Photorealistic Virtual Try-on with Generative Models

Luyang Zhu

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

Doctor of Philosophy

University of Washington

2024

Reading Committee:
Ira Kemelmacher-Shlizerman, Chair
Steven M. Seitz
Brian Curless

Program Authorized to Offer Degree:
Computer Science and Engineering

University of Washington

Abstract

Photorealistic Virtual Try-on with Generative Models

Luyang Zhu

Chair of the Supervisory Committee:
Ira Kemelmacher-Shlizerman
Computer Science and Engineering

Virtual try-on (VTO) is revolutionizing the online apparel shopping experience, enabling customers to see how a particular fashion item would look on them. Despite significant progress, current VTO methods still encounter challenges such as accurately warping garments under large pose gap and heavy occlusion, as well as preserving body shape and identity of the person under the new garment. Additionally, most research focuses on upper-body VTO, whereas a full-body VTO that allows for garment mix-and-match is more desirable in real-world scenarios. In my thesis, I address above challenges by developing generative models tailored for the VTO task.

First, I propose TryOnDiffusion, the first method capable of try-on synthesis at 1024×1024 resolution for various body poses and shapes while preserving garment details. Previous methods either focus on garment detail preservation without effective pose and shape variation, or allow try-on with the desired shape and pose but lack garment details. In this project, I show that the underlying reason for this challenge is a widely-used two-stage pipeline consisting of an explicit warping model and a blending GAN model. To solve this issue, I propose a diffusion-based architecture that unifies two UNets (referred to as Parallel-UNet), which can warp the garment implicitly with cross attention, in addition to warping and blending in a single network pass.

Next, I present M&M VTO, which extends TryOnDiffusion from upper body VTO to full

body VTO, allowing to mix and match multiple garments. To preserve intricate garment details required by full body VTO, I propose a single-stage diffusion model in the pixel space that is trained progressively. To solve a common identity loss problem in current VTO methods, I design a novel architecture named VTO UNet Diffusion Transformer (VTO-UDiT) to disentangle denoising from person specific features, allowing for a highly effective finetuning strategy. Furthermore, M&M VTO also supports garment layout editing via text inputs finetuned on multi-modal foundation models.

Finally, I show how we can train generative models on synthetic datasets for 3D clothed human reconstruction, which is an important component towards VTO in the 3D world. I propose reconstructing NBA players, which takes as input a single photo of a clothed player in any basketball pose and outputs a high resolution mesh and 3D pose for that player. Key to my approach is a deep neural skinning approach for creating poseable, skinned models of NBA players, and a large database of meshes derived from the video game. Although trained only on synthetic data, the proposed pipeline generalizes well to real-world images even under heavy occlusion.

TABLE OF CONTENTS

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my PhD advisors, Ira Kemelmacher-Shlizerman, Steve Seitz and Brian Curless, for their invaluable support throughout my PhD studies. Their profound knowledge and insightful feedbacks have significantly shaped my thinking and research direction. Their constant encouragement and rigorous standards pushed me to refine my work to its best form. Their impact extends far beyond my time in the PhD program and will resonate with me throughout my career.

I am grateful to the individuals who supported and guided me during my internships at NVIDIA, Adobe and Google: Dieter Fox, Arsalan Mousavian, Yu Xiang, Hammad Mazhar, Jozef van Eenbergen, Shoubhik Debnath, Ruben Villegas, Jimei Yang, Jun Saito, Aaron Hertzmann, Davis Rempe, Dawei Yang, Yingwei Li, Nan Liu, Hao Peng, Chris Lee, Srivatsan Varadharajan, Tyler Zhu, Alan Yang, Chunhui Gu, Varsha Ramakrishnan, Andreas Lugmayr, Innfarn Yoo, Yasamin Jafarian, Fitsum Reda, William Chan, Chitwan Saharia, Mohammad Norouzi, Daniel Watson and Ricardo Martin-Brualla. Their contributions made these experiences not only memorable but also profoundly influential in shaping my future career path.

I am thankful for having had the opportunity to meet and work alongside many researchers, labmates, and postdoctoral fellows at UW who made my experience thoroughly enjoyable: Adriana Schulz, Linda Shapiro, Ranjay Krishna, Gilbert Bernstein, Kostas Rematas, Soumyadip Sengupta, Aditya Sankar, Edward Zhang, Aleksander Holynski, Keunhong Park, Jeong Joon Park, Chung-Yi Weng, Xuan Luo, Isaac Tian, Roy Or-El, James Noeckel, Yifan Wang, Kuo-Hao Zeng, Adam Fishman, Nikita Haduong, Xiaojuan Wang, Yuxuan Mei, Vivek Jayaram, Teerapat Jenrungrot, Johanna Karras, Jingwei Ma, Alice Gao,

Mengyi Shan, Benlin Liu, Meng-Li Shih, Bowei Chen, Ben Jones, Haisen Zhao and Chenming Wu.

This thesis would not be possible without the love and support of my family—my parents and my parents-in-law. To my parents, thank you for instilling in me the values of hard work and perseverance, and for the countless sacrifices you made to see me succeed. To my parents-in-law, thank you for embracing me as your own and for your generous support and wise counsel when I needed it most.

Most importantly, I owe a profound debt of gratitude to my wife, Aurelia, for choosing me as her husband and entering my world. Her steadfast love and support throughout these years have been my anchor. Her resilience and understanding, especially during the most challenging times, provided the peace and stability I needed to persevere. Her encouragement was a constant source of inspiration that pushed me to strive harder. It is with the deepest admiration and overwhelming gratitude that I dedicate this thesis to her, thank you for being my cornerstone, my confidante, and my joy.

DEDICATION

to my wife, Aurelia, for being my constant source of strength and joy

Chapter 1

INTRODUCTION

Figure 1.1: Cher's Closet presented in the 1995 film *Clueless* [69]. The main character Cher Horowitz used this digital closet to mix and match her outfits (left), and virtual try-on outfits (right).

Virtual try-on (VTO) has long been a dream for many people. In the 1995 movie *Clueless* [69], the main character, Cher Horowitz, had a digital closet (Figure 1.1) that let her browse her clothes on a computer, mix and match outfits, and preview how they would look on her. At the time, every girl wanted a closet like Cher's to help pick out their "OOTD" (outfit of the day) and see how different clothes would look on them. But back then, Cher's high-tech closet was just a fantasy seen in the movies, far from reality.

As artificial intelligence advances and large fashion datasets [114] become available, VTO technology is evolving and increasingly shaping the future of online apparel shopping (Figure 1.2). For retailers, VTO reduces the chances of returns due to fit or style issues. It also provides valuable insights into customer preferences and behaviors, helping businesses better

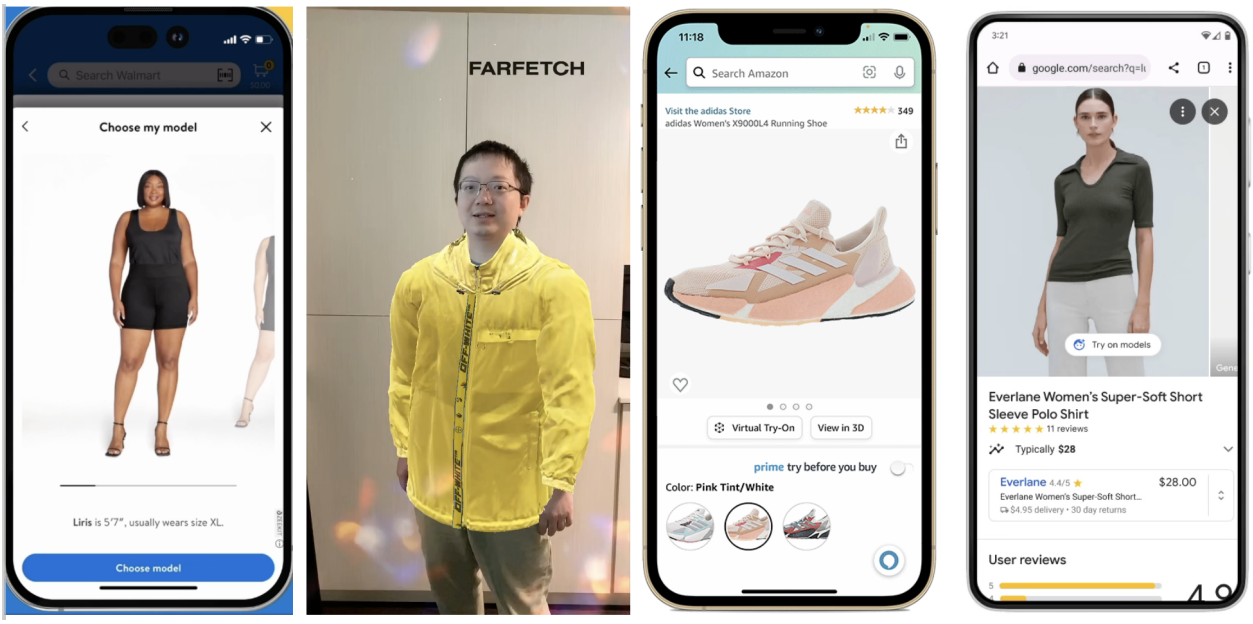

Figure 1.2: Multiple companies released their VTO features. From left to right: Walmart apparel VTO, Snapchat VTO filter for Farfetch, Amazon shoe VTO and Google apparel VTO.

tailor their products and marketing approaches. For consumers, VTO offers the convenience of trying on clothes from home, which saves time and simplifies the shopping process. By seeing how products look on them, customers can make better-informed decisions, boosting their confidence in purchases, particularly for items like clothing.

Despite significant progress in this field, current VTO methods still face challenges. Those include, 1) garment quality rendering in the VTO experience, e.g., photorealism and realistic warping, 2) person identity preservation, e.g., to make sure that the input person's body shape and identity is shown correctly with the new garment.

The first challenge stems from the complex task of non-rigidly warping a garment to fit a target body shape without distorting its patterns and texture. Related works [31, 105] addresses this challenge through a two-step pipeline. They first train a warping model to

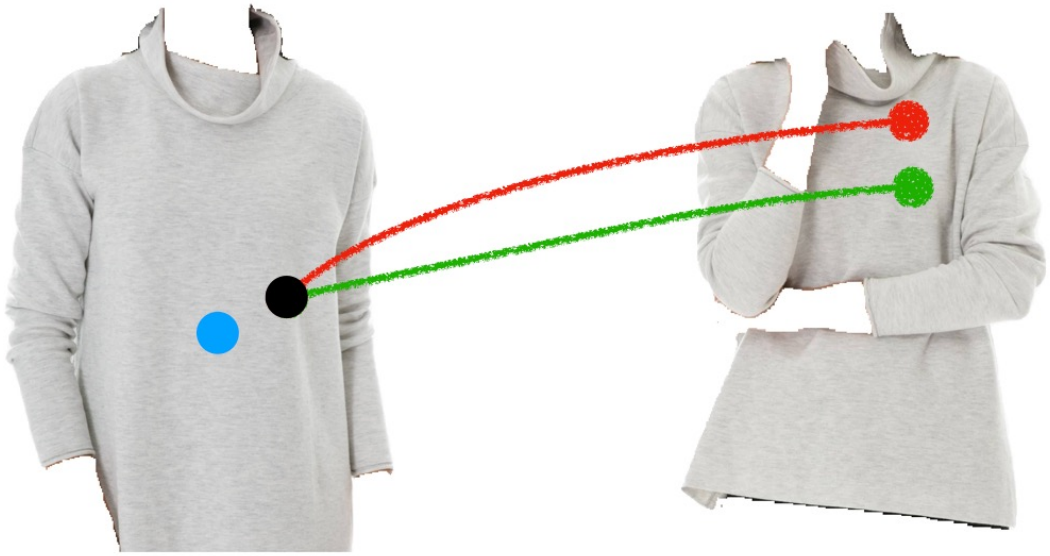

Figure 1.3: Problems for explicit warping. Ill-posed one-to-one pixel correspondence when occlusion exists: For the blue pixel in the input garment, there is no corresponding pixel in the groundtruth garment due to self-occlusion. Inadequate to supervise flow fields training with image-space perceptual loss: For the black pixel, both red and green pixels in thr groundtruth garment can lead to a small perceptual loss.

estimate a dense flow field, and use this field to explicitly warp the source garment. Second, they train a blending model to blend the warped garment and the target person. As shown in Figure 1.3, the explicit warping is problematic as it requires an invalid one-to-one pixel correspondence when occlusion exists. Additionally, obtaining groundtruth flow fields from real images is challenging. Most methods circumvent this issue by using the perceptual loss in the image space between the warped and groundtruth garment. However, there exists multiple flow solutions that can yield small perceptual losses, indicating that supervising the warping model solely with perceptual loss is also inadequate.

The second challenge arises from the disparity between the training and testing data.

Ideally, training data for the virtual try-on would include a triplet of a person in garment A, the same person in the same pose wearing garment B, another person in a different pose in garment A. However, obtaining such data at scale is practically impossible. Consequently, most methods [65, 177, 195, 82, 194, 47, 40, 68, 193, 8] are trained on paired samples and tested on unpaired ones. To prevent the leakage of the original garment, most methods employ a clothing-agnostic representation that completely eliminates the garment information. Unfortunately, this representation also removes body parts from the person input, causing issues with person identity preservation.

In addition to the aforementioned main challenges, there are other less explored issues. Firstly, most research focuses on upper body VTO, but there's a clear preference for the mix and match VTO that can try on multiple garments for the full body. Secondly, the current research typically doesn't support VTO that allows for different garment layouts, like rolling sleeves up or down, limiting how users can visualize and adjust their outfits.

In this thesis, I explore methods for photorealistic VTO with generative models. In particular, my contributions include: 1) The first diffusion models designed for the VTO task that can achieve realistic garment warping under very large pose differences and heavy occlusions. 2) A single-stage diffusion model designed for the mix and match VTO that can synthesize intricate garment details, preserve person identity with minimal space requirement and edit garment layout with text inputs. 3) A new method for creating poseable, skinned models of clothed human, and a large database of meshes derived from the NBA2K19 video game [33]. Below I introduce each of these contributions.

Realistic garment warping. In Chapter 3, I propose TryOnDiffusion, a diffusion-based architecture that unifies two UNets (referred to as Parallel-UNet), which allows us to preserve garment details and warp the garment for significant pose and body change in a single network. Our key observation is that the process of constructing cost volumes in flow-based models [170] is essentially the same as creating attention maps for scaled dot-product attention [174]. Moreover, in the context of VTO, our main concern is obtaining the final warped image rather than getting an accurate intermediate flow field, where the latter is

Figure 1.4: TryOnDiffusion is able to synthesize realistic garment warping even under heavy occlusion and extreme pose differences.

more complex. Therefore, I propose to replace the traditional explicit grid sampling with a learnable weighted sum. This leads to the implicit warping mechanism via the cross-attention operation. Inspired from the success of perceptual loss and style loss [85, 143], I further propose to use the same network to perform warping and blending, which allows the two processes to exchange information at the feature level rather than at the pixel level. As shown in Figure 1.4, TryOnDiffusion can achieve realistic garment warping even under extreme body pose differences.

Mix and match VTO. In Chapter 4, I present M&M VTO–a mix and match VTO method .

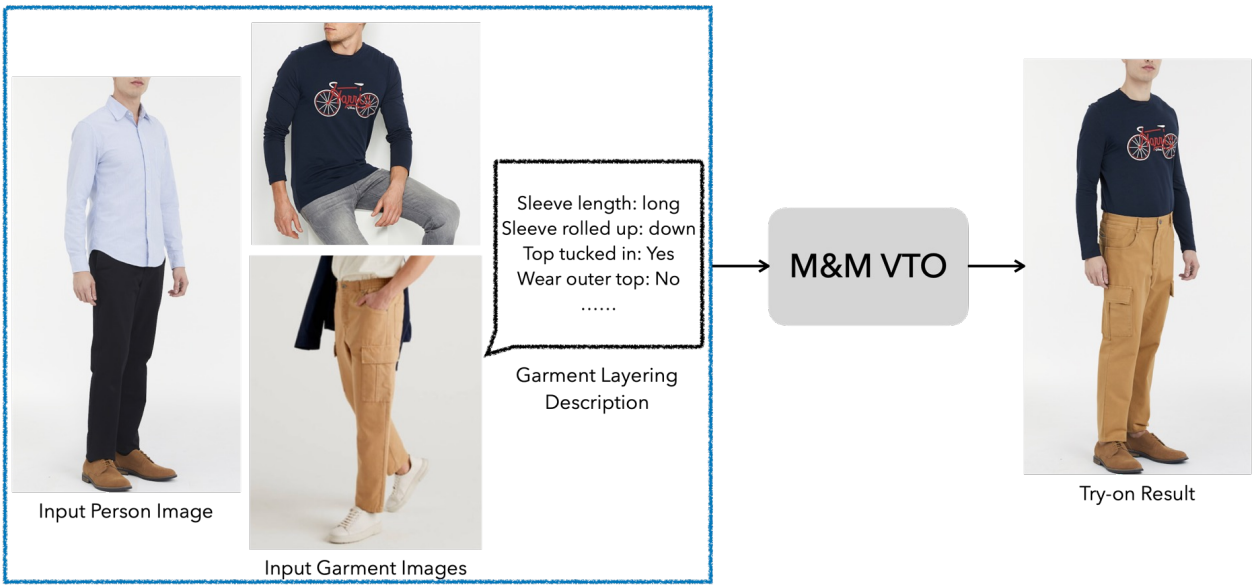

Figure 1.5: Task definition for M&M VTO. Given an image of upper body garment, an image of lower body garment, a text description of garment layout, and an image of a person, M&M VTO generates a visualization of how those garments (in the desired layout) would look like on the given person.

As shown in Figure 1.5, M&M VTO takes as input multiple garment images, text description for garment layout and an image of a person, and output a visualization of the given person wearing garments in the layout specified by the text description. Compared to upper body VTO, the mix and match VTO, which includes the full body from head to feet, requires the model to synthesize more intricate details because garments occupy a smaller portion of the image. I observe that cascaded diffusion models failed to achieve this task as the groundtruth in the base diffusion model does not contain intricate details due to excessive downsampling. Super-res diffusion models can not hallucinate those details if the base diffusion model does not include them. To overcome this issue, I suggest using a single-stage diffusion model that is progressively trained from lower to higher resolution data. The training at lower resolutions is aimed at learning mid-level structures and the warping process, whereas the training at higher resolutions concentrates on capturing high-frequency details. I also propose

a novel architecture named VTO UNet Diffusion Transformer (VTO-UDiT), to disentangle denoising from person specific features, allowing for a highly effective finetuning strategy for identity preservation.

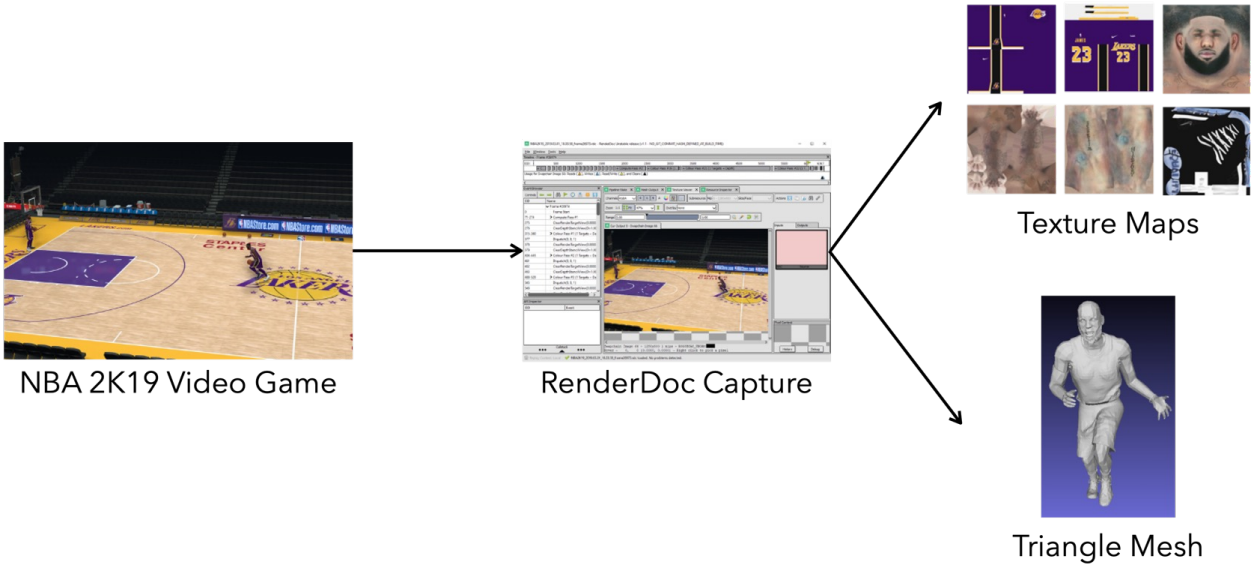

Figure 1.6: The data collection pipeline for the NBA2K dataset. Video game is a good data source for scaling up 3D clothed human meshes.

Skinned models of clothed human. 3D clothed human reconstruction from a single image is a crucial element for enhancing the VTO experience. It enables consumers to visualize how garments would appear on them from various viewpoints. However, current approaches in this field are constrained by the availability of large-scale datasets, which are costly to collect. To address this challenge, I have created techniques that can efficiently collect 3D clothed human datasets from video game engines. As shown in Figure 1.6, textured triangle mesh can be extracted from the GBuffer by intercepting calls between the game engine and GPU using graphics debugging softwares like RenderDoc [146]. Given this dataset, I am able to train skinned models of clothed human, parametrized as neural networks, that can capture pose-dependent garment details.

Chapter 2

RELATED WORK

In this chapter, I conduct a survey of literature necessary for creating photorealistic virtual try-on (VTO) with generative models. I begin by examining research on image-based VTO, detailing the breakthroughs and limitations of milestone papers. Next, I review the literature of diffusion models, a special family of generative models that I used for my projects. For this section, I will mainly focus on the application rather than the theory of diffusion models. Finally, I will briefly introduce methods for 3D clothed human reconstruction.

2.1 Image-Based Virtual Try-on

Image-based VTO methods [65, 177, 195, 82, 194, 31, 47, 40, 68, 193, 105, 8, 120, 197, 145] aim to create an image of a target individual wearing a source garment, given a pair of images (target person, source garment). An optimal try-on result should meet the following requirements: 1) The target person's identity, including factors like skin tone, body size, and tattoos, should be maintained. 2) The source garment should be warped properly to fit the target person's pose and body shape, even in cases of heavy occlusion. 3) Detailed texture patterns should be preserved, and garment folds should be realistically recreated to align with the new poses. 4) The try-on results should be high resolution and include all the details customers wish to see. A number of studies have been proposed to meet these criteria.

Thin-Plate-Spline Warping. The seminal work, VITON [65], is a significant contribution to the literature, paving the way for solutions in image-based VTO. Many subsequent works have built upon its framework and attempted to enhance it from various perspectives. VITON proposes a coarse-to-fine pipeline, guided by the thin-plate-spline [17] (TPS) warping

of source garments. Specifically, it first trains an encoder-decoder network to predict the coarse try-on result and the warped garment mask, given the clothing-agnostic representation and the layflat garment. The network is trained with L1 loss for the warped mask and perceptual loss for the coarse try-on result. Following this, TPS parameters are estimated based on shape context matching [11], which is then used to compute the warped garment. A refinement neural network is then trained to estimate the alpha mask, given the coarse try-on result and the TPS-warped garment. This alpha mask is used to generate the final try-on result by blending the two inputs of the refinement network. The refinement network is trained with perceptual loss between the final try-on result and groundtruth.

Two notable contributions from VITON are worth highlighting. Firstly, VITON found a way to relax the training data requirements for the try-on task. In theory, we need triplets of data samples to train a try-on system: person A wearing garment A in pose A, any person wearing garment B in pose B (or a layflat image of garment B), and person A wearing garment B in pose A. The first two images serve as input to the try-on system, and the last one is the groundtruth to supervise the training. However, in practice, it is prohibitively expensive to collect these desired triplets on a large scale. By introducing clothing-agnostic representations, VITON bypasses the need for triplets, only requiring a pair of images: a layflat garment and a person wearing the same garment. This kind of training data is readily available and easily accessible because many online retailers provide them by default for their products. Specifically, VITON extracts the following clothing-agnostic representations from the person input image: 2D human pose heatmaps, a rough body shape mask, and face/hair regions of the person. These representations prevent the garment information leak while partially preserving the person's identity. Secondly, VITON breaks down image-based virtual try-on into two stages: a warping model and a blending model, a pipeline that has been adopted by the majority of subsequent works.

One significant challenge for VITON is accurately warping the garment, especially under heavy occlusion and large pose gaps. VITON employs shape context matching to estimate TPS warping parameters. However, shape context matching, based on deterministic feature

descriptors, may not be robust enough under heavy occlusions. CP-VTON [177] seeks to replace shape context matching with a learnable geometry matching module, where TPS parameters are directly estimated through a neural network. Specifically, it first extracts features from clothing-agnostic representations and the garment image using two networks. Then a correlation tensor is computed from extracted features by dot-product. Finally, a regression network is used to predict TPS parameters from the correlation tensor. Since TPS warping is differentiable, CP-VTON can train the geometry matching module using the L1 loss between the warped garment and the groundtruth garment. Although CP-VTON improves upon VITON, it still relies on TPS warping, which struggles with large deformations. This is primarily because the sparse control points used in TPS warping fails to model complex deformations, leading to over-bending or distortion of the garment.

Apprearance Flow Warping. To overcome the above issue, ClothFlow [64] directly estimates dense flow fields with a neural network. Specifically, ClothFlow utilizes two feature pyramid networks to refine the estimation of the flow field progressively in a cascaded warping manner. However, it is almost impossible to get the dense supervision for the flow field on real images, thus ClothFlow resorts to training the network with a warping loss in the image space. This is ill-posed due to the infinite solutions of flow fields that exist to warp one image to another, especially for garment images where repetitive texture pattern is very common. Consequently, flow network easily falls into the local minima and fails to capture fine details. SDAFN [8] tries to mitigate the ill-posed problem by predicting multiple flow fields and combining warped features through deformable attention [204]. However, occlusions might invalidate the dense correspondence between two images and lead to the pixel-squeezing artifacts.

High-Resolution VTO. Another significant issue for above methods is that they can only synthesize try-on results up to a resolution of 256×256. VITON-HD [31] is the first work that can generate 1024×768 try-on images. The key contribution of VITON-HD is a novel architecture for the blending model, designed specifically to address the misalignment between the

warped garment and the desired garment region. Inspired from SPADE [130], VITON-HD introduces a blending model equipped with ALIgnment-Aware Segment (ALIAS) normalization. ALIAS normalization standardizes activations separately according to the aligned and misaligned area. These normalized activations are then modulated by affine parameters conditioned on the warped mask and the predicted human parsing map. The aligned area introduced in ALIAS normalization is the intersection of the warped mask and predicted human parsing, while the misaligned area is the subtraction of the aligned area from the warped mask. HR-VITON [105] further improves VITON-HD by simultaneously predicting human parsing maps and garment warping flows, which allows for information exchange between two tasks. To enhance the try-on performance in real-world applications, HR-VITON further introduces discriminator rejection to filter out incorrect human parsing map predictions.

Parser-free VTO. Human parsing prediction is a critical element of image-based try-on systems. A realistic try-on result relies heavily on the accuracy of the human parsing results. However, even the most advanced human parsing algorithms, such as Graphonomy [50], are susceptible to errors, particularly for garment types that were not present during training. WUTON [82] attempts to solve this issue by training a parser-based teacher network first, followed by a parser-free student network. The groundtruth for student network is the fake try-on results generated by the teacher network, leading to the student network mimicing the teacher's behavior. One disadvantage of WUTON is that any imperfections present in the teacher network are also passed on to the student network due to the nature of the knowledge distillation. PF-AFN [47] offers a solution to this problem by proposing a "teacher-tutor-student" knowledge distillation scheme. Rather than using the fake try-on result produced by the teacher network as the direct supervision, PF-AFN uses it as input to the student network, and supervises the student training with real person images. By doing this, PF-AFN can discard the clothing-agnostic representations and ensures that the supervising information for the student network is free of artifacts. Moreover, PF-AFN also distills the garment warping flow based on the quality of the teacher's try-on output. It's important to note, however, that PF-AFN struggles to retain the identity of the target

person. The main reason is that the fake try-on results produced by the parser-based teacher network (also the input of the student network) can not retain the person identity as it relies on the clothing-agnostic representations.

Multi-Garment VTO. The aforementioned methods are primarily focused on the upper-body try-on, which significantly limits their practical applications. DiOr [36] presents the first try-on system that supports full-body try-on and different garment interactions. The key innovation of DiOr is a recurrent generation pipeline that can sequentially dress a person with garments. Consequently, trying on the same set of garments in different orders will result in different garment layerings. However, the try-on quality of DiOr can be limited, especially in challenging scenarios such as rare body poses, unusual garment shapes, and complex garment layering. The performance of DiOr should improve with larger scale training data as well as more advanced warping and blending models. COTTON [28] proposes a Clothing-Oriented Transformation TryOn Network (COTTON) for multi-garment VTO. To achieve different garment interactions like tuck-in or tuck-out, COTTON introduces adjustable clothing landmarks to provide clothing size information based only on images.

Warping-Free VTO. Another interesting work to mention in the literature is TryOnGAN [107]. TryOnGAN does not adopt the two-stage pipeline as proposed in VITON that requires paired training data. Instead, it trains a pose-conditioned StyleGAN2 [91] on unpaired fashion images and running optimization in the latent space to achieve try-on. By optimizing the latent space, however, it loses garment details that are less represented by the latent space. This becomes evident when garments have a pattern or details like pockets, or special sleeves.

2.2 Diffusion Models

Diffusion models [165, 167, 73] have recently emerged as the most powerful family of generative models. They consists of a deterministic forward process and a learnable backward process. The forward process gradually corrupts data samples by adding random noise, while

the backward process learns to denoise the data from a pure Gaussian noise step by step. Unlike GANs [51, 20], diffusion models have better training stability and mode coverage, allowing them to model the underlying distribution even the size of the training set is in billions level.

Text-to-image Diffusion Models. Diffusion models have achieved state-of-the-art results on various image generation tasks, especially for the text-to-image generation [157, 141, 151]. Imagen employs cascaded diffusion models [74] to directly synthesize images in the pixel space given a text prompt. The key contribution of Imagen is to use a large launguage model [140] pretrained on text only to encode the text prompt. Imagen finds that using a larger language model can boost both sample fidelity and image-text alignment, which is much more effective than increasing the size of the image diffusion model. latent diffusion model (LDM) [151] is another popular text-to-image model. Unlike Imagen which directly operates in the pixel space, LDM first trains a VAE [95] to encode images into a smaller latent space, and then train a diffusion model in the latent space conditioned on the text prompt. Thus stable diffusion is more efficient in terms of the model size and the inference time.

ControlNet [198] extends LDM with spatial conditioning controls. ControlNet freeze the pretrained large diffusion models, and reuses the encoder as a backbone to learn a diverse set of conditional controls. To ensure that finetuning does not affect the underlying distribution of pretrained diffusion models, the trainable conditional encoder is initialized to zeros. Although being successful, ControlNet only works well for image-to-image translation problems where input and output pixels are perfectly aligned. However, it is not directly applicable to the VTO task as VTO involves highly non-linear transformations like garment warping.

Text-to-video Diffusion Models. Recently, several works try to tackle the harder problem of the text-to-video generation. Following Imagen, Imagen video [72] utilizes cascaded diffusion models to generate 1280×768 24FPS videos. The cascaded diffusion models are parametrized as 3D UNets, including a base video generation model and several spatial and

temporal video super-resolution models. Since the total model size is larger than 11B parameters, Imagen video also applies progressive distillation for fast and high quality sampling. Stable Video Diffusion (SVD) [15, 14] inserts temporal layers into a pretrained image LDM and finetunes it on the video data to generate latent key frames. In addition, it trains a interpolation LDM and a spatial diffusion model upsampler on the video data to increase the resolution and frame rates. For the long video generation, SVD also trains a video prediction LDM by conditioning on starting frames. Compared to Imagen video, SVD is much more resource efficient. Recently, Sora [22] can generate imppressive high-fidelity video of one minute long. Sora first compress raw videos into spacetime latent patches with a VAE. Then it trains a diffusion transformer [134] in the latent space conditioned on the text for video generation. By scaling up the size of model and dataset, Sora can be seen as the first step for building a simulator of the physical world.

Identity Preservation of Diffusion Models. Despite good generalization, text-to-image diffusion models often struggle with preserving the identity of a specific subject. The seminal work DreamBooth [154] finetuned Imagen on a few images of a given subject,such that it learns to bind a unique identifier with that specific subject. To avoid overfitting to the target subject, DreamBooth apply a class-specific prior preservation loss, which encourages the model to generate diverse instances belong to the subject's class. To get rid of the lengthy fine-tuning processes of DreamBooth, InstantID [178] proposes to train an IdentityNet conditioned on face landmarks and embeddings, whose features are used to cross attend the original UNet. In addition, InstantID learns a lightweight adaptor with decoupled cross-attention, allowing for the conditioning of visual inputs. Thus, InstantID does not require finetuning for each new subject, and can preserve the identity of the target subject given a single reference image.

Image Editing with Diffusion Models. The success of text-to-image diffusion models led to text-based image editing. DiffEdit [34] infers a region mask automatically based on spatial difference in noise prediction from original and new input prompts. Then DiffEdit

get the initial latent of the input image using DDIM inversion process [166]. Finally, DiffEdit per forms image editing during DDIM process by compositing original and new noisy images using the inferred mask. Prompt-to-Prompt (P2P) [70] finds that the spatial layout of generated images are based on cross-attention maps. Thus, P2P proposes a mask-free image editing method by developing different strategies for editing cross-attention maps. InstructPix2Pix [21] can directly edit images with a diffusion model given the input image and editing prompt. To achieve this, InstructPix2Pix first builds a dataset consisting of image editing examples. Specifically, InstructPix2Pix first prompts GPT-3 [23] to generate triplets of text edits, including a prompt for the original image, an editing prompt and a prompt for the edited image. Then InstructPix2Pix converts two image prompts into corresponding images using the P2P technique. Given this dataset, InstructPix2Pix trains a diffusion model conditioned on the input image and the editing prompt, which can edits images in a matter of seconds.

2.3 3D Clothed Human Reconstruction

3D clothed human reconstruction can be classified into two categories. The first category reconstructs clothed human based on parametric human body models like SMPL [115]. The second category directly estimates the geometry of clothed human in different 3D representations, without the help of parametric human body models.

Parametric Methods. Parametric human body models like SMPL are trained on human captures with minimal clothing, thus they are not able to model diverse garment worn by the people. To tackle this challenge, Weng *et al.* [182] demonstrate 2D warping of depth and normal maps from a single photo silhouette. Specifically, Given a photo of a person, Weng *et al.* first run off-the-shelf algorithms to extract 2D pose and person segmentation. Then, they fit a SMPL model to the 2D pose, and render the fitted model into the image space as a SMPL silhouette, a normal map and a skinning map. After that, they warp the normal/skinning map to align the SMPL silhouette with the person segmentation. The warped normal map is used to rebuild a depth map which are combined with warped skinning mas to create

a clothed human mesh. Alternatively, Xiang *et al.* [186] propose to model clothed human with a two-layer mesh representation, where the garment mesh is on top of the body mesh. To better model the interaction between the two-layer meshes, Xiang *et al.* propose to use a temporal convolution network to predict the clothing latent code based on a sequence of input skeletal poses.

Non-parametric Methods. As we mentioned above, non-parametric methods propose various 3D representations to model clothed humans by training on representative synthetic data. PIFu [159] introduce an efficient implicit representation that aligns the pixels of 2D images with the 3D shape of their corresponding objects. Using this representation, PIFu trains an end-to-end deep learning model on synthetic dataset for reconstructing highly detailed clothed humans. PIFuHD [160] extends PIFu to high-resolution images by proposing a multi-level architecture. The coarse-level model takes as input lower-resolution images and focuses on global reasoning. The features from the coarse-level model, as well as higher-resolution images are fed into the fine-level model to estimate highly detailed geometry.

Chapter 3

TRYONDIFFUSION: A TALE OF TWO UNETS

Figure 3.1: TryOnDiffusion generates apparel try-on results with a significant body shape and pose modification, while preserving garment details at 1024×1024 resolution. Input images (target person and garment worn by another person) are shown in the corner of the results.

This chapter presents the collaborative research project with Dawei Yang, Tyler Zhu,

Fitsum Reda, William Chan, Chitwan Saharia, Mohammad Norouzi, Ira Kemelmacher-Shlizerman. The findings from this work were initially published in CVPR 2023 [203]. The subsequent analysis and comparisons to related studies in this chapter are based on the prevailing state-of-the-art during that time.

Virtual apparel try-on aims to visualize how a garment might look on a person based on an image of the person and an image of the garment. Virtual try-on has the potential to enhance the online shopping experience, but most try-on methods only perform well when body pose and shape variation is small. A key open problem is the non-rigid warping of a garment to fit a target body shape, while not introducing distortions in garment patterns and texture [65, 177, 31].

When pose or body shape vary significantly, garments need to warp in a way that wrinkles are created or flattened according to the new shape or occlusions. Related works [31, 105, 176] have been approaching the warping problem via first estimating pixel displacements, e.g., optical flow, followed by pixel warping, and postprocessing with perceptual loss when blending with the target person. Fundamentally, however, the sequence of finding displacements, warping, and blending often creates artifacts, since occluded parts and shape deformations are challenging to model accurately with pixel displacements. It is also challenging to remove those artifacts later in the blending stage even if it is done with a powerful generative model. As an alternative, TryOnGAN [107] showed how to warp without estimating displacements, via a conditional StyleGAN2 [91] network and optimizing in generated latent space. While the generated results were of impressive quality, outputs often lose details especially for highly patterned garments due to the low representation power of the latent space.

In this paper, we present TryOnDiffusion that can handle large occlusions, pose changes, and body shape changes, while preserving garment details at 1024×1024 resolution. TryOnDiffusion takes as input two images: a target person image, and an image of a garment worn by another person. It synthesizes as output the target person wearing the garment. The garment might be partially occluded by body parts or other garments, and requires significant deformation. Our method is trained on 4 Million image pairs. Each pair has the

same person wearing the same garment but appears in different poses.

TryOnDiffusion is based on our novel architecture called Parallel-UNet consisting of two sub-UNets communicating through cross attentions [174]. Our two key design elements are implicit warping and combination of warp and blend (of target person and garment) in a single pass rather than in a sequential fashion. Implicit warping between the target person and the source garment is achieved via cross attention over their features at multiple pyramid levels which allows to establish long range correspondence. Long range correspondence performs well, especially under heavy occlusion and extreme pose differences. Furthermore, using the same network to perform warping and blending allows the two processes to exchange information at the feature level rather than at the color pixel level which proves to be essential in perceptual loss and style loss [85, 143]. We demonstrate the performance of these design choices in Sec. 3.5.

To generate high quality results at 1024×1024 resolution, we follow Imagen [157] and create cascaded diffusion models. Specifically, Parallel-UNet based diffusion is used for 128×128 and 256×256 resolutions. The 256×256 result is then fed to a super-resolution diffusion network to create the final 1024×1024 image.

In summary, the main contributions of our work are: 1) try-on synthesis at 1024×1024 resolution for a variety of complex body poses, allowing for diverse body shapes, while preserving garment details (including patterns, text, labels, etc.), 2) a novel architecture called Parallel-UNet, which can warp the garment implicitly with cross attention, in addition to warping and blending in a single network pass. We evaluated TryOnDiffusion quantitatively and qualitatively, compared to recent state-of-the-art methods, and performed an extensive user study. The user study was done by 15 non-experts, ranking more than 2K distinct random samples. The study showed that our results were chosen as the best 92.72% of the time compared to three recent state-of-the-art methods.

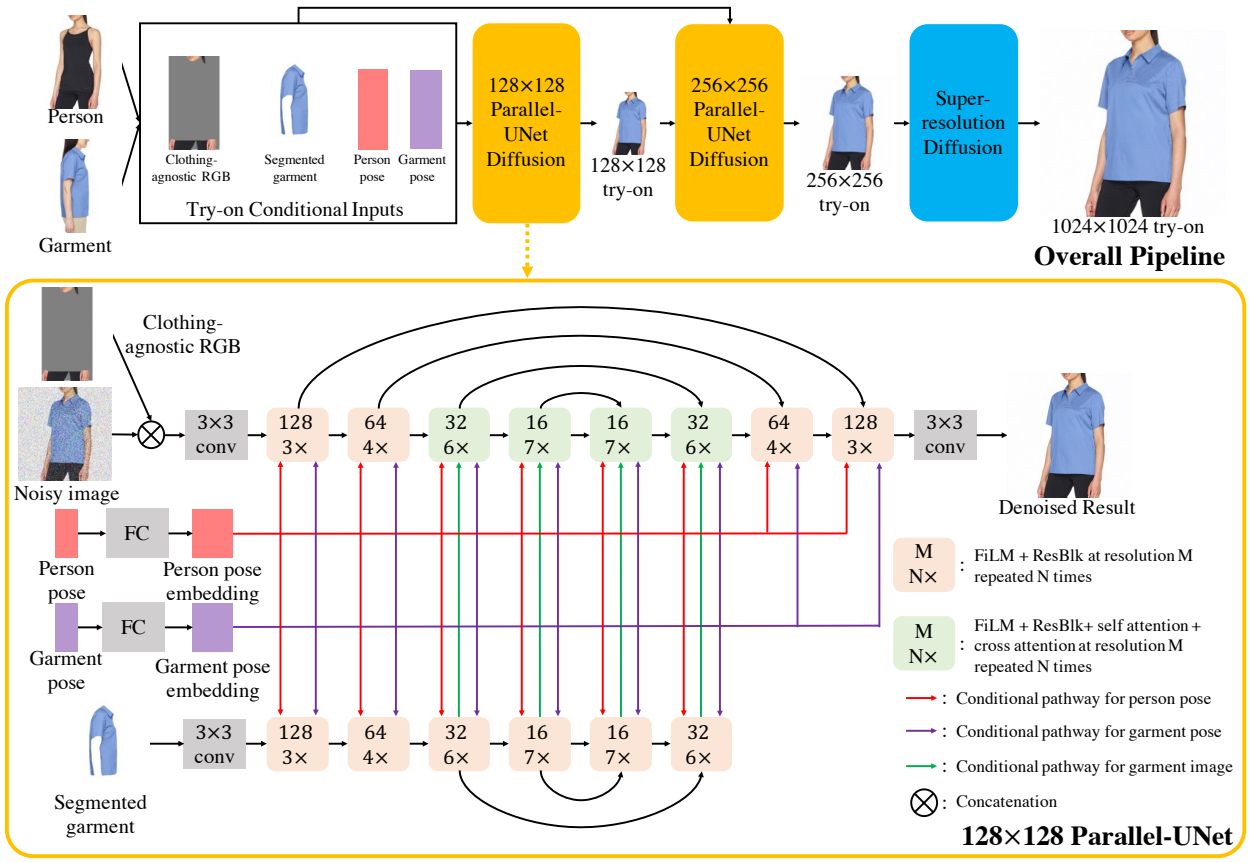

Figure 3.2: Overall pipeline (top): During preprocessing step, the target person is segmented out of the person image creating "clothing agnostic RGB" image, the target garment is segmented out of the garment image, and pose is computed for both person and garment images. These inputs are taken into 128×128 Parallel-UNet (key contribution) to create the 128×128 try-on image which is further sent as input to the 256×256 Parallel-UNet together with the try-on conditional inputs. Output from 256×256 Parallel-UNet is sent to standard super resolution diffusion to create the 1024×1024 image. The architecture of 128×128 Parallel-UNet is visualized at the bottom, see text for details. The 256×256 Parallel-UNet is similar to the 128 one, and provided in supplementary for completeness.

3.1 Related Work

Image-Based Virtual Try-On. Given a pair of images (target person, source garment), image-based virtual try-on methods generate the look of the target person wearing the source garment. Most of these methods [65, 177, 195, 82, 194, 31, 47, 40, 68, 193, 105, 8, 120, 197, 145] decompose the try-on task into two stages: a warping model and a blending model. The seminal work VITON [65] proposes a coarse-to-fine pipeline guided by the thin-plate-spline (TPS) warping of source garments. ClothFlow [64] directly estimates flow fields with a neural network instead of TPS for better garment warping. VITON-HD [31] introduces alignment-aware generator to increase the try-on resolution from 256×192 to 1024×768. HR-VITON [105] further improves VITON-HD by predicting segmentation and flow simultaneously. SDAFN [8] predicts multiple flow fields for both the garment and the person, and combines warped features through deformable attention [204] to improve quality.

Despite great progress, these methods still suffer from misalignment brought by explicit flow estimation and warping. TryOnGAN [107] tackles this issue by training a pose-conditioned StyleGAN2 [91] on unpaired fashion images and running optimization in the latent space to achieve try-on. By optimizing the latent space, however, it loses garment details that are less represented by the latent space. This becomes evident when garments have a pattern or details like pockets, or special sleeves.

We propose a novel architecture which performs implicit warping (without computing flow) and blending in a single network pass. Experiments show that our method can preserve details of the garment even under heavy occlusions and various body poses and shapes.

Diffusion Models. Diffusion models [165, 167, 73] have recently emerged as the most powerful family of generative models. Unlike GANs [51, 20], diffusion models have better training stability and mode coverage. They have achieved state-of-the-art results on various image generation tasks, such as super-resolution [158], colorization [156], novel-view synthesis [181] and text-to-image generation [157, 141, 151, 154]. Although being successful, state-of-the-art diffusion models utilize a traditional UNet architecture [153, 73] with channel-wise concate-

nation [158, 156] for image conditioning. The channel-wise concatenation works well for image-to-image translation problems where input and output pixels are perfectly aligned (e.g., super-resolution, inpainting and colorization). However, it is not directly applicable to our task as try-on involves highly non-linear transformations like garment warping. To solve this challenge, we propose Parallel-UNet architecture tailored to try-on, where the garment is warped implicitly via cross attentions.

3.2 Pipeline Overview

Fig. 3.2 provides an overview of our method for virtual try-on. Given an image I_p of person p and an image I_g of a different person in garment g, our approach generates try-on result I_{tr} of person p wearing garment g. Our method is trained on paired data where I_p and I_g are images of the same person wearing the same garment but in two different poses. During inference, I_p and I_g are set to images of two different people wearing different garments in different poses. We begin by describing our preprocessing steps, and a brief paragraph on diffusion models. Then we describe in subsections our contributions and design choices.

Preprocessing of inputs. We first predict human parsing map (S_p, S_g) and 2D pose keypoints (J_p, J_g) for both person and garment images using off-the-shelf methods [50, 129]. For garment image, we further segment out the garment I_c using the parsing map. For person image, we generate clothing-agnostic RGB image I_a which removes the original clothing but retains the person identity. Note that clothing-agnostic RGB described in VITON-HD [31] leaks information of the original garment for challenging human poses and loose garments. We thus adopt a more aggressive way to remove the garment information. Specifically, we first mask out the whole bounding box area of the foreground person, and then copy-paste the head, hands and lower body part on top of it. We use S_p and J_p to extract the non-garment body parts. We also normalize pose keypoints to the range of $[0, 1]$ before inputting them to our networks. Our try-on conditional inputs are denoted as $\mathbf{c}_{\text{tryon}} = (I_a, J_p, I_c, J_g)$.

Brief overview of diffusion models. Diffusion models [165, 73] are a class of genera-

tive models that learn the target distribution through an iterative denoising process. They consist of a Markovian forward process that gradually corrupts the data sample $\mathbf{x}$ into the Gaussian noise $\mathbf{z}_T$, and a learnable reverse process that converts $\mathbf{z}_T$ back to $\mathbf{x}$ iteratively. Diffusion models can be conditioned on various signals such as class labels, texts or images. A conditional diffusion model $\hat{\mathbf{x}}_\theta$ can be trained with a weighted denoising score matching objective:

$$\mathbb{E}_{\mathbf{x},\mathbf{c},\boldsymbol{\epsilon},t}[w_t \|\hat{\mathbf{x}}_\theta(\alpha_t \mathbf{x} + \sigma_t \boldsymbol{\epsilon}, \mathbf{c}) - \mathbf{x}\|_2^2] \tag{3.1}$$

where $\mathbf{x}$ is the target data sample, $\mathbf{c}$ is the conditional input, $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is the noise term. α_t, σ_t, w_t are functions of the timestep t that affect sample quality. In practice, $\hat{\mathbf{x}}_\theta$ is reparameterized as $\hat{\boldsymbol{\epsilon}}_\theta$ to predict the noise that corrupts $\mathbf{x}$ into $\mathbf{z}_t := \alpha_t \mathbf{x} + \sigma_t \boldsymbol{\epsilon}$. At inference time, data samples can be generated from Gaussian noise $\mathbf{z}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ using samplers like DDPM [73] or DDIM [166].

Cascaded Diffusion Models for Try-On. Our cascaded diffusion models consist of one base diffusion models and two super-resolution (SR) diffusion models.

The base diffusion model is parameterized as a 128×128 Parallel-UNet (see Fig. 3.2 bottom). It predicts the 128×128 try-on result I_{tr}^{128}, taking in the try-on conditional inputs $\mathbf{c}_{\text{tryon}}$. Since I_a and I_c can be noisy due to inaccurate human parsing and pose estimations, we apply noise conditioning augmentation [74] to them. Specifically, random Gaussian noise is added to I_a and I_c before any other processing. The levels of noise augmentation are also treated as conditional inputs following [74].

The 128×128→256×256 SR diffusion model is parameterized as a 256×256 Parallel-UNet. It generates the 256×256 try-on result I_{tr}^{256} by conditioning on both the 128×128 try-on result I_{tr}^{128} and the try-on conditional inputs $\mathbf{c}_{\text{tryon}}$ at 256×256 resolution. I_{tr}^{128} is directly downsampled from the ground-truth during training. At test time, it is set to the prediction from the base diffusion model. Noise conditioning augmentation is applied to all conditional input images at this stage, including I_{tr}^{128}, I_a and I_c.

The 256×256→1024×1024 SR diffusion model is parameterized as Efficient-UNet intro-

duced by Imagen [157]. This stage is a pure super-resolution model, with no try-on conditioning. For training, random 256×256 crops, from 1024×1024, serve as the ground-truth, and the input is set to 64×64 images downsampled from the crops. During inference, the model takes as input 256×256 try-on result from previous Parallel-UNet model and synthesizes the final try-on result I_{tr} at 1024×1024 resolution. To facilitate this setting, we make the network fully convolutional by removing all attention layers. Like the two previous models, noise conditioning augmentation is applied to the conditional input image.

3.3 Parallel-UNet

The 128×128 Parallel-UNet can be represented as

$$\boldsymbol{\epsilon}_t = \boldsymbol{\epsilon}_\theta(\mathbf{z}_t, t, \mathbf{c}_{\text{tryon}}, \mathbf{t}_{\text{na}}) \tag{3.2}$$

where t is the diffusion timestep, $\mathbf{z}_t$ is the noisy image corrupted from the ground-truth at timestep t, $\mathbf{c}_{\text{tryon}}$ is the try-on conditional inputs, $\mathbf{t}_{\text{na}}$ is the set of noise augmentation levels for different conditional images, and $\boldsymbol{\epsilon}_t$ is predicted noise that can be used to recover the ground-truth from $\mathbf{z}_t$. The 256×256 Parallel-UNet takes in the try-on result I_{tr}^{128} as input, in addition to the try-on conditional inputs $\mathbf{c}_{\text{tryon}}$ at 256×256 resolution. Next, we describe two key design elements of Parallel-UNet.

Implicit warping. The first question is: how can we implement implicit warping in the neural network? One natural solution is to use a traditional UNet [153, 73] and concatenate the segmented garment I_c and the noisy image $\mathbf{z}_t$ along the channel dimension. However, channel-wise concatenation[158, 156] can not handle complex transformations such as garment warping (see Sec. 3.5). This is because the computational primitives of the traditional UNet are spatial convolutions and spatial self attention, and these primitives have strong pixel-wise structural bias. To solve this challenge, we propose to achieve implicit warping using cross attention mechanism between our streams of information (I_c and $\mathbf{z}_t$). The cross attention is based on the scaled dot-product attention introduced by [174]:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d}})V \tag{3.3}$$

where $Q \in \mathbb{R}^{M \times d}, K \in \mathbb{R}^{N \times d}, V \in \mathbb{R}^{N \times d}$ are stacked vectors of query, key and value, M is the number of query vectors, N is the number of key and value vectors and d is the dimension of the vector. In our case, the query and key-value pairs come from different inputs. Specifically, Q is the flattened features of $\mathbf{z}_t$ and K, V are the flattened features of I_c. The attention map $\frac{QK^T}{\sqrt{d_k}}$ computed through dot-product tells us the similarity between the target person and the source garment, providing a learnable way to represent correspondence for the try-on task. We also make the cross attention multi-head, allowing the model to learn from different representation subspaces.

Combining warp and blend in a single pass. Instead of warping the garment to the target body and then blending with the target person as done by prior works, we combine the two operations into a single pass. As shown in Fig. 3.2, we achieve it via two UNets that handle the garment and the person respectively.

The person-UNet takes the clothing-agnostic RGB I_a and the noisy image $\mathbf{z}_t$ as input. Since I_a and $\mathbf{z}_t$ are pixel-wise aligned, we directly concatenate them along the channel dimension at the beginning of UNet processing.

The garment-UNet takes the segmented garment image I_c as input. The garment features are fused to the target image via cross attentions defined above. To save model parameters, we early stop the garment-UNet after the 32×32 upsampling block, where the final cross attention module in person-UNet is done.

The person and garment poses are necessary for guiding the warp and blend process. They are first fed into the linear layers to compute pose embeddings separately. The pose embeddings are then fused to the person-UNet through the attention mechanism, which is implemented by concatenating pose embeddings to the key-value pairs of each self attention layer [157]. Besides, pose embeddings are reduced along the keypoints dimension using CLIP-style 1D attention pooling [139], and summed with the positional encoding of diffusion timestep t and noise augmentation levels $\mathbf{t}_{\text{na}}$. The resulting 1D embedding is used to modulate features for both UNets using FiLM [41] across all scales.

3.4 Implementation Details

Training and Inference. All three models are trained with batch size 256 for 500K iterations using the Adam optimizer [94]. The learning rate linearly increases from 0 to 10^{-4} for the first 10K iterations and is kept constant afterwards. We follow classifier-free guidance [75] and train our models with conditioning dropout: conditional inputs are set to 0 for 10% of training time. All of our test results are generated with the following schedule: The base diffusion model is sampled for 256 steps using DDPM; The 128×128→256×256 SR diffusion model is sampled for 128 steps using DDPM; The final 256×256→1024×1024 SR diffusion model is sampled for 32 steps using DDIM. The guidance weight is set to 2 for all three stages. During training, levels of noise conditioning augmentation are sampled from uniform distribution $\mathcal{U}([0, 1])$. At inference time, they are set to constant values based on grid search, following [157]. TryOnDiffusion was implemented in JAX [19]. All three diffusion models are trained on 32 TPU-v4 chips for 500K iterations (around 3 days for each diffusion model). After trained, we run the inference of the whole pipeline on 4 TPU-v4 chips with batch size 4, which takes around 18 seconds for one batch.

Parallel-UNet Architecture. Figure 3.3 provides the architecture of 256×256 Parallel-UNet. Compared to the 128×128 version, 256×256 Parallel-UNet makes the following changes: 1) In addition to the try-on conditional inputs $\mathbf{c}_{\text{tryon}}$, the 256×256 Parallel-UNet takes as input the try-on result I_{tr}^{128}, which is first bilinearly upsampled to 256×256, and then concatenated to the noisy image $\mathbf{z}_t$; 2) the self attention and cross attention modules only happen at 16×16 resolution; 3) extra UNet blocks at 256×256 resolution are used; 4) the repeated times of UNet blocks are different as indicated by the Figures.

For both 128×128 and 256×256 Parallel-UNet, normalization layers are parametrized as Group Normalization [184]. The number of group is set to $\min(32, \lfloor \frac{C}{4} \rfloor)$, where C is the number of channels for input features. The non-linear activation is set to swish [42] across the whole model. The residual blocks used in each scale have a main pathway of GroupNorm→swish→conv→GroupNorm→swish→conv. The input to the residual block is

processed by a separate convolution layer and added to the output of the main pathway as the skip connection. The number of feature channels for UNet blocks in 128×128 Parallel-UNet is set to $128, 256, 512, 1024$ for resolution $128, 64, 32, 16$ respectively. The number of feature channels for UNet blocks in 256×256 Parallel-UNet is set to $128, 128, 256, 512, 1024$ for resolution $256, 128, 64, 32, 16$ respectively. The positional encodings of diffusion timstep t and noise augmentation levels $\mathbf{t}_{\text{na}}$ are not shown in the figures for cleaner visualization. They are used for FiLM [41] as described in Sec. 3.3. The 128×128 Parallel-UNet has 1.13B parameters in total while the 256×256 Parallel-UNet has 1.06B parameters.

3.5 Experiments

Datasets. We collect a paired training dataset of 4 Million samples. Each sample consists of two images of the same person wearing the same garment in two different poses. For test, we collect 6K unpaired samples that are never seen during training. Each test sample includes two images of *different* people wearing *different* garments under *different* poses. Both training and test images are cropped and resized to 1024×1024 based on detected 2D human poses. Our dataset includes both men and women captured in different poses, with different body shapes, skin tones, and wearing a wide variety of garments with diverse texture patterns. In addition, we also provide results on the VITON-HD dataset [31].

Test datasets	Ours		VITON-HD	
Methods	FID ↓	KID ↓	FID ↓	KID ↓
TryOnGAN [107]	24.577	16.024	30.202	18.586
SDAFN [8]	18.466	10.877	33.511	20.929
HR-VITON [105]	18.705	9.200	30.458	17.257
Ours	**13.447**	**6.964**	**23.352**	**10.838**

Table 3.1: Quantitative comparison to 3 baselines. We compute FID and KID on our 6K test set and VITON-HD's unpaired test set. The KID is scaled by 1000 following [90].

Methods	Random	Challenging
TryOnGAN [107]	1.75%	0.45%
SDAFN [8]	2.42%	2.20%
HR-VITON [105]	2.92%	1.30%
Ours	**92.72%**	**95.80%**
Hard to tell	0.18%	0.25%

Table 3.2: Two user studies. "Random": 2804 random input pairs (out of 6K) were rated by 15 non-experts asked to select the best result or choose "hard to tell". "Challenging": 2K pairs with challenging body poses were selected out of 6K and rated in same fashion. Our method significantly outperforms others in both studies.

	SDAFN [8]	Ours	Hard to tell
Random	5.24%	**77.83%**	16.93%
Challenging	3.96%	**93.99%**	2.05%

Table 3.3: User study comparing SDAFN [8] to our method at 256×256 resolution.

Comparison with other methods. We compare our approach to three methods: TryOnGAN [107], SDAFN [8] and HR-VITON [105]. For fair comparison, we re-train all three methods on our 4 Million samples until convergence. Without re-training, the results of these methods are worse. Released checkpoints of SDAFN and HR-VITON also require layflat garment as input, which is not applicable to our setting. The resolutions of the related methods vary, and we present each method's results in their native resolution: SDAFN's at 256×256, TryOnGAN's at 512×512 and HR-VITON at 1024×1024.

Quantitative comparison. Table 3.1 provides comparisons with two metrics. Since our test dataset is unpaired, we compute Frechet Inception Distance (FID) [71] and Kernel Inception Distance (KID) [13] as evaluation metrics. We computed those metrics on both

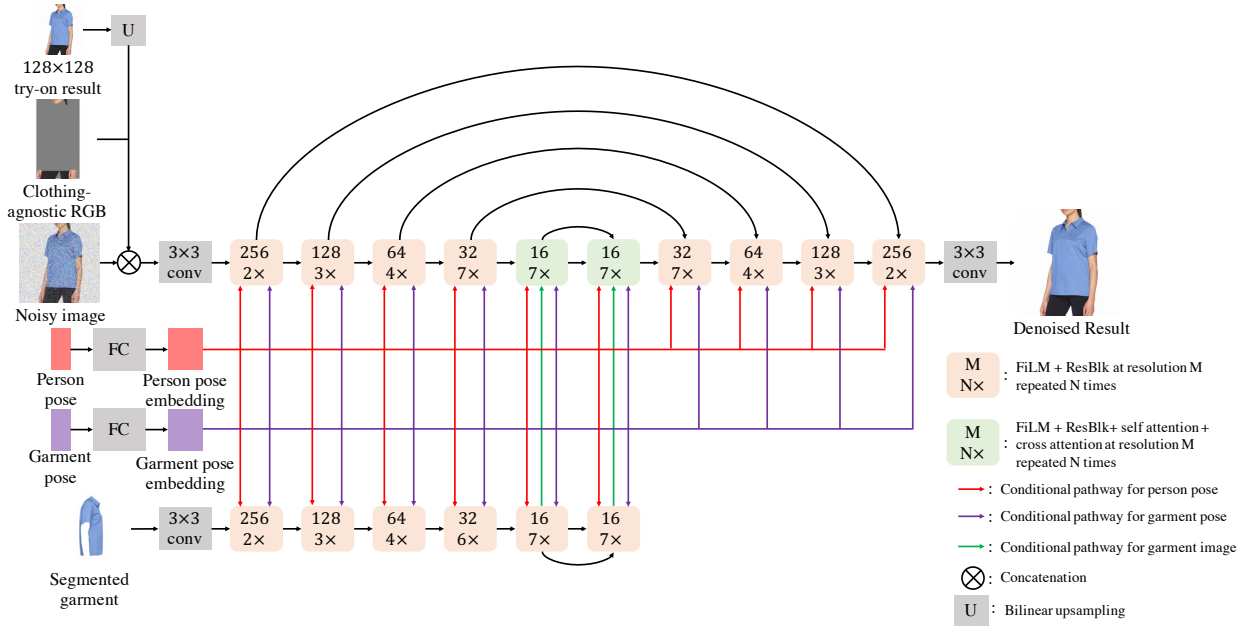

Figure 3.3: Architecture of 256×256 Parallel-UNet.

test datasets (our 6K, and VITON-HD) and observe a significantly better performance with our method.

User study. We ran two user studies to objectively evaluate our methods compared to others at scale. The results are reported in Table 3.2. In first study (named "random"), we randomly selected 2804 input pairs out of the 6K test set, ran all four methods on those pairs, and presented to raters. 15 non-expert raters (on crowdsource platform) have been asked to select the best result out of four or choose "hard to tell" option. Our method was selected as best for 92.72% of the inputs. In a second study (named "challenging"), we performed the same setup but chose 2K input pairs (out of 6K) with more challenging poses. The raters selected our method as best for 95.8% of the inputs. For fair comparison, we also run a new user study to compare SDAFN [8] vs our method at SDAFN's 256×256 resolution. To generate a 256×256 image with our method, we only run inference on the first two stages of our cascaded diffusion models and ignore the 256×256→1024×1024 SR diffusion. Table 3.3

shows results consistent with the user study reported in the paper.

Qualitative comparison. In Figures 3.4, 3.5, and 3.6, we provide visual comparisons to all baselines on our 6K test datasets. In Figures 3.7 and 3.8, we provide visual comparisons to all baselines on VITON-HD test datasets. Note that many of the chosen input pairs have quite different body poses, shapes and complex garment materials–all limitations of most previous methods–thus we don't expect them to perform well but present here to show the strength of our method. Specifically, we observe that TryOnGAN struggles to retain the texture pattern of the garments while SDAFN and HR-VITON introduce warping artifacts in the try-on results. In contrast, our approach preserves fine details of the source garment and seamlessly blends the garment with the person even if the poses are hard or materials are complex (Figure 3.4, row 4). Note also how TryOnDiffusion generates realistic garment wrinkles corresponding to the new body poses (Figure 3.4, row 1). In Figure 3.9 and 3.10, we provide qualitative comparison to state-of-the-art methods on simple cases. We select input pairs from our 6K test dataset with minimum garment warp and simple texture pattern. Baseline methods perform better for simple cases than for challenging cases. However, our method is still better at garment detail preservation and blending (of person and garment). We also compare to HR-VITON [105] using their released checkpoints. Note that original HR-VITON is trained on frontal garment images, so we select input garments satisfying this constraint to avoid unfair comparison. Figure 3.11 shows that our method is still better than HR-VITON under its optimal cases using its released checkpoints.

Ablation 1: Cross attention vs concatenation for implicit warping. The implementation of cross attention is detailed in Sec. 3.3. For concatenation, we discard the garment-UNet, directly concatenate the segmented garment I_c to the noisy image $\mathbf{z}_t$, and drop cross attention modules in the person-UNet. We apply these changes to each Parallel-UNet, and keep the final SR diffusion model same. Figure 3.12 (left) and 3.13 show that cross attention is better at preserving garment details under significant body pose and shape changes. Table 3.4 (row 1 and 3) shows that cross attention is better in terms of FID and KID.

Test datasets	Ours		VITON-HD	
Methods	FID ↓	KID ↓	FID ↓	KID ↓
Ablation 1	15.691	7.956	25.093	12.360
Ablation 2	14.936	7.235	28.330	17.339
Ours	**13.447**	**6.964**	**23.352**	**10.838**

Table 3.4: Quantitative comparison for ablation studies. We compute FID and KID on our 6K test set and VITON-HD's unpaired test set. The KID is scaled by 1000 following [90].

Test datasets	Ours		VITON-HD	
Train set size	FID ↓	KID ↓	FID ↓	KID ↓
10K	16.287	8.975	25.040	11.419
100K	14.667	7.073	23.983	**10.732**
4M	**13.447**	**6.964**	**23.352**	10.838

Table 3.5: Quantitative results for the effects of the training set size. We compute FID and KID on our 6K test set and VITON-HD's unpaired test set. The KID is scaled by 1000 following [90].

Ablation 2: Combining warp and blend vs sequencing two tasks. Our method combines both steps in one network pass as described in Sec. 3.3. For the ablated version, we train two base diffusion models while SR diffusion models are intact. The first base diffusion model handles the warping task. It takes as input the segmented garment I_c, the person pose J_p and the garment pose J_g, and predicts the warped garment I_{wc}. The second base diffusion model performs the blending task, whose inputs are the warped garment I_{wc}, clothing-agnostic RGB I_a, person pose J_p and garment pose J_g. The output is the try-on result I_{tr}^{128} at 128×128 resolution. The conditioning for (I_c, I_a, J_p, J_g) is kept unchanged. I_{wc} in the second base diffusion model is processed by a garment-UNet, which is the same as I_c.

Figure 3.12 (right) and 3.14 visualize the results of both methods. We can see that sequencing warp and blend causes artifacts near the garment boundary, while a single network can blend the target person and the source garment nicely. Table 3.4 (row 2 and 3) further shows that using a single network is better in terms of FID and KID.

Ablation 3: Training Dataset Size. We further investigate the effect of the training dataset size. We retrained our method from scratch on 10K and 100K random pairs from our 4M set and report quantitative results (FID and KID) on two different test sets in Table 3.5. Fig. 3.15 also shows visual results for our models trained on different dataset sizes.

Additional Qualitative Results. Figure 3.16, 3.17 and 3.18 show TryOnDiffusion results on variety of people and garments for both men and women. Figure 3.19 shows try-on results for a challenging case, where input person wearing garment with no folds, and input garment with folds. We can see that our method can generate realistic folds according to the person pose instead of copying folds from the garment input.

Limitations. First, our method exhibits garment leaking artifacts in case of errors in segmentation maps and pose estimations during preprocessing. Fortunately, those [129, 50] became quite accurate in recent years and this does not happen often. Second, representing identity via clothing-agnostic RGB is not ideal, since sometimes it may preserve only part of the identity, e.g., tatooes won't be visible in this representation, or specific muscle structure. Third, our train and test datasets have mostly clean uniform background so it is unknown how the method performs with more complex backgrounds. Finally, this work focused on upper body clothing and we have not experimented with full body try-on, which is left for future work. Figure 3.20 demonstrates failure cases.

3.6 Summary and Future Work

We presented a method that allows to synthesize try-on given an image of a person and an image of a garment. Our results are overwhelmingly better than state-of-the-art, both in the

quality of the warp to new body shapes and poses, and in the preservation of the garment. Our novel architecture Parallel-UNet, where two UNets are trained in parallel and one UNet sends information to the other via cross attentions, turned out to create state-of-the-art results. In addition to the exciting progress for the specific application of virtual try-on, we believe this architecture is going to be impactful for the general case of image editing, which we are excited to explore in the future. Finally, we believe that the architecture could also be extended to videos, which we also plan to pursue in the future.

Figure 3.4: Comparison with TryOnGAN [107], SDAFN [8] and HR-VITON [105]. First column shows the input (person, garment) pairs. TryOnDiffusion warps well garment details including text and geometric patterns even under extreme body pose and shape changes.

Figure 3.5: Comparison with TryOnGAN [107], SDAFN [8] and HR-VITON [105] on challenging cases for women. Compared to baselines, TryOnDiffusion can preserve garment details for heavy occlusions as well as extreme body pose and shape differences. Please zoom in to see details.

Figure 3.6: Comparison with TryOnGAN [107], SDAFN [8] and HR-VITON [105] on challenging cases for men. Compared to baselines, TryOnDiffusion can preserve garment details for heavy occlusions as well as extreme body pose and shape differences. Please zoom in to see details.

Figure 3.7: Comparison with state-of-the-art methods on VITON-HD dataset [31]. All methods were trained on the same 4M dataset and tested on VITON-HD.

Figure 3.8: Comparison with state-of-the-art methods on VITON-HD unpaired testing dataset [31]. All methods were trained on the same 4M dataset and tested on VITON-HD. Please zoom in to see details

Figure 3.9: Comparison with TryOnGAN [107], SDAFN [8] and HR-VITON [105] on simple cases for women. We select input pairs with minimum garment warp and simple texture pattern. Baseline methods perform better for simple cases than for challenging cases. However, our method is still better at garment detail preservation and blending (of person and garment). Please zoom in to see details.

Figure 3.10: Comparison with TryOnGAN [107], SDAFN [8] and HR-VITON [105] on simple cases for men. We select input pairs with minimum garment warp and simple texture pattern. Baseline methods perform better for simple cases than for challenging cases. However, our method is still better at garment detail preservation and blending (of person and garment). Please zoom in to see details.

Figure 3.11: Comparison with HR-VITON released checkpoints for frontal garment (optimal for HR-VITON). Please zoom in to see details.

Figure 3.12: Qualitative results for ablation studies. Left: cross attention versus concatenation for implicit warping. Right: One network versus two networks for warping and blending. Zoom in to see differences highlighted by green boxes.

Figure 3.13: Cross attention vs concatenation for implicit warping. Green boxes highlight differences, please zoom in to see details.

Figure 3.14: Combining warp and blend vs sequencing two tasks. Two networks (column 3) represent sequencing two tasks. One network (column 4) represents combining warp and blend. Green boxes highlight differences, please zoom in to see details.

Figure 3.15: Quanlitative results for effects of the training set size. Please zoom in to see details.

Figure 3.16: TryOnDiffusion on eight target people (columns) dressed by five garments (rows). Zoom in to see details.

Figure 3.17: 4 women trying on 5 garments.

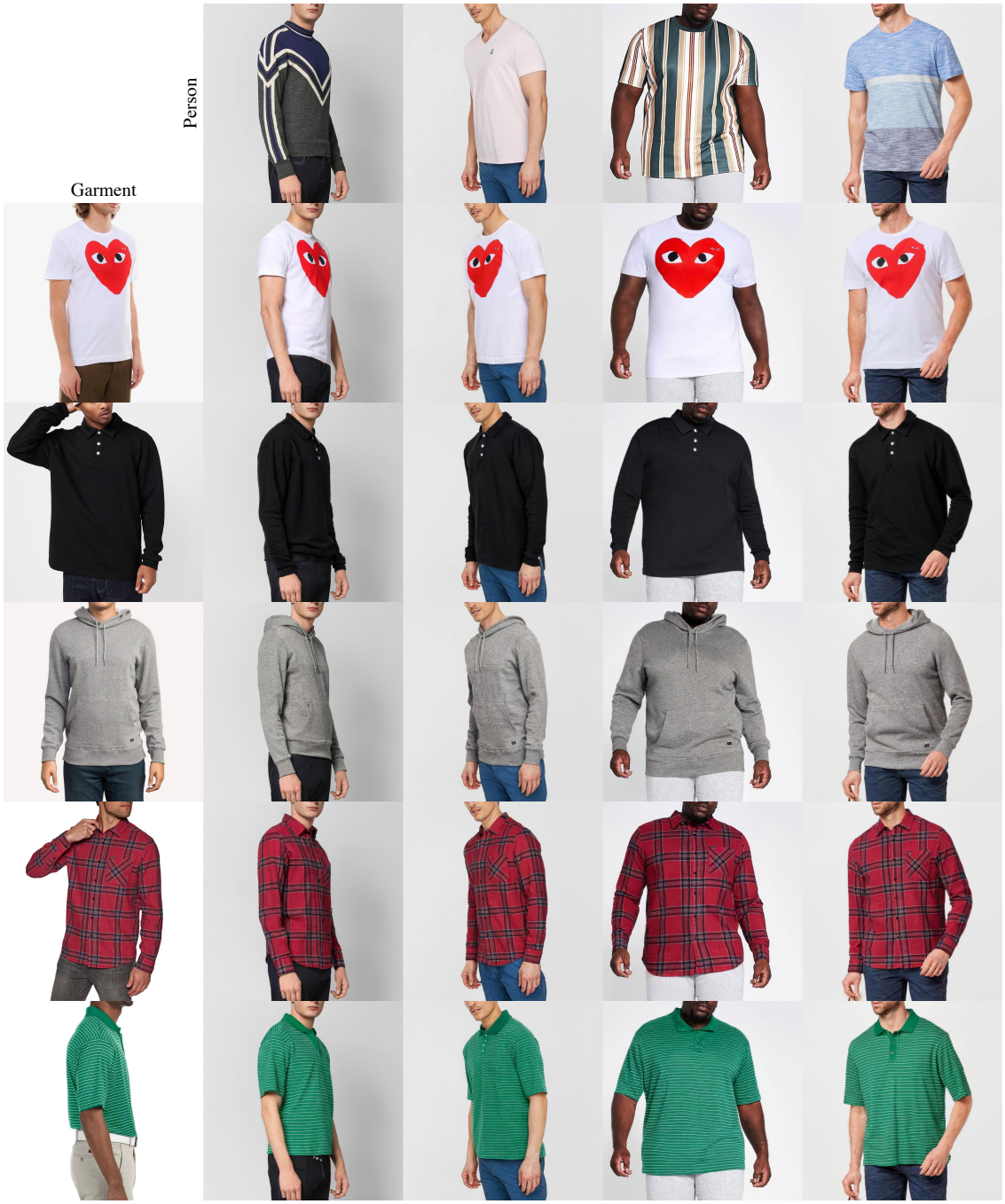

Figure 3.18: 4 men trying on 5 garments.

Figure 3.19: Try-on results for input person wearing garment with no folds, and input garment with folds.

Figure 3.20: Failures happen due to erroneous garment segmentation (left) or garment leaks into the Clothing-agnostic RGB image (right).

Chapter 4

M&M VTO: MULTI-GARMENT VIRTUAL TRY-ON AND EDITING

Figure 4.1: Given an input person image, multiple garments, M&M VTO can output a virtual try-on visualization of how those garments would look on the person. Our model performs well across various body shapes, poses, and garments. In addition, it allows layout to be changed, e.g., "roll up the sleeves" (top rightmost column), and "tuck in the shirt and roll down the sleeves" (bottom rightmost column).

This chapter presents the collaborative research project with Yingwei Li, Nan Liu, Hao Peng, Dawei Yang, Ira Kemelmacher-Shlizerman. The findings from this work will be published in CVPR 2024 [201].

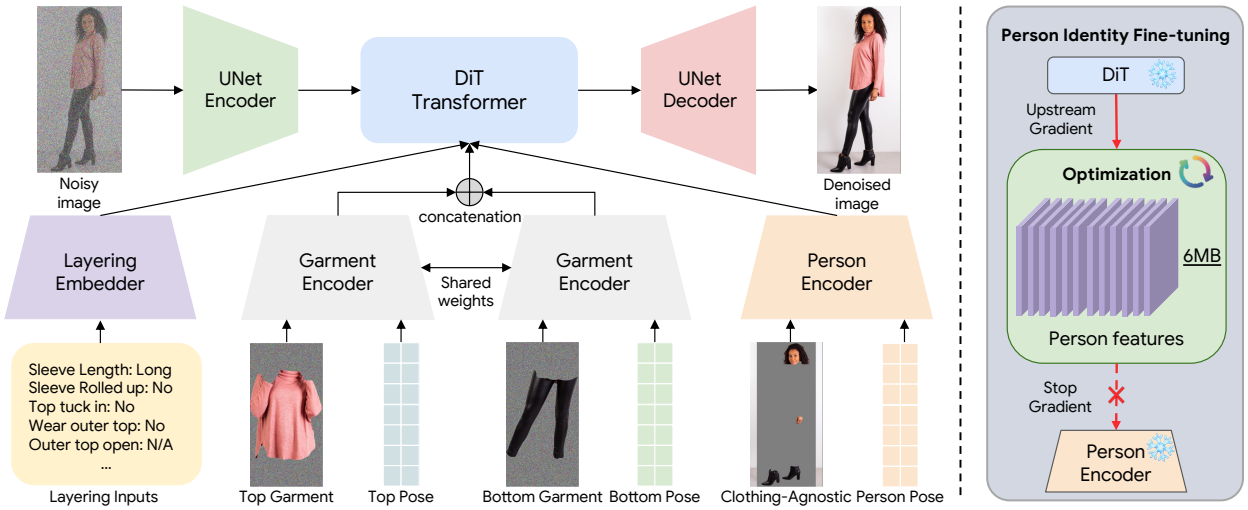

Figure 4.2: **Overview of M&M VTO. Left**: Given multiple garments (top and bottom in this case, full-body garment not shown for this example), layout description, and a person image, our method enables multi-garment virtual try-on. **Right**: By freezing all the parameters, we optimize person feature embeddings extracted from the person encoder to improve person identity for a specific input image. The fine-tuning process recovers the information lost via agnostic computation.

Virtual try-on (VTO) is the task of synthesizing how a person would look in various garments based on provided garment photos and a person photo. Ideally the synthesis is high resolution, showcasing the intricate details of garments, while at the same time representing the body shape, pose, and identity of the person accurately. In this paper, we focus specifically on multiple garment VTO and editing. For example, a user of our method would provide one or more photos for garments, e.g., shirt, pants, and one photo of a person, with additional optional text input to request a layout, e.g., "shirt tucked out, rolled sleeves".

The garment photos could be either a product photo (layflat) or the garment as worn by a different person. The person photo would be a full field-of-view photo showing the person head to toe. Our method, which we named, M&M VTO, outputs a visualization of how the person looks in those garments. Figure 4.1 shows a couple of examples.

Redefining the VTO problem as multiple-garment VTO, rather than the commonly targeted single garment VTO, allowed us to deeply rethink architecture design and solve several open problems in multi, as well as single VTO networks, in addition to opening up the new possibilities for mix and match and editing layouts.

Two of the most challenging VTO problems are (1) how to preserve the small but important details of garments while warping the garment to match various body shapes, and (2) how to preserve the identity of the person without leaking the original garments that the person was wearing to the final result. state-of-the-art methods came close for single garment VTO by leveraging the power of diffusion, and building networks that denoise while warping, e.g., Parallel-Unet [203]. To address (1), however, the network requires to max out the number of parameters and a memory heavy Parallel-UNet to warp a single garment. For (2) a "clothing-agnostic" representation is typically used for the person image to erase the current garment to be replaced by VTO, but at same time it removes a significant amount of identity information, with the network needing to hallucinate the rest, resulting in loss of characteristics like tattoos, body shape or muscle information.

With more garments, as in multi-garment VTO, the number of pixels needed to go through the network triples, so the same number of parameters would create a lower quality VTO. Similarly, showing head to toe person and allowing multiple garments, means 'clothing-agnostic' representation leaves even less of the identity of the person–if just a shirt needs to be replaced, the network can still see how the bottom part of that person looks like (and shape of the legs), while if all garments are changing the agnostic would preserve even less information about the person.

Our solution, M&M VTO, is three-fold as depicted in Figure 4.2. First, we designed a single-stage diffusion model to directly synthesize 1024×512 images with no need for extra

super-resolution(SR) stages as commonly done by state-of-the-art image generation techniques. We found that as we expand the scope of VTO, having cascaded design is detrimental as the base model's low resolution assumes excessive downsampling of ground truth during training, thus losing forever garment details; as SR models depend heavily on the base model, if the details disappear they can not be upsampled effectively. Training a single stage base model just on higher resolution data, however, does not solve the problem, as the model doesn't converge even with ideas proposed in [76, 29]. Instead we designed a progressive training strategy where model training begins with lower-resolution images and gradually moves to higher-resolution ones during the single stage training. Such a design naturally benefits training at higher resolutions by utilizing the prior learned at lower resolutions, allowing the model to better learn and refine high-frequency details.

Second, to solve the identity loss (and/or clothing leakage) during the 'clothing-agnostic' process, we propose a space saving finetuning strategy. Rather than finetuning the entire model during post processing, as commonly done by techniques like DreamBooth [154], we choose to finetune person features only. We designed a VTO UNet Diffusion Transformer (VTO-UDiT) to isolate encoding of person features from the denoising process. In addition to producing much higher quality results, this design also drastically reduces finetuned model size per new individual, going from 4GB to 6MB.

Third, we created text based labels representing various garment layout attributes, e.g., rolled sleeves, tucked in shirt, and open jacket. We formulated attribute extraction as an image captioning task and finetuned a PaLI-3 model [30] using only $1.5k$ labeled images. This allows us to automatically extract accurate labels for the whole training set.

Above three design choices are critical in producing high quality VTO results for multi-garment scenarios. We perform detailed ablation studies, and comparisons to state-of-the-art papers to illustrate each design choice. Our method significantly outperforms others. The user study shows that our method is chosen as best 78.5% of the time compared to state-of-the-art on multiple-garment VTO task.

4.1 Related Work

In this section we will focus on related work relevant to our three key design choices described above. For a comprehensive list of recent papers in virtual try-on we also invite the reader to review this list[1].

Image-Based Virtual Try-On. The seminal VITON method [65] proposed a warping model that estimates pixel displacements between the original garment image and target warp. Based on those displacements, it warped the garment, and then used a blending model to combine the warped garment with the person image, showing one of the first promising results for VTO. Many works followed, to improve pixel displacement estimation. [177] proposed thin plate splines, [195] predicted target segmentation and parsing for improved warping, student-teacher approach and distillation were proposed by [82, 47]. Other efforts include adaptive parsing and second order constraint on thin plate splines [194], optimization to remove misalignments [31], leveraging dance videos to improve warping [40], regularizing [193], and using self and cross attention to improve flow computation [8]. With the rise of StyleGAN, [68] proposed StyleGAN for optical flow, [105] proposed a generator-discriminator approach, [192, 109, 188] reported improved results for flow compute and inpainting by utilization of landmarks, and [28] incorporated size information.

While results were improving, there was an inherent difficulty in warping garments *explicitly–pixel wise*, as there is too much variation in folds, logos, texture where a garment image needs to warp to a new body shape. Rather than estimating flows directly, [107] proposed to interpolate StyleGAN coefficients to create try-on, still lacking complex textures, though, due to the averaging nature of StyleGAN. TryOnDiffusion [203] introduced a diffusion-based [165, 73, 167] Parallel-UNet enabling implicit warping and blending in the same model via cross-attention, showing significantly better results. Key limitations of that approach were incomplete garment details due to base model being only 128×128 resolution, and identity preservation. Finally, most of those methods are focused on single garment

[1] https://github.com/minar09/awesome-virtual-try-on

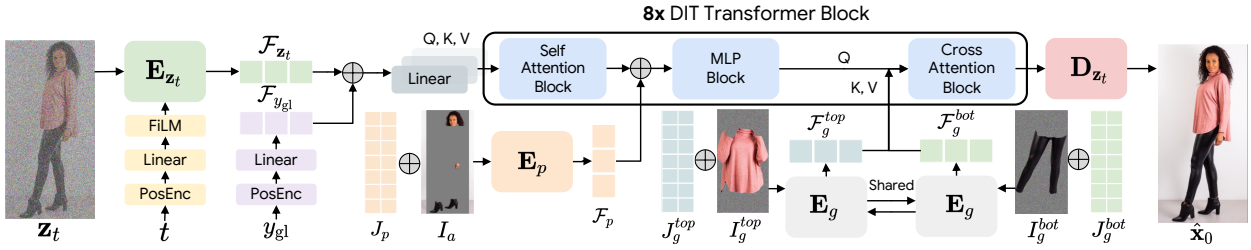

Figure 4.3: **VTO-UDiT architecture.** For image inputs, UNet encoders ($\mathbf{E}_{\mathbf{z}_t}$, $\mathbf{E}_p$, $\mathbf{E}_g$) extract features maps ($\mathcal{F}_{\mathbf{z}_t}$, $\mathcal{F}_p$, $\mathcal{F}_g^\kappa$) from $\mathbf{z}_t$, I_a, I_c^κ, respectively, with $\kappa \in \{\text{upper}, \text{lower}, \text{full}\}$. Diffusion timestep t and garment attributes y_{gl} are embedded with sinusoidal positional encoding, followed by a linear layer. The embeddings ($\mathcal{F}_t$ and $\mathcal{F}_{y_{\text{gl}}}$) are then used to modulate features with FiLM [41] or concatenated to the key-value feature of self-attention in DiT similar to [157]. Following [203], spatially aligned features($\mathcal{F}_{\mathbf{z}_t}$, $\mathcal{F}_p$) are concatenated whereas $\mathcal{F}_g^\kappa$ are implicitly warped with cross-attention blocks. The final denoised image $\hat{\mathbf{x}}_0$ is obtained with decoder $\mathbf{D}_{\mathbf{z}_t}$, which is architecturally symmetrical to $\mathbf{E}_{\mathbf{z}_t}$.

try-on only.

Finetuning Diffusion. As finetuning is a general concept and wasn't much used for VTO, we will review recent works for any general finetuning. Sometimes also called personalization [44], finetuning is the task of adjusting an existing, say text to image generation model, to a specific task, e.g., style transfer. Dreambooth [154] showed fantastic results by finetuning on a few images, and accompanying text, to bind a unique identifier with a specific subject. [45, 108] learned encoders to transfer visual concept into textual embeddings. [1] created a network that maps noise timestamp and layer to text token space. To improve multi-concept composition [112], Custom Diffusion [102] optimized concept embeddings along with key and value projection matrices of cross attention layers in the text-to-image model. In contrast, our approach is tailored for VTO and requires only **6**MB of parameters per person during the inference phase.

Image Editing with Diffusion Models. Editing of general images with diffusion initially

utilized image masks [128, 121, 116, 156, 34, 7]. SDEdit [121] added noise to the inputs and then subsequently denoised them through a stochastic process. Palette [156] trained a conditional diffusion model for specific edit tasks. BlendedDiffusion [7], inspired by CLIP guided diffusion [35], utilized CLIP text encoder [139] and spatial masks to edit images by blending noised input images with locally generated contents. Requiring masks is not applicable to VTO tasks e.g., tuck this shirt in.

The success of text to image diffusion models [128, 141, 72, 151] led to text-based image editing [34, 70, 21, 122, 93, 162, 172, 92, 179]. For example, DiffEdit [34] infers a region mask based on text instructions, and then guides image editing using inverted noise resulted from DDIM inversion process [166]. Prompt-to-Prompt (P2P) [70] edits images using only text by manipulating the cross-attention scores conditioned on inverted latents. Null-text inversion [122] optimized on null-text embeddings by minimizing differences between latent codes from unconditional inversion process and conditional one. InstructPix2Pix [21] directly manipulates image in the denoising process by using finetuned Stable Diffusion trained on paired examples generated using P2P technique with given editing instructions. Generally, text based editing, while allowing for easier input (compared to masks), often creates the edit but fails to preserve original image details, e.g., in VTO case the original garment details are lost with such techniques. We solve it via VTO specific finetuning on PaLI-3 and then using it as condition in the network.

4.2 Generating M&M VTO Data

Given a person image I_p, an upper-body garment image I_g^{upper}, a lower-body garment image I_g^{lower} and a full-body garment image I_g^{full}, our method synthesizes VTO result I_{tr} for person p. Optionally, a layout attribute is provided as input as well. We begin by describing training data and its preprocessing.

M&M VTO is trained on pairs–person image I_p, and a garment image I_g. I_g can be an image of a garment laid out on a flat surface (layflat), or an image of a person wearing the garment (most often in another pose). As the pair assumes that they share only one

or two garments rather than all three of upper, lower and full, we do the following simple process. We compute a garment embedding for each of the three garments (determined by segmentation) and compare which one appears on the person image. The ones that do not are set to 0.

Each pair is then processed following [203]. Conditional inputs $\mathbf{c}_{\text{tryon}}$ includes clothing-agnostic RGB I_a, segmented garment I_c^κ, 2D pose keypoints J_p for the person image I_p and 2D pose keypoints J_g^κ for garment images I_g^κ (J_g^κ is a vector with all -1's if I_g^κ is a layflat garment image). To make sure that background is as tight as possible (allowing for the model to fully focus on garments) we crop and resize all images to 1024×512, approximately resembling aspect ratio of a photograph of a head to toe person.

We also introduce a layout input y_{gl}, defining desired attributes of the garments. We only focus on attributes that one can do in real-life, for example: roll up sleeves, tuck in the shirt, etc. rather than changing texture or garment properties. One way to calculate attributes of each garment is by training a classifier for each attribute. We chose instead to finetune a large vision language model (PaLI-3 [30]). Specifically, we convert all attributes into a formatted text and formulate it as an image captioning task. There are two advantages for this formulation. First, vision language models have strong priors trained on large datasets and can utilize the correlation between different garment layout attributes (e.g. the sleeve can not be rolled up if the sleeve type is sleeveless). Second, using a single model can also accelerate the training data generation process. Thanks to the strong prior encoded in the PaLI-3 model, we are able to get very accurate garment attributes by finetuning PaLI-3 with only 1,500 images. To get y_{gl} for each training sample, we first extract garment layout attributes relevant to the garment type κ by running finetuned PaLI-3 on I_g^κ, and then concatenate those attributes into a single vector.

4.3 Single Stage Diffusion Model for Garment Details

Cascaded diffusion models, i.e., lower resolution diffusion base model, followed by super resolution models, have shown great success for text to synthetic image generation [179, 72].

Similarly, for VTO [203] followed a similar setup where three stages were used. For multi-garment VTO, however, such design is performing poorly, as the base model doesn't have enough capacity to create intricate warps and occlusions based on person's body shape. We observed that high-frequency garment details are smoothed and blurred out if images are downsampled by more than 2 times. Thus, it is impossible for base diffusion models trained to preserve those garment details as their groundtruth images do not include them.

Ideally we would just synthesize 1024×512 images with the base model directly. This turned out to be a challenging task, as if the cross-attention is applied at a lower resolution, the high frequency image details are destroyed by excessive downsampling of feature maps, and the model tends to learn a global structure for the warping. On the other hand, applying cross-attention at a higher resolution does not converge under random initialization from our initial experiments.

To tackle this challenge, we use an effective progressive training paradigm for M&M VTO. The key idea is to initialize the higher resolution diffusion models using a pre-trained lower resolution one. Specifically, we first train a base diffusion model to synthesize 512×256 try-on results $I_{\text{tr}}^{512\times256}$, where the cross-attention happens in 32×16. After that, we continue to train the *exact same model* to synthesize 1024×512 try-on results $I_{\text{tr}}^{1024\times512}$, where the cross-attention happens in 64×32 with the same architecture. Note that our training algorithm does not require modifying or adding new components to the architecture, all we need is to train the model with data in different resolutions, which is easy to implement.

4.4 Efficient Finetuning for Person Identity

A key challenge of current VTO methods is the loss of person identity due to the use of clothing-agnostic representation. To tackle this problem, we propose a space-efficient fine-tuning strategy based on our VTO-UDiT architecture.

4.4.1 VTO-UDiT Architecture.

The VTO-UDiT network (Figure 4.3) is represented as

$$\hat{\mathbf{x}}_0 = \mathbf{x}_\theta(\mathbf{z}_t, t, \mathbf{c}_{\text{tryon}}) \tag{4.1}$$

where t is the diffusion timestep, $\mathbf{z}_t$ is the noisy image corrupted from the ground-truth $\mathbf{x}_0$ at timestep t, $\mathbf{c}_{\text{tryon}}$ is the try-on conditional inputs, and $\hat{\mathbf{x}}_0$ is the predicted clean image at timestep t. In practice, we follow [72] to set the network output in $\mathbf{v}$-space to avoid color drift issues in higher resolution diffusion models. Given the predicted $\hat{\mathbf{v}}_t$, we compute $\hat{\mathbf{x}}_0 = \alpha_t \mathbf{z}_t - \sigma_t \hat{\mathbf{v}}_t$, where $\alpha_t, \sigma_t \in (0, 1)$ control the signal-to-noise ratio.

Inspired by [76], we change the Parallel-UNet architecture [203] into a UDiT architecture where the transformer block is implemented as DiT [134]. With the combination of UNet and DiT, the model benefits from light weight UNet as image encoders and the heavy DiT blocks to process in lower resolution feature maps for attention operations.

Moreover, the design of UDiT fully disentangle the encoding process of $\mathbf{c}_{\text{tryon}}$ from the denoising process, which is critical for person feature finetuning described later in Section 4.4. More specifically, 1) Different UNet encoders are used to process the input images without information exchange. 2) Only $\mathbf{E}_{\mathbf{z}_t}$ takes diffusion timestep t embedding as as input, while $\mathbf{E}_p$ and $\mathbf{E}_g$ do not, to fully disentangle conditional features from diffusion denoising. 3)Unlike Parallel-UNet [203] which updates both conditional features and noisy image features in parallel, VTO-UDiT fixes the conditional features and only updates diffusion features during the forward pass of DiT blocks.

Also, note that all UNet encoders are fully convolutional and free of attention operations, which is preferable for progressive training mentioned in Section 4.3.

Finetuning on Synthetic Data. As described in Sec. 4.4.1, person feature $\mathcal{F}_p$ is independent of diffusion or garment related features, and is kept fixed for DiT blocks where conditioning happens. Thus, we are able to directly finetune the person features instead of the whole diffusion model. This greatly reduces the optimizable weights from 4GB to 6MB.

Furthermore, we found finetuning on person features will not cause the model to overfit on the particular garments worn by the target person as shown in Section 4.6.

The finetuning process needs to learn how to warp garments from varying sizes and poses on the target person, however, acquiring pairs of images of same garment and various shapes and sizes is impractical. Instead we use pretrained M&M VTO to prepare a synthetic dataset. We segment out garments worn by the target person image, and try-on the garment on multiple person images across various poses (e.g. different torso orientations and arm positions) and body shapes (from 2XS to 2XL), resulting in 150 samples. Since our pretrained M&M VTO can accurately preserve but warp garment details to new pose and shape, the quality of the synthetic finetuning data is high, and allows us to reconstruct the person identity when tested on unseen garments.

4.5 Implementation Details

Training and inference. M&M VTO is trained in two stages. For the first stage, the model is trained on 512×256 images for $600K$ iterations. In the second stage, the model is initialized from the pretrained checkpoint of the first stage and trained on 1024×512 images for an additional $200K$ iterations. For both training stages, the batch size is set to 1024, and the learning rate linearly increases from 0 to 10^{-4} in the first $10K$ steps and is kept unchanged afterwards. We parameterize the model output in v-space following [161] while the $L2$ loss is computed in ϵ-space. All conditional inputs are set to 0 in 10% of the training time for classifier-free guidance (CFG) [75]. Test results are generated by sampling M&M VTO for 256 steps using ancestral sampler [73].

Garment attributes. We summarize as follows the full set of attributes used as layout conditioning input y_{gl}.

1. What is the type of the sleeve?

    (a) Not applicable

(b) Sleeveless

(c) Short sleeve

(d) Middle sleeve

(e) Long sleeve

2. Is the sleeve rolled up?

(a) Not applicable

(b) Sleeve type is not long

(c) Yes

(d) No

3. Is the top garment tucked in?

(a) Not applicable

(b) Not wearing top garment

(c) Can not determine

(d) Yes

(e) No

4. Is the person wearing outer top?

(a) Not applicable

(b) Yes

(c) No

5. Is the outer top closed (e.g. zipper up or button on)?

(a) Not applicable

(b) Not wearing outer top

(c) Can not determine

(d) Yes

(e) No

We selected 1,500 images and asked human labelers to answer all questions for each image. After that, we converted question-answer pairs into a formatted text, where different question-answer pairs are separated by semicolon while the question and answer within each pair are separated by colon. The resulting 1,500 image-caption samples were used to finetune PaLI-3 [30] model. Finally, we ran inference of the finetuned model on our train and test data, and converted the formatted text back into class labels.

4.6 Experiments

In this section, we describe datasets, comparisons and ablations.

Datasets. Our model is trained on two types of datasets: 1) "garment paired" dataset of 17 Million samples, where each sample consists of two images of the same garment in two different poses/body shapes, 2) "layflat paired" dataset of 1.8 Million samples, where each sample consists of an image with garment laid out on a flat surface and an image of a person wearing the garment. For testing, we use two sets: 1) we collected 8,300 triplets (top, bottom, person) that are *unseen* during training, 2) we use DressCode [125] just for comparison with other methods that use it.

Comparison of VTO. We compared with three representative state-of-the-art methods: TryOnDiffusion [203], GP-VTON [188], and LaDI-VTON [124]. Other methods don't provide code at the time of submission. Our 8,300 triplets test set was used to compare to TryonDiffusion, and DressCode triplets unpaired test set was used to compare to GP-VTON, LaDI-VTON and TryonDiffusion. As TryOnDiffusion was trained only on tops, and person images, we retrained it on our dataset for upper-body, lower-body, and full-body garments

Test datasets	Ours 8,300		DressCode	
Methods	FID ↓	KID ↓	FID ↓	KID ↓
GP-VTON [188]	N/A	N/A	38.392	33.909
LaDI-VTON [124]	N/A	N/A	19.346	9.305
TryOnDiffusion [203]	19.459	17.617	15.944	5.363
Ours-DressCode	N/A	N/A	18.725	8.250
Ours	**18.145**	**15.227**	**14.019**	**2.772**

Table 4.1: **Quantitative Comparison of FID [71] and KID [13] .** We evaluate on our 8,300 triplets test set and DressCode triplets test set. GP-VTON [188] and LaDI-VTON [124] are trained on layflat garments, thus we report only on DressCode test set. All baselines are run twice sequentially, first for tops then for bottoms try-on (See Section 4.6).

TryOnDiffusion [203]	Ours	Hard to tell
1526	**6512**	262

Table 4.2: **User Study on our** 8,300 **triplets test set.** In the user study, 16 non-experts were asked to either select the best result or opt for "hard to tell."

separately. For GP-VTON and LaDI-VTON, we used officially released checkpoints trained on DressCode. Then we ran inference sequentially first to produce top VTO, and then bottom VTO. To ensure a fair comparison, we also trained our method exclusively on the DressCode dataset. In Figure 4.4, 4.5, 4.6 and 4.7, we showcase qualitative results from our 8,300 triplets test set, comparing them against those generated by TryOnDiffusion [203]. Further qualitative comparisons on the DressCode triplets test set against all baselines are provided in Figure 4.8 and 4.9. These results highlight our method's superior ability to retain garment details and layout. Table 4.1 shows that our method outperform baselines in terms of FID [71] and KID [13] (scaled by 1000 following [90]). We also provide user study in Table 4.2 for our 8,300 triplets test set. In the user study, 16 non-experts were asked to either select the best result or opt for "hard to tell." The findings indicate that users generally prefer M&M VTO over other methods.

Comparison of Editing. We evaluate our approach by comparing with several text-

Methods	US ↑
P2P + NI [122]	0
IP2P [21]	1
Imagen editor [179]	10
DiffEdit [34]	0
SDXL inpainting [135]	4
Ours	**169**
Hard to tell	16

Table 4.3: **User Study for try-on editing.** We conducted user study on 200 images. The users are required to select the best method that can successfully perform the editing task while maintaining the property of input person and garments.

guided image editing methods. Inpainting mask free: Prompt-to-Prompt (P2P) [70] + Null inversion [122] (P2P + NI) and InstructPix2Pix (IP2P) [21] using a target text prompt and an input image that we wish to perform editing on. With inpainting mask: Imagen editor [179], DiffEdit [34] and SDXL inpainting [135]. To automatically obtain masks for these two baselines, we use human pose estimations to mask out belly regions for "tuck in top garment" or "tuck out top garment" or the arm regions for "roll up sleeve" or "roll down sleeve". Figure 4.10, 4.11, 4.12 and 4.13 present qualitative comparisons on different layout editing tasks. These examples demonstrate that our method can interpret garment layout concepts more effectively, allowing for more precise edits of the targeted part without affecting other areas. We furtber conducted a user study with 200 images to compare garment layout editing. The results in Table 4.3 indicate that our method are preferred by users 84.5% of the time, outperforming the baseline methods.

Finetuning Comparison. We chose 4 person images with challenging body shapes or poses for our person finetuning comparison. For each person image, we randomly picked 100 top and bottom garment combinations, then generated try-on results using all baseline

Methods	US ↑
Finetuned full model	19
Finetuned person encoder	20
Ours without finetuning	95
Ours with finetuning	**265**
Hard to tell	1

Table 4.4: **User Study for person finetuning.** We carried out a user study involving 400 images across 4 subjects, where we randomly select 100 top + bottom input garments for each subject. The participants were asked to choose the method that best maintains the identity of the person (including body pose and shape) as well as the details of input garments.

methods as well as our own. We compare to three baselines: non-finetuned model, finetuning the full model and finetuning the person encoder. For the latter two baselines, we have incorporated the class-specific prior preservation loss, as utilized in DreamBooth [154], to prevent overfitting to the clothing worn by the target person. For our approach, we don't apply such regularization technique as we found our method does not suffer from overfitting. Figure 4.14, 4.15, 4.16 and 4.17 provide qualitative results. Without finetuning, the person's arms, legs, or torso may appear unnaturally slim or wide, and certain challenging poses can not be accurately recovered. However, if we finetune the entire model or the person encoder, it tends to overfit to the clothing worn by the target subject. Our finetuning approach successfully retains both the person's identity and the intricate details of the input garments. The user study results, detailed in Table 4.4, show our finetuning method significantly outperforming the baselines.

Ablation for Single Stage Model vs. Cascaded. Our method generates 1024×512 try-on images in a single stage. For the cascaded variant, we trained a 512×256 base diffusion model, followed by a 512×256 → 1024×512 SR diffusion model. Both models share the same

Methods	FID ↓	KID ↓
Cascaded	18.523	**15.218**
From Scratch	21.645	15.781
Ours	**18.145**	15.227

Table 4.5: **Quantitative results for ablation studies.** We report FID and KID on our 8,300 triplets test set.

architecture as our single-stage model, with the distinction that the SR model concatenates the low-resolution image to the noisy image. Table 4.5 (1st and 3rd rows) presents the FID and KID metrics on our 8,300 triplets test set, comparing our single-stage model with the cascaded variant. Additionally, Figure 4.18 offers more qualitative results. While our method does not surpass the cascaded variant in terms of FID and KID scores with significant margin, the qualitative results indicate that it excels at preserving complex garment details, such as texts and logos. This observation aligns with insights from [135, 97], which suggest that FID and KID are more effective at capturing overall visual composition rather than the nuances of fine-grained visual aesthetics.

Ablation for Progressive Training vs. Training from Scratch. We train an identical model from scratch on 1024×512 data, without leveraging any model pretrained in lower resolutions. Table 4.5 (2nd and 3rd rows) reveals that our progressive training strategy yields better results than training from scratch when considering FID and KID scores on our 8,300 triplets test set. Figure 4.19 highlights that our progressively trained model more effectively manages garment warping under significant pose variations, whereas the ablated version struggles with learning implicit garment warping through cross-attention.

	TryOnDiffusion [203]	Ours
SSIM ↑	0.883	**0.908**
LPIPS ↓	0.165	**0.096**

Table 4.6: SSIM and LPIPS scores on our 1,000 paired test data.

Comparison on Paired Test Set. We have also collected $1,000$ paired test set (not seen during training. Each pair has same person wearing the garment but under two poses). Table 4.6 shows that our method achieves better SSIM and LPIPS for the paired data compared to TryOnDiffusion [203]. Figure 4.20 shows qualitative results, where our method can better preserve intricate garment details.

Dress VTO Qualitative Results. Figure 4.21 and 4.22 present try-on results for the dress category (denoted as I_g^{full}). Note that our method is able to synthesize realistic folds and wrinkles in dress, well aligned with the person's pose, while preserving the intricate details of the garment.

4.7 Discussion

Limitations. Firstly, our approach isn't designed for layout editing tasks, such as "Open the outer top." As demonstrated in Figure 4.23 (left), a random shirt is generated by the model, as no specific information is provided from inputs about what should be inpainted in the open area. Secondly, our method struggles with uncommon garment combinations found in the real world, like a long coat paired with skirts. As shown in the right example of Figure 4.23, the model tends to split the long coat in an attempt to show the skirts, because it learned from examples where both garments are typically visible during training. Thirdly, our model faces challenges when dealing with upper-body clothing from different images, e.g. pairing a shirt from one photo with an outer coat from another. This issue mainly stems from the difficulty in finding training pairs where one image clearly shows a shirt without any cover, while another displays the same shirt under an outer layer. As a result, the model struggles to accurately remove the shirt when it's covered by an outer layer during testing. Finally, note that our method visualizes how an item might look on a person, accounting for their body shape, but it doesn't yet include size information nor solves for exact fit.

Conclusion. We present a method that can synthesize multi-garment try-on results given an image of person and images of upper-body, lower-body and full-body garments. Our

novel architecture VTO-UDiT as well as progressive training strategy, enabled better than state-of-the-art results, particularly in preserving fine garment details and person identity. Furthermore, our method allows for explicit control of garment layout via conditioning the model with garment attributes obtained from a finetuned vision-language model.

Figure 4.4: **Qualitative comparison against TryOnDiffusion [203] on our 8,300 triplets test set part one.** Our method can generate better garment details and layouts. Red boxes highlight errors of TryOnDiffusion. Please zoom in to see details.

Figure 4.5: **Qualitative comparison against TryOnDiffusion [203] on our 8,300 triplets test set part two.** Our method can generate better garment details and layouts. Red boxes highlight errors of TryOnDiffusion. Please zoom in to see details.

Figure 4.6: **Qualitative comparison against TryOnDiffusion [203] on our 8,300 triplets test set part three.** Our method can generate better garment details and layouts. Red boxes highlight errors of TryOnDiffusion. Please zoom in to see details.

Figure 4.7: **Qualitative comparison against TryOnDiffusion [203] on our 8,300 triplets test set part four.** Our method can generate better garment details and layouts. Red boxes highlight errors of TryOnDiffusion. Please zoom in to see details.

Figure 4.8: **Qualitative comparison against GP-VTON [188], LaDI-VTON [124] and TryOnDiffusion [203] on DressCode[125] triplets test set part one.** Ours-DressCode represents our method trained only on DressCode dataset. Red boxes highlight errors of baselines. Please zoom in to see details.

Figure 4.9: **Qualitative comparison against GP-VTON [188], LaDI-VTON [124] and TryOnDiffusion [203] on DressCode[125] triplets test set part two.** Ours-DressCode represents our method trained only on DressCode dataset. Red boxes highlight errors of baselines. Please zoom in to see details.

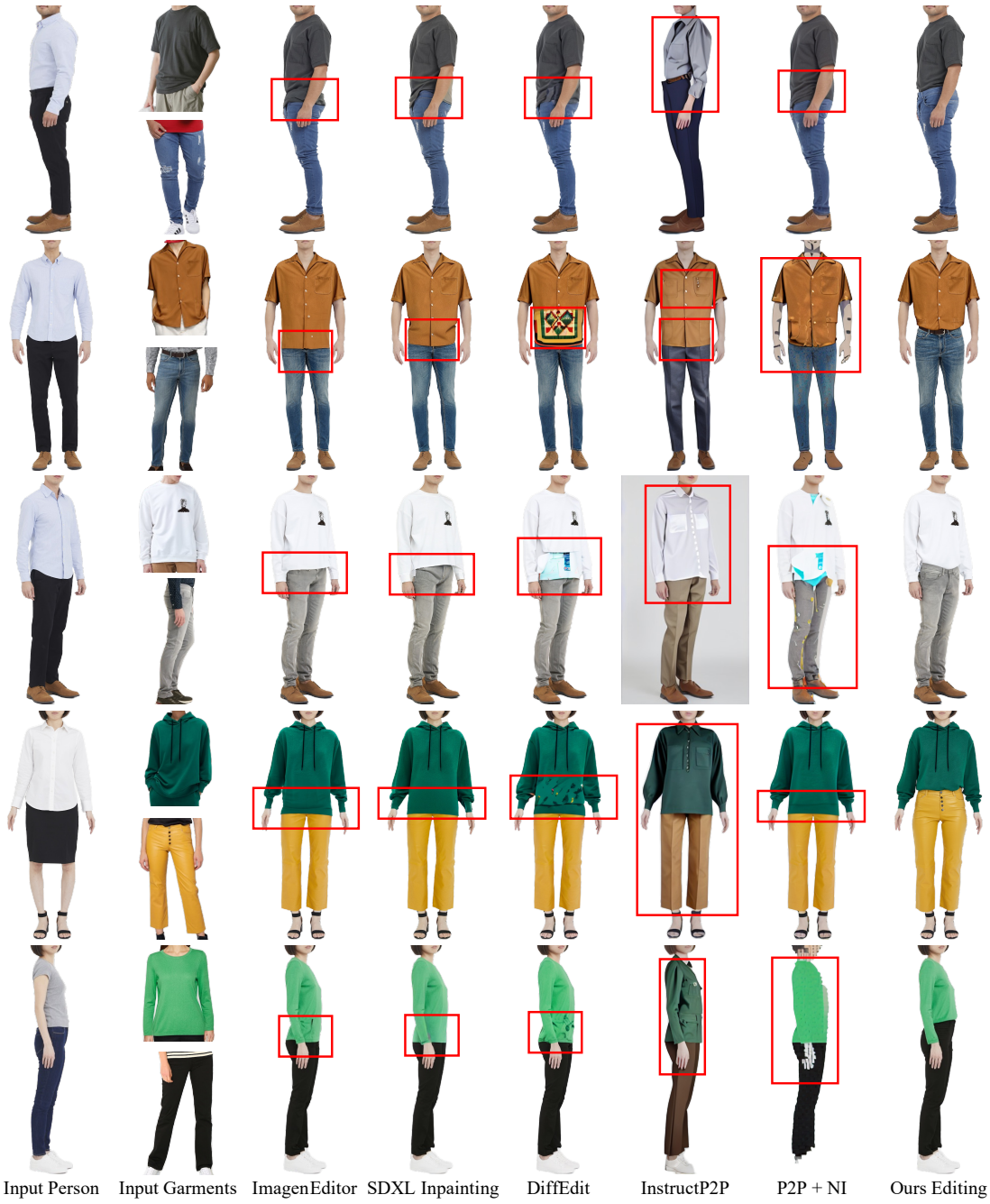

Figure 4.10: **Qualitative comparison for editing instruction: "tuck in the shirt".** Please zoom in to see how our method can perform the desired editing while preserving garment details. Red boxes highlight errors of baselines.

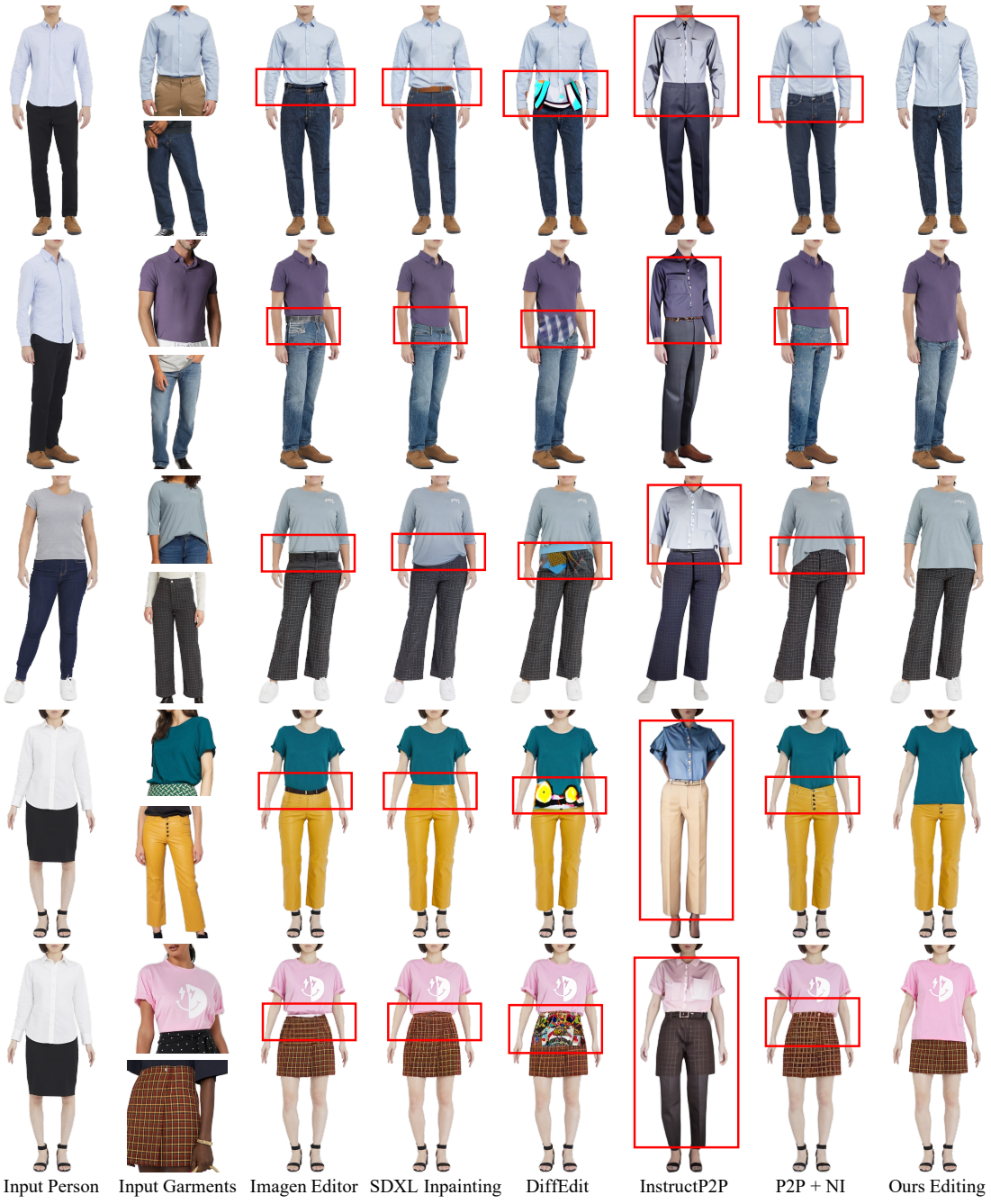

Figure 4.11: **Qualitative comparison for editing instruction: "tuck out the shirt".** Please zoom in to see how our method can perform the desired editing while preserving garment details. Red boxes highlight errors of baselines.

Figure 4.12: **Qualitative comparison for editing instruction: "roll down the sleeve".** Please zoom in to see how our method can perform the desired editing while preserving garment details. Red boxes highlight errors of baselines.

Figure 4.13: **Qualitative comparison for editing instruction: "roll up the sleeve".** Please zoom in to see how our method can perform the desired editing while preserving garment details. Red boxes highlight errors of baselines.

Figure 4.14: **Qualitative comparison for person finetuning of subject 1.** Please zoom in to see how our method can preserve both person identity and garment details. Red boxes highlight errors of baselines.

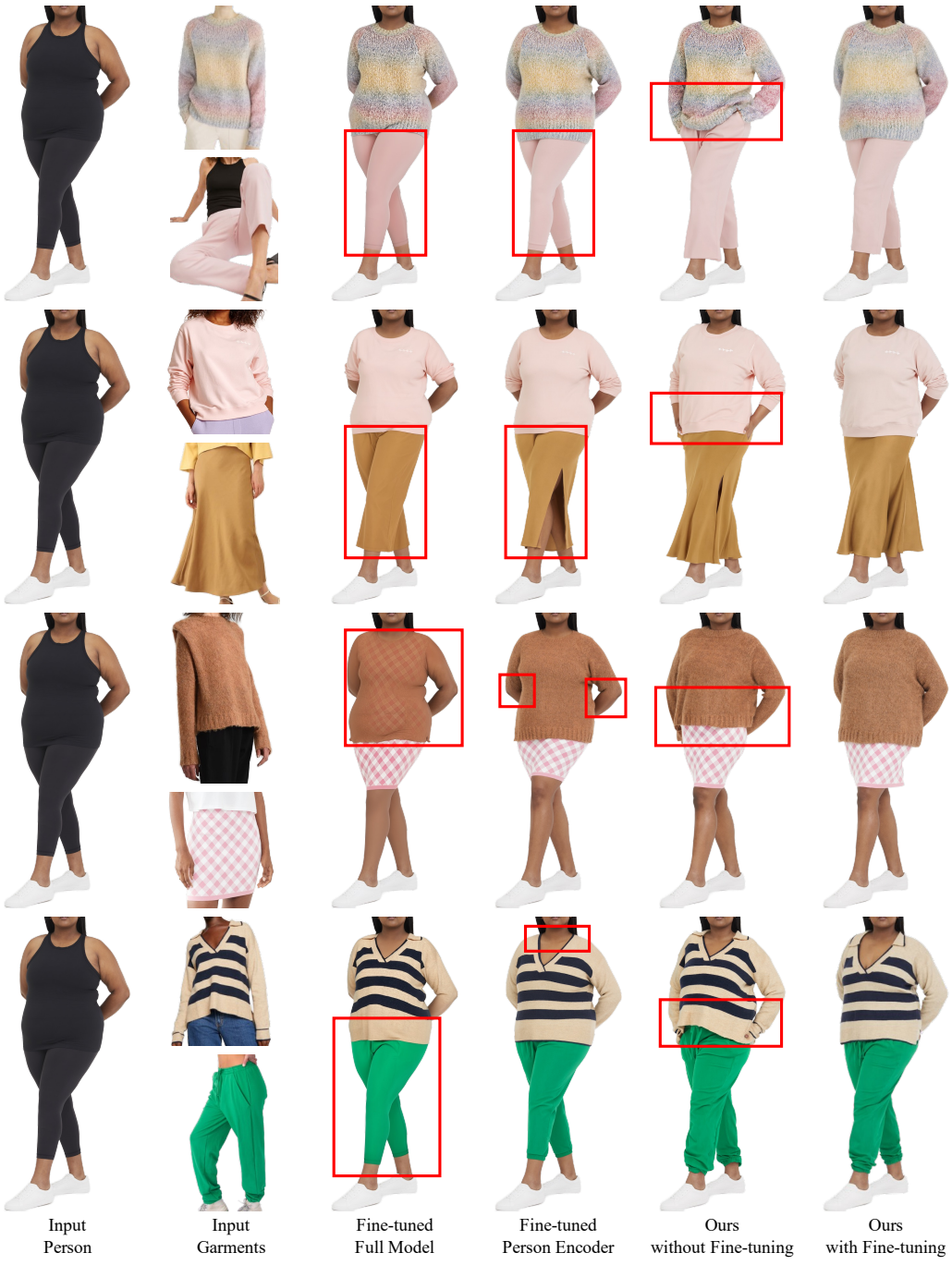

Figure 4.15: **Qualitative comparison for person finetuning of subject 2.** Please zoom in to see how our method can preserve both person identity and garment details. Red boxes highlight errors of baselines.

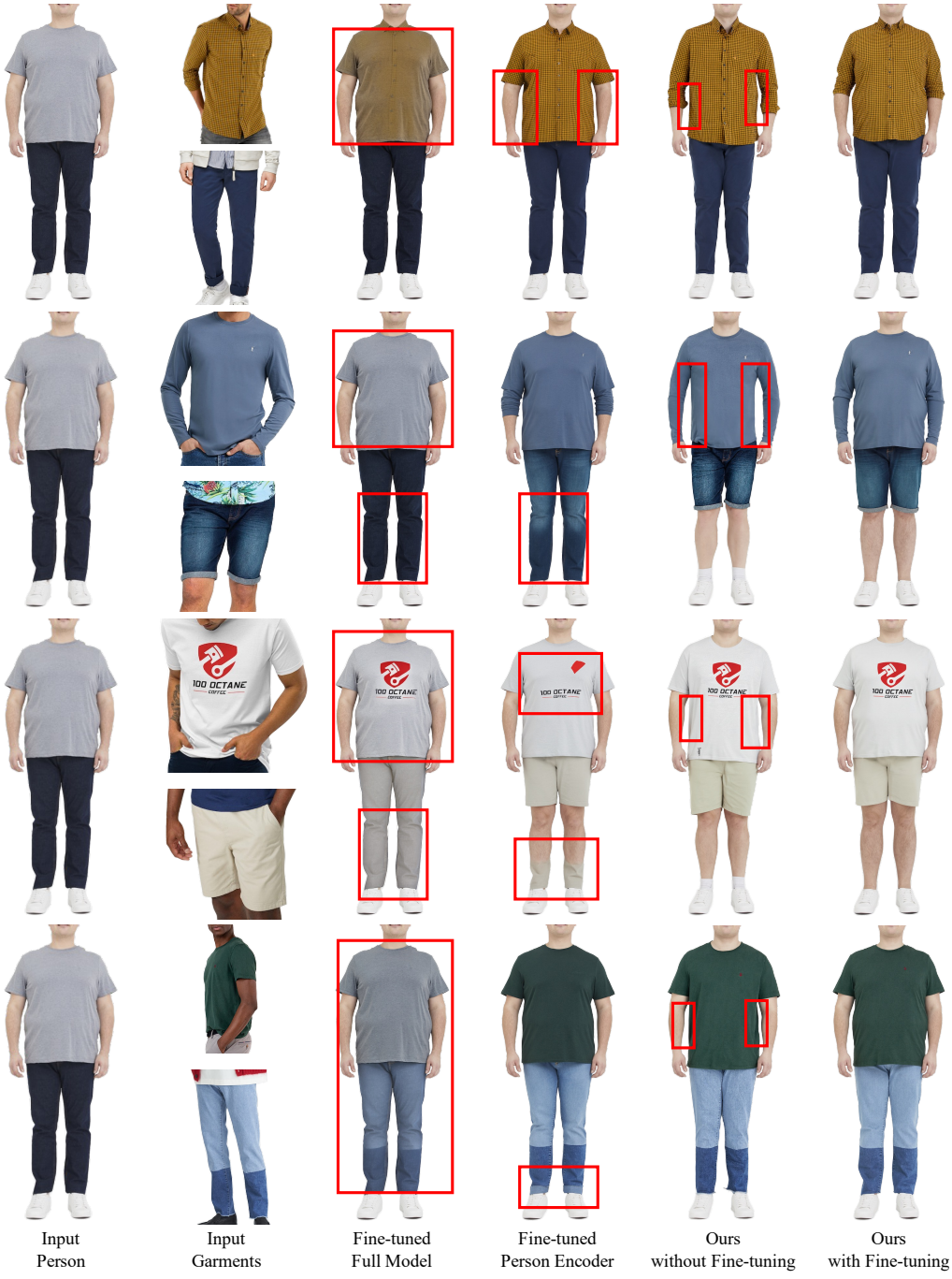

Figure 4.16: **Qualitative comparison for person finetuning of subject 3.** Please zoom in to see how our method can preserve both person identity and garment details. Red boxes highlight errors of baselines.

Figure 4.17: **Qualitative comparison for person finetuning of subject 4.** Please zoom in to see how our method can preserve both person identity and garment details. Red boxes highlight errors of baselines.

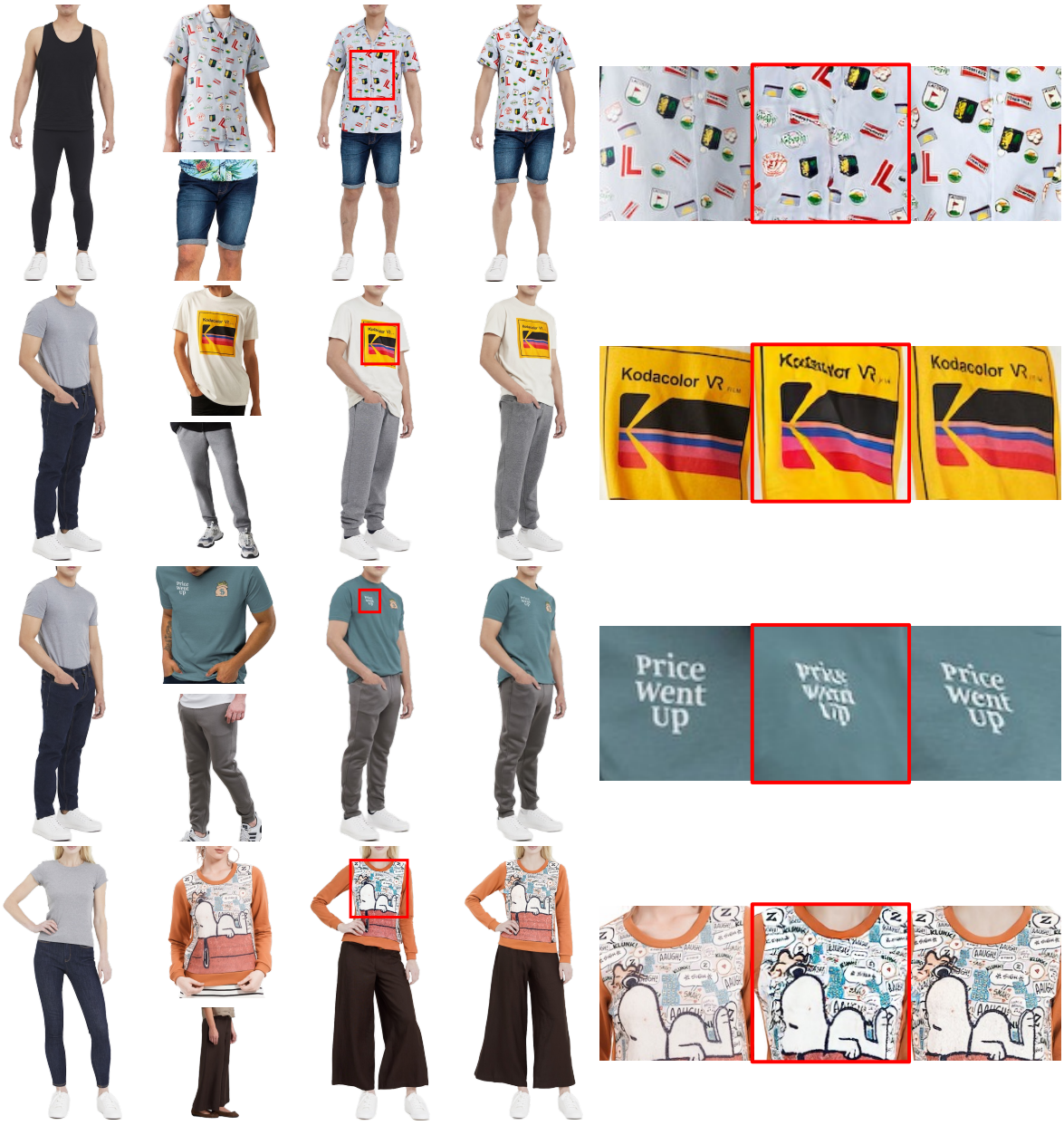

Figure 4.18: **Qualitative comparison for single stage model vs cascaded.** Our proposed single stage model can preserve fine garment details like text and logos under large pose differences. The last three columns visualize zoom-ins of red boxes for input, cascaded variant and single stage model respectively. Please zoom in to see details.

Figure 4.19: **Qualitative comparison for progressive training vs training from scratch.** Training from scratch can not handle complicated garment warping. Red boxes highlight errors of the training from scratch variant. Please zoom in to see details.

Figure 4.20: **Qualitative comparison on our** $1,000$ **paired test data.** Red boxes highlight errors of baselines. Zoom in to see details.

Figure 4.21: **Qualitative results for Dress VTO part one.** Our approach effectively manages complex garment warping and generates realistic wrinkles that align with the person's pose. Please zoom in to see details.

Figure 4.22: **Qualitative results for Dress VTO part two.** Our approach effectively manages complex garment warping and generates realistic wrinkles that align with the person's pose. Please zoom in to see details.

Figure 4.23: **Failure Cases.** Our model could generate random clothing given layout information. As shown in the left example, given "outer top open", the model generates a random inner top. In addition, the model could lead to failures when dealing with rare garment combinations. For example, given a long coat and skirt combination, it creates a half open coat, shown in the right image.

Chapter 5

RECONSTRUCTING NBA PLAYERS

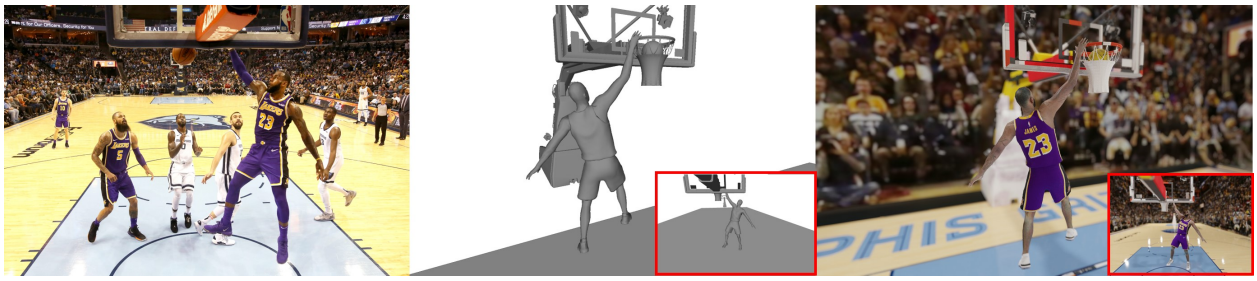

Figure 5.1: Single input photo (left), estimated 3D posed model that is viewed from **a new** camera position (middle), same model with video game texture for visualization purposes. The insets show the estimated shape from the input camera viewpoint. (Court and basketball meshes are extracted from the video game) *Photo Credit: [127]*

This chapter presents the collaborative research project with Konstantinos Rematas, Brian Curless, Steven Seitz, Ira Kemelmacher-Shlizerman. The findings from this work were initially published in ECCV 2020 [202]. The subsequent analysis and comparisons to related studies in this chapter are based on the prevailing state-of-the-art during that time.

Given regular, broadcast video of an NBA basketball game, we seek a complete 3D reconstruction of the players, viewable from any camera viewpoint. This reconstruction problem is challenging for many reasons, including the need to infer hidden and back-facing surfaces, and the complexity of basketball poses, e.g., reconstructing jumps, dunks, and dribbles.

Human body modeling from images has advanced dramatically in recent years, due in large part to availability of 3D human scan datasets, e.g., CAESAR [150]. Based on this

data, researchers have developed powerful tools that enable recreating realistic humans in a wide variety of poses and body shapes [115], and estimating 3D body shape from single images [159, 182]. These models, however, are largely limited to the domains of the source data – people in underwear [150], or clothed models of people in static, staged poses [147]. Adapting this data to a domain such as basketball is extremely challenging, as we must not only match the physique of an NBA player, but also their unique basketball poses.

Sports video games, on the other hand, have become extremely realistic, with renderings that are increasingly difficult to distinguish from reality. The player models in games like NBA2K [33] are meticulously crafted to capture each player's physique and appearance (Figure 5.3). Such models are ideally suited as a training set for 3D reconstruction and visualization of real basketball games.

In this paper, we present a novel dataset and neural networks that reconstruct high quality meshes of basketball players and retarget these meshes to fit frames of real NBA games. Given an image of a player, we are able to reconstruct the action in 3D, and apply new camera effects such as close-ups, replays, and bullet-time effects (Figure 5.1).

Our new dataset is derived from the video game NBA2K (with approval from the creator, Visual Concepts), by playing the game for hours and intercepting rendering instructions to capture thousands of meshes in diverse poses. Each mesh provides detailed shape and texture, down to the level of wrinkles in clothing, and captures all sides of the player, not just those visible to the camera. Since the intercepted meshes are not rigged, we learn a mapping from pose parameters to mesh geometry with a novel *deep skinning* approach. The result of our skinning method is a detailed deep net basketball body model that can be retargeted to any desired player and basketball pose.

We also introduce a system to fit our retargetable player models to real NBA game footage by solving for 3D player pose and camera parameters for each frame. We demonstrate the effectiveness of this approach on synthetic and real NBA input images, and compare with the state-of-the-art in 3D pose and human body model fitting. Our method outperforms the state-of-the-art methods when reconstructing basketball poses and players even when these

methods, to the extent possible, are retrained on our new dataset. This paper focuses on basketball shape estimation, and leaves texture estimation as future work.

Our biggest contributions are, first, a deep skinning approach that produces high quality, pose-dependent models of NBA players. A key differentiator is that we leverage *thousands of poses* and capture detailed geometric variations as a function of pose (e.g., folds in clothing), rather than a small number of poses which is the norm for datasets like CAESAR (1-3 poses/person) and modeling methods like SMPL (trained on CAESAR and ∼45 poses/person). While our approach is applicable to any source of registered 3D scan data, we apply it to reconstruct models of NBA players from NBA2K19 game play screen captures. As such, a second key contribution is pose-dependent models of different basketball players, and raw capture data for the research community. Finally, we present a system that fits these player models to images, enabling 3D reconstructions from photos of NBA players in real games. Both our skinning and pose networks are evaluated quantitatively and qualitatively, and outperform the current state-of-the-art.

One might ask, why spend so much effort reconstructing mesh models that already exist (within the game)? NBA2K's rigged models and in-house animation tools are proprietary IP. By reconstructing a posable model from intercepted meshes (eliminating requirement of proprietary animation and simulation tools), we can provide these best-in-the-world models of basketball players to researchers for the first time (with the company's support). These models provide a number of advantages beyond existing body models such as SMPL. In particular, they capture not just static poses, but human body dynamics for running, walking, and many other challenging activities. Furthermore, the plentiful pose-dependent data enables robust reconstruction even in the presence of heavy occlusions. In addition to producing the first high quality reconstructions of basketball from regular photos, our models can facilitate synthetic data collection for ML algorithms. Just as simulation provides a critical source of data for many ML tasks in robotics, self-driving cars, depth estimation, etc., our derived models can generate much more simulated content under any desired conditions (we can render any pose, viewpoint, combination of players, against any background, etc.)

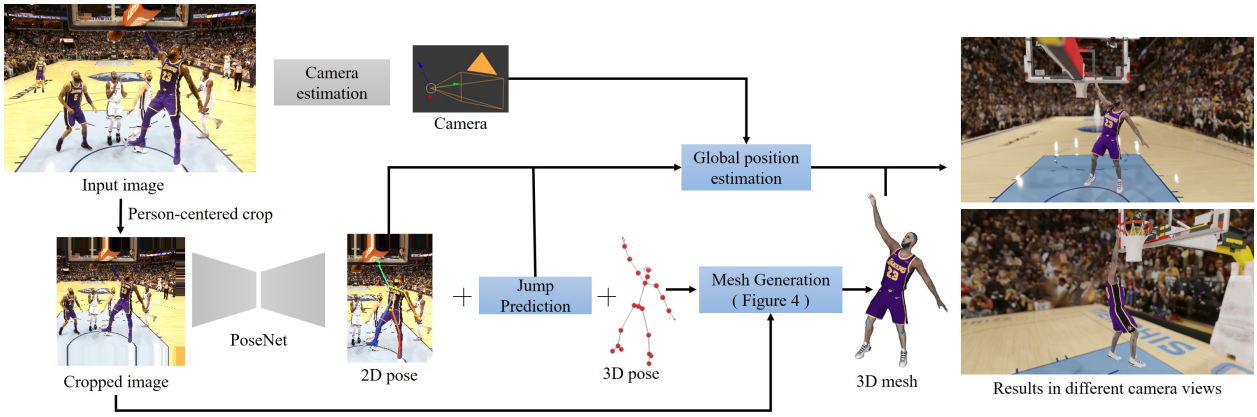

Figure 5.2: Overview: Given a single basketball image (top left), we begin by detecting the target player using [25, 163], and create a person-centered crop (bottom left). From this crop, our PoseNet predicts 2D pose, 3D pose, and jump information. The estimated 3D pose and the cropped image are then passed to mesh generation networks to predict the full, clothed 3D mesh of the target player. Finally, to globally position the player on the 3D court (right), we estimate camera parameters by solving the PnP problem on known court lines and predict global player position by combining camera, 2D pose, and jump information. Blue boxes represent novel components of our method.

5.1 Related Work

Video Game Training Data. Recent works [149, 148, 101, 144] have shown that, for some domains, data derived from video games can significantly reduce manual labor and labeling, since ground-truth labels can be extracted automatically while playing the game. E.g., [24, 144] collected depth maps of soccer players by playing the FIFA soccer video game, showing generalization to images of real games. Those works, however, focused on low level vision data, e.g., optical flow and depth maps rather than full high quality meshes. In contrast, we collect data that includes 3D triangle meshes, texture maps, and detailed 3D body pose, which requires more sophisticated modeling of human body pose and shape.

Sports 3D reconstruction. Reconstructing 3D models of athletes playing various sports from images has been explored in both academic research and industrial products. Most previous methods use multiple camera inputs rather than a single view. Grau *et al.* [54, 53] and Guillemaut *et al.* [60, 59] used multiview stereo methods for free viewpoint navigation. Germann *et al.* [48] proposed an articulated billboard presentation for novel view interpolation. Intel demonstrated 360 degree viewing experiences[1], with their True View [79] technology by installing 38 synchronized 5k cameras around the venue and using this multi-view input to build a volumetric reconstruction of each player. This paper aims to achieve similar reconstruction quality but from a *single* image.

Rematas *et al.* [144] reconstructed soccer games from monocular YouTube videos. However, they predicted only depth maps, thus can not handle occluded body parts and player visualization from all angles. Additionally, they estimated players' global position by assuming all players are standing on the ground, which is not a suitable assumption for basketball, where players are often airborne. The detail of the depth maps is also low. We address all of these challenges by building a basketball specific player reconstruction algorithm that is trained on meshes and accounts for complex airborne basketball poses. Our result is a detailed mesh of the player from a single view, but comparable to multi-view reconstructions. Our reconstructed mesh can be viewed from any camera position.

3D human pose estimation. Large scale body pose estimation datasets [80, 118, 175] enabled great progress in 3D human pose estimation from single images [119, 117, 171, 63, 123]. We build on [119] but train on our new basketball pose data, use a more detailed skeleton (35 joints including fingers and face keypoints), and an explicit model of jumping and camera to predict global position. Accounting for jumping is an important step that allows our method outperform state-of-the-art pose.

3D human body shape reconstruction. Parametric human body models [5, 115, 136, 152, 87, 131] are commonly fit to images to derive a body skeleton, and provide a framework

[1] https://www.intel.com/content/www/us/en/sports/technology/true-view.html

to optimize for shape parameters [16, 87, 131, 185, 103, 77, 196]. [182] further 2D warped the optimized parametric model to approximately account for clothing and create a rigged animated mesh from a single photo. [88, 133, 89, 100, 132, 61, 200, 99] trained a neural network to directly regress body shape parameters from images. Most parametric model based methods reconstruct undressed humans, since clothing is not part of the parametric model.

Clothing can be modeled to some extent by warping SMPL [115] models, e.g., to silhouettes: Weng *et al.* [182] demonstrated 2D warping of depth and normal maps from a single photo silhouette, and Alldeick *et al.* [3, 2, 4] addressed multi-image fitting. Alternatively, given predefined garment models [12] estimated a clothing mesh layer on top of SMPL.

Non-parametric methods [173, 126, 159, 138] proposed voxel [173] or implicit function [159] representations to model clothed humans by training on representative synthetic data. Xu *et al.* [190, 191] and Habermann *et al.* [62] assumed a pre-captured multi-view model of the clothed human, retargeted based on new poses.

We focus on single-view reconstruction of players in NBA basketball games, producing a complete 3D model of the player pose and shape, viewable from any camera viewpoint. This reconstruction problem is challenging for many reasons, including the need to infer hidden and back-facing surfaces, and the complexity of basketball poses, e.g., reconstructing jumps, dunks, and dribbles. Unlike prior methods modeling undressed people in various poses or dressed people in a frontal pose, we focus on modeling clothed people under challenging basketball poses and provide a rigorous comparison with the state-of-the-art.

5.2 The NBA2K Dataset

Imagine having thousands of 3D body scans of NBA players, in every conceivable pose during a basketball game. Suppose that these models were extremely detailed and realistic, down to the level of wrinkles in clothing. Such a dataset would be instrumental for sports reconstruction, visualization, and analysis. This section describes such a dataset, which we call *NBA2K*, after the video game from which these models derive. These models of

Figure 5.3: Our novel NBA2K dataset examples, extracted from the NBA2K19 video game. Our NBA2K dataset captures 27,144 basketball poses spanning 27 subjects, extracted from the NBA2K19 video game.

course are not literally player scans, but are produced by professional modelers for use in the NBA2K19 video game, based on a variety of data including high resolution player photos, scanned models and mocap data of some players. While they do not exactly match each player, they are among the most accurate 3D renditions in existence (Figure 5.3).

Our NBA2K dataset consists of body mesh and texture data for several NBA players, each in around 1000 widely varying poses. For each mesh (vertices, faces and texture) we also provide its 3D pose (35 keypoints including face and hand fingers points) and the corresponding RGB image with its camera parameters. While we used meshes of 27 real famous players to create many of figures in this paper, we do not have permission to release models of current NBA players. Instead, we additionally collected the same kind of data for 28 synthetic players and retrained our pipeline on this data. The synthetic player's have the same geometric and visual quality as the NBA models and their data along with trained models will be shared with the research community upon publication of this paper. Our released meshes, textures, and models will have the same quality as what's in the paper, and

span a similar variety of player types, but not be named individuals. Visual Concepts [33] has approved our collection and sharing of the data.

The data was collected by playing the NBA2K19 game and intercepting calls between the game engine and the graphics card using RenderDoc [**?**]. The program captures all drawing events per frame, where we locate player rendering events by analyzing the hashing code of both vertex and pixel shaders. Next, triangle meshes and textures are extracted by reverse-engineering the compiled code of the vertex shader. The game engine renders players by body parts, so we perform a nearest neighbor clustering to decide which body part belongs to which player. Since the game engine optimizes the mesh for real-time rendering, the extracted meshes have different mesh topologies, making them harder to use in a learning framework. We register the meshes by resampling vertices in texture space based on a template mesh. After registration, the processed mesh has 6036 vertices and 11576 faces with fixed topology across poses and players (point-to-point correspondence), has multiple connected components (not a watertight manifold), and comes with no skinning information. We also extract the rest-pose skeleton and per-bone transformation matrix, from which we can compute forward kinematics to get full 3D pose. One way to decide which frames to capture is to let the game use its AI where two teams play against each other, however we found that the variety of poses captured in this manner is rather limited. It captures mostly walking and running people, while we target more complex basketball moves. Instead, we have people play the game and proactively capture frames where dunk, dribble, shooting, and other complex basketball moves occur.

5.3 From Single Images to Meshes

Figure 5.2 shows our full reconstruction system, starting from a single image of a basketball game, and ending with output of a complete, high quality mesh of the target player with pose and shape matching the image. Next, we describe the individual steps to achieve the final results.

5.3.1 3D Pose in World Coordinates

2D pose, jump, and 3D pose estimation Since our input meshes are not rigged (no skeletal information or blending weights), we propose a neural network called *PoseNet* to estimate the 3D pose and other attributes of a player from a single image. This 3D pose information will be used later to facilitate shape reconstruction. PoseNet takes a single image as input and is trained to output 2D body pose, 3D body pose, a binary jump classification (is the person airborne or not), and the jump height (vertical height of the feet from ground). The two jump-related outputs are key for global position estimation and are our novel addition to existing generic body pose estimation.

From the input image, we first extract ResNet [187] features (from layer 4) and supply them to four separate network branches. The output of the 2D pose branch is a set of 2D heatmaps (one for each 2D keypoint) indicating where the particular keypoint is located. The output of the 3D pose branch is a set of XYZ location maps (one for each keypoint) [119]. The location map indicates the possible 3D location for every pixel. The 2D and 3D pose branches use the same architecture as [187]. The *jump branch* estimates a class label, and the *jump height branch* regresses the height of the jump. Both networks use a fully connected layer followed by two linear residual blocks [117] to get the final output.

The PoseNet model is trained using the following loss:

$$\mathcal{L}_{pose} = \omega_{2d}\mathcal{L}_{2d} + \omega_{3d}\mathcal{L}_{3d} + \omega_{bl}\mathcal{L}_{bl} + \omega_{jht}\mathcal{L}_{jht} + \omega_{jcls}\mathcal{L}_{jcls} \tag{5.1}$$

where $\mathcal{L}_{2d} = \|H - \hat{H}\|_1$ is the loss between predicted (H) and ground truth ($\hat{H}$) heatmaps, $\mathcal{L}_{3d} = \|L - \hat{L}\|_1$ is the loss between predicted (L) and ground truth ($\hat{L}$) 3D location maps, $\mathcal{L}_{bl} = \|B - \hat{B}\|_1$ is the loss between predicted (B) and ground truth ($\hat{B}$) bone lengths to penalize unnatural 3D poses (we pre-computed the ground truth bone length over the training data), $\mathcal{L}_{jht} = \|h - \hat{h}\|_1$ is the loss between predicted (h) and ground truth ($\hat{h}$) jump height, and $\mathcal{L}_{jcls}$ is the cross-entropy loss for the jump class. For all experiments, we set $\omega_{2d} = 10,\ \omega_{3d} = 10,\ \omega_{bl} = 0.5,\ \omega_{jht} = 0.4$, and $\omega_{jcls} = 0.2$.

Global Position To estimate the global position of the player we need the camera parameters of the input image. Since NBA courts have known dimensions, we generate a synthetic 3D field and align it with the input frame. Similar to [144, 26], we use a two-step approach. First, we provide four manual correspondences between the input image and the 3D basketball court to initialize the camera parameters by solving PnP [106]. Then, we perform a line-based camera optimization similar to [144], where the projected lines from the synthetic 3D court should match the lines on the image. Given the camera parameters, we can estimate a player's global position on (or above) the 3D court by the lowest keypoint and the jump height. We cast a ray from the camera center through the image keypoint; the 3D location of that keypoint is where the ray-ground height is equal to the estimated jump height.

5.3.2 Mesh Generation

Reconstruction of a complete detailed 3D mesh (including deformation due to pose, cloth, fingers and face) from a single image is a key technical contribution of our method. To achieve this we introduce two sub-networks (Figure 5.4): *IdentityNet* and *SkinningNet*. IdentityNet takes as input an image of a player whose rest mesh we wish to infer, and outputs the person's rest mesh by deforming a template mesh. The template mesh is the average of all training meshes and is the same starting point for any input. The main benefit of this network is that it allows us to estimate the body size and arm span of the player according to the input image. SkinningNet takes the rest pose personalized mesh and the 3D pose as input, and outputs the posed mesh. To reduce the learning complexity, we pre-segment the mesh into six parts: head, arms, shirt, pants, legs and shoes. We then train a SkinningNet on each part separately. Finally, we combine the six reconstructed parts into one, while removing interpenetration of garments with body parts. Details are described below.

IdentityNet. We propose a variant of 3D-CODED [56] to deform the template mesh. We first use ResNet [67] to extract features from input images. Then we concatenate template mesh vertices with image features and send them into an AtlasNet decoder [57] to predict

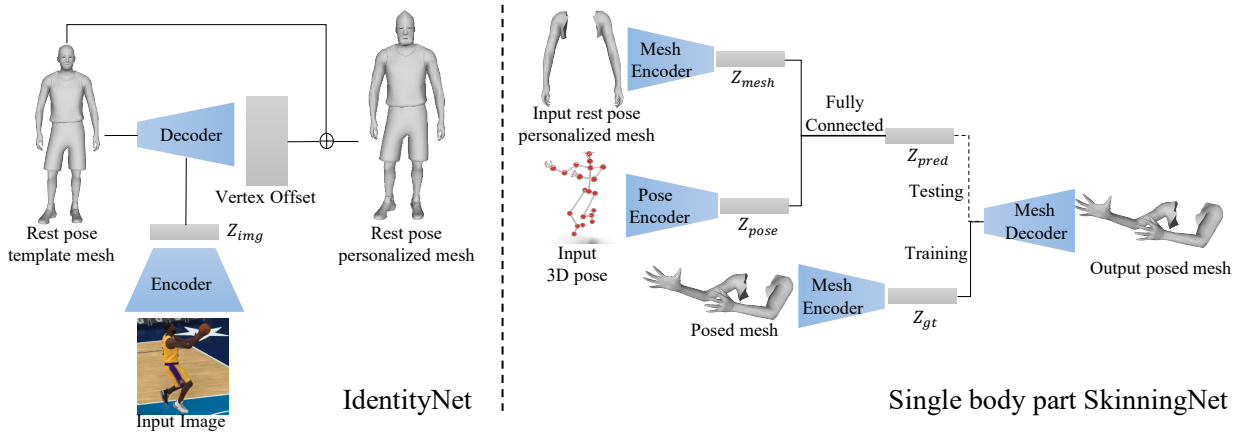

Figure 5.4: Mesh generation contains two sub networks: IdentityNet and SkinningNet. IdentityNet deforms a rest pose template mesh (average rest pose over all players in the database), into a rest pose personalized mesh given the image. SkinningNet takes the rest pose personalized mesh and 3D pose as input and outputs the posed mesh. There is a separate SkinningNet per body part, here we illustrate the arms.

per vertex offsets. Finally, we add this offset to the template mesh to get the predicted personalized mesh. We use the L1 loss between the prediction and ground truth to train IdentityNet.

SkinningNet. We propose a TL-embedding network [49] to learn an embedding space with generative capability. Specifically, the 3D keypoints $K_{pose} \in R^{35\times3}$ are processed by the pose encoder to produce a latent code $Z_{pose} \in R^{32}$. The rest pose personalized mesh vertices $V_{rest} \in R^{N\times3}$(where N is the number of vertices in a mesh part) are processed by the mesh encoder to produce a latent code $Z_{rest} \in R^{32}$. Then Z_{pose} and Z_{rest} are concatenated and fed into a fully connected layer to get $Z_{pred} \in R^{32}$. Similarly, the ground truth posed mesh vertices $V_{posed} \in R^{N\times3}$ are processed by another mesh encoder to produce a latent code $Z_{gt} \in R^{32}$. Z_{gt} is sent into the mesh decoder during training while Z_{pred} is sent into the mesh decoder during testing.

The Pose encoder is comprised of two linear residual blocks [117] followed by a fully connected layer. The mesh encoders and shared decoder are built with spiral convolutions [18]. SkinningNet is trained with the following loss:

$$\mathcal{L}_{skin} = \omega_Z \mathcal{L}_Z + \omega_{mesh} \mathcal{L}_{mesh} \tag{5.2}$$

where $\mathcal{L}_Z = \|Z_{pred} - Z_{gt}\|_1$ forces the space of Z_{pred} and Z_{gt} to be similar, and $\mathcal{L}_{mesh} = \|V_{pred} - V_{posed}\|_1$ is the loss between decoded mesh vertices V_{pred} and ground truth vertices V_{posed}. The weights of different losses are set to $\omega_Z = 5,\ \omega_{mesh} = 50$.

Combining body part meshes. Direct concatenation of body parts results in interpenetration between the garment and the body. Thus, we first detect all body part vertices in collision with clothing as in [131], and then follow [168, 169] to deform the mesh by moving collision vertices inside the garment while preserving local rigidity of the mesh. This detection-deformation process is repeated until there is no collision or the number of iterations is above a threshold (10 in our experiments).

5.4 Implementation Details

In this section, we provide more details of different components in our pipeline.

5.4.1 PoseNet

The input is a single, person-centered image with dimensions 256×256. We extract ResNet [187] features from layer 4 and supply them to four separate network branches (2D pose, 3D pose, jump class, jump height). The 2D and 3D pose branches consist of 3 set of Deconvolution-BatchNorm-ReLu blocks. For the jump class, we use a fully connected layer followed by two linear residual blocks [117] to get the final output and we use the same network architecture for the jump height branch. We estimate both the jump class and the jump height because the jump class can serve as a threshold to reject the inaccurate jump height prediction in the global position estimation.

The 2D pose branch outputs a set of 2D 64×64 heatmaps, one for every keypoint, indicating where a particular keypoint is located. Similarly, the 3D pose branch outputs a set of 2D 64×64 location maps [119], where each location map indicates the possible 3D location for every pixel. Each location map has 3 channels that encode the XYZ position of a keypoint with respect to pelvis. To generate the ground truth heatmaps, we first transform the 2D pose from its original image resolution (256×256) to 64×64 resolution, and then generate a 2D Gaussian map centered at each joint location. For ground truth XYZ location maps, we put the 3D joint location at the position where the heatmap has non-zero value. To obtain the final output, we take the location of the maximum value in every keypoint heatmap to get the 2D pose at 64×64 resolution and use it to sample the 3D pose from the XYZ location maps. After that, the 2d pose is transformed to original 256×256 resolution. The ground truth jump height is directly extracted from the game, and the jump class is set to 1 if the jump height is greater than 0.1m.

5.4.2 Global Position

Since a basketball court with players typically has more occlusions (and curved lines) than a soccer field, we found the traditional line detection method used in [144] fails. To get robust line features, we train a pix2pix [81] network to translate basketball images to court line masks. For the training data, we use synthetic data from NBA2K, where the predefined 3D court lines are projected to image space using the extracted camera parameters. To demonstrate the robustness of our line feature extraction method, we provide the results on synthetic data in Figure 5.5 and real data in Figure 5.6.

After estimating the camera parameters, we place the player mesh in 3D by considering its 2D pose in the image and the jumping height:

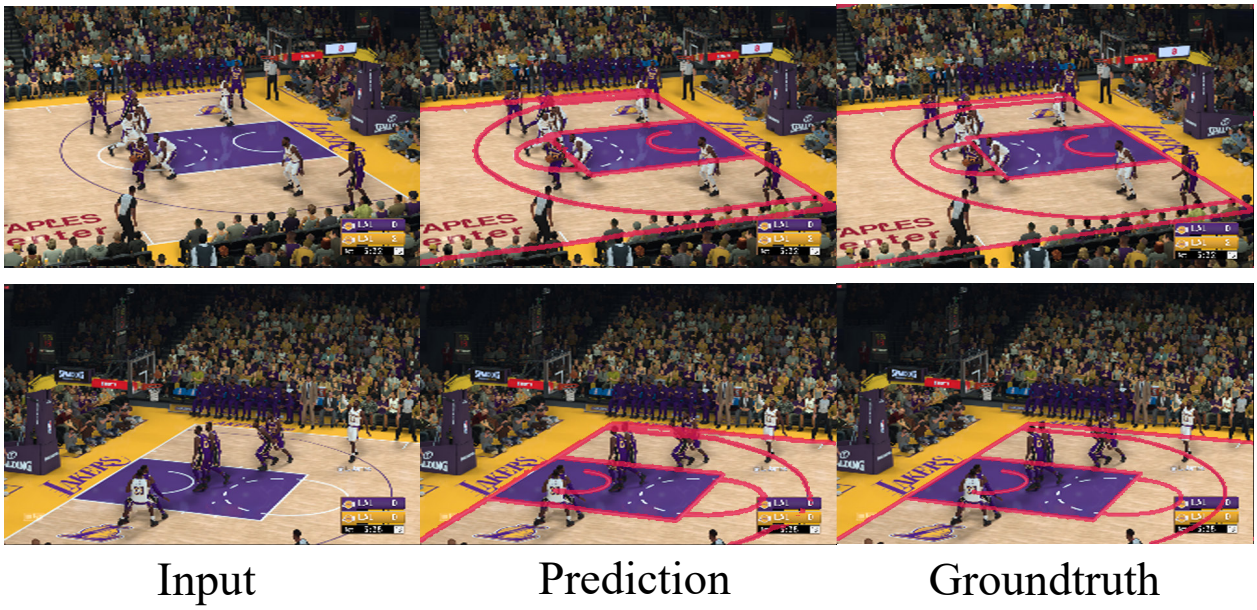

Figure 5.5: **Court line generation on synthetic data.** For every example, from left to right: input image, predicted court lines overlaid on the input image, ground truth court lines overlaid on the input image.

$$V_c = \begin{bmatrix} (x_p - p_x)\frac{z_c}{f} \\ (y_p - p_y)\frac{z_c}{f} \\ z_c \end{bmatrix} \tag{5.3}$$

$$y_w = R_2 \cdot (V_c - T) \tag{5.4}$$

where R_2 is the second column of the extrinsic rotation matrix; T is the extrinsic translation; f is focal length; (p_x, p_y) is the principle point; V_c is the camera coordinates of the lowest joint (e.g. foot); y_w is the world coordinate y-component of the lowest joint, which equals the predicted jump height; (x_p, y_p) are the pixel coordinates of the lowest joints. Substituting Eqn. 5.3 into Eqn. 5.4, we can solve for z_c (camera coordinate in z-component for lowest

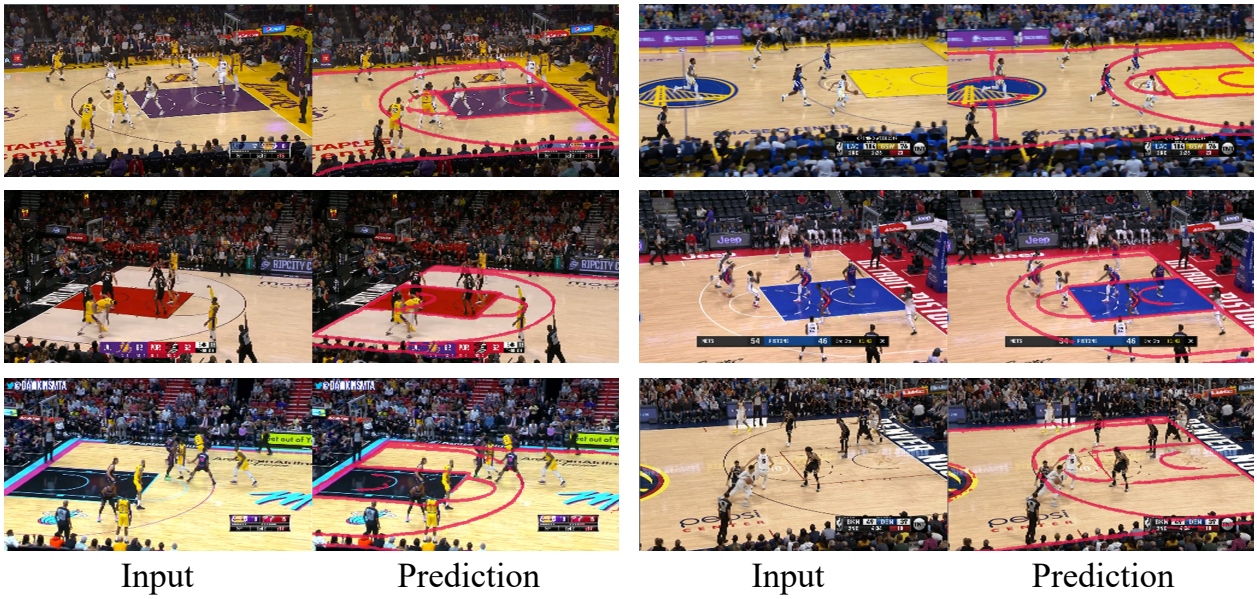

Figure 5.6: **Court line generation on real data.** For every example, left is input image, right is predicted court lines overlaid on the input image.

joints), from which we can further compute the global position of the player. In Figure 5.7, we show our results of global position estimation. We can see that our method can accurately place players (both airborne and on the ground) on the court due to accurate jump estimation.

5.4.3 SkinningNet

The pose encoder is comprised of linear residual block [117] followed by a fully connected layer. The linear residual block consists of four FC-BatchNorm-ReLu-Dropout blocks with skip connection from the input to the output. For the mesh part, we denote Spiral Convolution [18] as SC, mesh downsampling and upsampling operator [6] as DS and US. The mesh encoder consists of four SC-ELU [32]-DS blocks, followed by a FC layer. The mesh decoder consists of a FC layer, four US-SC-ELU blocks, and a SC layer for final processing.

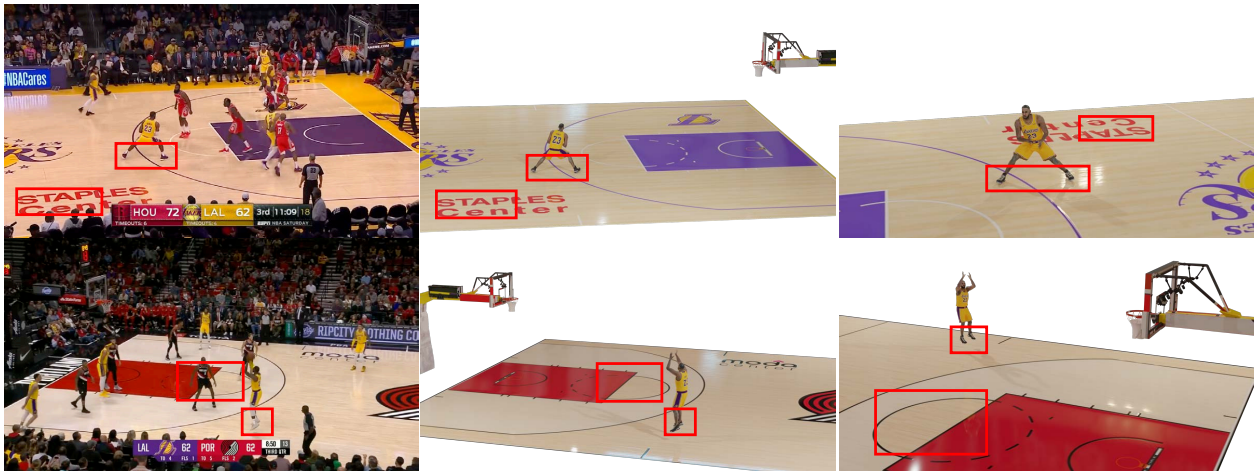

Figure 5.7: **Global position estimation. Please zoom in to see details.** From left to right: input images, two views of the estimated location (middle and right). Note the location of players with respect to court lines (marked with red boxes).

We follow COMA [6] to perform the mesh sampling operation where vertices are removed by minimizing quadric errors [46] during down-sampling and added using barycentric interpolation during up-sampling. In table 5.1, we provide detailed settings for the mesh encoders and decoders of different body parts.

Training details. For training IdentityNet and SkinningNet, we use batch size of 16 for 200 epochs and optimize with the Adam solver [94] with weight decay set to 5×10^{-5}. Learning rate for IdentityNet is 0.0002 while learning rate for SkinningNet is 0.001 with a decay of 0.99 after every epoch. The weights of different losses are set to $\omega_Z = 5, \omega_{mesh} = 50$.

5.4.4 Combining body part meshes

We first detect all the body part vertices in collision with clothing as in [131], and then follow [168, 169] to deform the mesh by moving collision vertices inside the garment while preserving local rigidity of the mesh. This detection-deformation process is repeated until

	head	arm	shoes	shirt	pant	leg
NV	348	842	937	2098	1439	372
DS Factor	(2,2,1,1)	(2,2,2,1)	(2,2,2,1)	(4,2,2,2)	(2,2,2,2)	(2,2,1,1)
NZ	32 for all body parts					
Filter Size	(16,32,64,64) for encoders, (64,32,16,16,3) for decoders					
Dilation	(2,2,1,1) for encoders, (1,1,2,2,2) for decoders					
Step Size	(2,2,1,1) for encoders, (1,1,2,2,2) for decoders					

Table 5.1: **Network architecture for mesh encoders and decoders of different body parts.** NV represents vertices numbers, DS factor represents downsampling factors. NZ represents the hidden size of latent vector. Filter Size represents the output channel of SC. Dilation represents dilation ratio for SC. Step size represents hops for SC.

there is no collision or the number of iterations is above a threshold (10 in our experiments). Before each mesh deformation step, collision vertices are first moved in the direction opposite their vertex normals by 10mm. Then we optimize the remaining vertex positions of body parts by minimizing the following loss:

$$\mathcal{L}_{pen} = \omega_{data}\mathcal{L}_{data} + \omega_{lap}\mathcal{L}_{lap} + \omega_{el}\mathcal{L}_{el} \tag{5.5}$$

$\mathcal{L}_{data} = \|V - V^*\|_2$ forces optimized vertices V to stay close to the SkinningNet inferred vertices $V^* = V(Z_{pred})$, $\mathcal{L}_{lap} = \|\Delta_V - \Delta_{V^*}\|_F$ is the Frobenius norm of Laplacian difference between the optimized and inferred meshes, and $\mathcal{L}_{el} = \|\frac{E}{E^*} - 1\|$ encourages the optimized edge length E to be same as the inferred edge length E^*. Each of these losses is taken as a sum over all vertices or edges. We set $\omega_{data} = 1, \omega_{lap} = 0.1, \omega_{el} = 0.1$ respectively. We use an L-BFGS solver [111], running for 20 iterations. Note that detected collision vertices, after being moved inward, are fixed during the optimization process. This hard constraint ensures the optimization will not move these vertices outside garments in future iterations. Figure 5.8 shows results before and after interpenetration optimization for two examples.

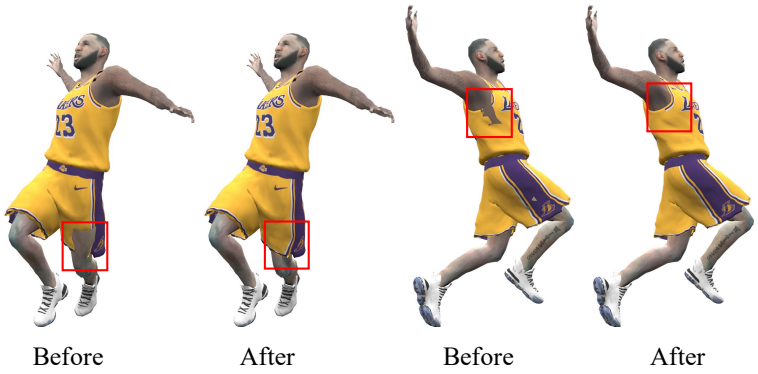

Figure 5.8: **Before and after interpenetration optimization.** Note the garment in the red square. Ground truth textures are used to better visualize the intersection.

	HMR [88]	CMR [100]	SPIN [99]	Ours(Reg+BL)	Ours(Loc)	Ours(Loc+BL)
MPJPE	115.77	82.28	88.72	81.66	66.12	**51.67**
MPJPE-PA	78.17	61.22	59.85	63.70	52.73	**40.91**

Table 5.2: **Quantitative comparison of 3D pose estimation to state-of-the-art.** The metric is mean per joint position error with (MPJPE-PA) and without (MPJPE) Procrustes alignment. Baseline methods are fine-tuned on our NBA2K dataset.

	HMR [88]	SPIN [99]	SMPLify-X [131]	PIFu [159]	Ours
CD	22.411	14.793	47.720	23.136	**4.934**
EMD	0.137	0.125	0.187	0.207	**0.087**

Table 5.3: **Quantitative comparison of our mesh reconstruction to state-of-the-art.** We use Chamfer distance denoted by CD (scaled by 1000, lower is better), and Earth-mover distance denoted by EMD (lower is better) for comparison. Both distance metrics show that our method significantly outperforms state-of-the-art for mesh estimation. All related works are retrained or fine-tuned on our data, see text.

5.5 Experiments

Dataset Preparation. We evaluate our method with respect to the state-of-the-art on our NBA2K dataset. We collected 27,144 meshes spanning 27 subjects performing various basketball poses (about 1000 poses per player). PoseNet training requires generalization on real images. Thus, we augment the data to 265,765 training examples, 37,966 validation examples, and 66,442 testing examples. Augmentation is done by rendering and blending meshes into various random basketball courts. For IdentityNet and SkinningNet, we select 19,667 examples from 20 subjects as training data and test on 7,477 examples from 7 unseen players. To further evaluate generalization of our method, we also provide qualitative results on real images. Note that textures are extracted from the game and not estimated by our algorithm.

5.5.1 3D Pose, Jump, and Global Position Evaluation

We evaluate pose estimation by comparing to state-of-the-art SMPL-based methods that released training code. Specifically we compare with HMR [88], CMR [100], and SPIN [99]. For fair comparison, we fine-tuned their models with 3D and 2D ground-truth NBA2K poses. Since NBA2K and SMPL meshes have different topology we do not use mesh vertices and SMPL parameters as part of the supervision. Table 5.2 shows comparison results for 3D pose. The metric is defined as mean per joint position error (MPJPE) with and without procrustes alignment. The error is computed on 14 joints as defined by the LSP dataset [86]. Our method outperforms all other methods even when they are fine-tuned on our NBA2K dataset (lower number is better).

To further evaluate our design choices, we compare the location-map-based representation (used in our network) with direct regression of 3D joints, and also evaluate the effect of bone length (BL) loss on pose prediction. A direct regression baseline is created by replacing our deconvolution network with fully connected layers [117]. The effectiveness of BL loss is evaluated by running the network with and without it. As shown in Table 5.2, both location

maps and BL loss can boost the performance.

5.5.2 3D Mesh Evaluation

Quantitative Results. Table 5.3 shows results of comparing our mesh reconstruction method to the state-of-the-art on NBA2K data. We compare to both undressed (HMR [88], SMPLify-X [131], SPIN [99]) and clothed (PIFu [159]) human reconstruction methods. For fair comparison, we retrain PIFU on our NBA2K meshes. SPIN and HMR are based on the SMPL model where we do not have groundtruth meshes, so we fine-tuned with NBA2K 2D and 3D pose. SMPLify-X is an optimization method, so we directly apply it to our testing examples. The meshes generated by baseline methods and the NBA2K meshes do not have one-to-one vertex correspondence, thus we use Chamfer (CD) and Earth-mover (EMD) as distance metrics. Prior to distance computations, all predictions are aligned to ground-truth using ICP. We can see that our method outperforms both undressed and clothed human reconstruction methods even when they are trained on our data.

Qualitative Results. Figure 5.9 and 5.10 qualitatively compare our results with the best performing SMPL-based methods SPIN [99] and SMPLify-X [131]. These two methods do not reconstruct clothes, so we focus on the pose accuracy of the body shape. Our method generates more accurate body shape for basketball poses, especially for hands and fingers. Figure 5.11 and 5.12 qualitatively compare with PIFu [159], a state-of-the-art *clothed* human reconstruction method. Our method generates detailed geometry such as shirt wrinkles under different poses while PIFu tends to over-smooth faces, hands, and garments. Figure 5.13 further visualizes garment details in our reconstructions. Figure 5.14 and 5.15 show results of our method on real images, demonstrating robust generalization.

5.5.3 Ablative Study

Comparison with SMPL-NBA. We follow the idea of SMPL [115] to train a skinning model from NBA2K registered mesh sequences. The trained body model is called SMPL-NBA. Since we don't have rest pose meshes for thousands of different subjects, we cannot

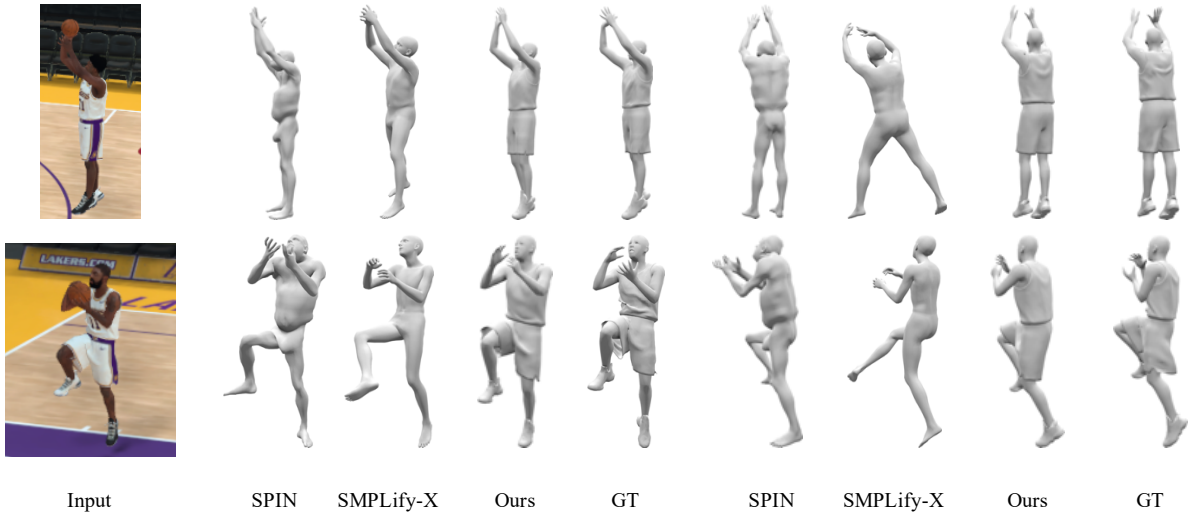

Figure 5.9: **Comparison with SMPL-based methods.** Column 1 is input, columns 2-5 are reconstructions in the image view, columns 6-9 are visualizations from a novel view. Note the significant difference in body pose between ours and SMPL-based methods.

learn a meaningful PCA shape basis as SMPL did. Thus, we focus on the pose dependent part and fit the SMPL-NBA model to 2000 meshes of a single player. We use the same skeleton rig as SMPL to drive the mesh. Since our mesh is comprised of multiple connected parts, we initialize the skinning weights using a voxel-based heat diffusion method [39]. The whole training process of SMPL-NBA is the same as the pose parameter training of SMPL. We fit the learned model to predicted 2D keypoints and 3D keypoints from PoseNet following SMPLify [16]. Figure 5.16 compares SkinningNet with SMPL-NBA, showing that SMPL-NBA has severe artifacts for garment deformation – an inherent difficulty for traditional skinning methods. It also suffers from twisted joints which is a common problem when fitting per bone transformation to 3D and 2D keypoints.

Comparison with Other Geometry Learning Methods. Figure 5.17 compares SkinningNet with two state-of-the-art mesh-based shape deformation networks: 3D-CODED [56] and CMR [100]. The baseline methods are retrained on the same data as SkinningNet for

Figure 5.10: **Comparison with SMPL-based methods on real images.** Column 1 is input, columns 2-4 are reconstructions in the image view, columns 5-7 are visualizations from a novel viewpoint. Note the significant difference in body pose between ours and SMPL-based methods; our results are qualitatively much more similar to what is seen in the input images. In addition, SMPL-based methods do not handle clothing.

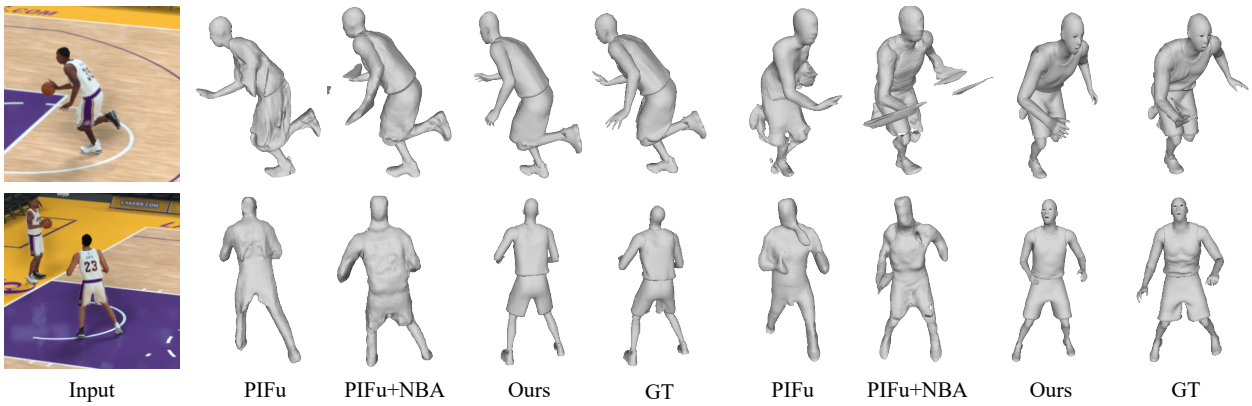

Figure 5.11: **Comparison with PIFu [159].** Column 1 is input, columns 2-5 are reconstructions in the image viewpoint, columns 6-9 are visualizations from a novel view. PIFu significantly over-smooths shape details and produces lower quality reconstruction even when trained on our dataset (PIFu+NBA).

fair comparison. For 3D-CODED, we take 3D pose as input instead of a point cloud to deform the template mesh. For CMR, we only use their mesh regression network (no SMPL regression network) and replace images with 3D pose as input. Both methods use the same 3D pose encoder as SkinningNet. The input template mesh is set to the prediction of IdentityNet. Unlike baseline methods, SkinningNet does not suffer from substantial deformation errors when the target pose is far from the rest pose. Table 5.4 provides further quantitative results based on mean per vertex position error (MPVPE) with and without procrustes alignment.

5.6 Discussion

We have presented a novel system for state-of-the-art, detailed 3D reconstruction of complete basketball player models from single photos. Our method includes 3D pose estimation, jump estimation, an identity network to deform a template mesh to the person in the photo (to

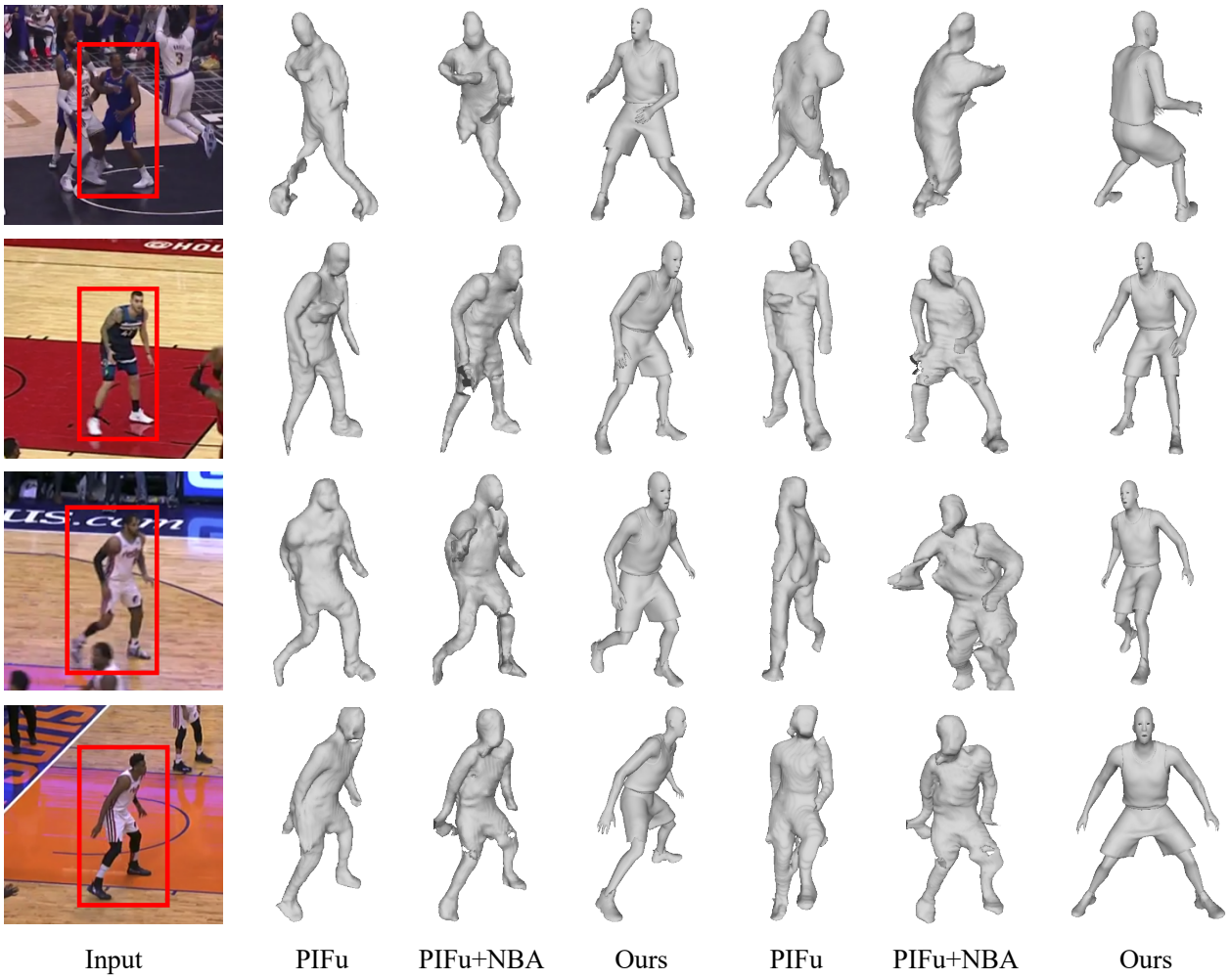

Figure 5.12: **Comparison with PIFu [159] on real images.** Column 1 is input (red box shows the target player), columns 2-4 are reconstructions in the image view, columns 5-7 are reconstructions in a novel view. PIFu fails to reconstruct high quality human shapes from real images, even when the players are in nearly standing poses.

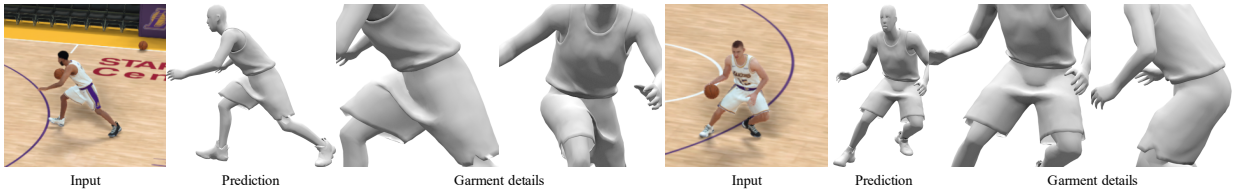

Figure 5.13: **Garment details at various poses.** For each input image, we show the predicted shape, close-ups from two viewpoints.

	CMR [100]	3D-CODED [56]	Ours
MPVPE	85.26	84.22	**76.41**
MPVPE-PA	64.32	63.13	**54.71**

Table 5.4: **Quantitative comparison with 3D-CODED [56] and CMR [100].** The metric is mean per vertex position error in mm with (MPVPE-PA) and without (MPVPE) Procrustes alignment. All baseline methods are trained on the NBA2K data.

estimate rest pose shape), and finally a skinning network that retargets the shape from rest pose to the pose in the photo. We thoroughly evaluated our method compared to prior art; both quantitative and qualitative results demonstrate substantial improvements over the state-of-the-art in pose and shape reconstruction from single images. For fairness, we retrained competing methods to the extent possible on our new data. Our data, models, and code will be released to the research community.

Limitations and future work In Figure 5.18, we provide examples where our approach fails to reconstruct a correct 3D shape from single view images. This paper focuses solely on high quality shape estimation of basketball players, and does not estimate texture – a topic for future work. Additionally IdentityNet can not model hair and facial identity due to lack of details in low resolution input images. Finally, the current system operates on single

Figure 5.14: **Results on real images.** For each example, column 1 is the input image, 2-3 are reconstructions rendered in different views. 4-5 are corresponding renderings using texture from the video game, just for visualization. Our technical method is focused only on shape recovery. *Photo Credit: [78]*

image input only; a future direction is to generalize to video with temporal dynamics.

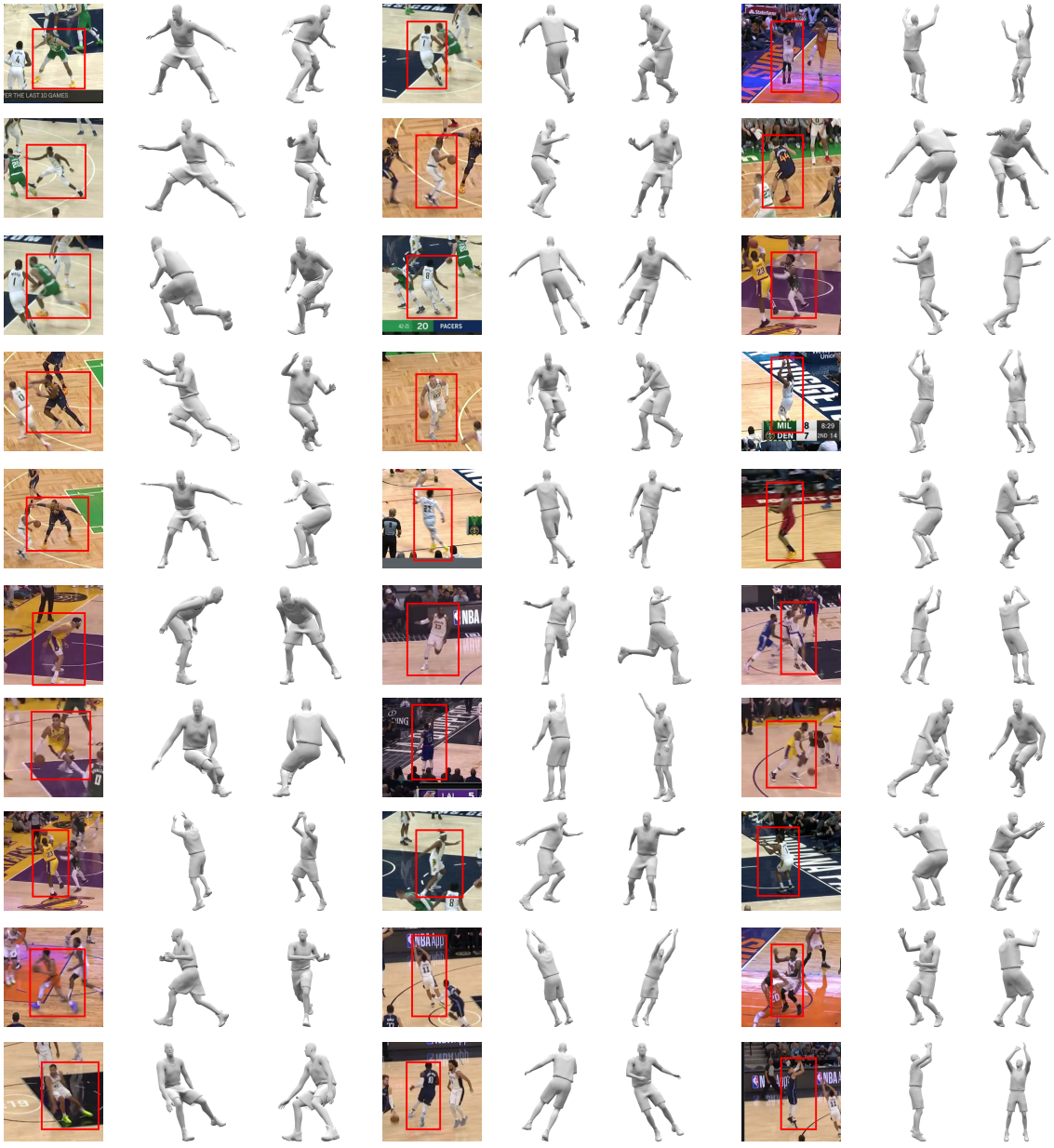

Figure 5.15: **Qualitative Results on real images. Please zoom in to see details.** For every example, left is input (red box shows the target player), middle is reconstruction in the image view, right is reconstruction in a novel view. Our method generalizes well on real images under a variety of poses.

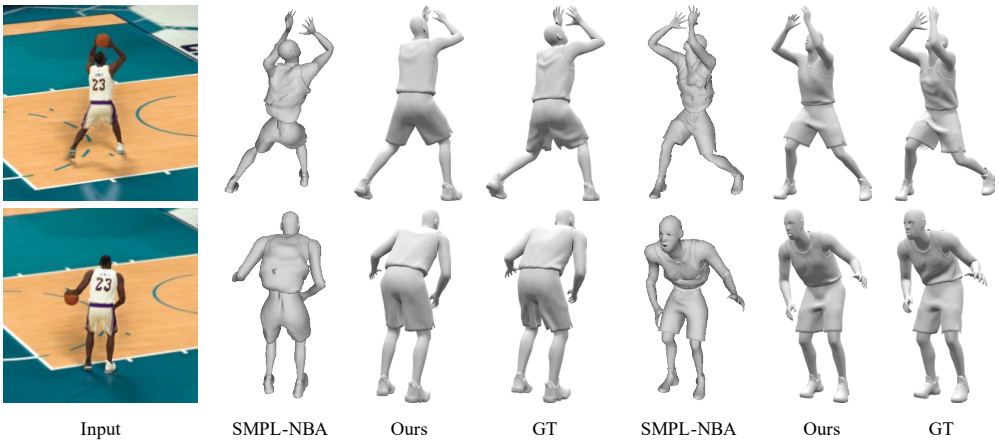

Figure 5.16: **Comparison with SMPL-NBA.** Column 1 is input, columns 2-4 are reconstructions in the image view, columns 5-7 are visualizations from a novel viewpoint. SMPL-NBA fails to model clothing and the fitting process is unstable.

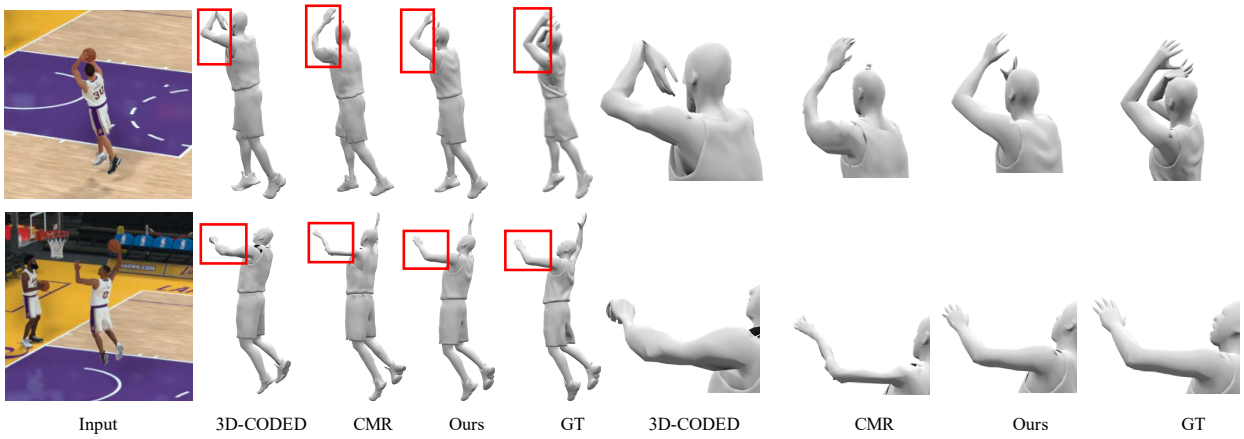

Figure 5.17: **Comparison with 3D-CODED [56] and CMR [100].** Column 1 is input, columns 2-5 are reconstructions in the image view, columns 6-9 are zoomed-in version of the red boxes. The baseline methods exhibit poor deformations for large deviations from the rest pose.

Figure 5.18: **Typical failure cases of our approach.** Column 1 is input (red box shows the target player), columns 2-3 are reconstructions in the image view, columns 4-5 are reconstructions in a novel view, columns 6-7 are zoomed-in versions of main errors. Failures include erroneous pose due to heavy occlusion in multi-person scenes (first and second example), incorrect orientation of head and hands (third and fourth example).

Chapter 6

CONCLUSION

My thesis presented several methods for reconstructing and synthesizing photorealistic humans. I explore diffusion-based image synthesis for applications like VTO, as well as 3D reconstruction of clothed humans, which could aid content creation in augmented or virtual reality environments. Below I summarize each of my contributions.

6.1 Main Contributions

Diffusion-based architecture for garment warping. In Chapter 3, I introduced TryOnDiffusion, the first diffusion model specifically designed for the VTO task. To handle garment warping across large pose differences, I proposed a novel architecture named Parallel-UNet, which employs cross-attention to implicitly warp garments. This approach circumvents the ill-posed problem of explicitly warping the image using dense flow fields, which can be challenging for handling large deformations and heavy occlusions.

Single-stage pixel-space diffusion model for garment detail synthesis. TryOnDiffusion employs cascaded diffusion models in pixel space to generate VTO results. In Chapter 4, I demonstrated that cascaded diffusion models often struggle to accurately synthesize intricate garment details, such as small texts and logos. To address this limitation, I proposed a progressive training strategy for a single-stage diffusion model in the pixel space. This strategy involves initially training the model on lower resolution images to capture mid-level structures, followed by continuing the training on the same model at a higher resolution to focus on capturing high-frequency details.

Space-efficient finetuning strategy for identity preservation. A common challenge in current VTO methods is preserving the identity of the person. In Chapter 4, I introduced

a space-efficient finetuning strategy to address this issue of identity preservation. I proposed a new architecture called VTO-UDiT, which separates the diffusion denoising process from the encoding of person features. Using this architecture, I can finetune the person feature while freezing all other components for a given target subject. This approach significantly reduces the model size needed for finetuning—from 4GB to 6MB—facilitating large-scale deployment.

A large dataset of clothed human meshes from video game. Building a large dataset of 3D clothed humans can be costly and cumbersome, due to the expenses involved in hiring actors, designing motions, and processing raw captures. In Chapter 5, I presented methods for collecting a 3D clothed human dataset from video games, which can be scaled up at a low cost. By intercepting calls between the game engine and the graphics card, I demonstrated how various pieces of information can be extracted from the rendering buffer, including triangle meshes, texture maps, skeletal poses, and camera parameters. I further showed that generative models trained on this synthetic dataset can effectively generalize to real-world images, even those with heavy occlusion.

Deep neural skinning for clothed humans. Previous skinning approaches are trained on a limited number of captures per subject with minimal clothing. Thus, they are not able to model detailed geometric variations of garments as a function of pose—for example, generating clothing folds that conform to specific human poses. In Chapter 5, I proposed to model the skinning function implicitly via a deep neural network, and train this network on thousands of captures per subject. The resulting model is capable of producing high-quality, pose-dependent garments. Additionally, I introduced a system that fits these skinning models to 2D images, enabling 3D reconstructions from real-world photos.

6.2 Future Work

In this section, I explore several promising future directions by addressing a practical and significant question: What are the most desirable features to bring VTO closer to the real

shopping experience?

Video-based virtual try-on. Although current methods can generate impressive try-on images, it still can not compete with the real shopping experience. One important reason is that image-based VTO can only provide a static view of how the garment would look on a person. On contrary, real shopping experience allows customers to see the garment from different angles. Furthermore, customers can see the garment dynamics when they move their body, which is important for loose garments like dresses. Thus, generating try-on videos is necessary to bring the virtual try-on experience to the next level. Given that the number of fashion videos is considerably less than that of fashion images, it's not viable to train a video-based try-on diffusion model from scratch. Instead, we can begin by introducing temporal layers into the pre-trained M&M VTO model and finetuning it with video data. To synthesize high-FPS videos, We can follow a similar progressive training strategy proposed by M&M VTO, but apply it to the temporal dimension instead. Besides real world videos, we can also scale up the dataset using synthetic data collected from the game engine and commercial software. These synthetic data can provide more supervision like normal maps, which can be used as a regularization loss [84] to better capture the movement of wrinkles and folds. By scaling up dataset and model size, we should be able to generate temporally consistent try-on videos capturing garment dynamics.

Virtual try-on anyone. TryOnDiffusion and M&M VTO are trained on data with clean backgrounds, which limits their ability to generalize well to in-the-wild images. The current commercial strategy for VTO involves hiring models of various body sizes and skin tones, capturing their photos in a studio under different poses, and applying VTO algorithms to these images. As a result, consumers can only view VTO results on models who resemble them. To allow consumers to upload their own images and see VTO results directly on themselves, it is essential to collect and curate a more diverse dataset, including in-the-wild images with extreme lighting, noisy backgrounds, selfie poses with occlusion, and camera foreshortening. To collect a high-quality in-the-wild dataset, we also need to train better

off-the-shelf detectors for 2D human pose and human parsing map, and develop more robust pairing algorithms. A promising solution for pairing is to collect hand-labelled positive and negative pairs, and finetune a multi-modal foundation model for this task. Additionally, the finetuning strategy proposed in M&M VTO can be beneficial as a postprocessing step to further enhance identity preservation.

4D human synthesis and editing. Given a video sequence as input, generating and editing free-viewpoint videos of dynamic human is very important for applications like VTO, as it allows consumers to view how they look from different viewpoints under multiple poses. It is also useful for other applications like VR/AR and game production. HumanNeRF [183] can achieve the 4D generation by jointly optimizing the appearance field in the canonical pose and a motion field. However, HumanNeRF is not able to editing the appearance in a temporal and viewpoint consistent way. One possible solution for the 4D editing is to utilize the Instruct-NeRF2NeRF [66] pipeline. Specifically, we can gradually updates the dynamic human reconstruction by iteratively updating training images with edited images given by image-to-image translation models. Another possible solution is to first apply video-based editing methods on the input video sequence, and then train a dynamic human rendering model on the edited video.

BIBLIOGRAPHY

[1] Yuval Alaluf, Elad Richardson, Gal Metzer, and Daniel Cohen-Or. A neural space-time representation for text-to-image personalization, 2023. 54

[2] Thiemo Alldieck, Marcus Magnor, Bharat Lal Bhatnagar, Christian Theobalt, and Gerard Pons-Moll. Learning to reconstruct people in clothing from a single RGB camera. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 92

[3] Thiemo Alldieck, Marcus Magnor, Weipeng Xu, Christian Theobalt, and Gerard Pons-Moll. Video based reconstruction of 3d people models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 92

[4] Thiemo Alldieck, Gerard Pons-Moll, Christian Theobalt, and Marcus Magnor. Tex2shape: Detailed full human body geometry from a single image. In *IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2019. 92

[5] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. Scape: shape completion and animation of people. In *ACM transactions on graphics (TOG)*, pages 408–416. ACM, 2005. 91

[6] Soubhik Sanyal Anurag Ranjan, Timo Bolkart and Michael J. Black. Generating 3D faces using convolutional mesh autoencoders. In *European Conference on Computer Vision (ECCV)*, pages 725–741. Springer International Publishing, 2018. 101, 102

[7] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18208–18218, 2022. 55

[8] Shuai Bai, Huiling Zhou, Zhikang Li, Chang Zhou, and Hongxia Yang. Single stage virtual try-on via deformable attention flows. In *European Conference on Computer Vision*, pages 409–425. Springer, 2022. 4, 8, 10, 21, 27, 28, 29, 34, 35, 36, 39, 40, 53

[9] Alberto Baldrati, Davide Morelli, Giuseppe Cartella, Marcella Cornia, Marco Bertini, and Rita Cucchiara. Multimodal garment designer: Human-centric latent diffusion models for fashion image editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

[10] Ilya Baran and Jovan Popović. Automatic rigging and animation of 3d characters. In *ACM Transactions on graphics (TOG)*, page 72. ACM, 2007.

[11] Serge Belongie, Jitendra Malik, and Jan Puzicha. Shape matching and object recognition using shape contexts. *IEEE transactions on pattern analysis and machine intelligence*, 24(4):509–522, 2002. 9

[12] Bharat Lal Bhatnagar, Garvita Tiwari, Christian Theobalt, and Gerard Pons-Moll. Multi-garment net: Learning to dress 3d people from images. In *IEEE International Conference on Computer Vision (ICCV)*. IEEE, oct 2019. 92

[13] Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans. *arXiv preprint arXiv:1801.01401*, 2018. 28, 62

[14] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 14

[15] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 14

[16] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J. Black. Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In *Computer Vision – ECCV 2016*, Lecture Notes in Computer Science. Springer International Publishing, October 2016. 92, 107

[17] Fred L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Transactions on pattern analysis and machine intelligence*, 11(6):567–585, 1989. 8

[18] Giorgos Bouritsas, Sergiy Bokhnyak, Stylianos Ploumpis, Michael Bronstein, and Stefanos Zafeiriou. Neural 3d morphable models: Spiral convolutional networks for 3d shape representation learning and generation. In *The IEEE International Conference on Computer Vision (ICCV)*, 2019. 98, 101

[19] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018. 26

[20] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018. 13, 21

[21] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2023. 15, 55, 63

[22] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. 2024. 14

[23] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 15

[24] Kiana Calagari, Mohamed Elgharib, Piotr Didyk, Alexander Kaspar, Wojciech Matuisk, and Mohamed Hefeeda. Gradient-based 2-d to 3-d conversion for soccer videos. In *ACM Multimedia*, pages 605–619, 2015. 90

[25] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields. In *arXiv preprint arXiv:1812.08008*, 2018. 90

[26] Peter Carr, Yaser Sheikh, and Iain Matthews. Pointless calibration: Camera parameters from gradient-based alignment to edge images. In *WACV*, 2012. 96

[27] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet: An Information-Rich 3D Model Repository. Technical Report arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015.

[28] 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. 12, 53

[29] Ting Chen. On the importance of noise scheduling for diffusion models. *arXiv preprint arXiv:2301.10972*, 2023. 52

[30] Xi Chen, Xiao Wang, Lucas Beyer, Alexander Kolesnikov, Jialin Wu, Paul Voigtlaender, Basil Mustafa, Sebastian Goodman, Ibrahim Alabdulmohsin, Piotr Padlewski, et al. Pali-3 vision language models: Smaller, faster, stronger. *arXiv preprint arXiv:2310.09199*, 2023. 52, 56, 61

[31] 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 14131–14140, 2021. 2, 8, 10, 18, 21, 22, 27, 37, 38, 53

[32] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*, 2015. 101

[33] Visual Concepts. VISUAL CONCEPTS. https://vcentertainment.com. 4, 88, 94

[34] Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance. *arXiv preprint arXiv:2210.11427*, 2022. 14, 55, 63

[35] Katherine Crowson. Clip guided diffusion hq 256x256. https://colab.research.google.com/drive/12a_Wrfi2_gwwAuN3VvMTwVMz9TfqctNj. 55

[36] Aiyu Cui, Daniel McKee, and Svetlana Lazebnik. Dressing in order: Recurrent person image generation for pose transfer, virtual try-on and outfit editing. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 14638–14647, 2021. 12

[37] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[38] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

[39] Olivier Dionne and Martin de Lasa. Geodesic voxel binding for production character meshes. In *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 173–180. ACM, 2013. 107

[40] 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*, pages 3480–3489, 2022. 4, 8, 21, 53

[41] Vincent Dumoulin, Ethan Perez, Nathan Schucher, Florian Strub, Harm de Vries, Aaron Courville, and Yoshua Bengio. Feature-wise transformations. *Distill*, 3(7):e11, 2018. 25, 27, 54

[42] Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Networks*, 107:3–11, 2018. 26

[43] Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. RMPE: Regional multi-person pose estimation. In *ICCV*, 2017.

[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, 2022. 54

[45] Rinon Gal, Moab Arar, Yuval Atzmon, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. Encoder-based domain tuning for fast personalization of text-to-image models. 02 2023. 54

[46] Michael Garland and Paul S Heckbert. Surface simplification using quadric error metrics. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 209–216. ACM Press/Addison-Wesley Publishing Co., 1997. 102

[47] 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. 4, 8, 11, 21, 53

[48] Marcel Germann, Alexander Hornung, Richard Keiser, Remo Ziegler, Stephan Würmlin, and Markus Gross. Articulated billboards for video-based rendering. In *Computer Graphics Forum*, pages 585–594. Wiley Online Library, 2010. 91

[49] Rohit Girdhar, David F Fouhey, Mikel Rodriguez, and Abhinav Gupta. Learning a predictable and generative vector representation for objects. In *European Conference on Computer Vision*, pages 484–499. Springer, 2016. 97

[50] Ke Gong, Yiming Gao, Xiaodan Liang, Xiaohui Shen, Meng Wang, and Liang Lin. Graphonomy: Universal human parsing via graph transfer learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7450–7459, 2019. 11, 22, 32

[51] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 13, 21

[52] Google. Google Virtual Try-On. https://blog.google/products/shopping/ai-virtual-try-on-google-shopping/.

[53] Oliver Grau, Adrian Hilton, Joe Kilner, Gregor Miller, Tim Sargeant, and Jonathan Starck. A free-viewpoint video system for visualization of sport scenes. *SMPTE motion imaging journal*, 116(5-6):213–219, 2007. 91

[54] Oliver Grau, Graham A Thomas, A Hilton, J Kilner, and J Starck. A robust free-viewpoint video system for sport scenes. In *2007 3DTV conference*, pages 1–4. IEEE, 2007. 91

[55] Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan Russell, and Mathieu Aubry. AtlasNet: A Papier-Mâché Approach to Learning 3D Surface Generation. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[56] Thibault Groueix, Matthew Fisher, Vladimir G Kim, Bryan C Russell, and Mathieu Aubry. 3d-coded: 3d correspondences by deep deformation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 230–246, 2018. 96, 107, 111, 114

[57] Thibault Groueix, Matthew Fisher, Vladimir G Kim, Bryan C Russell, and Mathieu Aubry. A papier-mâché approach to learning 3d surface generation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 216–224, 2018. 96

[58] Jiatao Gu, Shuangfei Zhai, Yizhe Zhang, Josh Susskind, and Navdeep Jaitly. Matryoshka diffusion models. *arXiv preprint arXiv:2310.15111*, 2023.

[59] Jean-Yves Guillemaut and Adrian Hilton. Joint multi-layer segmentation and reconstruction for free-viewpoint video applications. *IJCV*, 2011. 91

[60] Jean-Yves Guillemaut, Joe Kilner, and Adrian Hilton. Robust graph-cut scene segmentation and reconstruction for free-viewpoint video of complex dynamic scenes. In *ICCV*, 2009. 91

[61] Riza Alp Guler and Iasonas Kokkinos. Holopose: Holistic 3d human reconstruction in-the-wild. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 92

[62] Marc Habermann, Weipeng Xu, , Michael Zollhoefer, Gerard Pons-Moll, and Christian Theobalt. Livecap: Real-time human performance capture from monocular video. *ACM Transactions on Graphics, (Proc. SIGGRAPH)*, 2019. 92

[63] Ikhsanul Habibie, Weipeng Xu, Dushyant Mehta, Gerard Pons-Moll, and Christian Theobalt. In the wild human pose estimation using explicit 2d features and intermediate 3d representations. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 91

[64] 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*, pages 10471–10480, 2019. 10, 21

[65] 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*, pages 7543–7552, 2018. 4, 8, 18, 21, 53

[66] Ayaan Haque, Matthew Tancik, Alexei A Efros, Aleksander Holynski, and Angjoo Kanazawa. Instruct-nerf2nerf: Editing 3d scenes with instructions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19740–19750, 2023. 119

[67] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 96

[68] 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. 4, 8, 21, 53

[69] Amy Heckerling. Clueless, July 1995. https://www.imdb.com/title/tt0112697/. 1

[70] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022. 15, 55, 63

[71] 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. *Advances in neural information processing systems*, 30, 2017. 28, 62

[72] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al.

Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 13, 55, 56, 58

[73] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 12, 21, 22, 23, 24, 53, 59

[74] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *J. Mach. Learn. Res.*, 23:47–1, 2022. 13, 23

[75] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 26, 59

[76] Emiel Hoogeboom, Jonathan Heek, and Tim Salimans. simple diffusion: End-to-end diffusion for high resolution images. *arXiv preprint arXiv:2301.11093*, 2023. 52, 58

[77] Yinghao Huang, Federica Bogo, Christoph Lassner, Angjoo Kanazawa, Peter V Gehler, Javier Romero, Ijaz Akhter, and Michael J Black. Towards accurate marker-less human shape and pose estimation over time. In *2017 International Conference on 3D Vision (3DV)*, pages 421–430. IEEE, 2017. 92

[78] Getty Images. Getty Images. https://www.gettyimages.com. 112

[79] Intel. Intel True View. www.intel.com/content/www/us/en/sports/technology/true-view.html. 91

[80] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1325–1339, 2013. 91

[81] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017. 99

[82] Thibaut Issenhuth, Jérémie Mary, and Clément Calauzènes. 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. 4, 8, 11, 21, 53

[83] Alec Jacobson, Ilya Baran, Jovan Popovic, and Olga Sorkine. Bounded biharmonic weights for real-time deformation. *ACM Trans. Graph.*, 30(4):78, 2011.

[84] Yasamin Jafarian, Tuanfeng Y Wang, Duygu Ceylan, Jimei Yang, Nathan Carr, Yi Zhou, and Hyun Soo Park. Normal-guided garment uv prediction for human re-texturing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4627–4636, 2023. 118

[85] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016. 5, 19

[86] Sam Johnson and Mark Everingham. Clustered pose and nonlinear appearance models for human pose estimation. In *Proceedings of the British Machine Vision Conference*, 2010. doi:10.5244/C.24.12. 105

[87] Hanbyul Joo, Tomas Simon, and Yaser Sheikh. Total capture: A 3d deformation model for tracking faces, hands, and bodies. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8320–8329, 2018. 91, 92

[88] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Computer Vision and Pattern Regognition (CVPR)*, 2018. 92, 104, 105, 106

[89] Angjoo Kanazawa, Jason Y. Zhang, Panna Felsen, and Jitendra Malik. Learning 3d human dynamics from video. In *Computer Vision and Pattern Regognition (CVPR)*, 2019. 92

[90] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. *Advances in Neural Information Processing Systems*, 33:12104–12114, 2020. 27, 31, 62

[91] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8110–8119, 2020. 12, 18, 21

[92] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6007–6017, 2023. 55

[93] Gwanghyun Kim, Taesung Kwon, and Jong Chul Ye. Diffusionclip: Text-guided diffusion models for robust image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2426–2435, 2022. 55

[94] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 26, 102

[95] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 13

[96] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2017.

[97] Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. *arXiv preprint arXiv:2305.01569*, 2023. 65

[98] Alexander Kolesnikov, Alexey Dosovitskiy, Dirk Weissenborn, Georg Heigold, Jakob Uszkoreit, Lucas Beyer, Matthias Minderer, Mostafa Dehghani, Neil Houlsby, Sylvain Gelly, Thomas Unterthiner, and Xiaohua Zhai. An image is worth 16x16 words: Transformers for image recognition at scale. 2021.

[99] Nikos Kolotouros, Georgios Pavlakos, Michael J. Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *Proceedings of the IEEE International Conference on Computer Vision*, 2019. 92, 104, 105, 106

[100] Nikos Kolotouros, Georgios Pavlakos, and Kostas Daniilidis. Convolutional mesh regression for single-image human shape reconstruction. In *CVPR*, 2019. 92, 104, 105, 107, 111, 114

[101] Philipp Krähenbühl. Free supervision from video games. In *CVPR*, 2018. 90

[102] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *CVPR*, 2023. 54

[103] Christoph Lassner, Javier Romero, Martin Kiefel, Federica Bogo, Michael J Black, and Peter V Gehler. Unite the people: Closing the loop between 3d and 2d human representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6050–6059, 2017. 92

[104] Verica Lazova, Eldar Insafutdinov, and Gerard Pons-Moll. 360-degree textures of people in clothing from a single image. In *International Conference on 3D Vision (3DV)*, sep 2019.

[105] Sangyun Lee, Gyojung Gu, Sunghyun Park, Seunghwan Choi, and Jaegul Choo. High-resolution virtual try-on with misalignment and occlusion-handled conditions. In *Proceedings of the European conference on computer vision (ECCV)*, 2022. 2, 8, 11, 18, 21, 27, 28, 30, 34, 35, 36, 39, 40, 53

[106] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. Epnp: An accurate o (n) solution to the pnp problem. *International journal of computer vision*, 81(2):155, 2009. 96

[107] Kathleen M Lewis, Srivatsan Varadharajan, and Ira Kemelmacher-Shlizerman. Try-ongan: Body-aware try-on via layered interpolation. *ACM Transactions on Graphics (TOG)*, 40(4):1–10, 2021. 12, 18, 21, 27, 28, 34, 35, 36, 39, 40, 53

[108] Dongxu Li, Junnan Li, and Steven C.H. Hoi. Blip-diffusion: Pre-trained subject representation for controllable text-to-image generation and editing. 05 2023. 54

[109] Zhi Li, Pengfei Wei, Xiang Yin, Zejun Ma, and Alex C Kot. Virtual try-on with pose-garment keypoints guided inpainting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22788–22797, 2023. 53

[110] Shanchuan Lin, Bingchen Liu, Jiashi Li, and Xiao Yang. Common diffusion noise schedules and sample steps are flawed. *arXiv preprint arXiv:2305.08891*, 2023.

[111] Dong C Liu and Jorge Nocedal. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1-3):503–528, 1989. 103

[112] Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B Tenenbaum. Compositional visual generation with composable diffusion models. In *European Conference on Computer Vision*, pages 423–439. Springer, 2022. 54

[113] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021.

[114] Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1

[115] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6):248, 2015. 15, 88, 91, 92, 106

[116] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11461–11471, 2022. 55

[117] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. A simple yet effective baseline for 3d human pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2640–2649, 2017. 91, 95, 98, 101, 105

[118] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *2017 International Conference on 3D Vision (3DV)*, pages 506–516. IEEE, 2017. 91

[119] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. Vnect: Real-time 3d human pose estimation with a single rgb camera. *ACM Transactions on Graphics (TOG)*, 36(4):44, 2017. 91, 95, 99

[120] 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*, pages 5084–5093, 2020. 8, 21

[121] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021. 55

[122] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6038–6047, 2023. 55, 63

[123] Gyeongsik Moon, Juyong Chang, and Kyoung Mu Lee. Camera distance-aware top-down approach for 3d multi-person pose estimation from a single rgb image. In *The IEEE Conference on International Conference on Computer Vision (ICCV)*, 2019. 91

[124] Davide Morelli, Alberto Baldrati, Giuseppe Cartella, Marcella Cornia, Marco Bertini, and Rita Cucchiara. Ladi-vton: latent diffusion textual-inversion enhanced virtual try-on. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 8580–8589, 2023. 61, 62, 72, 73

[125] 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 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2231–2235, 2022. 61, 72, 73

[126] Ryota Natsume, Shunsuke Saito, Zeng Huang, Weikai Chen, Chongyang Ma, Hao Li, and Shigeo Morishima. Siclope: Silhouette-based clothed people. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4480–4490, 2019. 92

[127] USA TODAY Network. USA TODAY Network. https://www.commercialappeal.com. 87

[128] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 55

[129] George Papandreou, Tyler Zhu, Nori Kanazawa, Alexander Toshev, Jonathan Tompson, Chris Bregler, and Kevin Murphy. Towards accurate multi-person pose estimation in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4903–4911, 2017. 22, 32

[130] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2337–2346, 2019. 11

[131] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 91, 92, 98, 102, 104, 106

[132] Georgios Pavlakos, Nikos Kolotouros, and Kostas Daniilidis. Texturepose: Supervising human mesh estimation with texture consistency. In *ICCV*, 2019. 92

[133] Georgios Pavlakos, Luyang Zhu, Xiaowei Zhou, and Kostas Daniilidis. Learning to estimate 3d human pose and shape from a single color image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 459–468, 2018. 92

[134] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023. 14, 58

[135] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 63, 65

[136] Gerard Pons-Moll, Javier Romero, Naureen Mahmood, and Michael J. Black. Dyna: A model of dynamic human shape in motion. *ACM Transactions on Graphics, (Proc. SIGGRAPH)*, 34(4):120:1–120:14, August 2015. 91

[137] Michael JD Powell. An efficient method for finding the minimum of a function of several variables without calculating derivatives. *The computer journal*, 7(2):155–162, 1964.

[138] Albert Pumarola, Jordi Sanchez, Gary Choi, Alberto Sanfeliu, and Francesc Moreno-Noguer. 3DPeople: Modeling the Geometry of Dressed Humans. In *ICCV*, 2019. 92

[139] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 25, 55

[140] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67, 2020. 13

[141] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 13, 21, 55

[142] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021.

[143] Fitsum Reda, Janne Kontkanen, Eric Tabellion, Deqing Sun, Caroline Pantofaru, and Brian Curless. Film: Frame interpolation for large motion. In *The European Conference on Computer Vision (ECCV)*, 2022. 5, 19

[144] Konstantinos Rematas, Ira Kemelmacher-Shlizerman, Brian Curless, and Steve Seitz. Soccer on your tabletop. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4738–4747, 2018. 90, 91, 96, 99

[145] 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, pages 13535–13544, 2022. 8, 21

[146] RenderDoc. RenderDoc. https://renderdoc.org. 7

[147] RenderPeople. RenderPeople. https://renderpeople.com. 88

[148] Stephan R. Richter, Zeeshan Hayder, and Vladlen Koltun. Playing for benchmarks. In *ICCV*, 2017. 90

[149] Stephan R. Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *ECCV*, 2016. 90

[150] Kathleen M Robinette, Sherri Blackwell, Hein Daanen, Mark Boehmer, and Scott Fleming. Civilian American and European Surface Anthropometry Resource (CAESAR), final report. volume 1. summary. Technical report, SYTRONICS INC DAYTON OH, 2002. 87, 88

[151] 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*, pages 10684–10695, 2022. 13, 21, 55

[152] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6), November 2017. 91

[153] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 21, 24

[154] 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. *arXiv preprint arXiv:2208.12242*, 2022. 14, 21, 52, 54, 64

[155] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Wei Wei, Tingbo Hou, Yael Pritch, Neal Wadhwa, Michael Rubinstein, and Kfir Aberman. Hyperdreambooth: Hypernetworks for fast personalization of text-to-image models, 2023.

[156] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*, pages 1–10, 2022. 21, 22, 24, 55

[157] 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, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 2022. 13, 19, 21, 24, 25, 26, 54

[158] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 21, 22, 24

[159] Shunsuke Saito, , Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. *arXiv preprint arXiv:1905.05172*, 2019. 16, 88, 92, 104, 106, 109, 110

[160] Shunsuke Saito, Tomas Simon, Jason Saragih, and Hanbyul Joo. Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 84–93, 2020. 16

[161] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022. 59

[162] Shelly Sheynin, Oron Ashual, Adam Polyak, Uriel Singer, Oran Gafni, Eliya Nachmani, and Yaniv Taigman. Knn-diffusion: Image generation via large-scale retrieval. *arXiv preprint arXiv:2204.02849*, 2022. 55

[163] Tomas Simon, Hanbyul Joo, Iain Matthews, and Yaser Sheikh. Hand keypoint detection in single images using multiview bootstrapping. In *CVPR*, 2017. 90

[164] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXifv preprint arXiv:2209.14792*, 2022.

[165] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015. 12, 21, 22, 53

[166] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 15, 23, 55

[167] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019. 12, 21, 53

[168] Olga Sorkine and Marc Alexa. As-rigid-as-possible surface modeling. In *Symposium on Geometry processing*, volume 4, pages 109–116, 2007. 98, 102

[169] Olga Sorkine, Daniel Cohen-Or, Yaron Lipman, Marc Alexa, Christian Rössl, and H-P Seidel. Laplacian surface editing. In *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 175–184, 2004. 98, 102

[170] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8934–8943, 2018. 4

[171] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 529–545, 2018. 91

[172] Dani Valevski, Matan Kalman, Yossi Matias, and Yaniv Leviathan. Unitune: Text-driven image editing by fine tuning an image generation model on a single image. *arXiv preprint arXiv:2210.09477*, 2022. 55

[173] Gül Varol, Duygu Ceylan, Bryan Russell, Jimei Yang, Ersin Yumer, Ivan Laptev, and Cordelia Schmid. BodyNet: Volumetric inference of 3D human body shapes. In *ECCV*, 2018. 92

[174] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 4, 19, 24

[175] Timo von Marcard, Roberto Henschel, Michael J Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 601–617, 2018. 91

[176] Walmart. Walmart Virtual Try-On. https://www.walmart.com/cp/virtual-try-on/4879497. 18

[177] Bochao Wang, Huabin Zheng, Xiaodan Liang, Yimin Chen, Liang Lin, and Meng Yang. Toward characteristic-preserving image-based virtual try-on network. In *Proceedings of the European conference on computer vision (ECCV)*, pages 589–604, 2018. 4, 8, 10, 18, 21, 53

[178] Qixun Wang, Xu Bai, Haofan Wang, Zekui Qin, and Anthony Chen. Instantid: Zero-shot identity-preserving generation in seconds. *arXiv preprint arXiv:2401.07519*, 2024. 14

[179] Su Wang, Chitwan Saharia, Ceslee Montgomery, Jordi Pont-Tuset, Shai Noy, Stefano Pellegrini, Yasumasa Onoe, Sarah Laszlo, David J Fleet, Radu Soricut, et al. Imagen editor and editbench: Advancing and evaluating text-guided image inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18359–18369, 2023. 55, 56, 63

[180] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018.

[181] Daniel Watson, William Chan, Ricardo Martin-Brualla, Jonathan Ho, Andrea Tagliasacchi, and Mohammad Norouzi. Novel view synthesis with diffusion models. *arXiv preprint arXiv:2210.04628*, 2022. 21

[182] Chung-Yi Weng, Brian Curless, and Ira Kemelmacher-Shlizerman. Photo wake-up: 3d character animation from a single photo. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5908–5917, 2019. 15, 88, 92

[183] Chung-Yi Weng, Brian Curless, Pratul P. Srinivasan, Jonathan T. Barron, and Ira Kemelmacher-Shlizerman. HumanNeRF: Free-viewpoint rendering of moving people from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16210–16220, June 2022. 119

[184] Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018. 26

[185] Donglai Xiang, Hanbyul Joo, and Yaser Sheikh. Monocular total capture: Posing face, body, and hands in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 92

[186] Donglai Xiang, Fabian Prada, Timur Bagautdinov, Weipeng Xu, Yuan Dong, He Wen, Jessica Hodgins, and Chenglei Wu. Modeling clothing as a separate layer for an animatable human avatar. *ACM Transactions on Graphics (TOG)*, 40(6):1–15, 2021. 16

[187] Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking. In *European Conference on Computer Vision (ECCV)*, 2018. 95, 98

[188] Zhenyu Xie, Zaiyu Huang, Xin Dong, Fuwei Zhao, Haoye Dong, Xijin Zhang, Feida Zhu, and Xiaodan Liang. Gp-vton: Towards general purpose virtual try-on via collaborative local-flow global-parsing learning. In *Proceedings of the IEEE/CVF Conference*

on Computer Vision and Pattern Recognition, pages 23550–23559, 2023. 53, 61, 62, 72, 73

[189] Yuliang Xiu, Jiefeng Li, Haoyu Wang, Yinghong Fang, and Cewu Lu. Pose Flow: Efficient online pose tracking. In *BMVC*, 2018.

[190] Feng Xu, Yebin Liu, Carsten Stoll, James Tompkin, Gaurav Bharaj, Qionghai Dai, Hans-Peter Seidel, Jan Kautz, and Christian Theobalt. Video-based characters: Creating new human performances from a multi-view video database. *ACM Trans. Graph.*, 30(4):32:1–32:10, July 2011. 92

[191] Weipeng Xu, Avishek Chatterjee, Michael Zollhöfer, Helge Rhodin, Dushyant Mehta, Hans-Peter Seidel, and Christian Theobalt. Monoperfcap: Human performance capture from monocular video. *ACM Trans. Graph.*, 2018. 92

[192] Keyu Yan, Tingwei Gao, Hui Zhang, and Chengjun Xie. Linking garment with person via semantically associated landmarks for virtual try-on. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17194–17204, June 2023. 53

[193] Han Yang, Xinrui Yu, and Ziwei Liu. Full-range virtual try-on with recurrent tri-level transform. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3460–3469, June 2022. 4, 8, 21, 53

[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*, pages 7850–7859, 2020. 4, 8, 21, 53

[195] Ruiyun Yu, Xiaoqi Wang, and Xiaohui Xie. Vtnfp: An image-based virtual try-on network with body and clothing feature preservation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10511–10520, 2019. 4, 8, 21, 53

[196] Andrei Zanfir, Elisabeta Marinoiu, and Cristian Sminchisescu. Monocular 3d pose and shape estimation of multiple people in natural scenes-the importance of multiple scene constraints. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2148–2157, 2018. 92

[197] Jinsong Zhang, Kun Li, Yu-Kun Lai, and Jingyu Yang. Pise: Person image synthesis and editing with decoupled gan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7982–7990, 2021. 8, 21

[198] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *IEEE International Conference on Computer Vision (ICCV)*, 2023. 13

[199] Xingyi Zhou, Qixing Huang, Xiao Sun, Xiangyang Xue, and Yichen Wei. Towards 3d human pose estimation in the wild: A weakly-supervised approach. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[200] Hao Zhu, Xinxin Zuo, Sen Wang, Xun Cao, and Ruigang Yang. Detailed human shape estimation from a single image by hierarchical mesh deformation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 92

[201] Luyang Zhu, Yingwei Li, Nan Liu, Hao Peng, Dawei Yang, and Ira Kemelmacher-Shlizerman. M&m vto: Multi-garment virtual try-on and editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2024. 50

[202] Luyang Zhu, Konstantinos Rematas, Brian Curless, Steve Seitz, and Ira Kemelmacher-Shlizerman. Reconstructing nba players. In *Proceedings of the European Conference on Computer Vision (ECCV)*, August 2020. 87

[203] 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)*, pages 4606–4615, June 2023. 18, 51, 53, 54, 56, 57, 58, 61, 62, 65, 66, 68, 69, 70, 71, 72, 73

[204] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 10, 21