FabricDiffusion: High-Fidelity Texture Transfer for 3D Garments Generation from In-The-Wild Clothing Images

CHENG ZHANG*, Carnegie Mellon University, United States of America
YUANHAO WANG*, Carnegie Mellon University, United States of America
FRANCISCO VICENTE CARRASCO, Carnegie Mellon University, United States of America
CHENGLEI WU, Google Inc., United States of America
JINLONG YANG, Google Inc., Switzerland
THABO BEELER, Google Inc., Switzerland
FERNANDO DE LA TORRE, Carnegie Mellon University, United States of America

Fig. 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.

*Equal contribution. Website: https://humansensinglab.github.io/fabric-diffusion.

Authors' Contact Information: Cheng Zhang, Carnegie Mellon University, United States of America, chzhang@tamu.edu; Yuanhao Wang, Carnegie Mellon University, United States of America, yuanhao4@andrew.cmu.edu; Francisco Vicente Carrasco, Carnegie Mellon University, United States of America, fvicente@andrew.cmu.edu; Chenglei Wu, Google Inc., United States of America, chengleiwu@gmail.com; Jinlong Yang, Google Inc., Switzerland, jinlongy@google.com; Thabo Beeler, Google Inc., Switzerland, thabo.beeler@gmail.com; Fernando De la Torre, Carnegie Mellon University, United States of America, ftorre@cs.cmu.edu.

We introduce FabricDiffusion, a method for transferring fabric textures from a single clothing image to 3D garments of arbitrary shapes. Existing approaches typically synthesize textures on the garment surface through 2D-to-3D texture mapping or depth-aware inpainting via generative models. Unfortunately, these methods often struggle to capture and preserve texture details, particularly due to challenging occlusions, distortions, or poses in the input image. Inspired by the observation that in the fashion industry, most garments are constructed by stitching sewing patterns with flat, repeatable textures, we cast the task of clothing texture transfer as extracting distortion-free, tileable texture materials that are subsequently mapped onto the UV space of the garment. Building upon this insight, we train a denoising

diffusion model with a large-scale synthetic dataset to rectify distortions in the input texture image. This process yields a flat texture map that enables a tight coupling with existing Physically-Based Rendering (PBR) material generation pipelines, allowing for realistic relighting of the garment under various lighting conditions. We show that FabricDiffusion can transfer various features from a single clothing image including texture patterns, material properties, and detailed prints and logos. Extensive experiments demonstrate that our model significantly outperforms state-to-the-art methods on both synthetic data and real-world, in-the-wild clothing images while generalizing to unseen textures and garment shapes.

CCS Concepts: • **Computing methodologies → Appearance and texture representations**;

Additional Key Words and Phrases: Texture transfer, BRDF material, diffusion model, synthetic data, 3D garments reconstruction

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 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 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 [Gao et al. 2024; Majithia et al. 2022; Mir et al. 2020] or multi-view depth-aware inpainting by distilling a pre-trained 2D generative model [Richardson et al. 2023; Yeh et al. 2024; Zeng 2023]. 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 [Korosteleva and Lee 2021; Liu et al. 2023]: 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 [Cazenavette et al. 2022; Diamanti et al. 2015; Efros and Freeman 2023; Efros and Leung 1999; Guarnera et al. 2017; Hao et al. 2023; Li et al. 2022; Lopes et al. 2024; Rodriguez-Pardo et al. 2023, 2019; Schröder et al. 2014; Tu et al. 2022; Wei et al. 2009; Wu et al. 2019; Yeh et al. 2022]. 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 [Ho et al. 2020; Rombach et al. 2022] 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 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 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 [Lopes et al. 2024], allowing for accurate relighting of the garment under different lighting conditions. In summary,

---

[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 "undistorted", "distortion-free", "unwarped", and "flat" are used interchangeably in this paper to describe the textures free from geometric distortions.

FabricDiffusion represents a state-of-the-art approach capable of extracting undistorted texture maps from real-world clothing images to produce realistic 3D garments.

2 Related Work

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

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 [Gao et al. 2024; Majithia et al. 2022; Mir et al. 2020] and (2) conducting depth-aware inpainting supervised by a pre-trained image generative model [Rombach et al. 2022] to guarantee multi-view consistency [Richardson et al. 2023; Yeh et al. 2024; Zeng 2023; Zhang et al. 2024]. 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 [Korosteleva and Lee 2021]) 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 [Liu et al. 2023] or automatically reconstructed from reference images [Chen et al. 2022; Li et al. 2023; Liu et al. 2023]. Concurrently, DeepIron [Kwon and Lee 2024] 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 [Li et al. 2024] and WordRobe [Srivastava et al. 2024] are recent work that focus on text-based garment generation, whereas our approach transfers textures using image guidance. Another relevant work, Garment3DGen [Sarafianos et al. 2024], 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.

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. [2023] 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 [Lopes et al. 2024] 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 [Gal et al. 2022] 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 [Moritz et al. 2017], by using Guided Correspondence [Zhou et al. 2023a], by finding repeated patterns in images using pre-trained CNN features [Rodriguez-Pardo et al. 2019], by manipulating the latent space of pre-trained GANs [Rodriguez-Pardo and Garces 2022], or by modifying the noise sampling process of a diffusion model, i.e., rolled-diffusion [Vecchio et al. 2023]. We found that a simple circular padding strategy following [Zhou et al. 2022] 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 [Casas and Comino-Trinidad 2023; Deschaintre et al. 2018; Henzler et al. 2021; Vecchio and Deschaintre 2024; Vecchio et al. 2021, 2024]. 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 [Sartor and Peers 2023], 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 it into our pipeline, our model generates high-quality material maps for realistic 3D garment rendering.

2.3 Diffusion-based Image Generation

Our model architecture is inspired by the recent advancements in diffusion-based image generation models [Ho et al. 2020; Rombach et al. 2022; Sohl-Dickstein et al. 2015]. 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 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 2. We first introduce the problem statement in Section 3.1, followed by procedures for constructing synthetic training examples in Section 3.2. In Section 3.3, we detail our specific approach of texture map generation. Finally, we describe PBR materials generation and garment rendering in Section 3.4.

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
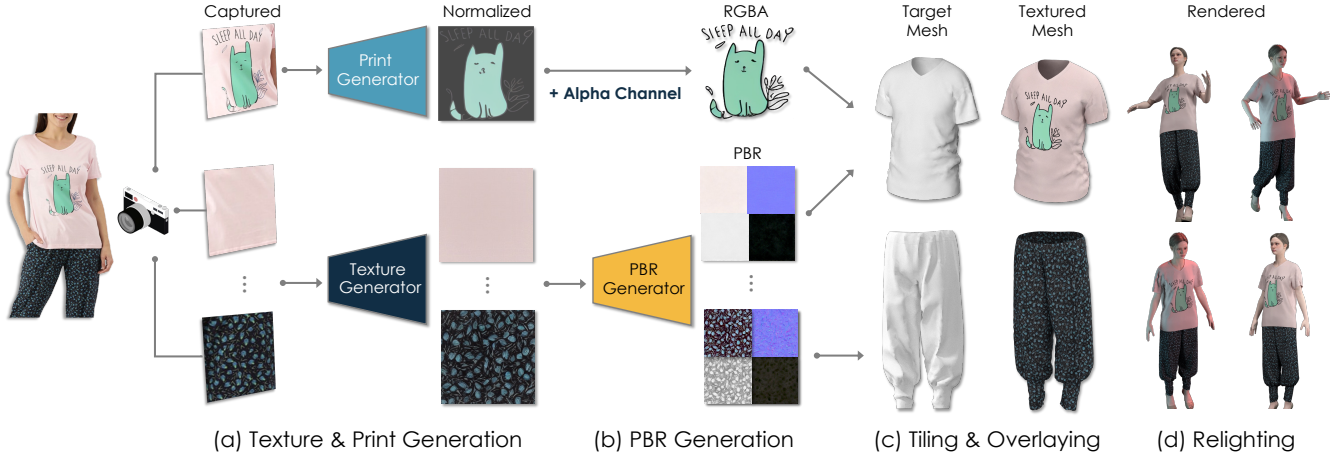
Fig. 2. **Overview of FabricDiffusion.** Given a real-life clothing image and region captures of its fabric materials and prints, (a) our model extracts normalized textures and prints, and (b) then generates high-quality Physically-Based Rendering (PBR) materials and transparent prints. (c) These materials 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.

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 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:

$$\tilde{x} \sim G_\theta(x, \epsilon), \epsilon \sim \mathcal{N}(0, \mathbf{I}). \tag{1}$$

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.

3.2 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 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 [McAuley et al. 2012]. Figure 3 illustrates the overall pipeline.

Paired training examples construction. For each material, we collect the ground-truth diffuse albedo ($k_d \in \mathbb{R}^3$), normal ($k_n \in \mathbb{R}^3$), roughness ($k_r \in \mathbb{R}^2$), and metallic ($k_m \in \mathbb{R}^2$) material maps. To create distorted rendered images that mimic real-world surface deformation and lighting, we map these material maps onto a raw
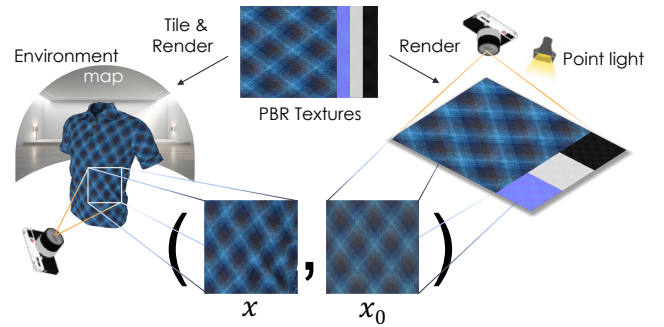

Fig. 3. **Pipeline of paired training data construction.** Given the textures of a PBR material, we apply them to both the target raw 3D garment mesh and the plain mesh. The 3D garment is rendered using an environment map, while the plain mesh is illuminated using a point light from above. The resulting rendered images (x, x_0) from both meshes serve as the paired training examples for training our texture generative model (Section 3.2).

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). 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 [Sartor and Peers 2023; Zhou et al.

2023b, 2022; Zhou and Kalantari 2021], which require orthogonal close-up views of the materials and/or a flashing image as input. As will be described in Section 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 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 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 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), 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 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 [Ho et al. 2020; Sohl-Dickstein et al. 2015] 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) [Rombach et al. 2022] to improve the efficiency and quality of diffusion models by operating in the latent space of a pre-trained variational autoencoder [Kingma and Welling 2013] 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:

$$x_t = \sqrt{\gamma(t)}\mathcal{E}(x_0) + \sqrt{1 - \gamma(t)}\epsilon, \tag{2}$$

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

$$L(\theta) = \mathbb{E}_{\mathcal{E}(x), \epsilon \sim \mathcal{N}(0, \mathbf{I}), t}\left[\|\epsilon - \epsilon_\theta(x_t, t, \mathcal{E}(x))\|^2\right], \tag{3}$$

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 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 4.2, this design is the key to producing faithful texture maps that differs from existing per-example optimization-based texture extraction approaches [Lopes et al. 2024; Richardson et al. 2023].

Model architecture and training. Any diffusion-based architecture for conditional image generation can realize Equation 3. Specifically, we use Stable Diffusion [Rombach et al. 2022], 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 [Deschaintre et al. 2023].

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 [Zhou et al. 2022]. Unlike TileGen, which uses a StyleGAN-like architecture [Karras et al. 2020] 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 [Zhang and Agrawala 2024], 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

---

[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.

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

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:

$$\text{RGB}(i, j) = \max\left[0, \frac{\tilde{x}(i, j) - 0.1}{0.9}\right], \qquad (4)$$

where $\tilde{x}$ is the generated texture (Equation 1). The alpha channel value at each pixel (i, j) is thus determined by the following criteria:

$$A(i, j) = \begin{cases} 1 & \text{if } \tilde{x}(i, j) \geq 0.1, \\ \tilde{x}(i, j)/0.1 & \text{otherwise.} \end{cases} \qquad (5)$$

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 4.2 and Figure 5, our method can handle complex prints and logos and output RGBA print images that can be overlaid onto the fabric texture.

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 [Sartor and Peers 2023] 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 captured 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.

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 4.1, followed by detailing the experimental results in Section 4.2. Finally, we conduct ablation studies and show several real-world applications in Section 4.3.

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 3.2, 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 [Zhang et al. 2018], SSIM [Wang et al. 2004], MS-SSIM [Wang et al. 2003], DISTS [Ding et al. 2020], and FLIP [Andersson et al. 2020]. To evaluate the tileability of the generated textures, we adopt the metric proposed by TexTile [Rodriguez-Pardo et al. 2024]. For the image-to-garment texture transfer task, we additionally report FID [Heusel et al. 2017] and CLIP-score in CLIP image feature space [Gal et al. 2022; Radford et al. 2021] 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* [Richardson et al. 2023], 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 [Gal et al. 2022] for personalization). (2) *Material Palette* [Lopes et al. 2024], which focuses on texture extraction and PBR materials estimation from a single image using generative models. (3) *MatFusion* [Sartor and Peers 2023], 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.

4.2 Experimental Results

FabricDiffusion on real-world clothing images. We first show the results of our method on real-world images in Figure 4. Our method effectively transfers both texture patterns and material properties from 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 5 shows some examples. We highlight two key advantages of our design that benefit

Fig. 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.

Input	Captured	Normalized	Generated PBR textures	Textured 3D garment

Fig. 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.

| | Input | FabricDiffusion (ours) | Material Palette [Lopes et al. 2024] | TEXTure [Richardson et al. 2023] |

Fig. 6. **Comparison on image-to-garment texture transfer.** FabricD-iffusion faithfully captures and preserves the texture pattern from the input clothing. We observe texture irregularities and artifacts for Material Palette [Lopes et al. 2024] and TEXTure [Richardson et al. 2023].

Table 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 [Lopes et al. 2024] 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 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 subpar, particularly in estimating the diffuse and roughness maps, due the differences in physical properties between fabric materials and general objects. Please see Table 3 for quantitative evaluation on rendered images and Figure 7 for a qualitative comparison between FabricDiffusion and Material Palette.

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

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, our method can generate prints with a transparent background, enabling practical usage in garment appearance modeling.

Image-to-garment texture transfer. In Figure 6, we compare our method with Material Palette [Lopes et al. 2024] and TEX-Ture [Richardson et al. 2023] 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 structured patterns, like the plaid on a skirt. Please refer to Table 1 for quantitative results.

PBR materials extraction. We also compare our method to Material Palette [Lopes et al. 2024] and MatFusion [Sartor and Peers 2023] on PBR materials extraction. In Table 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 7 shows visual comparisons between FabricDiffusion and Material Palette. While Material Palette [Lopes et al. 2024] 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. Particularly, we use render-aware metrics like FLIP [Andersson et al.
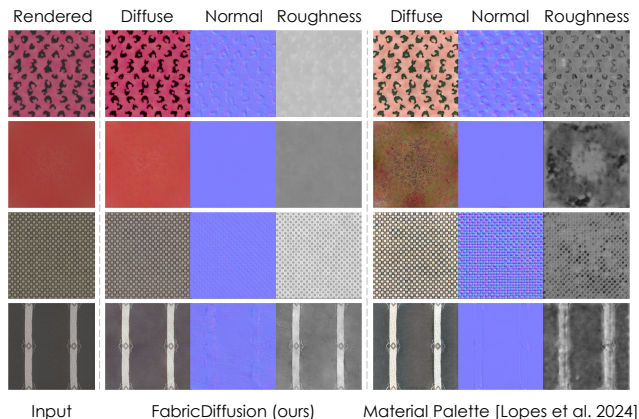
Fig. 7. **Qualitative comparison on PBR materials extraction.** Material Palette [Lopes et al. 2024] can hardly capture fabric materials while our FabricDiffusion model is capable of recovering physical properties for fabric textures especially on roughness and diffuse maps.

Table 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**

2020] and perceptual metrics like LPIPS and DISTS. FabricDiffusion consistently achieve better performance over other approaches.

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 [Rodriguez-Pardo et al. 2024], where higher values indicate better tileability. The results show that the MaterialPalette [Lopes et al. 2024] 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.

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 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 3.4, 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 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.

Effect of the capture scale. In Figure 8-(b), we further study the effect of the size of the captured region. By varying the scale of the

Table 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**

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 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 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.

5 Discussion, Limitation, and Conclusion

In this paper, we introduce FabricDiffusion, a new method for transferring fabric textures and prints from a single real-world clothing image onto 3D garments with arbitrary shapes. Our method, trained entirely using synthetic rendered images, is able to generate undistorted texture and prints from in-the-wild clothing images. While our method demonstrates strong generalization abilities with real photos and diverse texture patterns, it faces challenges with certain inputs, as shown in Figure 11. Specifically, FabricDiffusion may produce errors when reconstructing non-repetitive patterns and struggles to accurately capture fine details in complex prints or logos, especially since our focus is on prints with uniform backgrounds, moderate complexity, and moderate distortion. In the future, we plan to address these challenges by enhancing texture transfer for more complex scenarios and improving performance on difficult fabric categories, such as leather. Additionally, we plan to expand our method to handle a broader range of material maps, including transmittance, to further extend its applicability.

References

Pontus Andersson, Jim Nilsson, Tomas Akenine-Möller, Magnus Oskarsson, Kalle Åström, and Mark D Fairchild. 2020. FLIP: A Difference Evaluator for Alternating Images. *Proc. ACM Comput. Graph. Interact. Tech.* 3, 2 (2020), 15–1.

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

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

Xipeng Chen, Guangrun Wang, Dizhong Zhu, Xiaodan Liang, Philip Torr, and Liang Lin. 2022. Structure-preserving 3D garment modeling with neural sewing machines. *Advances in Neural Information Processing Systems* 35 (2022), 15147–15159.

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

Fig. 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.

Fig. 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.

Fig. 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.
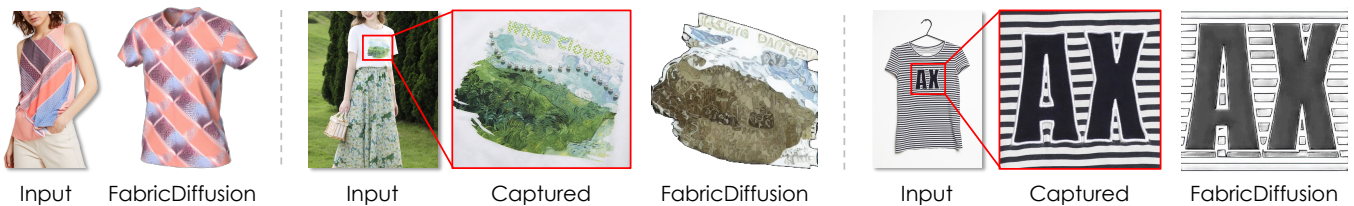

Fig. 11. **Limitations of FabricDiffusion.** Our method may struggle to reconstruct specific inputs such as complex (e.g., non-repetitive) patterns (left), fine details in complex prints (middle), and prints over non uniform fabric (right).

Valentin Deschaintre, Diego Gutierrez, Tamy Boubekeur, Julia Guerrero-Viu, and Belen Masia. 2023. *The visual language of fabrics.* Technical Report.

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

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

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

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

Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. 2022. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618* (2022).

Daiheng Gao, Xu Chen, Xindi Zhang, Qi Wang, Ke Sun, Bang Zhang, Liefeng Bo, and Qixing Huang. 2024. 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).*

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

Guoqing Hao, Satoshi Iizuka, Kensho Hara, Edgar Simo-Serra, Hirokatsu Kataoka, and Kazuhiro Fukui. 2023. Diffusion-based Holistic Texture Rectification and Synthesis. In *SIGGRAPH Asia 2023 Conference Papers.* 1–11.

Philipp Henzler, Valentin Deschaintre, Niloy J Mitra, and Tobias Ritschel. 2021. Generative modelling of BRDF textures from flash images. *arXiv preprint arXiv:2102.11861* (2021).

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems* 30 (2017).

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems* 33 (2020), 6840–6851.

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

Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).

Maria Korosteleva and Sung-Hee Lee. 2021. Generating datasets of 3d garments with sewing patterns. *arXiv preprint arXiv:2109.05633* (2021).

Hyun-Song Kwon and Sung-Hee Lee. 2024. DeepIron: Predicting Unwarped Garment Texture from a Single Image. In *Eurographics.*

Boqian Li, Xuan Li, Ying Jiang, Tianyi Xie, Feng Gao, Huamin Wang, Yin Yang, and Chenfanfu Jiang. 2024. GarmentDreamer: 3DGS Guided Garment Synthesis with Diverse Geometry and Texture Details. *arXiv preprint arXiv:2405.12420* (2024).

Xueting Li, Xiaolong Wang, Ming-Hsuan Yang, and Sifei Liu. 2022. Scraping Textures from Natural Images for Synthesis and Editing. In *European Conference on Computer Vision.* Springer, 391–408.

Yifei Li, Hsiao-yu Chen, Egor Larionov, Nikolaos Sarafianos, Wojciech Matusik, and Tuur Stuyck. 2023. DiffAvatar: Simulation-Ready Garment Optimization with Differentiable Simulation. *arXiv preprint arXiv:2311.12194* (2023).

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

Ivan Lopes, Fabio Pizzati, and Raoul de Charette. 2024. Material Palette: Extraction of Materials from a Single Image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.*

Sahib Majithia, Sandeep N Parameswaran, Sadbhavana Babar, Vikram Garg, Astitva Srivastava, and Avinash Sharma. 2022. 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.* 3428–3438.

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

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

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

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning.* PMLR, 8748–8763.

Elad Richardson, Gal Metzer, Yuval Alaluf, Raja Giryes, and Daniel Cohen-Or. 2023. Texture: Text-guided texturing of 3d shapes. In *ACM SIGGRAPH 2023 Conference Proceedings.* 1–11.

Carlos Rodriguez-Pardo, Dan Casas, Elena Garces, and Jorge Lopez-Moreno. 2024. TexTile: A Differentiable Metric for Texture Tileability. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 4439–4449.

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

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

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

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.* 10684–10695.

Nikolaos Sarafianos, Tuur Stuyck, Xiaoyu Xiang, Yilei Li, Jovan Popovic, and Rakesh Ranjan. 2024. Garment3DGen: 3D Garment Stylization and Texture Generation. *arXiv preprint arXiv:2403.18816* (2024).

Sam Sartor and Pieter Peers. 2023. Matfusion: a generative diffusion model for svbrdf capture. In *SIGGRAPH Asia 2023 Conference Papers.* 1–10.

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

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning.* PMLR, 2256–2265.

Astitva Srivastava, Pranav Manu, Amit Raj, Varun Jampani, and Avinash Sharma. 2024. WordRobe: Text-Guided Generation of Textured 3D Garments. *arXiv preprint arXiv:2403.17541* (2024).

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

Giuseppe Vecchio and Valentin Deschaintre. 2024. MatSynth: A Modern PBR Materials Dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 22109–22118.

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

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

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

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

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

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

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

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. 2024. TextureDreamer: Image-guided Texture Synthesis through Geometry-aware Diffusion. *arXiv preprint arXiv:2401.09416* (2024).

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

Xianfang Zeng. 2023. Paint3D: Paint Anything 3D with Lighting-Less Texture Diffusion Models. *arXiv preprint arXiv:2312.13913* (2023).

Lvmin Zhang and Maneesh Agrawala. 2024. Transparent Image Layer Diffusion using Latent Transparency. *arXiv preprint arXiv:2402.17113* (2024).

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition.* 586–595.

Shangzhan Zhang, Sida Peng, Tao Xu, Yuanbo Yang, Tianrun Chen, Nan Xue, Yujun Shen, Hujun Bao, Ruizhen Hu, and Xiaowei Zhou. 2024. MaPa: Text-driven

Photorealistic Material Painting for 3D Shapes. *arXiv preprint arXiv:2404.17569* (2024).

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

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

Xilong Zhou and Nima Khademi Kalantari. 2021. Adversarial Single-Image SVBRDF Estimation with Hybrid Training. In *Computer Graphics Forum*, Vol. 40. Wiley Online Library, 315–325.

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