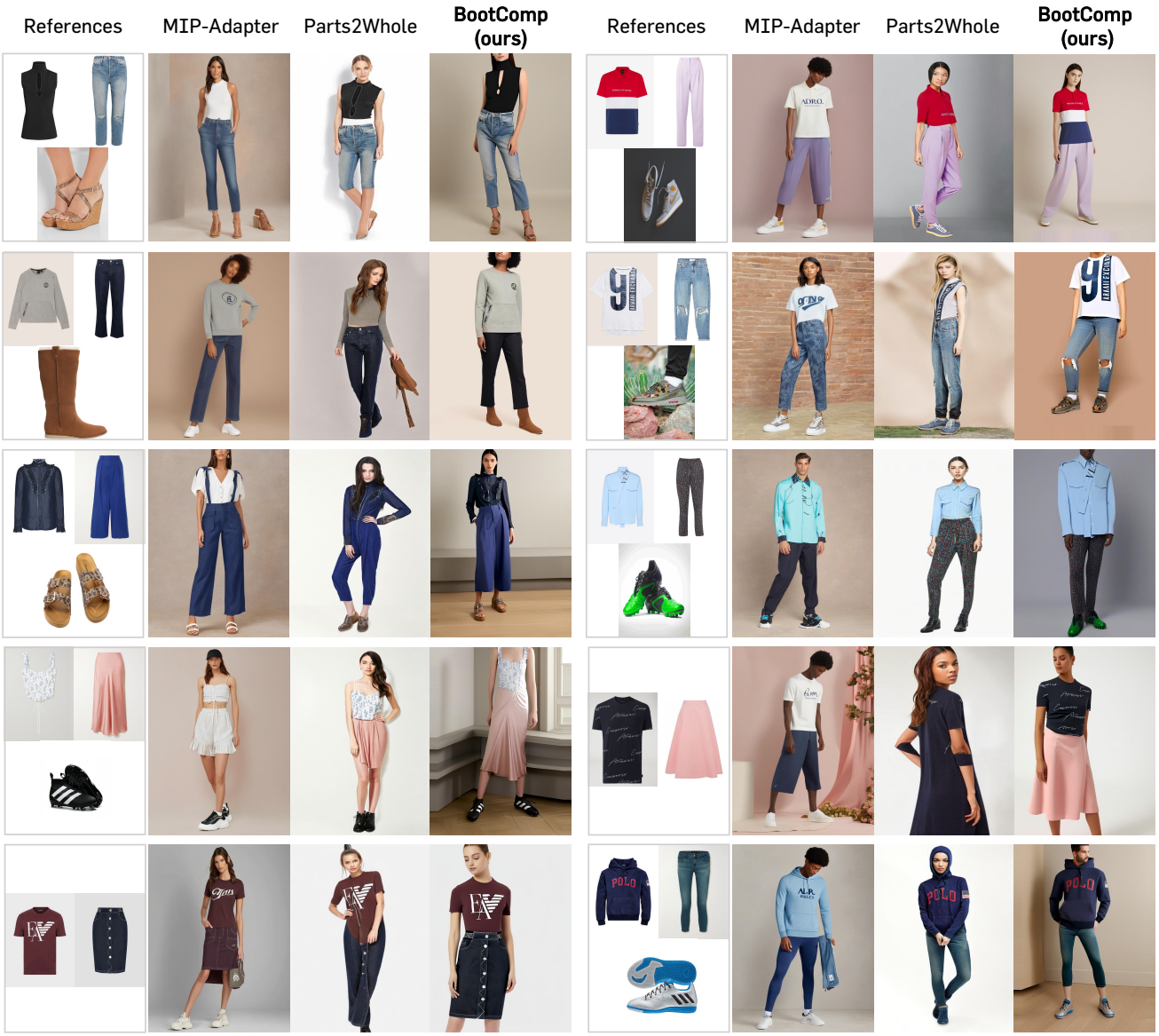


Figure 8. **More qualitative comparisons.** BootComp generates realistic human images wearing multiple reference garments, faithfully preserving fine-details of each garment, while baselines often generate inconsistent garment images and blend reference garments.

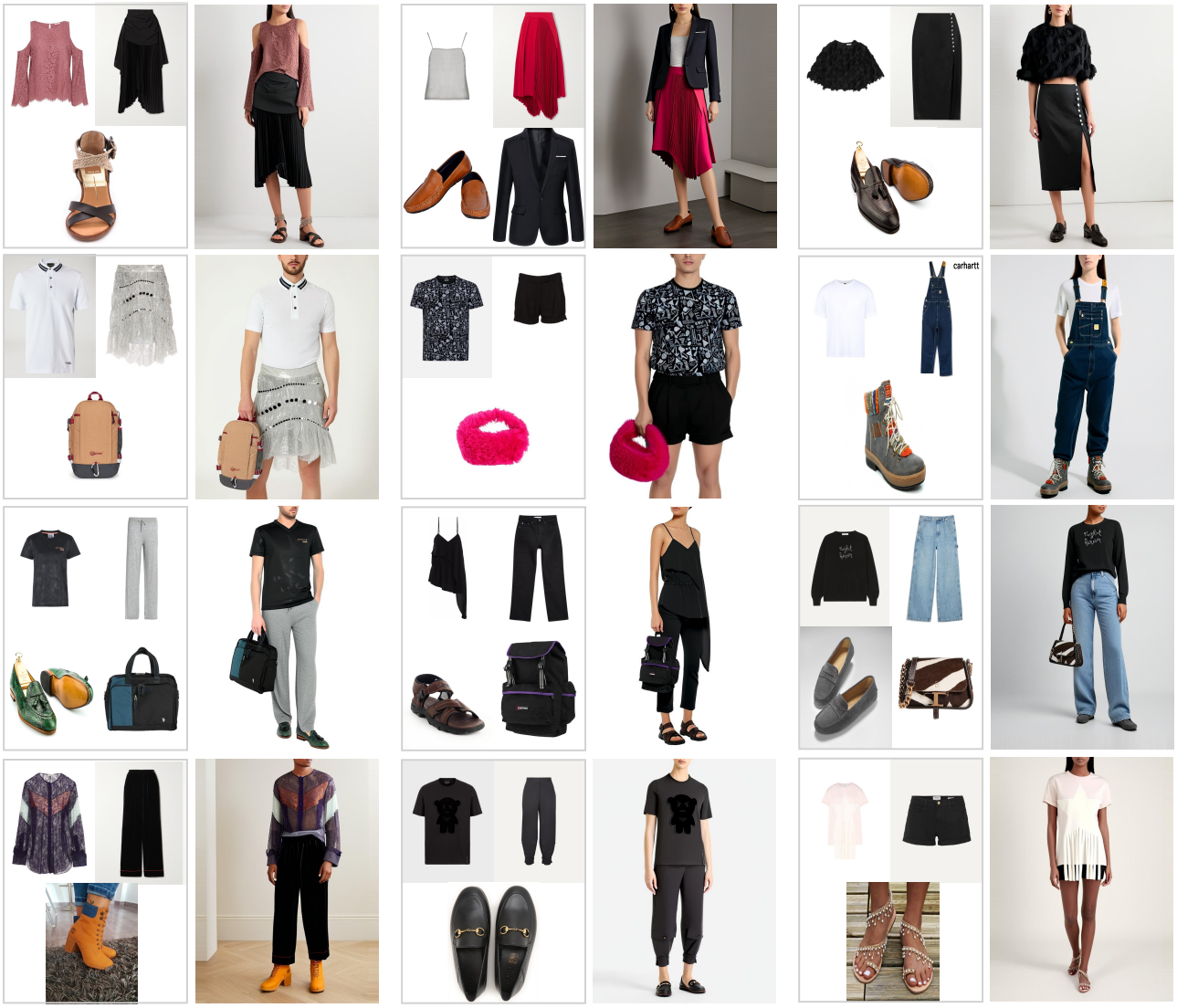


Figure 9. **Generated human images by BootComp.** BootComp can realistically dress humans with diverse categories of garments, including bags and shoes, which are not available for previous approaches. BootComp is capable of dressing complex combinations such as jackets and inner layers (First row, second column) and less common garments such as overalls (Second row, third column). Also, BootComp can address challenging garments such as asymmetric-length garments and sandals (Third row, second column), and garments with unique details (Last row, third column).

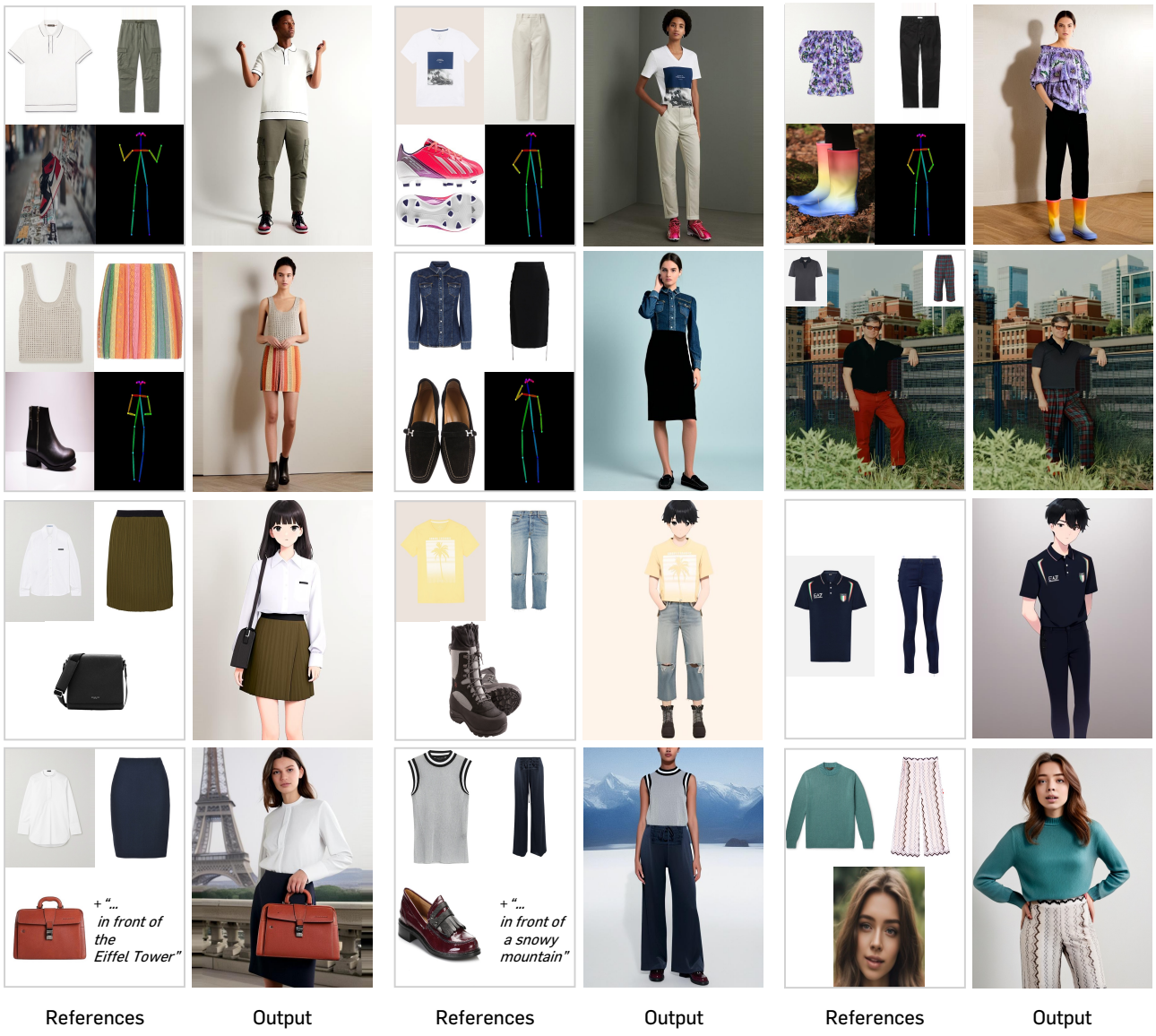


Figure 10. **Application results by BootComp.** BootComp is capable of generating human images with various conditions. By using structural conditions, it can control poses in the generated images. With text prompts, BootComp can manipulate the backgrounds of images. Additionally, it supports personalized generation through virtual try-on and face-based generations.

## References


[1] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2023. 1

[2] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 1

[3] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1

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

[5] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 3

[6] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 1

[7] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 3

[8] Shitao Xiao, Yueze Wang, Junjie Zhou, Huaying Yuan, Xingrun Xing, Ruiran Yan, Shuting Wang, Tiejun Huang, and Zheng Liu. Omnigen: Unified image generation. *arXiv preprint arXiv:2409.11340*, 2024. 3

[9] Xianggang Yu, Mutian Xu, Yidan Zhang, Haolin Liu, Chongjie Ye, Yushuang Wu, Zizheng Yan, Chenming Zhu, Zhangyang Xiong, Tianyou Liang, et al. Mvimgnet: A large-scale dataset of multi-view images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9150–9161, 2023. 3