Generative 3D Garment Modeling with Sparse Visual Cues

Yuanhao Wang

CMU-RI-TR-25-39

May 4, 2025

The Robotics Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA

Thesis Committee:
Fernando De la Torre, *chair*
Jun-Yan Zhu
Yehonathan Litman

Submitted in partial fulfillment of the requirements for the degree of Master of Science in Robotics.

To my parents and friends.

Abstract

Professional fashion designers rely on advanced software to create highly detailed 3D garments. However, as digital apparel becomes integral to virtual environments and personalized experiences, there is a growing need for intuitive tools that enable non-experts to design and interact with 3D garments. To broaden accessibility, these tools should function with minimal input, raising a key question: How can we enable high-quality 3D garment generation and manipulation using only sparse visual cues?

This thesis addresses this challenge by leveraging the strong priors of large pre-trained vision foundation models to tackle two core problems: (1) reconstructing and editing 3D garment assets from a single-view image and (2) transferring textures from an in-the-wild image to existing 3D garment models. To this end, we present two complementary systems: GarmentCrafter for 3D garment reconstruction and modification and FabricDiffusion for texture transfer, together democratizing 3D garment creation.

GarmentCrafter enables non-professional users to generate and modify 3D garments from a single image. Existing single-view reconstruction methods often rely on generative models to hallucinate novel views based on a reference image and camera pose but struggle with cross-view consistency. GarmentCrafter addresses this by integrating progressive depth prediction and image warping to approximate novel views, followed by a multi-view diffusion model that refines occluded and unknown clothing regions. By jointly inferring RGB and depth, it enforces cross-view coherence, reconstructing detailed and geometrically accurate garments.

Complementing this, FabricDiffusion transfers fabric textures from a single image onto 3D garments of arbitrary shapes. Inspired by the observation that in fashion industry most garments are constructed by stitching sewing patterns with flat, repeatable textures, we recast texture transfer as extracting distortion-free, tileable textures that can be mapped onto a garment's UV space. Building upon this insight, we train a denoising diffusion model with a large-scale synthetic dataset to rectify distortions in the input texture image, producing a flat texture map that integrates seamlessly with Physically-Based Rendering (PBR) pipelines. This enables realistic relighting under various lighting conditions, and preserves intricate texture details with high visual fidelity.

Together, these systems form a unified framework for generative 3D garment modeling from sparse inputs. They significantly lower the barrier for 3D content creation by allowing users to work with minimal visual guidance while still achieving high levels of realism, detail, and geometric accuracy. By bridging the gap between limited visual input and high-quality 3D output, this thesis takes a step toward making accessible, scalable, and customizable garment modeling a reality.

Acknowledgments

I would like to express my sincere gratitude to my advisor, Professor Fernando De la Torre, for his steadfast support and guidance throughout the past two years. His mentorship has profoundly influenced my development as a researcher - shaping not only my technical skills, but also my ability to think critically and pursue meaningful research directions. I am deeply appreciative of the trust he placed in me and the opportunities he provided to explore and grow.

I am especially grateful to Dr. Cheng Zhang for his invaluable support and collaboration. Our numerous discussions, ranging from high-level research questions to implementation details, greatly enriched my research experience. I am particularly thankful for his dedication and commitment during intensive periods of work, especially in the lead-up to deadlines. His mentorship has been instrumental to my progress.

I would also like to thank Dr. Thabo Beeler and Dr. Jinlong Yang for their mentorship and insightful guidance. I am fortunate to have had the opportunity to collaborate with many talented individuals, and I extend my appreciation to Francisco, Gonçalo, Chenglei, and Alex for their contributions and support.

Finally, I would like to thank my parents for their unwavering encouragement, and my lab mates and friends at CMU for creating an intellectually stimulating and supportive environment that made this journey both productive and memorable.

Funding

Thanks to Dr. Thabo Beeler and Google AR for partial financial support for this work.

Contents

When this dissertation is viewed as a PDF, the page header is a link to this Table of Contents.

List of Figures

List of Tables

Chapter 1

Introduction

There is an increasing interest to experience apparel in 3D for virtual try-on applications and e-commerce as well as an increasing demand for 3D clothing assets for games, virtual reality and augmented reality applications. While there is an abundance of 2D images of fashion items online, and recent generative AI algorithms democratize the creative generation of such images, the creation of high-quality 3D clothing assets remains a significant challenge.

Traditional 3D garment modeling requires expertise in professional design software such as CLO3D or Marvelous Designer, making it inaccessible to non-experts. Moreover, manual creation is time-consuming, as it involves intricate processes like pattern drafting, fabric simulation, and texture mapping. Recent advancements in generative AI have enabled impressive progress in 2D image synthesis, yet translating these capabilities to 3D remains non-trivial. Existing single-view 3D reconstruction methods often rely on deep learning models to synthesize multi-view images[39, 56, 63, 91, 108, 117], but they struggle with cross-view consistency, leading to suboptimal results. Similarly, texture transfer techniques for 3D garments frequently suffer from loss of fine-grained details due to inaccurate registrations and complex surface topology [30, 68, 71].

To bridge this gap, this thesis explores learning-based methods that leverage rich priors from large-scale vision foundation models to simplify 3D garment creation and editing. Specifically, we focus on two key tasks: (1) generating 3D garment models from a single 2D image and (2) transferring texture patterns from in-the-wild clothing images to arbitrary 3D garment models. To this end, we propose two complementary systems: **GarmentCrafter**, which reconstructs high-quality 3D garments with rich geometric and texture details by introducing Progressive Novel View Synthesis (PNVS) mechanism to ensure cross-view consistency, and **FabricDiffusion**, which leverages a data-driven approach to extract distortion-free texture materials for seamless mapping onto garment surfaces. Together, these methods significantly lower the barrier to high-quality 3D garment generation, making it more accessible to common users.

This thesis contributes to the field of 3D vision and generative modeling by introducing novel techniques for garment reconstruction and texture synthesis. Through extensive experiments, we demonstrate that our approaches significantly outperform existing methods in terms of visual fidelity, geometric accuracy, and texture realism. Furthermore, our models generalize well to diverse garment types, materials, and textures, making them applicable across various use cases. By making our models publicly available, we hope to advance research in 3D clothing generation and facilitate new innovations in fashion, gaming, and virtual reality applications.

Chapter 2

Single-View 3D Garment Reconstruction and Editing

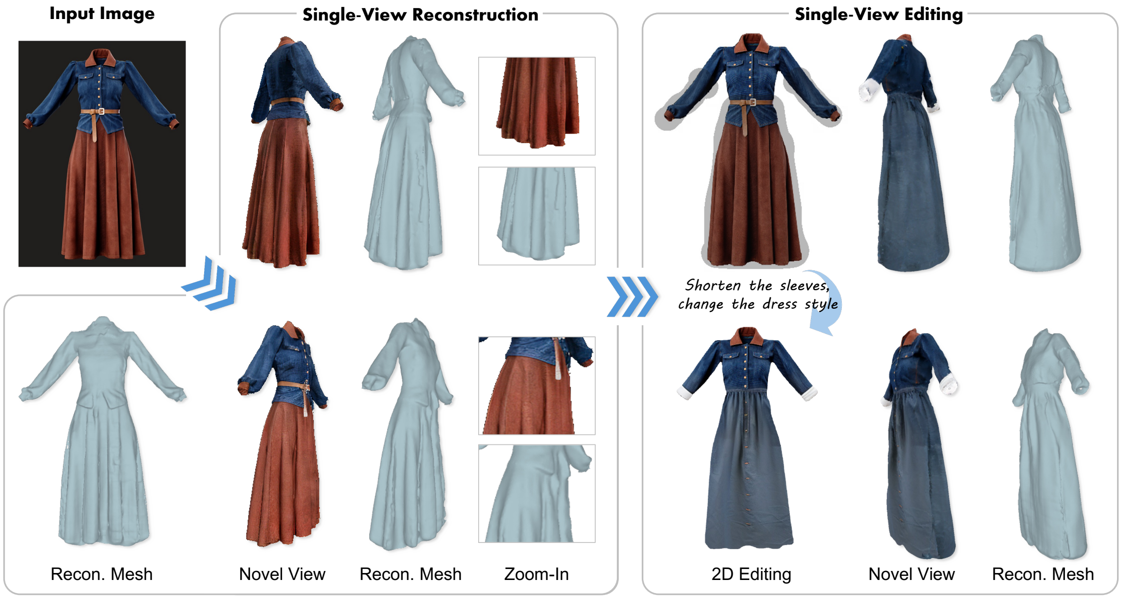

Figure 2.1: From a real-world clothing image, GarmentCrafter synthesizes high-quality novel views, enabling the reconstruction of garment meshes with accurate geometry and rich detail. Additionally, users can easily apply 2D edits (e.g., modifying parts or surface details) using off-the-shelf tools on a single image, and GarmentCrafter seamlessly applies these edits across the 3D model with multi-view consistency.

2.1 Introduction

Professional fashion designers use sophisticated software to create and edit garments in 3D, crafting highly detailed virtual apparels [1, 2, 3, 4]. However, as digital garments become integral to virtual environments and personalized digital experiences [12, 32, 42, 85, 116], there is a growing demand for intuitive tools that allow non-professional users to design and interact with 3D garments. For broader accessibility, such tools should allow users to work with 3D garments with minimal input, ideally from just a single image. This raises a key question: *How can we create and edit 3D garments with simple manipulations in an image?*

Recent advancements in image generation models [78, 84, 86, 106] and image editing techniques [10, 74, 77, 107, 131, 138] have enabled high-quality garment design in 2D. Yet, achieving the same level of control and realism for 3D garments remains challenging for common users. Currently, state-of-the-art methods on single-view 3D garments rely either on 1) deforming, matching, and registration with the human body prior [65] and/or predefined garment templates [8, 19, 30, 57, 61, 68, 88], or 2) novel view synthesis techniques [63, 111] that use pre-trained 2D diffusion models conditioned on a reference image and target pose. However, they often fall short in capturing accurate, realistic geometry and appearance.

Two characteristics of garments pose challenges. First, garments exhibit diverse shapes, complex geometries, and rich textures, making template-based methods limited in their ability to generalize across clothing styles. Most existing methods prioritize either geometry [19, 67] or texture [79, 125], rarely balancing both [30, 68, 88]. Second, the fine details in garments demand stronger multi-view consistency. Existing novel view synthesis methods [64, 117], conditioned on a reference image and target pose, often neglect critical semantic connections across different views.

How can we ensure that a pixel in one view corresponds to a point visible in another, with consistent appearance? In this paper, we propose a different approach, *progressive novel view synthesis*, to enhance cross-view coherence. Our method begins by estimating the depth of the input image and warping projected points to approximate unseen views. We then apply a multi-view diffusion model to complete missing and occluded regions based on the evolving camera pose. Furthermore, we incorporate a monocular depth estimation model to generate depth maps that

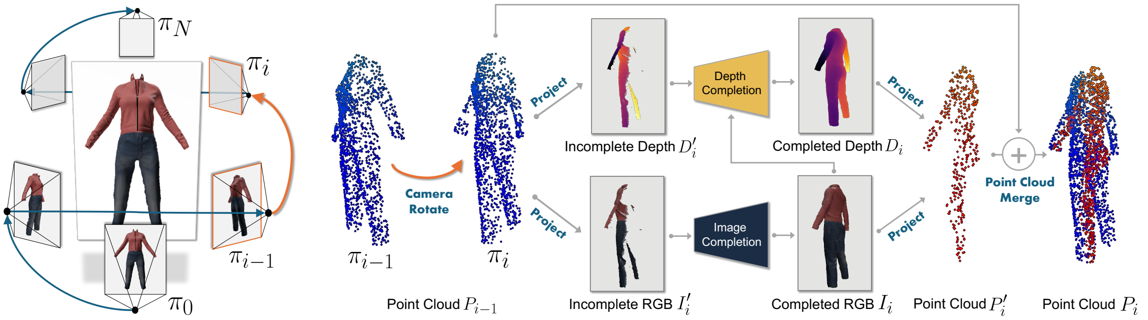

Figure 2.2: **An illustration of progressive novel view synthesis in GarmentCrafter. Left:** Given a garment image, our method performs depth-aware novel view synthesis along a predefined zigzag camera trajectory. **Right:** For each camera rotation from π_{i-1} to π_i, we project the current point cloud P_{i-1} into the image space based on camera pose π_i, resulting in incomplete RGB and depth images. Our diffusion model completes the RGB image using the warped view, input image, and camera pose as conditions, while a depth completion network refines the depth map based on the completed RGB, warped depth, and camera pose. The re-projected point cloud P_i' is then merged with P_{i-1} to produce an updated point cloud P_i. This iterative process continues until a full 3D representation of the garment is achieved.

remain consistent with the warped depths. Unlike existing novel view synthesis, our key insight is to use the depth-based warped image as an additional condition to guide cross-view alignment. By progressively synthesizing views and depths along a predefined camera trajectory, our method gradually refines the geometry and texture of the garment across viewpoints.

We name our method *GarmentCrafter*, a novel solution for 3D garment creation and editing while users just need to operate on a single-view image, as shown in Figure 2.1. Specifically, GarmentCrafter not only generates high-quality 3D garments but also extends garment editing from 2D to 3D. Thanks to our progressive novel view synthesis, users can make local edits (e.g., editing surface details) or perform part-based manipulations (e.g., modifying garment parts) directly on a single-view image, with precise effects reflected in 3D space — capabilities that are absent in the existing methods [88]. Trained on large-scale 3D garment datasets [9, 21, 139], GarmentCrafter demonstrates superior performance on held-out 3D garment data as well as in-the-wild clothing images. Extensive experiments show that our method outperforms state-of-the-art 2D-to-3D garment reconstruction approaches in terms of geometric accuracy, visual fidelity, and cross-view consistency.

2.2 Related Work

Single-View 3D Garment Reconstruction and Editing. Reconstructing 3D garments from a single image has been widely explored, with existing methods approaching the task from several perspectives. One line of work relies on parametric body templates, such as SMPL [8, 19, 44, 72], or employs 2D shape priors and keypoint-based techniques [130] to optimize garment structure. Another category of work uses explicit or implicit 3D parametric garment models [8, 20, 30, 57, 67, 68, 88, 137] to capture garment shape and support pose-guided deformations. Additionally, some methods incorporate garment sewing patterns [7, 15, 17, 43, 61, 119, 139], offering flexibility by reconstructing garments from 2D panels. However, these works often struggle to capture diverse garment styles and fine surface details (e.g., wrinkles), and lack support for intuitive garment manipulation, such as modifying surface details or garment parts. In contrast, GarmentCrafter prioritizes novel view synthesis for detailed geometry and texture reconstruction, without relying on garment templates or human body priors, allowing it to handle a wide range of garment styles. Furthermore, single-view edits can also be seamlessly extended to the 3D model. Note that, our focus in this paper is on garments in a rest pose — well suited to the fashion industry, where ease of adjustment is essential.

Novel View Synthesis from Sparse Images. Our method is inspired by novel view synthesis. Popular approaches such as Neural Radiance Fields (NeRFs) [70] and 3D Gaussian Splatting (3D-GS) [48] rely on numerous posed inputs, limiting their use in single-view scenarios. Recently, distillation from pre-trained 2D generative models has emerged as a promising solution for hallucinating novel views from limited input, with applications in human digitization [5, 34, 37, 51, 87, 114, 115, 129] and object-centric reconstruction [41, 41, 62, 63, 64, 75, 92, 96, 111, 136]. However, these methods often lack cross-view consistency and high-quality details, crucial for garment-focused tasks. Unlike models that sample views independently, our method takes semantic cues (i.e., wrapped images) from other views as an additional condition for view synthesis. This might be reminiscent of scene-level approaches, such as Perpetual View Synthesis [11, 18, 45, 60, 99, 123], which condition on warped images for neighbor view image completion. However, we note that scene-centric methods often lack the precision needed for object-centric cases (e.g., garment manipulation) and overlook loop closure for garment shape completion. Our work represents a novel attempt of progressive view synthesis with a predefined camera trajectory for garment reconstruction and editing.

Image-to-3D Reconstruction. Our approach builds on recent advancements in image-to-3D reconstruction, where most methods distill pre-trained generative models via per-scene optimization [16, 54, 75, 94, 105] or multi-view diffusion techniques [41, 62, 63, 64, 91, 104, 136]. With the availability of large-scale 3D datasets [21, 22], Large Reconstruction Models (LRMs) [39, 56, 97, 117, 118] are being trained for feed-forward image-to-3D generation. Unlike Zero-1-to-3 and its variants [63], our method leverages diffusion models to progressively condition on warped images with carefully designed camera trajectory and error reduction methods to enhance cross-view consistency. Additionally, we curated a 3D garment dataset, incorporating assets from existing 3D collections [9, 21, 139], allowing our model to synthesize highly detailed, multi-view images and corresponding depth maps. This process yields multi-view images alongside accurate depth maps, enabling high-quality mesh reconstruction through standard point cloud-to-mesh methods [47]. While we demonstrate point aggregation and mesh reconstruction in our work, our primary focus is on advancing the multi-view and depth synthesis stages rather than optimizing the point-to-mesh conversion process itself.

2.3 Approach

We first present problem statement in Section 2.3.1, followed by our proposed progressive novel view synthesis in Section 2.3.2. We introduce garment-centric applications enabled by our method in Section 2.3.3. We describe the details of data curation and model training methods in Section 2.3.4.

2.3.1 Problem Definition

Given a single-view garment image I_0, our goal is to generate consistent novel views with detailed RGB textures and accurate depths, which support both single-view 3D reconstruction and editing. Specifically, we first estimate a depth map D_0 based on the input I_0. Then, we project every pixel in the foreground of the garment to the world space, creating a colored point cloud P_0. Our goal is to complete this point cloud by sequentially incorporating information from synthesized novel views. To achieve this, we propose an progressive 3D completion process with a predefined camera trajectory $\boldsymbol{\pi} = \{\pi_1, \pi_2, ..., \pi_N\}$ that forms a closed loop around the garment object. Figure 2.2 illustrates the overall framework. Next, we elaborate the details of an arbitrary step in the following sections.

2.3.2 Progressive Novel View Synthesis

Overview. At the step i of the progressive novel view synthesis (see Figure 2.2), we first project the existing point cloud P_{i-1} to the image plane of camera $\pi_i \in \boldsymbol{\pi}$, producing an incomplete image I'_i and an incomplete depth map D'_i. We then apply an image completion model to inpaint the missing areas in I'_i, resulting in I_i. Next, we use an monocular depth estimation model to estimate the corresponding depth map D_i consistent with the known depths in D'_i. Finally, we integrate I_i and D_i with the existing point cloud to obtain a merged P_i. By following a predefined camera trajectory, our method can generate view-dependent images and corresponding depths that enable high-quality garment reconstruction and edit with improved cross-view consistency.

Conditional Image Generation. At step i, the goal is to synthesize $I_i \in \mathbb{R}^{H\times W\times 3}$, the image of the garment object from the viewpoint of camera π_i, given the input image I_0, the projected image I'_i, and the relative camera rotation $R_i \in \mathbb{R}^{3\times 3}$ and translation $T_i \in \mathbb{R}^3$ from π_0 to π_i. We aim to train a model f_{img} such that:

$$I_i = f_{\text{img}}(I_0, I'_i, R_i, T_i), \tag{2.1}$$

where I_i is the synthesized complete image that retains the appearance of I'_i in the known regions, and synthesizes plausible appearance in the unknown regions that remain perceptually consistent with I'_i and the original input I_0.

To learn f_{img}, we fine-tune a denoising diffusion model, leveraging its strong generalization capabilities in image generation. Specifically, we adopt a latent diffusion architecture based on Stable Diffusion [84] with an image encoder $\mathcal{E}$, a denoising network ϵ_θ, and a decoder $\mathcal{D}$. At denoising step $s \in S$, let z_s denote the noisy latent of the target image $x = I_i$, and let $\boldsymbol{c} = c(I_0, I'_i, R_i, T_i)$ be the embedding of the anchor view image, target view projected image, and relative camera extrinsics. We optimize the following latent diffusion objective:

$$\mathcal{L}(\theta) = \mathbb{E}_{\mathcal{E}(I_0),\mathcal{E}(I'_i),\epsilon\sim\mathcal{N}(0,\mathbf{I}),s}\left[\|\epsilon - \epsilon_\theta(z_s, s, \boldsymbol{c})\|^2\right]. \tag{2.2}$$

Unlike existing multi-view diffusion models (e.g., [63, 91]), which synthesize novels views from an arbitrary input viewpoint, we unify our garment-centric task by fixing the input image to a near-frontal view of the garment. This allows R_i and T_i to be interpreted as the absolute camera transformation from the frontal view. Furthermore, in addition to conditioning on the anchor view image, we incorporate the warped image (i.e., I'_i in Figure 2.2 and Equation 2.1) at the target view as an additional condition input, which provides a strong prior that enhances cross-view consistency in garment reconstruction, as demonstrated in Section 2.4.4.

Conditional Depth Generation. After obtained complete RGB image I_i, we learn a depth model f_{depth} to estimate the depth map $D_i \in \mathbb{R}^{H\times W\times 1}$ conditioned on the warped incomplete depth map D'_i as follows:

$$D_i = f_{\text{depth}}(I_i, D'_i) \tag{2.3}$$

Similar to the conditional image generation, we enforce depth preservation in known regions by framing the task as metric depth estimation. To ensure consistency, we align the depth values of D_i and D'_i during training. The model is optimized using an $\mathcal{L}_1$ loss:

$$\mathcal{L}_1 = ||(D_i - \hat{D}_i)\cdot m||, \tag{2.4}$$

where $\hat{D}_i$ is the ground-truth depth, and m is the foreground mask. To train f_{depth}, we fine-tune the pretrained human foundation model, Sapiens [49], leveraging its strong priors for human-related tasks. To condition the model on D'_i, we concatenate D'_i with I_i as input and add an extra channel to the first projection layer of Sapiens model. The weights of the added channel are initialized to zero.

Point Cloud Merging and Projection. To integrate novel view observations (i.e., I_i and D_i) into the existing point cloud P_{i-1}, we first identify the inpainted regions from the image model. Pixels in these regions are projected into world space and merged with P_{i-1} to form P_i, with expanded borders to include overlapping regions. To minimize stitching artifacts, we align the depth map of the inpainted regions with the warped depth map of P_{i-1}. When projecting a partial point cloud to a novel view, only surfaces facing the camera should be rendered. To enforce this, we track the orientation of each point. For a point x added at step i, its orientation vector v is derived from the normal direction of the corresponding pixel in D_i. During projection, a point is ignored if dot $(v, v_0) < 0$, where v_0 is the viewing direction. After completing all steps along the camera trajectory, we optionally sample a few random views for additional inpainting to recover any occluded regions. Please see supplementary for additional details.

2.3.3 Garment Digitization and Editing

Garment Digitization. Our method enables garment digitization from a single image by progressively synthesizing novel views, generating multi-view consistent RGBD images and a colored point cloud. This output serves as an intermediate representation for various 3D reconstruction. In this work, we employ Screened Poisson surface reconstruction [47] to convert the point cloud into a textured mesh. Specifically, we project multi-view RGBD images to form a colored point cloud, where each point encodes geometry and color. The Screened Poisson method then interpolates these attributes, mapping textures onto mesh vertices.

Interactive Editing. Redesigning a 3D garment model typically requires significant expertise, making it impractical for most users. GarmentCrafter provides an intuitive alternative, allowing users to edit a rendered image of the garment from a selected view, which is then lifted into 3D. In this work, we focus on two types of edits: (1) *Part-based Editing*: Modifies the geometry or texture of specific garment parts, such as sleeves or pant legs. Users can add, remove, or resize components. (2) *Local surface editing*: Adjusts the geometry and texture of localized regions, such as adding a pocket or modifying the neckline design.

The garment part editing is achieved with the following strategy. Given a 3D garment object G, the user selects an anchor view π and edits the rendered image I to obtain I_{edit}. We first identify the edited region in I_{edit} and remove the corresponding garment parts from G, leaving a partial garment G' that remains unchanged. This reformulates the task as single-view 3D garment part reconstruction, conditioned on G'. We then follow the process described in Section 2.3.2 with two modifications: (1) At each step along the camera trajectory, the conditional image and depth are generated by combining the projected point cloud with observations from the partial garment G'. (2) After computing image and depth maps, only pixels within the edited region are projected and merged with the existing point cloud. The final output is a colored point cloud of the edited parts, which is then merged with G'. For local surface editing, instead of removing and reconstructing an entire garment part, we apply the same process to a localized surface region.

2.3.4 Data Preparation and Training

We construct the training dataset by simulating inference. For each 3D garment, we sample 6 uniform views at 0° elevation (following the full camera trajectory) and 4 additional random views between 60° and $-30°$ for inpainting.

Training Data for Reconstruction. We follow the zigzag camera trajectory (Figure 2.2) and at each step i, we form a training pair for the image generation model f_{img}: $\{(I'_i, I_0, R_i, T_i), I_i\}$, where I'_i is the projected image, I_0 is the anchor view, and (R_i, T_i) are the relative camera transformations. Similarly, the depth generation model f_{depth} is trained with $\{(D'_i, I_i), D_i\}$, where D'_i is the projected depth, and D_i is the ground-truth depth. We merge the point cloud with I_i and D_i before proceeding. Finally, we repeat the process for four random views to simulate inpainting.

Training Data for Editing. For 3D editing, we generate training data by randomly removing parts of a 3D garment to create a partial known model. At each step, we create a partial image I''_i and depth map D''_i by merging I'_i and D'_i with known observations. The training pairs become $\{(I''_i, I_0, R_i, T_i), I_i\}$ for f_{img} and $\{(D''_i, I_i), D_i\}$ for f_{depth}.

Joint Training. To learn a unified model for both reconstruction and editing, we combine their training data. We randomly apply small rotations to the 3D object when generating the training data, enabling the model to handle in-the-wild inputs that may not be well-posed. Please refer to the supplementary materials for details.

2.4 Experiments

We present experimental results of our method on single-view garment reconstruction and editing. Please see supplementary for additional details, analyses, and results.

2.4.1 Datasets, Metrics, and Baselines

Datasets. We validate GarmentCrafter using 3D garment assets from a number of sources. (1) Curated dataset: We collect ∼700 3D garments with diverse shape and texture from Artstation[1]. (2) Objaverse 1.0 (Garment) [21]: the original v1.0 dataset contains more than 800K 3D objects, where most of the existing method trained on [63, 117, 120]. We manually curated a subset only contain ∼900 high-quality garment assets. (3) BEDLAM [9]: 114 garments, each has many textures, ∼1600 garments in total. (4) Cloth4D [139]: ∼1100 artists made garments.

Quantitative Metrics. (1) Texture and appearance quality: we evaluate the novel view synthesis using commonly used LPIPS [127], PSNR [40], SSIM [110]. (2) Geometry quality: we measure the performance using geometric errors with Chamfer distance (bi-directional point-to-mesh) between ground-truth and reconstructed meshes.

Baselines. We compare GarmentCrafter with state-of-the-art models for image-to-3D object and image-to-garment reconstruction. (1) InstantMesh [117]: object reconstruction by generating novel views using Zero-1-to-3++ [91]. (2) CRM [108]: generate six orthographic views for 3D object reconstruction. (3) Hunyuan3D-1.0 [120]: a newly released model for high-quality image-to-3D object reconstruction. (4) Garment3DGen [88]: a state-of-the-art garment-specific model based on template optimization, with templates initialized by InstantMesh [117]. As the texture code is not released, we compare only mesh geometry.

2.4.2 Results on Single-View Reconstruction

We evaluate GarmentCrafter on single-view reconstruction using a held-out test dataset of 150 garment assets. For each test case, we sample 12 views with alternating elevations of 0° and 20° and azimuth angles evenly spaced over 360°. To assess image

[1] https://www.artstation.com/

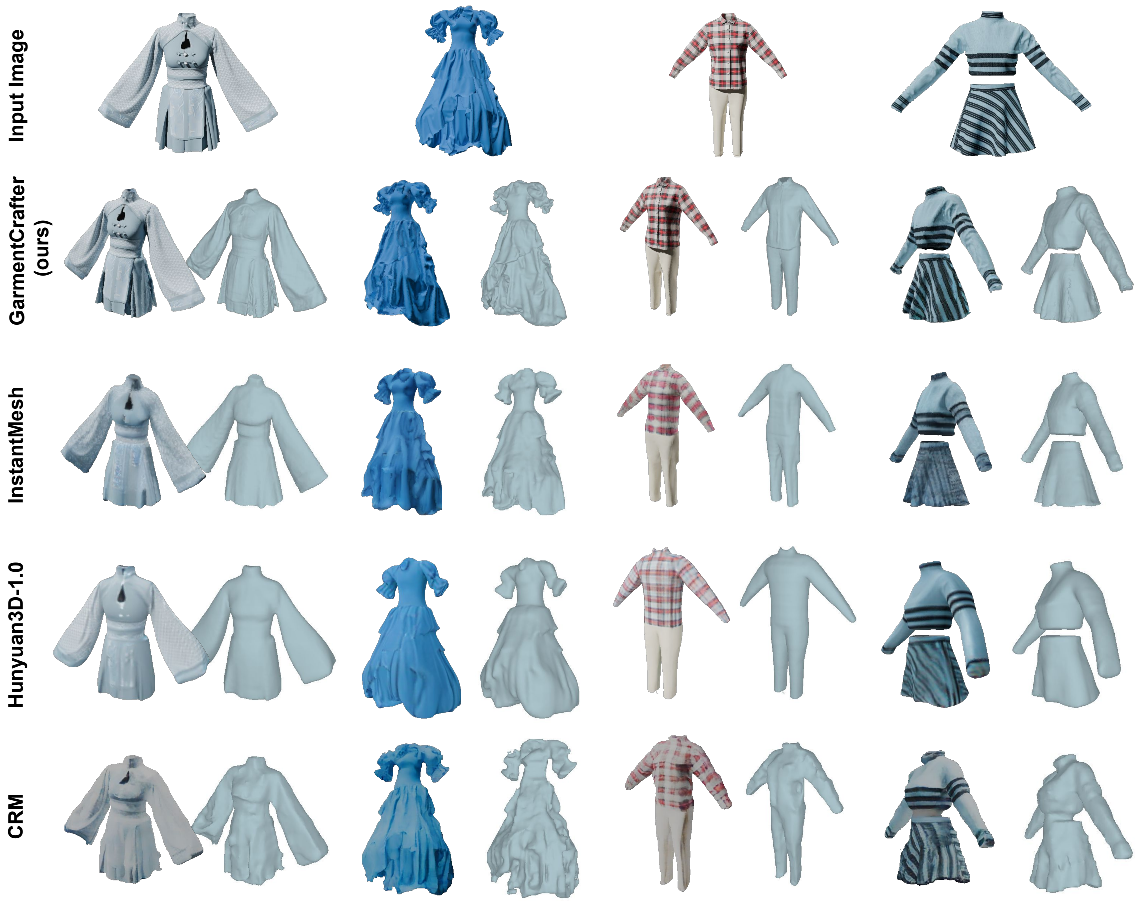

Figure 2.3: **Qualitative comparison on single-view 3D garment reconstruction with state-of-the-art methods.** Our method demonstrates better performance in handling complex texture patterns and geometric structures compared to InstantMesh [117], Hunyuan3D-1.0 [120], and Convolutional Reconstruction Model (CRM) [108].

quality, we convert the generated point clouds to meshes using a classical surface reconstruction method and render multi-view images. For geometry evaluation, we compute the Chamfer distance directly between the generated point cloud and the ground-truth mesh.

Qualitative Results. Figure 2.3 shows qualitative comparisons, where GarmentCrafter demonstrates superior texture and geometry generation compared to all other baselines. Our method, benefiting from consistent multi-view generation, produces sharp textures and intricate geometric details, whereas other baselines often result in blurry textures and overly smoothed geometries. Figure 2.4 shows additional

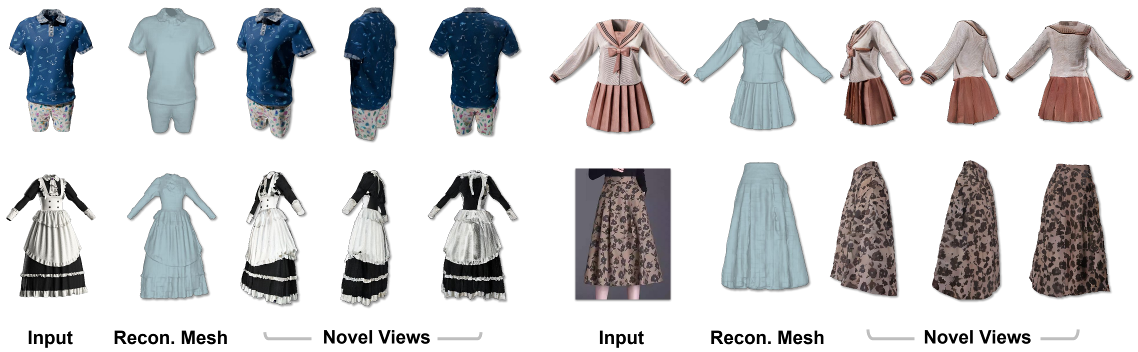

Figure 2.4: More qualitative results of GarmentCrafter on single-view reconstruction. Please see supplementary for more results.

Table 2.1: **Quantitative comparison of texture and geometry quality.** InstantMesh⋆: with fine-tuned Zero-1-to-3++ on our garment data for a fair comparison. CRM and Hunyuan3D-1.0 require significant computing for full fine-tuning, making it impractical. Garment3DGen does not provide texture reconstruction code.

	Appearance			Geometry
	LPIPS↓	PSNR↑	SSIM↑	Chamfer↓
InstantMesh⋆ [117]	<u>0.1848</u>	<u>19.14</u>	0.7944	0.0139
CRM [108]	0.2213	17.51	<u>0.8131</u>	0.0127
Hunyuan3D-1.0 [120]	0.2216	17.77	0.7794	<u>0.0121</u>
Garment3DGen [88]	–	–	–	0.0123
	0.1190	**22.36**	**0.8317**	**0.0044**

qualitative results of GarmentCrafter.

Quantitative Results on Texture Quality. We conduct a quantitative analysis of texture quality on our held-out test dataset and show results in Table 2.1. Across all image quality metrics, GarmentCrafter consistently surpasses baseline methods, demonstrating its effectiveness in producing high-fidelity textures and preserving fine-grained details.

Quantitative Results on Geometry Quality. We present quantitative geometry evaluation results in Table 2.1. GarmentCrafter outperforms baseline methods in terms of Chamfer distance, highlighting its enhanced ability to capture detailed surface geometries in 3D garment shapes.

Table 2.2: Ablation study on Progressive Novel View Synthesis (P-NVS) and analysis on multi-view consistency. We show results with and without P-NVS. CVCS: Cross-View Consistency Score.

P-NVS	LPIPS ↓	PSNR ↑	SSIM ↑	CVCS↑
	0.1195	21.512	0.8369	0.9030
	0.1052	**22.776**	**0.8557**	**0.9512**

2.4.3 Results on Single-View Editing

We present qualitative results on single-view editing in Figure 2.6, showcasing various types of edits, including resizing, element swapping, and surface editing. GarmentCrafter successfully applies 3D edits that are consistent with the 2D edits, while preserving cross-view consistency.

2.4.4 Analyses and Ablation Studies

Importance of Progressive Novel View Synthesis. A key insight of our method is to progressively synthesize novel view by conditioning the generation on the projected images. We conduct an ablation study on the effect of projected image conditioning. For each test case, we select an anchor view π_1, and a second camera view, π_2, at a 60° azimuthal angle relative to π_1. We compare the performance of our image model with or without projected image conditioning at synthesizing view π_2 in Table 2.2. We observe a drop in performance measured in image similarity metrics when removing the projected condition.

Analysis on multi-view consistency. Common image metrics (e.g., LPIPS, PSNR, and SSIM) measure similarity but do not directly reflect cross-view consistency. To address this, we propose a new metric, the Cross-View Consistency Score (CVCS), to gain deeper insights into the consistency performance of our model.

$$\text{CVCS} = 1 - \frac{\Sigma|I - I'| \cdot m'}{\Sigma m'} \tag{2.5}$$

where I is the synthesized image at camera view π, I' is a partial image projected from an observed view π_0 with known depth, and m' is a binary mask indicating the projection regions. This assumes π and π_0 are relatively close.

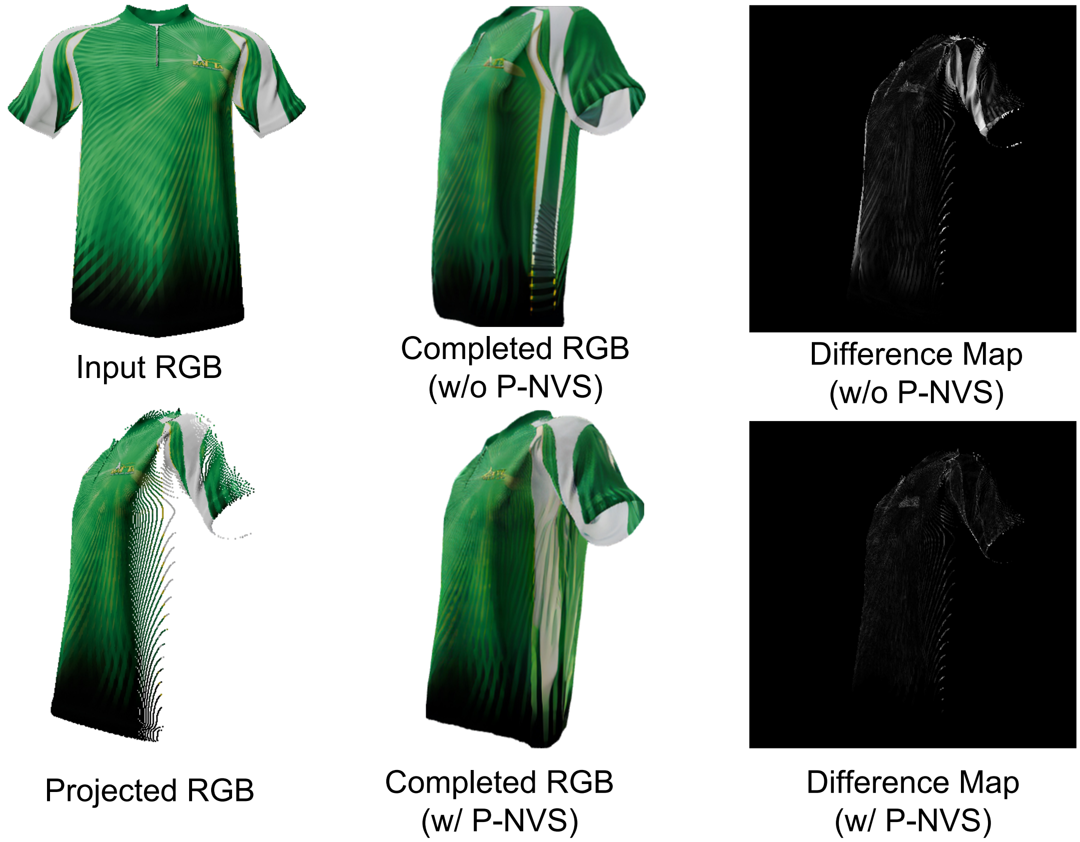
Input RGB
Completed RGB (w/o P-NVS)
Difference Map (w/o P-NVS)
Projected RGB
Completed RGB (w/ P-NVS)
Difference Map (w/ P-NVS)

Figure 2.5: **Analysis of projected image conditioning.** Left: we show original input and projected RGB images. Middle: completed RGB images with and without Progressive Novel View Synthesis (P-NVS). Right: difference between completed and projected images, showing our novel view aligns more closely with the ground-truth projected RGB. Zoom-in for details.

We use the CVCS metric to ablate the impact of P-NVS. As shown in Table 2.2, GarmentCrafter achieves superior cross-view consistency with P-NVS. We further validates this claim with a visual example in Figure 2.5. While both model synthesizes plausible novel views, GarmentCrafter with P-NVS aligns more closely with the input observation.

Effect of Trajectory on Loop Closure. For better loop closure, we use a “zigzag” camera trajectory where we rotate the camera to left and right alternatively and converge at the center back of the garment (see Figure 2.2). This design aims to better capture overlapping views, thereby improving reconstruction accuracy. We validate this design choice by comparing the quality of the 3D meshes generated using zigzag

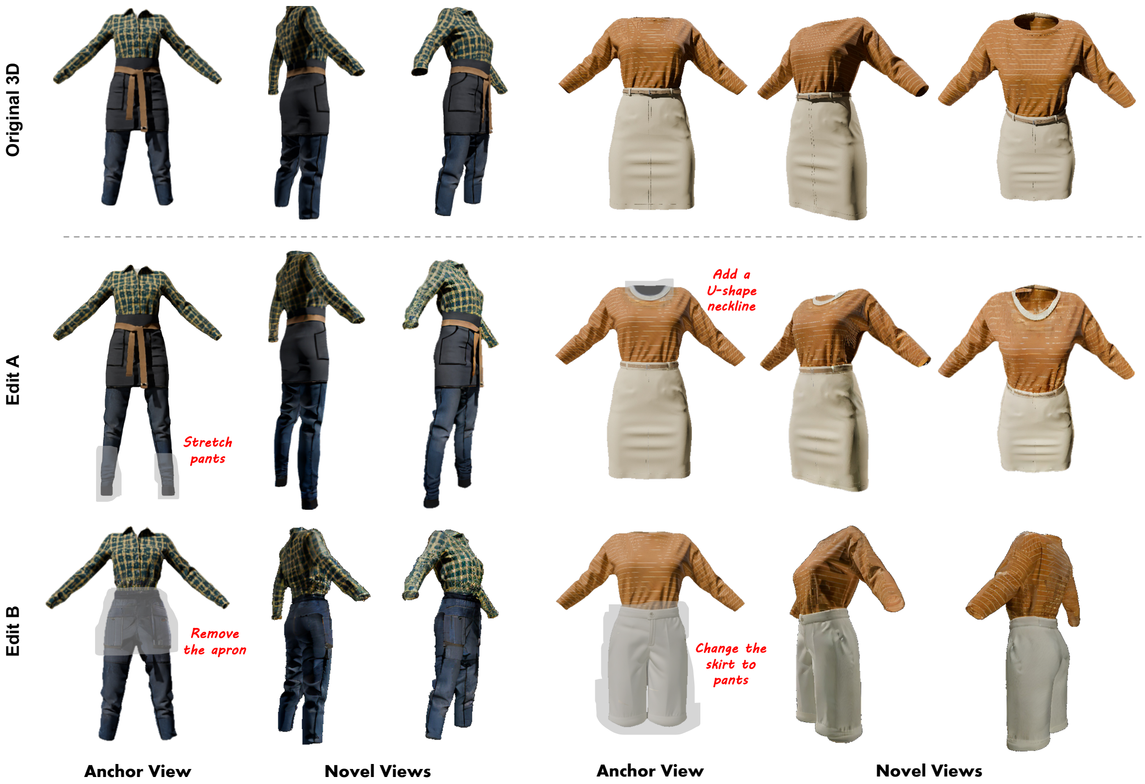

Figure 2.6: **Qualitative results on single-view 3D garment editing.** GarmentCrafter enables single-view edit such as modify the geometry and surface details of the garment, with the changes accurately reflected across the 3D model. Please see supplementary for more results.

and sequential trajectories. We report quantitative results in Table 2.3. We find that our chosen trajectory achieves better performance across both image and geometry metrics. We additionally show a qualitative comparison in Figure 2.7. When using a circular trajectory, achieving loop closure from the side view is challenging; the generated geometry (left sleeve) often conflicts with prior predictions, leading to model failure.

Table 2.3: **Ablation study on camera trajectory selection.** We study two types camera trajectory for progressive novel view synthesis. **Circular**: the camera moves around the object in regular steps, either clockwise or counterclockwise. **Zigzag**: the camera alternates directions with each step, as shown in Figure 2.2. Results indicate that our proposed zigzag achieves better appearance and geometry quality compared to using circular trajectory. We show an actual example in Figure 2.7 for qualitative analyses.

Trajectory	LPIPS ↓	PSNR ↑	SSIM ↑	Chamfer ↓
Circular	0.1503	20.79	0.8130	0.0054
Zigzag (ours)	**0.1454**	**21.22**	**0.8173**	**0.0044**

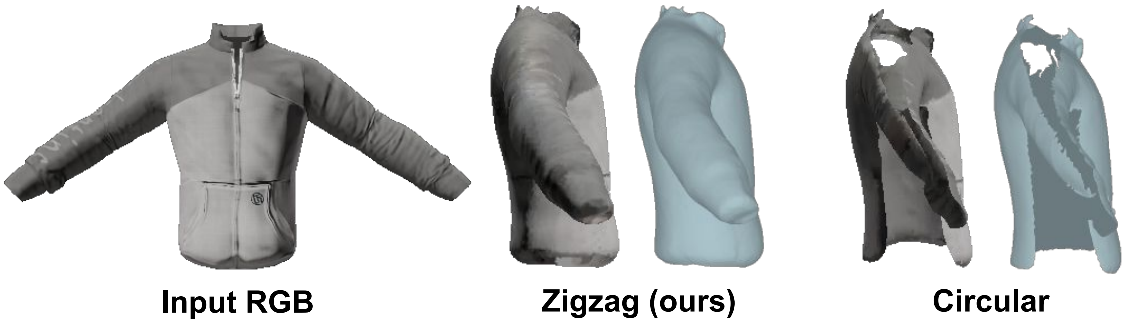

Figure 2.7: **Camera trajectory selection for loop closure.** Zigzag achieves better loop closure, while the circular trajectory struggles with side-view closure, leading to geometric conflicts and model failure. We argue that there are numerous ways to select camera trajectories, our proposed approach just offers an intuitive solution tailored for single-view garment reconstruction and editing.

Chapter 3

Texture Transfer for 3D Garments Generation from Clothing Images

3.1 Introduction

There is an increasing interest to experience apparel in 3D for virtual try-on applications and e-commerce as well as an increasing demand for 3D clothing assets for games, virtual reality and augmented reality applications. While there is an abundance of 2D images of fashion items online, and recent generative AI algorithms democratize the creative generation of such images, the creation of high-quality 3D clothing assets remains a significant challenge. In this work we explore how to transfer the appearance of clothing items from 2D images onto 3D assets, as shown in Figure 3.1.

Extracting the fabric material and prints from such imagery is a challenging task, since the clothing items in the images exhibit strong distortion and shading variation due to wrinkling and the underlying body shape, in addition to general illumination variation and occlusions. To overcome these challenges, we propose a generative approach capable of extracting high-quality physically-based fabric materials and prints from a single input image and transfer them to 3D garment meshes of arbitrary shapes. The result may be rendered using Physically Based Rendering (PBR) to realistically reproduce the garments, for example, in a game engine under novel

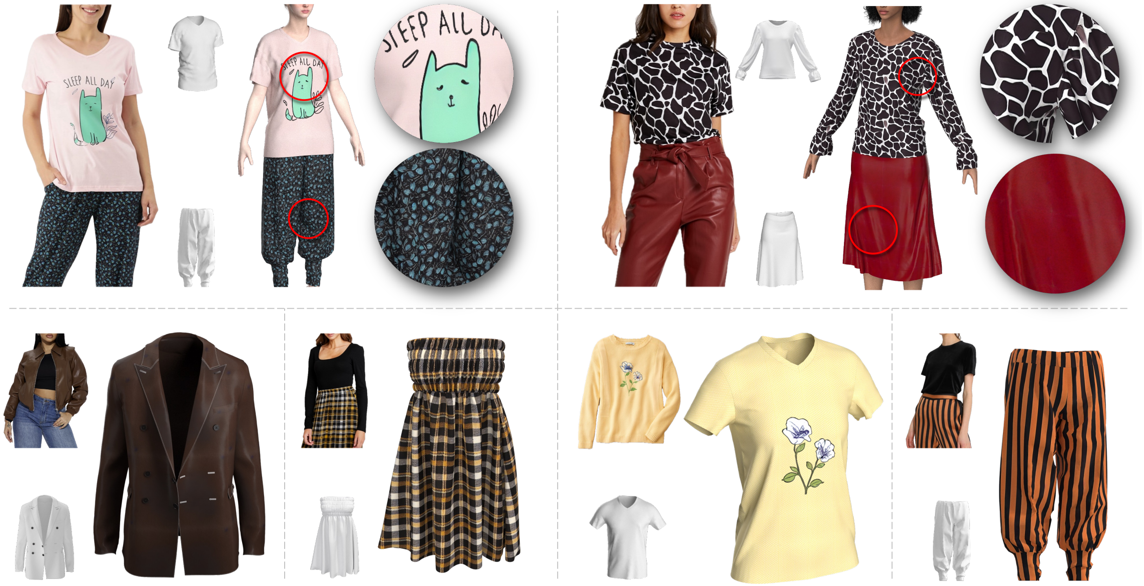

Figure 3.1: Given a real-world 2D clothing image and a raw 3D garment mesh, we propose FabricDiffusion to automatically extract high-quality texture maps and prints from the reference image and transfer them to the target 3D garment surface. Our method can handle different types textures, patterns, and materials. Moreover, FabricDiffusion is capable of generating not only diffuse albedo but also roughness, normal, and metallic texture maps, allowing for accurate relighting and rendering of the produced 3D garment across various lighting conditions.

environment illumination and cloth deformation.

Existing methods for example-based 3D garments texturing primarily focus on direct texture synthesis onto 3D meshes using techniques such as 2D-to-3D texture mapping [30, 68, 71] or multi-view depth-aware inpainting by distilling a pre-trained 2D generative model [79, 122, 124]. However, these approaches often lead to irregular and low-quality textures due to the inherent inaccuracies of 2D-to-3D registration and the stochastic nature of generative processes. Moreover, they struggle to faithfully represent texture details or disentangle garment distortions, resulting in significant degradation in texture continuity and quality.

In this work, we seek to overcome these limitations by drawing inspiration from the real-world garment creation process in the fashion industry [52, 61]: most 3D garments are typically modeled from 2D sewing patterns with normalized[1] and tileable texture maps. This allows us to approach the texturing process from a novel angle, where obtaining such texture maps enables more accurate and realistic garment rendering across various poses and environments. Interestingly, if we take the 3D mesh away from our task of texture transfer, there has been a long history of development in 2D exemplar-based texture map extraction and synthesis [14, 25, 27, 28, 31, 33, 58, 66, 81, 82, 90, 98, 112, 113, 121]. Nevertheless, there remains a significant gap in effectively correcting the geometric distortion or calibrating the appearance (e.g., lighting) of the fabric present in the input reference images.

How can we translate a clothing image to a normalized and tileable texture map? At first glance, solving this ill-posed inverse problem is challenging, and may require developing sophisticated frameworks to model the explicit mapping. Instead, we investigate a feed-forward pathway to simulate the texture distortion and lighting conditions from its normalized form to that on a 3D garment mesh. Then, we propose to train a denoising diffusion model [38, 84] using paired texture images (i.e., both the distorted and normalized) to generate normalized and tileable texture images. Such an objective makes the training procedure fairly straightforward, which we see as a key strength. As a result, generating normalized texture images becomes solving a supervised distribution mapping problem of translating distorted texture patches

[1] We define "normalized" as a canonical texture space devoid of geometric distortions, illumination variations, shadows, and other inconsistencies present in the real-life input images. Terms such as "undistored", "distortion-free", "unwarped", and "flat" are used interchangeably in this paper to describe the textures free from geometric distortions.

back to a unified normalized space.

However, acquiring such paired training data from real clothing at scale is infeasible. To address this issue, we develop a large-scale synthetic dataset comprising over 100k textile color images, 3.8k material PBR texture maps, 7k prints (e.g., logos), and 22 raw 3D garment meshes. These PBR textures and prints are carefully applied to the raw 3D garment meshes and then rendered using PBR techniques under diverse lighting and environmental conditions, simulating real-world scenarios. For each fabric captures from the textured 3D garment, we render a corresponding image using ground-truth PBR textures, which are applied to a flat mesh under a controlled illumination condition, i.e., orthogonal close-up views with a pointed lighting from above. The captured texture inputs along with their ground-truth flat mesh render are used to train our diffusion model. Figure 3.3 illustrates the pipeline of training data construction.

We name our method FabricDiffusion and systematically study the performance on both synthetic data and real-world scenarios. Despite being trained entirely on synthetic rendered examples, FabricDiffusion achieves zero-shot generalization to in-the-wild images with complex textures and prints. Furthermore, the outputs of FabricDiffusion seamlessly integrate with existing PBR material estimation pipelines [89], allowing for accurate relighting of the garment under different lighting conditions. In summary, FabricDiffusion represents a state-of-the-art approach capable of extracting undistorted texture maps from real-world clothing images to produce realistic 3D garments.

3.2 Related Work

Our method built upon recent and seminal work on image-based 3D garment modeling, exemplar-based texture and material extraction, and diffusion-based image generation.

3.2.1 Image-based 3D Garment Modeling

Image-to-mesh texture transfer.

Existing methods on 2D-to-3D texture transfer typically involve (1) learning a 2D-to-3D registration [30, 68, 71] and (2) conducting depth-aware inpainting supervised by a pre-trained image generative model [84] to guarantee multi-view consistency [79, 122, 124, 128]. However, these methods often fail to capture the high frequency details of the texture or leads to irregular textures. In this work, we tackle the problem of texturing 3D garments from a drastically different angle, aiming to extract normalized texture maps from a single real-life clothing image so that we can easily apply them to the 2D UV space (i.e., sewing pattern [52]) of the 3D garment mesh for realistic rendering.

Image-based sewing pattern generation.

We argue that a major cause of the quality gap observed in generated textures is not the capacity of the generation networks, but rather from a suboptimal choice of representations for the texture generation operating from the reference image to the 3D mesh. Unfortunately, there has been little progress in leveraging the idea of generating texture maps that can be used in the 2D UV space, despite the availability of sewing patterns for 3D garments as the sewing pattern can either be manually created by technical artists [61] or automatically reconstructed from reference images [17, 59, 61]. Concurrently, DeepIron [53] is the only work that leverages the similar idea of transferring the texture using sewing pattern representation. Unlike our method, they aim to transfer entire garments without PBR texture maps and exhibits subpar performance in real-world scenarios for practical usages.

3D garment generation.

Recently, there has been growing interest in 3D garment generation using generative models. For instance, GarmentDreamer [55] and WordRobe [95] are recent work that focus on text-based garment generation, whereas our approach transfers textures using image guidance. Another relevant work, Garment3DGen [88], can reconstruct both textures and geometry from a single input image. However, unlike Garment3DGen, our work focuses on generating distortion-free texture and prints and has the additional capability of generating standard PBR materials.

3.2.2 Exemplar-based Texture and Material Extraction

The literature on exemplar-based texture and material extraction is vast. We focus on representative works that are related to ours.

Texture map extraction.

We recast the task of image-to-3D garment texture transfer as generating texture maps from reference clothing image patches. Hao et al. [33] trained a diffusion model to rectify distortions and occlusions in natural texture images. However, it does not extract tileable texture patches or PBR materials for fabrics. More recently, Material Palette [66] addressed a similar problem by using a diffusion-based generative model to extract PBR materials. Their approach relies on personalization methods such as textual inversion [29] to represent the exemplar patch without normalizing the patch into a canonical space, i.e., distortion-free with unified lighting.

Tileable texture synthesis.

Previous work have attempted to synthesize tileable textures with a variety of methods, such as by maximizing perceived texture stationary [73], by using Guided Correspondence [135], by finding repeated patterns in images using pre-trained CNN features [81], by manipulating the latent space of pre-trained GANs [80], or by modifying the noise sampling process of a diffusion model, i.e., rolled-diffusion [102]. We found that a simple circular padding strategy following [133] performs well with our model architecture for addressing tileable texture generation.

BRDF material estimation.

A significant body of research exists on BRDF material estimation from a single image [13, 23, 35, 100, 101, 103]. Our model produces normalized texture maps in a canonical space, enabling compatibility with existing Bidirectional Reflective Distribution Function (BRDF) material estimation pipelines such as MatFusion [89], which can be integrated seamlessly with our output normalized textures. By fine-tuning the pre-trained MatFusion model with fabric PBR texture data and incorporate

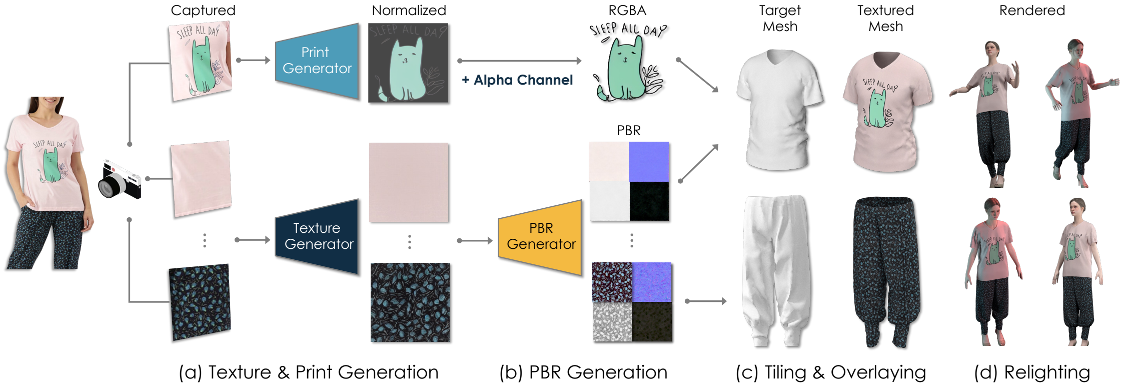

Figure 3.2: Overview of FabricDiffusion. Given a real-life clothing image and region captures of its fabric materials and prints, (a) our model extracts normalized textures and prints, and (b) then generates high-quality Physically-Based Rendering (PBR) materials and transparent prints. (c) These materials and prints can be applied to the target 3D garment meshes of arbitrary shapes (d) for realistic relighting. Our model is trained purely with synthetic data and achieves zero-shot generalization to real-world images.

it into our pipeline, our model generates high-quality material maps for realistic 3D garment rendering.

3.2.3 Diffusion-based Image Generation

Our model architecture is inspired by the recent advancements in diffusion-based image generation models [38, 84, 93]. In this work, we fine-tune the pre-trained image generative model using carefully created synthetic data, enabling texture normalization, which includes distortion removal, lighting calibration, and shadow elimination.

3.3 Method

We propose FabricDiffusion to extract normalized, tileable texture images and materials from a real-world clothing image, and then apply them to the target 3D garment. The overall framework is illustrated in Figure 3.2. We first introduce the problem statement in Section 3.3.1, followed by procedures for constructing synthetic training examples in Section 3.3.2. In Section 3.3.3, we detail our specific approach of texture map generation. Finally, we describe PBR materials generation and garment rendering in Section 3.3.4.

3.3.1 Problem Statement

Given an input clothing image I and a captured texture region x, which may exhibit various distortions and illuminations due to occlusion and poses present in the input image, our goal is learn a mapping function g that takes the captured patch x and outputs the corresponding normalized texture map $\tilde{x}$, effectively correcting the distortions. The texture map $\tilde{x}$ needs to retain the intrinsic properties of the original captured region, such as color, texture pattern, and material characteristics.

As mentioned in Section 3.1, we formulate the generation of normalized texture maps from a real-life clothing patch as a distribution mapping problem. Specifically, the mapping function g can be modeled by a generative process:

$$\tilde{x} \sim G_\theta(x, \epsilon), \epsilon \sim \mathcal{N}(0, \mathbf{I}), \tag{3.1}$$

where the generative model G_θ, parameterized by θ, takes the input patch x as a condition and samples from Gaussian noise to generate the distortion-free texture map $\tilde{x}$ in a canonical space. To train the generator G, we must create a large number of paired training examples (x, x_0) across various types of textures. Here x is the input capture and x_o is the corresponding ground-truth normalized texture. After the model training, we expect to align the sampled output $\tilde{x}$ with the distribution of normalized textures.

3.3.2 Synthetic Paired Training Data Construction

Collecting paired training examples with real clothing poses significant challenges. In contrast, we found that PBR textures — the fundamental unit for appearance modeling in 3D apparel creation — are much more accessible from public sources (see Section 3.4.1 for details on dataset collection). Given these observations, we propose to build synthetic environments for constructing distorted and flat rendered training pairs using the PBR material model [69]. Figure 3.3 illustrates the overall pipeline.

Paired training examples construction.

For each material, we collect the ground-truth diffuse albedo ($k_d \in^3$), normal ($k_n \in^3$), roughness ($k_r \in^2$), and metallic ($k_m \in^2$) material maps. To create distorted rendered images that mimic real-world surface deformation and lighting, we map these material maps onto a raw garment mesh sampled from 22 common garment types. The PBR textures are tiled appropriately and illuminated using four environment maps with white lights to avoid color biases. During rendering, we capture frontal views of the garment and randomly crop patches from the rendered images to match the original fabric texture size.

Separately, we render the same texture material on a plane mesh to create flat rendered images as ground-truths (image x_0 in Figure 3.3). For illumination, we use a fixed point light above the surface center and a fixed orthogonal camera for rendering. This method is highly beneficial as it provides supervision to align the distorted rendered images on the 3D garment to a canonical space of normalized, flat images with a unified lighting condition.

In fact, our flat image rendering and capturing approach may be reminiscent of the input format used in well-known SVBRDF material estimation methods [89, 132, 133, 134], which require orthogonal close-up views of the materials and/or a flashing image as input. As will be described in Section 3.3.4, the output normalized textures from our method can be effectively integrated with SVBRDF material estimation models to generate high-quality PBR material maps.

Paired prints (e.g., logos) construction.

In additional to general textures, we aim to transfer clothing details by creating warped and flat pairs of print images. We map the print to a random location on the garment mesh and blend it with a uniformly colored background texture. Unlike flat texture generation on a plane mesh, we use the original print image with a transparent background as the flat image.

Scaling up training data with Pseudo-BRDF materials.

While the texture material maps are easier to acquire than real clothing, we raise the question: Do we really need a large amount of real BRDF material maps for paired

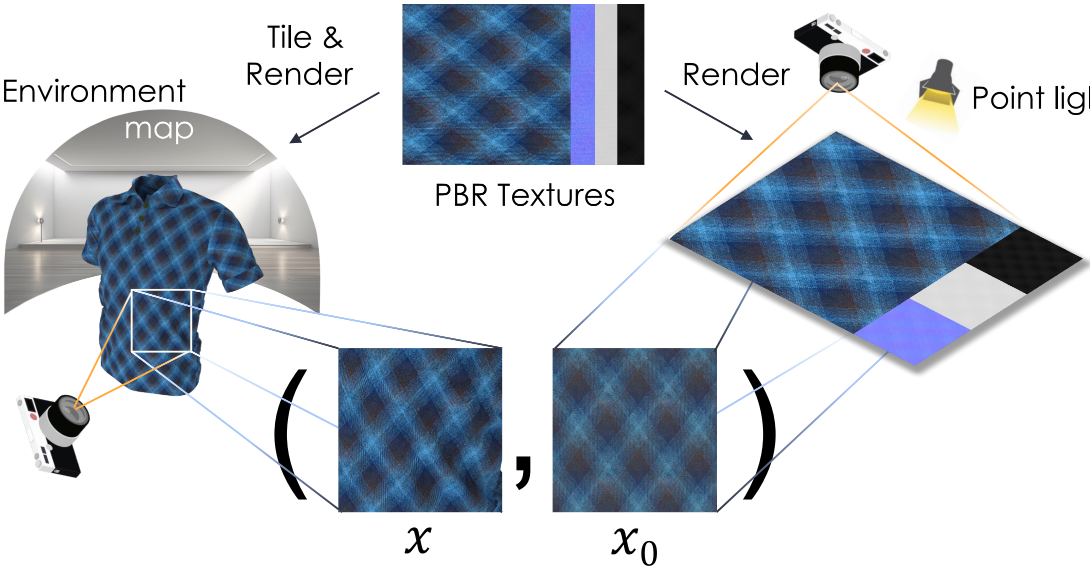

Figure 3.3: Pipeline of paired training data construction. Given the textures of a PBR material, we apply them to both the target raw 3D garment mesh and the plain mesh. The 3D garment is rendered using an environment map, while the plain mesh is illuminated using a point light from above. The resulting rendered images (x, x_0) from both meshes serve as the paired training examples for training our texture generative model (Section 3.3.2).

training data construction, and what if we cannot obtain enough data?

In this work, we are able to collect a BRDF dataset comprises 3.8k assets in total (see Section 3.4.1 for details), covering a broad spectrum of fabric materials. However, the texture patterns in this dataset exhibit limited diversity because it is not large enough to model the appearance of fabric textures in our real life, given the vast range of colors, patterns, and materials. To address this, we augmented the dataset by gathering 100k textile color images featuring a wide array of patterns and designs, which are then used to generate pseudo-BRDF[2] materials. Specifically, the color image served as the albedo map, while the roughness map was assigned a uniform value α sampled from the distribution $\mathcal{N}(0.708, 0.193^2)$, with 0.708 and 0.193 representing the population mean and standard deviation of the mean roughness values of the real BRDF dataset, respectively. The metallic map was assigned a uniform value $\max(\beta, 0)$, where $\beta \sim \mathcal{U}(-0.05, 0.05)$, and the normal map was kept

[2] Since the normal, roughness, and metallic maps of the 100k textile images are sampled instead of ground truth, they are referred to as pseudo-BRDF data.

flat.

We use a combination of real (3.8k) and pseudo-BRDF (100k) materials to create paired rendered images for training our texture generation model. During paired training examples construction, both real and pseudo-BRDF have x and x_0 (as illustrated in Figure 3.3), representing distorted and flat textures, respectively. Intuitively, the primary goal of our texture generator is to eliminate geometric distortions, and our generated pseudo rendered images, serve this purpose effectively.

3.3.3 Normalized Texture Generation via FabricDiffusion

Given the paired training images, we build a denoising diffusion model to learn the distribution mapping from the input capture to the normalized texture map. Next, we detail our training objective, model architecture and training, and the design for tileable texture generation and alpha-channel-enabled[3] prints generation.

Training objective of conditional diffusion model.

Diffusion models [38, 93] are trained to capture the distribution of training images through a sequential Markov chains of adding random noise into clean images and denoising pure noise to clean images. We leverage Latent Diffusion Model (LDM) [84] to improve the efficiency and quality of diffusion models by operating in the latent space of a pre-trained variational autoencoder [50] with encoder $\mathcal{E}$ and decoder $\mathcal{D}$. In our case, given the paired training data (x, x_0), where x is the distorted patch and x_0 is the normalized texture, the feed-forward process is formulated by adding random Gaussian noise into the latent space of image x_0:

$$x_t = \sqrt{\gamma(t)}\mathcal{E}(x_0) + \sqrt{1-\gamma(t)}\epsilon, \tag{3.2}$$

where x_t is a noisy latent of the original clean input x_0, $\epsilon \sim \mathcal{N}(0, \mathbf{I})$, $t \in [0, 1]$, and $\gamma(t)$ is defined as a noise scheduler that monotonically descends from 1 to 0. By adding the distorted image x as the condition, the reverse process aims to denoise Gaussian noises back to clean images by iteratively predicting the added noises at

[3] Alpha-channel-enabled prints are images with transparency that can be overlaid onto existing images for realistic composition and rendering.

each reverse step. We minimize the following latent diffusion objective:

$$L(\theta) = \mathbb{E}_{\mathcal{E}(x),\epsilon\sim\mathcal{N}(0,\mathbf{I}),t}\left[\|\epsilon - \epsilon_\theta(x_t, t, \mathcal{E}(x))\|^2\right], \tag{3.3}$$

where ϵ_θ denotes model parameterized by a neural network, x_t is the noisy latent for each timestep t, and $\mathcal{E}(x)$ is the condition.

Recalling Equation 3.1, the above formulation incorporates input-specific information (i.e., the captured patch x) into the training process for generating normalized textures. As will be shown in the experimental results in Section 3.4.2, this design is the key to producing faithful texture maps that differs from existing per-example optimization-based texture extraction approaches [66, 79].

Model architecture and training.

Any diffusion-based architecture for conditional image generation can realize Equation 3.3. Specifically, we use Stable Diffusion [84], a popular open-source text-conditioned image generative model pre-trained on large-scale text and image pairs. To support image conditioning, we use additional input channels to the first convolutional layer, where the latent noise x_t is concatenated with the conditioned image latent $\mathcal{E}(x)$. The model's initial weights come from the pre-trained Stable Diffusion v1.5, while the newly added channels are initialized to zero, speeding up training and convergence. We eliminate text conditioning, focusing solely on using a single image as the prompt. This approach addresses the challenge of generating normalized texture maps, which text prompts struggle to describe accurately [24].

Circular padding for seamless texture generation.

To ensure the generated texture maps are tileable, we employ a simple yet effective circular padding strategy inspired by TileGen [133]. Unlike TileGen, which uses a StyleGAN-like architecture [46] and needs to replace both regular and transposed (e.g., upsampling or downsampling) convolutions, we only apply circular padding to all regular convolutional layers, thanks to the flexibility of diffusion models.

Transparent prints generation.

The vanilla Stable Diffusion model can only output RGB images, lacking the capability to generate layered or transparent images, which is in stark contrast to our demand for prints transfer. Instead of redesigning the existing generative model [126], we propose a simple and effective recipe to post-process the generated RGB print images for computing an additional alpha channel. We hypothesize that the alpha map for prints can be approximated as binary – either fully transparent or fully opaque. Based on this assumption, we assign a new RGB value for each pixel (i, j) as follows:

$$\mathrm{RGB}(i, j) = \max\left[0, \frac{\tilde{x}(i, j) - 0.1}{0.9}\right], \tag{3.4}$$

where $\tilde{x}$ is the generated texture (Equation 3.1). The alpha channel value at each pixel (i, j) is thus determined by the following criteria:

$$\mathrm{A}(i, j) = \begin{cases} 1 & \text{if } \tilde{x}(i, j) \geq 0.1, \\ \tilde{x}(i, j)/0.1 & \text{otherwise.} \end{cases} \tag{3.5}$$

This approach assigns full opacity (alpha value of 1) to pixels where the initial value exceeds a certain threshold, and scales down the alpha value for other pixels, designating them as transparent background. As will be shown in Section 3.4.2 and Figure 3.5, our method can handle complex prints and logos and output RGBA print images that can be overlaid onto the fabric texture.

3.3.4 PBR Materials Generation and Garment Rendering

Our FabricDiffusion model is able to generate a normalized texture map that is tileable, flat, and under a unified lighting, ensuring compatibility with the SVBRDF material estimation method. The goal of this work is not to develop a new material estimation method but to demonstrate the compatibility of our approach with existing methods. MatFusion [89] is a state-of-the-art model trained on approximately 312k SVBRDF maps, most of which are non-fabric or non-clothing materials. We fine-tune this model using our dataset of real fabric BRDF materials. Specifically, we use our normalized textures as inputs, with the material maps (k_d, k_n, k_r, k_m) as ground-truths for model fine-tuning.

The generated PBR material maps can be used for tiling in the garment sewing pattern. The remaining question is how to determine the scale for tiling? We consider two specific strategies: (1) Proportion-aware tiling. We use image segmentation to calculate the proportion of the caputured region relative to the segmented clothing, maintaining a similar ratio when tiling the generated texture onto the sewing pattern. (2) User-guided tiling. We emphasize that an end-to-end automatic tilling method may not be optimal, as user involvement is often necessary to resolve ambiguities and provide flexibility in fashion industries.

3.4 Experiments

We validate FabricDiffusion with both synthetic data and real-world images across various scenarios. We begin by introducing the experimental setup in Section 3.4.1, followed by detailing the experimental results in Section 3.4.2. Finally, we conduct ablation studies and show several real-world applications in Section 3.4.3.

3.4.1 Setup

Dataset.

We detail the process of collecting BRDF texture, print, and garment datasets. (1) Fabric BRDF dataset. This dataset includes 3.8k real fabric materials and 100k pseudo-BRDF textures (RGB only). We reserved 200 real BRDF materials for testing the PBR generator and 800 pseudo-BRDF materials (combined with the 200 real materials) for testing the texture generator. (2) 3D garment dataset. We collected 22 3D garment meshes for training and 5 for testing. Using the method in Section 3.3.2, we created 220k flat and distorted rendered image pairs for training and 5k pairs for testing. (3) Logos and prints dataset. This dataset contains 7k prints and logos in PNG format. We generated pseudo-BRDF materials with specific roughness and metallic values and a flat normal map. Dark prints were converted to white if necessary. By compositing these onto 3D garments, we produced 82k warped print images.

Evaluation protocols and tasks.

We compare FabricDiffusion to state-of-the-art methods on two tasks: (1) Image-to-garment texture transfer. Our ultimate goal is to transfer the textures and prints from the reference image to the target garment. We evaluate FabricDiffusion and compare it to baseline methods using both synthetic and real-world test examples. (2) PBR materials extraction. We provide both quantitative and qualitative results on PBR materials estimation using our testing BRDF materials dataset.

Evaluation metrics

We evaluate the quality of generated textures and garments using commonly used metrics: LPIPS [127] , SSIM [110], MS-SSIM [109], DISTS [26], and FLIP [6]. To evaluate the tileability of the generated textures, we adopt the metric proposed by TexTile [83]. For the image-to-garment texture transfer task, we additionally report FID [36] and CLIP-score in CLIP image feature space [29, 76] to evaluate the visual similarity of the textured garment with the original input clothing.

Baseline methods.

We compare with state-of-the-art methods that support image-to-mesh texture transfer, including: (1) TEXTure [79], the most representative method for texturing a 3D mesh based on a small set of sample images through per-subject optimization (i.e., textual inversion [29] for personalization). (2) Material Palette [66], which focuses on texture extraction and PBR materials estimation from a single image using generative models. (3) MatFusion [89], for PBR materials estimation for general materials, not specifically fabric or clothing. We fine-tuned the pre-trained MatFusion model with our curated fabric BRDF training examples, resulting in improved performance.

3.4.2 Experimental Results

FabricDiffusion on real-world clothing images.

We first show the results of our method on real-world images in Figure 3.4. Our method effectively transfers both texture patterns and material properties from

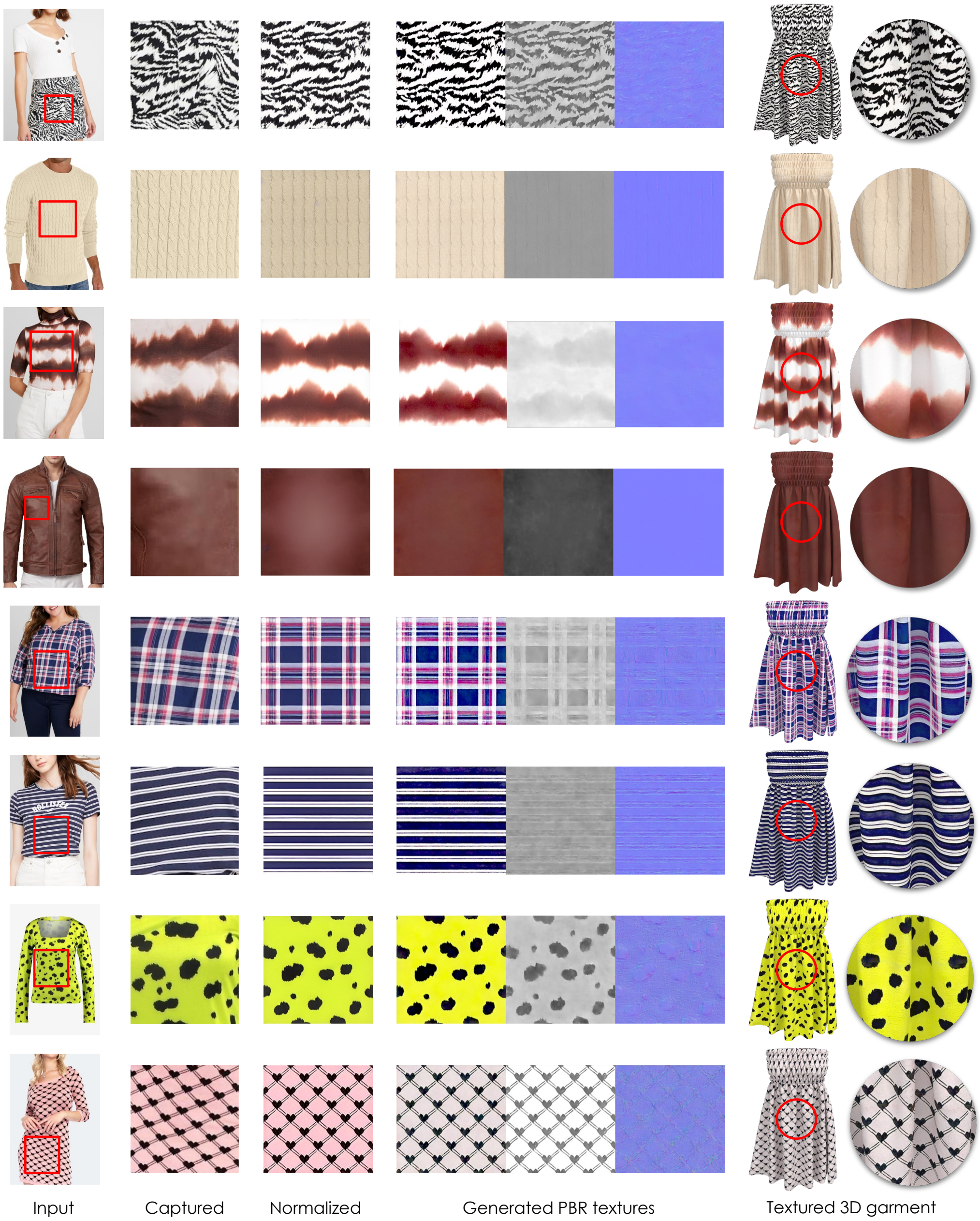

Figure 3.4: Results on texture transfer on real-world clothing images. Our method can handle real-world garment images to generate normalized texture maps, along with the corresponding PBR materials. The PBR maps can be applied to the 3D garment for realistic relighting and rendering.

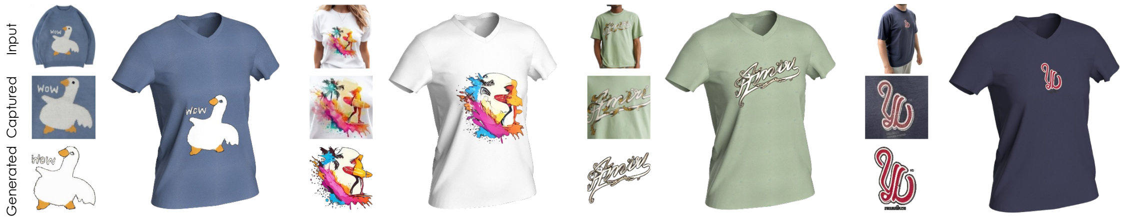

Figure 3.5: Results on prints and logos transfer on real-world images. Given a real-life garment image with prints and/or logos, and the cropped patch of the region where the print is located. Our method generates a distortion-free and transparent print element, which can be applied to the target 3D garment for realistic rendering. Note that the background texture is transferred using our method as well.

various types of clothing to the target 3D garment. Notably, our method is capable of recovering challenging materials such as knit, translucent fabric, and leather. We attribute this success to our construction of paired training examples that seamlessly couples the PBR generator with the upstream texture generator. Since we focus on non-metallic fabrics, the metallic map is omitted in the visualizations in the section. Please be referred to Appendix for more details and results.

FabricDiffusion on detailed prints and logos.

In addition to texture patterns and material properties, our FabricDiffusion model can transfer detailed prints and logos. Figure 3.5 shows some examples. We highlight two key advantages of our design that benefit the recovery of prints and logos. First, our conditional generative model corrects geometry distortion caused by human pose or camera perspective. Second, as detailed in Section 3.3.3, our method can generate prints with a transparent background, enabling practical usage in garment appearance modeling.

Image-to-garment texture transfer.

In Figure 3.6, we compare our method with Material Palette [66] and TEXTure [79] for image-to-garment texture transfer. We present the results on real-world clothing images featuring a variety of textures, ranging from micro to macro patterns and prints. Our observations indicate that FabricDiffusion not only recovers repetitive patterns, such as scattered stars or camouflage, but also maintains the regularity of

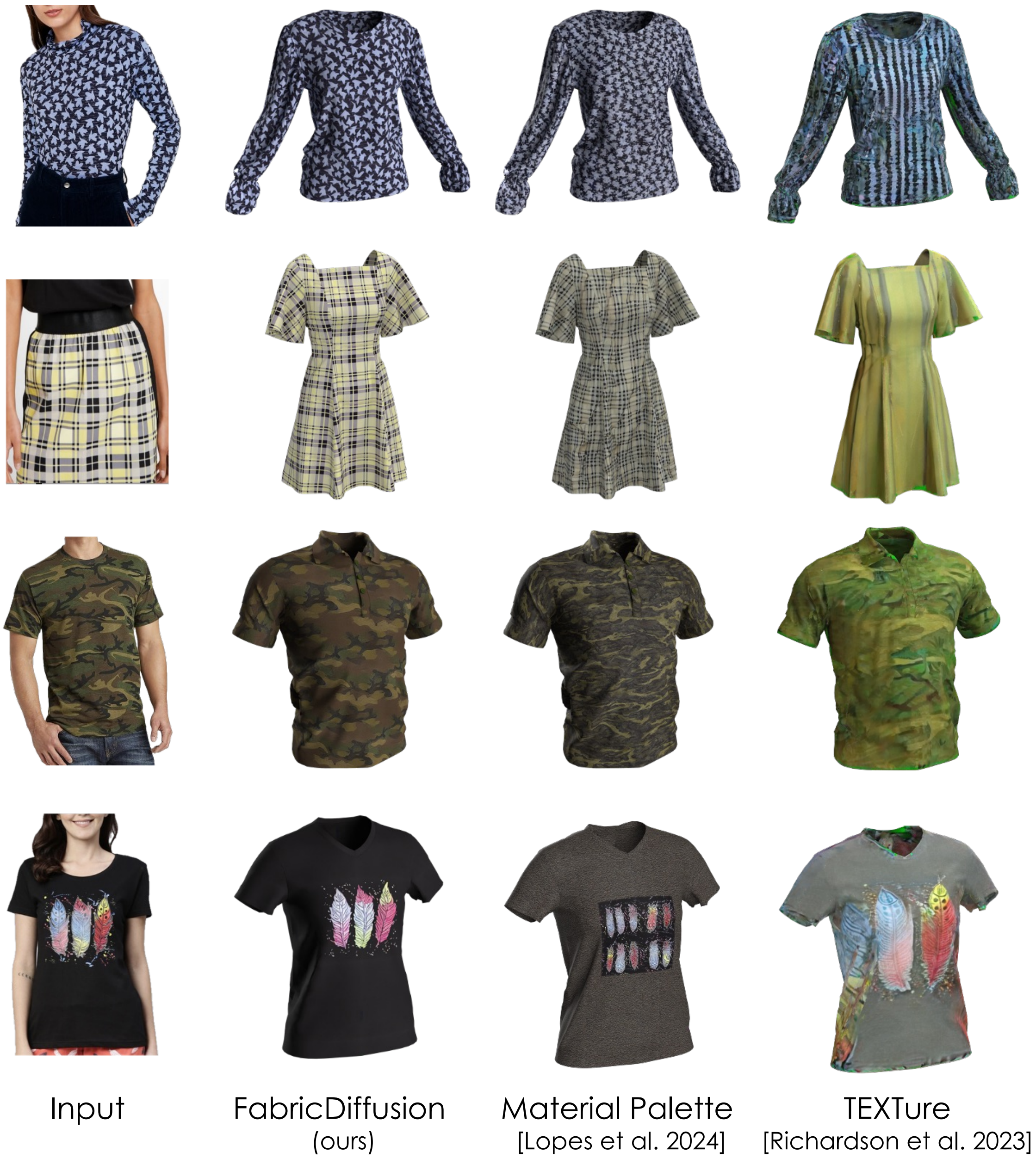

Figure 3.6: Comparison on image-to-garment texture transfer. FabricDiffusion faithfully captures and preserves the texture pattern from the input clothing. We observe texture irregularities and artifacts for Material Palette [66] and TEXTure [79].

Table 3.1: Quantitative comparison on image-to-garment clothing texture transfer. Performances evaluated on synthetic testing data. Our method succeeds at faithfully extracting and transferring textures from images, whereas Material Palette [66] exhibits significant artifacts, resulting in suboptimal performance, particularly on FID.

	FID ↓	LPIPS↓	SSIM↑	MS-SSIM↑	DISTS↓	CLIP-s↑
Material Palette	34.39	0.20	0.75	0.73	0.28	0.94
FabricDiffusion (ours)	**12.44**	**0.16**	**0.79**	**0.77**	**0.19**	**0.97**

Table 3.2: Quantitative comparison with state-of-the-art methods on PBR material extraction. Results are evaluated on the real PBR test examples. By fine-tuning MatFusion with additional fabric PBR training data, our method achieves superior performance across most material maps. Material Palette performs worse, particularly in estimating the diffuse and roughness maps, due the differences in physical properties between fabric materials and general objects. Please see Table 3.3 for quantitative evaluation on rendered images and Figure 3.7 for a qualitative comparison between and Material Palette.

	MSE↓			SSIM↑		
	Diff.	Norm.	Rough.	Diff.	Norm.	Rough.
Material Palette	0.0515	0.0136	0.1287	0.2213	0.3028	0.2920
MatFusion	0.0896	0.0127	0.0806	0.2190	**0.3902**	0.4922
FabricDiffusion (ours)	**0.0287**	**0.0094**	**0.0559**	**0.3157**	0.3827	**0.5039**

structured patterns, like the plaid on a skirt. Please refer to Table 3.1 for quantitative results.

PBR materials extraction.

We compare our method to Material Palette [66] and MatFusion [89] on PBR materials extraction. In Table 3.2, we present a comparison of pixel-level MSE and SSIM between the generated material maps and the ground-truths. Our FabricDiffusion material generator, fine-tuned from the base MatFusion model with additional fabric BRDF training examples, demonstrates superior performance. Additionally, Figure 3.7 shows visual comparisons between FabricDiffusion and Material Palette. While Material Palette [66] struggles to accurately capture fabric materials, our FabricDiffusion model excels in recovering the physical properties for fabric textures, particularly in roughness and diffuse maps. We also evaluate different methods on the rendered images and show the results in Table 3.3. Particularly, we use render-aware metrics like FLIP [6] and perceptual metrics like LPIPS and DISTS. FabricDiffusion consistently

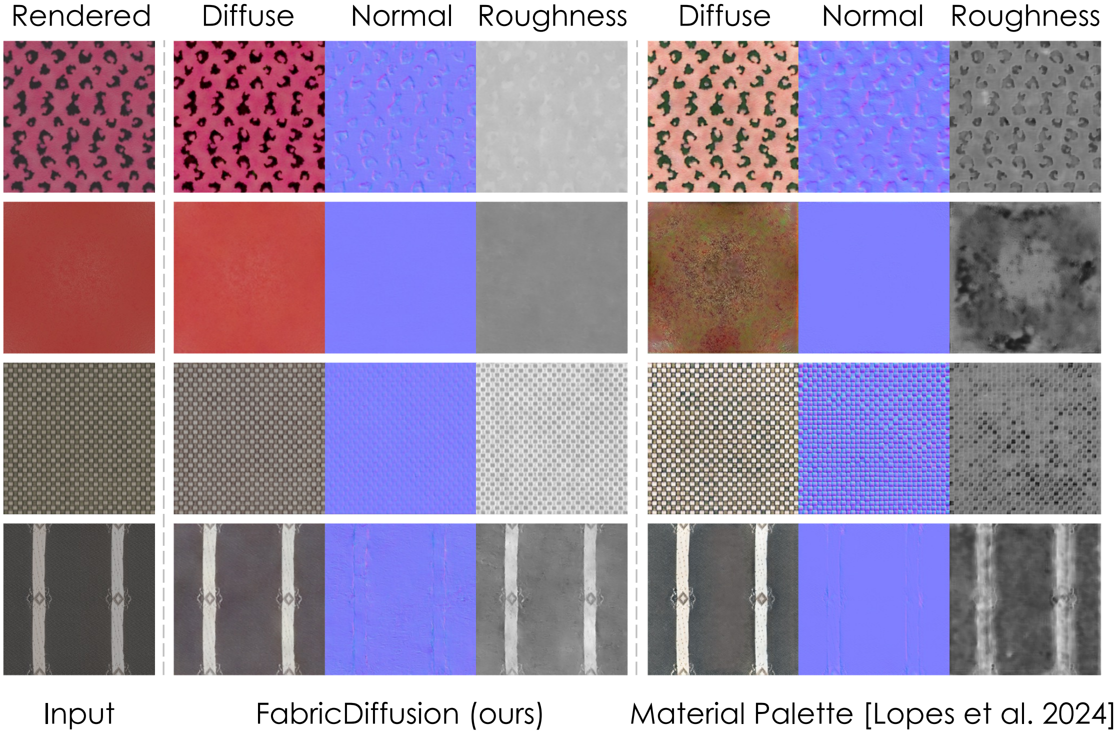

Figure 3.7: Qualitative comparison on PBR materials extraction. Material Palette [66] can hardly capture fabric materials while our FabricDiffusion model is capable of recovering physical properties for fabric textures especially on roughness and diffuse maps.

achieve better performance over other approaches.

3.4.3 Ablations, Analyses, and Applications

Ablation on circular padding and tileability analysis.

We conduct an ablation study to evaluate the impact of circular padding using the TexTile metric [83], where higher values indicate better tileability. The results show that the MaterialPalette [66] achieves a score of 0.54. Our method without circular padding scores 0.47, while with circular padding, our method improves significantly, reaching a score of 0.62.

Table 3.3: Quantitative comparison on rendered materials. We adopt render-aware and perceptual metrics and compare the quality of rendered generated texture. FabricDiffusion outperforms other methods.

	MSE↓	SSIM↑	DISTS↓	LPIPS↓	FLIP↓
Material Palette	0.0531	0.2838	0.3388	0.4463	0.5812
MatFusion	0.1032	0.3233	0.3790	0.5697	0.7009
FabricDiffusion (ours)	**0.0284**	**0.4102**	**0.3035**	**0.3836**	**0.4411**

Ablation on pseudo-BRDF data.

We compare the performance of using combined real-BRDF and pseudo-BRDF data versus using only real-BRDF data. The results, summarized in Table 3.4, demonstrate that the inclusion of pseudo-BRDF data alongside real-BRDF data improves performance across all metrics.

Effect of the capture location.

In Section 3.3.4, we explored how FabricDiffusion can be integrated into an end-to-end framework for 3D garment design. To assess whether the generated texture remains consistent with the input, Figure 3.8-(a) shows the results of varying the location of a fixed-size capture region. The results indicate that FabricDiffusion consistently produces similar texture patterns, regardless of the location of the captured region.

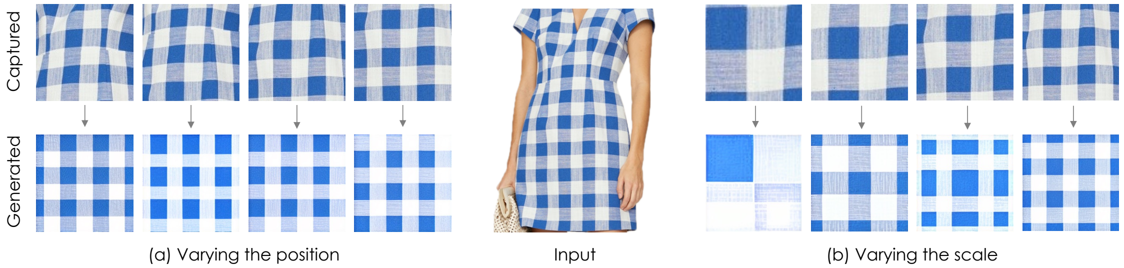

Figure 3.8: Ablation study on varying the position and scale of the captured texture. Given an input clothing image, we evaluate (a) varying the position with a fixed capture size and (b) varying the scale for texture extraction. Our method successfully recovers the input texture despite variation in the location or resolution of the captured image. Since we care about distributions, none of the generated images are cherry- or lemon-pick.

Table 3.4: Ablation study on pseudo-BRDF data. We compare the performance of using combined versus only real-BRDF data. Combined data effectively improve the performance.

Real-BRDF	Pseudo-BRDF	FID↓	LPIPS↓	DISTS↓	CLIP-s↑
✓		19.17	0.19	0.26	0.96
✓	✓	**12.44**	**0.16**	**0.19**	**0.97**

Effect of the capture scale.

In Figure 3.8-(b), we further study the effect of the size of the captured region. By varying the scale of the captured region, FabricDiffusion recovers the texture pattern from the input patch, demonstrating robustness to changes in resolution.

Multi-material texture transfer.

Since FabricDiffusion works on patches, it can be applied to multi-material garments as well as evidenced in Figure 3.10. This suggests that FabricDiffusion can serve as a basic building block for multi-material garment texture transfer.

Compatibility with AI-Generated Images.

We explore the possibility of enhancing FabricDiffusion with AI-generated images and demonstrate the results in Figure 3.9. In addition to real-life clothing, we use an advanced text-to-image model to create apparel images and the apply FabricDiffusion to transfer their textures to the target 3D garments. This opens up new creative possibilities for designers, allowing them to envision and materialize entirely novel textures and patterns through simple text descriptions.

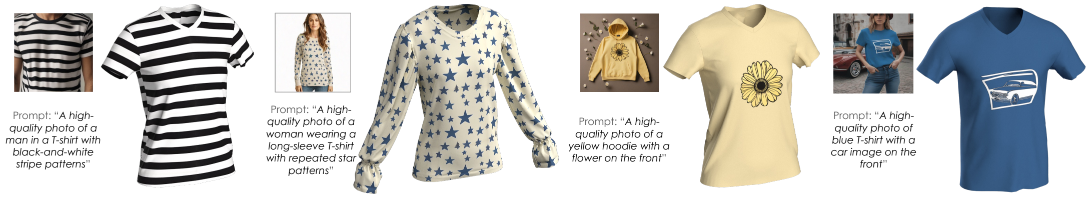
Prompt: "A high-quality photo of a man in a T-shirt with black-and-white stripe patterns"
Prompt: "A high-quality photo of a woman wearing a long-sleeve T-shirt with repeated star patterns"
Prompt: "A high-quality photo of a yellow hoodie with a flower on the front"
Prompt: "A high-quality photo of blue T-shirt with a car image on the front"

Figure 3.9: Compatibility with generative apparel. FabricDiffusion can extract the textures from the output image of a text-to-image generative model and apply them to a target 3D garment of arbitrary shapes. We highlight that our method can handle imperfect textures, such as the broken black stripes in the first example. For each example, we show the input text prompt (bottom-left), the generated 2D image by Stable Diffusion XL (top-left), and the textured 3D garment (right) created by our FabricDiffusion method.

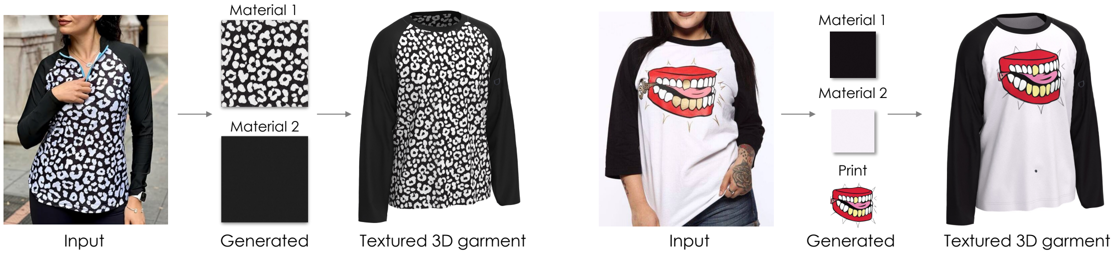

Figure 3.10: Multi-material textures transfer. Given a clothing image containing multiple texture patterns, materials, and prints, FabricDiffusion can transfer each distinct element to separate regions of the target 3D garment.

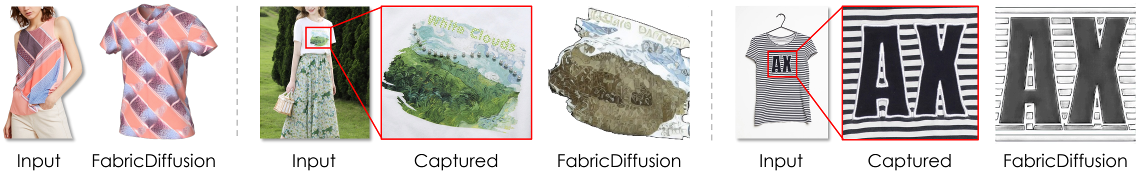

Figure 3.11: Limitations of FabricDiffusion. Our method may struggle to reconstruct specific inputs such as complex (e.g., non-repetitive) patterns (left), fine details in complex prints (middle), and prints over non uniform fabric (right).

Chapter 4

Conclusions and Future Work

This thesis presents novel approaches for democratizing 3D garment creation by leveraging large-scale vision foundation models. We address two fundamental challenges: (1) reconstructing and editing 3D garments from a single-view image and (2) transferring textures from in-the-wild clothing images to arbitrary 3D garment models. To this end, we propose GarmentCrafter, a method that ensures cross-view consistency in single-view 3D garment reconstruction and editing, and FabricDiffusion, a framework for extracting distortion-free textures and mapping them seamlessly onto 3D garments. These contributions significantly enhance the accessibility of 3D fashion design, allowing non-expert users to generate and modify high-quality 3D clothing assets with minimal effort.

Through extensive experiments, we demonstrate that our methods significantly outperform state-of-the-art approaches in terms of geometric and texture quality. Our models generalize well across diverse garment styles, texture patterns, and material types, showcasing their applicability in virtual fashion, gaming, and augmented/virtual reality environments.

Nevertheless, the proposed methods have certain limitations. GarmentCrafter is designed for garments in a rest pose and does not generalize to arbitrary deformations or dynamic motion. Additionally, it reconstructs only the external surface, without capturing inner layers or underlying structures. FabricDiffusion may introduce errors when reconstructing non-repetitive patterns and struggles to preserve fine details in complex prints or logos, particularly when dealing with highly intricate designs, non-uniform backgrounds, or severe distortions. Future work could explore solutions to these challenges, such as incorporating pose-aware reconstruction, multi-layer garment modeling, and enhanced texture synthesis techniques for greater accuracy and generalization.

Bibliography

[1] CLO3D. https://www.clo3d.com/en/. 2.1

[2] Style3D. https://www.linctex.com/. 2.1

[3] TUKA3D. https://tukatech.com/tuka3d/. 2.1

[4] Browzwear. https://browzwear.com/. 2.1

[5] Thiemo Alldieck, Mihai Zanfir, and Cristian Sminchisescu. Photorealistic monocular 3d reconstruction of humans wearing clothing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1506–1515, 2022. 2.2

[6] Pontus Andersson, Jim Nilsson, Tomas Akenine-Möller, Magnus Oskarsson, Kalle Åström, and Mark D Fairchild. Flip: A difference evaluator for alternating images. *Proc. ACM Comput. Graph. Interact. Tech.*, 3(2):15–1, 2020. 3.4.1, 3.4.2

[7] Floraine Berthouzoz, Akash Garg, Danny M Kaufman, Eitan Grinspun, and Maneesh Agrawala. Parsing sewing patterns into 3d garments. *Acm Transactions on Graphics (TOG)*, 32(4):1–12, 2013. 2.2

[8] Bharat Lal Bhatnagar, Garvita Tiwari, Christian Theobalt, and Gerard Pons-Moll. Multi-garment net: Learning to dress 3d people from images. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5420–5430, 2019. 2.1, 2.2

[9] Michael J Black, Priyanka Patel, Joachim Tesch, and Jinlong Yang. Bedlam: A synthetic dataset of bodies exhibiting detailed lifelike animated motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8726–8737, 2023. 2.1, 2.2, 2.4.1

[10] 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. 2.1

[11] Shengqu Cai, Eric Ryan Chan, Songyou Peng, Mohamad Shahbazi, Anton

Obukhov, Luc Van Gool, and Gordon Wetzstein. Diffdreamer: Towards consistent unsupervised single-view scene extrapolation with conditional diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2139–2150, 2023. 2.2

[12] Andrés Casado-Elvira, Marc Comino Trinidad, and Dan Casas. Pergamo: Personalized 3d garments from monocular video. In *Computer Graphics Forum*, volume 41, pages 293–304. Wiley Online Library, 2022. 2.1

[13] Dan Casas and Marc Comino-Trinidad. Smplitex: A generative model and dataset for 3d human texture estimation from single image. *arXiv preprint arXiv:2309.01855*, 2023. 3.2.2

[14] George Cazenavette, Tongzhou Wang, Antonio Torralba, Alexei A Efros, and Jun-Yan Zhu. Wearable imagenet: Synthesizing tileable textures via dataset distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2278–2282, 2022. 3.1

[15] Cheng-Hsiu Chen, Jheng-Wei Su, Min-Chun Hu, Chih-Yuan Yao, and Hung-Kuo Chu. Panelformer: Sewing pattern reconstruction from 2d garment images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 454–463, 2024. 2.2

[16] Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 22246–22256, 2023. 2.2

[17] Xipeng Chen, Guangrun Wang, Dizhong Zhu, Xiaodan Liang, Philip Torr, and Liang Lin. Structure-preserving 3d garment modeling with neural sewing machines. *Advances in Neural Information Processing Systems*, 35:15147–15159, 2022. 2.2, 3.2.1

[18] Jaeyoung Chung, Suyoung Lee, Hyeongjin Nam, Jaerin Lee, and Kyoung Mu Lee. Luciddreamer: Domain-free generation of 3d gaussian splatting scenes. *arXiv preprint arXiv:2311.13384*, 2023. 2.2

[19] Enric Corona, Albert Pumarola, Guillem Alenya, Gerard Pons-Moll, and Francesc Moreno-Noguer. Smplicit: Topology-aware generative model for clothed people. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11875–11885, 2021. 2.1, 2.2

[20] R Daněřek, Endri Dibra, Cengiz Öztireli, Remo Ziegler, and Markus Gross. Deepgarment: 3d garment shape estimation from a single image. In *Computer Graphics Forum*, volume 36, pages 269–280. Wiley Online Library, 2017. 2.2

[21] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali

Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13142–13153, 2023. 2.1, 2.2, 2.4.1

[22] Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, et al. Objaverse-xl: A universe of 10m+ 3d objects. *Advances in Neural Information Processing Systems*, 36, 2024. 2.2

[23] Valentin Deschaintre, Miika Aittala, Fredo Durand, George Drettakis, and Adrien Bousseau. Single-image svbrdf capture with a rendering-aware deep network. *ACM Transactions on Graphics (ToG)*, 37(4):1–15, 2018. 3.2.2

[24] Valentin Deschaintre, Diego Gutierrez, Tamy Boubekeur, Julia Guerrero-Viu, and Belen Masia. The visual language of fabrics. Technical report, 2023. 3.3.3

[25] Olga Diamanti, Connelly Barnes, Sylvain Paris, Eli Shechtman, and Olga Sorkine-Hornung. Synthesis of complex image appearance from limited exemplars. *ACM Transactions on Graphics (TOG)*, 34(2):1–14, 2015. 3.1

[26] Keyan Ding, Kede Ma, Shiqi Wang, and Eero P Simoncelli. Image quality assessment: Unifying structure and texture similarity. *IEEE transactions on pattern analysis and machine intelligence*, 44(5):2567–2581, 2020. 3.4.1

[27] Alexei A Efros and William T Freeman. Image quilting for texture synthesis and transfer. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 571–576. 2023. 3.1

[28] Alexei A Efros and Thomas K Leung. Texture synthesis by non-parametric sampling. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1033–1038, 1999. 3.1

[29] 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. *arXiv preprint arXiv:2208.01618*, 2022. 3.2.2, 3.4.1, 3.4.1

[30] Daiheng Gao, Xu Chen, Xindi Zhang, Qi Wang, Ke Sun, Bang Zhang, Liefeng Bo, and Qixing Huang. Cloth2tex: A customized cloth texture generation pipeline for 3d virtual try-on. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2024. 1, 2.1, 2.2, 3.1, 3.2.1

[31] Giuseppe Claudio Guarnera, Peter Hall, Alain Chesnais, and Mashhuda Glencross. Woven fabric model creation from a single image. *ACM Transactions on Graphics (TOG)*, 36(5):1–13, 2017. 3.1

[32] Chen Guo, Tianjian Jiang, Xu Chen, Jie Song, and Otmar Hilliges. Vid2avatar:

3d avatar reconstruction from videos in the wild via self-supervised scene decomposition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12858–12868, 2023. 2.1

[33] Guoqing Hao, Satoshi Iizuka, Kensho Hara, Edgar Simo-Serra, Hirokatsu Kataoka, and Kazuhiro Fukui. Diffusion-based holistic texture rectification and synthesis. In *SIGGRAPH Asia 2023 Conference Papers*, pages 1–11, 2023. 3.1, 3.2.2

[34] Vishnu Mani Hema, Shubhra Aich, Christian Haene, Jean-Charles Bazin, and Fernando De la Torre. Famous: High-fidelity monocular 3d human digitization using view synthesis. *arXiv preprint arXiv:2410.09690*, 2024. 2.2

[35] Philipp Henzler, Valentin Deschaintre, Niloy J Mitra, and Tobias Ritschel. Generative modelling of brdf textures from flash images. *arXiv preprint arXiv:2102.11861*, 2021. 3.2.2

[36] 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. 3.4.1

[37] I Ho, Jie Song, Otmar Hilliges, et al. Sith: Single-view textured human reconstruction with image-conditioned diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 538–549, 2024. 2.2

[38] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 3.1, 3.2.3, 3.3.3

[39] Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. LRM: Large reconstruction model for single image to 3d. *arXiv preprint arXiv:2311.04400*, 2023. 1, 2.2

[40] Alain Hore and Djemel Ziou. Image quality metrics: Psnr vs. ssim. In *2010 20th international conference on pattern recognition*, pages 2366–2369. IEEE, 2010. 2.4.1

[41] Hanzhe Hu, Zhizhuo Zhou, Varun Jampani, and Shubham Tulsiani. Mvd-fusion: Single-view 3d via depth-consistent multi-view generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9698–9707, 2024. 2.2

[42] Liangxiao Hu, Hongwen Zhang, Yuxiang Zhang, Boyao Zhou, Boning Liu, Shengping Zhang, and Liqiang Nie. Gaussianavatar: Towards realistic human avatar modeling from a single video via animatable 3d gaussians. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*,

pages 634–644, 2024. 2.1

[43] Moon-Hwan Jeong, Dong-Hoon Han, and Hyeong-Seok Ko. Garment capture from a photograph. *Computer Animation and Virtual Worlds*, 26(3-4):291–300, 2015. 2.2

[44] Boyi Jiang, Juyong Zhang, Yang Hong, Jinhao Luo, Ligang Liu, and Hujun Bao. Bcnet: Learning body and cloth shape from a single image. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX 16*, pages 18–35. Springer, 2020. 2.2

[45] Biliana Kaneva, Josef Sivic, Antonio Torralba, Shai Avidan, and William T Freeman. Infinite images: Creating and exploring a large photorealistic virtual space. *Proceedings of the IEEE*, 98(8):1391–1407, 2010. 2.2

[46] 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. 3.3.3

[47] Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing*, volume 7, 2006. 2.2, 2.3.3

[48] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):1–14, 2023. 2.2

[49] Rawal Khirodkar, Timur Bagautdinov, Julieta Martinez, Su Zhaoen, Austin James, Peter Selednik, Stuart Anderson, and Shunsuke Saito. Sapiens: Foundation for human vision models. In *European Conference on Computer Vision*, pages 206–228. Springer, 2024. 2.3.2

[50] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 3.3.3

[51] Muhammed Kocabas, Jen-Hao Rick Chang, James Gabriel, Oncel Tuzel, and Anurag Ranjan. Hugs: Human gaussian splats. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 505–515, 2024. 2.2

[52] Maria Korosteleva and Sung-Hee Lee. Generating datasets of 3d garments with sewing patterns. *arXiv preprint arXiv:2109.05633*, 2021. 3.1, 3.2.1

[53] Hyun-Song Kwon and Sung-Hee Lee. Deepiron: Predicting unwarped garment texture from a single image. In *Eurographics*, 2024. 3.2.1

[54] Youngjoong Kwon, Baole Fang, Yixing Lu, Haoye Dong, Cheng Zhang, Francisco Vicente Carrasco, Albert Mosella-Montoro, Jianjin Xu, Shingo Takagi, Daeil Kim, et al. Generalizable human gaussians for sparse view synthesis. In

Proceedings of the European Conference on Computer Vision (ECCV), 2024. 2.2

[55] Boqian Li, Xuan Li, Ying Jiang, Tianyi Xie, Feng Gao, Huamin Wang, Yin Yang, and Chenfanfu Jiang. Garmentdreamer: 3dgs guided garment synthesis with diverse geometry and texture details. *arXiv preprint arXiv:2405.12420*, 2024. 3.2.1

[56] Jiahao Li, Hao Tan, Kai Zhang, Zexiang Xu, Fujun Luan, Yinghao Xu, Yicong Hong, Kalyan Sunkavalli, Greg Shakhnarovich, and Sai Bi. Instant3d: Fast text-to-3d with sparse-view generation and large reconstruction model. *arXiv preprint arXiv:2311.06214*, 2023. 1, 2.2

[57] Ren Li, Corentin Dumery, Benoît Guillard, and Pascal Fua. Garment recovery with shape and deformation priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1586–1595, 2024. 2.1, 2.2

[58] Xueting Li, Xiaolong Wang, Ming-Hsuan Yang, Alexei A Efros, and Sifei Liu. Scraping textures from natural images for synthesis and editing. In *European Conference on Computer Vision*, pages 391–408. Springer, 2022. 3.1

[59] Yifei Li, Hsiao-yu Chen, Egor Larionov, Nikolaos Sarafianos, Wojciech Matusik, and Tuur Stuyck. Diffavatar: Simulation-ready garment optimization with differentiable simulation. *arXiv preprint arXiv:2311.12194*, 2023. 3.2.1

[60] Andrew Liu, Richard Tucker, Varun Jampani, Ameesh Makadia, Noah Snavely, and Angjoo Kanazawa. Infinite nature: Perpetual view generation of natural scenes from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14458–14467, 2021. 2.2

[61] Lijuan Liu, Xiangyu Xu, Zhijie Lin, Jiabin Liang, and Shuicheng Yan. Towards garment sewing pattern reconstruction from a single image. *ACM Transactions on Graphics (TOG)*, 42(6):1–15, 2023. 2.1, 2.2, 3.1, 3.2.1

[62] Minghua Liu, Chao Xu, Haian Jin, Linghao Chen, Mukund Varma T, Zexiang Xu, and Hao Su. One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization. *Advances in Neural Information Processing Systems*, 36, 2024. 2.2

[63] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9298–9309, 2023. 1, 2.1, 2.2, 2.3.2, 2.4.1

[64] Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. Syncdreamer: Generating multiview-consistent images from a single-view image. *arXiv preprint arXiv:2309.03453*, 2023. 2.1, 2.2

[65] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. SMPL: A skinned multi-person linear model. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 851–866. 2023. 2.1

[66] Ivan Lopes, Fabio Pizzati, and Raoul de Charette. Material palette: Extraction of materials from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. (document), 3.1, 3.2.2, 3.3.3, 3.4.1, 3.4.2, 3.6, 3.1, 3.4.2, 3.7, 3.4.3

[67] Zhongjin Luo, Haolin Liu, Chenghong Li, Wanghao Du, Zirong Jin, Wanhu Sun, Yinyu Nie, Weikai Chen, and Xiaoguang Han. Garverselod: High-fidelity 3d garment reconstruction from a single in-the-wild image using a dataset with levels of details. *arXiv preprint arXiv:2411.03047*, 2024. 2.1, 2.2

[68] Sahib Majithia, Sandeep N Parameswaran, Sadbhavana Babar, Vikram Garg, Astitva Srivastava, and Avinash Sharma. Robust 3d garment digitization from monocular 2d images for 3d virtual try-on systems. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3428–3438, 2022. 1, 2.1, 2.2, 3.1, 3.2.1

[69] Stephen McAuley, Stephen Hill, Naty Hoffman, Yoshiharu Gotanda, Brian Smits, Brent Burley, and Adam Martinez. Practical physically-based shading in film and game production. In *ACM SIGGRAPH 2012 Courses*, pages 1–7. 2012. 3.3.2

[70] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision (ECCV)*, 2020. 2.2

[71] Aymen Mir, Thiemo Alldieck, and Gerard Pons-Moll. Learning to transfer texture from clothing images to 3d humans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7023–7034, 2020. 1, 3.1, 3.2.1

[72] Gyeongsik Moon, Hyeongjin Nam, Takaaki Shiratori, and Kyoung Mu Lee. 3d clothed human reconstruction in the wild. In *European conference on computer vision*, pages 184–200. Springer, 2022. 2.2

[73] Joep Moritz, Stuart James, Tom SF Haines, Tobias Ritschel, and Tim Weyrich. Texture stationarization: Turning photos into tileable textures. In *Computer graphics forum*, volume 36, pages 177–188. Wiley Online Library, 2017. 3.2.2

[74] Xingang Pan, Ayush Tewari, Thomas Leimkühler, Lingjie Liu, Abhimitra Meka, and Christian Theobalt. Drag your gan: Interactive point-based manipulation on the generative image manifold. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–11, 2023. 2.1

[75] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022. 2.2

[76] 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. 3.4.1

[77] Amit Raj, Patsorn Sangkloy, Huiwen Chang, James Hays, Duygu Ceylan, and Jingwan Lu. Swapnet: Image based garment transfer. In *Computer Vision–ECCV 2018: 15th European Conference, Munich, Germany, September 8–14, 2018, Proceedings, Part XII 15*, pages 679–695. Springer, 2018. 2.1

[78] 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. 2.1

[79] Elad Richardson, Gal Metzer, Yuval Alaluf, Raja Giryes, and Daniel Cohen-Or. Texture: Text-guided texturing of 3d shapes. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–11, 2023. (document), 2.1, 3.1, 3.2.1, 3.3.3, 3.4.1, 3.4.2, 3.6

[80] Carlos Rodriguez-Pardo and Elena Garces. Seamlessgan: Self-supervised synthesis of tileable texture maps. *IEEE Transactions on Visualization and Computer Graphics*, 29(6):2914–2925, 2022. 3.2.2

[81] Carlos Rodriguez-Pardo, Sergio Suja, David Pascual, Jorge Lopez-Moreno, and Elena Garces. Automatic extraction and synthesis of regular repeatable patterns. *Computers & Graphics*, 83:33–41, 2019. 3.1, 3.2.2

[82] Carlos Rodriguez-Pardo, Henar Dominguez-Elvira, David Pascual-Hernandez, and Elena Garces. Umat: Uncertainty-aware single image high resolution material capture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5764–5774, 2023. 3.1

[83] Carlos Rodriguez-Pardo, Dan Casas, Elena Garces, and Jorge Lopez-Moreno. Textile: A differentiable metric for texture tileability. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4439–4449, 2024. 3.4.1, 3.4.3

[84] 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. 2.1, 2.3.2, 3.1, 3.2.1, 3.2.3, 3.3.3, 3.3.3

[85] Boxiang Rong, Artur Grigorev, Wenbo Wang, Michael J Black, Bernhard

Thomaszewski, Christina Tsalicoglou, and Otmar Hilliges. Gaussian garments: Reconstructing simulation-ready clothing with photorealistic appearance from multi-view video. *arXiv preprint arXiv:2409.08189*, 2024. 2.1

[86] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022. 2.1

[87] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2304–2314, 2019. 2.2

[88] Nikolaos Sarafianos, Tuur Stuyck, Xiaoyu Xiang, Yilei Li, Jovan Popovic, and Rakesh Ranjan. Garment3dgen: 3d garment stylization and texture generation. In *3DV*, 2025. 2.1, 2.2, 2.4.1, 2.1, 3.2.1

[89] Sam Sartor and Pieter Peers. Matfusion: a generative diffusion model for svbrdf capture. In *SIGGRAPH Asia 2023 Conference Papers*, pages 1–10, 2023. 3.1, 3.2.2, 3.3.2, 3.3.4, 3.4.1, 3.4.2

[90] Kai Schröder, Arno Zinke, and Reinhard Klein. Image-based reverse engineering and visual prototyping of woven cloth. *IEEE transactions on visualization and computer graphics*, 21(2):188–200, 2014. 3.1

[91] Ruoxi Shi, Hansheng Chen, Zhuoyang Zhang, Minghua Liu, Chao Xu, Xinyue Wei, Linghao Chen, Chong Zeng, and Hao Su. Zero123++: a single image to consistent multi-view diffusion base model. *arXiv preprint arXiv:2310.15110*, 2023. 1, 2.2, 2.3.2, 2.4.1

[92] Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. *arXiv preprint arXiv:2308.16512*, 2023. 2.2

[93] 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. 3.2.3, 3.3.3

[94] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019. 2.2

[95] Astitva Srivastava, Pranav Manu, Amit Raj, Varun Jampani, and Avinash Sharma. Wordrobe: Text-guided generation of textured 3d garments. *arXiv preprint arXiv:2403.17541*, 2024. 3.2.1

[96] Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. *arXiv preprint arXiv:2309.16653*, 2023. 2.2

[97] Jiaxiang Tang, Zhaoxi Chen, Xiaokang Chen, Tengfei Wang, Gang Zeng, and Ziwei Liu. Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In *European Conference on Computer Vision*, pages 1–18. Springer, 2025. 2.2

[98] Peihan Tu, Li-Yi Wei, and Matthias Zwicker. Clustered vector textures. *ACM Transactions on Graphics (TOG)*, 41(4):1–23, 2022. 3.1

[99] Joseph Tung, Gene Chou, Ruojin Cai, Guandao Yang, Kai Zhang, Gordon Wetzstein, Bharath Hariharan, and Noah Snavely. Megascenes: Scene-level view synthesis at scale. In *European Conference on Computer Vision*, pages 197–214. Springer, 2025. 2.2

[100] Giuseppe Vecchio and Valentin Deschaintre. Matsynth: A modern pbr materials dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22109–22118, 2024. 3.2.2

[101] Giuseppe Vecchio, Simone Palazzo, and Concetto Spampinato. Surfacenet: Adversarial svbrdf estimation from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12840–12848, 2021. 3.2.2

[102] Giuseppe Vecchio, Rosalie Martin, Arthur Roullier, Adrien Kaiser, Romain Rouffet, Valentin Deschaintre, and Tamy Boubekeur. Controlmat: a controlled generative approach to material capture. *ACM Transactions on Graphics*, 2023. 3.2.2

[103] Giuseppe Vecchio, Renato Sortino, Simone Palazzo, and Concetto Spampinato. Matfuse: controllable material generation with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4429–4438, 2024. 3.2.2

[104] Vikram Voleti, Chun-Han Yao, Mark Boss, Adam Letts, David Pankratz, Dmitry Tochilkin, Christian Laforte, Robin Rombach, and Varun Jampani. Sv3d: Novel multi-view synthesis and 3d generation from a single image using latent video diffusion. In *European Conference on Computer Vision*, pages 439–457. Springer, 2025. 2.2

[105] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A Yeh, and Greg Shakhnarovich. Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12619–12629, 2023. 2.2

[106] Junyan Wang, Zhenhong Sun, Zhiyu Tan, Xuanbai Chen, Weihua Chen, Hao Li,

Cheng Zhang, and Yang Song. Towards effective usage of human-centric priors in diffusion models for text-based human image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8446–8455, 2024. 2.1

[107] Tongxin Wang and Mang Ye. Texfit: Text-driven fashion image editing with diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 10198–10206, 2024. 2.1

[108] Zhengyi Wang, Yikai Wang, Yifei Chen, Chendong Xiang, Shuo Chen, Dajiang Yu, Chongxuan Li, Hang Su, and Jun Zhu. Crm: Single image to 3d textured mesh with convolutional reconstruction model. In *European Conference on Computer Vision*, pages 57–74. Springer, 2025. (document), 1, 2.4.1, 2.3, 2.1

[109] Zhou Wang, Eero P Simoncelli, and Alan C Bovik. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, volume 2, pages 1398–1402. Ieee, 2003. 3.4.1

[110] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 2.4.1, 3.4.1

[111] 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. 2.1, 2.2

[112] Li-Yi Wei, Sylvain Lefebvre, Vivek Kwatra, and Greg Turk. State of the art in example-based texture synthesis. *Eurographics 2009, State of the Art Report, EG-STAR*, pages 93–117, 2009. 3.1

[113] Hong-yu Wu, Xiao-wu Chen, Chen-xu Zhang, Bin Zhou, and Qin-ping Zhao. Modeling yarn-level geometry from a single micro-image. *Frontiers of Information Technology & Electronic Engineering*, 20(9):1165–1174, 2019. 3.1

[114] Yuliang Xiu, Jinlong Yang, Dimitrios Tzionas, and Michael J Black. Icon: Implicit clothed humans obtained from normals. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13286–13296. IEEE, 2022. 2.2

[115] Yuliang Xiu, Jinlong Yang, Xu Cao, Dimitrios Tzionas, and Michael J Black. Econ: Explicit clothed humans optimized via normal integration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 512–523, 2023. 2.2

[116] Yuliang Xiu, Yufei Ye, Zhen Liu, Dimitrios Tzionas, and Michael J Black. Puzzleavatar: Assembling 3d avatars from personal albums. *arXiv preprint arXiv:2405.14869*, 2024. 2.1

[117] Jiale Xu, Weihao Cheng, Yiming Gao, Xintao Wang, Shenghua Gao, and Ying Shan. Instantmesh: Efficient 3d mesh generation from a single image with sparse-view large reconstruction models. *arXiv preprint arXiv:2404.07191*, 2024. (document), 1, 2.1, 2.2, 2.4.1, 2.3, 2.1

[118] Yinghao Xu, Zifan Shi, Wang Yifan, Hansheng Chen, Ceyuan Yang, Sida Peng, Yujun Shen, and Gordon Wetzstein. Grm: Large gaussian reconstruction model for efficient 3d reconstruction and generation. *arXiv preprint arXiv:2403.14621*, 2024. 2.2

[119] Shan Yang, Tanya Ambert, Zherong Pan, Ke Wang, Licheng Yu, Tamara Berg, and Ming C Lin. Detailed garment recovery from a single-view image. *arXiv preprint arXiv:1608.01250*, 2016. 2.2

[120] Xianghui Yang, Huiwen Shi, Bowen Zhang, Fan Yang, Jiacheng Wang, Hongxu Zhao, Xinhai Liu, Xinzhou Wang, Qingxiang Lin, Jiaao Yu, et al. Hunyuan3d-1.0: A unified framework for text-to-3d and image-to-3d generation. *arXiv preprint arXiv:2411.02293*, 2024. (document), 2.4.1, 2.3, 2.1

[121] Yu-Ying Yeh, Zhengqin Li, Yannick Hold-Geoffroy, Rui Zhu, Zexiang Xu, Miloš Hašan, Kalyan Sunkavalli, and Manmohan Chandraker. Photoscene: Photorealistic material and lighting transfer for indoor scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18562–18571, 2022. 3.1

[122] Yu-Ying Yeh, Jia-Bin Huang, Changil Kim, Lei Xiao, Thu Nguyen-Phuoc, Numair Khan, Cheng Zhang, Manmohan Chandraker, Carl S Marshall, Zhao Dong, et al. Texturedreamer: Image-guided texture synthesis through geometry-aware diffusion. *arXiv preprint arXiv:2401.09416*, 2024. 3.1, 3.2.1

[123] Hong-Xing Yu, Haoyi Duan, Junhwa Hur, Kyle Sargent, Michael Rubinstein, William T Freeman, Forrester Cole, Deqing Sun, Noah Snavely, Jiajun Wu, et al. Wonderjourney: Going from anywhere to everywhere. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6658–6667, 2024. 2.2

[124] Xianfang Zeng. Paint3d: Paint anything 3d with lighting-less texture diffusion models. *arXiv preprint arXiv:2312.13913*, 2023. 3.1, 3.2.1

[125] Cheng Zhang, Yuanhao Wang, Francisco Vicente Carrasco, Chenglei Wu, Jinlong Yang, Thabo Beeler, and Fernando De la Torre. FabricDiffusion: High-fidelity texture transfer for 3d garments generation from in-the-wild images. In *ACM SIGGRAPH Asia*, 2024. 2.1

[126] Lvmin Zhang and Maneesh Agrawala. Transparent image layer diffusion using latent transparency. *arXiv preprint arXiv:2402.17113*, 2024. 3.3.3

[127] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang.

The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 2.4.1, 3.4.1

[128] Shangzhan Zhang, Sida Peng, Tao Xu, Yuanbo Yang, Tianrun Chen, Nan Xue, Yujun Shen, Hujun Bao, Ruizhen Hu, and Xiaowei Zhou. Mapa: Text-driven photorealistic material painting for 3d shapes. *arXiv preprint arXiv:2404.17569*, 2024. 3.2.1

[129] Zerong Zheng, Tao Yu, Yebin Liu, and Qionghai Dai. Pamir: Parametric model-conditioned implicit representation for image-based human reconstruction. *IEEE transactions on pattern analysis and machine intelligence*, 44(6):3170–3184, 2021. 2.2

[130] Bin Zhou, Xiaowu Chen, Qiang Fu, Kan Guo, and Ping Tan. Garment modeling from a single image. In *Computer graphics forum*, volume 32, pages 85–91. Wiley Online Library, 2013. 2.2

[131] Shizhe Zhou, Hongbo Fu, Ligang Liu, Daniel Cohen-Or, and Xiaoguang Han. Parametric reshaping of human bodies in images. *ACM transactions on graphics (TOG)*, 29(4):1–10, 2010. 2.1

[132] Xilong Zhou and Nima Khademi Kalantari. Adversarial single-image svbrdf estimation with hybrid training. In *Computer Graphics Forum*, volume 40, pages 315–325. Wiley Online Library, 2021. 3.3.2

[133] Xilong Zhou, Milos Hasan, Valentin Deschaintre, Paul Guerrero, Kalyan Sunkavalli, and Nima Khademi Kalantari. Tilegen: Tileable, controllable material generation and capture. In *SIGGRAPH Asia 2022 conference papers*, pages 1–9, 2022. 3.2.2, 3.3.2, 3.3.3

[134] Xilong Zhou, Milos Hasan, Valentin Deschaintre, Paul Guerrero, Yannick Hold-Geoffroy, Kalyan Sunkavalli, and Nima Khademi Kalantari. Photomat: A material generator learned from single flash photos. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–11, 2023. 3.3.2

[135] Yang Zhou, Kaijian Chen, Rongjun Xiao, and Hui Huang. Neural texture synthesis with guided correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18095–18104, 2023. 3.2.2

[136] Zhizhuo Zhou and Shubham Tulsiani. Sparsefusion: Distilling view-conditioned diffusion for 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12588–12597, 2023. 2.2

[137] Heming Zhu, Lingteng Qiu, Yuda Qiu, and Xiaoguang Han. Registering explicit to implicit: Towards high-fidelity garment mesh reconstruction from single images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and*

Pattern Recognition, pages 3845–3854, 2022. 2.2

[138] 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*, pages 1346–1356, 2024. 2.1

[139] Xingxing Zou, Xintong Han, and Waikeung Wong. Cloth4d: A dataset for clothed human reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12847–12857, 2023. 2.1, 2.2, 2.4.1