FLOWVVTON: FLOW-GUIDED MASK-FREE VIDEO VIRTUAL TRY-ON

Anonymous authors
Paper under double-blind review

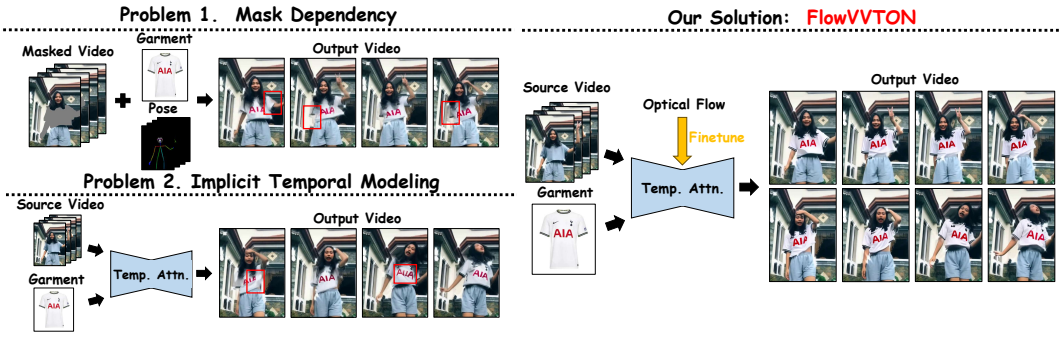

Figure 1: FlowVVTON performs mask-free video virtual try-on. Given only a source video and a target garment image, our method generates temporally consistent try-on results across all frames without any human parsing masks, pose keypoints, or region annotations. Temporal consistency is enforced during training via a flow-warped latent loss applied across at multiple decoder levels of the generation model.

ABSTRACT

Video virtual try-on aims to transfer a target garment onto a moving person across video frames. Existing methods often rely on human parsing masks or pose keypoints that can become unreliable under large motions and occlusions, causing boundary artifacts and temporal inconsistency. A further limitation is that most approaches rely solely on attention mechanisms for temporal modeling, providing no explicit motion supervision. To address both limitations, we propose FlowVVTON, a mask-free framework that eliminates parsing mask dependency while introducing explicit motion supervision. Our key innovation is a flow-warped latent loss that uses optical flow solely during training to align features across frames at multiple layers of the generation model, enforcing multi-scale temporal consistency. We incorporate this loss into a two-stage training strategy that first establishes mask-free spatial alignment and then introduces flow-guided temporal supervision. Together, these designs improve temporal consistency and reduce boundary artifacts, enabling robust garment transfer under large body motions. Experiments on TikTokDress show that FlowVVTON outperforms baselines by substantial margins, including a 5.7× reduction in unpaired VFID-R compared with SwiftTry, while requiring no segmentation masks, pose keypoints, or region annotations as model inputs.

1 INTRODUCTION

Video virtual try-on aims to realistically transfer a target garment onto a moving person across a video sequence. Unlike image-based try-on Ho et al. (2020); Rombach et al. (2022), which operates on static frames, video try-on must satisfy *temporal consistency*: the transferred garment must remain stable across frames without flickering, texture drift, or boundary artifacts, even as the person moves, rotates, or crosses their limbs. This capability has significant commercial value for online shopping, where dynamic garment visualization can reduce return rates, as well as for virtual content creation.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

The dominant paradigm in video try-on conditions generation on human parsing masks or pose keypoints obtained from off-the-shelf estimators such as OpenPose Cao et al. (2021) or DensePose Güler et al. (2018). These preprocessing tools segment the body into semantic regions (arms, torso, background) and provide spatial guidance for garment placement. However, they introduce a critical vulnerability: under fast motion, self-occlusion, or challenging lighting, mask and keypoint predictions become unreliable. Because mask extraction is the first stage of the pipeline, errors propagate and amplify through subsequent generation steps, manifesting as jagged garment edges, texture bleeding into the background, and temporal instability.

A second, related limitation is that most methods rely **solely** on attention mechanisms to maintain spatiotemporal consistency. They insert 3D convolutions or temporal attention layers into the denoising network and depend entirely on data-driven learning to capture cross-frame correspondence. While effective for slow, predictable motions, these attention layers provide no explicit motion supervision—the model must infer how pixels move between frames purely from statistical patterns in the training data. Under rapid movements or large body rotations, this implicit approach breaks down: without direct knowledge of pixel-level motion, textures drift, flicker, or deform.

Both challenges share a common root: the absence of explicit, pixel-level motion information. **Optical flow** offers a natural solution: a dense, continuous field that directly encodes the physical motion of every visible surface, unlike discrete masks that are brittle to estimation errors. Our central question is therefore: *can optical flow, used purely as a training-time supervision signal, enable robust mask-free video try-on?*

We answer this question affirmatively with **FlowVVTON**, a mask-free video try-on framework that eliminates parsing mask dependency while introducing explicit motion supervision. FlowVVTON takes only source video frames and a target garment image as input, no parsing masks, pose keypoints, or region annotations. Our key innovation is a **flow-warped latent loss**: during training, optical flow is used to align features across frames at multiple layers of the generation model. The loss minimizes the mean squared error between flow-warped features and their counterparts in another frame, complementing temporal attention with explicit, multi-scale motion supervision. Crucially, optical flow is used only as a training-time supervision signal; at inference, the model generates temporally consistent video without any flow computation. We incorporate this loss into a **two-stage training strategy**: first, we establish mask-free spatial alignment by fine-tuning on video frames. Then, we train temporal attention with flow-guided supervision to enforce cross-frame consistency. To support this training, we prepare synthetic paired data from TikTokDress using off-the-shelf video try-on models. Together, these designs improve temporal consistency and reduce boundary artifacts, enabling robust garment transfer under large body motions. Extensive experiments on TikTokDress demonstrate that FlowVVTON outperforms existing baselines in both reconstruction quality and temporal consistency, including a $5.7\times$ reduction in unpaired VFID-R compared with SwiftTry.

Our contributions are:

- **A mask-free video try-on framework for large-motion scenarios.** FlowVVTON combines mask-free generation with explicit motion supervision to reduce boundary artifacts and improve robustness to large body motions, using only source video frames and a target garment image as inputs.
- **A flow-warped latent loss for multi-scale temporal consistency.** The loss aligns features across frames at multiple layers of the generation model, complementing temporal attention with explicit motion supervision. Optical flow is used only during training, **requiring no flow computation at inference**.
- **A two-stage training strategy with flow-guided temporal supervision.** We first establish mask-free spatial alignment by fine-tuning on video frames, then train temporal attention with the flow-warped latent loss to enforce cross-frame consistency.

2 Related Work

2.1 Image Virtual Try-On

Image virtual try-on transfers a garment onto a person image. Early GAN-based methods Han et al. (2018); Wang et al. (2018); Han et al. (2019); Ge et al. (2021); Choi et al. (2021) used warp-and-

blend pipelines effective under controlled conditions. Diffusion models Ho et al. (2020); Rombach et al. (2022) now dominate, using iterative denoising with garment feature injection via textual inversion, cross-attention fusion, or lightweight conditioning Morelli et al. (2023); Kim et al. (2024); Xu et al. (2024a); Choi et al. (2024); Chong et al. (2025a); Zhou et al. (2025); Takemoto & Koshinaka (2026), with recent efforts targeting single-step generation Fang et al. (2024a); Sun et al. (2026). Mask-free image methods eliminate parsing dependency through pseudo-data training or unified architectures Zhang et al. (2025); Wan et al. (2025); Jo et al. (2025); Guo et al. (2025); Sun et al. (2025). We adopt a pretrained mask-free image model as our base.

2.2 Video Virtual Try-On

Video try-on adds temporal consistency: the garment must stay consistent across frames under motion. Early methods Dong et al. (2019); Jiang et al. (2022) processed frames independently with post-hoc smoothing but lacked joint spatiotemporal modeling. Video diffusion models enabled end-to-end temporal generation, from SVD-based approaches with hierarchical attention and local refinement Fang et al. (2024b); Xu et al. (2024b); He et al. (2024); Li et al. (2025a;c); Nguyen et al. (2025) to recent DiT-based architectures Peebles & Xie (2023) with spatiotemporal encoding, stage-wise training, and keyframe-guided injection Zheng et al. (2024); Chong et al. (2025b); Li et al. (2025b); Zuo et al. (2025); He et al. (2026); Zeng et al. (2025); Zheng et al. (2026). All of these, however, rely on human parsing masks or pose keypoints Cao et al. (2021); Güler et al. (2018).

2.3 Mask-Free Try-On

Mask-free methods avoid error propagation from imperfect upstream parsers. Image-domain works use pseudo-data training, teacher-student distillation, or unified architectures Zhang et al. (2025); Wan et al. (2025); Jo et al. (2025); Guo et al. (2025); Sun et al. (2025); Wang et al. (2025). Extending mask-free try-on to video is harder. The latest mask-free video methods—PEMF-VTO Chang et al. (2025), TripVVT Shao et al. (2026), and VanAST Cha et al. (2026)—replace dense masks with sparse keypoints, coarse bounding boxes, or pose-driven animation, but still require spatial annotations or mask-based teachers during training, and their code has not been open-sourced at the time of writing. FlowVVTON eliminates masks entirely, using only raw video frames and a garment image as input, with flow-warped latent loss for temporal consistency.

2.4 Optical Flow for Video Generation

Optical flow has been used to improve video generation through flow-based losses Wu et al. (2025); Bhowmik et al. (2025), as a conditioning signal via latent warping or noise correlation Wang et al. (2026); Nam et al. (2025); Liao et al. (2025); Liang et al. (2024); Burgert et al. (2025), or injected into model internals through frozen attention or warped embeddings Guo et al. (2024); Xue et al. (2025); Koroglu et al. (2025); Gökmen et al. (2025). In FlowVVTON, a per-layer flow-warped latent loss aligns adjacent-frame UNet features under motion constraints, providing explicit geometric supervision for temporal consistency.

3 Method

Our framework is built on a Latent Diffusion Model (LDM) Rombach et al. (2022) backbone. All operations are performed in the latent space of a pretrained VAE, which compresses input video frames $V \in \mathbf{R}^{F\times H\times W\times 3}$ and the garment image I_g into compact latent codes z_v and z_g.

3.1 Model Architecture

Overview. The architecture consists of two UNets with shared base weights: a **reference UNet** for the garment image I_g, and a **main UNet** for video generation. The reference UNet extracts multi-scale garment features and injects them into the main UNet via mutual self-attention (frame features as queries, garment features as keys/values), enabling mask-free garment transfer without explicit warping.

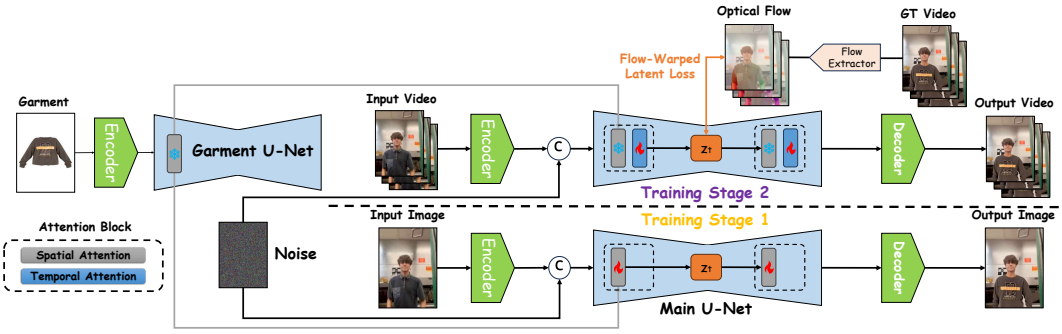

Figure 2: Overview of FlowVVTON. A reference UNet extracts garment features and injects them into the main UNet via mutual self-attention. The main UNet with temporal attention generates video frames from concatenated noisy and source latents. At every layer of the main UNet, a flow-warped latent loss enforces temporal consistency by aligning features across adjacent frames under optical flow guidance.

The main UNet is a 2D UNet inflated to video by inserting temporal attention layers after each spatial block, containing spatial self-attention (with garment feature injection) and temporal self-attention across F frames. Only temporal attention layers are trained; all spatial weights and the reference UNet remain frozen.

Mask-free input. At each denoising step t, the main UNet receives the concatenation of the noisy latent z_t and the VAE-encoded source video z_{src}:

$$\text{Input}_{\text{main}} = \text{Concat}(z_t, z_{\text{src}}). \tag{1}$$

z_{src} encodes the full original video without any masking. The model learns to distinguish garment areas from background purely through feature-level contrast, eliminating the error-prone mask extraction step.

Flow-warped latent loss. Optical flow serves two roles during training—a spatial alignment signal and a temporal regularizer—and is not required at inference. The loss operates within the main UNet on *clean features*: a clean forward pass (noise-free latents at timestep 0) yields multi-scale features $\mathbf{z}_l(t)$ for each frame t and layer l. Since optical flow is unreliable at occlusion boundaries, we use forward-backward consistency checking: the forward-backward error $\text{FBE}(p) = \|p - p''\|_2$ is converted into a confidence map $\mathbf{C}(p) = \exp(-\text{FBE}(p)/\tau)$ with $\tau = 3.0$, which down-weights unreliable pixels.

Given frames x_t and x_{t+1}, we compute backward flow $F_{t+1\to t}$ via RAFT Teed & Deng (2020) and downsample it to each UNet layer's resolution—coarse layers capture large body movements, fine layers track subtle texture displacements. The per-layer loss warps frame $t{+}1$ backward and measures MSE consistency with frame t:

$$\mathcal{L}_l = \frac{\sum_p \mathbf{C}_l(p) \cdot \left\|\mathbf{z}_l(t,p) - \mathcal{W}\big(\mathbf{z}_l(t{+}1),\, F_l\big)(p)\right\|_2^2}{\sum_p \mathbf{C}_l(p) + \varepsilon}, \tag{2}$$

where $\mathcal{W}$ is the bilinear warping operator defined as $\mathcal{W}(F, f)(p) = F(p + f(p))$, implemented via differentiable grid sampling: F is sampled at warped coordinates $p + f(p)$ using bilinear interpolation, F_l is the flow downsampled to layer l, and $\mathbf{C}_l(p)$ is the confidence map. The per-layer losses are aggregated with level weights w_l (2.0 for highest-resolution, 1.5 mid, 1.0 coarse) and per-layer variance normalization to prevent domination by high-activation layers.

For long-range stability, we compute the loss at multiple strides $s \in \{1, 2, 4\}$:

$$\mathcal{L}_{\text{flow}} = \sum_{s\in\{1,2,4\}} \alpha_s \cdot \sum_l w_l\, \mathcal{L}_l^{(s)}, \tag{3}$$

with $\alpha_1{=}1.0$, $\alpha_2{=}0.5$, $\alpha_4{=}0.25$. Flow $F_{t+s\to t}$ is computed directly on sub-sampled frames, avoiding compounding interpolation errors. By design, the loss only involves the main UNet; the reference UNet is frozen.

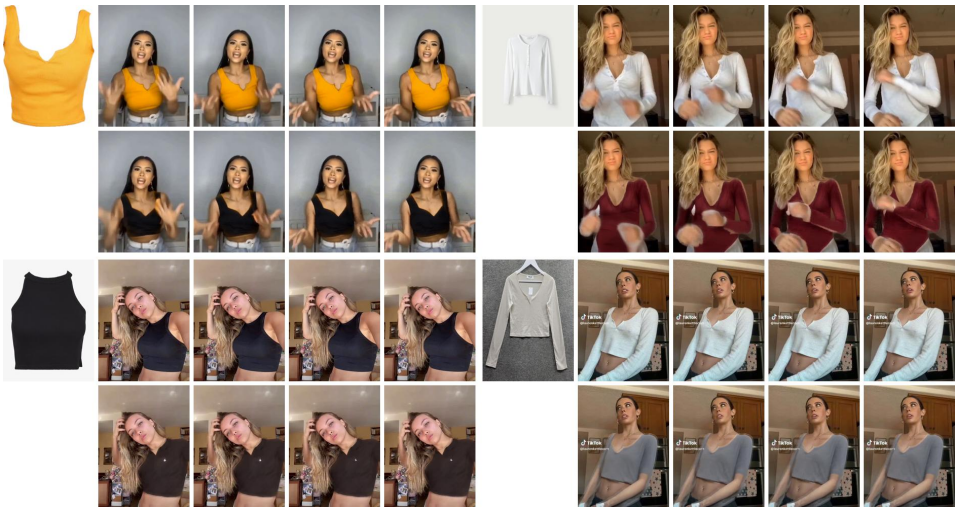

Figure 3: Examples from the synthesized training dataset. Each group shows a garment image (left), synthesized source-side frames generated by the external synthesis model (top row), and original ground-truth (GT) frames (bottom row).

3.2 Data Preparation

Paired video try-on data is impractical to collect at scale. We prepare training data from TikTokDress (693 training, 124 test videos): an off-the-shelf video try-on model generates garment-transfer results for randomly sampled garments. Synthesized videos serve as training inputs; original videos serve as ground truth. Optical flow is computed on clean GT videos, providing more reliable motion supervision. Figure 3 shows examples of the synthesized training data.

3.3 Training Strategy

We adopt a two-stage training pipeline.

Stage 1: Image-level spatial alignment. We fine-tune a mask-free image try-on model on frames sampled from TikTokDress for 50K steps, establishing per-frame garment transfer without masks. Temporal attention layers remain frozen.

Stage 2: Video-level temporal training with flow guidance. We activate temporal attention layers, freeze spatial weights from Stage 1, and introduce $\mathcal{L}_{\text{flow}}$ across all main UNet layers. Temporal attention learns cross-frame patterns while the flow loss enforces geometric consistency—a combination key to stable mask-free generation.

Training objective. The total loss combines standard diffusion denoising with the flow-warped latent loss:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{ddpm}} + \lambda \mathcal{L}_{\text{flow}}, \tag{4}$$

where $\mathcal{L}_{\text{ddpm}}$ is the noise prediction MSE with Min-SNR weighting Hang et al. (2023) ($\gamma = 5.0$) and λ controls the flow loss weight. Zero-terminal SNR Lin et al. (2024) is enabled for better noise scheduling at inference.



Figure 4: Stage 1 vs. Stage 2 results. Top row: GT frames. Middle row: Stage 1 outputs after image-level fine-tuning lack cross-frame consistency. Bottom row: after Stage 2 flow-guided temporal training, garment textures remain stable across frames with temporal artifacts removed.

Table 1: Inference efficiency on a 59-frame video (384×512), single NVIDIA A6000.

Method	Time (s/frame)	GPU Memory (GB)
SwiftTry	2.31	7.6
CatV2TON	2.44	21.0
MagicTryOn	13.8	3.8 (offload)
FlowVVTON (Ours)	**1.60**	7.6

4 EXPERIMENTS

4.1 DATASETS AND METRICS

Datasets. We use the training data described in the Data Preparation section and evaluate under paired (same garment as source) and unpaired (different garment) settings on the TikTokDress test set.

Metrics. For paired evaluation we report SSIM (structural similarity), LPIPS (perceptual distance), and VFID (Video Fréchet Inception Distance) with I3D and ResNeXt backbones. VFID_{I3D} is sensitive to spatiotemporal dynamics, while VFID_{RN} captures spatial texture quality. For unpaired evaluation we report VFID_{I3D} and VFID_{RN} as distribution-level measures, plus CLIP-Score, which measures the semantic alignment between generated frames and the target garment embedding; higher is better. ↑ = higher is better, ↓ = lower is better.

4.2 IMPLEMENTATION DETAILS

Architecture. We use the pretrained VAE (ft-mse) from Stable Diffusion v1.5, frozen. The reference UNet is a frozen 2D UNet from a pretrained mask-free image try-on model. The main UNet is 3D-inflated with temporal attention layers after each spatial block; only these are trained.

Training. Stage 1 fine-tunes an image try-on model on TikTokDress frames for 50K steps. Stage 2 trains temporal attention layers on our synthetic dataset for 100K steps, with AdamW (8-bit, lr=1×10^{-5}, constant schedule), batch size 1, 16-frame sequences at stride 4, resolution 384×512, fp16 mixed precision, gradient clipping at 1.0, and gradient checkpointing.

Flow computation. RAFT Teed & Deng (2020) with 32 update iterations computes flow on the fly. Forward-backward consistency filters unreliable regions ($\tau = 3.0$). Flow loss weight $\lambda = 1.0$, level weights: down_0/up_3: 2.0; down_1/up_2: 1.5; down_2/up_1: 1.0. Multi-scale strides $\{1, 2, 4\}$ with weights $\{1.0, 0.5, 0.25\}$.

Inference. DDIM with 25 steps, CFG scale 3.5, context windows of 24 frames with 4-frame overlap, on a single NVIDIA A6000 GPU.

Inference efficiency. Table 1 compares per-frame inference time and GPU memory on a 59-frame test video (384×512) with a single NVIDIA A6000 GPU. FlowVVTON achieves 1.60 s/frame, faster than SwiftTry (2.31 s/frame), because no optical flow computation is required at inference and the model reuses ShiftCaching-style window inference. CatV2TON requires 21 GB memory due to its full DiT backbone with 24-frame windows. MagicTryOn relies on CPU offloading to fit a 14B-parameter DiT within 3.8 GB, which drastically increases per-frame latency to 13.8 s. FlowVVTON and SwiftTry both use UNet backbones and stay within 7.6 GB, making them practical for consumer GPUs.

4.3 QUALITATIVE COMPARISON

Figure 5 compares FlowVVTON with SwiftTry. FlowVVTON produces sharper garment textures with fewer boundary artifacts, particularly under large motions where SwiftTry's mask-based pipeline produces misaligned edges. Figure 6 shows additional results across diverse garments and motions.



Figure 5: Qualitative comparison with SwiftTry. FlowVVTON produces fewer boundary artifacts and sharper garment textures, particularly under large motions where mask-based methods degrade.



Figure 6: Qualitative results of FlowVVTON across diverse garments and motion patterns. Our method preserves garment textures and maintains temporal stability under large body motions, without mask-induced boundary artifacts.

4.4 Quantitative Comparison

We evaluate under two settings: paired (same garment, same pose) and unpaired (different garment). The baselines cover both UNet-based (SwiftTry, ours) and DiT-based (CatV2TON, MagicTryOn) architectures.

Table 2 reports quantitative results. FlowVVTON achieves the best results across all metrics in both settings, outperforming SwiftTry by substantial margins. The 6× reduction in paired VFID-R (0.71 vs. 4.30) and 5.7× reduction in unpaired VFID-R (0.60 vs. 3.42) confirm that the flow-warped latent loss provides fundamentally better temporal supervision than implicit temporal attention. The unpaired CLIP-Score further shows that FlowVVTON achieves the best semantic alignment between generated frames and target garments. Notably, CatV2TON and MagicTryOn perform worse than SwiftTry, suggesting that DiT-based architectures do not compensate for the absence of explicit temporal constraints.

Evaluation on ViViD. To assess generalization, we additionally evaluate on the ViViD-Test benchmark. In the paired setting, MagicTryOn leads on every metric (SSIM 0.936, LPIPS 0.033, VFID-I 5.382, VFID-R 0.234), which we attribute to its strong image-level foundation and garment-preserving strategy producing outputs close to the ground-truth appearance. FlowVVTON ranks second on both paired VFID metrics; in particular, its paired VFID-R (1.725) is 45% lower than SwiftTry's (3.145), indicating that flow-based supervision improves temporal quality even where per-frame fidelity, rather than temporal modeling, is the primary challenge.

In the unpaired setting, CatV2TON attains the highest CLIP-Score (0.457), followed closely by FlowVVTON (0.442). Notably, MagicTryOn's advantage does not transfer to garment fidelity: it

Table 2: Quantitative evaluation on TikTokDress test set. U- prefix denotes unpaired evaluation. ↑/↓ = higher/lower is better. **Bold** = best, underline = second best.

Method	SSIM↑	LPIPS↓	VFID-I↓	VFID-R↓	U-VFID-I↓	U-VFID-R↓	U-CLIP↑
SwiftTry	0.877	0.090	31.31	4.30	32.06	3.42	0.297
CatV2TON	0.673	0.264	50.13	7.30	50.33	7.75	0.266
MagicTryOn	0.765	0.144	37.05	4.62	34.46	4.02	0.288
FlowVVTON (Ours)	**0.905**	**0.072**	**29.94**	**0.71**	**30.99**	**0.60**	**0.324**

achieves the lowest CLIP-Score (0.392) despite the best VFID values, suggesting its outputs follow the source-video distribution rather than the target garment. This reveals a fidelity–realism trade-off: methods biased toward reconstructing the original video attain better VFID at the cost of garment alignment, whereas FlowVVTON balances the two, ranking second on CLIP-Score and U-VFID-I. Overall, FlowVVTON outperforms SwiftTry—the closest mask-based UNet baseline—on every metric in both settings, further validating the effectiveness of flow-based supervision beyond TikTokDress. The relative ordering also shifts compared with TikTokDress: MagicTryOn's image-level fidelity dominates on ViViD, while flow-guided supervision remains decisive under the large motions of TikTokDress, consistent with our claim that explicit motion supervision matters most when temporal demands are high.

Table 3: Evaluation on ViViD-Test. U- prefix denotes unpaired evaluation; VFID-I/VFID-R use I3D/ResNeXt backbones. ↑/↓ = higher/lower is better. **Bold** = best, underline = second best.

Method	SSIM↑	LPIPS↓	VFID-I↓	VFID-R↓	U-VFID-I↓	U-VFID-R↓	U-CLIP↑
SwiftTry	0.838	0.097	23.341	3.145	26.11	3.53	0.438
CatV2TON	0.806	0.107	24.139	4.100	31.57	2.26	**0.457**
MagicTryOn	**0.936**	**0.033**	**5.382**	**0.234**	**5.34**	**0.24**	0.392
FlowVVTON (Ours)	0.839	0.093	21.976	1.725	26.00	2.68	0.442

4.5 Ablation Study

Table 4 analyzes each component's contribution.

Loss components. Removing the flow-warped latent loss (λ=0) causes the most severe degradation: VFID-I rises from 30.99 to 32.48 and VFID-R from 0.60 to 1.24, confirming that denoising alone cannot enforce temporal consistency. Removing confidence weighting degrades VFID-R to 0.67.

Multi-scale strides. Stride-1 only achieves the best VFID-I (30.95) but worse VFID-R (0.72). Strides $\{1, 2, 4\}$ achieve VFID-R of 0.60—a 17% improvement—by combining short-range smoothness, mid-range alignment, and long-range stability. Stride-8 slightly degrades performance as flow estimates become less reliable.

Table 4: Ablation study (unpaired evaluation). ↓ = lower is better. **Bold** = best, underline = second best.

Configuration	VFID-I	VFID-R
Loss components		
w/o $\mathcal{L}_{\text{flow}}$	32.48	1.24
w/o confidence	31.99	0.67
Multi-scale strides		
stride 1	**30.95**	0.72
strides $\{1, 2\}$	31.19	0.68
strides $\{1, 2, 4\}$	30.99	**0.60**
strides $\{1, 2, 4, 8\}$	31.03	0.67

Flow quality. Reducing RAFT iterations from 32 to 16 degrades VFID-R from 0.60 to 0.61, while 48 iterations yields only marginal improvement (VFID-I 30.92) with higher cost. We select 32 iterations.

4.6 User Study

We conducted a blind user study comparing our method against SwiftTry, CatV2TON, and MagicTryOn. Participants rated randomly sampled video pairs on a scale of 1–10 with model identity hidden and randomized order. 13 participants provided approximately 300 ratings per method. We apply per-user z-score normalization to account for individual rating biases. The normalized scores (Table 5) reveal a clear ranking: ours $\gg$ MagicTryOn $>$ SwiftTry $\approx$ CatV2TON. The 1.04-point gap between our method and the best baseline exceeds the 0.16-point range among the three baselines, indicating a pronounced and consistent user preference. Notably, even the best method achieves a raw mean of only 4.39 out of 10, indicating TikTokDress remains a challenging dataset with substantial room for improvement.

Table 5: User study results (1–10 scale). Raw = raw mean, Std = standard deviation, Norm = per-user z-score normalized mean.

Model	Raw	Std	Norm
SwiftTry	2.45	1.20	−0.37
CatV2TON	2.36	1.05	−0.40
MagicTryOn	2.60	1.67	−0.24
Ours	**4.39**	2.15	**+0.80**

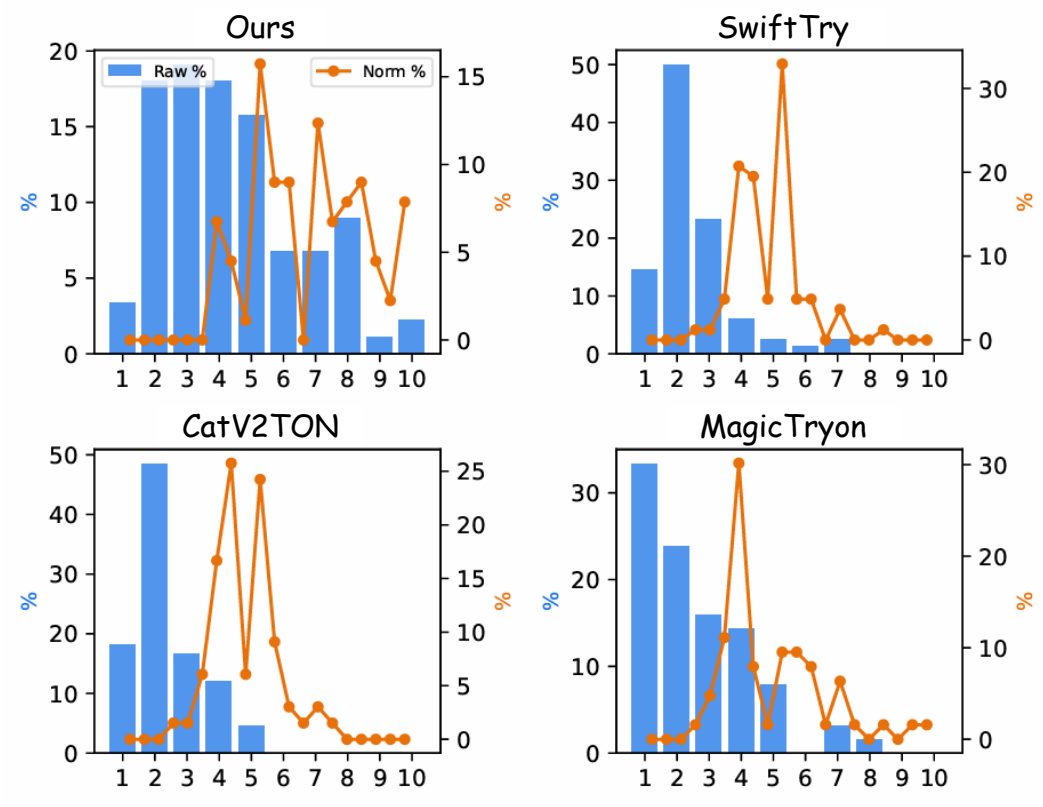

Figure 7: Per-model rating distributions. Blue bars: raw rating histogram (percentage). Orange curve: per-user z-score normalized ratings, rescaled onto the 1–10 axis for visual alignment (actual z-score values in Table 5).

5 Conclusion

We presented FlowVVTON, a mask-free video virtual try-on framework. By using optical flow solely as a training-time loss via a per-layer flow-warped latent loss, FlowVVTON achieves robust temporal consistency and reduces boundary artifacts under large motions. Limitations include dependency on flow quality during training, UNet-specific architecture, and current resolution constraints, and occasional sleeve-length transfer failures: when transferring a short-sleeved garment, the result may still exhibit short sleeves even where the target garment is long-sleeved.

AI USE STATEMENT

In this work, we used generative AI tools to assist with polishing the language of the manuscript and organizing the formatting. We have not used generative AI tools for the methodology design, the implementation of experiments, or the generation of experimental results. All AI-assisted content was reviewed and verified by the authors. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

Video try-on enables positive applications such as reducing e-commerce return rates through dynamic garment visualization. Realistic video synthesis can, however, be misused to fabricate content; our method focuses on garment transfer rather than identity manipulation, and we encourage the development of detection methods for synthetic media. The user study in this work involved voluntary participants who rated anonymized video pairs; no personal or identifiable information was collected. All datasets used in this work (TikTokDress, ViViD) are publicly available benchmarks.

REPRODUCIBILITY STATEMENT

We report complete implementation details, including the architecture, training hyperparameters, and the configuration of the flow-warped latent loss, in the Implementation Details section and the Training Details appendix. Baseline training protocols and evaluation protocols are documented in the Baseline Training Details and Evaluation Details appendices. All metrics are computed with publicly available implementations, and our code and model checkpoints will be released upon acceptance.

AUTHOR CONTRIBUTIONS

All authors contributed equally to the research idea, method design, experiments, and writing of this paper.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their constructive feedback.

REFERENCES

Aritra Bhowmik, Denis Korzhenkov, Cees GM Snoek, Amirhossein Habibian, and Mohsen Ghafoorian. Moalign: Motion-centric representation alignment for video diffusion models. *arXiv preprint arXiv:2510.19022*, 2025.

Ryan Burgert, Yuancheng Xu, Wenqi Xian, Oliver Pilarski, Pascal Clausen, Mingming He, Li Ma, Yitong Deng, Lingxiao Li, Mohsen Mousavi, et al. Go-with-the-flow: Motion-controllable video diffusion models using real-time warped noise. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13–23. IEEE, 2025.

Zhe Cao, Gines Hidalgo Martinez, Tomas Simon, Shih-En Wei, and Yaser A Sheikh. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(1), 2021.

Hyunsoo Cha, Wonjung Woo, Byungjun Kim, and Hanbyul Joo. Vanast: Virtual try-on with human image animation via synthetic triplet supervision. In *CVPR*, 2026.

Tianyu Chang, Xiaohao Chen, Zhichao Wei, Xuanpu Zhang, Qingguo Chen, Weihua Luo, Peipei Song, and Xun Yang. Pemf-vto: Point-enhanced video virtual try-on via mask-free paradigm. *IEEE Transactions on Consumer Electronics*, 2025.

Seunghwan Choi, Sunghyun Park, Minsoo Lee, and Jaegul Choo. Viton-hd: High-resolution virtual try-on via misalignment-aware normalization. In *CVPR*, 2021.

Yisol Choi, Sangkyung Kwak, Kyungmin Lee, Hyungwon Choi, and Jinwoo Shin. Improving diffusion models for authentic virtual try-on in the wild. *arXiv preprint arXiv:2403.05139*, 2024.

Zheng Chong, Xiao Dong, Haoxiang Li, Wenqing Zhang, Hanqing Zhao, Dongmei Jiang, Xiaodan Liang, et al. Catvton: Concatenation is all you need for virtual try-on with diffusion models. In *International Conference on Learning Representations*, volume 2025, pp. 66586–66601, 2025a.

Zheng Chong, Wenqing Zhang, Shiyue Zhang, Jun Zheng, Xiao Dong, Haoxiang Li, Yiling Wu, Dongmei Jiang, and Xiaodan Liang. Catv2ton: Taming diffusion transformers for vision-based virtual try-on with temporal concatenation. *arXiv preprint arXiv:2501.11325*, 2025b.

Haoye Dong, Xiaodan Liang, Xiaohui Shen, Bowen Wu, Bing-Cheng Chen, and Jian Yin. Fw-gan: Flow-navigated warping gan for video virtual try-on. In *ICCV*, 2019.

Naiyu Fang, Lemiao Qiu, Shuyou Zhang, Zili Wang, and Kerui Hu. Pg-vton: A novel image-based virtual try-on method via progressive inference paradigm. *IEEE Transactions on Multimedia*, 26, 2024a.

Zixun Fang, Wei Zhai, Aimin Su, Hongliang Song, Kai Zhu, Mao Wang, Yu Chen, Zhiheng Liu, Yang Cao, and Zheng-Jun Zha. Vivid: Video virtual try-on using diffusion models. *arXiv preprint arXiv:2405.11794*, 2024b.

Yuying Ge, Yibing Song, Ruimao Zhang, Chongjian Ge, Wei Liu, and Ping Luo. Parser-free virtual try-on via distilling appearance flows. *arXiv preprint arXiv:2103.04559*, 2021.

Ahmet Berke Gökmen, Yiğit Ekin, Bahri Batuhan Bilecen, and Aysegul Dundar. Ropecraft: Training-free motion transfer with trajectory-guided rope optimization on diffusion transformers. In *NeurIPS*, 2025.

Rıza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *CVPR*, 2018.

Hailong Guo, Bohan Zeng, Yiren Song, Wentao Zhang, Jiaming Liu, and Chuang Zhang. Any2anytryon: Leveraging adaptive position embeddings for versatile virtual clothing tasks. In *ICCV*, 2025.

Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. In *ICLR*, 2024.

Xintong Han, Zuxuan Wu, Zhe Wu, Ruichi Yu, and Larry S Davis. Viton: An image-based virtual try-on network. In *CVPR*, 2018.

Xintong Han, Xiaojun Hu, Weilin Huang, and Matthew R Scott. Clothflow: A flow-based model for clothed person generation. In *ICCV*, 2019.

Tiankai Hang, Shuyang Gu, Chen Li, Jianmin Bao, Dong Chen, Han Hu, Xin Geng, and Baining Guo. Efficient diffusion training via min-snr weighting strategy. In *ICCV*, 2023.

Qingdong He, Xueqin Chen, Yanjie Pan, Peng Tang, Pengcheng Xu, Zhenye Gan, Chengjie Wang, Xiaobin Hu, Jiangning Zhang, and Yabiao Wang. The devil is in the details: Enhancing video virtual try-on via keyframe-driven details injection. In *CVPR*, 2026.

Zijian He, Peixin Chen, Guangrun Wang, Guanbin Li, Philip HS Torr, and Liang Lin. Wildvidfit: Video virtual try-on in the wild via image-based controlled diffusion models. In *ECCV*, 2024.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 33, 2020.

Jianbin Jiang, Tan Wang, He Yan, and Junhui Liu. Clothformer: Taming video virtual try-on in all module. In *CVPR*, 2022.

Youngjoo Jo, Minho Park, and Dong-oh Kang. Up-vton: A unified virtual try-on framework supporting mask, mask-free, and prompt-driven guidance. In *ICCVW*, 2025.

Jeongho Kim, Gyojung Gu, Minho Park, Sunghyun Park, and Jaegul Choo. Stable viton: Learning semantic correspondence with latent diffusion model for virtual try-on. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8176–8185. IEEE, 2024.

Mathis Koroglu, Hugo Caselles-Dupré, Guillaume Jeanneret, and Matthieu Cord. Onlyflow: Optical flow based motion conditioning for video diffusion models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 6216–6226. IEEE, 2025.

Dong Li, Wenqi Zhong, Wei Yu, Yingwei Pan, Dingwen Zhang, Ting Yao, Junwei Han, and Tao Mei. Pursuing temporal-consistent video virtual try-on via dynamic pose interaction. In *CVPR*, 2025a.

Guangyuan Li, Siming Zheng, Hao Zhang, Jinwei Chen, Junsheng Luan, Binkai Ou, Lei Zhao, Bo Li, and Peng-Tao Jiang. Magictryon: Harnessing diffusion transformer for garment-preserving video virtual try-on. *arXiv preprint arXiv:2505.21325*, 2025b.

Siqi Li, Zhengkai Jiang, Jiawei Zhou, Zhihong Liu, Xiaowei Chi, and Haoqian Wang. Realvvt: Towards photorealistic video virtual try-on via spatio-temporal consistency. *arXiv preprint arXiv:2501.08682*, 2025c.

Feng Liang, Bichen Wu, Jialiang Wang, Licheng Yu, Kunpeng Li, Yinan Zhao, Ishan Misra, Jia-Bin Huang, Peizhao Zhang, Peter Vajda, et al. Flowvid: Taming imperfect optical flows for consistent video-to-video synthesis. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8207–8216. IEEE, 2024.

Xinyao Liao, Xianfang Zeng, Liao Wang, Gang Yu, Guosheng Lin, and Chi Zhang. Motionagent: Fine-grained controllable video generation via motion field agent. In *ICCV*, 2025.

Shanchuan Lin, Bingchen Liu, Jiashi Li, and Xiao Yang. Common diffusion noise schedules and sample steps are flawed. In *WACV*, 2024.

Davide Morelli, Alberto Baldrati, Giuseppe Cartella, Marcella Cornia, Marco Bertini, and Rita Cucchiara. Ladi-vton: Latent diffusion textual-inversion enhanced virtual try-on. In *ACM Multimedia*, 2023.

Hyelin Nam, Jaemin Kim, Dohun Lee, and Jong Chul Ye. Optical-flow guided prompt optimization for coherent video generation. In *CVPR*, 2025.

Hung Nguyen, Quang Qui-Vinh Nguyen, Khoi Nguyen, and Rang Nguyen. Swifttry: Fast and consistent video virtual try-on with diffusion models. In *AAAI*, 2025.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, 2023.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

Dingbao Shao, Song Wu, Shenyi Wang, Ye Wang, Ziheng Tang, Fei Liu, Jiang Lin, Xinyu Chen, Qian Wang, Ying Tai, et al. Tripvvt: A large-scale triplet dataset and a coarse-mask baseline for in-the-wild video virtual try-on. *arXiv preprint arXiv:2604.27958*, 2026.

Xianbing Sun, Yan Hong, Jiahui Zhan, Jun Lan, Huijia Zhu, Weiqiang Wang, Liqing Zhang, and Jianfu Zhang. Ds-vton: An enhanced dual-scale coarse-to-fine framework for virtual try-on. *arXiv preprint arXiv:2506.00908*, 2025.

Xianbing Sun, Jiahui Zhan, Liqing Zhang, and Jianfu Zhang. Directtryon: One-step virtual try-on via straightened conditional transport. *arXiv preprint arXiv:2605.12939*, 2026.

Kosuke Takemoto and Takafumi Koshinaka. Sift-vton: Geometric correspondence supervision on cross-attention for virtual try-on. *arXiv preprint arXiv:2605.01296*, 2026.

Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *ECCV*, 2020.

Zhenchen Wan, Dongting Hu, Weilun Cheng, Tianxi Chen, Zhaoqing Wang, Feng Liu, Tongliang Liu, Mingming Gong, et al. Mf-viton: High-fidelity mask-free virtual try-on with minimal input. *arXiv preprint arXiv:2503.08650*, 2025.

Aowen Wang, Wei Li, Hao Luo, Mengxing Ao, Chenyu Zhu, Xinyang Li, and Fan Wang. Jcomvton: Jointly controllable multi-modal diffusion transformer for mask-free virtual try-on. *arXiv preprint arXiv:2508.17614*, 2025.

Bochao Wang, Huabin Zheng, Xiaodan Liang, Yimin Chen, Liang Lin, and Meng Yang. Toward characteristic-preserving image-based virtual try-on network. In *ECCV*, 2018.

Ge Wang, Songlin Fan, Hangxu Liu, Quanjian Song, Hewei Wang, and Jinfeng Xu. Consistent video editing as flow-driven image-to-video generation. In *CVPR*, 2026.

Kuanting Wu, Kei Ota, and Asako Kanezaki. Flowloss: Dynamic flow-conditioned loss strategy for video diffusion models. In *MVA*, 2025.

Yuhao Xu, Tao Gu, Weifeng Chen, and Chengcai Chen. Ootdiffusion: Outfitting fusion based latent diffusion for controllable virtual try-on. *arXiv preprint arXiv:2403.01779*, 2024a.

Zhengze Xu, Mengting Chen, Zhao Wang, Linyu Xing, Zhonghua Zhai, Nong Sang, Jinsong Lan, Shuai Xiao, and Changxin Gao. Tunnel try-on: Excavating spatial-temporal tunnels for high-quality virtual try-on in videos. In *ACM Multimedia*, 2024b.

Haotian Xue, Qi Chen, Zhonghao Wang, Xun Huang, Eli Shechtman, Jinrong Xie, and Yongxin Chen. Mogan: Improving motion quality in video diffusion via few-step motion adversarial post-training. *arXiv preprint arXiv:2511.21592*, 2025.

Jianhao Zeng, Yancheng Bai, Ruidong Chen, Xuanpu Zhang, Lei Sun, Dongyang Jin, Ryan Xu, Nannan Zhang, Dan Song, and Xiangxiang Chu. Eevee: Towards close-up high-resolution video-based virtual try-on. *arXiv preprint arXiv:2511.18957*, 2025.

Xuanpu Zhang, Dan Song, Pengxin Zhan, Tianyu Chang, Jianhao Zeng, Qingguo Chen, Weihua Luo, and An-An Liu. Boow-vton: Boosting in-the-wild virtual try-on via mask-free pseudo data training. In *CVPR*, 2025.

Jun Zheng, Fuwei Zhao, Youjiang Xu, Xin Dong, and Xiaodan Liang. Viton-dit: Learning in-the-wild video try-on from human dance videos via diffusion transformers. *arXiv preprint arXiv:2405.18326*, 2024.

Jun Zheng, Zhengze Xu, Mengting Chen, Jing Wang, Jinsong Lan, Xiaoyong Zhu, Kaifu Zhang, Bo Zheng, and Xiaodan Liang. itryon: Mastering interactive video virtual try-on with spatial-semantic guidance. *arXiv preprint arXiv:2605.21431*, 2026.

Zijian Zhou, Shikun Liu, Xiao Han, Haozhe Liu, Kam Woh Ng, Tian Xie, Yuren Cong, Hang Li, Mengmeng Xu, Juan-Manuel Pérez-Rúa, et al. Learning flow fields in attention for controllable person image generation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2491–2501. IEEE, 2025.

Tongchun Zuo, Zaiyu Huang, Shuliang Ning, Ente Lin, Chao Liang, Zerong Zheng, Jianwen Jiang, Yuan Zhang, Mingyuan Gao, and Xin Dong. Dreamvvt: Mastering realistic video virtual try-on in the wild via a stage-wise diffusion transformer framework. *arXiv preprint arXiv:2508.02807*, 2025.

A TRAINING DETAILS

Forward pass and garment injection. The reference UNet encodes the garment image in WRITE mode, storing self-attention features (K, V). The denoising UNet reads those features in READ mode, injecting garment appearance without explicit warping. The 8-channel input concatenates noisy target latents with source latents.

Algorithm 1 Garment injection forward pass

Input: noisy target latent z_t, source latent z_{src}, timestep t, garment latent z_g
$z_t \leftarrow \text{Concat}(z_t,\ z_{\text{src}})$ {8-channel input}
ref_t $\leftarrow$ zeros_like(t)
ref_unet$(z_g,$ ref_t$)$ {WRITE mode: store K,V}
reader.update(writer) {copy garment features}
$\epsilon_\theta \leftarrow$ denoising_unet$(z_t,\ t)$.sample {READ mode}
return ϵ_θ

We remove and reinitialize the mismatched `conv_in` weights before loading the remaining pre-trained weights with `strict=False`. Trainable parameters are kept in float32 for GradScaler safety.

Stage 1 loss computation. SNR-weighted MSE with v-prediction and zero terminal SNR. The min-SNR weight for v-prediction is $w_t = \min(\text{SNR}_t, \gamma)/(\text{SNR}_t + 1)$, where $\text{SNR}_t = \alpha_t^2/\sigma_t^2$. Unconditional forward passes are applied with a probability of 0.1 to enable CFG training. During Stage 1, only the denoising UNet mid and decoder blocks are optimized. The VAE encoder, reference UNet, denoising UNet encoder, and other frozen components remain unchanged.

Stage 2 flow-warped latent loss. During Stage 2, only newly introduced temporal attention layers are optimized; all spatial parameters learned in Stage 1 remain frozen. The flow-warped latent loss is applied to multi-scale feature maps of the trainable main UNet; specifically, features from decoder levels are used for consistency supervision. Optical flow is computed using RAFT during training and is not required during inference.

Algorithm 2 Flow-warped latent loss

Input: clean features $\{\mathbf{z}_l(t)\}_{l,t}$, multi-stride flow $\{F_{t+s\to t,\,l}\}_{t,s,l}$, confidence maps $\{\mathbf{C}_l\}_l$, level weights $\{w_l\}$, stride weights $\{\alpha_s\}$
$\mathcal{L}_{\text{flow}} \leftarrow 0$
for $s \in \{1, 2, 4\}$ **do**
  for l in decoder levels **do**
    $\mathcal{L}_l^{(s)} \leftarrow 0$
    for $t \leftarrow 0$ **to** $T - s - 1$ **do**
      $\mathbf{z}_{\text{warp}} \leftarrow \mathcal{W}\big(\mathbf{z}_l(t{+}s),\ F_{t+s\to t,\,l}\big)$
      $\mathcal{L}_l^{(s)}$ += $\frac{\sum_p \mathbf{C}_l(p)\cdot\|\mathbf{z}_l(t,p)-\mathbf{z}_{\text{warp}}(p)\|_2^2}{\sum_p \mathbf{C}_l(p)+\varepsilon}$
    end for
    $\mathcal{L}_l^{(s)}$ /= $(T - s)$
    $\mathcal{L}_{\text{flow}}$ += $\alpha_s \cdot w_l \cdot \mathcal{L}_l^{(s)}$
  end for
end for
return $\mathcal{L}_{\text{flow}}$

The total objective is $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{denoise}} + \lambda \cdot \mathcal{L}_{\text{flow}}$.

Input and annotation clarification. During both training and inference, FlowVVTON receives only source video frames and a garment image as input. Human parsing masks, pose keypoints, and region annotations are never provided to the model. Optical flow is used exclusively as a training-time auxiliary supervision signal and is discarded during inference. The off-the-shelf video try-on

model used for data preparation is an external synthesis tool; its outputs serve only as training targets and do not imply that masks are used by FlowVVTON. The external synthesis model is only used for offline training data preparation and is not involved in inference.

Ablation through configuration flags. The ablation study manipulates three config-level parameters. The flow-warped loss is toggled by setting `lfl_loss_weight` to 0, which skips the entire feature extraction and warping block. Confidence weighting is controlled by whether forward-backward consistency maps are loaded and multiplied into the per-pixel loss; setting `use_confidence=false` replaces $\mathbf{C}_l(p)$ with a constant 1.0. The multi-scale stride set is specified as `multi_lfl_strides` (e.g., `[1]` for stride-1 only, `[1,2,4]` for the full variant). Stride weights α_s are configured via `multi_lfl_stride_weights` and level weights w_l via `lfl_level_weights`.

B Baseline Training Details

SwiftTry. Trained with the official two-stage pipeline: Stage 1 on VITON-HD image pairs, Stage 2 on TikTokDress videos with human parsing masks and dense pose as spatial conditions. Temporal attention modules are trained with standard denoising loss; no flow-based supervision is used.

CatV2TON. A DiT-based architecture trained with temporal concatenation. Stage 1 pretrains on image try-on data; Stage 2 fine-tunes the full DiT backbone on TikTokDress videos with garment-aware spatiotemporal RoPE and human parsing masks as input conditions. Training follows the official configuration and released weights.

MagicTryOn. Also DiT-based, trained with a garment-preserving strategy that decouples garment appearance cues and injects them via decomposition priors. Stage 1 establishes image-level try-on capability; Stage 2 trains on TikTokDress with human parsing masks. We directly evaluate publicly released checkpoints without additional fine-tuning.

All baselines are evaluated using their publicly released checkpoints at the same resolution (384×512) and on the same TikTokDress test set. Since these methods require their original conditioning inputs (masks, pose), we provide the corresponding conditions only for baseline evaluation.

C Evaluation Details

SSIM uses `torchmetrics` with `data_range=1.0`, input in $[0, 1]$. LPIPS uses `net_type=squeeze`, input in $[-1, 1]$. VFID uses I3D and 3D ResNeXt-101 feature extractors with Fréchet distance. All metrics are computed on full frames, with GT resized to each prediction's native resolution. GT videos are 540×720; predictions vary. GT is resized via bilinear interpolation to preserve full-frame content. When $T_{\text{gt}} \neq T_{\text{pred}}$, both are truncated to $\min(T_{\text{gt}}, T_{\text{pred}})$.

D Additional Qualitative Results

Additional qualitative results on TikTokDress and ViViD, demonstrating mask-free try-on performance across diverse garments, motions, and real-world scenarios.

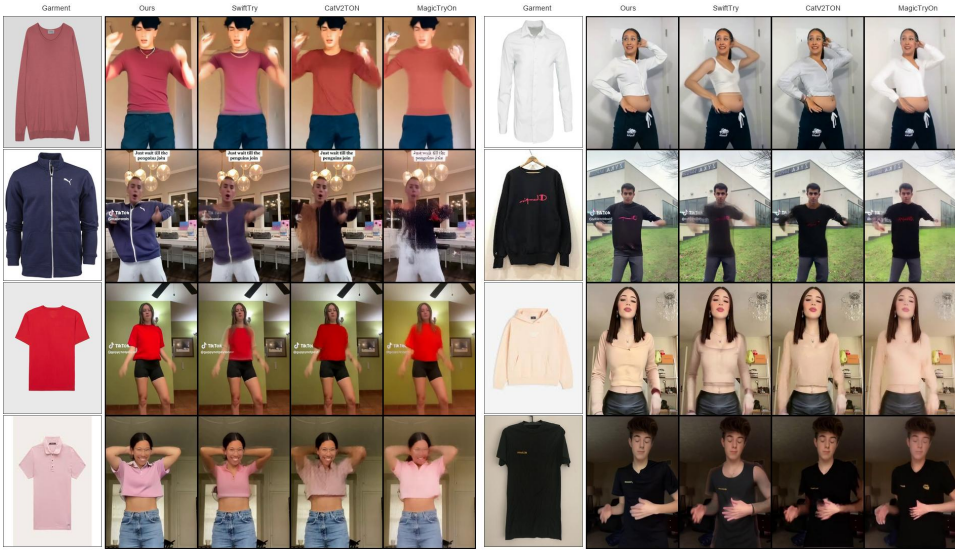

Figure 8: Cross-method comparison on TikTokDress. Each row shows garment image followed by FlowVVTON (ours), SwiftTry, CatV2TON, and MagicTryOn at frame 10.

Algorithm 3 Chunked metric computation

Input: GT video $V_g \in \mathbb{R}^{T_g \times C \times H_g \times W_g}$, prediction $V_p \in \mathbb{R}^{T_p \times C \times H_p \times W_p}$, chunk_size
$T \leftarrow \min(T_g, T_p)$
$V_g \leftarrow \text{Interpolate}(V_g,\ \text{size} = (H_p, W_p))$
$V_g \leftarrow V_g[:T],\ V_p \leftarrow V_p[:T]$
$s \leftarrow 0$
while $s < T$ **do**
  $e \leftarrow \min(s + \text{chunk_size},\ T)$
  $g \leftarrow V_g[s:e],\ p \leftarrow V_p[s:e]$
  if LPIPS **then**
    $g \leftarrow (g \times 2) - 1,\ p \leftarrow (p \times 2) - 1$
  end if
  metric.update(p, g)
  $s \leftarrow s + \text{chunk_size}$
end while
return metric.compute()

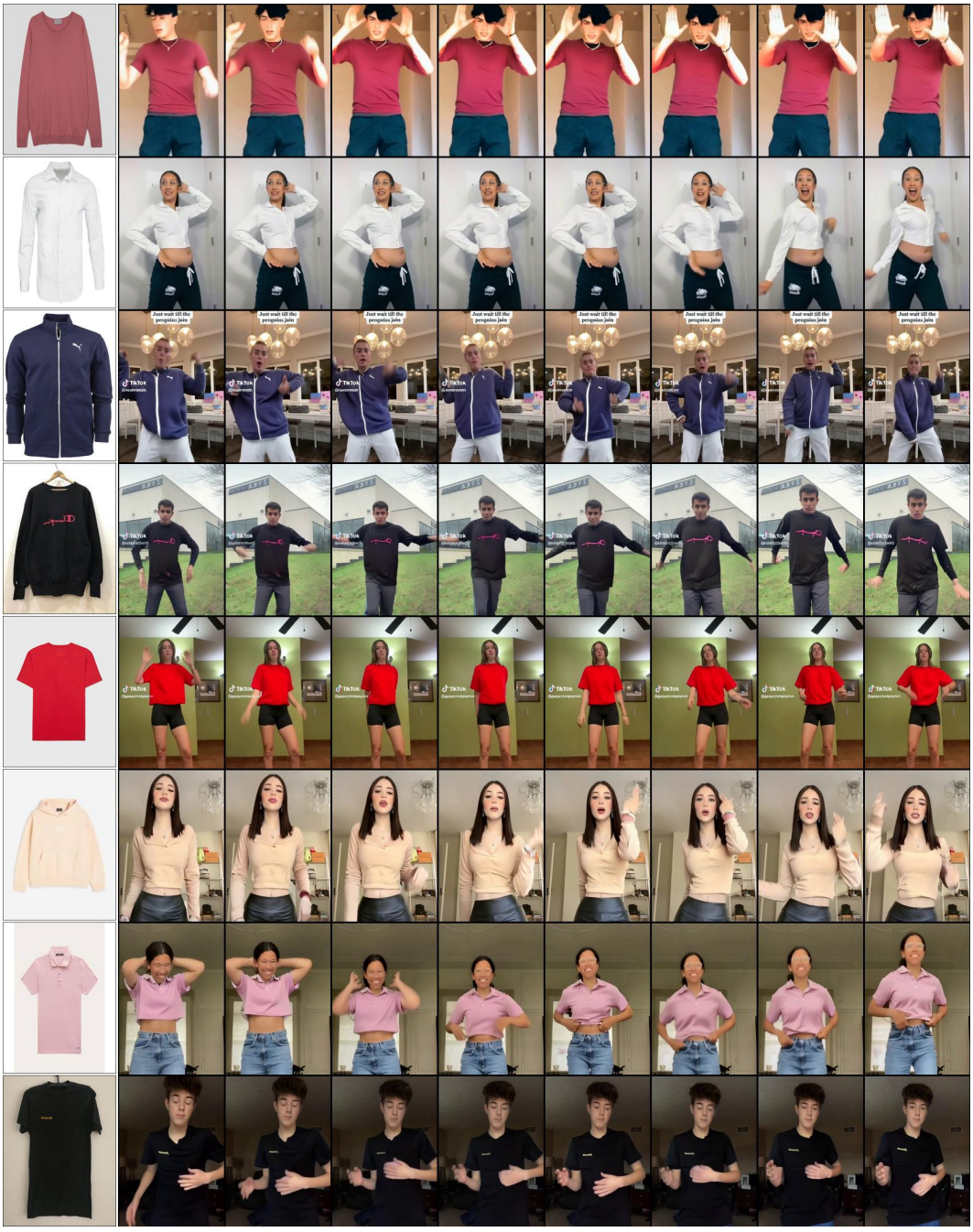

Figure 9: Qualitative results on TikTokDress.

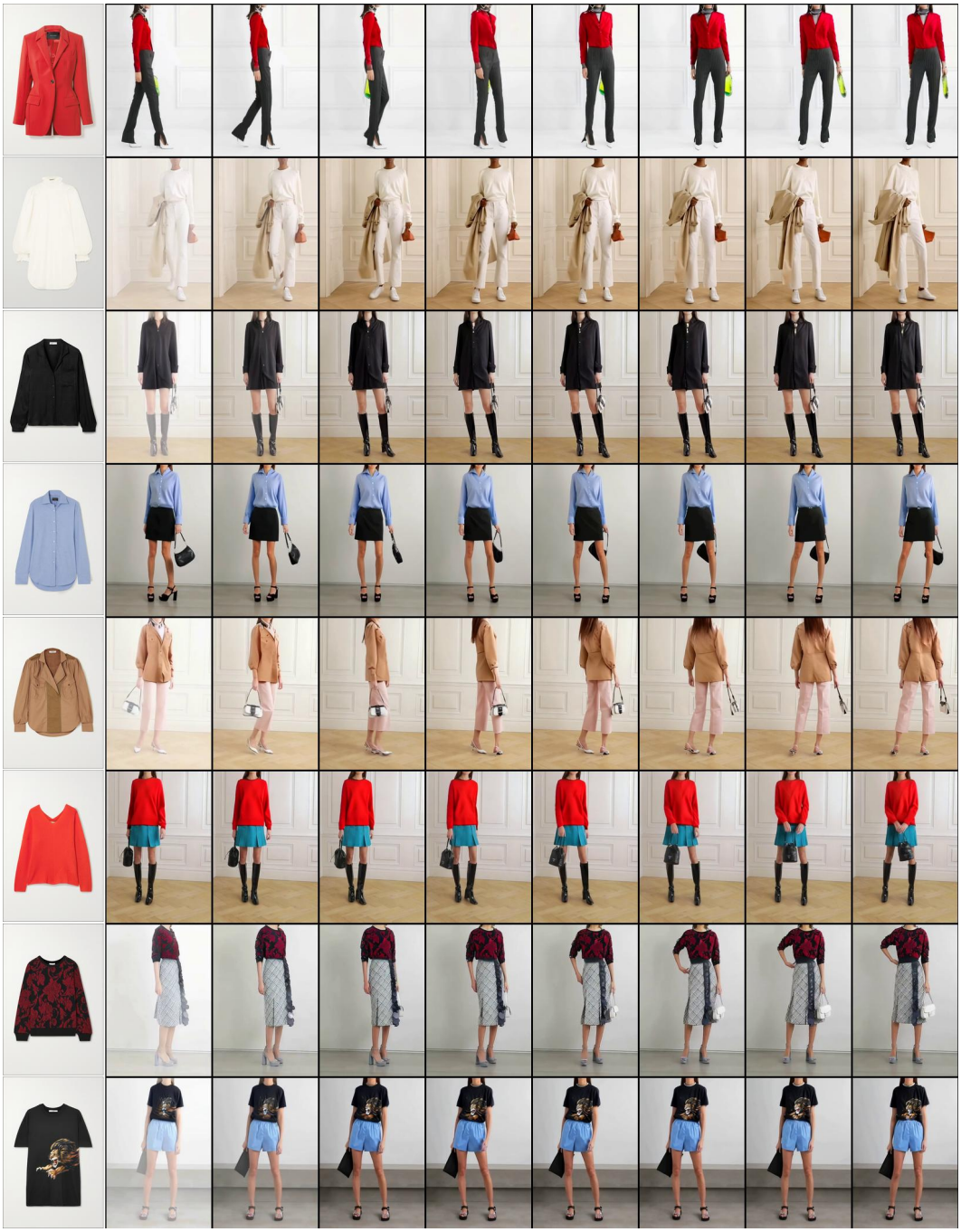

Figure 10: Qualitative results on ViViD.