

---

# Appendix: ShoeFit: A New Dataset and Dual-image-stream DiT Framework for Virtual Footwear Try-On

---

Anonymous Author(s)

Affiliation

Address

email

- 1 • In Section A, we provide a more detailed discussion about the limitations of ShoeFit.
- 2 • In Section B, we provide additional details about the MVShoes datasets, including data
- 3 processing and data statistics.
- 4 • In Section C, we provide preliminary knowledge about the FLUX (6) model, as well as how
- 5 LayeredRefAttention modules are applied in the Single Stream DiT Block.
- 6 • In Section D, we provide additional details about the experimental setup, including baseline
- 7 training, data augmentation, and hyperparameters.
- 8 • In Section E, we present a multitude of ShoeFit generated images, including more ablation
- 9 results, more comparisons with the baselines, with additional results displayed in challenging
- 10 scenarios.

## 11 A Limitations and Future Work

12 We propose ShoeFit, a dual-stream DiT frame-  
13 work that addresses the critical challenges  
14 of viewpoint misalignment and background-  
15 induced color distortion in VFTON by Multi-  
16 View Conditioning and LayeredRefAttention  
17 modules. However, certain limitations persist.  
18 Firstly, similar to the common constraints faced  
19 by existing generative models, the model oc-  
20 casionally struggles to accurately render small  
21 logos and intricate text due to their small portion  
22 in images and high-frequency variations. We il-  
23 lustrate this limitation in Fig. 1. These small  
24 logos are particularly critical for e-commerce  
25 sellers as they often convey brand information  
26 that consumers care about. Therefore, we plan  
27 to develop a detailed supplementary condition-  
28 ing method in future research to pre-detect and  
29 enhance the injection of such patterns, funda-  
30 mentally addressing this issue. Secondly, we  
31 aim to introduce explicit 3D geometric and ma-  
32 terial priors in the future to achieve more robust  
33 multi-view representation and refined visual fidelity.



Figure 1: Similar to the common constraints faced by existing generative models, ShoeFit occasionally struggles to accurately render small logos and intricate text due to their small portion in images and high-frequency variations.

## 34 B MVShoes Dataset Supplement

35 In the main body of the paper, we provide a brief description of the data processing pipeline and  
36 a rough visual presentation of the dataset statistics in terms of Rich Scenes and Diverse Footwear  
37 Categories due to space constraints. In this section, we offer more detailed information on the data  
38 processing pipeline to ensure reader comprehension. Additionally, we present the specific distribution  
39 statistics of MVShoes in a quantitative table format.

40 **Data Processing Details.** Given a dataset comprising raw images of various shoes and human  
41 models, we employ Qwen2.5-VL (1), which operates with 7 billion parameters, to distinguish  
42 between footwear images and human model images. The prompting framework utilized is as follows:  
43 “Analyze the provided image and determine whether it depicts a model image or a shoe image. A  
44 model image is defined as one that portrays a human model’s lower body or legs adorned with shoes,  
45 while a shoe image solely comprises images of shoes without any human presence. Assign a value of  
46 ‘1’ if it qualifies as a model image and ‘0’ if it does not. ”

47 Following this classification, we extract image features from the shoe images using the CLIP (13)  
48 model, implementing a similarity threshold of 0.9 to effectively eliminate duplicate images within  
49 the dataset. Subsequently, we perform segmentation by SAM (5) to isolate the shoe regions in  
50 each image, aligning the DINO (11) features to filter out shoes-model data pairs exhibiting inner  
51 similarities greater than 0.8.

52 This procedure facilitates the initial construction of human-footwear try-on triplets, which are then  
53 subjected to manual filtering to address any potential errors or oversights. Additionally, we exclude  
54 visually blurred images and compile statistical analyses pertaining to shoe categories and try-on  
55 scenarios. In accordance with methodologies established in DWPose (17) and GroundingDINO (10),  
56 we extract foot poses and shoe masks, ultimately resulting in the generation of high-resolution,  
57 category-comprehensive try-on triplets. All models referenced above are open-source, and their  
58 respective URLs are provided as follows:

- 59 • Qwen2.5-VL-7B (1): <https://huggingface.co/Qwen/Qwen2.5-VL-7B-Instruct>
- 60 • SAM (5): <https://github.com/facebookresearch/segment-anything>
- 61 • CLIP (13): <https://github.com/openai/CLIP>
- 62 • DINO (11): <https://github.com/facebookresearch/dinov2>
- 63 • DWPose (17): <https://github.com/IDEA-Research/DWPose>
- 64 • GroundingDINO (10): <https://github.com/IDEA-Research/GroundingDINO>

65 **Dataset Statistics.** The dataset features diverse shoe categories and rich human scenes. We present  
66 the quantitative distribution statistics of MVShoes in Tab. 1 and Tab. 2. We also provide more visual  
67 samples from MVShoes datasets in Fig. 2

## 68 C Method Supplement

### 69 C.1 Preliminary

70 **FLUX.1** Our ShoeFit is an extension of Stable Diffusion 3 (14) and FLUX.1 (6), which are the  
71 most commonly used text-to-image diffusion models based on Flow Matching(9) and DiT(12).  
72 FLUX.1 (6) employs a variational autoencoder (4) (VAE) that consists of an encoder  $\mathcal{E}$  and a decoder  
73  $\mathcal{D}$  to enable image representations in the latent space. It is also equipped with rotary positional  
74 embeddings (RoPE)(15) and denoise-text-stream attention layers to improvement performance.  
75 FLUX.1 implements an actual two-dimensional RoPE scheme for encoding spatial positions in the  
76 latent space:

$$\omega_d = \frac{1}{\theta^{2d/D}}, \text{ for } d = 0, 1, \dots, D/2 - 1, \quad (1)$$

77 where  $\theta$  is typically set to 10000. The position encoding applies a rotation matrix:

$$\begin{bmatrix} \cos(\omega_d \cdot \mathbf{pos}) & -\sin(\omega_d \cdot \mathbf{pos}) \\ \sin(\omega_d \cdot \mathbf{pos}) & \cos(\omega_d \cdot \mathbf{pos}) \end{bmatrix} \quad (2)$$

Table 1: Shoe Category Statistics. We report the sample count for each subcategory, with its proportion indicated in parentheses.

Primary Categories	Subcategory
<i>Casual Shoes</i> : 3158 (43.23%)	Lifestyle Casual Shoes: 803 (10.99%) Sneakers: 1428 (19.55%) Canvas Shoes: 141 (1.93%) Children Casual Shoes: 99 (1.36%) Men Casual Shoes: 687 (9.40%)
<i>Athletic Shoes</i> : 558 (7.64%)	Running Shoes: 322 (4.41%) Basketball Shoes: 121 (1.66%) Training Shoes: 45 (0.62%) Football Shoes: 70 (0.96%)
<i>High-Heel Shoes</i> : 895 (12.25%)	Classical High-heel Shoes: 340 (4.65%) Ladies Casual Shoes: 401 (5.49%) Mary Jane Shoes: 154 (2.11%)
<i>Boots</i> : 1045 (14.31%)	Ankle Boots: 416 (5.69%) Chelsea Boots: 56 (0.77%) High Boots: 153 (2.09%) Snow Boots: 183 (2.51%) Martin Boots: 237 (3.24%)
<i>Sandals</i> : 279 (3.82%)	Flip Flops: 183 (2.51%) Beach Sandals: 36 (0.49%) Strap Sandals: 60 (0.82%)
<i>Dress Shoes</i> : 389 (5.33%)	Loafers: 209 (2.86%) Formal Leather Shoes: 180 (2.46%)
<i>Slippers</i> : 981 (13.43%)	Clogs: 82 (1.12%) Thong Slippers: 18 (0.25%) House Slippers: 881 (12.06%)

Table 2: Human scene statistics.

Scene Types	Number of samples	Percentage
<i>Top-down Foot</i>	1719	23.53%
<i>Horizontal Foot</i>	4330	59.27%
<i>Half-body Model</i>	833	11.40%
<i>Full-body Model</i>	423	5.79%

78 This rotation is applied to query and key vectors in the attention mechanism, enabling the model to  
79 capture relative positional relationships in the latent space.

80 **Flow Matching** Flow matching (9) aligns the flow of information between noise  $\epsilon$  and data  
81 distributions by optimizing a velocity field  $u_t$ , which progressively converts noise into data over  
82 time. This technique ensures that the generative model maps the noise distribution to the actual data  
83 distribution in a structured manner. The text encoders (13)  $\tau_\theta$  are employed to deal with the given  
84 text prompt  $y$ . The flow matching loss is defined as follows:

$$\mathcal{L} = E_{t, p_t(z|\epsilon), p(\epsilon), y} \left[ \|v_\Theta(z, t, \tau_\theta(y)) - u_t(z|\epsilon)\|^2 \right]. \quad (3)$$

85 In this context,  $v_\Theta(z, t, \tau_\theta(y))$  signifies the conditional velocity field determined by the weights of  
86 the neural network, while  $u_t(z|\epsilon)$  represents the vector field created by the model to delineate the  
87 probabilistic trajectory between the noise and actual data distributions. The symbol  $E$  stands for the  
88 expectation, which involves either integration or summation over time  $t$ , latent variables  $z$ , conditions  
89  $y$ , and noise  $\epsilon$ . This expectation computes the mean of the squared differences for all conditions,  
90 ensuring that the model’s performance is evaluated over numerous instances to yield a dependable  
91 estimate of its generative capability.

## 92 C.2 LayeredRefAttention in Single-stream DiT Block

93 The original FLUX is a text-to-image model composed of a series of stacked MM-DiT (double-  
 94 stream) blocks, followed by a series of stacked single-stream DiT blocks. Due to space constraints,  
 95 the framework of the LayeredRefAttention module shown in the main text is its structure within  
 96 the double-stream DiT blocks. Here, we provide how it is used within a single-stream DiT block in  
 97 Fig. 4. Similar to the way in double-stream blocks, we employ a Squeeze-and-Excitation (SE) block  
 98 (3) followed by global average pooling over spatial dimensions to compute channel-specific weights:

$$P^{(n)} = \text{AvgPool}(SE(C_I^{(n)})) \in \mathbb{R}^{b \times c}. \quad (4)$$

99 Subsequently, we employ two linear layers  $F(\cdot)$  to extract background modulation parameters  
 100 and modulate foreground shoe features, filtering irrelevant environmental lighting and background  
 101 reflection to ensure faithful material preservation of shoes as:

$$\beta_{scale}^{(n)}, \beta_{shift}^{(n)} = F(P^{(n)}), \quad C_{fg}^{(n)} = (1 + \beta_{scale}^{(n)}) \cdot (LN(C_I^{(n)} \odot M_{fg}^{(n)})) + \beta_{shift}^{(n)}, \quad (5)$$

102 where  $LN(\cdot)$  means layer normalization and  $M_{fg}^{(n)}$  represents the binary shoe masks for conditioning  
 103 product image. The subsequent operations are then performed as in regular single-stream DiT blocks.

## 104 D Implementation Supplement

### 105 D.1 Baseline Training Details

106 We here provide detailed descriptions of the  
 107 training processes for the baseline models, focus-  
 108 ing specifically on the training configurations.  
 109 All models are trained on MVShoes at a res-  
 110 olution of  $768 \times 768$ , utilizing 6305 pairs for  
 111 training and 1000 pairs for testing.

112 For Flux.1 Fill (7), we implement the training  
 113 and inference pipeline by concatenating  
 114 footwear images with human model images. In  
 115 the trainable components, we apply LoRA (2)  
 116 with a rank of 64 to all attention modules in the  
 117 model and a Flux ControlNet (19) comprising  
 118 6 single layers and 6 double layers, injecting  
 119 the poses at every denoising step. The model is  
 120 trained for 6 days on 8 80GB-A100 GPUs using  
 121 DeepSpeed ZeRO-2, with a batch size of 4. We  
 122 utilize the AdamW optimizer, setting a constant  
 123 learning rate of  $3e-5$  for training, and operate  
 124 the model on a single A100 GPU for 25 steps  
 125 during inference.

126 For Flux.1 Redux (8), we adopt the same train-  
 127 ing settings as those used for FLUX.1-Fill. Ad-  
 128 ditionally, we incorporate two linear layers for  
 129 projecting SigLip (18) features of the product shoe images, training these two layers concurrently  
 130 with the attention LoRAs and ControlNet. The parameters of SigLip remain frozen throughout the  
 131 training process. We retain the same inference settings as employed in FLUX.1-Fill.

132 For OOTDiffusion (16), we execute the training and inference pipelines based on the official code.  
 133 We also implement ControlNet to facilitate pose injection. The model is trained for 6 days on 4  
 134 80GB-A100 GPUs with DeepSpeed ZeRO-2, at a batch size of 8. During inference, the model is run  
 135 on a single A10 GPU for 25 steps.

136 For CatVTOn-Flux-Lite-2V, we maintain the text stream for FLUX-lite and concatenate footwear  
 137 images with human model images. The model undergoes training for 7 days on 12 80GB-A100 GPUs  
 138 utilizing DeepSpeed ZeRO-2, with a batch size of 2. All other training and inference configurations  
 139 align with those of ShoeFit-2V.

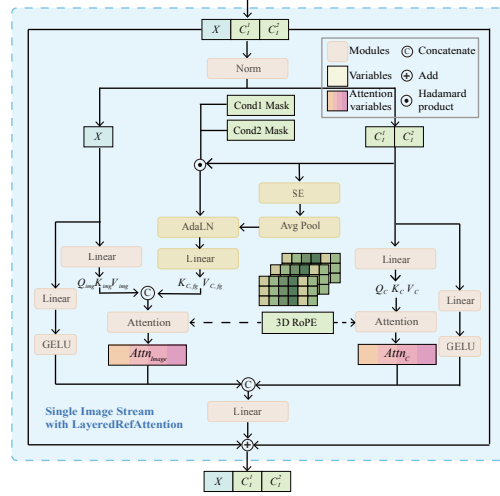


Figure 4: We provide how LayeredRefAttention is used within a single-stream DiT block.



## 140 D.2 Data Augmentation

141 We have implemented data augmentation techniques that could potentially enhance the model’s  
142 generalization ability as well as its fidelity performance. Specifically, the data augmentation operations  
143 include (a) horizontal flipping of images, (b) resizing footwear and human figures through padding  
144 (up to 10% of the image size), (c) randomly adjusting the image’s hue within a range of -5 to +5, and  
145 (d) randomly adjusting the image’s contrast within a specified range (between 0.8 and 1.2 times the  
146 original contrast). Each of these operations occurs independently with a 50% probability. Moreover,  
147 these operations are simultaneously applied to both the footwear and model images.

## 148 D.3 LayeredRefAttention Hyperparameters

149 In the LayeredRefAttention layer, we primarily introduced a new Linear layer and an SE block.  
150 The Linear layer for the foreground follows the same dimensions as other linear layers within the  
151 module, which is the *hidden size*. The SE block takes input features with the same dimensions  
152 of *hidden size*, and internally, we use a compression rate of  $reduction = 4$  for SE Block, as  
153  $Linear(channels, channels/4)$ ,  $ReLU()$ ,  $Linear(channels/4, channels)$ .

## 154 E More visual results

155 Fig. 3 provides more results about the ablation study. We highlight the improvements by all three  
156 components of the method in red.

157 Fig. 5 provides more results on MVShoes for comparisons between the baselines and ShoeFit. For a  
158 fair comparison, we report the results of the single-view conditioning version of all methods. Our  
159 method substantially improves rendering fidelity and robustness under challenging real-world product  
160 shoes, establishing a new benchmark in high-fidelity footwear try-on synthesis.

161 Fig. 6 provides more results on MVShoes for inspection to demonstrate that ShoeFit synthesizes  
162 high-fidelity and detail-faithful try-on results.



Figure 2: We provide more visual samples from MVShoes datasets.

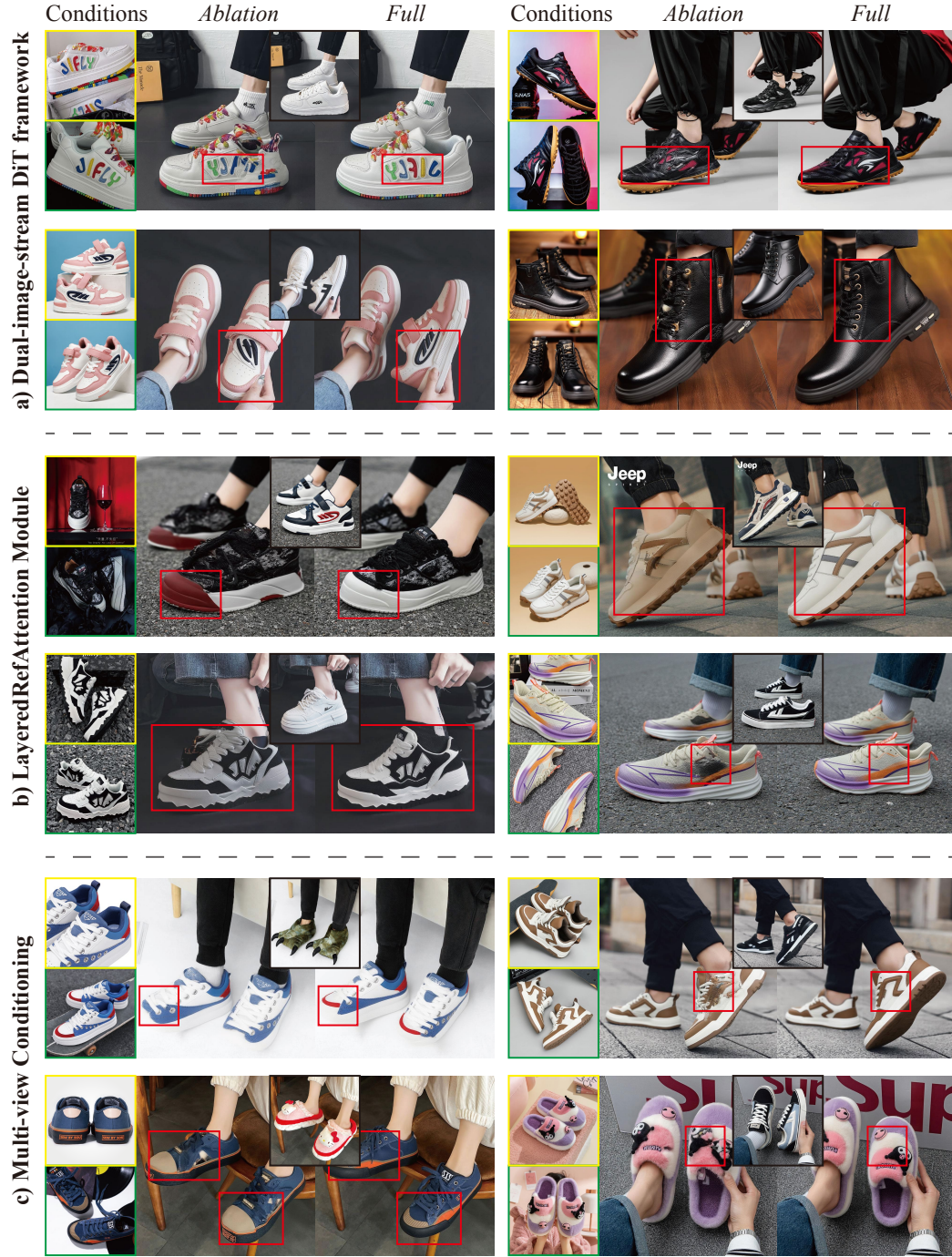


Figure 3: More visual results of the ablation study. We highlight the improvements by all three components of the method in red. Best viewed when zoomed in.





Figure 5: More visual comparisons on the MVShoes by ShoeFit. Best viewed when zoomed in.

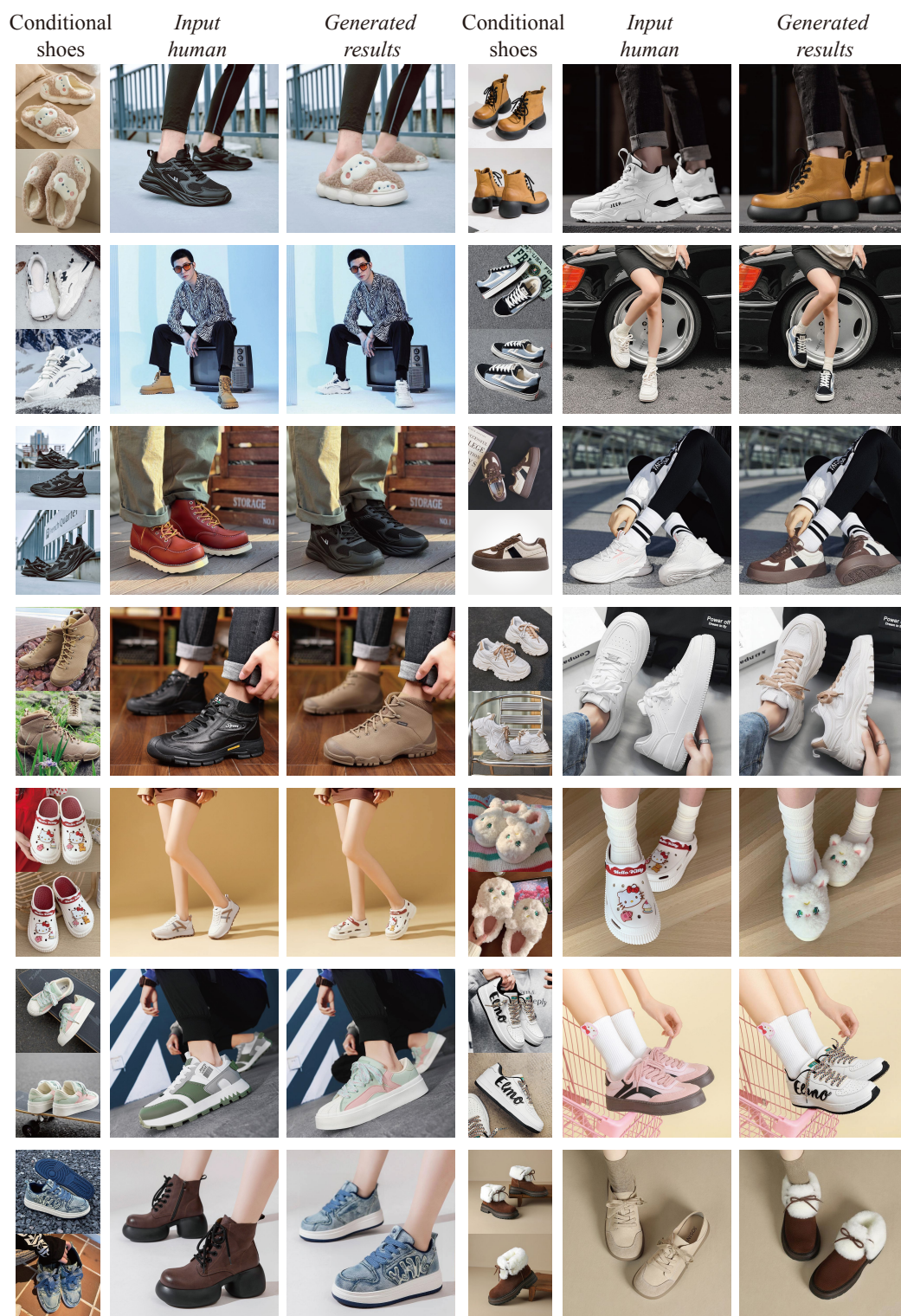


Figure 6: More visual results on the MVShoes by ShoeFit. Best viewed when zoomed in.



## References

- [1] Bai, J., Bai, S., Yang, S., Wang, S., Tan, S., Wang, P., Lin, J., Zhou, C., Zhou, J.: Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond (2023), <https://arxiv.org/abs/2308.12966>
- [2] Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al.: Lora: Low-rank adaptation of large language models. *ICLR* **1**(2), 3 (2022)
- [3] Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp. 7132–7141 (2018)
- [4] Kingma, D.P., Welling, M.: Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013)
- [5] Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., et al.: Segment anything. *arXiv preprint arXiv:2304.02643* (2023)
- [6] forest labs, B.: Flux.1-dev. <https://github.com/black-forest-labs/flux> (2024), <https://github.com/black-forest-labs/flux>
- [7] forest labs, B.: Flux.1-fill-dev. <https://huggingface.co/black-forest-labs/FLUX.1-Fill-dev> (2024), <https://huggingface.co/black-forest-labs/FLUX.1-Fill-dev>
- [8] forest labs, B.: Flux.1-redux-dev. <https://huggingface.co/black-forest-labs/FLUX.1-Redux-dev> (2024), <https://huggingface.co/black-forest-labs/FLUX.1-Redux-dev>
- [9] Lipman, Y., Chen, R.T., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747* (2022)
- [10] Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Jiang, Q., Li, C., Yang, J., Su, H., Zhu, J., Zhang, L.: Grounding dino: Marrying dino with grounded pre-training for open-set object detection (2024), <https://arxiv.org/abs/2303.05499>
- [11] Oquab, M., Darcet, T., Moutakanni, T., Vo, H.V., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., Howes, R., Huang, P.Y., Xu, H., Sharma, V., Li, S.W., Galuba, W., Rabbat, M., Assran, M., Ballas, N., Synnaeve, G., Misra, I., Jegou, H., Mairal, J., Labatut, P., Joulin, A., Bojanowski, P.: Dinov2: Learning robust visual features without supervision (2023)
- [12] Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: *Proceedings of the IEEE/CVF international conference on computer vision*. pp. 4195–4205 (2023)
- [13] Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: *International conference on machine learning* (2021)
- [14] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2022)
- [15] Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., Liu, Y.: Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing* **568**, 127063 (2024)
- [16] Xu, Y., Gu, T., Chen, W., Chen, C.: Ootdiffusion: Outfitting fusion based latent diffusion for controllable virtual try-on. *arXiv preprint arXiv:2403.01779* (2024)
- [17] Yang, Z., Zeng, A., Yuan, C., Li, Y.: Effective whole-body pose estimation with two-stages distillation (2023), <https://arxiv.org/abs/2307.15880>
- [18] Zhai, X., Mustafa, B., Kolesnikov, A., Beyer, L.: Sigmoid loss for language image pre-training (2023), <https://arxiv.org/abs/2303.15343>
- [19] Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 3836–3847 (2023)