

# Fashion-VDM: Video Diffusion Model for Virtual Try-On Supplementary Material

CCS Concepts: • **Computing methodologies** → **Computer graphics**; **Computer vision**.

Additional Key Words and Phrases: Virtual Try-On, Video Synthesis, Diffusion Models

## ACM Reference Format:

. 2024. Fashion-VDM: Video Diffusion Model for Virtual Try-On Supplementary Material. In *SIGGRAPH Asia 2024 Conference Papers (SA Conference Papers '24)*, December 3–6, 2024, Tokyo, Japan. ACM, New York, NY, USA, 5 pages. <https://doi.org/10.1145/3680528.3687623>

## 1 EXAMPLES OF FAILURE CASES

We show two examples of failure cases of our method in Figure 1. In row 1, we show artifacts that appear in the body/garment boundary, due to an imperfect person segmentation in the clothing-agnostic image. Imperfect segmentation is a common cause of such artifacts, and may also incorrectly leak regions from the original garment. In our human evaluation (Section 3.1), 10/17 videos that were failed had agnostic errors. In general, although our preprocessing methods are state-of-the-art, other types of preprocessing errors occur limit the quality of Fashion-VDM. In total, 70% of videos not chosen by human raters had errors in one or more inputs. As shown in row 2, body shape misrepresentation (e.g. slimming) occurs, because the clothing-agnostic images remove all body parts, besides hands, feet, and head, thus they do not include detailed information about body size.

### 1.1 Training and Inference Details

We train our model on 16 TPU-v4’s for approximately 2 weeks, including all training phases. Our image baseline model is trained for 1M iterations with a batch size of 8 and resolution  $512 \times 384$ px using the Adam optimizer with a linearly decaying learning rate of  $1e^{-4}$  to  $1e^{-5}$  over 1M steps and 10K warm-up steps. Each phase of progressive temporal training is initialized from the previous checkpoint and trained for 150K iterations, following the order of phases described in the Section 2. For all phases, we incorporate dropout for each conditional input independently 10% of the time. We train with an L2 loss on  $\epsilon$ .

During inference, we use the DDPM sampler [Ho et al. 2020] with 1000 refinement steps. Each video takes approximately 8 minutes to synthesize with split-CFG and 5 minutes without split-CFG.

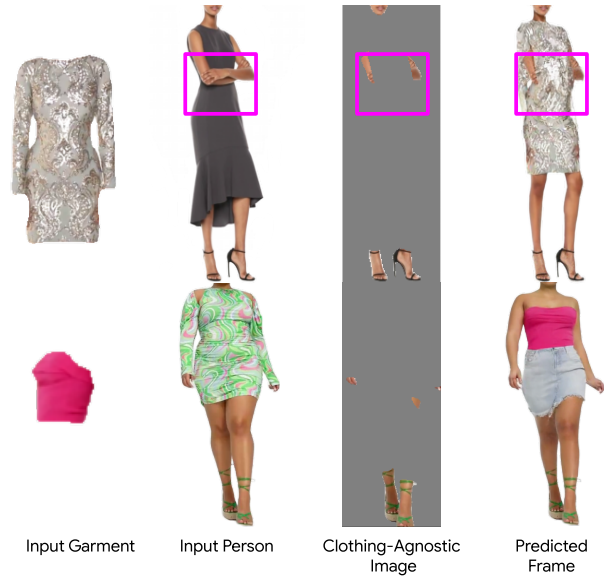


Fig. 1. **Failure Cases.** Errors in the person segmentation may lead to artifacts (top row). Fashion-VDM may incorrectly represent body shape (bottom row).

## 2 PROGRESSIVE TRAINING DETAILS

The overall progressive temporal training strategy is depicted in Figure 2. We first train a base image model from scratch on image data at 512px resolution and batch size 8 for 1M iterations. Then, we inflate the base architecture with temporal blocks and continue training the model using our joint image-video training strategy. In these temporal training phases, half of the batches are from the image dataset and the other half are batches of consecutive video frames from the video dataset. When training with an image batch, we skip the temporal blocks entirely in the forward and backward passes. At each successive phase of temporal training, we initialize the model from the previous phase’s checkpoint and double the training video length:  $8 \rightarrow 16 \rightarrow 32 \rightarrow 64$ . We train each temporal training phase for 150K iterations. Once the video length becomes prohibitively large in memory at 64-frames, we introduce temporal downsampling and upsampling layers to the model. At test time, our model generates  $512 \times 384$ px videos up to 64-frames in one inference pass with a single network.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

*SA Conference Papers '24, December 3–6, 2024, Tokyo, Japan*

© 2024 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-1131-2/24/12.

<https://doi.org/10.1145/3680528.3687623>

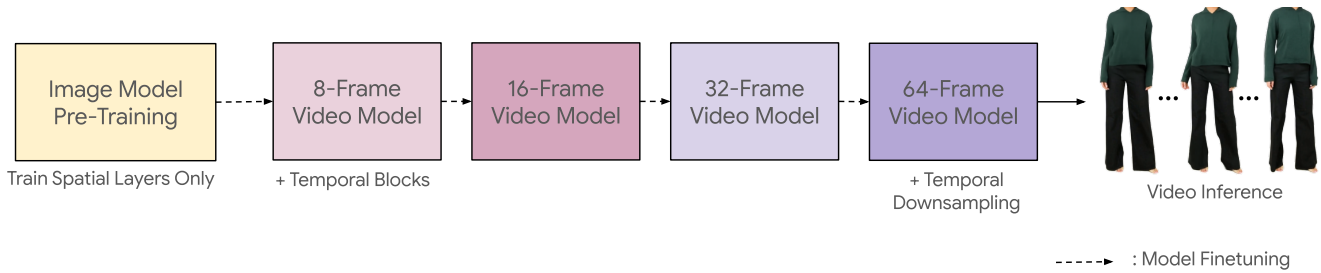


Fig. 2. **Progressive Training Strategy.** Fashion-VDM is trained in multiple phases of increasing frame length. We first pretrain an image model, by training only the spatial layers on our image dataset. In subsequent phases, we train temporal and spatial layers on increasingly long batches of consecutive frames from our video dataset.

$(w_\theta, w_p, w_g, w_c)$	UBC Test Dataset			Our Test Dataset		
	FID ↓	FVD ↓	CLIP ↑	FID ↓	FVD ↓	CLIP ↑
(1, 0, 1, 1)	136	1053	0.712	126	779	0.632
(1, 1, 1, 1)	95	644	0.748	<b>52</b>	<b>242</b>	0.667
(1, 1, 1, 3)	100	653	0.745	84	588	0.664
(1, 1, 3, 3)	99	454	0.756	55	262	0.662
(1, 3, 3, 3)	104	481	0.763	63	284	<b>0.671</b>
(1, 1, 3, 1)	<b>62</b>	<b>253</b>	<b>0.770</b>	76	385	0.661

Table 1. Quantitative Ablation of Split-CFG Weights. We compute FID, FVD, and CLIP scores of our full model using different split-CFG weights. Bolded values indicate best scores within the dataset.

### 3 SPLIT-CFG WEIGHTS ABLATIONS

We quantitatively evaluate our choice of split-CFG weights for both datasets on a held-out validation set. The results are shown in Table 1. Calibrating these weights correctly is not only beneficial to preserving garment fidelity, as shown by the FID score, but also increasing temporal consistency, as shown by the FVD score. Intuitively, by increasing the similarity of the output garment to the input garment, there is less allowed variability in the appearance of each frame, thus increased temporal smoothness. Based on these results, we employ weights (1, 1, 3, 1) for UBC and weights (1, 1, 1, 1) for our test dataset.

#### 3.1 User Study

In addition to qualitative and quantitative evaluations, we perform user studies for our state-of-the-art comparisons. The results are shown in Table 2. Our user studies are conducted by 5 human raters who are unfamiliar with the method. For each sample, the raters were asked to select which video performs best in each category: temporal smoothness, garment fidelity to the input garment image, and person fidelity to the input person video. The scores on both UBC test dataset and our test dataset reported are fraction of total votes divided by the total number of videos. Fashion-VDM exceeds other methods on all three user preference categories for both datasets.



Fig. 3. **Split-CFG Ablation with UBC-Only Model.** When Fashion-VDM is trained on the limited UBC dataset only, we observe overfitting to the largely plain garments in the UBC train dataset. However, we find that increasing garment image guidance ( $w_g$ ) in split-CFG significantly increases garment details.

#### 3.2 UBC-Only Model

We initialize this model from our pretrained image model, which is comparable to an open source image diffusion model, like Stable Diffusion [Rombach et al. 2021], which are trained on even larger image datasets, including LAION 5B [Schuhmann et al. 2022]. We then train progressively using both image data and UBC video data, following the same progressive training scheme as the full model.

The UBC-only model exceeds all baselines on the UBC test dataset quantitatively, but is qualitatively worse at preserving intricate garment details and patterns. This is expected, given the limited size and lack of diversity of UBC training dataset. However, we discovered that increasing the split-CFG garment weight significantly improves

	UBC Test Dataset			Our Test Dataset		
	Video Smoothness ↑	Person Fidelity ↑	Garment Fidelity ↑	Video Smoothness ↑	Person Fidelity ↑	Garment Fidelity ↑
TryOn Diffusion	0.01	0.00	0.00	0.03	0.01	0.00
Magic Animate	0.03	0.00	0.03	0.02	0.01	0.00
Animate Anyone	0.03	0.03	0.03	0.04	0.01	0.05
Ours (Full)	0.93	0.97	0.94	0.91	0.96	0.95

Table 2. User Study. Our study indicates that users overwhelmingly prefer Fashion-VDM to other baselines in terms of video smoothness, person fidelity, and garment fidelity on both test datasets.

lost garment details, even more so than with the full model. We qualitatively show this in Figure 4. This implies that when training with limited data, split-CFG becomes even more crucial to preserving the conditioning image details.

We provide qualitative examples generated by our model trained only on the UBC dataset [Zablotskaia et al. 2019] in Figure 4. While the results are still smooth and temporally consistent, the model struggles to maintain complex patterns and garment shape details. This is likely due to overfitting to the limited size and scope of the UBC training dataset, consisting of 500 videos of women in dresses.

## REFERENCES

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising Diffusion Probabilistic Models. arXiv:arXiv:2006.11239

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2021. High-Resolution Image Synthesis with Latent Diffusion Models. arXiv:arXiv:2112.10752

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. 2022. LAION-5B: An open large-scale dataset for training next generation image-text models. arXiv:arXiv:2210.08402

Polina Zablotskaia, Aliaksandr Siarohin, Bo Zhao, and Leonid Sigal. 2019. DwNet: Dense warp-based network for pose-guided human video generation. arXiv:arXiv:1910.09139



Fig. 4. **Qualitative Results for UBC-Only Model.** Our model trained only on UBC data generates temporally consistent, smooth try-on videos for plain and simple patterned garments, but struggles to preserve intricate patterns and complex garment shapes.



Fig. 5. **Additional Qualitative Results.** We showcase video try-on results generated by Fashion-VDM using swapped test videos from the UBC dataset [Zablot-skaia et al. 2019] and our own collected test dataset. Note that the input garment image and input person frames come from different videos.