SwiftTry: Fast and Consistent Video Virtual Try-On with Diffusion Models

Anonymous submission

Abstract

Given an input video of a person and a new garment, the objective of this paper is to synthesize a new video where the person is wearing the specified garment while maintaining spatiotemporal consistency. While significant advances have been made in image-based virtual try-ons, extending these successes to video often results in frame-to-frame inconsistencies. Some approaches have attempted to address this by increasing the overlap of frames across multiple video chunks, but this comes at a steep computational cost due to the repeated processing of the same frames, especially for long video sequence. To address these challenges, we reconceptualize video virtual try-on as a conditional video inpainting task, with garments serving as input conditions. Specifically, our approach enhances image diffusion models by incorporating temporal attention layers to improve temporal coherence. To reduce computational overhead, we introduce ShiftCaching, a novel technique that maintains temporal consistency while minimizing redundant computations. Furthermore, we introduce the TikTokDress dataset, a new video try-on dataset featuring more complex backgrounds, challenging movements, and higher resolution compared to existing public datasets. Extensive experiments show that our approach outperforms current baselines, particularly in terms of video consistency and inference speed. Code and dataset will be made available upon acceptance.

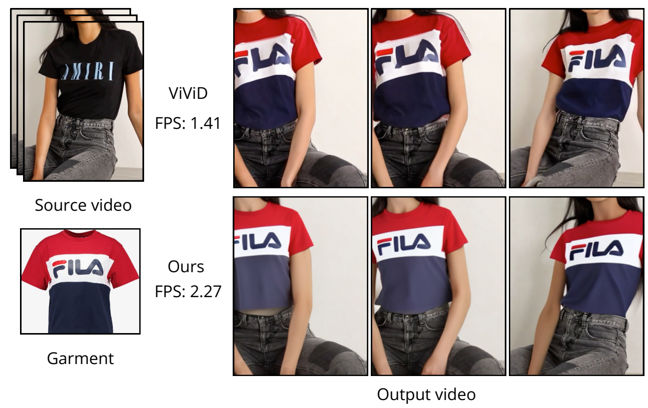

Figure 1: Results of our SwiftTry compared with those of ViViD (Fang et al. 2024), the previous method for video try-on. Our method maintains detail and consistency in the texture of the garments and runs over 60% faster.

Introduction

Video virtual try-on is an emerging research area (Chen et al. 2021; Rogge et al. 2014; Pumarola et al. 2019; Dong et al. 2019b; Kuppa et al. 2021; Zhong et al. 2021; Jiang et al. 2022; He et al. 2024; Xu et al. 2024b; Fang et al. 2024; Zheng et al. 2024) with significant potential in fashion and e-commerce. The ability to realistically visualize how a garment looks on a person in a video could transform online shopping. However, despite recent progress in image-based virtual try-on (He, Song, and Xiang 2022; Choi et al. 2021; Lee et al. 2022; Xie et al. 2023; Zhu et al. 2023; Kim et al. 2023), extending these capabilities to video remains challenging due to the need for spatiotemporal consistency and the high computational costs of processing long sequences.

A significant challenge in video virtual try-on is balancing the need for temporal coherence with the computational demands of processing long video sequences. Previous methods (Xu et al. 2024b; He et al. 2024; Fang et al. 2024) often struggle with temporal inconsistencies, leading to visual artifacts and flickering between frames, which undermines the realism of the virtual try-on experience. Additionally, the high computational cost of rendering high-quality results over extended sequences hampers the practicality of these approaches for real-world applications.

Another challenge is lacking an evaluation dataset. The first public video try-on dataset VVT (Dong et al. 2019b) only covers basic pattern garments, form-fitting T-shirts, uniform backgrounds, static camera angles, and repetitive human motions. Recently, ViViD (Fang et al. 2024) released the first practical dataset for video virtual try-on; however, it struggles to handle in-the-wild scenarios, such as complex movements and diverse backgrounds, making it challenging to meet the demands of practical applications. Moreover, one reason for the poor quality video try-on result is primarily due to masks extracted using human parsing segmentation (Li et al. 2020) applied on each frame of the video.

In this paper, we tackle these challenges by introducing two key contributions. First, we present a new high-quality dataset named **TikTokDress** consisting of 817 videos which is specifically designed for training and evaluating video virtual try-on models. This dataset features realistic scenes, diverse garment types, and complex movements, providing a robust foundation for advancing research in this field. Second, we introduce a novel video virtual try-on framework named **SwiftTry**, as shown in Fig. 1 which significantly reduces the computational cost of processing long video sequences while maintaining temporal consistency. Our framework is inspired by state-of-the-art diffusion-based image virtual try-on methods (Kim et al. 2023; Xu et al. 2024a; Choi et al. 2024) and incorporates temporal attention in the UNet architecture to train on video try-on data. During inference, we introduce a new technique called ShiftCaching, which ensures temporal coherence and smooth transitions between video clips while also reducing redundant computation compared to previous methods. Extensive experimental results indicate that our proposed SwiftTry framework, incorporating the aforementioned techniques, substantially surpasses other video virtual try-on methods in performance.

In summary, the contributions of our work are as follows:

- We propose a new technique for video inference named ShiftCaching, which can ensure temporal smoothness between video clips and reduce redundant computation.
- We introduce and curate a new video virtual try-on dataset, TikTokDress, which encompasses a wide range of backgrounds and complex movements and features high-resolution videos, filling a gap that exists in previous video virtual try-on datasets.

Related Work

Image Virtual Try-On. Traditional image virtual try-on methods (Han et al. 2018; Wang et al. 2018a; Dong et al. 2019a; Yang et al. 2020; Ge et al. 2021; He, Song, and Xiang 2022; Choi et al. 2021; Lee et al. 2022; Xie et al. 2023) commonly employ a two-stage pipeline based on **GANs** (Goodfellow et al. 2014). In this approach, the target clothing is first warped and then fused with the person image to create the try-on effect. Various techniques have been utilized for clothing warping, including thin-plate spline (TPS) warping (Han et al. 2018), spatial transformer networks (STN) (Li et al. 2021), and flow estimation (Xie et al. 2023). Despite these advances, such methods often face limitations in generalization, resulting in significant performance degradation when applied to person images with complex backgrounds.

Recently, **diffusion models** have markedly enhanced the realism of images in generative tasks, leading to their increasing adoption in virtual try-on research. For instance, TryOnDiffusion (Zhu et al. 2023) presents a virtual try-on method utilizing two U-Nets, but it requires a large dataset of image pairs depicting the same person in various poses, which can be difficult to acquire. StableVITON (Kim et al. 2023) conditions the garment in a ControlNet (Zhang, Rao, and Agrawala 2023)-style using a zero cross-attention block, while IDM-VTON (Choi et al. 2024) proposes GarmentNet to encode low-level features combined with high-level semantic features extracted via IP-Adapter (Ye et al. 2023). Despite these advancements, extending these existing image virtual try-on methods for video often results in significant inter-frame inconsistency and flickering, which adversely affects the overall quality of the generated results.

Video Virtual Try-On. Several efforts have been made to develop virtual try-on for videos. FW-GAN (Dong et al. 2019b) integrates an optical flow prediction module from Video2Video (Wang et al. 2018b) to warp preceding frames to the current frame, enabling the synthesis of temporally coherent subsequent frames. MV-TON (Zhong et al. 2021) introduces a memory refinement module that retains and refines features from previous frames. ClothFormer (Jiang et al. 2022) utilizes a vision transformer in its try-on generator to minimize blurriness and temporal artifacts, and it features an innovative warping module that combines TPS-based and appearance-based methods to address issues like incorrect warping due to occlusions. Among **diffusion-based** methods, Tunnel Try-On (Xu et al. 2024b) is the first to apply diffusion models for video virtual try-on, effectively handling camera movement and maintaining consistency, although its demo videos are limited to only a few seconds long. ViViD (Fang et al. 2024) introduced a large-scale video try-on dataset with multiple categories, but it remains limited by simple backgrounds and movements, which constrains its ability to maintain long-term consistency and coherence. In this paper, we propose a novel technique that ensures temporal smoothness and coherence across video clips, complemented by a caching technique (Ma, Fang, and Wang 2024) that reduces redundant computations during long video inference.

Methods

Problem Statement: Given a source video $V = \{I_1, I_2, \ldots, I_N\} \in \mathbb{R}^{N\times3\times H\times W}$ of a person and a garment image $g \in \mathbb{R}^{3\times H\times W}$, where N, H, and W represent the video length, frame height, and frame width, respectively, our goal is to synthesize an target video $\hat{V} = \{\hat{I}_1, \hat{I}_2, \ldots, \hat{I}_N\} \in \mathbb{R}^{N\times3\times H\times W}$ of the person wearing the garment, while preserving the motion of the person, the background in V, and the color and texture of g.

It is important to note that collecting both source and target videos of the same person with identical motion and gestures, differing only in the garment, is extremely challenging. As a result, most video try-on approaches adopt a **self-supervised training** method, where only a single video is used, and the garment regions are masked. The model is then trained to inpaint the masked regions using guidance from the garment image.

In the next section, we first describe our overall video try-on architecture and then discuss in detail the ShiftCaching technique – one of our main contributions.

Overall Architecture

Our approach consists of two stages: first, training a diffusion-based image try-on model, and then extending it to work with video data by incorporating temporal attention into every block of the Main UNet.

In the first stage, inspired by StableVITON (Kim et al. 2023), we design **diffusion-based image try-on model** with two submodules: the Garment UNet and the Main UNet, as illustrated in Fig. 2. The *Main UNet* is a modified inpainting model, initialized with the pretrained weights from Stable

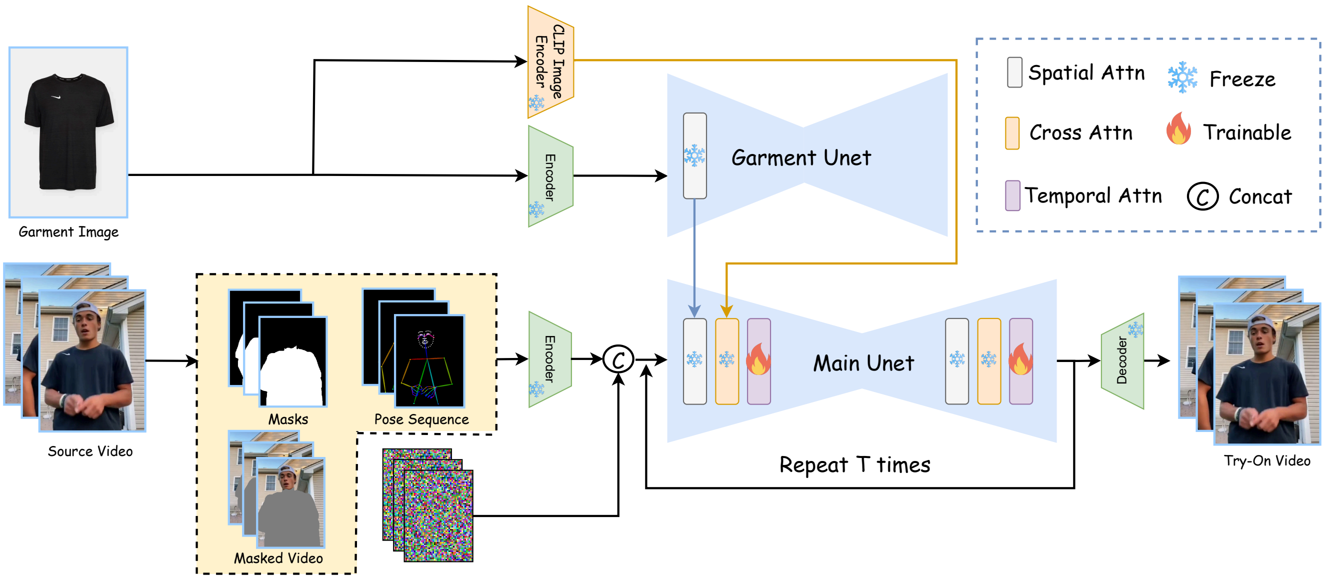

Figure 2: Overview of Stage 2 of our SwiftTry framework. Stage 1 is similar except the input is single image frame and does not have temporal attention layers. Given an input video and a garment image, our method first extracts the masked video, corresponding masks, and pose sequence. The masked video is encoded into latent space by the VAE Encoder, which is then concatenated with noise, masks, and pose features before being processed by the Main U-Net. To inpaint the garment during the denoising process, we use a Garment U-Net and a CLIP encoder to extract both low- and high-level garment features. These features are integrated into the Main U-Net through spatial and cross-attention mechanisms.

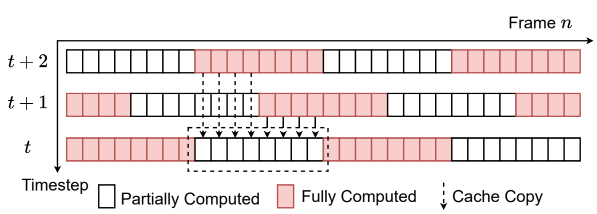

Figure 3: Illustration of fully and partially computed frames with a chunk size $N = 8$ and shift $\Delta = 4$. In the partially computed chunk, one half uses cached features from timestep $t+2$, and the other half uses features from $t+1$.

Overlapping size	$\text{VFID}_{\text{I3D}} \downarrow$	FPS $\uparrow$
$S = 0$	9.040	**1.544**
$S = 4$	8.822	1.176
$S = 8$	8.947	0.801
$S = 15$	**8.675**	0.104

Table 1: Trade-off between speed (FPS) and consistency (VFID_{I3D}) with different overlap sizes of prior method.

Diffusion (Rombach et al. 2022). It takes as input four channels of latent noise, four channels of latent representations of the masked image (i.e., the person image with the clothes region masked), and one channel for the binary mask representing the inpainting region. To further enhance the generation, we add the pose skeleton as an additional control, represented by a pose map rendered from DW-Pose (Yang et al. 2023). This results in a 13-channel input fed into the Main UNet, which predicts the cleaned latent over T timesteps. Finally, the cleaned latent is passed through a decoder to produce the output image. The *Garment UNet* has a similar architecture to the Main UNet but only takes the garment image as its input, rather than the multiple channels used in the Main UNet. This module is designed to extract both detailed and high-level features from the garment, guiding the Main UNet to accurately replicate the garment's appearance through its spatial and cross-attention layers. We use the VITON-HD dataset (Choi et al. 2021) to train our network during this stage.

In the second stage of designing our **diffusion-based video try-on model**, we ensure temporal consistency across video frames by adapting the Main UNet from the image try-on model. This adaptation involves converting its 2D layers into pseudo-3D layers (Guo et al. 2023; Wu et al. 2023; Zhou et al. 2022) and adding a temporal attention layer after the Spatial and Cross-Attention layers to capture temporal correlations between frames. The architecture of the modified UNet blocks is illustrated in Fig. 2. In the temporal attention layer, the features are reshaped into the shape of $(H \times W) \times N \times C$, where C is the number of feature channels and $H \times W$ is the batch-size dimension, to compute self-attention along the temporal dimension. This allows a location in the latent space of frame t to interact with the same location in other frames within a chunk of N frames. This design is highly efficient as it avoids the expense of full 3D attention by factorizing it into two consecutive steps:

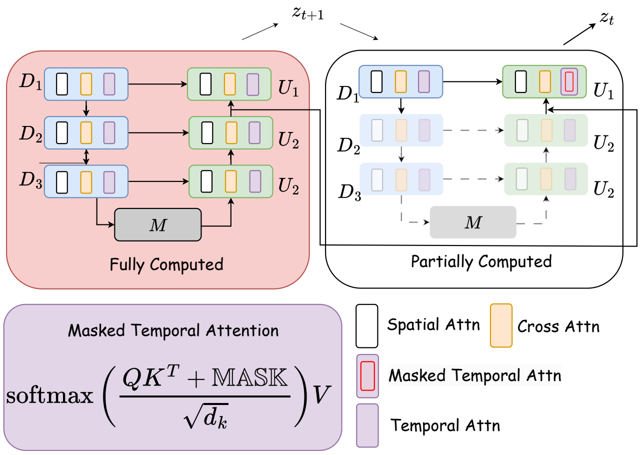

Figure 4: Comparison of a fully computed frame and a partially computed frame. The partially computed frame uses Masked Temporal Attention instead of standard Temporal Attention to address mismatches in cached features.

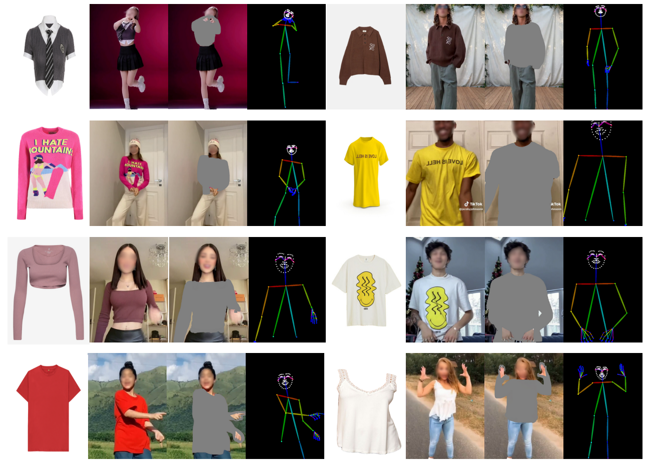

Figure 5: Example videos from the TikTokDress dataset highlighting diversity in skin tones, genders, camera angles, and clothing types.

spatial attention (2D) and temporal attention (1D). This approach allows a location in one frame to exchange information with every location in all other frames. Additionally, we incorporate sinusoidal positional encoding to help the model recognize the position of each frame in the video, following (Guo et al. 2023). In this stage, we train only the temporal attention layer while keeping the other layers unchanged, using a video dataset.

ShiftCaching Technique

Due to memory constraints, current video diffusion-based virtual try-on methods can only generate video chunks of 16 frames at a time. Previous approaches (Fang et al. 2024; He et al. 2024; Xu et al. 2024b) use a temporal aggregation technique (Tseng, Castellon, and Liu 2023; Xu et al. 2023) to stitch overlapping video clips into longer sequences. In this process, the long video is divided into overlapping clips with an overlap size S, typically set to $N/2$ or $N/4$. At each denoising timestep t, the overlapping noise predictions are merged using a simple averaging technique. However, this method involves a trade-off: a smaller overlap size, such as $S = 4$, can cause temporal flickering and texture artifacts, while a larger overlap size, such as $S = 15$, improves consistency but greatly slows down the process as shown in Tab. 1.

To achieve good temporal coherence and smoothness without recomputing the overlapped regions, we propose a **shifting mechanism** during inference. Specifically, we divide the long video into non-overlapping chunks ($S = 0$). At each DDIM sampling timestep t, we shift these chunks by a predefined value Δ between two consecutive frames, allowing the model to process different compositions of noisy chunks at each step. An example of a fixed $\Delta = 4$ applied to a chunk with length $N = 8$ is illustrated in Fig. 3.

To further accelerate the inference process, we can skip a random chunk to reduce redundant computation during denoising. However, dropping chunks without adjustment can lead to abrupt changes in noise levels in the final results. Following (Ma, Fang, and Wang 2024), which notes that adjacent denoising steps share significant similarities in high-level features, we instead perform **partial computations** on the Main U-Net. Specifically, we use a cache to copy the latest features from the fully computed timestep z_{t+1} (Red frame) and use these features to partially compute the current latent z_t (White frame), bypassing the deeper blocks of the UNet, as illustrated in Fig. 4.

When performing partial computations on a chunk, the cached features typically include a first half from timestep $t+2$ and a second half from timestep $t+1$, which can cause mismatches between the two halves. To address this, we introduce a **Masked Temporal Attention** mechanism. This mechanism incorporates a special mask with size $N \times N$ during the softmax attention calculation to set the attention matrix values to 0, preventing information transfer from less accurate features (timestep $t + 2$) to more accurate features (timestep $t + 1$), while allowing transfer from good features to bad features. This mechanism ensures both smoothness and high quality in the partially computed cells.

TikTokDress Dataset

Public datasets for single-image virtual try-ons, such as VITON-HD and DressCode, often suffer from simple backgrounds and limited human poses, despite offering high-quality images. These datasets are also restricted to single-image scenarios. Similarly, the VVT dataset, a standard for video virtual try-on, has notable drawbacks, including uniform movements, white backgrounds, and low resolution (256×192), making it unsuitable for real-world applications, particularly in the short-video industry where higher resolution is crucial. Given that real-world videos are typically recorded on mobile phones, which introduces variations in background, camera position, and lighting, there's a need for a more robust dataset. To address these gaps, we introduce TikTokDress, a high-resolution video virtual try-on

Label	Gender		Skin tone			Camera position			Distance		Action		Background	
	Male	Female	White	Yellow	Black	Bottom	Top	Center	Near	Far	Move	Stay	Dynamic	Static
Counting	267	550	541	124	152	576	7	231	570	247	275	542	94	724

Table 2: Data statistics illustrating our dataset's diversity and complexity. The table breaks down attributes such as gender, skin tone, camera positions, distances (Near/Far), and actions (Move/Stay), which indicate whether the actor is moving or stationary.

Figure 6: Row a) shows SAM 2 (Ravi et al. 2024) failures due to sensitivity to prompts (green for positive, red for negative), with manual corrections needed for challenging areas (green rectangles). Row b) demonstrates our approach's effectiveness in accurately covering garment areas, improving try-on results for casual clothing.

dataset that includes complex backgrounds, diverse movements, and gender representation. Each video is paired with the corresponding garment and is annotated with detailed human poses and precise binary cloth masks, enhancing its value for real-world applications.

First, the quality of garment masks in our dataset is crucial for enhancing try-on results, as shown in the Supplementary Material. While existing datasets like VITON-HD (Choi et al. 2021), DressCode (Morelli et al. 2022), and VVT (Dong et al. 2019b) use a standard segmenter (Li et al. 2020), it struggles with complex videos, resulting in subpar performance. In contrast, TikTokDress offers **manually corrected**, highly accurate garment masks, leading to significantly improved video try-on quality.

Second, our TikTokDress dataset captures a broad range of human poses and dynamic movements, such as dancing, common in short-form videos. As shown in Fig. 5, it includes variations in camera distance and diverse backgrounds, from indoor to outdoor settings with complex lighting. Additionally, it features a variety of clothing types, from casual T-shirts to structured garments like sweaters and challenging attire such as chainmail tops, addressing real-world challenges in video virtual try-on.

Video collection and annotation.

Our dataset consists of short TikTok clips (10-30 seconds) featuring various dance routines, as shown in Fig. 5. We expanded the TikTok Dataset (Jafarian and Park 2021) by adding videos to increase diversity in backgrounds, skin tones, and clothing styles, resulting in 817 video-garment pairs. Videos with excessive motion blur or low-quality garments were excluded. To ensure accurate garment matching, we manually selected high-quality matches from fashion retail websites. The dataset includes over 270,000 RGB frames extracted at 30 frames per second. We also calculated 2D keypoints and dense pose information using DWPose (Yang et al. 2023) and DensePose (Güler, Neverova, and Kokkinos 2018). Dataset statistics are summarized in Tab. 2, highlighting diversity in gender, skin tone, and camera positions.

Creating high-quality garment masks for each video was challenging due to the need for precise segmentation in every frame. We used SAM 2 (Ravi et al. 2024) to extract masks for both clothing and arms, but its sensitivity to prompt points and specific frames (see Fig. 6-(a)) required an additional solution. We developed an algorithm (described in the Supplemental Material) to automate the selection of optimal frames and prompts, improving efficiency across various scenarios. Despite these advancements, complex or unusual garments still needed manual intervention, which we detail with examples in the Supplementary Material. This meticulous process was essential for ensuring the dataset's high quality and reliability.

Experiments

Datasets: We evaluate our approach on the VVT dataset (Dong et al. 2019b) and our new TikTokDress dataset. The VVT dataset, a standard benchmark for video virtual try-on, includes 791 paired videos of individuals and clothing images, with 661 for training and 130 for testing, all at 256×192 resolution. The videos feature simple movements against plain backgrounds. In contrast, the TikTokDress dataset offers a more complex challenge, with varied backgrounds, dynamic movements, and diverse body poses. It comprises 693 training videos and 124 testing videos at 540×720 resolution, totaling 232,843 frames for training and 39,705 frames for testing.

Metrics: We evaluate our approach using image-based and video-based metrics in both paired and unpaired settings, as outlined in (Jiang et al. 2022). In paired settings, we use SSIM (Wang et al. 2004) and LPIPS (Zhang et al. 2018) to assess reconstruction quality. In unpaired settings, we measure visual quality and temporal consistency with Video Fréchet Inception Distance (VFID) (Dong et al. 2019b). Additionally, we measure inference speed in frames per second (FPS) to demonstrate speed improvements.

Implementation details: The training process is divided into two stages. In the first stage, we focus on inpainting and preserving detailed garment textures using the VITON-HD dataset (Choi et al. 2021). We fine-tune the Garment UNet, Pose Encoder, and Main UNet decoder, initializing

Method	VVT				
	LPIPS ↓	SSIM ↑	$VFID_{I3D}$ ↓	$VFID_{RN}$ ↓	FPS ↑
CP-VTON	0.535	0.459	6.361	12.10	N/A
FBAFN	0.157	0.870	4.516	8.690	N/A
StableVITON	0.184	0.760	17.068	11.254	0.2412
StableVITON+AA	0.270	0.683	12.597	3.336	1.165
FWGAN	0.283	0.675	8.019	12.15	N/A
MVTON	0.068	0.853	8.367	9.702	N/A
ClothFormer	0.081	**0.921**	3.967	5.048	N/A
Tunnel Try-On	**0.054**	<u>0.913</u>	**3.345**	4.614	N/A
ViViD†	0.119	0.829	6.788	<u>0.853</u>	<u>1.409</u>
WildVidFit	N/A	N/A	4.202	N/A	N/A
SwiftTry (ours)	<u>0.066</u>	0.887	<u>3.589</u>	**0.534**	**2.270**

Table 3: Comparisons on the VVT dataset (Dong et al. 2019b). † means our re-evaluation from the provided code.

Method	TikTokDress			
	LPIPS ↓	SSIM ↑	$VFID_{I3D}$ ↓	FPS ↑
ViViD†	0.129	0.824	5.638	1.409
SwiftTry w/o ShiftCaching	0.075	**0.891**	**3.865**	1.177
SwiftTry (ours)	**0.074**	0.888	4.231	**2.270**

Table 4: Comparisons on the TikTokDress dataset.

the Main UNet and Garment UNet with pretrained weights from SD 1.5, while keeping the VAE Encoder, Decoder, and CLIP image encoder weights unchanged. In the second stage, we incorporate temporal attention layers into the previously trained model, initializing these new modules with pretrained weights from AnimateDiff (Guo et al. 2023).

Comparisons with Prior Approaches

We compare our approach with other video virtual try-on methods using the VVT and TikTokDress datasets. As most methods are closed-source, we rely on reported results and available generated videos for comparison. For GAN-based methods, we evaluate against FW-GAN (Dong et al. 2019b), MV-TON (Zhong et al. 2021), and ClothFormer (Jiang et al. 2022). For diffusion-based methods, we compare with Tunnel Try-On (Xu et al. 2024b), ViViD (Fang et al. 2024), and WildVidFit (He et al. 2024). We re-evaluate ViViD (Fang et al. 2024) on the VVT dataset, as it is the only method with available inference code and pre-trained weights.

Quantitative results: Tab. 3 presents the comparison on the VVT dataset. Our approach excels in the VFID metric, indicating superior visual quality and consistency, and also performs competitively in SSIM and LPIPS scores. While ClothFormer achieves a high SSIM score, its VFID is lower due to the limitations of its GAN-based method. Our ShiftCaching technique enhances performance, increasing the frame rate to 2.27 FPS – over 1.5 times faster. We also evaluated our method on the TikTokDress dataset, as detailed in Tab. 4. Our analysis shows that while these methods produce accurate individual frames, they often struggle with flickering and inconsistencies due to poor temporal coherence and motion handling across frames.

Variant	LPIPS↓	SSIM↑	$VFID_{I3D}$↓	$VFID_{RN}$↓	FPS↑
FS	0.061	0.882	8.971	0.864	1.544
RS	0.060	0.883	**8.878**	**0.853**	1.544
FS, P 50%	0.060	0.883	8.932	0.887	**2.270**
RS, P 50%	0.060	0.883	8.938	0.888	**2.270**

Table 5: Study of our Temporal layers with ShiftCaching. FS: Fixed Shift, RS: Random Shift, P: Partially Computed.

Variant	LPIPS↓	SSIM↑	$VFID_{I3D}$↓	$VFID_{RN}$↓
FA	0.060	0.883	8.932	0.887
HA	**0.059**	**0.886**	**8.679**	**0.796**
QA	0.061	0.882	8.990	0.909
CA	0.086	0.854	13.520	5.501

Table 6: Ablation study of our Temporal layers with ShiftCaching. FA: Full Attention, HA: Half Attention, QA: Quarter Attention, CA: Causal Attention.

Variant	LPIPS↓	SSIM↑	$VFID_{I3D}$↓	$VFID_{RN}$↓	FPS↑
8	0.062	0.880	9.312	1.027	0.914
16	**0.061**	**0.881**	8.822	0.851	1.176
24	0.163	0.821	**8.724**	**0.800**	**1.723**

Table 7: Ablation study of testing with 8, 16, and 24 frames, with the default training set to 16 frames.

Qualitative results: As shown in Fig. 8 and Fig. 7, the textures on the garment vary between frames. Additionally, there are significant jitters between adjacent frames with these methods, which can be observed more intuitively in videos provided in our Supplementary Material.

Ablation Study

We conducted ablation studies on the VVT dataset to investigate various factors affecting the performance of SwiftTry.

Study on the ShiftCaching Technique is shown in Tab. 5. The results indicate that using random shifts provides the best consistency. When combined with partial computation of 50% of the frames, this approach accelerates inference by 1.5 times while maintaining comparable quantitative metrics to other methods.

Study on Different Types of Masks in Masked Temporal Attention is shown in Tab. 6. The results reveal that Half Attention yields the best performance. This suggests that allowing only the bad features (from timestep $t+2$) to access the good features (from timestep $t+1$) and allowing only the good features to interact with each other, produces the optimal results. Detailed explanations of different masking attention are described in our Supplementary Material.

Impact of Inference Video Chunk Length is examined in Tab. 7. The study reveals that matching the training and inference video chunk lengths – both set to $N = 16$ – yields

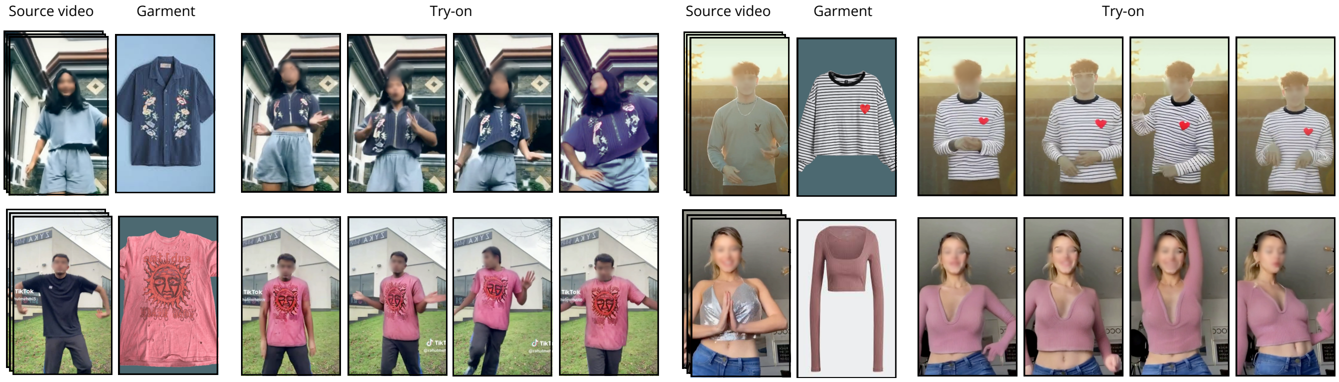

Figure 7: Qualitative results of our method on the TikTokDress dataset.

Figure 8: Qualitative comparison with prior method on the VVT dataset.

the best results.

Conclusion

In conclusion, we have proposed a novel technique, Shift-Caching, which ensures temporal smoothness across video clips while effectively reducing redundant computations during video inference. This advancement enhances the efficiency and quality of video virtual try-on, making it more practical for real-world applications. Additionally, we have introduced a new dataset, TikTokDress, designed specifically for video virtual try-on. This dataset stands out for its diverse range of backgrounds, complex movements, and high-resolution videos, addressing the limitations of existing datasets and providing a valuable resource for future research in this area.

References

Chen, C.-Y.; Lo, L.; Huang, P.-J.; Shuai, H.-H.; and Cheng, W.-H. 2021. Fashionmirror: Co-attention feature-remapping virtual try-on with sequential template poses. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 13809–13818.

Choi, S.; Park, S.; Lee, M.; and Choo, J. 2021. Viton-hd: High-resolution virtual try-on via misalignment-aware normalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 14131–14140.

Choi, Y.; Kwak, S.; Lee, K.; Choi, H.; and Shin, J. 2024. Improving Diffusion Models for Virtual Try-on. *arXiv preprint arXiv:2403.05139*.

Dong, H.; Liang, X.; Shen, X.; Wang, B.; Lai, H.; Zhu, J.; Hu, Z.; and Yin, J. 2019a. Towards Multi-Pose Guided Virtual Try-On Network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Dong, H.; Liang, X.; Shen, X.; Wu, B.; Chen, B.-C.; and Yin, J. 2019b. Fw-gan: Flow-navigated warping gan for video virtual try-on. In *Proceedings of the IEEE/CVF international conference on computer vision*, 1161–1170.

Fang, Z.; Zhai, W.; Su, A.; Song, H.; Zhu, K.; Wang, M.; Chen, Y.; Liu, Z.; Cao, Y.; and Zha, Z.-J. 2024. ViViD: Video Virtual Try-on using Diffusion Models. *arXiv preprint arXiv:2405.11794*.

Ge, Y.; Song, Y.; Zhang, R.; Ge, C.; Liu, W.; and Luo, P. 2021. Parser-free virtual try-on via distilling appearance flows. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8485–8493.

Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative Adversarial Networks. arXiv:1406.2661.

Güler, R. A.; Neverova, N.; and Kokkinos, I. 2018. Densepose: Dense human pose estimation in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7297–7306.

Guo, Y.; Yang, C.; Rao, A.; Wang, Y.; Qiao, Y.; Lin, D.; and Dai, B. 2023. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*.

Han, X.; Wu, Z.; Wu, Z.; Yu, R.; and Davis, L. S. 2018. Viton: An image-based virtual try-on network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7543–7552.

He, S.; Song, Y.-Z.; and Xiang, T. 2022. Style-based global appearance flow for virtual try-on. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3470–3479.

He, Z.; Chen, P.; Wang, G.; Li, G.; Torr, P. H.; and Lin, L. 2024. WildVidFit: Video Virtual Try-On in the Wild via Image-Based Controlled Diffusion Models. *arXiv preprint arXiv:2407.10625*.

Jafarian, Y.; and Park, H. S. 2021. Learning High Fidelity Depths of Dressed Humans by Watching Social Media Dance Videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 12753–12762.

Jiang, J.; Wang, T.; Yan, H.; and Liu, J. 2022. Clothformer: Taming video virtual try-on in all module. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10799–10808.

Kim, J.; Gu, G.; Park, M.; Park, S.; and Choo, J. 2023. StableVITON: Learning Semantic Correspondence with Latent Diffusion Model for Virtual Try-On. *arXiv preprint arXiv:2312.01725*.

Kuppa, G.; Jong, A.; Liu, X.; Liu, Z.; and Moh, T.-S. 2021. ShineOn: Illuminating design choices for practical video-based virtual clothing try-on. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 191–200.

Lee, S.; Gu, G.; Park, S.; Choi, S.; and Choo, J. 2022. High-Resolution Virtual Try-On with Misalignment and Occlusion-Handled Conditions. *arXiv preprint arXiv:2206.14180*.

Li, K.; Chong, M. J.; Zhang, J.; and Liu, J. 2021. Toward accurate and realistic outfits visualization with attention to details. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 15546–15555.

Li, P.; Xu, Y.; Wei, Y.; and Yang, Y. 2020. Self-correction for human parsing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(6): 3260–3271.

Ma, X.; Fang, G.; and Wang, X. 2024. Deepcache: Accelerating diffusion models for free. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15762–15772.

Morelli, D.; Fincato, M.; Cornia, M.; Landi, F.; Cesari, F.; and Cucchiara, R. 2022. Dress code: high-resolution multi-category virtual try-on. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2231–2235.

Pumarola, A.; Goswami, V.; Vicente, F.; De la Torre, F.; and Moreno-Noguer, F. 2019. Unsupervised image-to-video clothing transfer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 0–0.

Ravi, N.; Gabeur, V.; Hu, Y.-T.; Hu, R.; Ryali, C.; Ma, T.; Khedr, H.; Rädle, R.; Rolland, C.; Gustafson, L.; Mintun, E.; Pan, J.; Alwala, K. V.; Carion, N.; Wu, C.-Y.; Girshick, R.; Dollár, P.; and Feichtenhofer, C. 2024. SAM 2: Segment Anything in Images and Videos. *arXiv preprint arXiv:2408.00714*.

Rogge, L.; Klose, F.; Stengel, M.; Eisemann, M.; and Magnor, M. 2014. Garment replacement in monocular video sequences. *ACM Transactions on Graphics (TOG)*, 34(1): 1–10.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695.

Tseng, J.; Castellon, R.; and Liu, K. 2023. Edge: Editable dance generation from music. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 448–458.

Wang, B.; Zheng, H.; Liang, X.; Chen, Y.; Lin, L.; and Yang, M. 2018a. Toward characteristic-preserving image-based virtual try-on network. In *Proceedings of the European conference on computer vision (ECCV)*, 589–604.

Wang, T.-C.; Liu, M.-Y.; Zhu, J.-Y.; Liu, G.; Tao, A.; Kautz, J.; and Catanzaro, B. 2018b. Video-to-video synthesis. *arXiv preprint arXiv:1808.06601*.

Wang, Z.; Bovik, A. C.; Sheikh, H. R.; and Simoncelli, E. P. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4): 600–612.

Wu, J. Z.; Ge, Y.; Wang, X.; Lei, S. W.; Gu, Y.; Shi, Y.; Hsu, W.; Shan, Y.; Qie, X.; and Shou, M. Z. 2023. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 7623–7633.

Xie, Z.; Huang, Z.; Dong, X.; Zhao, F.; Dong, H.; Zhang, X.; Zhu, F.; and Liang, X. 2023. GP-VTON: Towards General Purpose Virtual Try-on via Collaborative Local-Flow Global-Parsing Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 23550–23559.

Xu, Y.; Gu, T.; Chen, W.; and Chen, C. 2024a. OOTDiffusion: Outfitting Fusion based Latent Diffusion for Controllable Virtual Try-on. *arXiv e-prints*, arXiv–2403.

Xu, Z.; Chen, M.; Wang, Z.; Xing, L.; Zhai, Z.; Sang, N.; Lan, J.; Xiao, S.; and Gao, C. 2024b. Tunnel Try-on: Excavating Spatial-temporal Tunnels for High-quality Virtual Try-on in Videos. *arXiv preprint arXiv:2404.17571*.

Xu, Z.; Zhang, J.; Liew, J. H.; Yan, H.; Liu, J.-W.; Zhang, C.; Feng, J.; and Shou, M. Z. 2023. Magicanimate: Temporally consistent human image animation using diffusion model. *arXiv preprint arXiv:2311.16498*.

Yang, H.; Zhang, R.; Guo, X.; Liu, W.; Zuo, W.; and Luo, P. 2020. Towards Photo-Realistic Virtual Try-On by Adaptively Generating-Preserving Image Content. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Yang, Z.; Zeng, A.; Yuan, C.; and Li, Y. 2023. Effective whole-body pose estimation with two-stages distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 4210–4220.

Ye, H.; Zhang, J.; Liu, S.; Han, X.; and Yang, W. 2023. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*.

Zhang, L.; Rao, A.; and Agrawala, M. 2023. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 3836–3847.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 586–595.

Zheng, J.; Zhao, F.; Xu, Y.; Dong, X.; and Liang, X. 2024. VITON-DiT: Learning In-the-Wild Video Try-On from Human Dance Videos via Diffusion Transformers. *arXiv preprint arXiv:2405.18326*.

Zhong, X.; Wu, Z.; Tan, T.; Lin, G.; and Wu, Q. 2021. Mv-ton: Memory-based video virtual try-on network. In *Proceedings of the 29th ACM International Conference on Multimedia*, 908–916.

Zhou, D.; Wang, W.; Yan, H.; Lv, W.; Zhu, Y.; and Feng, J. 2022. Magicvideo: Efficient video generation with latent diffusion models. *arXiv preprint arXiv:2211.11018*.

Zhu, L.; Yang, D.; Zhu, T.; Reda, F.; Chan, W.; Saharia, C.; Norouzi, M.; and Kemelmacher-Shlizerman, I. 2023. TryOnDiffusion: A Tale of Two UNets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4606–4615.

Reproducibility Checklist

1. **This paper**:

   Question: Includes a conceptual outline and/or pseudocode description of AI methods introduced

   Answer: [Yes]

   Question: Clearly delineates statements that are opinions, hypothesis, and speculation from objective facts and results

   Answer: [Yes]

   Question: Provides well marked pedagogical references for less-familiare readers to gain background necessary to replicate the paper

   Answer: [Yes]

2. **Does this paper make theoretical contributions?** [No]

3. **Does this paper rely on one or more datasets?** [Yes]

   Question: A motivation is given for why the experiments are conducted on the selected datasets (yes/partial/no/NA)

   Answer: [Yes]

   Question: All novel datasets introduced in this paper are included in a data appendix. (yes/partial/no/NA)

   Answer: [Yes] . We release a new dataset in this paper.

   Question: All novel datasets introduced in this paper will be made publicly available upon publication of the paper with a license that allows free usage for research purposes. (yes/partial/no/NA)

   Answer: [Yes] . We will publish after the paper acceptance.

   Question: All datasets drawn from the existing literature (potentially including authors' own previously published work) are accompanied by appropriate citations. (yes/no/NA)

   Answer: [Yes]

   Question: All datasets drawn from the existing literature (potentially including authors' own previously published work) are publicly available. (yes/partial/no/NA)

   Answer: [Yes]

   Question: All datasets that are not publicly available are described in detail, with explanation why publicly available alternatives are not scientifically satisficing. (yes/-partial/no/NA)

Answer: [NA] . All datasets used in this paper are publicly available.

4. **Does this paper include computational experiments?** [Yes]

   Question: Any code required for pre-processing data is included in the appendix. (yes/partial/no).

   Answer: [No]

   Question: All source code required for conducting and analyzing the experiments is included in a code appendix. (yes/partial/no)

   Answer: [Yes]

   Question: All source code required for conducting and analyzing the experiments will be made publicly available upon publication of the paper with a license that allows free usage for research purposes. (yes/partial/no)

   Answer: [Yes]

   Question: All source code implementing new methods have comments detailing the implementation, with references to the paper where each step comes from (yes/partial/no)

   Answer: [Yes]

   Question: If an algorithm depends on randomness, then the method used for setting seeds is described in a way sufficient to allow replication of results. (yes/partial/no/NA)

   Answer: [Yes]

   Question: This paper specifies the computing infrastructure used for running experiments (hardware and software), including GPU/CPU models; amount of memory; operating system; names and versions of relevant software libraries and frameworks. (yes/partial/no)

   Answer: [Yes]

   Question: This paper formally describes evaluation metrics used and explains the motivation for choosing these metrics. (yes/partial/no)

   Answer: [Yes]

   Question: This paper states the number of algorithm runs used to compute each reported result. (yes/no)

   Answer: [Yes]

   Question: Analysis of experiments goes beyond single-dimensional summaries of performance (e.g., average; median) to include measures of variation, confidence, or other distributional information. (yes/no)

   Answer: [No]

   Question: The significance of any improvement or decrease in performance is judged using appropriate statistical tests (e.g., Wilcoxon signed-rank). (yes/partial/no)

   Answer: [No]

   Question: This paper lists all final (hyper-)parameters used for each model/algorithm in the paper's experiments. (yes/partial/no/NA)

   Answer: [Yes]

   Question: This paper states the number and range of values tried per (hyper-) parameter during development of the paper, along with the criterion used for selecting the final parameter setting. (yes/partial/no/NA)

   Answer: [Yes]