

Generative AI for Lifelike Digital Garment Visualization

Johanna Suvi Karras

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2026

Reading Committee:

Ira Kemelmacher-Shlizerman, Chair

Steve Seitz

Brian Curless

Program Authorized to Offer Degree:

Computer Science & Engineering

© Copyright 2026

Johanna Suvi Karras

University of Washington

Abstract

Generative AI for Lifelike Digital Garment Visualization

Johanna Suvi Karras

Chair of the Supervisory Committee:

Ira Kemelmacher-Shlizerman

Computer Science & Engineering

The fashion industry is undergoing a rapid digital transformation. With consumers increasingly shopping for clothes online and brands seeking more engaging, cost-effective tools for showcasing clothing online, the demand for generative AI tools for garment visualization is growing. Despite rapid progress in this space, particularly in virtual try-on, existing methods still struggle to accurately capture physical realism, such as how a garment drapes and flows in motion or how a garment fits on a real body. In this thesis, I introduce a suite of generative AI-based methods that address five core dimensions of this challenge – fabric dynamics, multi-view consistency, precise conditioning, temporal smoothness, and garment fit – advancing the state-of-the-art in realistic digital garment visualization.

First, I present *DreamPose*, a diffusion-based method for synthesizing animated fashion videos from a still image. Given an input image and a driving pose sequence, DreamPose generates a video depicting realistic human and fabric motion. The novelty of this work is leveraging a pretrained text-to-image model (Stable Diffusion) for image animation – proposing novel architectural adaptations to ensure temporal consistency and enable joint image-and-pose conditioning. A key challenge with this method is maintaining precise input image fidelity. We address this through a subject-specific

finetuning strategy that improves person and garment identity, without sacrificing generalization to new pose sequences. DreamPose produces state-of-the-art results on the fashion video animation task across a diverse range of clothing styles and poses.

Next, I introduce *Fashion-VDM*, a video diffusion model for video virtual try-on. Given a garment image and a person video, the model synthesizes a temporally consistent try-on video that renders the target garment onto the person, while preserving the person’s identity and motion. Although image-based virtual try-on has achieved impressive results, existing video try-on methods rely on multiple networks and suffer from poor quality due to warping artifacts. Fashion-VDM addresses these limitations through a unified diffusion transformer architecture augmented with temporal blocks, split classifier-free guidance, and a progressive temporal training strategy. Fashion-VDM is the first diffusion-based video virtual try-on method, and qualitative and quantitative experiments demonstrate that it outperforms existing methods by a wide margin.

I then discuss *HoloGarment*, a novel view synthesis method for in-the-wild garments. Given one to three images or a video of a person wearing a garment, HoloGarment generates 360° views of that garment in a canonical pose. This is a challenging task due to the significant occlusions, body pose variations, and cloth deformations present in real-world imagery. Prior methods fail to handle these conditions, because existing 3D datasets mostly consist of static, unoccluded, and rigid objects. Our key insight is to bridge this domain gap through an implicit training paradigm that jointly leverages large-scale real video data and small-scale synthetic 3D data to optimize a shared 2D-3D garment embedding space. At inference time, this shared space enables HoloGarment to transform real-world garments into static, canonical 360° spin visualizations. Furthermore, HoloGarment can leverage this shared space to handle dynamic video-to-360° novel view synthesis via a garment-specific “atlas” finetuning stage that captures garment geometry and texture across all viewpoints into a single garment embedding. Extensive experiments demonstrate that HoloGarment achieves state-of-the-art performance on novel view synthesis of in-the-wild garments, robustly handling wrinkling, pose variation, and occlusion while maintaining photorealism, view consistency,

and geometric accuracy.

Finally, I describe how we take the first steps toward fit-aware virtual try-on – depicting not just how a garment looks, but how it actually fits a specific person. Despite being crucial to the shopping experience, fit is largely ignored by existing virtual try-on methods, which instead default to generating well-fitted results regardless of body or garment size. This limitation is largely due to the lack of VTO training data that includes both size information and ill-fit scenarios (where the garment is too tight or too loose on the person). To bridge this gap, we introduce *FIT* (Fit-Inclusive Try-on), a large-scale dataset of over one million try-on samples accompanied by precise body and garment measurements. *FIT* is constructed with a scalable synthetic pipeline: 3D garments are programmatically generated and draped using physics simulation, then transformed into photorealistic images through a novel re-texturing framework that strictly preserves person and garment geometry. Leveraging *FIT*, we train *Fit-VTO*, a baseline fit-aware virtual try-on model that sets the new state-of-the-art and provides a robust benchmark for future research.

Contents

1	Introduction	5
2	Background	7
2.1	Diffusion Models	7
2.2	Video Diffusion Models	9
2.3	Flow Matching Models	10
2.4	Conditioning Mechanisms	11
3	DreamPose: Fashion Image-to-Video Synthesis via Stable Diffusion	14
3.1	Related Works	16
3.2	Stable Diffusion	18
3.3	Method	19
3.4	Experiments	26
3.5	Limitations and Future Work	31
4	Fashion-VDM: Video Diffusion Model for Virtual Try-On	39
4.1	Related Works	41
4.2	Method	43
4.3	Experiments	49
4.4	Limitations and Future Work	58
5	HoloGarment: 360° Novel View Synthesis of In-the-Wild Garments	66
5.1	Related Works	68
5.2	Method	69

5.3	Experiments	75
5.4	Limitations and Future Work	81
6	FIT: A Large-Scale Dataset for Fit-Aware Virtual Try-On	89
6.1	Related Works	91
6.2	Fit-Inclusive Try-on (FIT) Dataset	94
6.3	Fit-Aware Virtual Try-On	98
6.4	Experiments	100
6.5	Scope and Limitations	105
7	Conclusion and Future Work	119
7.1	Foundation Models for Digital Fashion	119
7.2	Open Problems and Future Work	121

ACKNOWLEDGMENTS

First and foremost, I would like to express my deepest gratitude to my advisor, Ira Kemelmacher-Shlizerman, for her exceptional mentorship throughout my PhD. Ira has provided invaluable guidance on my research, challenged me to uphold rigorous standards, and created opportunities for me to grow as a researcher. Her constructive feedback, openness to ambitious new projects, and unwavering support have shaped my approach to research and equipped me with skills that will serve me throughout my career.

I would also like to thank my committee members, Steve Seitz and Brian Curless, for their guidance, support, and thoughtful feedback throughout my PhD milestones.

I am grateful to have had the opportunity to collaborate with fellow researchers in the Graphics and Imaging Lab (GRAIL) at UW who continue to inspire me with their intelligence, creativity, and insightful discussions: Ben Jones, Aleksander Holynski, Keunhong Park, Chung-Yi Weng, Xuan Luo, Isaac Tian, Roy Or-El, James Noeckel, Felix Hahnlein, Yifan Wang, Nikita Haduong, Xiaojuan Wang, Yuxuan Mei, Vivek Jayaram, Teerapat Jenrungrot, Luyang Zhu, Jingwei Ma, Alice Gao, Mengyi Shan, Benlin Liu, Meng-Li Shih, Bowei Chen, Susung Hong, and Yuanhao Wang.

I am further deeply grateful for the incredible opportunities I had during my PhD to collaborate with researchers at Nvidia and Google. In particular, I would like to acknowledge my mentors and co-authors: Ting-Chun Wang, Yingwei Li, Yasamin Jafarian, Andreas Lugmayr, Innfarn Yoo, Chris Lee, Nan Liu, Akash Sengupta, Varsha Ramakrishnan, and everyone else on the ARML team at Google. These collaborations were invaluable for my growth as a researcher and taught me about how to do research with real-world impact. In particular, I would like to extend my thanks to my manager at Google, Yingwei, whose patient and kind mentorship allowed me to grow tremendously as a researcher and engineer.

My PhD would not have been possible without the love and support of my family – my parents,

my older brothers, my sister-in-law, and my niece. To my parents, thank you for teaching me hard work, humility, and resilience through your example and for encouraging me to pursue challenging goals. To my brothers, thank you for paving the way for me both in academics and in life, as well as for your encouragements, guidance, and sense of humor. To my sister-in-law, thank you for being a kind, generous, and delightful addition to our family. To my niece, I hope you too develop a love and curiosity for learning and discovery.

Finally, I am profoundly grateful to my partner, Hayes, for choosing to love and support me every day. To Hayes – You make my life endlessly more vibrant and you have also become an irreplaceable part of it. You are at once my boyfriend, workout partner, emotional support animal, sous-chef, fellow adventurer, comedian, hype man, and most importantly, best friend. Thank you for being my biggest cheerleader during moments of success, as much as for being my biggest supporter during the challenging ones. I am so grateful to have had you by my side throughout this journey.

Chapter 1

INTRODUCTION

The fashion industry is going digital. Nearly half of consumers now purchase clothing online [22] and fashion designers are increasingly turning to advanced 3D garment design software, such as CLO [1], Marvelous Designer [30], and Style3D [161], to design their collections. Moreover, social media platforms have emerged as digital storefronts and powerful advertising channels for showcasing clothing brands, new products, and trending styles.

Along with this digital transition comes an opportunity for more engaging and informative media for presenting clothing online. No longer is fashion constrained to still images printed in magazines or on billboards. What if a traditional fashion campaign were instead a video, where you could watch the garments come to life? What if you could inspect a garment in 3D before purchasing it online? What if you could step into a virtual fitting room and be able to see how a garment fits you from every angle? Realizing this vision requires rethinking the tools and mediums through which fashion is created, presented, and experienced in the digital age.

Generative AI models are emerging as particularly powerful in enabling such capabilities. Virtual try-on tools, like Doppl [59], already allow users to visualize how a garment looks on themselves, without needing to enter a fitting room and while retaining the convenience of online shopping. In fact, virtual try-on currently represents one of the most sought-after features by online shoppers [22]. For advertisers, AI-powered photography platforms such as Botika [19] allow brands to generate studio-quality on-model imagery and are often more time- and cost-effective for creating campaigns, compared to organizing a traditional photoshoot. Generative AI is also accelerating the garment design process. For instance, Refabric [143] turns text prompts and mood boards into garment concepts, technical patterns, and 3D mockups and AI-powered fashion design tools, like Style3D [161], accelerate iteration loops and reduce material waste during prototyping.

The ultimate goal is to accurately visualize and simulate how a garment would *actually* look in real life. However, despite recent breakthroughs in virtual try-on, fashion image synthesis, and 3D garment reconstruction, existing tools often still fall short of capturing physical realism. Several key challenges towards realizing this goal include synthesizing: (1) realistic fabric dynamics and motion; (2) consistent multi-view imagery; and (3) geometric and fit accuracy that reflects how a garment conforms to the wearer’s body shape.

In this thesis, I introduce a set of generative AI-based methods that address these challenges. I begin with a background on recent generative models and controllability mechanisms for image and video synthesis in Chapter 2, which forms the technical foundation for the contributions that follow. I then present my works leveraging generative AI for realistic digital garment visualization: (1) a method for animating still fashion images given a driving pose sequence (Chapter 3); (2) a video diffusion model for video virtual try-on (Chapter 4); (3) a framework for generating 360° novel views of in-the-wild garments from images and videos (Chapter 5); and (4) a dataset and model for fit-aware virtual try-on conditioned on person–garment measurements (Chapter 6). Finally, I propose future research directions for open problems and remaining challenges in this space.

Chapter 2

BACKGROUND

In this chapter, I review the fundamental theory and key advances in generative modeling that underpin the methods and applications explored in this thesis. I begin with an overview of diffusion models for image and video generation, followed by a discussion of flow matching as a related generative paradigm. Lastly, I examine conditioning mechanisms that enable controllable generation with these models.

2.1 Diffusion Models

Diffusion models [37, 76, 160, 162, 164] have recently emerged as a powerful class of generative models, achieving state-of-the-art results in image synthesis [141, 147, 149], video generation [78, 128], and 3D generation [137, 159]. Diffusion models learn to map samples from a random noise distribution to a target data distribution through an iterative denoising process. This is achieved via a forward noising process and a learned reverse denoising process. In the forward process, a data sample x is gradually corrupted by adding Gaussian noise over T timesteps, producing a sequence of latent variables $\{x_t\}_{t=1}^T$ that approaches pure Gaussian noise $x_T \sim \mathcal{N}(0, I)$. A forward step is defined by:

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1 - \beta_t}x_{t-1}, \beta_t I) \quad (1)$$

where β_t represents a fixed noise schedule. In the reverse process, a neural network ϵ_θ is trained to iteratively invert this corruption:

$$p_\theta(x_{t-1}, x_t) = \mathcal{N}(\mu_\theta, \Sigma_\theta) \quad (2)$$

Rather than directly predicting the clean sample, modern diffusion models are commonly trained to predict the noise component ϵ added to a noisy sample x_t at timestep t [76]. Given a conditioning signal c (e.g., text, images, pose, etc.), the model predicts:

$$\hat{\epsilon}_t = \epsilon_\theta(x_t, t, c) \quad (3)$$

During training, the model is optimized using a mean squared error objective:

$$\mathcal{L} = \mathbb{E}_{z_t, \epsilon \sim \mathcal{N}(0, I)} [\|\epsilon - \epsilon_\theta(x_t, t, c)\|_2^2] \quad (4)$$

where ϵ denotes the ground-truth noise at timestep t .

Latent Diffusion Models: Training diffusion models from scratch in pixel space is computationally expensive. To reduce training cost, Latent Diffusion Models (LDMs) [147] perform the diffusion process in the latent space of a pretrained VAE [96]. By compressing the data while preserving semantic structure, LDMs significantly reduces the computational requirements and training time with only marginal reductions to visual fidelity. As a result, pretrained latent diffusion models have become a common backbone for a wide range of downstream tasks, including image editing [20], personalization [148], and guided generation [11].

Diffusion Transformer (DiT): While early diffusion models adopt a UNet backbone to parameterize the denoising network ϵ_θ [37, 76, 162], the Diffusion Transformer (DiT) [136] has shifted the paradigm towards Transformer-based diffusion backbones [172]. DiTs split images into patches, which are treated as tokens in a Transformer [172]. Unlike UNets, which rely on local convolutional inductive biases, Transformers use attention to model long-range dependencies and dense correspondences between inputs and outputs, thereby improving the model’s ability to reason about spatial structure and semantic relationships. Transformer-based architectures have become the standard for modern large-scale generative models, including Flux [16] and Stable Diffusion 3 [43], to achieve high fidelity, strong conditioning, and improved generalization across diverse generation tasks.

2.2 Video Diffusion Models

Video diffusion models (VDMs) extend image diffusion models to generate temporally coherent sequences of frames [17, 18, 62, 64, 71, 75, 78, 88, 158]. However, out-of-the-box image diffusion models [76, 147, 162] generate frames independently, leading to severe flickering and temporal artifacts when applied to video. Therefore, synthesizing temporally consistent frames over long time horizons is a quintessential challenge with video diffusion models.

To learn temporal consistency, early VDM’s adapt traditional text-to-image diffusion models by injecting temporal blocks, such as 3D convolutions and temporal attention layers, within the denoising UNet architecture [18, 75, 78]. However, training a VDM from scratch is expensive, requiring large-scale video datasets and extensive training. To minimize these requirements, some methods finetune a pretrained image model for video generation [69, 75, 158, 196]. However, these methods are limited to generating short clips, typically 16-24 frames, per forward pass, due to the increasing complexity of 3D convolutions and temporal attention over long sequences.

Long Video Generation To generate longer sequences, many VDMs incorporate autoregressive inference strategies to iteratively generate longer video sequences by conditioning previously-generated frames. A simple approach conditions future frames on past frames [78, 174]. Other approaches condition on sparse key-frames [26] or temporally overlapping frames [176, 191]. StreamingT2V [72] proposes dedicated short- and long-term memory modules: a conditional attention module maintains consistency across chunk transitions, while an appearance preservation module retains global scene features across the full sequence, enabling generation of videos exceeding 1200 frames.

Another line of works implement architectural adaptations to enable increasingly longer video generation. Imagen-Video [75] introduces a cascaded diffusion pipeline to generate low-resolution videos followed by spatial and temporal super-resolution stages. Other hierarchical generation approaches introduce nested or multi-scale architectures to progressively refine video outputs [61].

Another approach, latent video diffusion models (LVDMs) [17, 18, 62, 64, 71, 126, 179], has been proposed to enable longer sequences and higher resolutions by performing diffusion in a compressed latent space, significantly improving efficiency while maintaining visual quality.

Transformer-Based VDMs More recently, there has been a shift toward Transformer-based architectures for video generation, driven by their strong scalability and ability to model long-range temporal dependencies. Early works such as Phenaki [174] demonstrated the feasibility of generating long, temporally coherent videos using autoregressive Transformer-based models operating over compressed latent representations. More recent large-scale systems, including Sora [132] and Veo [57], further explore Transformer-centric or hybrid architectures for video generation, achieving significant improvements in realism, temporal consistency, and prompt adherence at unprecedented scales.

2.3 Flow Matching Models

Flow matching models [8, 111, 116] have recently emerged as a compelling alternative generative modeling framework to diffusion models. Rather than modeling a stochastic forward-and-reverse Markov chain as in diffusion, flow matching models learn a deterministic continuous normalizing flow parameterized as an ordinary differential equation (ODE) to transport samples from a source noise distribution π_0 to the target data distribution π_1 . Concretely, a neural network v_θ is trained to predict the velocity field v_t along the transport trajectory at timestep $t \in [0, 1]$:

$$dZ_t = v_\theta(x_t, t)dt \tag{5}$$

which transports noisy latent $x_0 \in \pi_0$ towards $x_1 \in \pi_1$.

A popular flow matching variant is rectified flow (RF) matching [116], which constrains the transport trajectories to straight lines. This linearization simultaneously simplifies the training objective—reducing it to a mean-squared regression against a constant target direction, as well as

enables faster inference since its trajectories can be accurately integrated with few sampling steps. The RF training objective is:

$$\mathcal{L} = \mathbb{E}_{x_0 \sim \pi_0, x_1 \sim \pi_1} [\|(x_1 - x_0) - v_\theta(x_t, t)\|_2^2] \quad (6)$$

where $x_t = (1 - t) \cdot x_0 + t \cdot x_1$ is linearly interpolated between x_0 and x_1 . In practice, x_0 is drawn from an isotropic Gaussian $\mathcal{N}(0, I)$ and x_1 from the training data.

The scalability and inference efficiency of rectified flows have driven their adoption across modern, state-of-the-art generative models. In image generation, Stable Diffusion 3 [43] and FLUX [16] combine rectified flow with multimodal transformer (MMDiT) architectures to achieve strong prompt alignment and visual fidelity, while InstaFlow [117] demonstrates that ReFlow and distillation can compress Stable Diffusion inference into a one-step evaluation. Rectified flow has since been extended to video generation, where the straight-trajectory inductive bias proves beneficial for modeling the higher-dimensional temporal domain: Open-Sora [212], CogVideoX [198], and Wan [175] all adopt rectified flow backbones and represent the current state-of-the-art in video synthesis.

2.4 Conditioning Mechanisms

The ability to control the content, style, and structure of generated outputs is critical to the real-world applicability of generative models and has motivated a rich body of research on conditioning mechanisms. While text-conditioned models [16, 43, 136, 140, 147, 149, 149] enable high-level semantic control through natural language, text alone is insufficient for specifying fine-grained visual attributes such as exact person and garment identity, precise geometry and motion, or spatial layout. This limitation has spurred significant efforts to develop richer conditioning mechanisms to incorporate additional modalities, such as images, depth maps, poses, etc.

Classifier & Classifier-Free Guidance: Classifier guidance [37] was among the first general-

purpose conditioning mechanisms for diffusion models, augmenting the score function with the log-likelihood gradient of a pretrained classifier, scaled by a constant w to control conditioning strength. Classifier-free guidance (CFG) [77] has since emerged as the preferred alternative, eliminating the need for a separate classifier by instead linearly interpolating between conditional and unconditional score estimates from a single jointly-trained model:

$$\tilde{v}_\theta(x_t, t, c) = v_\theta(x_t, t, \emptyset) + w \cdot (v_\theta(x_t, t, c) - v_\theta(x_t, t, \emptyset)) \quad (7)$$

where c represents the conditioning signal(s), $\emptyset$ represents the unconditional (null) signal(s), and $w > 1$ amplifies adherence to the condition.

CFG has been extended in several directions: Instruct-Pix2Pix [20] applies dual guidance over independent text and image conditioning signals, enabling flexible control over each modality independently. More recently, CFG distillation [16] has been proposed to bake a fixed guidance scale directly into the model weights during training, removing the need to run two forward passes at inference time and halving the associated computational overhead.

Subject Finetuning: One family of approaches performs subject-specific finetuning to bind a concept to the model’s learned representation space. Textual Inversion [44] pioneered this direction by optimizing a single new text embedding token to represent a subject, leaving all model weights unchanged. DreamBooth [148] extended this by finetuning the full diffusion model on a small set of subject images while using a class-preservation loss to prevent catastrophic forgetting, achieving significantly higher subject fidelity. Related methods extend this idea to video [88, 188] to achieve high-fidelity image conditioning. However, the requirement for per-subject finetuning limits practical applicability, as it incurs time and computational overhead during inference.

Lightweight Conditioning Modules: Another paradigm adapts pretrained models to new conditioning signals by training lightweight auxiliary modules while keeping the backbone frozen, avoiding the per-subject overhead of finetuning approaches. LoRA [81] injects trainable low-rank

decomposition matrices into the attention layers of a pretrained model, enabling efficient style or concept adaptation with a small parameter budget. ControlNet [207] attaches a trainable copy of the U-Net encoder to the frozen backbone via zero initialized convolutions, enabling spatially-aligned conditioning from signals such as edges, depth maps, and human pose. T2I-Adapter [130] achieves similar spatial conditioning with a lighter-weight architecture, using a small standalone encoder rather than a full U-Net copy. IP-Adapter [199] introduces decoupled cross-attention to inject image prompt features alongside text features, enabling reference-image conditioning without modifying the backbone. Uni-ControlNet [211] unifies local spatial controls (e.g. edge maps, depth) and global semantic controls (e.g. CLIP embeddings) within a single two-adapter framework, enabling flexible composition of multiple conditioning signals simultaneously. The benefit of these approaches is two-fold: preserving pretrained knowledge of the frozen T2I model and reducing the computational demands of finetuning.

Chapter 3

DREAMPOSE: FASHION IMAGE-TO-VIDEO SYNTHESIS VIA STABLE DIFFUSION

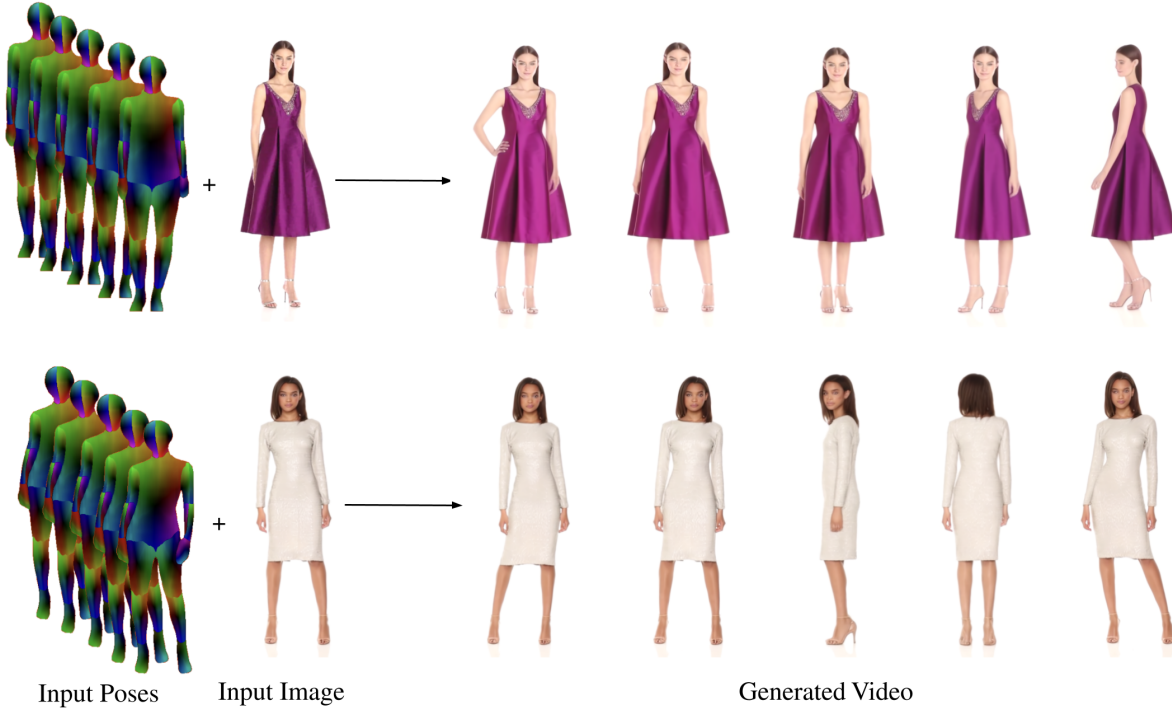


Figure 1: Given an image of a person and a sequence of body poses, DreamPose synthesizes a photorealistic video.

The project in this chapter was completed in collaboration with Aleksander Holynski, Ting-Chun Wang, and Ira Kemelmacher-Shlizerman. It was originally published in ICCV 2023 [88]. The analysis and comparisons in this chapter are based on the existing state-of-the-art during that time.

Introduction

Fashion photography is incredibly prevalent online, from social media platforms to online retail sites. Unfortunately, these still photographs are limited in the information they convey, and fail to capture many of the crucial nuances of a garment, such as how it drapes and flows when worn. Fashion *videos*, on the other hand, do showcase all these details, and for this reason are highly informative

for consumer decision-making. Despite this benefit, however, these videos are a relatively rare commodity.

In this chapter, we introduce DreamPose, a method that turns fashion photographs into realistic, animated videos, using a driving pose sequence. Our method is a diffusion video synthesis model based upon Stable Diffusion [147]. Given one or more images of a human and a pose sequence, DreamPose generates a high-quality video of the input subject following the pose sequence (Figure 1).

This is a challenging task in several ways. While image diffusion models have shown impressive, high-quality results [141, 147, 149], video diffusion models have yet to achieve the same quality of results and are often limited to “textural” motion or cartoon-like appearance [69, 75, 78, 158, 196]. Moreover, existing video diffusion models suffer from poor temporal consistency, motion jitter, lack of realism, and the inability to control the motion or detailed object appearance in the target video. This is partly because existing models are primarily conditioned on text, as opposed to other conditioning signals (e.g., motion) which may offer more fine-grained control. In contrast, our image-and-pose conditioning scheme allows for greater appearance fidelity and frame-to-frame consistency.

Our model is finetuned from an existing pretrained image diffusion model, which already effectively models the distribution of natural images. When using such a model, the task of image animation can effectively be simplified to finding the subspace of natural images consistent with the conditioning signals. To accomplish this, we redesign the encoder and conditioning mechanisms of the Stable Diffusion [147] architecture, in order to enable aligned-image and unaligned-pose conditioning. Further, we propose a two-stage finetuning scheme that consists of finetuning both UNet and VAE from one or more input images.

To summarize, our contributions include: (1) *DreamPose*: an image-and-pose conditioned diffusion method for still fashion image animation that achieves photorealistic results on a diverse range of garment patterns and shapes, (2) a simple, yet effective, pose conditioning approach that greatly

improves temporal consistency across frames, (3) a split CLIP-VAE encoder that increases the output fidelity to the conditioning image, (4) a finetuning strategy that effectively balances image fidelity and generalization to new poses.

3.1 Related Works

Still Image Animation. Still image animation refers to the task of generating a video from one or more input images. Existing, non-diffusion approaches often consist of multiple separate networks, such as for predicting the background [157, 186, 210], motion representation [79, 155–157, 180, 210], occlusion maps [156, 157, 186, 210], or depth maps [12]. However, multiple networks require separate training for each stage and potentially unavailable or imperfect ground-truth intermediate data, such as motion or depth. Especially with large and complex motion, these ground-truth estimates are harder to derive and more error-prone. Several more recent papers explore end-to-end single-network approaches, such as by merging optical flow and warping [142], replacing motion estimation networks entirely with a cross-attention modules [125], or generating animatable 3D humans using a NeRF representation [80].

Pose-Guided Human Image Synthesis. Body pose control is essential to many human-centric applications, including reposing, virtual try-on, image editing, and animation. As such, a large body of work has explored explicit pose guidance for human image synthesis. Early approaches based on Generative Adversarial Networks (GANs) rely on dense correspondences, such as optical flow, to align features between source and target poses [7, 33, 105, 145, 206, 218]. While effective for moderate pose changes, these methods often struggle with large deformations, occlusions, and preserving person and garment appearance. More recent methods replace explicit flow with attention-based mechanisms, leveraging self- and cross-attention to implicitly model correspondences between source appearance and target pose [27, 115, 145]. These approaches improve robustness to complex pose variations and enable more flexible feature warping. Relatively few diffusion-based works exist with pose guidance. Concurrent work PIDM [14] generates pose-conditioned human images,

but is designed for single-image pose transfer, and therefore not optimized for temporal consistency.

In contrast, our approach uses a unique multi-pose input representation to enhance temporal smoothness. Moreover, by leveraging pretrained Stable Diffusion [147], our method can be finetuned in 2 days with 2 A100 GPU’s using a small dataset of 323 videos. In contrast, PIDM [14] is trained from scratch on 101,966 Deep Fashion [51] image pairs with 4 A100s for 26 days.

Image-Guided Diffusion Models. While text conditioning enables high-level control in diffusion models [140, 147, 149], it is insufficient for specifying fine-grained information about the *exact* person identity and garment appearance. As a result, a large body of work has explored explicit image guidance for diffusion models. A common paradigm is to adapt pretrained text-to-image diffusion models, such as Stable Diffusion [147], for image-conditioned generation [11, 20, 45, 128, 148, 183]. Notably, DreamBooth [148] introduced subject-specific finetuning, where a unique token is learned to represent a specific subject, enabling high-fidelity personalization from a small set of images. Other works incorporate text to edit the appearance of existing images [20] and videos [128, 188]. In the fashion domain, DiffFashion [21] aims to edit a clothing item by transferring the style of a reference image and a semantic mask. PIDM [14] uses a separate textural encoder to encode image textures. Similar to PIDM, our image conditioning approach directly incorporates image embeddings in the cross-attention layers of the UNet, but these image embeddings come from a mixture of two pretrained encoders: CLIP and VAE. Moreover, with our subject-specific finetuning method, we achieve superior person identity preservation.

Diffusion Models for Video Synthesis. Many text-to-video diffusion models rely on adapting text-to-image diffusion models for video synthesis [69, 71, 75, 78, 158, 196]. While the results are promising, these methods still struggle to match the realism of text-to-image models. Quality is largely hindered due to the new challenges introduced by video synthesis, such as maintaining temporal consistency across frames and generating realistic motion. Training a video diffusion methods from scratch requires expensive computational resources, huge training datasets, and extensive training time. In contrast, an emerging family of works instead finetune pre-trained

latent image-to-video diffusion models [18, 62, 71, 188]. Concurrently to our work, Tune-A-Video fine-tunes a text-to-image pretrained diffusion model for text-and-image conditioned video generation [188]. However, like earlier video diffusion methods, Tune-A-Video’s results exhibit textural flickering and structural inconsistencies. Our work aims to address these issues in order to synthesize realistic human and fabric motion.

3.2 Stable Diffusion

Stable Diffusion [147] is a latent diffusion model that operates in the encoded latent space of an autoencoder (VAE [96]), therefore reducing computational complexity, with only minimal sacrifice to perceptual quality. Stable Diffusion is composed of two models: a variational autoencoder and a denoising UNet. The autoencoder consists of an encoder $\mathcal{E}$ that distills a frame x into a compact latent representation, $z = \mathcal{E}(x)$, and a decoder $\mathcal{D}$ that reconstructs the image from its latent representation, $x' = \mathcal{D}(z)$. During training, the latent features z are diffused in T timesteps by a deterministic Gaussian process to produce noisy features $\tilde{z}_T$, indistinguishable from random noise. In order to recover the original image, a time-conditioned UNet is trained to iteratively predict the noise of the latent features corresponding to each timestep $t \in \{1, \dots, T\}$. The UNet ϵ_θ objective function is:

$$L_{DM} = \mathbb{E}_{z, \epsilon \in \mathcal{N}(0,1)} [\|\epsilon - \epsilon_\theta(\tilde{z}_t, t, c)\|_2^2] \quad (8)$$

where c represents the embeddings of conditional information, such as text, image, segmentation mask, etc. In the case of text-to-image Stable Diffusion, c is obtained using a CLIP text encoder [138]. Finally, the predicted denoised latents z' are decoded to recover the predicted image $x' = \mathcal{D}(x')$.

Classifier-free guidance is a sampling mechanism that pushes the predicted noise distribution towards the conditional distribution through an implicit classifier [77]. This is practically achieved by dropout, a training scheme that, with a random probability, replaces real conditioning inputs

with null inputs ($\emptyset$). During inference, the conditional prediction is used to guide the unconditional prediction towards the conditional, using a guidance scalar weight s :

$$\epsilon_{\theta} = \epsilon_{\theta}(\tilde{z}_t, t, \emptyset) + s \cdot (\epsilon_{\theta}(\tilde{z}_t, t, c) - \epsilon_{\theta}(\tilde{z}_t, t, \emptyset)) \quad (9)$$

Since its release, Stable Diffusion and its pretrained checkpoints have been used by many for various image generation tasks [11, 20, 148]. Like these methods, our work leverages a pretrained Stable Diffusion model with subject-specific finetuning.

3.3 Method

Our method aims to produce photorealistic animated videos from a single image and a pose sequence. To achieve this, we finetune a pretrained Stable Diffusion model on a collection of fashion videos. This involves adapting the architecture of Stable Diffusion (which is a text-to-image model) to accept additional conditioning signals (image and pose) and to output temporally consistent content that can be viewed as a video. In the following sections, we begin by describing the architectural modifications (Section 3.3.2). Then, we describe the two-stage finetuning strategy (Section 3.3.3). Finally, we describe the inference process of generating an animated video from a still image, which involves a novel formulation of classifier-free guidance (Section 3.3.4).

3.3.1 Problem Formulation

Given input image x_0 and pose sequence $\{p_1, \dots, p_N\}$, DreamPose generates a video $\{x'_1, \dots, x'_N\}$, where x'_i is the i -th predicted frame corresponding to input pose p_i . Our method relies on a pretrained latent diffusion model [147], which is conditioned on an input image and a sequence of poses. At inference time, we generate each frame independently through a standard diffusion sampling procedure: starting with uniformly distributed Gaussian noise, the diffusion model is repeatedly queried with both conditioning signals to gradually denoise the noisy latent to a plausible estimate.

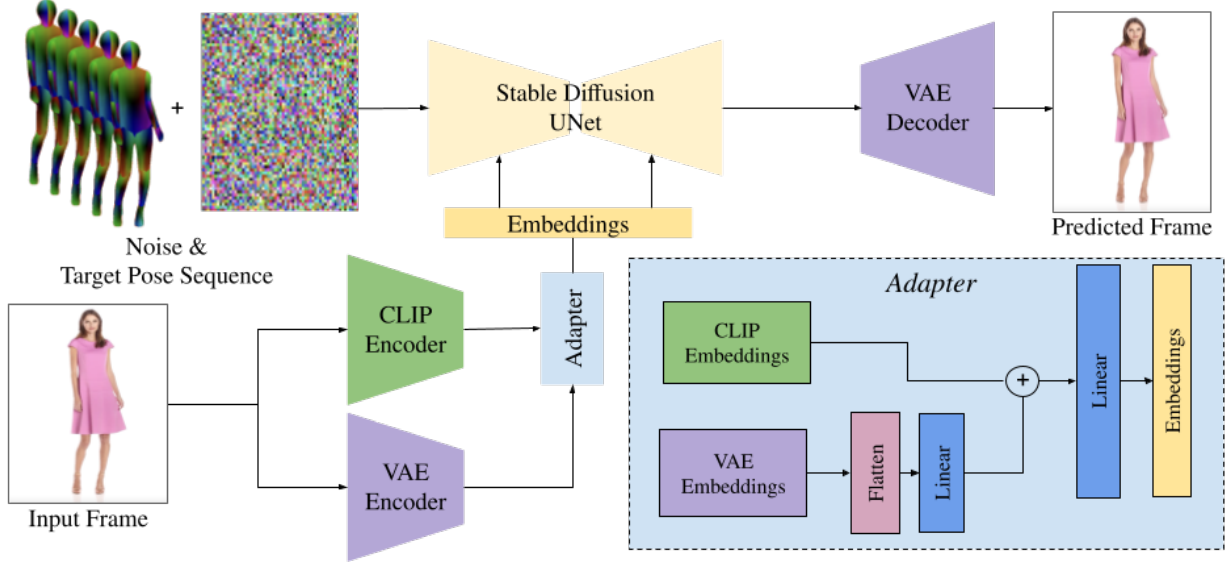


Figure 2: Architecture Overview. We modify the original Stable Diffusion architecture in order to enable image and pose conditioning. First, we replace the CLIP text encoder with a dual CLIP-VAE image encoder and adapter module (shown in the blue box). The adapter module jointly models and reshapes the pretrained CLIP and VAE input image embeddings. Then, we concatenate the target pose representation, consisting of 5 consecutive poses surrounding the target pose, to the input noise. During training, we finetune the denoising UNet and our Adapter module on the full dataset and further perform subject-specific finetuning of the UNet, Adapter, and VAE decoder on a single input image.

Finally, the predicted denoised latent z'_i is decoded to produce the predicted video frame $x'_i = \mathcal{D}(z'_i)$.

3.3.2 Architecture

The DreamPose model is a pose- and image-conditioned image generation model that modifies and finetunes the original text-to-image Stable Diffusion model for the purpose of image animation. The objectives of image animation include: (1) faithfulness to the provided input image, (2) visual quality, and (3) temporal stability across generated frames. As such, DreamPose requires an image conditioning mechanism that captures the global structure, person identity, and fine-grained details of the garment, as well as a method to effectively condition the output image on target pose while also enabling temporal consistency between independently sampled output frames. We describe our approach to achieving these goals in the sections below. A diagram of our architecture can be found

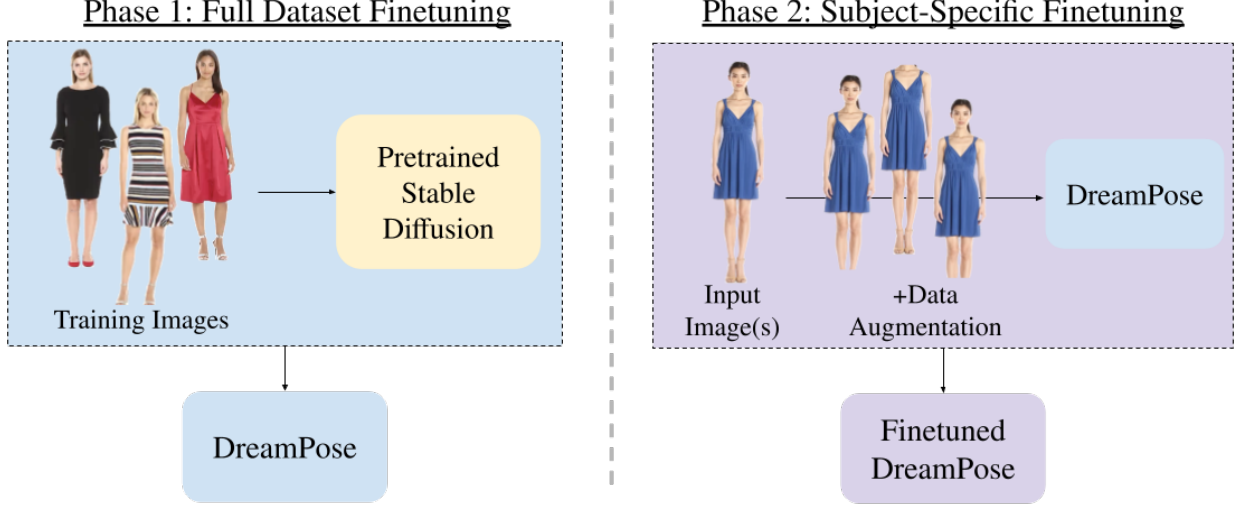


Figure 3: Two-Phase Finetuning Scheme. In the first phase, our method finetunes the modified Stable Diffusion model on the full dataset. In the second phase, the model is further finetuned on a single subject image.

in Figure 2. Full implementation details are provided in the appendix.

Split CLIP-VAE Encoder. In many prior works, such as InstructPix2Pix [20], image conditioning signals are often concatenated with the input noise to the denoising U-Net. While this is effective for conditioning signals that are spatially aligned with the desired output image, in our case, our network aims specifically to produce images which are *not* spatially aligned with the input image. As such, we explore alternative approaches for image conditioning. In particular, we implement image conditioning by replacing the CLIP text encoder with a custom conditioning adapter that combines the encoded information from pretrained CLIP image and VAE encoders.

A crucial objective when finetuning from a pretrained network is to make training gradients as meaningful as possible by making the input signals as similar as possible to those used in the original network training. This helps avoid regressions in network performance during finetuning, or loss of learned priors, which can come from noisy gradients (e.g., if the network does not know how to parse new forms of input signals). For this reason, most diffusion-based finetuning schemes [20, 207] will retain all original conditioning signals, and will initialize network weights

that interact with new (previously unseen) conditioning signals to zero.

For our purposes, given that Stable Diffusion is conditioned on CLIP embeddings of text prompts, and CLIP encodes both text and images to a shared embedding space, it may seem natural to simply replace the CLIP conditioning with the embedding derived from the conditioning image. While this would in theory pose a very small change to the original architecture and allow for image conditioning with minimal finetuning, we find that in practice that CLIP image embeddings alone are insufficient for capturing fine-grained details in the conditioning image. So, we instead additionally input the encoded latent embeddings from Stable Diffusion’s VAE. Adding these latent embeddings as conditioning has the added benefit of coinciding with the output domain of the diffusion model.

Since the architecture does not support VAE latents as a conditioning signal by default, we add an adapter module $\mathcal{A}$ that combines the CLIP and VAE embeddings to produce one embedding that is used in the network’s usual cross-attention operations. This adapter blends both the signals together and transforms the output into the typical shape expected by the cross-attention modules of the denoising U-Net. Initially, the weights corresponding to the VAE embeddings are set to zero, such that the network begins training with only the CLIP embeddings (as mentioned before, to mitigate network “shock” in training). We define the final image conditioning signal c_I as:

$$c_I = \mathcal{A}(c_{\text{CLIP}}, c_{\text{VAE}}) \quad (10)$$

Modified UNet. Unlike the image conditioning, the pose conditioning *is* image-aligned. As such, we concatenate the noisy latents $\tilde{z}_i$ with a target pose representation c_p . To account for noise in the poses (which are estimated from real videos using an off-the-shelf network [65]) and to maximize temporal consistency in the generated frames, we set c_p to consist of five consecutive pose frames: $c_p = \{p_{i-2}, p_{i-1}, p_i, p_{i+1}, p_{i+2}\}$. We observe that individual poses are prone to frame-to-frame jitter, but training the network with a set of consecutive poses increases the overall motion smoothness and temporal consistency. Architecturally, we modify the UNet input layer to take in 10 extra input

channels, initialized to zero, while the original channels corresponding to the noisy latents are unmodified from the pretrained weights.

3.3.3 Subject-Specific Finetuning

For initialization, the unmodified Stable Diffusion layers are initialized from a pretrained text-to-image Stable Diffusion checkpoint, except for the CLIP image encoder which is loaded from a separate pretrained checkpoint [138, 147]. As mentioned previously, the novel layers are initialized such that initially the new conditioning signals do not contribute to the network output.

Following initialization, DreamPose is finetuned in two stages (shown in Figure 3). The first phase finetunes the UNet and adapter module on the full training dataset in order to synthesize frames consistent with an input image and pose. The second phase refines the base model by finetuning the UNet and adapter module, then the VAE decoder, on one or more subject-specific input image(s) to create a subject-specific custom model used for inference.

Similar to other image-conditional diffusion methods [128, 137, 148], we find that sample-specific finetuning is essential to preserving the identity of the input image’s person and garment, as well as maintaining a consistent appearance across frames. However, simply training on a single frame and pose pair quickly leads to artifacts in the output videos, such as texture-sticking. To prevent this, we augment the image-and-pose pair at each step, such as by adding random cropping.

We also find that finetuning the VAE decoder is crucial for recovering sharper, more photorealistic details in the synthesized output frames. Refer to Figure 4 for an ablated comparison. Furthermore, we show in Figure 6 that even single image finetuning of the decoder allows increased pose guidance, without sacrificing the person identity or appearance.



Figure 4: Ablation of VAE Finetuning. We find that finetuning the VAE decoder, in addition to the UNet, during the subject-specific finetuning phase yields more photorealistic details and reduces high-frequency noise, compared to finetuning the UNet alone.

3.3.4 Pose and Image Classifier-Free Guidance

At inference time, we generate a video frame-by-frame from a single input image and a sequence of poses using the subject-specific model. We modulate the strength of image conditioning c_I and pose conditioning c_p during inference using dual classifier-free guidance [20]. The dual classifier-free guidance equation is modified from Equation 3.2 to be controlled by two guidance weights, s_I and s_p , which rule how similar the output image is to the input image c_I and input pose c_p , respectively:

$$\begin{aligned}
 \epsilon_{\theta}(z_t, c_I, c_p) &= \epsilon_{\theta}(z_t, \emptyset, \emptyset) \\
 &+ s_I(\epsilon_{\theta}(z_t, c_I, \emptyset) - \epsilon_{\theta}(z_t, \emptyset, \emptyset)) \\
 &+ s_p(\epsilon_{\theta}(z_t, c_I, c_p) - \epsilon_{\theta}(z_t, c_I, \emptyset))
 \end{aligned} \tag{11}$$

In Figure 6, we show the effect of varying the classifier free guidance weights (s_I, s_p). A large s_I ensures high appearance fidelity to the input image, while a large s_p ensures alignment to the input pose. In addition to strengthening our pose and image guidance, the decoupled classifier-free guidance prevents overfitting to the one input pose after subject-specific finetuning.



Figure 5: Qualitative comparisons of our method versus MRAA [157], TPSMM [210], and PIDM [14]. Our method produces more photorealistic details aligned with the input frame, such as garment folds, fine-grain patterns, and face identity. Our method can also better handle challenging cases, including pattern synthesis in occluded regions.

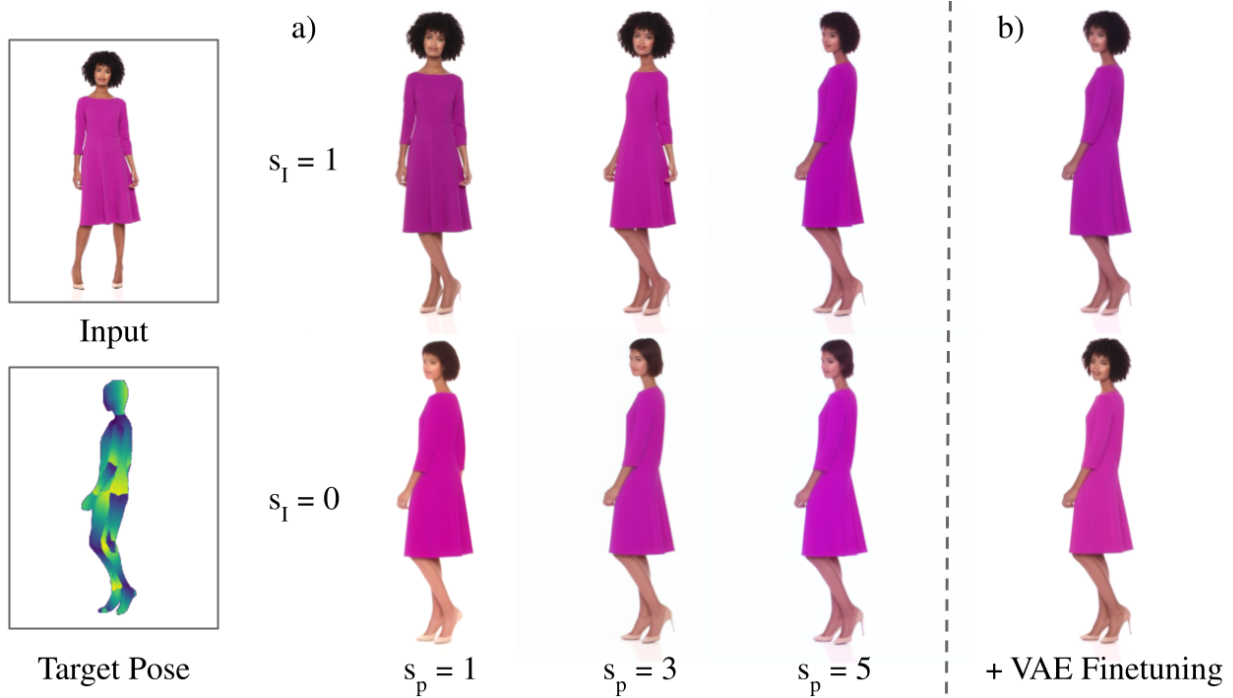


Figure 6: a) Pose and Image Classifier-Free Guidance. We demonstrate the effect of the relative weight between image and pose guidance weights, s_I and s_P . Results shown are after subject-specific finetuning. b) VAE decoder finetuning improves the appearance and person identity, even with larger relative pose guidance.

3.3.5 Dataset

We train and test our method on the UBC Fashion dataset [203]. We follow the provided train/test split of 339 training and 100 test videos. Each video has a frame rate of 30 frames/second and is approximately 12 seconds long. During training, we randomly sample pairs of frames from the training videos. We compute poses with DensePose [65].

3.4 Experiments

DreamPose is capable of generating state-of-the-art fashion videos from still images. In Figure 7, we showcase frames synthesized by DreamPose from a variety of input images and poses from the UBC Fashion dataset [203]. DreamPose handles diverse human and clothing appearances well, even

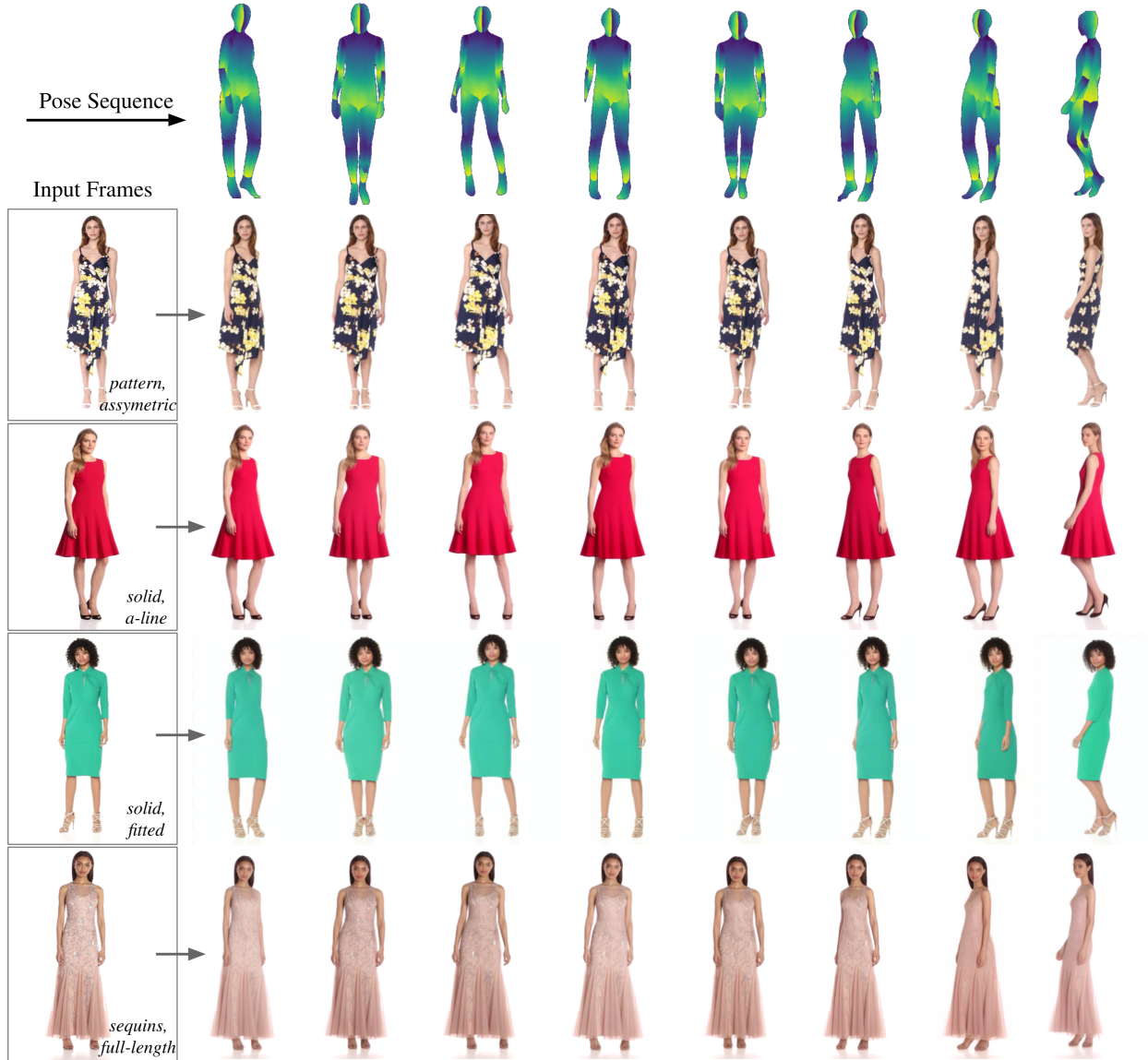


Figure 7: Qualitative Results. We showcase the results of our method on a variety of input frames and poses. DreamPose is capable of synthesizing photorealistic video frames consistent with a diverse range of patterns, fabric types, person identities, clothing shapes, and viewpoints.

from different viewpoints and loose garments. Moreover, for a given subject image, DreamPose can be conditioned on driving poses derived from a different video, as shown in the appendix Figure 11. We also show additional results on images from the DeepFashion dataset [51] in the appendix.

	L1 ↓	SSIM ↑	VGG ↓	LPIPS ↓	FID ↓	FVD (16f) ↓	AED ↓
MRAA [157]	0.0857	0.749	0.534	0.212	23.42	253.65	0.0139
TPSMM [210]	0.0858	0.746	0.547	0.213	22.87	247.55	0.0137
PIDM [14]	0.1098	0.713	0.629	0.288	30.279	1197.39	0.0155
Ours	0.0256	0.885	0.235	0.068	13.04	238.75	0.0110

Table 1: Quantitative comparisons of our method with MRAA, TPSMM, PIDM, and our method. Bolded values indicate best scores in each column.

3.4.1 Comparisons

We compare DreamPose quantitatively and qualitatively to two publicly available state-of-the-art conditional video synthesis methods, Motion Representations for Articulated Animation (MRAA) [157] and Thin-Plate Spline Motion Model (TPSMM) [210], and a concurrent diffusion-based pose transfer method, PIDM [14]. We train all methods from scratch on the UBC Fashion Dataset [203], using the provided training scripts and configurations. For evaluating MRAA [157] and TPSMM [210], we use the provided test scripts in the “AVD” mode.

Quantitative Analysis. We present our quantitative analysis in Table 1. We test all models on the UBC Fashion test set, consisting of 100 unique fashion videos, at 256px resolution [203]. For each video, we extract 50 frames for testing, where they are at least 50 frames away from the input frame. The full DreamPose model quantitatively outperforms all three methods in all metrics: L1, SSIM [181], VGG [87], LPIPS [208], FID [73], FVD [171], and Average Euclidean Distance (AED). Note that while PIDM produces realistic images, it struggles to preserve clothing and person identity, as indicated by the quantitative results. Moreover, there is poor temporal consistency frame-to-frame, resulting in a large FVD [171] score.

Qualitative Analysis. We qualitatively compare our method to MRAA, TPSMM, and PIDM in Figure 5. With these other methods, note that the person identity, fabric folds, and fine patterns are lost in new poses, whereas DreamPose accurately retains those details. Plus, during large pose changes, MRAA may produce disjointed or blurry limbs. Compared to PIDM, DreamPose produces higher-fidelity results, in terms of both face identity and clothing patterns. While PIDM synthesizes

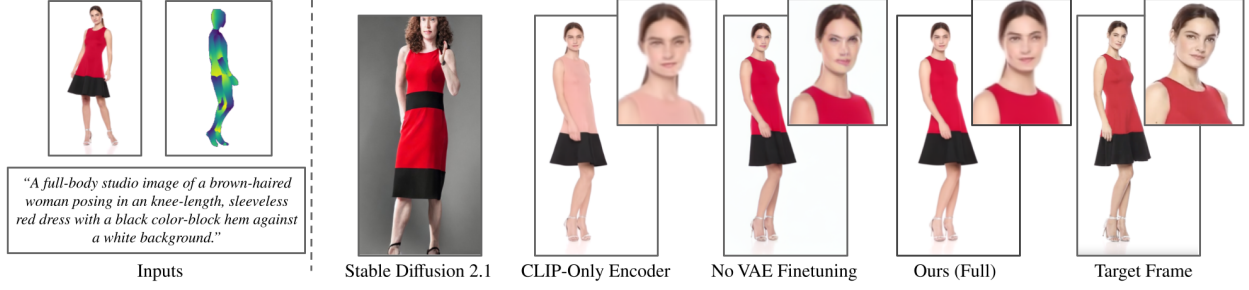


Figure 8: Qualitative Ablation of Image Conditioning. We compare results of the original text-to-image Stable Diffusion model, our model with CLIP-only image embeddings without finetuning, our model with CLIP-VAE image encoder embeddings without VAE finetuning, and our full model.

realistic faces, they do not necessarily align with the identity of the source person. Moreover, we find that both the identity and the dress appearance vary frame-to-frame. As such, PIDM does not work well as-is for video synthesis. We provide additional comparisons to PIDM on the DeepFashion dataset [51] in the appendix. Furthermore, we conduct a user study and provide the results in the supplementary material.

3.4.2 Ablation Studies

We perform comparisons of ablated versions of our method to verify our design choices. Namely, we compare five variants:

- (1) $\text{Ours}_{\text{CLIP}}$: We use a pretrained CLIP image encoder, instead of our dual CLIP-VAE encoder
- (2) $\text{Ours}_{\text{No-VAE-FT}}$: We do subject-specific finetuning of the UNet only, not the VAE decoder
- (3) $\text{Ours}_{1\text{-pose}}$: We concatenate only one target pose, instead of 5 consecutive poses, to the noise.
- (4) $\text{Ours}_{\text{smooth}}$: Like ablation $\text{Ours}_{1\text{-pose}}$, but apply temporal smoothing to the output frames.
- (5) $\text{Ours}_{\text{full}}$: Our full model, including subject-specific VAE finetuning, CLIP-VAE encoder, and 5-pose input.

	L1 ↓	SSIM ↑	VGG ↓	LPIPS ↓
Ours _{CLIP}	0.025	0.882	0.247	0.070
Ours _{No-VAE-FT}	0.025	0.897	0.210	0.057
Ours _{1-pose}	0.019	0.899	0.208	0.056
Ours _{smooth}	0.767	0.758	0.502	0.202
Ours _{full}	0.019	0.900	0.207	0.056

Table 2: Quantitative comparison of ablated versions of our method. Note that the single-pose version of our method, although achieving similar numerical results, fails to achieve the motion smoothness as the full method.

Quantitative Comparison. For each ablated version, we compute the L1, SSIM, VGG, and LPIPS for 100 predicted video frames selected from each of the 100 test videos of the UBC Fashion dataset [203]. As shown in Table 2, our full model outperforms the ablated versions in all four metrics.

Qualitative Comparison. We visually show the effectiveness of our full method in Figure 8. We compare results from the original text-conditioned Stable Diffusion model, our method with only a CLIP image encoder, our method with CLIP-VAE encoder, and our full method with CLIP-VAE encoder and subject-specific VAE finetuning.

The original Stable Diffusion model with text-only conditioning via CLIP text encoder is unable to preserve rich details of the garment or person identity. Simply replacing the text encoder with a CLIP image encoder helps capture most image details, but there is still information loss about the appearance. Subject-specific finetuning of the UNet, similar to DreamBooth [148], is critical to preserving photorealistic details in the face and garment. Furthermore, we find that also finetuning the VAE decoder on the input image greatly improves the sharpness of these details and does not lead to overfitting to the input pose. Lastly, with only a single input pose, there is noticeable flickering of the subject’s shape, especially around the feet and hair. Please refer to video qualitative comparisons of each ablated version on our project page.



Figure 9: Examples of failure cases of our method. Our method may merge limbs into underlying fabric textures (left), hallucinate features (middle), or predict front-facing person instead of a back-facing person (right).

3.5 Limitations and Future Work

In Figure 9, we show failure cases of our method. On rare occasions, we observe limbs disappearing into the fabric, hallucinated dress features, and directional misalignment when the target pose is facing backwards. We suspect that some of these failures could be alleviated with improved pose estimation, a larger dataset, or a segmentation mask. Additionally, while our method produces realistic results on most plain and simple-patterned fabrics, some of our results present minor flickering behavior on large and complex patterns. Achieving better temporal consistency on such patterns, ideally without subject-specific finetuning, is left to future work. Lastly, similar to other

diffusion models, our finetuning and inference times are slow compared to GAN or VAE methods. Finetuning the model on a specific subject takes approximately 10 minutes for the UNet and 20 minutes for the VAE decoder, in addition to an 18 second per-frame rendering time.

Conclusion

In this paper, we presented DreamPose, a novel diffusion-based method for still fashion image animation. Given a single image and pose sequence, we demonstrate how our method generates photorealistic fashion videos from only a single image – animating a diverse range of fabrics, patterns, and person identities.

Acknowledgments

This work was supported by NVIDIA and the UW Reality Lab, Meta, Google, OPPO, and Amazon.

Appendix

3.5.1 Implementation Details

Our experiments are trained on two NVIDIA A100 GPU’s with resolution 512x512. In our first phase of training, we finetune our base model UNet on the full training dataset for a total of 5 epochs at a learning rate of $5e-6$. We use an effective batch size of 16 (through 4 gradient accumulation steps). We implement a dropout scheme where null values replace the pose input 5% of the time, the input image 5% of the time, and both input pose and input image 5% of the time during training. We further finetune the UNet on a specific sample frame for another 500 steps with a learning rate of $1e-5$ and no dropout. Lastly, we finetune the VAE decoder only for 1500 steps with a learning rate of $5e-5$. During inference, we use a PNDM sampler for 100 denoising steps [113].

3.5.2 User Studies

We conducted two user studies involving 50 distinct Amazon Mechanical Turk workers to compare our method with state-of-the-art image animation approaches [157] [210] and evaluate the quality of our videos. In both surveys, workers evaluated results corresponding to 50 unique input images from the test set of the UBC Fashion dataset [203].

In the first user study, workers were asked their pair-wise preferences between our method and one of the other methods. For each input image, the workers were shown two videos: one containing the input image, our resulting video, and the MRAA resulting video and the other containing the input image, our resulting video, and the TPSMM resulting video. The ordering of our video and other video (MRAA or TPSMM) was randomized for each question. For each video, workers selected their preference between the videos. The results are shown in Table 3. Overall, the workers had a preference for our method over MRAA and TPSMM.

In the second user study, workers were asked to rate our videos and TPSMM videos on a scale of 0

to 5, where 0 corresponds a video that does not match the input image at all and 5 corresponds to a realistic animation of the input image. During training, workers were shown a video of a different dress for as an example of a "0" rating and a ground-truth video of the input image as an example of the "5" rating. The results are shown in Figure 10. Our videos achieved higher scores for image similarity and quality than TPSMM and 85% of users rated the results of our method a 3 or higher.

	# Responses	Total Responses	(%)
Ours > MRAA [157]	1637	2500	(65%)
Ours > TPSMM [210]	1417	2500	(57%)

Table 3: Results of User Study #1: Workers choose between pairs of videos corresponding to input images, either our result vs. MRAA result or our result vs. TPSMM result. Overall, participants preferred our method over both MRAA and TPSMM in terms of quality and similarity to the input image.

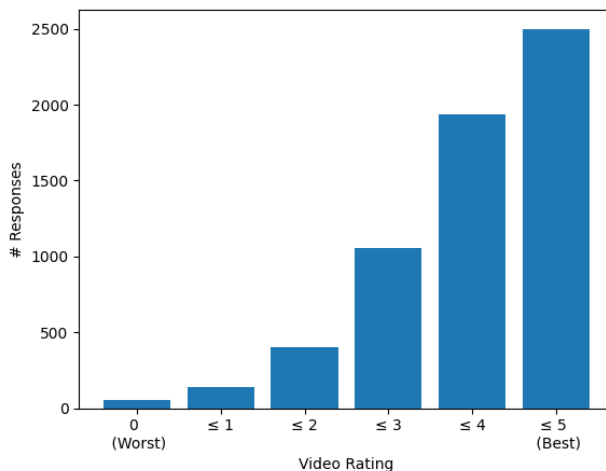


Figure 10: Results of User Study #2: Amazon Mechanical Turk worker ratings of our videos from 0 (video does not match input image) to 5 (video is a realistic animation of the input image). Overall, 85% of workers rated our method a 3 or higher.

3.5.3 Different Videos for Source Person and Driving Pose Sequence

We show in Figure 11 that DreamPose can animate an input image using motion from a video containing a different person and garment identity. As such, our method is applicable in practice when ground-truth motion is unavailable.



Figure 11: Qualitative results for conditioning on subject and pose from different videos.

3.5.4 Multiple Input Frames

While DreamPose demonstrates high-quality results with only a single input image, DreamPose can also be finetuned with an arbitrary number of input images of a subject. We showcase the results of training with multiple input images in Figure 12. We find that additional input images of a subject increase the quality and viewpoint consistency.

3.5.5 Deep Fashion Results

We demonstrate the effectiveness of our method on a popular dataset, DeepFashion, in Figure 13 [14, 51]. Although trained exclusively on the UBC Fashion video dataset, DreamPose performs well on unseen retail images, even to new backgrounds, model identities, accessories, and patterns.



Figure 12: Results after training with 1, 3, 5, and 7 input images. Increasing the number of input frames improves fidelity of pose, facial identity, and color.

3.5.6 Application to Pose Transfer

While adapted for image-to-video synthesis, DreamPose is also an effective pose transfer tool. In Figure 14, we compare DreamPose to two state-of-the-art pose transfer models: DynaST [115] and PIDM [14]. Our method is better able to preserve fine-details, such as shoe appearance, hemline, and face identity, than DynaST or PIDM.



Figure 13: DreamPose results on unseen samples from the DeepFashion dataset [51]. Despite being trained exclusively on the UBC Fashion Dataset, our method generalizes to new garments and model identities after subject-specific finetuning of the base model.



Figure 14: Comparison of Pose Transfer Results. We compare our method to two state-of-the-art pose transfer methods, DynaST [115] and PIDM [14].

Chapter 4

FASHION-VDM: VIDEO DIFFUSION MODEL FOR VIRTUAL TRY-ON



Figure 15: Fashion-VDM. Given an input garment image and a person video, Fashion-VDM generates a video of the person virtually trying on the given garment, while preserving their original identity and motion.

The project in this chapter was completed in collaboration with Yingwei Li, Nan Liu, Luyang Zhu, Innfarn Yoo, Andreas Lugmayr, Chris Lee, and Ira Kemelmacher-Shlizerman during my internship at Google. It was originally published in SIGGRAPH Asia 2024 [90]. The analysis and comparisons in this chapter are based on the existing state-of-the-art during that time.

Introduction

With the popularity of online clothing shopping and social media marketing, there is a strong demand for virtual try-on methods. Given a garment image and a person image, virtual try-on aims to show how the person would look wearing the given garment. In this paper, we explore *video* virtual try-on, where the input is a garment image and person video. The benefit of a video virtual

try-on (VVT) experience is that it would depict how a garment looks at different angles and how it drapes and flows in motion.

VVT is a challenging task, as it requires synthesizing realistic try-on frames from different viewpoints, while generating realistic fabric dynamics (e.g. folds and wrinkles) and maintaining temporal consistency between frames. Additional difficulty arises if the person and garment poses vary significantly, as this creates occluded garment and person regions that need to be hallucinated. Another challenge is the scarcity of try-on video data. Perfect ground truth data (i.e. two videos of different people wearing the same garment and moving in the exact same way) is difficult and expensive to acquire. In general, available human video data, such as UBC Fashion [203], are much more scarce and less diverse than image data, such as LAION 5B [152].

Past approaches to virtual try-on typically leverage dense flow fields to explicitly warp the source garment pixels onto the target person frames [38, 39, 86, 185, 213]. However, these flow-based approaches can introduce artifacts due to occlusions in the source frame, large pose deformations, and inaccurate flow estimates. Moreover, these methods are incapable of producing realistic and fine-grained fabric dynamics, such as wrinkling, folding, and flowing, as these details are not captured by appearance flows. A recent breakthrough in image-based virtual try-on uses a diffusion model [217], which implicitly warps the input garment under large pose gaps and heavy occlusion using spatial cross-attention. However, directly applying [217] or other image-based try-on methods for VVT in a frame-by-frame manner creates severe flickering artifacts and temporal inconsistencies.

Diffusion models [37, 76, 160, 162, 164] have shown promising results on various video synthesis tasks, such as text-to-video generation [78] and image-to-video generation [64, 82, 88]. However, a key challenge is generating longer videos, while maintaining temporal consistency and adhering to computational and memory constraints. Previous works use cascaded approaches [75], sliding windows inference [78, 191], past-frame conditioning [69, 104, 126], and transitions or interpolation [26, 176]. Yet, even with such schemes, longer videos are temporally inconsistent, contain artifacts, and lack realistic textures and details. We argue that, similar to context modeling for

LLMs [25], short-video generation models can be naturally extended for long-video generation by a temporally progressive finetuning scheme, without introducing additional inference passes or multiple networks.

A potential option for diffusion-based VVT is to apply an animation model to a single try-on image generated by an image try-on model. However, as this is not an end-to-end trained system, any image try-on errors will accumulate throughout the video. We argue that a single VVT model would overcome this issue by 1) injecting explicit person and garment conditioning information into the model and 2) having an end-to-end training objective.

We present *Fashion-VDM*, the first VVT method to synthesize temporally consistent, high-quality try-on videos, even on diverse poses and difficult garments. Fashion-VDM is a single-network, diffusion-based approach. To maintain temporal smoothness, we inflate the M&M VTO [216] architecture with 3D-convolution and temporal attention blocks. We maintain temporal consistency in videos up to 64-frames long with a single network by training in a temporally progressive manner. To address input person and garment fidelity, we introduce split classifier-free guidance (split-CFG) that enables increased control over each input signal. In our experiments, we also show that split-CFG increases realism, temporal consistency, and garment fidelity, compared to ordinary or dual CFG. Additionally, we increase garment fidelity and realism by training jointly with image and video data. Our results show that Fashion-VDM surpasses benchmark methods by a large margin and synthesizes state-of-the-art try-on videos.

4.1 Related Works

Virtual Try-On. Traditional image virtual try-on approaches first warp the target garment onto the input person, then refine the resulting image [9, 27, 32, 68, 70, 103, 127, 144, 194, 202, 206]. Similarly, for video virtual try-on (VVT), past methods often rely on multiple networks to predict intermediate values, such as optical flow, background masks, and occlusion masks, to warp the target

garment to the person in each frame of the video [38, 39, 86, 185, 213]. However, inaccuracies in these intermediate values lead to artifacts and misalignment. Some image try-on approaches incorporate optical flow estimation to alleviate this misalignment [9, 67, 103, 105]. For VVT, MV-TON [213] proposes a memory refinement module to correct inaccurate details in the generated frames by encoding past frames into latent space, then using this as external memory to generate new frames. ClothFormer [86] estimates an occlusion mask to correct for flow inaccuracies. Current state-of-the-art VVT methods achieve improved results by utilizing attention modules in the warping and fusing phases [86, 185].

In contrast to earlier flow-based methods, TryOnDiffusion [217] leverages a diffusion-based method conditioned with pose and garment for image virtual try-on. WarpDiffusion [192] tries to reduce the computational cost and data requirements by bridging warping and diffusion-based virtual try-on methods. StableVITON [94] avoids warping by finetuning pretrained latent diffusion [147] encoders for input person and garment conditioning via cross-attention blocks. Mix-and-match (M&M) VTO [216] extends single tryon task for mixmatch tryon application with a novel person embedding finetuning strategy.

Image and Video Training. Video datasets are often smaller and less diverse, compared to image datasets, as images are more abundant online. To alleviate this problem, [75, 78, 191] propose jointly leveraging image and video data for training. VDM [78] and Imagen Video [75] implement joint training by applying a temporal mask to image batches. MagicAnimate [191] applies joint training during the pretraining stage of their appearance encoder and pose ControlNet. We improve upon existing joint training schemes (see Section 4.2.7), ultimately demonstrating the benefit of joint image and video training for video try-on.

Subject Identity Preservation in Video. Several existing image-conditioned video diffusion models require subject-specific finetuning to achieve high-fidelity results aligned to the input image [17, 64, 88] which limits their applicability in practice. Other existing diffusion models animate an input image given a conditioning pose sequence [52, 64, 82, 88, 191]. DreamPose

uses a pre-trained (latent) Stable Diffusion model without temporal layers to generate videos in a frame-by-frame manner [88]. More recently, Animate Anyone [82] encodes the image using ReferenceNet and their diffusion model incorporates spatial, cross, and temporal attention layers to maintain consistency and preserve details, while MagicAnimate [191] introduces an appearance encoder to maintain the fidelity across the frames and generates a long video using temporally overlapping segments. In contrast, Fashion-VDM is a non-latent, temporally-aware video diffusion model, capable of synthesizing up to 64 consecutive frames in a single inference pass without subject-specific finetuning.

4.2 Method

We propose Fashion-VDM, a unified video diffusion model for synthesizing state-of-the-art virtual try-on (VTO) videos up to 64 frames long at 512px resolution. Our method introduces an end-to-end diffusion-based VVT architecture based on [216] (Section 4.2.4), split classifier-free guidance (split-CFG) for increased garment fidelity (Section 4.2.5), progressive temporal training for long-video generation (Section 4.2.6), and joint image-video training for improved garment fidelity (Section 4.2.7).

4.2.1 Problem Formulation

In video virtual try-on, the input is a video $\{I_p^0, I_p^1, \dots, I_p^{N-1}\}$ of a person p consisting of N frames and a single garment image I_g of another person wearing garment g . The goal is to synthesize a video $\{I_{tr}^0, I_{tr}^1, \dots, I_{tr}^{N-1}\}$, where I_{tr}^i denotes the i -th try-on video frame that preserves the identity and motion of the person p wearing the garment g .

4.2.2 Preliminary: M&M VTO

Our VTO-UDiT network architecture is inspired by [216], a state-of-the-art multi-garment *image* try-on diffusion model that also enables text-based control of garment layout. VTO-UDiT is represented by

$$\hat{x}_0 = x_\theta(z_t, t, c_{tr}) \quad (12)$$

where $\hat{x}_0$ is the predicted try-on image by the network x_θ , parameterized by θ , at diffusion timestep t , z_t is the noisy image, and c_{tr} is the conditioning inputs. VTO-UDiT is parameterized in v-space, following [150]. Each conditioning input is encoded separately by fully convolutional encoders and processed at the lowest resolution of the main UNet via DiT blocks [136], where conditioning features are processed with self-attention or cross-attention modules. However, while it shows impressive results for image try-on, VTO-UDiT cannot reason about temporal consistency when applied to video inputs.

4.2.3 Input Preprocessing

From the input video frames, we compute the clothing-agnostic frames $I_a = \{I_a^0, I_a^1, \dots, I_a^{N-1}\}$, person poses $J_p = \{J_p^0, J_p^1, \dots, J_p^{N-1}\}$, and person masks $\{M_p^0, M_p^1, \dots, M_p^{N-1}\}$. The clothing-agnostic frames mask out the entire bounding box area of the person in the frame, except for the visible body regions (head, hands, legs, and shoes), following TryOnDiffusion [217]. Optionally, the clothing-agnostic frames can keep the original bottoms, if doing top try-on only. From the input garment image I_g , we extract the garment segmentation image S_g , garment pose J_g , and garment mask M_g . The garment pose refers to the pose keypoints of the person wearing the garment before segmentation. We channel-wise concatenate M_p^i to I_a^i and M_g to I_g . Poses, masks, and segmentations are computed using an in-house equivalent of Graphonomy [53]. Both person and garment pose keypoints are preprocessed to be spatially aligned with the person frames and garment image, respectively.

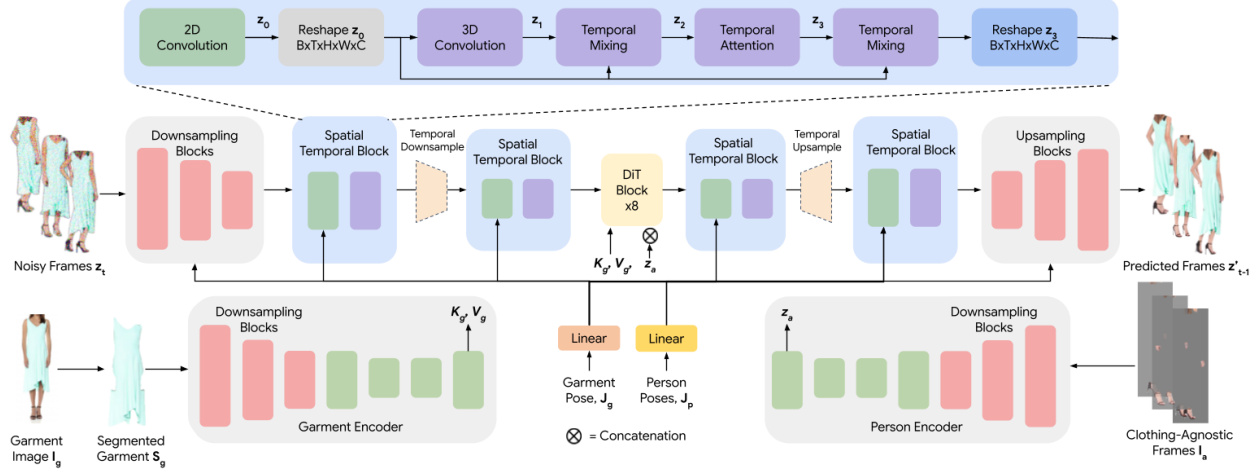


Figure 16: Fashion-VDM Architecture. Given a noisy video z_t at diffusion timestep t , a forward pass of Fashion-VDM computes a single denoising step to get the denoised video z'_{t-1} . Noisy video z_t is preprocessed into person poses J_p and clothing-agnostic frames I_a , while the garment image I_g is preprocessed into the garment segmentation S_g and garment poses J_g (Section 4.2.3). The architecture follows [216], except the main UNet contains 3D-Conv and temporal attention blocks to maintain temporal consistency. Additionally, we inject temporal down/upsampling blocks during 64-frame temporal training. Noisy video z_t is encoded by the main UNet and the conditioning signals, S_g and I_a , are encoded by separate UNet encoders. In the 8 DiT blocks at the lowest resolution of the UNet, the garment conditioning features are cross-attended with the noisy video features and the spatially-aligned clothing-agnostic features z_a and noisy video features are directly concatenated. J_g and J_p are encoded by single linear layers, then concatenated to the noisy features in all UNet 2D spatial layers.

4.2.4 Architecture

Our overall architecture is depicted in Figure 16. We adapt the VTO-UDiT architecture [217] by inflating the two lowest-resolution downsampling and upsampling blocks with temporal attention and 3D-Conv blocks, as shown in Figure 16. To be specific, after the 2D-Conv layers, we add a 3D-Conv block, a temporal attention block, and a temporal mixing block to linearly combine spatial and temporal features, as proposed in [18]. In the temporal mixing blocks, processed features after the spatial attention layer z_s are linearly combined with processed features after the temporal attention layer z_t via learned weighting parameter α :

$$z'_t = \alpha \cdot z_s + (1 - \alpha) \cdot z_t \quad (13)$$

During 64-frame training (see Section 4.2.6), we further inflate the model with temporal downsampling and upsampling blocks with factor 2, to reduce the memory footprint of the model. These blocks are added before and after the lowest-resolution spatial blocks, respectively.

The person and garment poses are encoded and used to condition all 2D spatial layers in the UNet. The 8 Diffusion Transformer (DiT) blocks [136] between the UNet encoder and decoder condition our model on the segmented garment and clothing-agnostic image features, as proposed by [216]. In each block, the garment images are cross-attended with the noisy target features, while the agnostic input images are concatenated to the noisy target features.

4.2.5 Split Classifier-Free Guidance

Standard classifier-free guidance (CFG) [77] is a sampling technique that pushes the distribution of inference results towards the input conditioning signal(s); however, it does not allow for disentangled guidance towards separate conditioning signals. Instruct-Pix2Pix [20] introduces dual-CFG, which separates the CFG weights for text and image conditioning signals.

We introduce *split*-CFG, a generalization of dual-CFG which allows independent control over multiple conditioning signals. See Algorithm 1. The inputs to `SPLIT-CFG` are the trained denoising UNet ϵ_θ , the list of all conditioning signal sets C , and the respective conditioning weights W . For each subset of conditioning signals $c_i \in C$, containing one or more conditional inputs, the algorithm computes the conditional result ϵ_i given c_i . Then, the weighted difference between the conditional result ϵ_i from the past conditional result ϵ_{i-1} is added to the prediction. In this way, the prediction is pushed in the direction of c_i .

Split-CFG is naturally dependent on the order of the conditioning signals. Intuitively, the first conditional output will have the largest distance from the null output, thus most affecting the final result. In our implementation, our conditioning groups C consist of (1) the empty set (unconditional inference), (2) the clothing-agnostic images ($\{I_a^0, \dots, I_a^{N-1}\}$), (3) all clothing-related

Algorithm 1: Split Classifier-Free Guidance

```
Split-CFG( $\epsilon_\theta, C, W$ )  
   $c \leftarrow \emptyset$ ; // current conditioning signals  
   $\hat{\epsilon}_\theta(z_t, C) \leftarrow w_0 \epsilon_\theta(z_t, \emptyset)$ ; // initialize prediction  
   $\hat{\epsilon}_0 \leftarrow \hat{\epsilon}_\theta(z_t, C)$ ; // store past prediction  
  for  $c_i$  in  $C$  do  
     $c \leftarrow c \cup \{c_i\}$ ; // update  $c$   
     $\hat{\epsilon}_i \leftarrow \epsilon_\theta(z_t, c)$ ; // store new prediction  
     $\hat{\epsilon}_\theta(z_t, C) \leftarrow \hat{\epsilon}_\theta(z_t, C) + w_i(\hat{\epsilon}_i - \hat{\epsilon}_{i-1})$ ;  
     $\hat{\epsilon}_{i-1} \leftarrow \hat{\epsilon}_i$ ; // update  $\hat{\epsilon}_{i-1}$   
  end  
  return  $\hat{\epsilon}_\theta(z_t, C)$ 
```

inputs (S_g, J_g, M_g) , and (4) lastly, all remaining conditioning inputs $(\{J_p^0, \dots, J_p^{N-1}\})$. We denote the respective weights of each term as $(w_\emptyset, w_p, w_g, w_{\text{full}})$. Empirically, we find this ordering yields the best results.

Overall, we find that controlling sampling via split-CFG not only enhances the frame-wise garment fidelity, but also increases photorealism (FID) the inter-frame consistency of video (FVD), compared to ordinary CFG.

4.2.6 Progressive Temporal Training

Our novel progressive temporal training enables up to 64-frame video generation in a single inference run (Figure 17). We first train a base image model from scratch on image data at 512px resolution and image batches of shape $B \times T \times H \times W \times C$, with batch size $B = 8$ and length $T = 1$, for 1M iterations. Then, we inflate the base architecture with temporal blocks and continue training the model using our joint image-video training strategy. In these temporal training phases, half of the batches are from the image dataset and the other half are batches of consecutive video frames ($B = 1$) from the video dataset. When training with an image batch, we skip the temporal blocks entirely in the forward and backward passes. At each successive phase of temporal training, we initialize the model from the checkpoint of the previous phase and double the training video

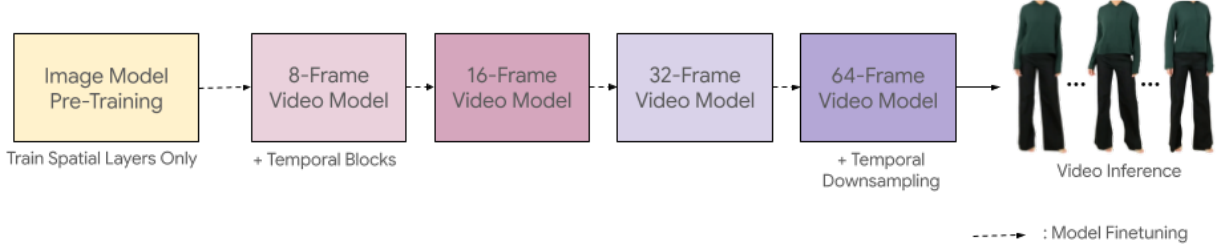


Figure 17: Progressive Training Strategy. Fashion-VDM is trained in multiple phases of increasing frame length. We first pretrain an image model, by training only the spatial layers on our image dataset. In subsequent phases, we train temporal and spatial layers on increasingly long batches of consecutive frames from our video dataset.

length: $T = 8 \rightarrow 16 \rightarrow 32 \rightarrow 64$. We train each temporal training phase for 150K iterations. Once the video length becomes prohibitively large in memory at 64-frames, we introduce temporal downsampling and upsampling layers to the model. At test time, our model generates $512 \times 384\text{px}$ videos up to 64-frames in one inference pass with a single network. The benefit of such a progressive process is a faster training speed and better multi-frame consistency. Additional details are provided in the appendix.

4.2.7 Joint Image and Video Training

Training the temporal phases solely with video data, which is much more limited in scale compared to image data, would disregard the image dataset entirely after the pretraining phase. We observe that video-only training in the temporal phases sacrifices image quality and fidelity for temporal smoothness. To combat this issue, we train the temporal phases jointly with 50% image batches and 50% video batches. We implement joint training via conditional network branching [83], i.e. for image batches, we skip updating the temporal blocks in the network. Unlike temporal masking strategies [75, 78], using conditional network branching allows us to include other temporal blocks (Conv-3D, temporal mixing) in addition to temporal attention. Critically, we also train with either image-only or video-only batches, rather than batches of video with appended images [75, 78]. This improves data diversity and training stability by not constraining the possible batches by the number of available video batches. We observe that improved garment fidelity and multi-view

realism, especially for synthesized details in occluded garment regions with joint image-video training compared to video-only training (see Figure 20).

4.3 Experiments

In this section, we describe our datasets (Section 4.3.1), evaluation metrics (Section 4.3.2), and results (Section 4.3.3). We provide training and inference details in the appendix.

4.3.1 Datasets

Our image dataset is a collection of publicly-crawled online fashion images, containing 17M paired images of people wearing the same garment in different poses. We also collect a video dataset of over 52K publicly-available fashion videos totalling 3.9M frames, which we use for the temporal training phases. During training, the garment image and person frames are randomly sampled from the same video. For evaluation, we collect a separate dataset of 5K videos, containing person videos paired with garment images from a *different* video. Our custom image and video datasets contain a diverse range of skin tones, body shapes, garments, genders, and motions. We also evaluate on the UBC test dataset [203] of 100 videos. For both test datasets, we randomly pair a garment frame from each video clip with three distinct other video clips to get swapped try-on datasets.

Reproducibility. To promote future work in this area and allow fair comparisons with our method, we plan to release a benchmark dataset, including sample paired person videos, garment images, and corresponding preprocessed inputs. We also analyze a version of our model trained and tested exclusively on publicly-available UBC video data [203] in Section 4.3.3.



Figure 18: Qualitative Results. We showcase video try-on results generated by Fashion-VDM using randomly paired person-garment test videos from the UBC dataset [203] and our own collected test dataset. Note that the input garment image and input person frames come from different videos.

	UBC Test Dataset			Our Test Dataset		
	FID ↓	FVD ↓	CLIP ↑	FID ↓	FVD ↓	CLIP ↑
<i>w/o</i> Split-CFG	145	687	0.745	78	450	0.663
<i>w/o</i> Joint Training	106	579	0.744	96	565	0.651
<i>w/o</i> Prog. Training	102	631	0.736	87	824	0.651
<i>w/o</i> Temporal Blocks	94	1019	0.739	95	565	0.642
Ours (Full)	86	515	0.752	71	377	0.669

Table 4: Quantitative Ablation Studies. For each ablated version of our model, we compute FID, FVD, and CLIP scores using both UBC and our test videos with randomly paired garments. Bolded values indicate the best score in each column.

4.3.2 Metrics

We evaluate our method using FID [73], FVD [171], and CLIP [138] scores in Tables 4 and 5. FID measures the similarity of the distributions of the predicted and ground-truth frames, which gives a measure of the realism of the generated video frames. FVD measures temporal consistency of video frames. We compute the CLIP image similarity between the segmented garments of the input garment image and predicted frames. In this way, the CLIP score gives us a measure of try-on garment fidelity.

4.3.3 Results

We showcase qualitative results of our full method in Figure 18 and provide more qualitative results in the appendix. Fashion-VDM is capable of synthesizing smooth, photorealistic try-on videos on a variety of input garment types, patterns, skin tones, genders, and motions.

UBC-Only Model. In order to provide a fair comparison to other methods [17, 64, 88], we train a version of Fashion-VDM using video data only from the UBC dataset. Similar to other methods, we leverage a pretrained image try-on diffusion models and further train using the publicly-available UBC dataset for the video stages. We show the quantitative results in Table 5 and provide further details, discussion, and qualitative examples in the appendix.



Figure 19: Split-CFG Ablation. We compare different split-cfg weights, where $(w_{\emptyset}, w_p, w_g, w_{\text{full}})$ correspond to the unconditional guidance, person-only guidance, person and cloth guidance, and full guidance terms, respectively.

4.3.4 Ablation Studies

We ablate each of our design choices with respect to garment fidelity, temporal smoothness, and photorealism. We report quantitative results for each ablated version in Table 4. All components are essential to improving realism (FID), temporal consistency (FVD), and garment fidelity (CLIP). Qualitatively, we find that split-CFG and joint training have the largest effect on person/garment fidelity and overall quality (Figure 21), while progressive training and temporal blocks affect the temporal smoothness (Figure 22). We discuss these effects in detail in the remainder of this section.

Split Classifier-Free Guidance.

Split-CFG improves per-frame person and garment fidelity, thereby improving overall inter-frame temporal consistency and photorealism. In Figure 19, we compare results generated with different split-CFG weights at inference time. By increasing the person guidance weight w_p from 0 to 1, the realism and identity of the input person are improved. Increasing the full-conditional weight w_{full} improves the garment fidelity, but not as much as by increasing the garment weight w_g alone, as in the last column. We provide quantitative split-CFG ablation results in the appendix. We also demonstrate that increasing w_g also increases fine-grained garment details when using a version of



Figure 20: Joint Training Ablation. Joint image and video training improves the realism of occluded views.

our model trained on limited video data (see appendix). This suggests split-CFG does not require extensive training to be useful and can be impactful in low-resource settings.

Joint Image-Video Training. We find that training with video data only in the temporal phases sacrifices garment fidelity compared to the base image model. Training jointly with images and videos increases the fidelity to garment details, even compared to the image baseline, as shown by the improved FID and CLIP scores in Table 4. The increased access to diverse data with joint image-video training also enables the model to synthesize more plausible occluded regions. For example, as shown in Figure 20, the jointly trained model is able to generate a hood with more realism than the video-only model.

Temporal Blocks. As seen in prior works [18, 75, 78], interleaving 3D-convolution and temporal attention blocks into the 2D UNet greatly improves temporal consistency. Removing temporal blocks entirely causes large temporal inconsistencies. For instance, in the top row of Figure 22, the ablated model without temporal blocks swaps the pants and body shape in each frame.

Progressive Temporal Training. To ablate our progressive training scheme, we train our image base model directly with 16-frame video batches for the same total number of iterations, but skipping the 8-frame training phase entirely. Progressive training enables more temporally smooth results with the same number of training iterations. This is supported by our quantitative findings in Table 4,



Figure 21: Garment Fidelity Ablations. We compare our full model with ablated versions without split-CFG and without joint image-video training in terms of garment fidelity. Both split-CFG and joint image-video training improve fine-grain garment details (top row) and novel view generation (bottom row).

which indicates worse FVD when not doing progressive training. Qualitatively, in Figure 22, the non-progressively trained model in the middle row exhibits temporal artifacts in the pants region and intermittently merges the pant legs into a skirt. We hypothesize that, given limited training iterations, it is easier to learn temporal consistency well across a small number of frames. Then, to transfer that knowledge to larger temporal windows only requires minimal additional training.

4.3.5 Comparisons to State-of-the-Art

We qualitatively and quantitatively compare our method to the state-of-the-art in diffusion-based try-on and animation, as no previous diffusion-based video try-on baselines with publicly-available code

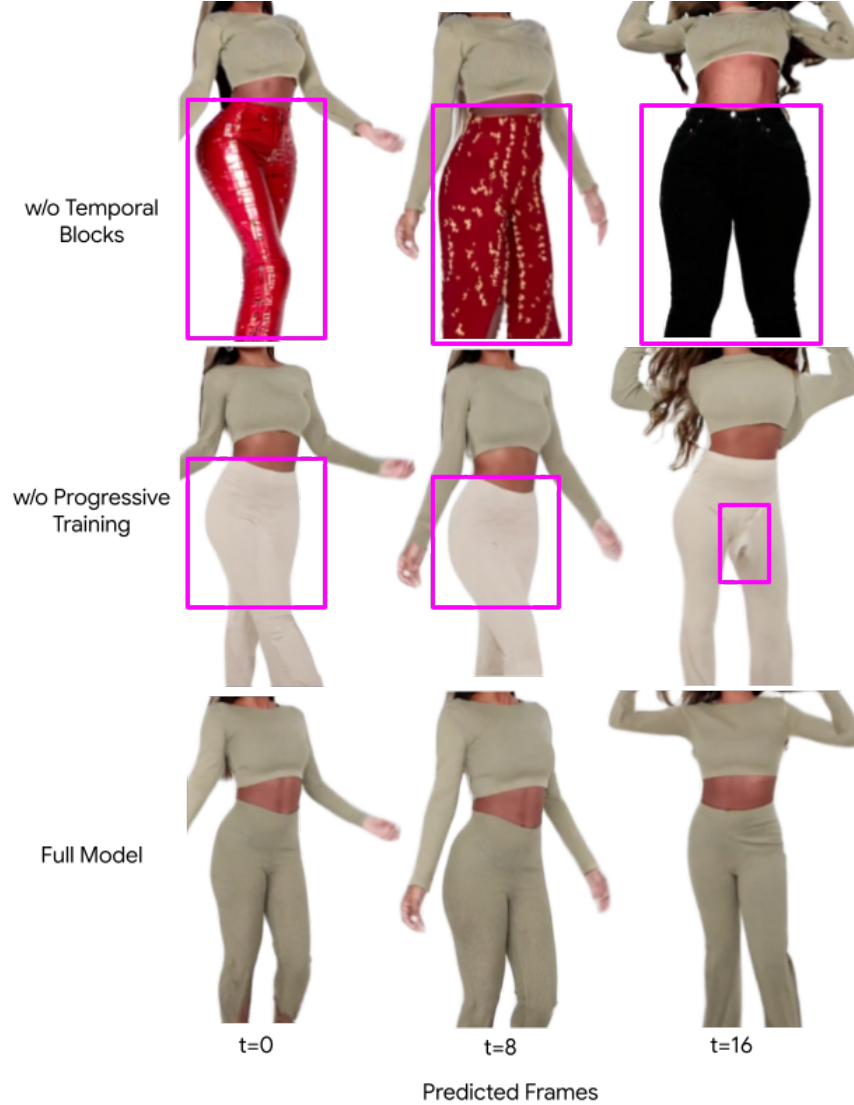


Figure 22: Temporal Smoothness Ablations. We compare video frames generated by our ablated model without temporal blocks (top row) and without progressive training (middle row) to our full model (bottom row). Both ablated versions exhibit large frame-to-frame inconsistencies and artifacts.

currently exist: (1) TryOn Diffusion [217] (2) MagicAnimate [191], and (3) Animate Anyone [82]. For (1), we generate try-on results in a frame-by-frame manner for each input frame to generate a video. For (2) and (3), we first generate a single try-on image from the first input frame and garment image using TryOn Diffusion, then use the extracted poses from the input frames to animate the result. In addition, we provide user survey results in the appendix.

Qualitative Results. We qualitatively compare Fashion-VDM to the baseline methods in Figure 23.



Figure 23: Qualitative Comparisons. Fashion-VDM outperforms past methods in garment fidelity and realism. Especially in cases of large disocclusion, our method synthesizes more realistic novel views.

	UBC Test Dataset			Our Test Dataset		
	FID ↓	FVD ↓	CLIP ↑	FID ↓	FVD ↓	CLIP ↑
TryOn Diffusion	94	1019	0.739	95	960	0.663
Magic Animate	155	1861	0.702	97	694	0.642
Animate Anyone	118	819	0.727	112	468	0.629
Ours (Full)	86	515	0.752	71	377	0.669
Ours (UBC-Only)	39	172	0.749	129	949	0.657

Table 5: Quantitative Comparisons. We compare Fashion-VDM to the baseline methods using the UBC test dataset [203] and our test dataset of internet videos. Fashion-VDM quantitatively outperforms other methods on all metrics.

In the top and bottom rows, we show how other methods exhibit large artifacts with large pose changes. In these examples, baseline methods struggle to preserve garment details and hallucinate plausible occluded views. Plus, both MagicAnimate and Animate Anyone create an overall cartoon-like appearance.

We observe that frame-by-frame TryOn Diffusion results exhibit lots of flickering and garment inconsistencies. MagicAnimate fails to preserve the correct background and also does not maintain a consistent garment appearance throughout the video. Animate Anyone also exhibits garment temporal inconsistency, especially with large viewpoint changes, and the human motion has an unrealistic, warping effect. Overall, Fashion-VDM synthesizes more natural-looking garment motion, such as folding, wrinkling, and flow, and better preserves garment appearance.

Quantitative Results. We compute FID scores on 300 16-frame videos of the UBC dataset and on 300 16-frame videos of our custom video test dataset. For both datasets, we compute FVD scores and CLIP on 100 distinct 16-frame videos. The results are displayed in Table 5. In our experiments, Fashion-VDM surpasses all baselines in both image quality (FID), video quality (FVD), and garment fidelity (CLIP). Although the UBC-only model excels in terms of all UBC metrics, we qualitatively observe over-smoothing and worse garment detail preservation, compared to the full version trained on our larger, more diverse video dataset.

4.4 Limitations and Future Work

The main limitations of Fashion-VDM include inaccurate body shape, artifacts, and incorrect details in occluded garment regions. See examples and further discussion in the appendix. Improbable details may be hallucinated in unseen garment regions, because the input image only shows one view of the garment. Future work might consider multi-view conditioning and individual person customization for improved garment and person fidelity. Other errors include minor aliasing for fine-grained patterns. Finally, our method does not simulate exact physical cloth dynamics, but rather realistic video try-on visualization. Establishing physics could be a great next step.

Discussion

We present Fashion-VDM, a diffusion-based video try-on model. Given an input garment image and person video, Fashion-VDM synthesizes a try-on video with the input garment fitted to the person in motion, maintaining realistic details and fabric dynamics. We show qualitatively and quantitatively that our method significantly surpasses existing state-of-the-art diffusion-based image try-on and animation methods.

Ethics Statement

While we believe our research creates a positive contribution to the research community by advancing the state-of-the-art in generative video diffusion, we also condemn its potential for misuse, including any spreading misinformation or manipulating human content for malicious purposes. While our method is trained on public data containing identifiable humans, we will not release any images or videos containing personally identifiable features, such as faces, tattoos, or logos to protect the privacy of these individuals.

Appendix

4.4.1 Additional Qualitative Results

In Figure 24, we provide additional examples of our method on test videos from both the UBC dataset [203] and our own collected test dataset.

4.4.2 Training and Inference Details

We train our model on 16 TPU-v4’s for approximately 2 weeks, including all training phases. Our image baseline model is trained for 1M iterations with a batch size of 8 and resolution 512×384 px using the Adam optimizer with a linearly decaying learning rate of $1e^{-4}$ to $1e^{-5}$ over 1M steps and 10K warm-up steps. Each phase of progressive temporal training is initialized from the previous checkpoint and trained for 150K iterations, following the order of phases described in the Section 4.2.6. For all phases, we incorporate dropout for each conditional input independently 10% of the time. We train with an L2 loss on ϵ .

During inference, we use the DDPM sampler [76] with 1000 refinement steps. Each video takes approximately 8 minutes to synthesize with split-CFG and 5 minutes without split-CFG.

4.4.3 Examples of Failure Cases

We show two examples of failure cases of our method in Figure 25. In row 1, we show artifacts that appear in the body/garment boundary, due to an imperfect person segmentation in the clothing-agnostic image. Imperfect segmentation is a common cause of such artifacts, and may also incorrectly leak regions from the original garment. In our human evaluation (Section 4.4.5), 10/17 videos that were failed had agnostic errors. In general, although our preprocessing methods are state-of-the-art, other types of preprocessing errors limit the quality of Fashion-VDM. In total, 70%



Figure 24: **Additional Qualitative Results.** We showcase video try-on results generated by Fashion-VDM using swapped test videos from the UBC dataset [203] and our own collected test dataset. Note that the input garment image and input person frames come from different videos.

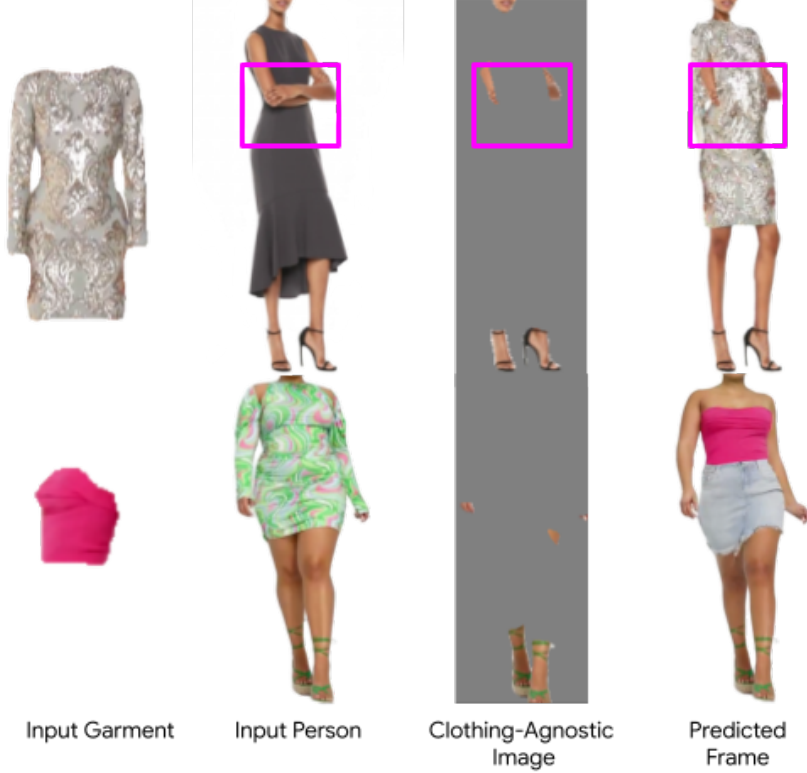


Figure 25: **Failure Cases.** Errors in the person segmentation may lead to artifacts (top row). Fashion-VDM may incorrectly represent body shape (bottom row).

of videos not chosen by human raters had errors in one or more inputs. As shown in row 2, body shape misrepresentation (e.g. slimming) occurs, because the clothing-agnostic images remove all body parts, besides hands, feet, and head, thus they do not include detailed information about body size.

4.4.4 Split-CFG Weights Ablations

We quantitatively evaluate our choice of split-CFG weights for both datasets on a held-out validation set. The results are shown in Table 6. Calibrating these weights correctly is not only beneficial to preserving garment fidelity, as shown by the FID score, but also increasing temporal consistency, as shown by the FVD score. Intuitively, by increasing the similarity of the output garment to the input garment, there is less allowed variability in the appearance of each frame, thus increased temporal smoothness. Based on these results, we employ weights $(1, 1, 3, 1)$ for UBC and weights $(1, 1, 1, 1)$

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

Table 6: Quantitative Ablation of Split-CFG Weights. We compute FID, FVD, and CLIP scores of our full model using different split-CFG weights.

for our test dataset.

4.4.5 User Study

In addition to qualitative and quantitative evaluations, we perform user studies for our state-of-the-art comparisons. The results are shown in Table 7. Our user studies are conducted by 5 human raters who are unfamiliar with the method. For each sample, the raters were asked to select which video performs best in each category: temporal smoothness, garment fidelity to the input garment image, and person fidelity to the input person video. The scores on both UBC test dataset and our test dataset reported are fraction of total votes divided by the total number of videos. Fashion-VDM exceeds other methods on all three user preference categories for both datasets.

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

Table 7: User Study. Our study indicates that users overwhelmingly prefer Fashion-VDM to other baselines in terms of video smoothness, person fidelity, and garment fidelity on both test datasets.



Figure 26: **Split-CFG Ablation with UBC-Only Model.** When Fashion-VDM is trained on the limited UBC dataset only, we observe overfitting to the largely plain garments in the UBC train dataset. However, we find that increasing garment image guidance (w_g) in split-CFG significantly increases garment details.

4.4.6 UBC-Only Model

We initialize this model from our pretrained image model, which is comparable to an open source image diffusion model, like Stable Diffusion [147], which are trained on even larger image datasets, including LAION 5B [152]. We then train progressively using both image data and UBC video data, following the same progressive training scheme as the full model.

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



Figure 27: Qualitative Results for UBC-Only Model. Our model trained only on UBC data generates temporally consistent, smooth try-on videos for plain and simple patterned garments, but struggles to preserve intricate patterns and complex garment shapes.

We qualitatively show this in Figure 27. This implies that when training with limited data, split-CFG becomes even more crucial to preserving the conditioning image details.

We provide qualitative examples generated by our model trained only on the UBC dataset [203] in Figure 27. While the results are still smooth and temporally consistent, the model struggles to maintain complex patterns and garment shape details. This is likely due to overfitting to the limited size and scope of the UBC training dataset, consisting of 500 videos of women in dresses.

Chapter 5

HOLOGARMENT: 360° NOVEL VIEW SYNTHESIS OF IN-THE-WILD GARMENTS

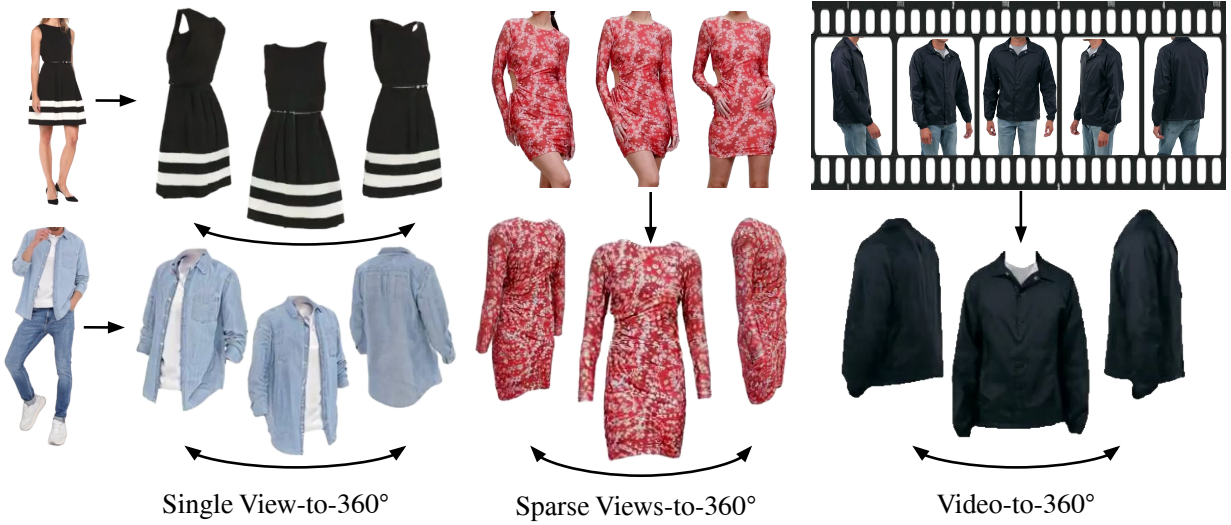


Figure 28: HoloGarment enables 360° novel view synthesis of real-world garments in images and videos.

The project in this chapter was completed in collaboration with Yingwei Li and Ira Kemelmacher-Shlizerman during my internship at Google. It was originally published in 2025 [89]. The analysis and comparisons in this chapter are based on the existing state-of-the-art during that time.

Introduction

The rise in online retail, virtual try-on, and digital fashion design is driving the demand for high-quality digital garment visualizations. While images and videos offer limited, pose-dependent views of a garment, they fall short of providing a full 360° garment representation – one that is free of occlusions and wrinkles. Furthermore, manually capturing such 360° views is highly impractical and costly. Therefore, there is strong interest in discovering an automatic method to generate

high-quality, high-fidelity novel views of garments from images and videos.

This is a challenging task, since real-world garments are inherently complex, containing deformations, occlusions, and pose variations when worn. Existing methods for novel view synthesis (NVS) of general objects [47, 189] only handle a fixed number of input views and are constrained to static and unoccluded objects. As a result, these approaches perform poorly on real-world garments and do not handle an arbitrary number of input views, such as from a video. While some methods handle finetuning on video [178, 204], they are prone to overfitting to the shape and appearance of the subject in the input frames, thus failing to generate unoccluded, static novel views of a standalone garment in a dynamic video.

Another challenge is the lack of diverse, realistic 3D garment datasets. While 2D garment data (images and videos) are abundantly available online, they are missing ground-truth 3D representations. As such, past works [10, 46, 98, 108, 129, 146, 151, 165] often leverage purely synthetic garment data [107, 129, 215], but tend to overfit to simplistic shapes and patterns and generalize poorly to diverse garment shapes and textures. To overcome these limitations, we ask: (1) Can we train a real-world garment NVS model by leveraging abundant real-world 2D data, even in the absence of paired ground-truth 3D assets? (2) In contrast to sparse-frame conditioning, can we leverage dense frames from a dynamic video sequence to learn a robust and high-fidelity garment representation for NVS?

In this paper, we propose HoloGarment, a video diffusion model for garment NVS from images and videos of in-the-wild dressed humans. Our key insight is a novel implicit training paradigm, where two or more distinct training tasks indirectly train a model to perform the target task for which ground truth data is not available. By implicitly training with a combination of real 2D data and synthetic 3D assets, our method learns a shared garment embedding space between both domains that enables real-world garment novel view synthesis. In doing so, we bypass the limitations of synthetic-only 3D datasets to handle challenging real-world garment images and videos. Furthermore, we introduce the notion of a garment “atlas”, a finetuned garment embedding optimized on a specific dynamic

video featuring a person wearing the garment. The “atlas” bridges the gap between finetuning (2D) and inference (3D) modalities, enabling the novel task of *video*-to-NVS generation, as well as eliminates the need for arbitrary input view selection. Our experiments showcase HoloGarment’s capability to generate high-quality, high-fidelity 360° novel views across a variety of garment types, including tops, dresses, jackets, rompers, and pants, even those containing occlusions, pose variations, and deformations. We further quantitatively and quantitatively demonstrate that our method achieves state-of-the-art results compared to related methods.

5.1 Related Works

Novel View Synthesis with Diffusion Models: Novel view synthesis refers to generating novel object views from limited observations. Many diffusion-based NVS methods use 3D datasets to finetune a pretrained text-to-image diffusion model [47, 114, 118, 153, 154, 177] or video diffusion model [101, 182, 214]. However, their reliance on 3D data limits their ability to handle real-world objects effectively. Several works have explored using 2D diffusion priors to enhance 3D consistency [109, 137, 154], but do not tackle cases where input views contain incomplete information (i.e. occlusions) or inconsistencies (i.e. deformations, pose changes). Existing NVS methods therefore struggle with diverse garments in complex, dynamic real-world scenarios. In this work, we address this challenging case directly by training a video diffusion model implicitly on real 2D videos and synthetic 3D assets, enabling our method to robustly generate consistent novel views of real-world garments.

3D Garment Reconstruction: Related to the task of garment NVS is 3D garment reconstruction, which aims to recover the 3D geometry of a garment in an image. One avenue of methods explores 2D sewing pattern estimation [10, 98, 108, 112, 129], which predicts flat patterns that can be draped onto the person’s 3D structure, but neglects to preserve texture details. Other recent methods focus on texture estimation by utilizing template garment meshes to achieve better realism in garment representation [46, 146, 151, 165].

A major limitation of both approaches is their reliance on limited synthetic 3D garment datasets, including DressCode [129] and GarverseLOD [123], which lack ground-truth textures. As a result, these methods do not generalize well to real-world garment inputs. In contrast, our approach eliminates the reliance on purely synthetic data, input meshes, and complex templates.

Garment Finetuning: Subject-specific finetuning, like DreamBooth [148] and similar video-specific methods [178, 204], customize a pretrained generative model to a specific subject. While this paradigm has been extended for human identity preservation [88, 216], it has not yet been applied to garment identity specifically, which comes with unique challenges. One specific limitation is that current subject-finetuning methods tend to overfit to the pose and shape of the target subject. As such, existing methods will replicate the motion, occlusions, deformations, and wrinkling of a dynamic garment video that is used for finetuning. This is undesirable for synthesizing a static video of a dynamic garment in an unoccluded, canonical A-pose. We address this by explicitly disentangling animation and spin motion via split temporal blocks in our network, while still sharing the same garment appearance encoder. As a result, we are able to finetune a garment-specific embedding on a dynamic garment video and still generate novel views of the garment in a static pose, without overfitting to the original dynamic motion.

5.2 Method

Given 1-3 images I_g of a garment g and a driving pose sequence $J = (J_{2D}, J_{3D})$ represented in 2D and 3D, HoloGarment generates novel garment views $\hat{V}_g$ following the driving poses. In this section, we introduce our model architecture (5.2.2) and implicit training strategy with real 2D and synthetic 3D garment data (5.2.3). Then, we describe how this unlocks image(s)-to-360° NVS (5.2.4), as well as video-to-360° NVS capabilities via finetuning a garment “atlas” (5.2.5).

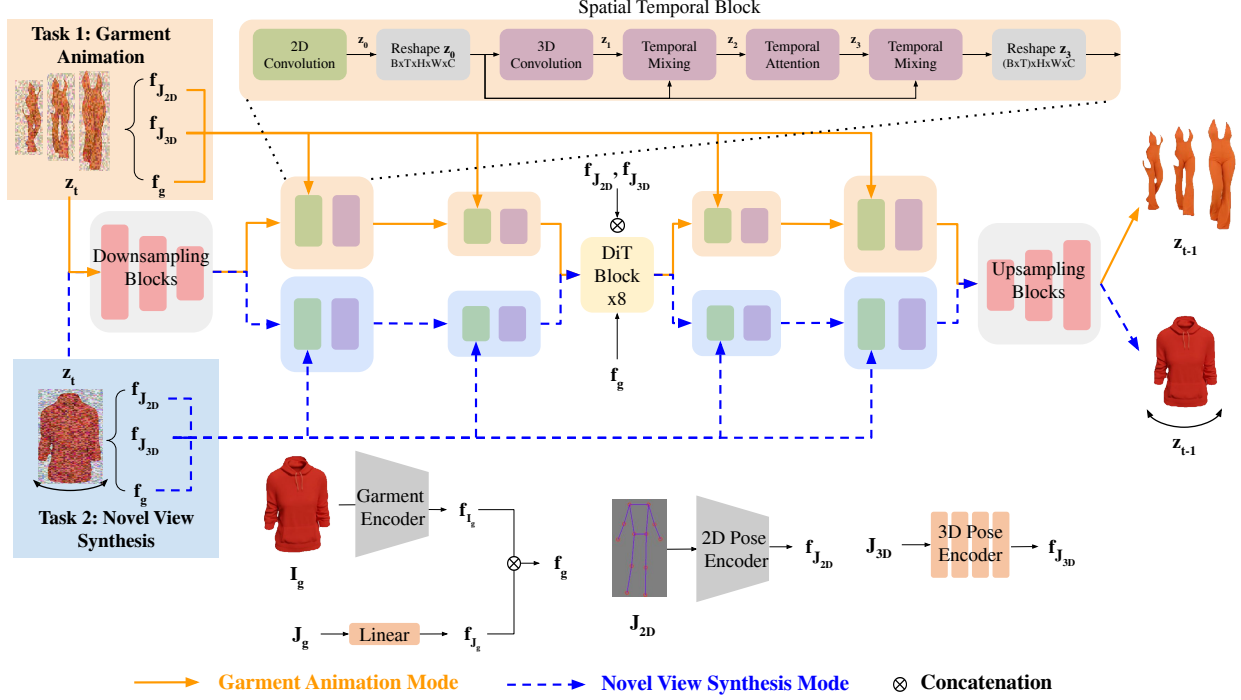


Figure 29: **Architecture.** Our VDM generates video frames conditioned on a garment image I_g and driving pose sequence (J_{2D}, J_{3D}) . By training separate sets of temporal blocks for each training task (orange for animation and blue for NVS), our VDM can effectively generate either dynamic or spin motion. Refer to Section 5.2.2 for details.

5.2.1 Preliminary: Video DiT

While conventional diffusion models often leverage a UNet backbone, the diffusion transformer (DiT) [136] model replaces this with a Transformer architecture, leading to superior scalability and performance. Fashion-VDM [90] extends DiT into a video model with temporal blocks (e.g. 3D convolutions and temporal attention layers) and progressive temporal training. Paired with parallel UNet encoders to disentangle person and garment conditioning signals [217], Fashion-VDM achieves superior performance for video try-on.

5.2.2 HoloGarment

At its core, HoloGarment consists of an image- and pose-conditioned video diffusion model (VDM) with trainable parameters θ . Its architecture (Figure 29) builds upon the video transformer

diffusion model proposed in Fashion-VDM [90]. However, it does not include any person image representation and the driving poses are encoded in both 2D and 3D. Our VDM additionally implements two identical sets of temporal blocks to separately handle video motion and 3D spin motion. We describe these adaptations in further detail below. Additional architecture and implementation details are provided in the appendix.

Garment and Pose Conditioning: Given a noisy video z_t at diffusion timestep t , a UNet encoder $\mathcal{E}_z$ encodes z_t into features f_z . Similarly, separate UNet encoders encode the input garment I_g and driving 2D poses J_{2D} into $f_{I_g} = \mathcal{E}_{I_g}(I_g)$, $f_{J_{2D}} = \mathcal{E}_{J_{2D}}(J_{2D})$. Meanwhile, the garment pose J_g and 3D driving pose sequence J_{3D} are encoded into $f_{J_g} = \mathcal{E}_{J_g}(J_g)$, $f_{J_{3D}} = \mathcal{E}_{J_{3D}}(J_{3D})$ by a single linear and 4 dense layers, respectively. Garment and pose features are subsequently concatenated: $f_g = f_{I_g} \oplus f_{J_g}$ and $f_j = f_{J_{2D}} \oplus f_{J_{3D}}$. Mathematically, at timestep t , the VDM performs one denoising step to recover noise $\hat{\epsilon}_t$.

$$\hat{\epsilon}_t = \text{VDM}_\theta(z_t, t, f_g, f_j) \quad (14)$$

Inside the VDM, these conditional features are processed in the VDM via transformer blocks (DiT [136]). Pose features f_j are spatially-aligned with f_z , so they are concatenated channel-wise before self-attention. Meanwhile, the non-spatially aligned garment features f_g and noisy video features f_z are cross-attended. In this manner, the garment features are implicitly warped to their target locations according to the driving poses [217].

Disjoint Temporal Blocks: To effectively disentangle dynamic and static spin motions, we implement our VDM with two identical, disjoint sets of temporal blocks. Each set consists of 3D-convolution, temporal attention, and temporal mixing blocks [18]. One set is trained exclusively on batches of *real*-world images and videos $\mathcal{D}_{\text{real}}$, while the other is trained only on batches of static *spin* renderings of 3D garment assets $\mathcal{D}_{\text{spin}}$. This architecture allows the VDM to better synthesize either dynamic or static spin motion, depending on the which set of temporal blocks are activated.

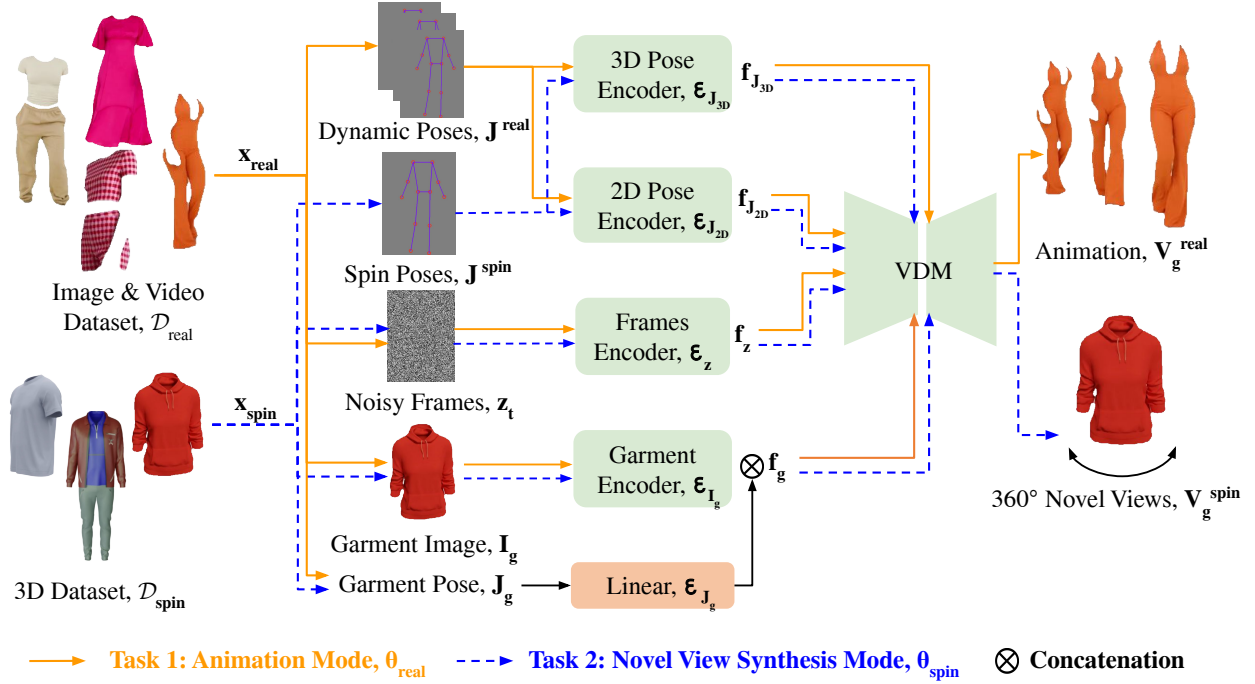


Figure 30: **Implicit Training Paradigm** The VDM is trained to generate either a garment animation given dynamic driving poses (solid orange path) or 360° novel views given static spin driving poses (blue dotted path). By training on both real 2D data and synthetic 3D data, the VDM implicitly learns canonical 360° NVS from real-world garment images, without paired real-world 3D data.

5.2.3 Implicit Training with Real and Synthetic Data

We introduce a novel implicit training paradigm to learn photorealistic garment NVS without real-world paired data. Implicit training leverages two or more related tasks for jointly training a model to perform a desired task. In contrast to *joint* training [78], where ground-truth task-specific data is supplemented with similar, non-task-specific data, an *implicit* training strategy leverages solely non-task-specific data to learn the target task.

In this case, the desired task is 3D-consistent NVS from real-world garment images, and the training tasks are (1) garment animation using real image and video data $\mathcal{D}_{\text{real}}$ and (2) NVS using synthetic 3D data $\mathcal{D}_{\text{spin}}$. Garment animation with real data trains the model to handle the desired *input style* – diverse, real-world garments, even under challenging conditions, like occlusions and wrinkling. NVS with synthetic 3D data trains the model to generate the desired *output style* – unoccluded, static 360° views (spin videos) of garments. As shown in Figure 30, we train our VDM by alternating

batches x_* from both datasets:

$$x_{\text{real}} = (V_g^{\text{real}}, I_g^{\text{real}}, J_g, J^{\text{real}}) \sim \mathcal{D}_{\text{real}} \quad (15)$$

$$x_{\text{spin}} = (V_g^{\text{spin}}, I_g^{\text{spin}}, J_g, J^{\text{spin}}) \sim \mathcal{D}_{\text{spin}} \quad (16)$$

During training, the VDM trains different temporal parameters for handling dynamic motion and static spin motion. For dynamic batches x_{real} , the VDM operates with θ_{real} and for spin batches x_{spin} , the VDM operates with θ_{spin} (Section 5.2.2). In this way, disjoint sets of temporal blocks are separately optimized for the different motion styles. After encoding the conditional inputs,

$$\hat{c}_t = \begin{cases} \text{VDM}_{\theta_{\text{real}}}(z_t, t, f_g^{\text{real}}, f_j^{\text{real}}) & x_{\text{real}} \sim \mathcal{D}_{\text{real}} \\ \text{VDM}_{\theta_{\text{spin}}}(z_t, t, f_g^{\text{spin}}, f_j^{\text{spin}}) & x_{\text{spin}} \sim \mathcal{D}_{\text{spin}} \end{cases} \quad (17)$$

Recall from Section 5.2.2, $f_g^{\text{real}} = \mathcal{E}_g(I_g^{\text{real}})$ and $f_g^{\text{spin}} = \mathcal{E}_g(I_g^{\text{spin}})$. Let F_G be the garment encoder’s ($\mathcal{E}_g$) embedding space for all garments $g \sim G$. Then,

$$f_g^{\text{real}}, f_g^{\text{spin}} \sim F_G \quad (18)$$

Thus, real and synthetic garment embeddings share an embedding space F_G , which is optimized for both sets of model parameters (θ_{real} or θ_{spin}). This implies that both input garment styles (real and synthetic) are compatible with both output motion styles (dynamic or static spin). Critically, this property enables HoloGarment to mix and match input garment styles with output motion styles.

5.2.4 Real-World Garment Image(s)-to-360° Garment

Our implicit training approach enables us to train a robust garment embedding space F_G on diverse, large-scale garment video data that is also compatible with the NVS task. As a result, we can accomplish the desired implicit task of *real* image-to-360° NVS. Given a real garment image I_g^{real}

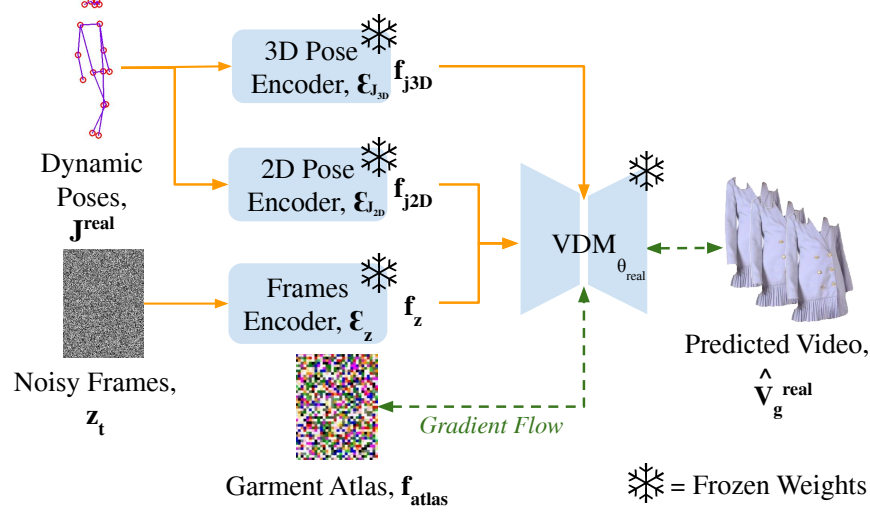


Figure 31: **Garment Atlas Finetuning** HoloGarment enables video-to-NVS by finetuning a garment-specific embedding, or “atlas”, on a real-world video. By utilizing this atlas during inference, HoloGarment generates photorealistic 360° novel views of the garment.

(task 1) and static spin pose sequence in a canonical A-pose J^{spin} (task 2), HoloGarment generates static 360° novel views of the input garment. During inference, the VDM operates with parameters θ_{spin} and denoises noisy frames z_t via iterative noise prediction:

$$\hat{e}_t = \text{VDM}_{\theta_{\text{spin}}}(z_t, t, f_g^{\text{real}}, f_j^{\text{spin}}) \quad (19)$$

Since $f_g^{\text{real}} \sim F_G$ (Eq. 18) is compatible with both θ_{real} and θ_{spin} , $\text{VDM}_{\theta_{\text{spin}}}$ generates consistent novel views of the real-world input garment style of task 1 in the output motion style of task 2: static, a-posed, and without occlusions, deformations, or wrinkling.

5.2.5 Video-to-360° Garment via Atlas Finetuning

As a result of our implicit training approach (5.2.3), the VDM’s garment embedding space F_G is shared between the 2D video animation and 3D NVS tasks. This enables another capability: finetuning a single garment embedding $f_g \sim F_G$ on a dynamic video (task 1) to run NVS (task 2). We call this finetuned garment-specific embedding the garment “atlas”, f_{atlas} , which can be leveraged

Algorithm 2: Garment Atlas Finetuning on a Dynamic Garment Video

Input: V_g^{real} : Input dynamic garment video

Output: f_{atlas} : The finetuned garment embedding.

Initialize:

- Freeze all parameters θ

- $f_{\text{atlas}} \leftarrow$ random embedding of shape f_g

for $i = 1$ **to** M **do**

- Sample frames $v_g \sim V_g^{\text{real}}$

- Sample timestep $t \sim \text{Uniform}(1, T)$

- Sample noise $\epsilon_t \sim \text{Gaussian}(0, I)$

- Compute poses $(J_{2D}^{\text{real}}, J_{3D}^{\text{real}})$ from v_g

- Compute pose embeddings f_j^{real}

- $z_t \leftarrow \text{AddNoise}(v_g, \epsilon_t, t)$

▷ Get noisy frames

- $\hat{\epsilon}_t \leftarrow \text{VDM}_{\theta_{\text{real}}}(z_t, t, f_{\text{atlas}}, f_j^{\text{real}})$

▷ Predict noise using VDM

- $\mathcal{L} = \|\hat{\epsilon}_t - \epsilon_t\|_2^2$

▷ Compute MSE loss

- $f_{\text{atlas}} \leftarrow \text{Update}(f_{\text{atlas}}, \mathcal{L})$

▷ Update garment atlas

end

for both tasks (Eq. 18). Note that this is in contrast with earlier subject finetuning methods [148], which finetune a model on the same task (text-to-image) as during inference (text-to-image).

The garment atlas finetuning strategy is shown in Figure 31. Initially, f_{atlas} is randomly initialized with same the shape as f_g . Then, freezing all other model parameters θ_{real} , f_{atlas} is finetuned on a specific garment video V_g for M iterations. Refer to Algorithm 2 for details. At inference, f_{atlas} replaces the original garment embeddings f_g ,

$$\hat{\epsilon}_t = \text{VDM}_{\theta_{\text{spin}}}(z_t, t, f_{\text{atlas}}, f_j^{\text{spin}}) \quad (20)$$

5.3 Experiments

In this section, we first describe our datasets (5.3.1), training and inference details (5.3.2), and evaluation metrics (5.3.3). Then, we evaluate our method on image-to-360° and video-to-360° novel view synthesis (NVS) (5.3.4), demonstrating quantitative and qualitative improvements over related methods (5.3.5) and ablated versions of our method (5.3.6). Garment animation results are included

in the appendix.

5.3.1 Datasets

We train our model on a combination of real-world and synthetic data. Our real-world dataset includes 17M crawled garment images and 52K garment videos, along with the UBC Fashion Video dataset [203], containing 500 train and 100 test videos. For synthetic data, we use 8,473 unique garment assets, from turboquid [3], objaverse [34], and other online sources. For each 3D garment, we render 32 views covering one full 360° orbit around the object center. Each image and video frame I is preprocessed using an in-house equivalent of Graphonomy [53] to compute the corresponding 2D person keypoints J_{2D} , garment segmentation I_g , and 2D pose of the garment image J_g . Each 2D pose J_{2D} is further preprocessed as a heat-map representation that is spatially-aligned with I . Mediapipe [122] is used to compute 3D person keypoints J_{3D} . During evaluation, each input pair consists of 1 or 3 real-world images or frames from a held-out dataset similar to (1) and a randomly selected pose sequence from (2), covering one full 360° spin.

5.3.2 Training and Inference Details

Training: We pretrain our base model without temporal layers on our custom garment image dataset for 1M iterations with batch size 8 [90]. Then, we train our full model with temporal layers for 589K iterations, approximately 3 days, on 16 TPU-v4’s. In this stage of training, we train jointly with 33% fashion image data, 33% fashion video data, and 33% novel views rendered from 3D garment assets, each with frame count 32 and resolution 512 x 384. For task (2) batches, 3D garment assets are rendered as 360-degree RGB spin videos, similar to [47]. However, different from [47], which uses camera position as conditioning, we compute J_{2D} and J_{3D} for both garment spins and real fashion videos, so that the motion representation is shared. Plus, 2D and 3D together encapsulate camera pose [121]. Finally, we finetune the model for an additional 50K iterations on 3D data only.



Figure 32: **Qualitative Results.** Our method generates 360° novel views of garments from single images, multiple images, or videos. Additional qualitative results are shown in the supplementary material.

Method	Custom Dataset				UBC Dataset			
	FID ↓	CLIP ↑	FVD ↓	SSIM ↑	FID ↓	CLIP ↑	FVD ↓	SSIM ↑
Garment3DGen	320	0.631	2477	0.002	310	0.638	2534	1.72e-5
Gemini 2.5 Flash Image ¹	156	0.836	–	0.700	137	0.880	–	0.742
SVC	130	0.855	1109	0.767	200	0.835	1144	0.745
CAT3D	152	0.861	1073	0.578	186	0.798	1137	0.625
ours _{video}	131	0.890	1088	0.730	128	0.872	1053	0.743
ours _{3D}	144	0.871	968	0.739	147	0.847	880	0.754
ours	128	0.872	875	0.729	127	0.881	880	0.771

Table 8: **Quantitative Comparisons.** HoloGarment outperforms related and ablated methods on all metrics. ¹Given that the Gemini 2.5 Flash Image model isn’t designed for temporal consistency, we chose to omit the FVD metric from our evaluation.

For both pretraining and training, we use an Adam optimizer with linearly decaying learning rate of $1e-4$ to $1e-5$ over a maximum of 1M iterations. We additionally add independent dropout for each conditional input with 10% probability per batch. After each forward pass, we compute the L2 loss on predicted noise ϵ_t at diffusion timestep t .

Inference: During inference, we use the DDPM sampler [76] with 1000 refinement steps to generate 32-frame videos. For evaluating our full and ablated models, we employ dual classifier-free guidance [20] with conditioning groups $(\emptyset, I_g, [J_{2D}, J_{3D}])$ and weights $(1, 5, 1)$.

5.3.3 Evaluation Metrics

We evaluate our method based on garment fidelity, multi-view consistency, and 3D realism. Garment fidelity is measured using the Fréchet Inception Distance FID [73] and CLIP [138] scores between the input garment images and the predicted garment images. For multi-view consistency and 3D realism, we compute the Fréchet Video Distance (FVD) [171] and structural similarity (SSIM) between our generated frames and ground-truth rendered 360° spin videos.



Figure 33: **Qualitative Comparisons.** HoloGarment demonstrates superior preservation of garment appearance, realism, canonical pose, and multiview consistency, as well as robustness to occlusions, compared to related works and the ablated versions.

5.3.4 360° Novel View Synthesis of In-the-Wild Garments

Image-to-360°: Figure 32 showcases qualitative results from our image-to-360° NVS method. HoloGarment synthesizes consistent and realistic novel views in a canonical target pose from single-view (top row) or multi-view inputs (middle row), even in the presence of occlusions, pose changes, and wrinkling in the input images. Moreover, HoloGarment handles a variety of garments, including tops, dresses, jackets, rompers, and pants.

Video-to-360°: To synthesize 360° novel views of a garment in a dynamic video, we finetune a garment embedding, which we call a garment “atlas” f_{atlas} , on a 128-frame real-world video for 500 iterations with batch size 32 and constant learning rate of $1e-3$. By only optimizing for f_{atlas} , we prevent undesired overfitting to the input video motion.

We showcase qualitative examples of our video-to-360° NVS method in the bottom row of Figure 32 and Figure 34. Our atlas finetuning strategy enables HoloGarment to consolidate an arbitrary number garment views, poses, and deformations into a unified 360° garment representation. We also show qualitative results of our single-view method on held-out synthetic 3D garment assets in Figure 35. Given a single view of a synthetic 3D asset, HoloGarment synthesizes plausible novel

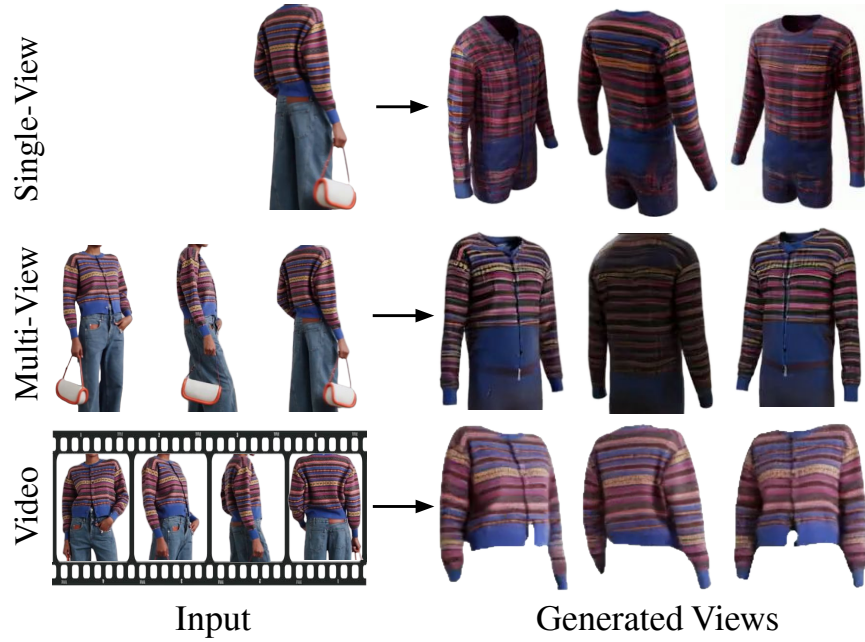


Figure 34: **Atlas Finetuning Ablation.** In contrast to single-view (*top row*) and multi-view conditioning (*middle row*), atlas finetuning on a video (*bottom row*) eliminates the dependency on input view selection by consolidating details from all video frames to improve garment texture details and multi-view consistency.

views that are consistent and retain high fidelity to the ground truth views.

5.3.5 Comparisons to State-of-the-Art

We compare our method to image-to-3D (Garment3DGen [151] and CAT3D [47]), image editing (Gemini 2.5 Flash Image [54]), and camera-controlled video generation (Stable Virtual Camera [214], Veo3 [58]) methods. Our results are presented in Figure 33 and Table 1. Qualitatively, HoloGarment produces superior results compared to Garment3DGen, CAT3D, and Stable Virtual Camera, and achieves visual quality on par with large, publicly available models like Gemini 2.5 Flash Image and Veo3. This is notable given that our model was trained on a significantly smaller dataset. Quantitatively, our approach consistently outperforms all compared methods across all metrics on both evaluation datasets (Table 8). Due to the high computational cost of large-scale evaluation, a full quantitative comparison with Veo3 is reserved for future work. Further discussion and implementation details for each method are provided in the appendix.

5.3.6 Ablations

Implicit Training Paradigm: As shown in Figure 33 and Table 8, our implicit video-and-3D training approach (ours) outperforms video-only training (ours_{video}) and synthetic 3D-only training (ours_{3D}) methods. Video-only training leads to the great photorealism and garment fidelity, but it fails to enable 3D-consistent motion generation or realistic novel views. Plus, the synthesized views contain holes where limbs or hair overlap with the garment in the input views.

3D-only training generates highly consistent and realistic garment spins, but exhibits poor garment fidelity and over-smoothed textures. Our jointly-trained model balances the benefits of video-only and 3D-only training, maintaining garment fidelity, photorealism, and multi-view consistency without holes.

Atlas Finetuning: We evaluate our atlas finetuning strategy quantitatively in Table 9 and qualitatively in Figure 34. Poorly-chosen input view(s) can drastically limit novel view realism. On the other hand, atlas finetuning enables our model to consolidate information from an arbitrary number of images, improving fidelity and realism. Compared to single-image conditioning, atlas finetuning improves performance across all metrics.

Method	FID ↓	CLIP ↑	FVD ↓	SSIM ↑
w/o Atlas	134	0.852	538	0.689
w/ Atlas	103	0.900	474	0.728

Table 9: **Atlas Finetuning Ablation.** We evaluate 40 results with and without atlas finetuning.

5.4 Limitations and Future Work

While our method improves over existing methods, it faces several limitations. Due to the limited diversity of the synthetic 3D garment dataset, HoloGarment struggles with unusual garment shapes (e.g. asymmetry or cut-outs). In Figure 36, we show two examples of failure cases of our method.

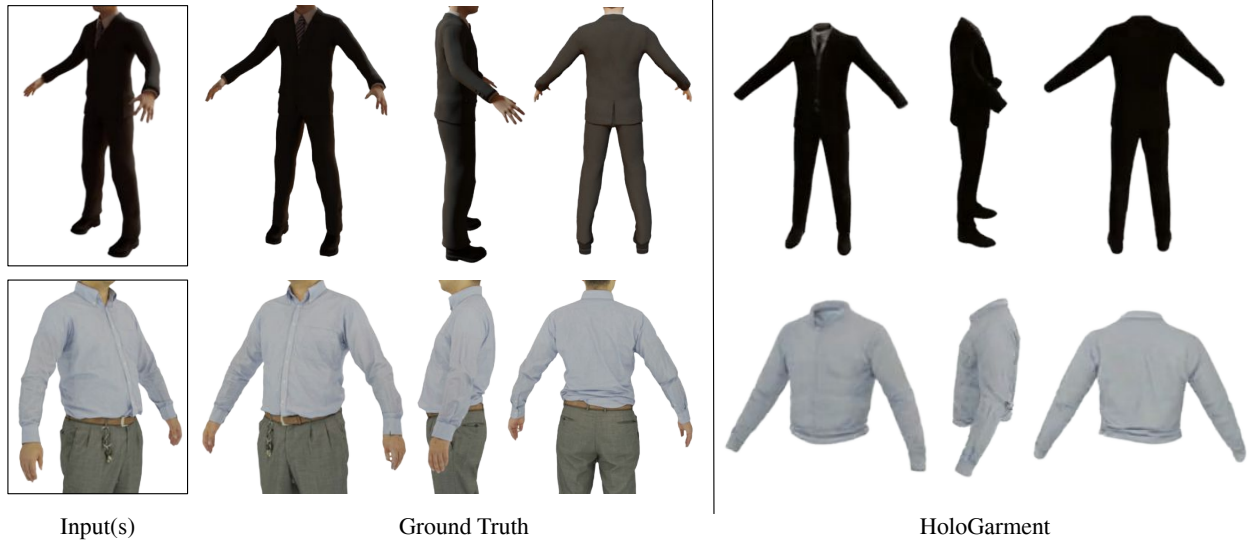


Figure 35: **Qualitative Results on Synthetic 3D Garments.** HoloGarment generates consistent 360-degree novel views from a single garment view that retain high-fidelity to their corresponding ground-truth synthetic 3D assets.



Figure 36: **Failure Cases.** Example failure cases of our method.

These include hallucinating bottoms for top-only garment inputs (*top*) and at times misrepresenting skirts as pants, especially when a slit is present in the input view (*bottom*). A larger synthetic garment dataset may remedy such issues. Other future work includes speeding up atlas finetuning (currently ~ 30 minutes on a single TPU) and increasing resolution via super-resolution network.

Conclusion

In this paper, we present HoloGarment, a method for synthesizing state-of-the-art novel views of garments in real-world images and videos. We introduce an implicit training scheme to optimize a video diffusion model for real-world garment image-to-360° novel-view synthesis (NVS) using a combination of large-scale 2D garment data and limited synthetic 3D garment assets. We further propose atlas finetuning, a strategy where a garment embedding, or “atlas”, is finetuned on a dynamic garment video to enable video-to-NVS capabilities.

Appendix

5.4.1 Additional Qualitative Results

In Figure 38, we show additional qualitative results from our method on image-to-360° few image-to-360° and video-to-360° generation.

5.4.2 Garment Animation

In Figure 37, we demonstrate HoloGarment’s ability to realistically animate real-world garments given an image and driving pose sequence (task 1). Due to the nature of the real-world image and video data, HoloGarment creates wrinkling and occlusions when operating on video data. Although the focus of our work is NVS, the ability for HoloGarment to perform well on the image animation task, is crucial for enabling video-to-NVS finetuning (Section 5.3.4).

5.4.3 Architecture Details

We show our overall video diffusion model (VDM) architecture in Figure 29. Our model is trained following v-prediction [150], where the model outputs the predicted noise ϵ'_t at timestep t , such that $z'_{t-1} = z_t + \epsilon'_t$. The $L2$ -loss is computed in ϵ space.

Input Preprocessing: The conditional inputs to our VDM are 1 or 3 segmented garment images I_g , their corresponding 2D poses J_g , and a sequence of driving 2D and 3D poses (J_{2D}, J_{3D}) . In the case of multiple input garment images, I_g is channel-wise concatenation of the segmented garment images. Additionally, I_g and its corresponding spatially-aligned 2D poses J_g^{2D} are concatenated channel-wise.

Conditioning Inputs: The noisy video z_t , garment signals $[I_g, J_g]$, and driving 2D poses J_{2D} are encoded by separate UNet encoders [217] into features f_z, f_g, f_{j2D} , respectively. The driving

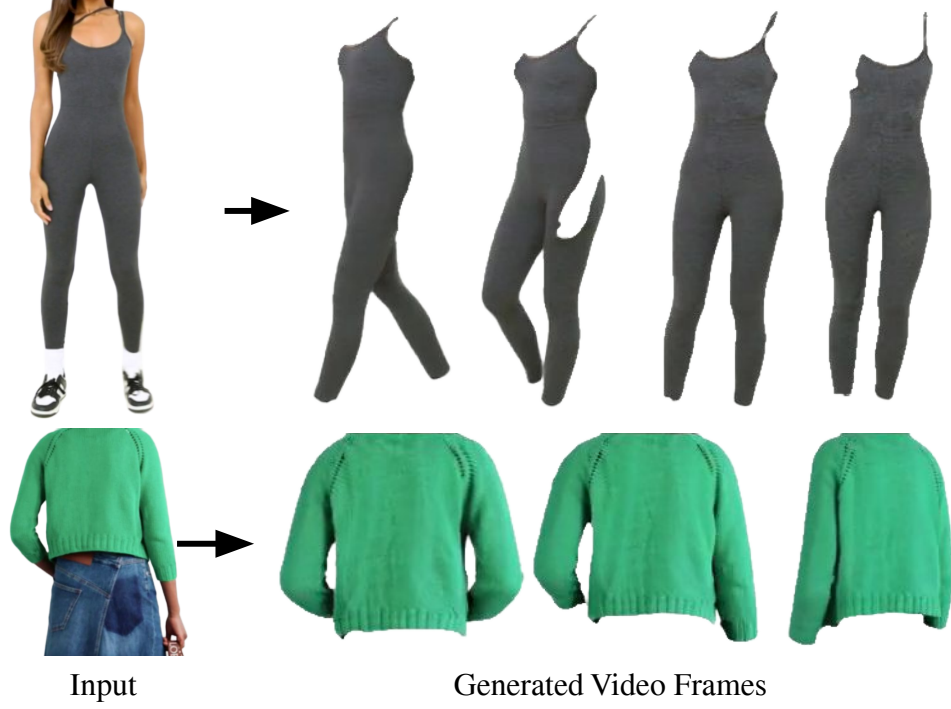


Figure 37: **Qualitative Animation Results.** HoloGarment generates realistic garment animations given a garment image and dynamic driving pose sequence.

3D poses J_{3D} are separately encoded by 4 dense layers into features f_{j3D} and reshaped to match f_{j2D} . Garment pose f_{J_g} is encoded by a single linear layer. The conditioning input embeddings f_g, f_{j2D}, f_{j3D} are processed by the DiT blocks [136] before the UNet decoder.

We concatenate the driving pose features, f_{j2D} and f_{j3D} , with the noisy video features f_z . The input garment image and pose features f_g are cross-attended with noisy video features f_z , in order to implicitly warp the input garment features to their target locations according to the driving poses [217], getting warped features f'_z .

UNet: Similarly to Fashion-VDM [90], we add 3D convolution, temporal attention, and temporal mixing blocks after the two lowest-resolution spatial layers in the UNet encoder and decoder. However, unlike Fashion-VDM, we duplicate these temporal blocks, such that one set only processes video batches and the other only processes 3D spin batches. We implement this switch in the network via conditional network branching. Note that only one branch of temporal blocks is activated at a

time. Finally, a UNet decoder decodes f'_z into the predicted noise ϵ_t .

Additional implementation details of our architecture include:

- The kernel sizes of our Conv2D and Conv3D blocks are $(3, 3)$ and $(4, 3, 3)$, respectively.
- Our 8 DiT blocks are implemented with 8 attention heads each and a hidden size of 2048 at feature resolution 32×24 .
- Our UNet encoders consist of 4 downsampling CNN blocks. Symmetrically, our UNet decoders consist of 4 upsampling CNN blocks.

5.4.4 Comparisons to State-of-the-Art Details

Gemini 2.5 Flash Image: We compare our method to Gemini 2.5 Flash Image [54] on garment novel view synthesis. For quantitative evaluation, we generate the front view of each input segmented garment using the prompt *"Generate a front-facing image of the garment in a-pose and without occlusions."* As Gemini 2.5 Flash Image is not designed for temporal consistency, we omit FVD in the quantitative comparison. For qualitative comparisons, we specify the target output angle of the garment in the prompt. Despite synthesizing high-quality, plausible novel views, Gemini 2.5 Flash Image is not designed for temporally-consistency, and struggles to generate smooth 360° novel views.

Veo3: We qualitatively compare with Veo3 [58] frame-to-video functionality through the Flow app of Google labs using the input segmented garment image as the input frame and the prompt *"Generate a 360-degree orbit of this garment in a-pose and without occlusions."* Veo3 generates high-quality, consistent orbits, but over-saturates the garment colors.

Stable Virtual Camera: We evaluate the official implementation of Stable Virtual Camera (SVC) [214] provided by the authors. We run SVC in single-image video generation mode on

the input segmented garment image, following an orbit trajectory for 32 target frames and using all other default parameters. In Figure 33, SVC fails to inpaint occluded regions (bottom row), synthesize plausible novel views (top row) or generate the garment in a canonical a-pose.

Garment3DGen: We follow the official Garment3DGen implementation. Garment3DGen does not provide a texturing tool, so we use a text-to-texture model [36], as suggested by the authors. We caption each input image using Gemini [167], then use FlashTex [36] to add texture to the 3D mesh. From the caption, we also determine the garment type and select the closest template mesh provided by the authors. As shown in Figure 33, the requirement of a template mesh severely limits Garment3DGen’s ability to generalize to different garment shapes. Plus, the generated textures show poor fidelity to the input garment image. In contrast, HoloGarment is independent of any template meshes or third-party texture generation methods, making it robust to diverse garment shapes and textures.

CAT3D: For comparisons with CAT3D, we follow the original authors’ single-image implementation. Figure 33 shows that CAT3D generates a flattened appearance in side and back views. Additionally, as shown in the bottom row, CAT3D is not robust to occlusions or pose variations, reproducing the input holes and wrinkling. Finally, unlike our method, CAT3D cannot warp the garment into a desired target pose, replicating the original garment pose.



Figure 38: **Additional Qualitative Results.** Demonstrating our method’s capability for 360° novel view synthesis of garments from a 1-3 input image(s) or a video. Results are best viewed in our supplementary video.

Chapter 6

FIT: A LARGE-SCALE DATASET FOR FIT-AWARE VIRTUAL TRY-ON



Figure 39: **The FIT Dataset.** We present FIT, a dataset and benchmark designed for *fit-aware* virtual try-on, featuring diverse garment fits (e.g., tight, loose) and precise size annotations. *Left:* Sample dataset triplets showing the conditioning garment image (*top*), the conditioning person image (*middle*), and the target try-on image (*bottom*). *Right:* Visualization of the corresponding person and garment measurement annotations. Backgrounds are removed for clarity.

The project in this chapter was completed in collaboration with Yuanhao Wang, Yingwei Li, and Ira Kemelmacher-Shlizerman. It will be published at SIGGRAPH 2026.

Introduction

The rising popularity of online shopping and social media has increased the demand for virtual try-on (VTO) systems. Driven by advances in generative models, recent VTO works [29, 63, 216] have achieved remarkable progress in synthesizing photorealistic try-on images. However, they often merely transfer garment *appearance* onto a person, neglecting to take into account the person

or garment sizes. As such, current VTO methods fail to address a fundamental question for any user: *"How will this garment actually fit me?"* This severely limits the accuracy and reliability of existing VTO tools to simulate a real-life try-on experience. Furthermore, it prevents users from experimenting with different sizes to achieve a desired fitted or oversized look. Consequently, there is significant commercial and research interest in developing a fit-aware VTO method.

Fit-aware try-on remains challenging due to the scarcity of real-world data annotated with precise person and garment measurements. Most existing VTO datasets [13, 27, 32, 50, 68, 112, 119, 129, 135, 215, 219] are curated by scraping catalog images from online retailers, which inherently lack "ill-fit" examples, i.e. the garment is too large or too small. Moreover, while some retailers provide size metadata, these annotations are often non structured and difficult to process at scale. Synthetic 3D garments created by artists offer an alternative, but this data suffers from limited scale and realism.

To fill this gap, we introduce FIT (**Fit-Inclusive Try-on**), the first large-scale, size-aware VTO benchmark explicitly designed to capture diverse upper-garment fit scenarios. By pivoting to a synthetic data generation pipeline (GarmentCode [99]), we overcome the limitations of real-world data collection. We procedurally create 3D garments with exact ground-truth measurements and simulate their drape onto a wide range of parametric bodies. This approach ensures not only size measurements, but also details like wrinkles, stretch, and garment coverage, are physically accurate. To close the domain gap between synthetic and real images, we employ a novel re-texturing pipeline designed to generate photorealistic textures for the synthetic renderings, while ensuring that the garment fit and body shape are preserved. To this end, we finetune a foundational image generation model, Flux.1-dev[16], to generate realistic person images from the synthetic normal maps and text-based garment descriptions.

Another critical bottleneck in VTO research is the lack of paired training data (identical subject and pose, different garments). Consequently, existing methods [29, 93, 95, 190, 216, 217] are forced to formulate VTO as a self-supervised reconstruction task, which limits real-world applications, or rely

on synthesized pseudo triplets [40, 63, 209], which suffer from inaccurate masking, identity loss, and size leakage. In contrast, our synthetic pipeline offers the unique advantage of controllability. We can simulate the same 3D subject in the same pose wearing multiple distinct garments, thereby generating ground-truth paired person data. Building on this insight, we further propose a novel framework for paired person image generation that ensures accurate 3D grounding and identity preservation.

Our dataset contains 1.13M training and 1K test samples of both men’s and women’s upper-garments. Each sample consists of a target try-on image, layflat garment image, a paired person image, as well as person and garment measurements. Our target try-on images cover diverse fit scenarios, including extreme ill-fits (e.g. a size 3XL draped onto a size XS person). By finetuning Flux.1-dev[16] with our custom dataset and a custom measurement encoder, we demonstrate a baseline fit-aware VTO model that accurately showcases garment fit.

To summarize, we present the following contributions:

1. We introduce **FIT**, the first large-scale dataset and benchmark explicitly designed for fit-aware virtual try-on, featuring precise metric annotations and diverse fit scenarios.
2. We develop a scalable synthetic data generation pipeline that leverages physics simulation and generative re-texturing to produce photorealistic try-on triplets with 3D grounding.
3. We demonstrate a novel, fit-aware virtual try-on model (**Fit-VTO**) that incorporates person and garment measurements to visualize not only garment appearance, but also accurate garment fit.

6.1 Related Works

Virtual Try-On Datasets. A primary bottleneck for fit-aware virtual try-on is the lack of datasets containing explicit size annotations or ill-fitting examples. Standard 2D benchmarks, such as

ViTON [68], ViTON-HD [27], DressCode [129], StreetTryOn [32], and LAION-Garment [63], predominantly feature well-fitted garments, lacking the diverse fit conditions required for size-aware training. While some datasets, including SIZER [170], SV-VTO [193], and Fit4Men [195] collect real-world samples for this purpose, they remain limited in scale and diversity. See Table 10.

Alternatively, 3D datasets [13, 112, 170, 215, 219] offer 3D models of clothed humans. However, extracting accurate garment measurements from raw meshes is often infeasible. GarmentCode [99] addresses this by introducing a domain-specific language for generating sewing patterns with explicit size parameters, enabling synthetic garment generation across varied garment and body sizes [97]. However, for extreme ill-fitting garment draping cases, GarmentCode tends to produce significant and frequent draping errors. Furthermore, raw 3D synthetic datasets [97, 106] generally suffer from their lack of realistic textures, which leads to poor real-world generalization. Although Sewformer [112] attempts to enhance realism via texture synthesis and SDEdit refinement, the results are still cartoonish and lack fit diversity. To bridge these gaps, we adapt GarmentCode for ill-fit scenarios, as well as introduce a novel pipeline for transforming synthetic GarmentCode renderings into photorealistic images.

Dataset	Realism	Ill-Fit	Measurements	Triplet	Scale
SV-VTO	✓	✓	✓	✓	1,524
SIZER	✓	✓	✓	✗	2,000
DeepFashion3D	✗	✗	✗	✗	2,078
ViTON-HD	✓	✗	✗	✗	11,647
Size4Men	✓	✓	✗	✗	13,000
LAION-Garment	✓	✗	✗	✓	60K
SewFactory	✓	✗	✓	✗	1M
GCD	✗	✗	✓	✗	115K
Ours	✓	✓	✓	✓	1.13M

Table 10: Comparison of related datasets. We compare FIT to several related datasets. For scale, we report the number of training images.

Image-Based Virtual Try-On. Image-based virtual try-on methods are generally categorized into two paradigms: *mask-based*, which utilize explicit segmentation maps to localize generation, and *mask-free*, which synthesize results directly without segmentation priors.

Mask-Based Methods These approaches formulate virtual try-on as a conditional inpainting task, where the target clothing region is masked and filled based on the garment image and human priors. Early warping-based works [27, 68] established a two-stage paradigm: warping the garment to the target body followed by refinement. Recent approaches have shifted toward single-stage diffusion-based architectures, achieving state-of-the-art photorealism [29, 32, 93, 190, 216, 217]. However, because these methods rely on inpainting within a fixed mask, they primarily focus on texture preservation and body alignment, largely neglecting the physical reality of garment sizing.

Mask-Free Methods. Another line of research [40, 41, 48, 49, 63, 85, 209] focus on mask-free architectures. Since real-world paired data is unavailable, these methods typically rely on generating “pseudo-triplets” via generative modeling to enable supervised training. A common strategy involves a “Teacher-Student” distillation framework, where a mask-based “teacher” model swaps garments on training images to generate synthetic ground-truth for a mask-free “student”. Similarly, Any2AnyTryOn [63] leverages a pretrained inpainting model to digitally replace garments in the try-on region. A fundamental bottleneck is that this training data is itself hallucinated, causing models to inherit the artifacts and geometric inconsistencies of the teacher. In contrast, our synthetic pipeline simulates actual draping dynamics on 3D bodies, yielding true ground-truth pairs with precise geometry and segmentation, effectively bypassing the error accumulation of 2D pseudo-triplet generation.

Fit and Size Control. While most VTO works ignore size, a few attempts have been made to incorporate fit information using geometric heuristics [24, 100, 193, 195]. For instance, [24] leverages clothing landmarks to transform garment size, while [100] uses body-to-clothing ratios to resize the conditioning segmentation maps. More recently, [193] and [195] introduce coarse fit conditioning based on descriptors (e.g., “tight” or “loose”). However, by relying on imprecise intermediate values or coarse labels, past methods struggle to generalize to complex poses and lack precise control. In contrast, our fit-aware model avoids noisy geometric heuristics by conditioning on exact metric measurements.

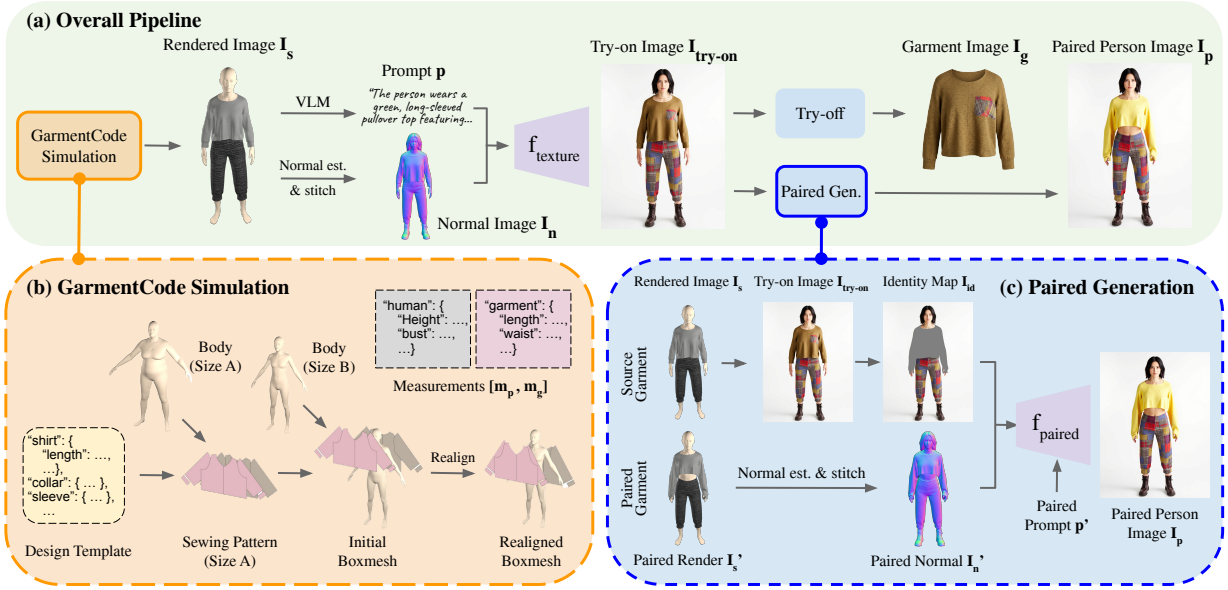


Figure 40: FIT dataset generation. (a) Overall pipeline: For each sample, we first simulate garment draping in 3D via GarmentCode, rendering a synthetic try-on image I_s (see (b)). Then, we generate a text prompt p (via VLM) describing the person and garment appearance, as well as a composite normal map I_n based on I_s . We use p and I_n to condition our re-texturing model f_{texture} to generate the photorealistic try-on image $I_{\text{try-on}}$. Finally, our model f_{paired} generates a paired person image I_p (see (c)) and a VLM synthesizes the corresponding layflat garment I_g . (b) GarmentCode simulation: Given a garment design template, we compute a sewing pattern with measurements m_g for a specific body size A . Then, we cross-drape the pattern onto a different target body of size B with person measurements m_p , using box-mesh realignment to prevent simulation failures. (c) To generate a paired person image (same person and pose, different garment), we start with a paired rendered image I'_s containing a *different* garment than in $I_{\text{try-on}}$ draped onto the *same* body. Next, we derive an identity map I_{id} by masking out the combined source and paired garment regions in $I_{\text{try-on}}$. Conditioned on I_{id} , the paired normal map I'_n , and a paired prompt p' , f_{paired} generates I_p .

6.2 Fit-Inclusive Try-on (FIT) Dataset

In this section, we describe the construction of the FIT dataset. We first report the dataset statistics in Section 6.2.1. We then detail our data generation pipeline, illustrated in Figure 40, which consists of the following steps: (1) procedurally generating garment assets with measurements m_g and simulating their drape across diverse sizes of bodies with measurements m_p via GarmentCode (Section 6.2.2); (2) transforming the synthetic renderings I_s into photorealistic try-on images $I_{\text{try-on}}$ via a geometry-preserving re-texturing framework (Section 6.2.3); (3) leveraging identity conditioning to generate a paired person reference image I_p featuring the same person wearing a

different garment (Section 6.2.4); and (4) synthesizing the corresponding layflat garment image I_g using an off-the-shelf VLM model [55] (Section 6.2.5).

6.2.1 Dataset Statistics

Our dataset consists of 1,137,282 training and 1000 test samples, each consisting of $(I_{\text{try-on}}, I_p, I_g, m_p, m_g)$. Our data covers 168 distinct body shapes (82 men’s, 86 women’s) in sizes XS-3XL, 528 body poses, as well as 158,483 unique top and garment designs. Our dataset covers a diverse range of fits, from loose to tight fits. We provide a histogram of each person/garment size combination in the appendix. The test dataset is balanced to match the overall distribution over gender, body sizes, and person/garment size combinations. See Figure 39 and the appendix for examples of our dataset.

6.2.2 GarmentCode Simulation

GarmentCode [99] is a parametric programming framework that enables the procedural generation and draping of 3D garment patterns, allowing for precise control over sizing and design details.

To generate try-on images with diverse fits, we implement a cross-draping strategy. We begin by sampling various garment templates and human body models with known measurements m_p . From a garment template, we generate sewing patterns fitted to multiple human bodies of varying sizes. We then simulate draping these sewing patterns onto a single target human model via GarmentCode’s custom implementation of Warp [124], thereby creating realistic ”tight” and ”loose” fit scenarios. However, direct cross-draping initially fails because the 3D box-mesh specified by standard sewing patterns is aligned with its original target body, causing severe misalignments when applied to a new body. We address this by explicitly realigning the initial box-mesh panels to the target mesh position before simulation. Please refer to the appendix for details. Furthermore, GarmentCode’s default implementation stitches top and bottom garments together into a unified mesh, preventing the appearance of ”tucked-out” shirts. We modify this behavior to drape the top and bottom garments

in two separate steps (typically simulating the bottom garment first) to ensure proper layering and realistic interactions between items. The draped 3d mesh is then reposed and rendered with different person poses to form a synthetic rendering image I_s .

Our procedural framework allows us to programmatically extract precise ground-truth garment measurements m_g in centimeters directly from the 2D sewing pattern specifications. We focus on five critical metrics used in standard sizing: garment length (high point shoulder to hem), bust circumference (width), and sleeve length for tops; and waist and out-seam length for bottoms. We also derive four key body measurements directly from GarmentCode’s parametric body model: height, bust, waist, and hips.

6.2.3 Synthetic-to-Photorealistic Retexturing

Our re-texturing pipeline is designed to transform the synthetic rendering I_s into a photorealistic image while strictly preserving the geometry of both the garment and the subject. Due to the lack of paired synthetic-to-real training data, we utilize surface normal maps as a geometry-preserving bridge between domains. Specifically, we finetune a diffusion model, f_{texture} (based on Flux.1-dev [16]), to synthesize photorealistic textures conditioned on an input normal map I_n and a text prompt p . f_{texture} is trained on real-world images with the following objective:

$$\hat{I}_{\text{try-on}} = f_{\text{texture}}(I_n, p) \quad (21)$$

where $I_n = N(I_{\text{try-on}})$ represents the normal map extracted from an off-the-shelf estimator N [92].

Despite utilizing normal maps, a significant domain gap persists between real-world and synthetic data. First, synthetic renderings I_s lack anatomical details, featuring bald heads and bare feet. To address this, we employ a composite refinement strategy: we prompt Nano Banana Pro [56] to inpaint realistic facial features, hair, and footwear onto I_s , estimate the normals of this enhanced image, and stitch the resulting head and feet regions onto the original synthetic normal map. This

ensures realistic semantic cues while leaving the body and garment geometry untouched. Second, Similarly, synthetic meshes lack intricate surface details, such as pockets, buttons, and seams. We observe that our model f_{texture} successfully inpaints these details when guided by appropriate text prompts. Similarly, due to GarmentCode’s limited controllability of material, the synthetic garments exhibit uniform, smooth fabric. To increase fabric diversity, we sample from 72 fabric types (e.g. leather, cotton, silk) and inject it into the text prompt. We further align the domains by augmenting the training data with random normal map blurring. This simulates the smoothness of synthetic normal maps and improves generation quality.

6.2.4 Paired Person Reference Image Generation

Our synthetic framework enables the generation of ground-truth paired data by exploiting procedural controllability. By fixing the subject’s shape and pose while draping two distinct garments, we obtain pairs of synthetic renderings (I_s, I'_s) , normal maps (I_n, I'_n) , garment masks (m_g, m'_g) , and prompts (p, p') .

First, we generate the primary try-on image $I_{\text{try-on}} = f_{\text{texture}}(I_n, p)$ using the re-texturing pipeline described in Section 6.2.3. Next, to synthesize the paired reference image I_p , we employ a conditional inpainting model f_{paired} :

$$I_p = f_{\text{paired}}(I_{\text{id}}, I'_n, p'), \quad (22)$$

where I_{id} represents the identity map, defined as $I_{\text{id}} = I_{\text{try-on}} \odot (\neg m_g \cap \neg m'_g)$. This operation preserves the skin and background from the try-on image while masking out the regions occupied by both the source and paired garments. Essentially, f_{paired} serves as a geometry-guided inpainter. To train f_{paired} , we utilize real human images, creates identity maps by estimating garment masks and applying random dilation to mimic the dual-garment masking seen at inference. Additionally, we limit our scope to upper-body try-on, hence we enforce identical bottom garment geometry

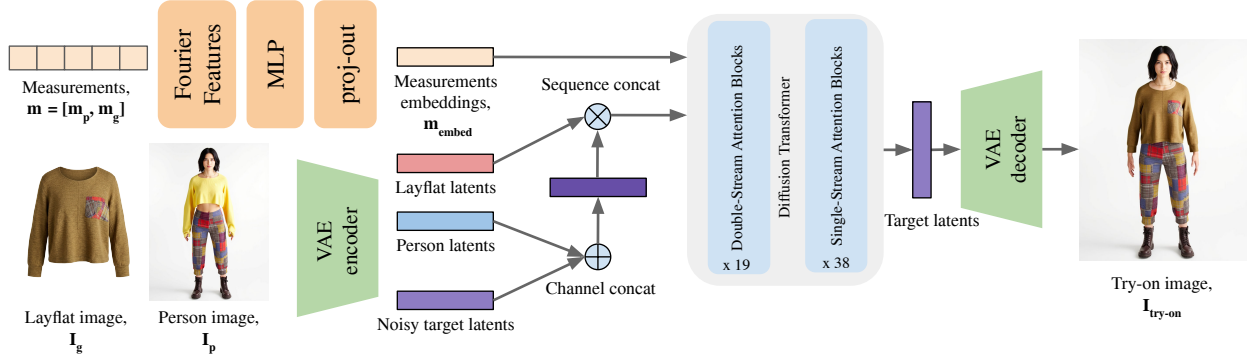


Figure 41: Fit-VTO architecture. Our architecture is a flow-based diffusion model based on Flux.1-dev [16] and finetuned with LoRA [81]. FiT-VTO generates a try-on image $I_{\text{try-on}}$ given a layflat garment image I_g , paired person image I_p , and person-garment measurements $m = [m_p, m_g]$. First, image inputs I_g and I_p are encoded into latents separately through a pre-trained VAE encoder. We replace the text embeddings in Flux.1-dev with custom measurement embeddings m_{embed} computed from m . Person latents are channel-concatenated with the noisy target latents, while layflat latents and m_{embed} are sequence-wise concatenated with z_t . After processing through the diffusion transformer, clean latents are decoded by the VAE decoder.

across pairs during simulation. In practice, we train a unified model for f_{texture} and f_{paired} following Eq. 22, but randomly dropping out I_{id} .

6.2.5 Layflat Image Generation

Motivated by the impressive image synthesis capability of Nano Banana Pro [56], we use it as an off-the-shelf virtual try-off model to generate a layflat garment image I_g from $I_{\text{try-on}}$. Please refer to the appendix for the exact prompts used.

6.3 Fit-Aware Virtual Try-On

Given an image I_p of person p , a garment image I_g of target garment g , target garment measurements m_g , and person measurements m_p , our Fit-VTO model f_{vto} synthesizes the predicted try-on result $\hat{I}_{\text{try-on}}$ of person p wearing g according to the measurements m_p and m_g .

$$\hat{I}_{\text{try-on}} = f_{\text{vto}}(I_p, I_g, m_p, m_g) \quad (23)$$

6.3.1 Dataset Preparation

To increase the robustness of our model to diverse, real-world garments and poses, we crawled 330,559 online fashion images and their corresponding layflat garment images I_g , to augment our FIT training dataset. Since ground-truth measurements are not available for online images, we set the measurements to null values (-1). For FIT data samples, all measurements m_p , m_g are normalized between 0 and 1.

6.3.2 Architecture

Our architecture (Figure 41) is a flow-matching diffusion model x_θ represented as:

$$\hat{v}_t = x_\theta(z_t, t, I_p, I_g, m_p, m_g) \quad (24)$$

where z_t is the noisy ground-truth image x_0 at diffusion timestep t and $\hat{v}_t$ is the predicted velocity. The model x_θ is trained to satisfy the consistency constraint where $\hat{v}_t$ approximates ground-truth velocity $v_t = x_0 - z_0$.

Our network x_θ is finetuned from the pretrained Flux.1-dev text-to-image model [16]. FLUX.1-dev is a powerful, 12 billion parameter text-to-image generator that employs a rectified flow formulation and a Multi-modal Diffusion Transformer (MMDiT) backbone for efficient, high-fidelity image synthesis. We finetune only the lightweight LoRA parameters, keeping the majority of the original model weights frozen.

Person and Garment Conditioning. We condition the model on paired person image I_p and garment image I_g . Since the I_p is pixel-aligned to the noisy target image z_t , we concatenate latents from I_p and z_t channel-wise. Since I_g latents need to be warped to z_t , we concatenate them along the sequence dimension after packing.

Measurement Conditioning. To condition on person and garment measurements, we remove the

CLIP and T5 text conditionings for Flux.1-dev and instead condition with measurement embeddings from our custom measurement encoder $\mathcal{E}_m$. We first concatenate person measurements m_p with garment measurements m_g into a measurement vector $m = [m_p, m_g] \in R^7$. Then, we compute the Fourier Feature Embeddings for each measurement with 8 Fourier frequency bands, mapping $m \rightarrow m_{\text{embed}} \in R^{7 \times 16}$. These embeddings are further processed by an MLP and projected to the hidden dimension R^{3072} of the MMDiT. Our model is conditioned on m_{embed} with positional encodings for each measurement via cross-attention, replacing the T5 text conditioning in the single-stream and double-stream blocks.

6.4 Experiments

We describe details of experiments in this section. We quantitatively and qualitatively evaluate the quality of our synthetic triplet data and demonstrate the effectiveness of our baseline fit-aware VTO model against state-of-the-art methods.

6.4.1 Implementation Details

For synthetic data generation, we initialize our re-texturing model from the pretrained Flux.1-dev [16] checkpoint and only finetune with LoRA layers [81] with rank 64 and alpha 64. The model is trained on a custom dataset of 50k real person images (see appendix for details). We adopt Prodigy optimizer with learning rate 1.0 and weight decay factor 0.01. The training is done on 8 H200 GPUs with a total batch size of 64 and 5k training iterations (1 day).

Our baseline VTO model initialized from Flux.1-dev checkpoint and finetuned using LoRA layers [81] with rank 128 and alpha 128. The measurement encoder is zero-initialized for stable early training. We finetune our model for 2M iterations on a mix of FIT training dataset and real-world images. The learning rate is 10^{-4} with 1000 warm-up steps and batch size is 64. All training is done on 64 TPU-v5's for 2 days. At inference, we set guidance scale to 1.0 and number of inference

steps is set to 50. We keep the same inference scheduler as the base Flux.1-dev release.

6.4.2 Evaluation Baselines, Datasets & Metrics

Paired Image Generation Evaluation. In this work, we propose a novel framework for generating pseudo-ground-truth paired-person images to enable mask-free VTO training. We benchmark against three baseline strategies: (1) **VLM-based**, which prompts Large Vision Language Models to swap garments while preserving context; (2) **VTO-based**, which utilizes off-the-shelf virtual try-on models for garment transfer; and (3) **Inpainting-based**, which replaces masked garment regions via generative inpainting. We implement these baselines using Nano Banana Pro, CatVTON [29], and FLUX-Controlnet-Inpainting [31].

To quantify how well the paired image I_p preserves the identity of the original $I_{\text{try-on}}$ in non-garment regions (i.e., background, head, and limbs), we compute the Masked L1 Distance $\mathcal{L}_{\text{id}}$:

$$\mathcal{L}_{\text{id}} = \frac{1}{N} \sum |(I_p - I_{\text{try-on}}) \odot M|, \quad (25)$$

where $M = \mathbf{1} - (m_g \cup m'_g)$ represents the binary mask of the non-garment regions, and N is the number of valid pixels in M . We randomly sampled 1000 cases from the dataset to compute this metric. The pixel values are between 0 and 255.

Fit-Aware VTO Evaluation. We compare our method qualitatively and quantitatively to Any2AnyTryon [63], Nano Banana Pro [56], COTTON [24], IDM-VTON [28], and ablated versions of our method. We provide implementation details about related methods in the appendix. We evaluate on the VITON-HD test dataset [27] to measure general try-on accuracy and the FIT test dataset to evaluate fit-aware try-on accuracy. For VITON-HD, we generate paired-person images according to Section 6.2.4.

We compute common VTO metrics – SSIM [181], FID [73], LPIPS [208], KID [15] – to evaluate image similarity between ground-truth and synthesized try-on images. We also implement a custom

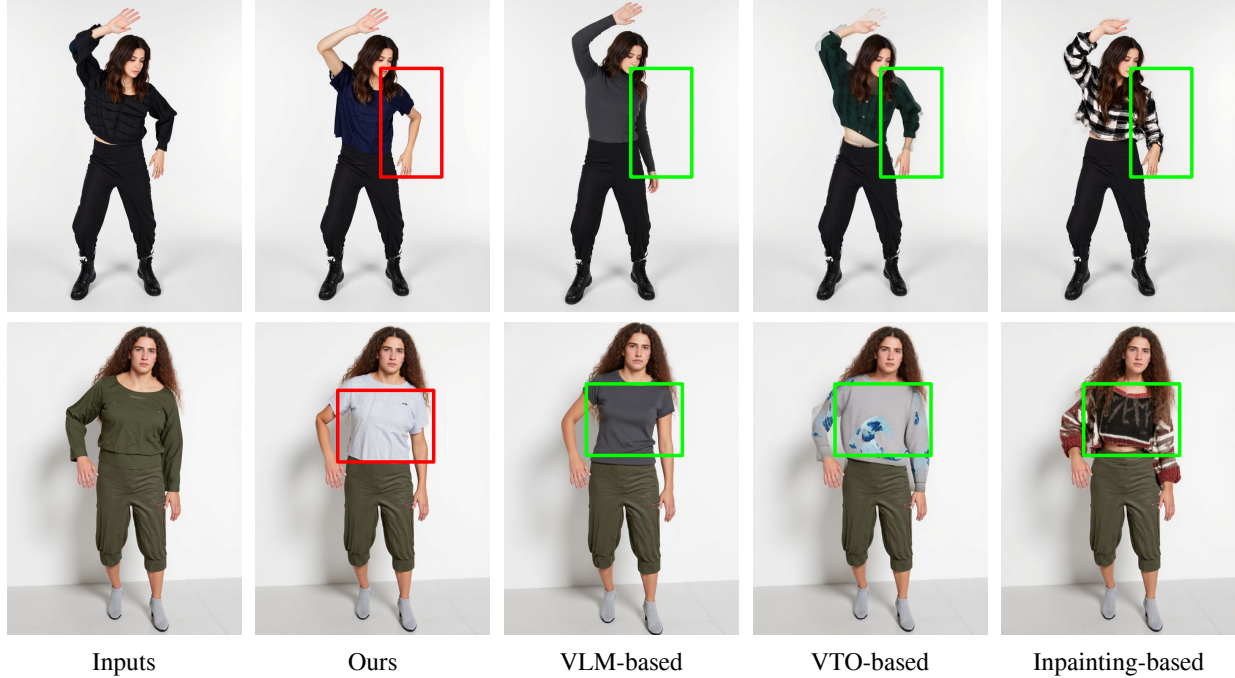


Figure 42: Paired Image Generation Comparison. VLM methods struggle with pose and shape preservation, while VTO and inpainting baselines introduce artifacts. Our approach yields highly consistent paired data.

metric (IoU), specifically designed for measuring size fidelity for the FIT dataset. IoU measures the Intersection-Over-Union of the garment mask in synthesized try-on image and the ground truth. We do not compute IoU for VITON-HD, as this dataset does not provide any size conditioning.

6.4.3 Paired Image Evaluation Results

We present a qualitative comparison of the generated paired-person images in Figure 42. Despite their impressive editing capabilities, VLM-based methods fail to guarantee identity preservation in non-garment regions (e.g., the left arm pose deviation in the top row). Furthermore, they often disregard the underlying body shape within the garment region (e.g., the inconsistent chest volume in the bottom row). Similarly, the VTO and Inpainting-based baselines introduce significant visual artifacts and struggle to maintain geometric consistency. In contrast, our approach achieves near-perfect identity and body shape preservation by explicitly conditioning on the identity map and ground-truth normals. Quantitative analysis confirms our visual findings: our method achieves

an $\mathcal{L}_{id}$ of 1.61, significantly outperforming VLM-based (4.45), VTO-based (2.29), and Inpainting-based (3.91) baselines. These results demonstrate that our pipeline successfully generates highly consistent paired data essential for robust VTO training.

	VITON-HD				FIT Dataset				
	SSIM $\uparrow$	FID $\downarrow$	LPIPS $\downarrow$	KID $\downarrow$	SSIM $\uparrow$	FID $\downarrow$	LPIPS $\downarrow$	KID $\downarrow$	IOU $\uparrow$
Any2AnyTryon [63]	0.758	14.186	0.152	2.413	0.819	25.059	0.209	3.939	0.783
Nano Banana Pro [55]	0.552	11.344	0.501	<u>0.624</u>	0.785	19.926	0.166	1.676	0.792
COTTON [24]	0.615	39.117	0.349	11.397	0.759	29.716	0.207	6.269	0.739
IDM-VTON [28]	0.849	9.115	0.077	0.471	0.739	31.229	0.246	6.819	0.789
Ours _{no FIT}	0.817	11.499	0.103	0.639	0.852	16.427	0.095	0.849	0.844
Ours _{text}	0.763	11.367	0.134	0.766	0.911	11.624	0.054	0.576	0.932
Ours _{FIT only}	0.732	14.651	0.192	1.061	<u>0.912</u>	<u>11.248</u>	<u>0.052</u>	<u>0.532</u>	<u>0.952</u>
Ours	0.817	11.391	0.102	0.651	0.914	10.381	0.050	0.144	0.955
Ours_{ft_vitonhd}	<u>0.833</u>	<u>9.320</u>	<u>0.087</u>	0.670	0.846	17.041	0.096	1.724	0.908

Table 11: Quantitative comparisons. We compare Fit-VTO to related methods and ablated versions of our method. Ours_{ft_vitonhd} refers to our method finetuned with VITON-HD training data. **Bolded** and underlined values indicate the best and second-best scores per column, respectively.

6.4.4 Fit-VTO Qualitative Results

We showcase qualitative results of Fit-VTO on the synthetic FIT test dataset in Figure 43. Our method synthesizes high-quality try-on images that maintain high fidelity to the person identity and garment appearance, while accurately reflecting realistic garment fit with respect to the person and garment measurements. Fit-VTO handles diverse fit cases, including tight fit, perfect fit, and loose fit. Our Fit-VTO method also generalizes to real-world images (VITON-HD [27]) without measurements, as shown in the bottom two rows of Figure 44.

To evaluate Fit-VTO’s ability to independently model person and garment size, we showcase the results of varying the garment size while keeping the person fixed in Figure 45. Fit-VTO realistically adjusts garment fit with respect to both garment and person sizes, while maintaining consistent person and garment appearance. In the appendix, we evaluate independent controllability of individual garment measurements and show that garment size controllability extends to images of real-world humans.

In Figure 44, we qualitatively compare our method to related works. Despite accurate texture warping, Any2AnyTryon [63], Nano Banana Pro [56], COTTON [24], and IDM-VTON [28] fail to accurately portray accurate garment fit according to the person and garment sizes. Nano Banana Pro, for example, produces aesthetically pleasing images, but lacks precise measurement grounding, leading to incorrect fit (e.g., overly loose or tight) relative to ground-truth (last column). COTTON also suffers from severe boundary artifacts due to errors in its pre-processing pipeline. In contrast, Fit-VTO respects person and garment measurements and visualizes accurate garment appearance.

6.4.5 Fit-VTO Quantitative Results

We report quantitative metrics against related methods in Table 11. Fit-VTO excels in nearly all VTO metrics on both real-world VITON-HD and synthetic FIT datasets. On VITON-HD, IDM-VTON’s attains slightly stronger results, which is partially explained by its training directly on VITON-HD, whereas our base method (“ours”) is not. However, with additional VITON-HD finetuning, our method achieves comparable performance to IDM-VTON on VITON-HD. On the FIT dataset, Fit-VTO achieves superior size-aware IoU score, even compared to size-conditioned COTTON. These results indicate that our method effectively delivers high appearance fidelity, as well as incorporates size information for try-on.

6.4.6 Ablations

In our ablations, we evaluate the impact of our FIT dataset, measurements encoder, and real-world data supervision. As summarized in Table 11, we compare (1) training without FIT data and only online fashion images ($\text{Ours}_{\text{no FIT}}$), (2) replacing our measurement encoder with pretrained T5 [139] and CLIP [138] text encoders used in the original Flux.1-dev [16] model ($\text{Ours}_{\text{text}}$), and (3) training with FIT data only ($\text{Ours}_{\text{FIT only}}$). See Figure 44 for qualitative comparisons.

The FIT-only model performs well on FIT data, but degrades considerably on VITON-HD, as further

evidenced in the bottom two rows of Figure 44. We attribute this to overfitting to the garments and poses FIT dataset, highlighting the importance of real-world training data for generalization. Conversely, the model trained without FIT data performs well on VITON-HD with respect to SSIM, FID, and LPIPS, but fails to model person-garment size relationships, as indicated by the significantly lower size-aware IoU. This demonstrates that real-world data with measurements predicted by VLM alone are insufficient for learning accurate fit. The text-only model performs moderately well on VITON-HD – likely because it better preserves the pretrained knowledge from Flux.1-dev – yet, this model fails to encode precise measurement information and exhibits a low IoU score. This indicates that pretrained text encoders are not well-designed to represent structured numerical size inputs. Row 1 in Figure 44 corroborates these findings: $\text{Ours}_{\text{no FIT}}$ and $\text{Ours}_{\text{text only}}$ exhibit significant errors in representing ill-fitting garment size, while our full method accurately show accurate garment fit.

Our full model achieves the best balance across both benchmarks, performing on par with the strongest variants on each domain while delivering high size-aware IoU on FIT. These results confirm that combining FIT supervision, real-world data, and our measurement encoder yield a model that is both robust to real imagery and sensitive to garment–person size relationships.

6.5 Scope and Limitations

Our work serves as a proof-of-concept demonstrating that synthetic data generation, grounded in physics-based simulation, is a promising way to overcome the scarcity of size-annotated data in virtual try-on. However, as an initial exploration, our current scope is intentionally constrained. We focus exclusively on upper-body garments in standardized front-facing views (full-body or cropped) and casual poses, thereby avoiding the complicated collision dynamics. Additionally, the structural diversity of our dataset is bounded by the capabilities of the GarmentCode engine, limiting our study to simple structural designs rather than complex, multi-layered apparel. Despite these constraints, our results validate the core hypothesis: that synthetic, physics-informed supervision can teach



Figure 43: Qualitative results. We show examples of Fit-VTO results on the synthetic FIT test dataset. Fit-VTO respects person and garment inputs, while also synthesizing realistic garment fit based on person and garment measurements (zoom in for details). For brevity, we approximate the full measurements with a size label (XS-3XL). See the appendix for our size categorization chart.



Figure 44: Qualitative comparisons. We compare Fit-VTO to related and ablated methods using synthetic FIT test data (top two rows) and real-world VITON-HD data (bottom two rows). Overall, our method best depicts the most accurate garment appearance and fit.

generative models to respect precise metric sizing. We believe this synthetic-to-real paradigm establishes a foundation for future research to scale up to complex, in-the-wild scenarios.

We also identify two specific technical limitations. First, accurately representing the degree of tightness in the data is challenging. While the feeling of wearing a tight garment or a very tight garment may vastly differ, the simulated appearance is almost identical – fitted to the skin. As such, our dataset and VTO model do not represent varying degrees of tightness well (see left column of Figure 43). Furthermore, our Fit-VTO model is sensitive to correlations in measurements, limiting its ability to independently alter single measurements. For example, an increase in width frequently leads to a slight increase in length and sleeve length, as well.



Figure 45: Independent size control. Fit-VTO realistically visualizes garment fit across various sizes on a fixed person size.

Conclusion

In this paper, we introduce **FIT**, the first large-scale dataset and benchmark for *fit-aware* virtual try-on (VTO) consisting of over 1.13M samples. We also present **Fit-VTO**, a novel fit-aware VTO model designed to leverage FIT’s rich person-garment size annotations. Across extensive comparisons to related and ablated methods, Fit-VTO demonstrates a clear advantage in modeling accurate garment fit, according to person and garment measurements.

Future Work: Our immediate next steps involve expanding the dataset scope beyond tops to include lower-body and full-body garments (e.g., pants, dresses). Additionally, while our current dataset supports basic pose variation, scaling up the diversity of poses and camera viewpoints remains a key objective to ensure robust performance across complex, real-world inputs.

Appendix

6.5.1 Additional FIT Dataset Details

In this section, we provide additional details and statistics about the FIT dataset.

Size Categorization. In our figures, we frequently abbreviate the full person-garment measurements $[m_p, m_g]$ with coarse size labels for the person and garment such as XS, L, XL. We determine these size labels based on average measurement ranges as shown in Table 14. Note that the grouping of size measurements into coarse size labels (e.g. XS, L, XL) are only used for visualization and grouping purposes, not Fit-VTO training or evaluation.

Garment Fit Distribution. Our FIT dataset covers a diverse range of fit scenarios. In Figure 46, we plot the distribution of person/garment size pairings as a histogram, showing that every reasonable fit scenario is represented, from very tight (e.g. size “XL” person wearing a size “M” garment) to very loose (e.g. size “XS” person wearing a size “2XL” garment). Implausible fit pairings where the garment is more than 3 sizes smaller than the person (e.g. size “3XL” person wearing a size “XS” garment) are not included.

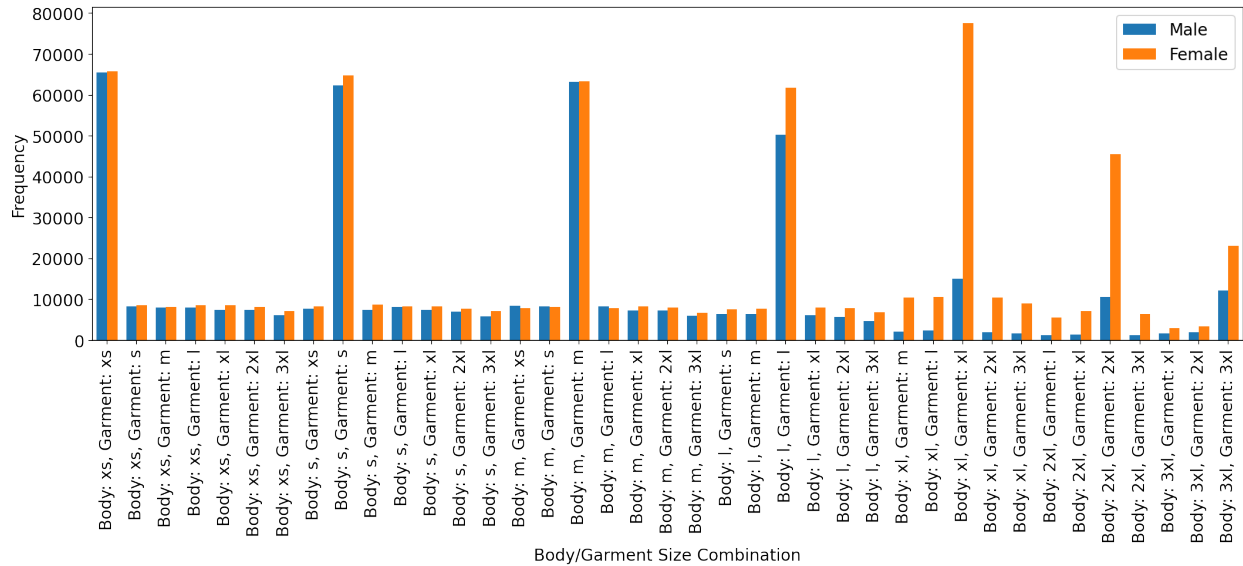


Figure 46: Garment fit distribution. We plot the frequency of each (body size, garment size) pairing in our dataset according to the size classification in Table 14.

Measurement Statistics. In Table 12 and Table 13, we report the minimum, mean, maximum, and standard deviations of the measurements for our body and garment meshes in the FIT dataset, respectively. Our dataset covers a wide range of body shapes and garment sizes.

Measurement	Men’s (min, mean, max, std)	Women’s (min, mean, max, std)
Bust	(87, 101, 141, 10)	(83, 100, 136, 13)
Height	(155, 174, 194, 8.0)	(151, 170, 196, 9.0)
Hips	(88, 101, 125, 6.0)	(89, 104, 127, 10)
Waist	(70, 86, 141, 13)	(61, 85, 130, 17)

Table 12: Body size statistics. We report the min, mean, max, and standard deviation of our garment measurements in cm.

Measurement	Men’s (min, mean, max, std)	Women’s (min, mean, max, std)
Width	(77, 112, 169, 16)	(75, 110, 169, 16)
Length	(29, 53, 76, 8.0)	(29, 51, 76, 7.5)
Sleeve Length	(0.0, 30, 79, 17)	(0, 29, 79, 17)

Table 13: Garment size statistics. We report the min, mean, max, and standard deviation of our garment measurements in cm.

	Bust		Waist		Hips	
Size	Men’s (min, max)	Women’s (min, max)	Men’s (min, max)	Women’s (min, max)	Men’s (min, max)	Women’s (min, max)
XS	(86, 91)	(79, 84)	(71, 76)	(58, 64)	(91, 96)	(84, 89)
S	(91, 96)	(86, 89)	(76, 81)	(66, 67)	(96, 101)	(91, 94)
M	(96, 101)	(90, 95)	(81, 86)	(71, 75)	(101, 106)	(97, 102)
L	(101, 106)	(96, 104)	(86, 91)	(86, 91)	(106, 111)	(106, 111)
XL	(106, 117)	(105, 116)	(91, 103)	(85, 97)	(111, 120)	(112, 121)
2XL	(111, 127)	(112, 125)	(96, 115)	(91, 105)	(120, 134)	(120, 130)
3XL	(127, 147)	(117, 135)	(115, 137)	(107, 127)	(134, 145)	(125, 137)

Table 14: Body size categorization statistics. We report the min and max for each body measurements and size label in cm.

Additional Dataset Examples. We present additional examples of try-on triplet data in our dataset in Figure 48.

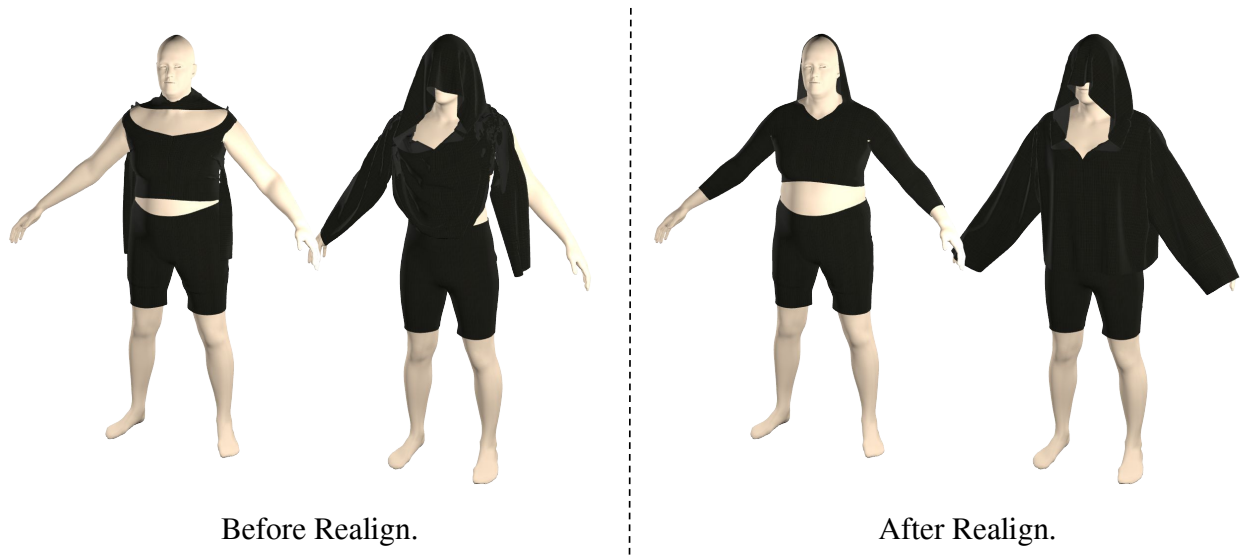


Figure 47: During cross-body draping, the initial boxmeshes are often misaligned with the target human models, causing draping failures (left). We explicitly realign the boxmesh to ensure successful simulation (right).

6.5.2 Additional Details on Data Generation Pipeline

Cross Draping vs. Linear Size Change. The standard GarmentCode pipeline computes sewing patterns based on a design template and target body parameters, yielding a 3D garment that is well-fitted to the wearer. To generate ill-fitting examples (e.g., oversized or undersized), a naive baseline would be to linearly scale the garment parameters. However, this fails to capture real-world sizing dynamics, as garment grading rules are non-linear and distinct from simple geometric scaling. To address this, we propose a cross-draping strategy. Instead of manipulating the mesh directly, we instantiate a separate “source” body in a different size, generate a pattern fitted to that body, and then drape the resulting garment onto the original target body. This process simulates the physical reality of a person wearing a garment designed for someone else, resulting in significantly more natural and realistic ill-fitting dynamics compared to simple linear scaling.

Boxmesh Realignment. Cross-draping a sewing pattern onto a target body mesh of a different size creates misalignments between the boxmesh panels and target body parts, which can lead to draping errors. We implement boxmesh realignment (Section 3.2 in main) as a critical step for successful



Figure 48: **Additional dataset examples.** In each example, we show the paired person image (top left), garment image (lower left), and the target try-on image (right).

cross-body draping. See Figure 47 for a visual comparison with and without boxmesh alignment.

To align a given sewing pattern p_{tgt} (might be ill-fit to size s_p body) with a target body mesh of size s_p , we use a different, well-fitted (i.e. generated on size s_p body) sewing pattern p_{ref} as a reference. We then align the panels of the p_{tgt} to the spatial locations of p_{ref} . This ensures that p_{tgt} is aligned to the target body mesh. Additionally, we observed that significant discrepancies between the human model’s arm angle and the initialized sleeve panel angle can cause arm-sleeve penetrations. To mitigate this, we adjust the sleeve angle to match the arm angle prior to simulation.

Retexturing Model Training Data. We train our retexturing model on a dataset of 50k real-world person images. We use the person images in VITON-HD and additionally scraped online images featuring modeling posing in front of the camera with a studio background. We use Sapiens [92] to estimate normal and segmentation maps and Gemini [167] to generate prompts describing the garment textures and designs. We enforce a structured prompt format containing two sentences (one per garment piece). This facilitates inference for paired generation: since only the top garment is swapped, we update the first sentence corresponding to the top, while the second sentence remains frozen to preserve the bottom garment.

Reposing. Since GarmentCode simulation produces results exclusively in a static A-pose, we employ a customized reposing pipeline to repose the 3D simulated meshes, thereby expanding the dataset’s diversity and improve model generalization. In total, we sample from 528 distinct target poses and repose each sample into a randomly chosen pose from the pool, prioritizing casual stances commonly encountered in real-world try-on scenarios.

6.5.3 Additional Resizing Results

We further evaluate independent controllability of individual garment measurements in Figure 49. Fit-VTO realistically adjusts specific garment dimensions with respect to measurement changes, while preserving the non-adjusted garment dimensions, as well as person and garment appearance.



Figure 49: Qualitative resizing results. Fit-VTO adapts garment fit according to individual garment measurements. We show results of independently shrinking and growing the length, width, and sleeve length with respect to the original value. Please zoom in for details.

To show how Fit-VTO generalizes to real-world images, we provide additional resizing results on real-world person images in Figure 51. In these examples, the person and person measurements are captured from real human subjects, and the garment layflat image and measurements are randomly chosen from the FIT test dataset.

6.5.4 Failure Cases

We show qualitative examples of the limitations of our method in Figure 50. These include a limited ability to represent varying degrees of garment tightness in GarmentCode [99], which we leave to future work. Another limitation is that garment measurements are often correlated in our FIT data (e.g. larger width correlates positively with larger length). As a result, with our Fit-VTO model, changing one measurement may lead to an unintentional change in another dimension.

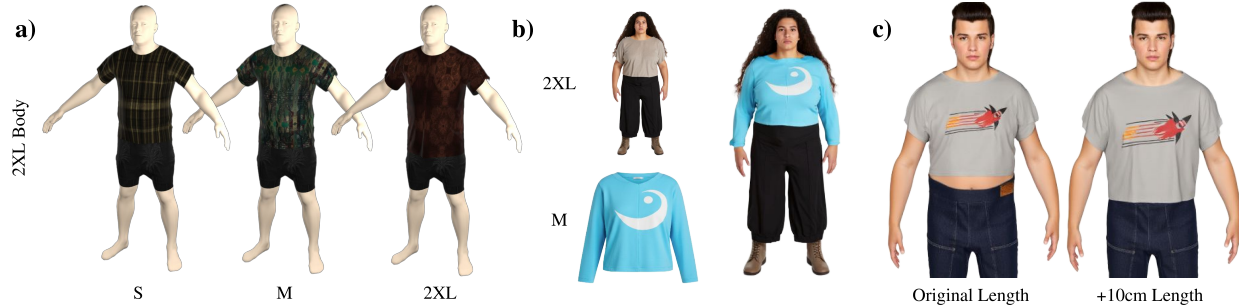


Figure 50: Failure cases. (a) GarmentCode [99] simulation does not model varying degrees of tightness well, leading to similar-looking fit for all garment sizes smaller than the body size. (b) As a result of (a), it is difficult to tell the level of tightness in FIT try-on images. (c) Due to the correlations in measurements across sizes, adjustments to individual measurements may also lead to undesired changes in other dimensions. In this example, increasing garment length also increases garment width.

6.5.5 Comparisons to State-of-the-Art

VITON-HD Preprocessing. Due to the lack of paired data, we generated pseudo paired-person images I_p for every image in the VITON-HD [27] using Nano Banana Pro (see Section 6.5.6 for prompts). When running our Fit-VTO method, we set the each person and garment measurement to the null value (-1), same as the dropout value used during training.

Implementation Details.

- **Any2AnyTryon [63]:** We used the model and code released from the official implementation. For all evaluations, we used the “dev_lora_any2any_multi” checkpoint.
- **COTTON [24]:** The officially released code and checkpoint trained on COTTON dataset was used. For evaluation on the VITON-HD test dataset, the default try-on mode was used. When running on FIT test dataset, the scaling parameter was computed as $r = \text{length}/\text{bust}$.
- **Nano Banana Pro [56]:** For comparisons to Nano Banana Pro [56], we input the paired-person image I_p , layflat garment image I_g , person-garment measurements m , and the prompt.

Edit $image_0$ so that the person wears garment in $image_1$ with size of person and garment described as {measurement description}.



Figure 51: Real-world resizing results. We show Fit-VTO try-on performance on real-world person images using varying garment sizes. Fit-VTO realistically shrinks and grows the garment fit according to uniform adjustments to the garment measurements—length, width, and sleeve length—with respect to their original values (1.0x). The size label to the left of each example corresponds to the person’s body size. Since real-world garment images with precise measurements are difficult to acquire, we use our synthetic garment images and measurements.

- **IDM-VTON [28]:** We used the model and code released from the official implementation. For VITON-HD, the agnostic masks from the original data release were used. For FIT dataset, agnostic masks were computed from IDM-VTON preprocessing code. All hyperparameters(e.g. number of diffusion steps) are set to be the recommended value from official release.

6.5.6 LLM and VLM Prompts

In the follow sections, we provide the exact prompts used for all calls to LLM (Gemini [167]) and VLM (Nano Banana Pro [56]) models in this paper.

Head & Shoes Generation.

Change the head to make it look photorealistic. Add realistic <hair style> hair, but the hair should always be behind the shoulder and never at the front. Add <shoe type> if feet are visible. Make sure that everything else stays identical, including the human pose, garment shape, size, design and position.

Prompt Generation.

Describe the garment in the image in two sentences. The first sentence should describe the top garment, and the second sentence should describe the bottom garment. Note that the input image is an illustration of the garment type, style and size - please ignore its existing texture. Please come up with some new description of the texture, logo and design. Add pocket, zipper, button, and other garment details if appropriate. Keep everything under 50 words.

Garment Try-Off.

Create an in-shop product image of the top garment only against a plain white background.

Paired Person Image Generation.

Generate a new image where the upper garment is changed, and keep everything else exactly the same, including the bottom garment, face, human pose, position etc.

Quality Assurance (QA).

We leverage our LLM to filter out draping errors in I_s that expose either person bust or groin area.

We use the following two prompts to detect such errors:

Does the garment cover the person's chest? If so, return 'pass'. If not, return 'fail'.

Does the image contain a bottom garment (skirt, pants, underwear, boxers, leggings, or shorts) that covers the person's groin area? If so, return 'pass'. If not, return 'fail'.

6.5.7 Usage of LLMs

In addition to using an LLM [167] as described in Sections 6.5.1, 6.5.5, and 6.5.6, we leveraged an LLM to improve the grammar and clarity of our writing.

Chapter 7

CONCLUSION AND FUTURE WORK

This thesis addresses the fundamental challenge at the intersection of generative AI and fashion: accurately depicting how digital garments look, move, and fit in the real world. Through my contributions spanning image animation, video virtual try-on, 360° novel view synthesis, and fit-aware virtual try-on, I have demonstrated that modern diffusion and flow matching models are powerful and flexible tools for tackling realistic digital garment visualization. In addition, by introducing a large-scale dataset of over one million virtual try-on samples and detailed measurements, I not only promote further research in fit-aware virtual try-on, but also demonstrate that synthetic data generation methods are a scalable and effective solution for data-scarce problems. Together, these advances bring us closer to bridging the gap between digital and real-world garments.

7.1 Foundation Models for Digital Fashion

Foundation models for generative computer vision have progressed tremendously over the past several years. State-of-the-art image generation models [56, 102, 133, 187] can produce highly photorealistic images from text and image inputs across diverse subjects and styles. Recent video generation models [58, 168, 169, 175] are able to synthesize long, temporally coherent videos with realistic motion and fine-grain details. In the 3D domain, Seed3D 2.0 [166] and SAM3D 2.0 [197] achieve state-of-the-art results on single-image object reconstruction, preserving high frequency geometric details and fine-grain texture.

Most of the works presented in this thesis were developed before general-purpose foundation models had reached the current level of quality and robustness – necessitating the task-specific architectures and training strategies described in each chapter. Today, these powerful models offer a compelling

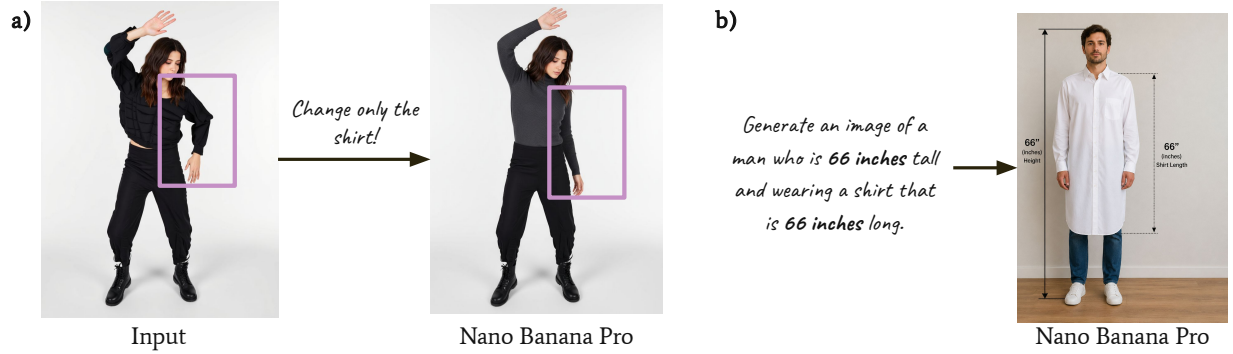


Figure 52: Limitations of modern foundation models. Despite remarkable visual quality, recent foundation models still struggle with precise conditioning and physical reasoning. **a)** In this example, Nano Banana Pro [56] inadvertently modifies both arm and hair regions. **b)** Nano Banana Pro [56] misrepresents a single length value as two different quantities in the same image.

starting point for future work in this space. In Chapter 6, I describe one such direction, showing how state-of-the-art VLMs [167] and image models [16, 56] can be leveraged across multiple stages of the fit-aware virtual try-on pipeline, including data generation (Section 6.2), virtual try-on (Section 6.3), and quality assurance (Section 6.5.6). As for my earlier works, an open-source video model, like Kling-OMNI [168] or Wan [175], could serve as a strong backbone for fashion image animation, video virtual try-on, or garment novel view synthesis, particularly with targeted task-specific finetuning. Likewise, a pretrained 3D reconstruction model [166, 197] could be a promising baseline for garment novel view synthesis or even 3D garment reconstruction.

However, despite remarkable visual quality, foundation models face key limitations when directly applied to digital fashion applications. First, while these models can often generalize to new tasks outside of their training domains, they often require additional finetuning to achieve a high level of the desired behavior, such as pose guidance or exact garment conditioning. In general, precise conditioning is challenging because foundation models’ non-deterministic generation process frequently introduces undesired changes to the input subjects – altering garment appearance, person identity, body shape, and body pose. As a result, these models struggle to faithfully preserve fine-grain person-and-garment details in the inputs which are essential for reliable fashion applications. See the example in Figure 52a. Compounding this challenge is the lack of fine-grained evaluation metrics to reliably detect and measure such failures, making it hard to systematically improve

these models for fashion-specific applications. A second major limitation is foundation models’ lack of understanding of the physics governing garment behavior. This leads to issues including physically implausible fabric motion in videos, an inability to understand or accurately represent garment measurements (Figure 52b), failure to respect fabric material properties such as stiffness and elasticity, and incorrect or inconsistent 3d garment geometry. Third, the computational demands of state-of-the-art foundation models pose a significant practical barrier for real-world fashion applications. These models are often far too expensive and time-consuming for everyday scenarios, such as running in real-time or on consumer devices. Addressing these limitations is the central challenge for the next generation of generative AI tools in fashion.

7.2 Open Problems and Future Work

The works in this thesis represent progress towards lifelike digital fashion using generative AI. However, generative AI today still falls short of the ultimate goal of providing consumers, retailers, and designers with the tools to visualize, design, and interact with garments in a digital setting that is equivalent to or even exceeds the real-world experience. Several open problems remain, including:

- **Fabric Awareness:** True garment fit is not only a function of the person and garment measurements, but also the material properties of the fabric – stiffness, weight, elasticity, etc. These properties also govern realistic garment wrinkling, drape, and flow in motion. As such, accurate fit-aware or video virtual try-on systems should consider both.
- **Fashion-Specific Evaluation:** A key challenge is the lack of fashion-specific evaluation metrics and benchmark datasets that reflect the specific properties of interest to these applications. As a result, comparing methods and measuring true progress in fashion AI remains difficult.
- **Real-Time Video Virtual Try-On:** High-quality, real-time video virtual try-on is still unsolved, even though such an experience would most naturally emulate a real-world try-on

experience for consumers.

To address these open problems, I propose the following directions for future research.

7.2.1 Fabric-Aware Virtual Try-On

A fundamental limitation of existing virtual try-on methods, including Fashion-VDM and Fit-VTO, is their inability to model physically accurate garment fabric. In reality, a garment’s drape, wrinkling, dynamics, and fit are governed not only by its geometry, but also by its material properties such as thickness, bending stiffness, stretching resistance, and friction. For example, a silk shirt will exhibit smooth, flowing drape; whereas, a leather garment of identical geometry will retain a rigid structure with sharp folds. Incorporating fabric material awareness into virtual try is therefore a natural and important direction for future work.

A key challenge is the lack of datasets with fabric property annotations. Existing VTO datasets, such as VITON-HD [27] and FIT [91], contain diverse garment imagery but do not provide information about underlying material properties, making supervised learning of fabric-aware models infeasible. Curating such a fabric-aware dataset for virtual try-on would be a valuable next step.

One potential direction is to estimate material properties directly from the images in existing datasets. Material Palette [120] generates PBR materials from real-world images. FabricDiffusion [205] transfers fabric textures from clothing images by extracting distortion-free, tileable materials that are subsequently mapped onto the UV space of a 3D garment as well as a PBR material. However, while these approaches capture surface appearance, they do not contain fabric-specific properties, such as garment weight, stretch, bending stiffness, friction, or thickness, which are essential for fabric-aware VTO. Moreover, inverse estimation is inherently ill-posed and error-prone, and existing datasets may lack sufficient diversity to robustly infer material behavior.

An alternative and promising direction is the collection of either real-world or synthetic fabric-aware datasets. Coarse fabric labels (e.g. “silk”, “cotton”, “denim”), which implicitly encode material

properties, could be scraped alongside garment images from online retailer metadata, though this data could be noisy and incomplete. Alternatively, more controlled data could be obtained through simulation platforms such as CLO [1], Style3D [2], and Marvelous Designer [30], which enable garment design and editing with controllable fabric and fit. These tools could be a valuable source for generating high-quality, paired VTO training data with fabric annotations.

Given a fabric-aware dataset, the next step would be to incorporate conditioning into a virtual try-on model. Several works explore physics-based garment simulation. For example, HOOD [60] uses graph neural networks to predict realistic garment dynamics conditioned on physical properties, while DiffusedWrinkles [173] and D-Garment [42] model garment deformations using diffusion models in UV space, conditioned on body shape, pose, and material signals. As discussed in Chapter 6, Fit-VTO leverages a custom encoder to condition a pretrained text-to-image model on garment measurements. Building on this approach, a promising research avenue is to integrate either numerical or textual material information via specialized encoder to a pretrained image or video foundation model.

Ultimately, fabric-aware virtual try-on requires bridging fabric properties and garment appearance. Developing representations that encode both the visual appearance and mechanical behavior of fabrics remains an open challenge, and is a key step toward realistic, controllable, and trustworthy virtual try-on systems.

7.2.2 Fashion-Specific Evaluation Protocol

A largely overlooked but critical challenge in fashion AI research is the absence of evaluation metrics and benchmark datasets that reflect the properties these systems are actually designed to produce. Widely adopted metrics such as FID [73], CLIP [138], and LPIPS [208] measure overall visual quality and coarse structural similarity, but correlate poorly with the fine-grained properties specific to fashion applications, including garment texture fidelity, person identity preservation, accurate body shape, and physically accurate fabric geometry. Overcoming this gap requires new

benchmark datasets with reliable ground truth and new metrics that are demonstrably aligned with human perceptual judgments.

Several recent works have begun to address this gap from the angle of perceptual benchmarking. VTONQA [184] introduced the first quality assessment (QA) dataset specifically designed for virtual try-on, comprising 8,132 images generated by 11 representative VTON models annotated with 24,396 mean opinion scores across clothing fit, body compatibility, and overall quality. Crucially, their analysis reveals that traditional image quality assessment metrics correlate poorly with human judgments on garment-specific factors like deformation and body-garment interactions. Similarly, VTBench [4] proposes a multi-dimensional evaluation framework spanning five dimensions: overall image quality, texture preservation, background, consistency, size adaptability, and hand-occlusion handling, benchmarking 15 recent VTON methods in a standardized setting with human preference annotations. OpenVTON-Bench [6] introduces a commercial-grade VTO dataset of 100,000 high-resolution try-on samples and leverages a VLM-as-a-Judge protocol to evaluate VTO images, as well as a mask-based metric for structural evaluation.

A parallel line of work has focused on developing reference-free metrics that can operate without ground-truth images. VTON-IQA [74] addresses this by constructing VTON-QBench, a large-scale human-annotated benchmark of 62,688 try-on images with over 431,800 quality annotations collected from 13,838 annotators – the largest human perceptual study in virtual try-on to date. The resulting model achieves reliable human-aligned image-level quality prediction without requiring reference images, which is essential for evaluating in-the-wild try-on systems. VTEdit-Bench [5] extends the evaluation scope beyond paired try-on to multi-reference image editing scenarios, introducing VTEdit-QA, a VLM-based metric that evaluates three dimensions: model consistency, cloth consistency, and image quality.

These recent works signal a broader shift in the field from general-purpose vision metrics toward more VTO-specific, human-grounded evaluation frameworks. However, despite this recent progress, several important gaps remain. First, existing protocols are almost exclusively focused on image-

based virtual try-on and do not address video try-on systems where additional dimensions such as temporal consistency, garment motion realism, and identity preservation across frames become critical. Second, these past works focus on pixel-level evaluation, without measuring physical correctness, like accurate fit, fabric drape, wrinkling, etc. The FIT dataset introduced in Chapter 6 is a step forward in this direction, leveraging 3D simulation to provide physically-accurate ground-truth for evaluating garment fit.

Future work could extend the FIT evaluation protocol to other categories of interest, such as body shape and fabric properties, leveraging simulation-grounded geometry comparisons that can quantify fit accuracy without relying solely on human annotation. Similarly, a valuable direction would be to develop automated metrics for physical geometry, such as landmark-based fit deviation scores. More broadly, the community would benefit from perceptual studies that establish which measurable properties best predict user satisfaction in real-world fashion applications, providing a principled foundation for the next generation of evaluation protocols.

7.2.3 Real-Time Video Virtual Try-On

Despite the advances demonstrated in Fashion-VDM and related works, high-quality video virtual try-on remains far from real-time deployment. Existing approaches typically require several minutes of inference on high-end GPUs, making them impractical for the interactive, consumer-facing applications that would most naturally emulate a real-world fitting room experience, such as live mirror try-on or mobile fitting apps.

A promising direction toward real-time VTO lies in reducing both the per-step computational cost and the number of required sampling steps. SwiftTry [131] takes a step in this direction by introducing “ShiftCaching”, which reduces redundant computation across diffusion timesteps by selectively updating a subset of frames, but does not yet reach interactive frame rates. Other general methods roughly fall into the following categories:

- *Latent video diffusion models* such as LTX-Video [66] operate in highly compressed latent spaces, significantly lowering the cost of each denoising step.
- *Distillation models* like consistency models [163] and progressive distillation [150] have demonstrated that diffusion models can be distilled into few-step or even single-step generators with minimal quality loss. Complementary approaches leverage step distillation to enable few-step or even single-step generation [110], trading additional training complexity for faster inference.
- *Flow matching* [111, 116], which replaces the stochastic diffusion process with a deterministic straight-line trajectory between noise and data, offers a more computationally efficient sampling paradigm and has been adopted in recent state-of-the-art video generation models [168, 169, 175].

Applying the above techniques to video virtual try-on, where the additional garment conditioning introduces complexity beyond general-purpose video generation, remains an open problem.

Real-time virtual try-on must also provide consistent streaming and temporal modeling. Most existing video try-on methods assume access to the full video clip during inference, which is incompatible with real-time operation. Real-time systems must instead operate causally by processing only past and present frames. Autoregressive video generation methods [23, 35, 72, 134, 200] address this setting by conditioning each generated frame on previously generated ones. Recent works such as Genie 2 [134] and VideoMAR [201], demonstrate that autoregressive generation can sustain high visual quality and temporal consistency over long horizons. To address temporal drift, self-forcing [84], is a technique that explicitly trains models to condition on their own previously generated frames rather than ground-truth frames. Integrating step-efficient sampling with causal autoregressive modeling and garment-and-person conditioning could enable video virtual try-on in real time.

Bibliography

- [1] Clo. <https://www.clo3d.com/en/>. Accessed: 2025-01-09.
- [2] Style3d. <https://www.style3d.ai/>. Accessed: 2026-04-22.
- [3] Turbosquid. <https://www.turbosquid.com/>. Accessed: 2025-01-09.
- [4] VTBench: Comprehensive benchmark suite towards real-world virtual try-on models. *arXiv preprint arXiv:2505.19571*, 2025.
- [5] VTEdit-Bench: A comprehensive benchmark for multi-reference image editing models in virtual try-on. *arXiv preprint arXiv:2603.11734*, 2025.
- [6] OpenVTON-Bench: A large-scale high-resolution benchmark for controllable virtual try-on evaluation. *arXiv preprint arXiv:2601.22725*, 2026.
- [7] Badour AlBahar, Jingwan Lu, Jimei Yang, Zhixin Shu, Eli Shechtman, and Jia-Bin Huang. Pose with style: Detail-preserving pose-guided image synthesis with conditional stylegan, 2021.
- [8] Michael Samuel Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *The Eleventh International Conference on Learning Representations*, 2023.
- [9] Shuai Bai, Huiling Zhou, Zhikang Li, Chang Zhou, and Hongxia Yang. Single stage virtual try-on via deformable attention flows. In Shai Avidan, Gabriel Brostow, Moustapha Cissé,

- Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision – ECCV 2022*, pages 409–425, Cham, 2022. Springer Nature Switzerland.
- [10] Seungbae Bang, Maria Korosteleva, and Sung-Hee Lee. Estimating garment patterns from static scan data. *Computer Graphics Forum*, 2021.
 - [11] Arpit Bansal, Hong-Min Chu, Avi Schwarzschild, Soumyadip Sengupta, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Universal guidance for diffusion models, 2023.
 - [12] Wenbo Bao, Wei-Sheng Lai, Chao Ma, Xiaoyun Zhang, Zhiyong Gao, and Ming-Hsuan Yang. Depth-aware video frame interpolation, 2019.
 - [13] Hugo Bertiche, Meysam Madadi, and Sergio Escalera. Cloth3d: Clothed 3d humans. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2020.
 - [14] Ankan Kumar Bhunia, Salman Khan, Hisham Cholakkal, Rao Muhammad Anwer, Jorma Laaksonen, Mubarak Shah, and Fahad Shahbaz Khan. Person image synthesis via denoising diffusion model, 2022.
 - [15] Mikolaj Binkowski, Danica J. Sutherland, Michal Arbel, and Arthur Gretton. Demystifying mmd gans. *ArXiv*, abs/1801.01401, 2018.
 - [16] Black Forest Labs. FLUX.1 [dev]: A 12 Billion Parameter Rectified Flow Transformer. Hugging Face Model Repository, Oct 2024. Model available at <https://huggingface.co/black-forest-labs/FLUX.1-dev>.
 - [17] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, Varun Jampani, and Robin Rombach. Stable video diffusion: Scaling latent video diffusion models to large datasets, November 2023.

- [18] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22563–22575, June 2023.
- [19] Botika. Botika: AI-generated fashion photography & virtual model platform. <https://botika.com>, 2023.
- [20] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions, 2022.
- [21] Shidong Cao, Wenhao Chai, Shengyu Hao, Yanting Zhang, Hangyue Chen, and Gaoang Wang. Diffashion: Reference-based fashion design with structure-aware transfer by diffusion models, 2023.
- [22] Capital One Shopping Research. Online clothing shopping statistics. <https://capitaloneshopping.com/research/online-clothing-shopping-statistics/>, 2025. Accessed: 2026-04-23.
- [23] Haoxuan Che, Xuanhua He, Quande Liu, Cheng Jin, and Hao Chen. Gamegen-x: Interactive open-world game video generation. In *The Thirteenth International Conference on Learning Representations*, 2025.
- [24] Chieh-Yun Chen, Yi-Chung Chen, Hong-Han Shuai, and Wen-Huang Cheng. Size does matter: Size-aware virtual try-on via clothing-oriented transformation try-on network. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7513–7522, 2023.
- [25] Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. Extending context window of large language models via positional interpolation. *arXiv preprint arXiv:2306.15595*, 2023.

- [26] Xinyuan Chen, Yaohui Wang, Lingjun Zhang, Shaobin Zhuang, Xin Ma, Jiashuo Yu, Yali Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. Seine: Short-to-long video diffusion model for generative transition and prediction, 2023.
- [27] Seunghwan Choi, Sunghyun Park, Minsoo Lee, and Jaegul Choo. Viton-hd: High-resolution virtual try-on via misalignment-aware normalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [28] Yisol Choi, Sangkyung Kwak, Kyungmin Lee, Hyungwon Choi, and Jinwoo Shin. Improving diffusion models for authentic virtual try-on in the wild. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.
- [29] Zheng Chong, Xiao Dong, Haoxiang Li, Shiyue Zhang, Wenqing Zhang, Xujie Zhang, Hanqing Zhao, and Xiaodan Liang. Catvton: Concatenation is all you need for virtual try-on with diffusion models, 2024.
- [30] CLO Virtual Fashion. Marvelous designer, 2024. Accessed: 2024-05-20.
- [31] Alimama Creative. Flux.1-dev-controlnet-inpainting-beta. <https://huggingface.co/alimama-creative/FLUX.1-dev-Controlnet-Inpainting-Beta>, 2024.
- [32] Aiyu Cui, Jay Mahajan, Viraj Shah, Preeti Gomathinayagam, Chang Liu, and Svetlana Lazebnik. Street tryon: Learning in-the-wild virtual try-on from unpaired person images. *arXiv preprint arXiv:2311.16094*, 2023.
- [33] Aiyu Cui, Daniel McKee, and Svetlana Lazebnik. Dressing in order: Recurrent person image generation for pose transfer, virtual try-on and outfit editing, 2021.
- [34] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. *arXiv preprint arXiv:2212.08051*, 2022.

- [35] Haoge Deng, Ting Pan, Haiwen Diao, Zhengxiong Luo, Yufeng Cui, Huchuan Lu, Shiguang Shan, Yonggang Qi, and Xinlong Wang. Autoregressive video generation without vector quantization. *arXiv preprint arXiv:2412.14169*, 2024.
- [36] Kangle Deng, Timothy Omernick, Alexander Weiss, Deva Ramanan, Jun-Yan Zhu, Tinghui Zhou, and Maneesh Agrawala. Flashtex: Fast relightable mesh texturing with lightcontrolnet. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol, editors, *Computer Vision – ECCV 2024*, pages 90–107, Cham, 2025. Springer Nature Switzerland.
- [37] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 2021.
- [38] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.
- [39] Xin Dong, Fuwei Zhao, Zhenyu Xie, Xijin Zhang, Daniel K. Du, Min Zheng, Xiang Long, Xiaodan Liang, and Jianchao Yang. Dressing in the wild by watching dance videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3480–3489, June 2022.
- [40] Chenghu Du, Shuqing Liu, Shengwu Xiong, et al. Greatness in simplicity: Unified self-cycle consistency for parser-free virtual try-on. *Advances in Neural Information Processing Systems*, 36:20287–20298, 2023.
- [41] Chenghu Du, Shengwu Xiong, Junyin Wang, Yi Rong, and Shili Xiong. Mitigating occlusions in virtual try-on via a simple-yet-effective mask-free framework. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.
- [42] Antoine Dumoulin, Adnane Boukhayma, Laurence Boissieux, Bharath Bhushan Damodaran,

- Pierre Hellier, et al. D-Garment: Physics-conditioned latent diffusion for dynamic garment deformations. *arXiv preprint arXiv:2504.03468*, 2025.
- [43] Patrick Esser, Qinsheng Sun, the Stable Diffusion 3 Team, et al. Scaling rectified flow transformers for high-resolution image synthesis. *arXiv preprint arXiv:2403.03206*, 2024.
- [44] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *International Conference on Learning Representations (ICLR)*, 2023.
- [45] Rinon Gal, Moab Arar, Yuval Atzmon, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. Designing an encoder for fast personalization of text-to-image models, 2023.
- [46] Daiheng Gao, Xu Chen, Xindi Zhang, Qi Wang, Ke Sun, Bang Zhang, Liefeng Bo, and Qixing Huang. Cloth2tex: A customized cloth texture generation pipeline for 3d virtual try-on, 2023.
- [47] Ruiqi Gao, Aleksander Holynski, Philipp Henzler, Arthur Brussee, Ricardo Martin-Brualla, Pratul Srinivasan, Jonathan T. Barron, and Ben Poole. Cat3d: Create anything in 3d with multi-view diffusion models, 2024.
- [48] Chongjian Ge, Yibing Song, Yuying Ge, Han Yang, Wei Liu, and Ping Luo. Disentangled cycle consistency for highly-realistic virtual try-on. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16928–16937, 2021.
- [49] Yuying Ge, Yibing Song, Ruimao Zhang, Chongjian Ge, Wei Liu, and Ping Luo. Parser-free virtual try-on via distilling appearance flows. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8485–8493, 2021.
- [50] Yuying Ge, Ruimao Zhang, Xiaogang Wang, Xiaoou Tang, and Ping Luo. Deepfashion2: A versatile benchmark for detection, pose estimation, segmentation and retrieval of clothing

- images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
- [51] Yuying Ge, Ruimao Zhang, Lingyun Wu, Xiaogang Wang, Xiaoou Tang, and Ping Luo. Deepfashion2: A versatile benchmark for detection, pose estimation, segmentation and re-identification of clothing images, 2019.
- [52] Rohit Girdhar, Mannat Singh, Andrew Brown, Quentin Duval, Samaneh Azadi, Sai Saketh Rambhatla, Akbar Shah, Xi Yin, Devi Parikh, and Ishan Misra. Emu video: Factorizing text-to-video generation by explicit image conditioning, 2023.
- [53] Ke Gong, Yiming Gao, Xiaodan Liang, Xiaohui Shen, Meng Wang, and Liang Lin. Graphonomy: Universal human parsing via graph transfer learning, 2019.
- [54] Google. Gemini 2.5 flash image. <https://gemini.google.com/>, 2025.
- [55] Google. Gemini 2.5 flash image. <https://gemini.google.com/>, 2025.
- [56] Google. Gemini 3 pro image. <https://gemini.google.com/>, 2025.
- [57] Google DeepMind. Google Veo: Generating high-quality video with high-fidelity, 2024. Accessed: 2026-04-09.
- [58] Google DeepMind. Veo 3 technical report, 2024.
- [59] Google Labs. Doppl: Try on looks and discover your style. <https://labs.google/doppl/>, 2025. Experimental app for virtual try-on.
- [60] Artur Grigorev, Michael J. Black, and Otmar Hilliges. HOOD: Hierarchical graphs for generalized modelling of clothing dynamics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16965–16974, 2023.
- [61] Jiatao Gu, Shuangfei Zhai, Yizhe Zhang, Josh Susskind, and Navdeep Jaitly. Matryoshka diffusion models, 2023.

- [62] Jiayi Gu, Shicong Wang, Haoyu Zhao, Tianyi Lu, Xing Zhang, Zuxuan Wu, Songcen Xu, Wei Zhang, Yu-Gang Jiang, and Hang Xu. Reuse and diffuse: Iterative denoising for text-to-video generation, 2023.
- [63] Hailong Guo, Bohan Zeng, Yiren Song, Wentao Zhang, Chuan Zhang, and Jiaming Liu. Any2anytryon: Leveraging adaptive position embeddings for versatile virtual clothing tasks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2025.
- [64] Yuwei Guo, Ceyuan Yang, Anyi Rao, Yaohui Wang, Yu Qiao, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning, 2023.
- [65] Rıza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild, 2018.
- [66] Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, Poriya Panet, Sapir Weissbuch, Victor Kulikov, Yaki Bitterman, Zeev Melumian, and Ofir Bibi. Ltx-video: Realtime video latent diffusion, 12 2024.
- [67] Xintong Han, Xiaojun Hu, Weilin Huang, and Matthew R. Scott. Clothflow: A flow-based model for clothed person generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.
- [68] Xintong Han, Zuxuan Wu, Zhe Wu, Ruichi Yu, and Larry S. Davis. Viton: An image-based virtual try-on network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [69] William Harvey, Saeid Naderiparizi, Vaden Masrani, Christian Weilbach, and Frank Wood. Flexible diffusion modeling of long videos, 2022.

- [70] Sen He, Yi-Zhe Song, and Tao Xiang. Style-based global appearance flow for virtual try-on. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3470–3479, June 2022.
- [71] Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent video diffusion models for high-fidelity video generation with arbitrary lengths, 2022.
- [72] Roberto Henschel, Levon Khachatryan, Daniil Hayrapetyan, Hayk Poghosyan, Vahram Tadevosyan, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Streamingt2v: Consistent, dynamic, and extendable long video generation from text. *arXiv preprint arXiv:2403.14773*, 2024.
- [73] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. 2017.
- [74] Yuki Hirakawa, Takashi Wada, Ryotaro Shimizu, Takuya Furusawa, Yuki Saito, Ryosuke Araki, Tianwei Chen, Fan Mo, and Yoshimitsu Aoki. Reference-free image quality assessment for virtual try-on via human feedback. *arXiv preprint arXiv:2603.13057*, 2025.
- [75] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P. Kingma, Ben Poole, Mohammad Norouzi, David J. Fleet, and Tim Salimans. Imagen video: High definition video generation with diffusion models, 2022.
- [76] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020.
- [77] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance, 2022.
- [78] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J. Fleet. Video diffusion models, 2022.
- [79] Aleksander Holynski, Brian Curless, Steven M. Seitz, and Richard Szeliski. Animating pictures with eulerian motion fields, 2020.

- [80] Fangzhou Hong, Zhaoxi Chen, Yushi Lan, Liang Pan, and Ziwei Liu. Eva3d: Compositional 3d human generation from 2d image collections, 2022.
- [81] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yanzhi Wang, Yu Chen, Lu Li, Xiangru Wang, Lu Wang, Yan Zhou, et al. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2308.03303*, 2023.
- [82] Li Hu, Xin Gao, Peng Zhang, Ke Sun, Bang Zhang, and Liefeng Bo. Animate anyone: Consistent and controllable image-to-video synthesis for character animation, 2023.
- [83] Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q Weinberger. Deep networks with stochastic depth. In *Computer Vision—ECCV 2016: 14th European Conference*, pages 646–661. Springer, 2016.
- [84] Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. *arXiv preprint arXiv:2506.08009*, 2025.
- [85] Thibaut Issenhuth, Jérémie Mary, and Clément Calauzenes. Do not mask what you do not need to mask: a parser-free virtual try-on. In *European Conference on Computer Vision*, pages 619–635. Springer, 2020.
- [86] Jianbin Jiang, Tan Wang, He Yan, and Junhui Liu. Clothformer: Taming video virtual try-on in all module. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10799–10808, June 2022.
- [87] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution, 2016.
- [88] Johanna Karras, Aleksander Holynski, Ting-Chun Wang, and Ira Kemelmacher-Shlizerman. Dreampose: Fashion video synthesis with stable diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 22680–22690, October 2023.

- [89] Johanna Karras, Yingwei Li, Yasamin Jafarian, and Ira Kemelmacher-Shlizerman. Hologarment: 360° novel view synthesis of in-the-wild garments, 2025.
- [90] Johanna Karras, Yingwei Li, Nan Liu, Luyang Zhu, Innfarn Yoo, Andreas Lugmayr, Chris Lee, and Ira Kemelmacher-Shlizerman. Fashion-vdm: Video diffusion model for virtual try-on, 2024.
- [91] Johanna Karras, Yuanhao Wang, Yingwei Li, and Ira Kemelmacher-Shlizerman. Fit: A large-scale dataset for fit-aware virtual try-on. *SIGGRAPH*, 2026.
- [92] Rawal Khirodkar, Timur Bagautdinov, Julieta Martinez, Su Zhaoen, Austin James, Peter Selednik, Stuart Anderson, and Shunsuke Saito. Sapiens: Foundation for human vision models. In *European Conference on Computer Vision*, pages 206–228. Springer, 2024.
- [93] Jeongho Kim, Guojung Gu, Minhoo Park, Sunghyun Park, and Jaegul Choo. Stableviton: Learning semantic correspondence with latent diffusion model for virtual try-on. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8176–8185, 2024.
- [94] Jeongho Kim, Gyojung Gu, Minhoo Park, Sunghyun Park, and Jaegul Choo. Stableviton: Learning semantic correspondence with latent diffusion model for virtual try-on, 2023.
- [95] Jeongho Kim, Hoiyeong Jin, Sunghyun Park, and Jaegul Choo. Promptdresser: Improving the quality and controllability of virtual try-on via generative textual prompt and prompt-aware mask. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16026–16036, 2025.
- [96] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.
- [97] Maria Korosteleva, Timur Levent Kesdogan, Fabian Kemper, Stephan Wenninger, Jasmin Koller, Yuhan Zhang, Mario Botsch, and Olga Sorkine-Hornung. GarmentCodeData: A

- dataset of 3d made-to-measure garments with sewing patterns. In *European Conference on Computer Vision (ECCV)*, 2024.
- [98] Maria Korosteleva and Sung-Hee Lee. Neuraltailor: Reconstructing sewing pattern structures from 3d point clouds of garments. 2022.
- [99] Maria Korosteleva and Olga Sorkine-Hornung. GarmentCode: Programming parametric sewing patterns. *ACM Transactions on Graphics (TOG)*, 42(6), 2023.
- [100] Minoru Kuribayashi, Koki Nakai, and Nobuo Funabiki. Image-based virtual try-on system with clothing-size adjustment. *arXiv preprint arXiv:2302.14197*, 2023.
- [101] Jeonggi Kwak, Erqun Dong, Yuhe Jin, Hanseok Ko, Shweta Mahajan, and Kwang Moo Yi. Vivid-1-to-3: Novel view synthesis with video diffusion models, 2023.
- [102] Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, et al. Flux.1 kontext: Flow matching for in-context image generation and editing in latent space, 2025.
- [103] Sangyun Lee, Gyojung Gu, Sunghyun Park, Seunghwan Choi, and Jaegul Choo. High-resolution virtual try-on with misalignment and occlusion-handled conditions. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision – ECCV 2022*, pages 204–219, Cham, 2022. Springer Nature Switzerland.
- [104] Seung Hyun Lee, Sieun Kim, Innfarn Yoo, Feng Yang, Donghyeon Cho, Youngseo Kim, Huiwen Chang, Jinkyu Kim, and Sangpil Kim. Soundini: Sound-guided diffusion for natural video editing, 2023.
- [105] Kathleen M Lewis, Srivatsan Varadharajan, and Ira Kemelmacher-Shlizerman. Tryongan: Body-aware try-on via layered interpolation, 2021.

- [106] Siran Li, Ruiyang Liu, Chen Liu, Zhendong Wang, Gaofeng He, Yong-Lu Li, Xiaogang Jin, and Huamin Wang. Garmagenet: A multimodal generative framework for sewing pattern design and generic garment modeling. *ACM Trans. Graph.*, 44(6), December 2025.
- [107] Yifei Li, Hsiao yu Chen, Egor Larionov, Nikolaos Sarafianos, Wojciech Matusik, and Tuur Stuyck. Diffavatar: Simulation-ready garment optimization with differentiable simulation, 2023.
- [108] Seungchan Lim, Sumin Kim, and Sung-Hee Lee. Spnet: Estimating garment sewing patterns from a single image, 2023.
- [109] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [110] Shanchuan Lin and Xiao Yang. Animatediff-lightning: Cross-model diffusion distillation, 2024.
- [111] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023.
- [112] Lijuan Liu, Xiangyu Xu, Zhijie Lin, Jiabin Liang, and Shuicheng Yan. Towards garment sewing pattern reconstruction from a single image. *ACM Trans. Graph.*, 42(6), 2023.
- [113] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds, 2022.
- [114] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object, 2023.

- [115] Songhua Liu, Jingwen Ye, Sucheng Ren, and Xinchao Wang. Dynast: Dynamic sparse transformer for exemplar-guided image generation, 2022.
- [116] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations*, 2023.
- [117] Xingchao Liu, Xiwen Zhang, Jianzhu Ma, Jian Peng, and Qiang Liu. InstafLOW: One step is enough for high-quality diffusion-based text-to-image generation, 2023.
- [118] Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. Syncdreamer: Generating multiview-consistent images from a single-view image, 2023.
- [119] Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [120] Ivan Lopes, Fabio Pizzati, and Raoul de Charette. Material palette: Extraction of materials from a single image. In *CVPR*, 2024.
- [121] Xiao Xin Lu. A review of solutions for perspective-n-point problem in camera pose estimation, 2018.
- [122] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. Mediapipe: A framework for building perception pipelines, 2019.
- [123] Zhongjin Luo, Haolin Liu, Chenghong Li, Wanghao Du, Zirong Jin, Wanhu Sun, Yinyu Nie, Weikai Chen, and Xiaoguang Han. Garverselod: High-fidelity 3d garment reconstruction

- from a single in-the-wild image using a dataset with levels of details. *ACM Transactions on Graphics*, 2024.
- [124] Miles Macklin. Warp: A high-performance python framework for gpu simulation and graphics. In *NVIDIA GPU Technology Conference (GTC)*, volume 3, 2022.
- [125] Arun Mallya, Ting-Chun Wang, and Ming-Yu Liu. Implicit warping for animation with image sets, 2022.
- [126] Kangfu Mei and Vishal Patel. Vidm: Video implicit diffusion models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(8):9117–9125, Jun. 2023.
- [127] Yifang Men, Yiming Mao, Yuning Jiang, Wei-Ying Ma, and Zhouhui Lian. Controllable person image synthesis with attribute-decomposed gan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.
- [128] Eyal Molad, Eliahu Horwitz, Dani Valevski, Alex Rav Acha, Yossi Matias, Yael Pritch, Yaniv Leviathan, and Yedid Hoshen. Dreamix: Video diffusion models are general video editors, 2023.
- [129] Davide Morelli, Matteo Fincato, Marcella Cornia, Federico Landi, Fabio Cesari, and Rita Cucchiara. Dress Code: High-Resolution Multi-Category Virtual Try-On. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.
- [130] Chong Mou, Xintao Wang, Liangbin Xie, Jian Zhang, Zhongang Qi, Ying Shan, and Xiaohu Qie. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 4296–4304, 2024.
- [131] Hung Nguyen, Quang Nguyen, Khoi Nguyen, and Rang Nguyen. Swifttry: Fast and consistent video virtual try-on with diffusion models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39:6200–6208, 04 2025.

- [132] OpenAI. Video generation models as world simulators, 2024. Accessed: [Insert Date Here].
- [133] OpenAI. ChatGPT Images 2.0. AI Image Generator, 2026. Accessed: YYYY-MM-DD.
- [134] Jack Parker-Holder, Philip Ball, Jake Bruce, Vibhavari Dasagi, Kristian Holsheimer, Christos Kaplanis, Alexandre Moufarek, Guy Scully, Jeremy Shar, Jimmy Shi, Stephen Spencer, Jessica Yung, Michael Dennis, Sultan Kenjeyev, Shangbang Long, Vlad Mnih, Harris Chan, Maxime Gazeau, Bonnie Li, Fabio Pardo, Luyu Wang, Lei Zhang, Frederic Besse, Tim Harley, Anna Mitenkova, Jane Wang, Jeff Clune, Demis Hassabis, Raia Hadsell, Adrian Bolton, Satinder Singh, and Tim Rocktäschel. Genie 2: A large-scale foundation world model. 2024.
- [135] Chaitanya Patel, Zhouyingcheng Liao, and Gerard Pons-Moll. Tailornet: Predicting clothing in 3d as a function of human pose, shape and garment style. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [136] William Peebles and Saining Xie. Scalable diffusion models with transformers, 2022.
- [137] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion, 2022.
- [138] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.
- [139] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.
- [140] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents, 2022.

- [141] Royi Rassin, Shauli Ravfogel, and Yoav Goldberg. Dalle-2 is seeing double: Flaws in word-to-concept mapping in text2image models, 2022.
- [142] Fitsum Reda, Janne Kontkanen, Eric Tabellion, Deqing Sun, Caroline Pantofaru, and Brian Curless. Film: Frame interpolation for large motion, 2022.
- [143] Refabric. Refabric: AI fashion design assistant. <https://refabric.com>, 2023.
- [144] Yurui Ren, Xiaoqing Fan, Ge Li, Shan Liu, and Thomas H. Li. Neural texture extraction and distribution for controllable person image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13535–13544, June 2022.
- [145] Yurui Ren, Xiaoming Yu, Junming Chen, Thomas H. Li, and Ge Li. Deep image spatial transformation for person image generation, 2020.
- [146] Elad Richardson, Gal Metzer, Yuval Alaluf, Raja Giryes, and Daniel Cohen-Or. Texture: Text-guided texturing of 3d shapes, 2023.
- [147] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.
- [148] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation, 2022.
- [149] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding, 2022.

- [150] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models, 2022.
- [151] Nikolaos Sarafianos, Tuur Stuyck, Xiaoyu Xiang, Yilei Li, Jovan Popovic, and Rakesh Ranjan. Garment3dgen: 3d garment stylization and texture generation, 2024.
- [152] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. Laion-5b: An open large-scale dataset for training next generation image-text models, 2022.
- [153] Ruoxi Shi, Hansheng Chen, Zhuoyang Zhang, Minghua Liu, Chao Xu, Xinyue Wei, Linghao Chen, Chong Zeng, and Hao Su. Zero123++: a single image to consistent multi-view diffusion base model, 2023.
- [154] Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation, 2023.
- [155] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. Animating arbitrary objects via deep motion transfer, 2018.
- [156] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. First order motion model for image animation, 2020.
- [157] Aliaksandr Siarohin, Oliver J. Woodford, Jian Ren, Menglei Chai, and Sergey Tulyakov. Motion representations for articulated animation. 2021.
- [158] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, Devi Parikh, Sonal Gupta, and Yaniv Taigman. Make-a-video: Text-to-video generation without text-video data, 2022.

- [159] Uriel Singer, Shelly Sheynin, Adam Polyak, Oron Ashual, Iurii Makarov, Filippos Kokkinos, Naman Goyal, Andrea Vedaldi, Devi Parikh, Justin Johnson, and Yaniv Taigman. Text-to-4d dynamic scene generation, 2023.
- [160] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Brendan Swersky. Deep unsupervised learning using nonequilibrium thermodynamics. *International Conference on Machine Learning*, 2015.
- [161] Bingjie Song, Xin Huang, Ruting Xie, Xue Wang, and Qing Wang. Style3d: Attention-guided multi-view style transfer for 3d object generation, 2024.
- [162] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *International Conference on Learning Representations*, 2020.
- [163] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. *arXiv preprint arXiv:2303.01469*, 2023.
- [164] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 2019.
- [165] Zhaoqi Su, Tao Yu, Yangang Wang, and Yebin Liu. Deepcloth: Neural garment representation for shape and style editing. *IEEE TPAMI*, 2020.
- [166] ByteDance Seed Team. Seed3d 2.0: Advancing high-fidelity simulation-ready 3d content generation. *ByteDance Seed*, 2026.
- [167] Gemini Team. Gemini: A family of highly capable multimodal models, 2023.
- [168] Kling Team. Kling-omni technical report. *arXiv preprint arXiv:2512.16776*, 2025.
- [169] HPC-AI Tech. Open-sora 2.0: Training a commercial-level video generation model in \$200k, 2025.

- [170] Garvita Tiwari, Bharat Lal Bhatnagar, Tony Tung, and Gerard Pons-Moll. Sizer: A dataset and model for parsing 3d clothing and learning size sensitive 3d clothing. In *European Conference on Computer Vision (ECCV)*. Springer, August 2020.
- [171] Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric and challenges. *arXiv preprint arXiv:1812.01717*, 2018.
- [172] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, pages 5998–6008, 2017.
- [173] Raquel Vidaurre, Elena Garces, and Dan Casas. DiffusedWrinkles: A diffusion-based model for data-driven garment animation. In *35th British Machine Vision Conference (BMVC)*, 2024.
- [174] Ruben Villegas, Mohammad Babaeizadeh, Pieter-Jan Kindermans, Hernan Moraldo, Han Zhang, Mohammad Taghi Saffar, Santiago Castro, Julius Kunze, and Dumitru Erhan. Phenaki: Variable length video generation from open domain textual description, 2022.
- [175] Teng Wan et al. Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.
- [176] Fu-Yun Wang, Wenshuo Chen, Guanglu Song, Han-Jia Ye, Yu Liu, and Hongsheng Li. Gen-l-video: Multi-text to long video generation via temporal co-denoising. *arXiv preprint arXiv:2305.18264*, 2023.
- [177] Peng Wang and Yichun Shi. Imagedream: Image-prompt multi-view diffusion for 3d generation, 2023.
- [178] Ting-Chun Wang, Ming-Yu Liu, Andrew Tao, Guilin Liu, Jan Kautz, and Bryan Catanzaro. Few-shot video-to-video synthesis. *ArXiv*, abs/1910.12713, 2019.

- [179] Xiang Wang, Hangjie Yuan, Shiwei Zhang, Dayou Chen, Jiuniu Wang, Yingya Zhang, Yujun Shen, Deli Zhao, and Jingren Zhou. Videocomposer: compositional video synthesis with motion controllability. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.
- [180] Yaohui Wang, Di Yang, Francois Bremond, and Antitza Dantcheva. Latent image animator: Learning to animate images via latent space navigation, 2022.
- [181] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.
- [182] Zhouxia Wang, Ziyang Yuan, Xintao Wang, Tianshui Chen, Menghan Xia, Ping Luo, and Ying Shan. Motionctrl: A unified and flexible motion controller for video generation, 2023.
- [183] Daniel Watson, William Chan, Ricardo Martin-Brualla, Jonathan Ho, Andrea Tagliasacchi, and Mohammad Norouzi. Novel view synthesis with diffusion models, 2022.
- [184] Xinyi Wei, Sijing Wu, Zitong Xu, Yunhao Li, Huiyu Duan, Xionghuo Min, and Guangtao Zhai. VTONQA: A multi-dimensional quality assessment dataset for virtual try-on. *arXiv preprint arXiv:2601.02945*, 2026.
- [185] Yi-Cheng Tien Wen-Jiin Tsai. Attention-based video virtual try-on. pages 209–216. ACM, Proceedings of the 2023 ACM International Conference on Multimedia Retrieval, June 2023.
- [186] Chung-Yi Weng, Brian Curless, and Ira Kemelmacher-Shlizerman. Photo wake-up: 3d character animation from a single photo, 2018.
- [187] Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, Yuxiang Chen, Zecheng Tang, Zekai Zhang, Zhengyi Wang, An Yang, Bowen Yu, Chen Cheng, Dayiheng Liu, Deqing Li, Hang Zhang, Hao Meng,

- Hu Wei, Jingyuan Ni, Kai Chen, Kuan Cao, Liang Peng, Lin Qu, Minggang Wu, Peng Wang, Shuting Yu, Tingkun Wen, Wensen Feng, Xiaoxiao Xu, Yi Wang, Yichang Zhang, Yongqiang Zhu, Yujia Wu, Yuxuan Cai, and Zenan Liu. Qwen-image technical report, 2025.
- [188] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Weixian Lei, Yuchao Gu, Wynne Hsu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation, 2022.
- [189] Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation, 2024.
- [190] Yuhao Xu, Tao Gu, Weifeng Chen, and Arlene Chen. Ootdiffusion: Outfitting fusion based latent diffusion for controllable virtual try-on. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 8996–9004, 2025.
- [191] Zhongcong Xu, Jianfeng Zhang, Jun Hao Liew, Hanshu Yan, Jia-Wei Liu, Chenxu Zhang, Jiashi Feng, and Mike Zheng Shou. Magicanimate: Temporally consistent human image animation using diffusion models, 2023.
- [192] xujie zhang, Xiu Li, Michael Kampffmeyer, Xin Dong, Zhenyu Xie, Feida Zhu, Haoye Dong, and Xiaodan Liang. Warpdiffusion: Efficient diffusion model for high-fidelity virtual try-on, 2023.
- [193] Yohei Yamashita, Chihiro Nakatani, and Norimichi Ukita. Size-variable virtual try-on with physical clothes size. *arXiv preprint arXiv:2412.06201*, 2024.
- [194] Han Yang, Ruimao Zhang, Xiaobao Guo, Wei Liu, Wangmeng Zuo, and Ping Luo. 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)*, June 2020.

- [195] Lu Yang, Yicheng Liu, Yanan Li, Xiang Bai, and Hao Lu. Fitcontroler: Toward fit-aware virtual try-on, 2025.
- [196] Ruihan Yang, Prakhar Srivastava, and Stephan Mandt. Diffusion probabilistic modeling for video generation, 2022.
- [197] Xitong Yang et al. Sam 3d: 3dfy anything in images. *arXiv preprint arXiv:2511.16624*, 2025.
- [198] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024.
- [199] Hu Ye, Jun Zhang, Sibio Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023.
- [200] Tianwei Yin, Qiang Zhang, Richard Zhang, William T Freeman, Fredo Durand, Eli Shechtman, and Xun Huang. From slow bidirectional to fast autoregressive video diffusion models. 2025.
- [201] Hu Yu, Biao Gong, Hangjie Yuan, Dandan Zheng, Weilong Chai, Jingdong Chen, Kecheng Zheng, and Feng Zhao. Videomar: Autoregressive video generation with continuous tokens. *NeurIPS 2026*, 2025.
- [202] Ruiyun Yu, Xiaoqi Wang, and Xiaohui Xie. Vtnfp: An image-based virtual try-on network with body and clothing feature preservation. page 10511–10520. Proceedings of the IEEE/CVF international conference on computer vision, 2019.
- [203] Polina Zablotskaia, Aliaksandr Siarohin, Bo Zhao, and Leonid Sigal. Dwnet: Dense warp-based network for pose-guided human video generation, 2019.
- [204] Egor Zakharov, Aliaksandra Shysheya, Egor Burkov, and Victor Lempitsky. Few-Shot Adversarial Learning of Realistic Neural Talking Head Models . In *2019 IEEE/CVF Interna-*

- tional Conference on Computer Vision (ICCV)*, pages 9458–9467, Los Alamitos, CA, USA, November 2019. IEEE Computer Society.
- [205] Cheng Zhang, Yuanhao Wang, Francisco Vicente, Chenglei Wu, Jinlong Yang, Thabo Beeler, and Fernando De la Torre. Fabricdiffusion: High-fidelity texture transfer for 3d garments generation from in-the-wild images. In *SIGGRAPH Asia 2024 Conference Papers*, SA '24, New York, NY, USA, 2024. Association for Computing Machinery.
 - [206] Jinsong Zhang, Kun Li, Yu-Kun Lai, and Jingyu Yang. Pise: Person image synthesis and editing with decoupled gan, 2021.
 - [207] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3836–3847, 2023.
 - [208] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric, 2018.
 - [209] 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 *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 26399–26408, 2025.
 - [210] Jian Zhao and Hui Zhang. Thin-plate spline motion model for image animation, 2022.
 - [211] Shihao Zhao, Dongdong Chen, Yen-Chun Chen, Jianmin Bao, Shaozhe Hao, Lu Yuan, and Kwan-Yee K. Wong. Uni-controlnet: All-in-one control to text-to-image diffusion models. *Advances in Neural Information Processing Systems*, 2023.
 - [212] Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all, 2024.

- [213] Xiaojing Zhong, Zhonghua Wu, Taizhe Tan, Guosheng Lin, and Qingyao Wu. Mv-ton: Memory-based video virtual try-on network. 2021.
- [214] Jensen (Jinghao) Zhou, Hang Gao, Vikram Voleti, Aaryaman Vasishta, Chun-Han Yao, Mark Boss, Philip Torr, Christian Rupprecht, and Varun Jampani. Stable virtual camera: Generative view synthesis with diffusion models. *arXiv preprint*, 2025.
- [215] Heming Zhu, Yu Cao, Hang Jin, Weikai Chen, Dong Du, Zhangye Wang, Shuguang Cui, and Xiaoguang Han. Deep fashion3d: A dataset and benchmark for 3d garment reconstruction from single images, 2020.
- [216] Luyang Zhu, Yingwei Li, Nan Liu, Hao Peng, Dawei Yang, and Ira Kemelmacher-Shlizerman. M&m VTO: multi-garment virtual try-on and editing. *CoRR*, abs/2406.04542, 2024.
- [217] Luyang Zhu, Dawei Yang, Tyler Zhu, Fitsum Reda, William Chan, Chitwan Saharia, Mohammad Norouzi, and Ira Kemelmacher-Shlizerman. Tryondiffusion: A tale of two unets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [218] Zhen Zhu, Tengpeng Huang, Baoguang Shi, Miao Yu, Bofei Wang, and Xiang Bai. Progressive pose attention transfer for person image generation, 2019.
- [219] Xingxing Zou, Xintong Han, and Waikeng Wong. Cloth4d: A dataset for clothed human reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.