TextureDreamer: Image-guided Texture Synthesis through Geometry-aware Diffusion

Yu-Ying Yeh[13] Jia-Bin Huang[23] Changil Kim[3] Lei Xiao[3] Thu Nguyen-Phuoc[3] Numair Khan[3]
Cheng Zhang[3] Manmohan Chandraker[1] Carl S Marshall[3] Zhao Dong[3] Zhengqin Li[3]

[1]University of California, San Diego [2]University of Maryland, College Park [3]Meta Reality Lab

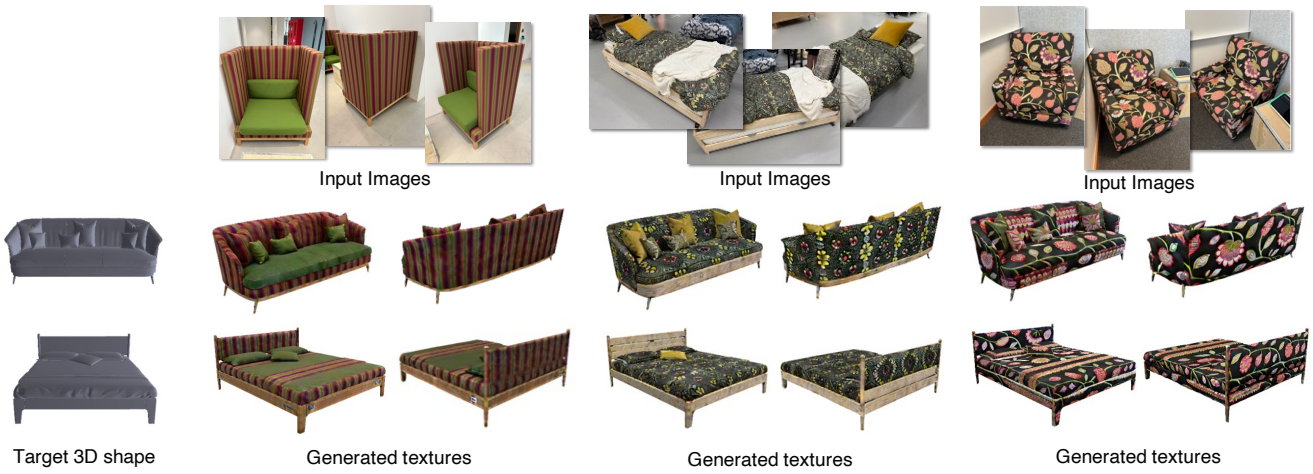

Figure 1. **Texture transfer from sparse images.** Given a small number of images and a target mesh, our method synthesizes geometry-aware texture that looks similar to the input appearances for diverse objects.

Abstract

We present TextureDreamer, a novel image-guided texture synthesis method to transfer relightable textures from a small number of input images (3 to 5) to target 3D shapes across arbitrary categories. Texture creation is a pivotal challenge in vision and graphics. Industrial companies hire experienced artists to manually craft textures for 3D assets. Classical methods require densely sampled views and accurately aligned geometry, while learning-based methods are confined to category-specific shapes within the dataset. In contrast, TextureDreamer can transfer highly detailed, intricate textures from real-world environments to arbitrary objects with only a few casually captured images, potentially significantly democratizing texture creation. Our core idea, personalized geometry-aware score distillation (PGSD), draws inspiration from recent advancements in diffuse models, including personalized modeling for texture information extraction, score distillation for detailed appearance synthesis, and explicit geometry guidance with ControlNet. Our integration and several essential modifications substantially improve the texture quality. Experiments on real images spanning different categories show that TextureDreamer can successfully transfer highly realistic, semantic meaningful texture to arbitrary objects, surpassing the visual quality of previous state-of-the-art. Project page: https://texturedreamer.github.io

1. Introduction

High-quality 3D content is indispensable for a wide range of critical applications, including AR/VR, robotics, film, and gaming. In recent years, remarkable progress has been made in democratizing 3D content creation pipelines, facilitated by advancements in 3D reconstruction [41, 44] and generative models [18, 62]. While substantial attention has been devoted to exploring the *geometry component* [8, 12, 67] and neural implicit representations [46], such as NeRF [41], creation of high-quality *textures* is relatively under-explored. Textures are pivotal in creating realistic, highly detailed appearances and are integral to various graphics pipelines, where industry has traditionally relied on professional, experienced artists to craft textures. This process usually involves manually authoring procedu-

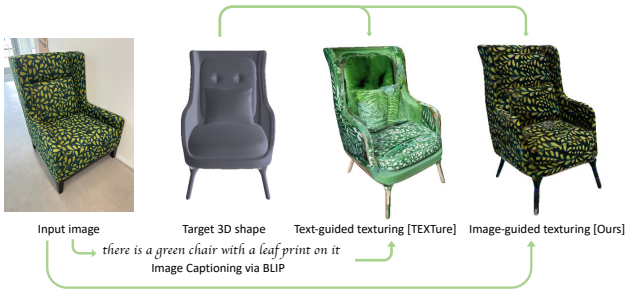

Figure 2. **Limitation of text-guided texturing.** Compared to text-guided texturing method which requires a captioning method to generate a text prompt which might not express all the details of the image, image-based guided texturing can be more effective and more expressive. Image captioning is predicted by BLIP [33], text-guided texturing is generated via TEXTure [55], and image-guided result is from our method.

ral graphs [1] and UV maps, making it expensive and inefficient. Automatically transferring the diverse visual appearance of objects around us to the texture of any target geometry would thus be highly beneficial.

We present *TextureDreamer*, a novel framework to create high-quality relightable textures from sparse images. Given 3 to 5 randomly sampled views of an object, we can transfer its texture to an target geometry that may come from a different category. This is an extremely challenging problem, as previous texture creation methods usually either require densely sampled views with aligned geometry [3, 32, 71], or can only work for category-specific shapes [4, 21, 48, 61]. Our framework draws inspiration from recent advancements in diffusion-based generative models [23, 62, 63]. Trained on billions of text-image pairs, these diffusion models enable text-guided image generation with extraordinary visual quality and diversity [54]. Pioneering works have applied these pre-trained 2D diffusion models to text-guided 3D content creation [35, 49, 66]. However, a common limitation among those methods is that *text-only input* may not be sufficiently expressive to describe complex, detailed patterns, as demonstrated in Figure 2. In contrast to text-guided methods, we effectively extract texture information from a small set of input images by fine-tuning the pre-trained diffusion model with a unique text token [16, 57]. Our framework, therefore, addresses the challenge of accurately describing complex textures.

The Score Distillation Sampling (SDS) [49, 65] is one core element that bridges pre-trained 2D diffusion models with 3D content creation. It is widely used to generate and edit 3D contents by minimizing the discrepancy between the distribution of rendered images and the distribution defined by the pre-trained diffusion models [35, 38]. Despite its popularity, two well-known limitations impede its ability to generate high-quality textures. First, it tends to create over-smoothed and saturated appearances due to the unusually high classifier-free guidance necessary for the method to converge. Second, it lacks the knowledge to generate a 3D-consistent appearance, often resulting in multi-face artifacts and mismatches between textures and geometry.

We propose two key design choices to tackle these challenges. Instead of using SDS, we build upon Variational Score Distillation (VSD) in our optimization approach, which can generate much more photorealistic and diverse textures. Initially introduced in ProlificDreamer [66], VSD treats the whole 3D representation as a random variable and aligns its distribution with the pre-trained diffusion model. It does not need a large classifier-free guidance weight to converge, which is essential to create a realistic and diverse appearance. However, naïvely applying VSD update does not suffice for generating high-quality textures in our application. We identify a simple modification that can improve texture quality while slightly reducing the computational cost. Additionally, VSD alone cannot fully solve the 3D consistency issue. Fine-tuning on sparse inputs makes converging harder, as observed by previous work [53]. We, therefore, explicitly condition our texture generation process on geometry information extracted from the given mesh by injecting rendered normal maps into the fine-tuned diffusion model through the ControlNet [70] architecture. Our framework, designated as personalized geometry aware score distillation (PGSD), can effectively transfer highly detailed textures to diverse geometry in a semantically meaningful and visually appealing manner. Extensive qualitative and quantitative experiments demonstrate that our framework substantially outperforms state-of-the-art texture-transfer methods.

2. Related Works

Texture synthesis and reconstruction Classical texture creation methods involve sampling from a distribution derived from the neighborhood [13, 28], tiling repetitive patterns [29, 56], or with generative approaches [43, 72]. These methods fall short in creating semantic meaningful textures. Texture reconstruction by fusing multi-view images onto the object surfaces [3, 32, 71] requires highly accurate geometry reconstruction. Numerous learning-based methods were proposed to learn texture creation from large-scale 3D datasets [4, 11, 21, 48, 61] but are confined to specific categories within the dataset. Recent works also use CLIP model [52] for text-guided texture generation of arbitrary objects [31, 37, 40, 42], but their texture qualities are usually low. In contrast, TextureDreamer can create semantically meaningful, high-quality textures for arbitrary objects using uncorrelated sparse images. Traditionally, textures are represented as a 2D image and projected to object surfaces through UV mapping. Leveraging the recent progress in neural implicit representation, our

method, along with recent developments in inverse rendering [5, 7, 17, 64] and 3D generation [7, 17], represents texture as a neural implicit texture field.

Diffusion models Diffusion models [62] have emerged as the state-of-the-art generative models [23, 63], demonstrating exceptional visual quality [54]. Its training and inference involve iteratively adding noise with different variances and denoise the data. Trained on internet-scale image-text pair datasets [54], these pre-trained models exhibit unprecedented capability in text-guided image synthesis and have proven successful in various image editing tasks. Recent works also manage to fine-tune pre-trained diffusion models on much smaller datasets or even a few images to facilitate customized/personalized image synthesis [57] and image generation conditioned on multi-modal data [70], such as normal and semantic maps. Building upon this progress, TextureDreamer can effectively extract texture information from sparse views and transfer it to a novel target object in a geometry-aware manner.

3D generation with 2D diffusion priors Diffusion-based 3D content creation has very recently gained substantial interest. Several methods directly train 3D diffusion models to generate 3D content in various representations, including point cloud [36], neural radiance filed [26], hypernetwork [14] and texture [69]. Others utilize pre-trained 2D diffusion models by either progressively fusing generated images from different views [2, 6, 9, 55] or optimizing the 3D representation through score distillation sampling [35, 38, 49] and its improved variations [27, 66]. While many methods concentrate on text-guided 3D generation, fewer attempt to leverage diffusion models to generate 3D content from images. A number of concurrent works fine-tune 2D diffusion models on large-scale 3D datasets for sparse view reconstruction [50, 60], primarily focusing on whole 3D object reconstruction. In contrast, TextureDreamer targets transferring textures from a small number of images to a target 3D shape with unmatched geometry. Dreambooth3D [53] and TEXTure [55] extract information from sparse views into a new text token and fine-tuned diffusion model weights, which can be used to generate personalized 3D object or texture unseen objects. TextureDreamer employs a similar method to extract information from sparse images. However, it differs from prior works on utilizing the extracted information for texture generation, leading to improvements in consistency and photorealism.

3. Method

We propose TextureDreamer, a framework which synthesizes geometry-aware texture for a given mesh with appearance similar to 3-5 input images of an object. In Section 3.1, we first introduce preliminaries on Dreambooth [57], ControlNet [70] and score distillation sampling [49, 65, 66]. In Section 3.2, we propose personalized geometry-aware score distillation (PGSD), which is our core technical contribution that enables high-quality image-guided texture transfer from sparse images to arbitrary geometries.

3.1. Preliminaries

Dreambooth [57] is a simple yet effective method to fine-tune pre-trained text-to-image diffusion models on a small number of input images for personalized text-guided image generation. It stores the subject's appearance into the diffusion model weights with a specific text-token *"[V]"*. Dreambooth is fine-tuned with two loss functions. Reconstruction loss is standard diffusion denoising supervision on the input images. Class-specific prior preservation loss is proposed to avoid language drift and loss of diversity caused by fine-tuning. It further supervises the pre-trained model with a large number of its own generated examples. TextureDreamer uses DreamBooth to distill texture information from input images. Instead of image synthesis, we apply the distilled information to a 3D object with different geometry.

ControlNet [70] proposes a novel architecture that adds spatial conditioning control to pre-trained diffusion models. The key insight is to reuse the large number of diffusion model parameters trained on billions of images and insert small convolution networks into the model with window size 1 and zero-initialized weights. It enables robust fine-tuning performance on small datasets with different types of 2D conditions, such as depth, normal, and edge maps. We utilize ControlNet models to ensure that our created textures are aligned with the given geometry.

Score Distillation Sampling [49, 65] is the core component of numerous methods that use pre-trained 2D diffusion models for 3D content creation [10, 35, 49]. It optimizes the 3D representation by pushing its rendered images to a high-dimensional manifold modeled by the pre-trained diffusion model. Let θ be the 3D representation and ϵ_ψ be the pre-trained diffusion model. The gradient back-propagated to the parameter θ is

$$\nabla_\theta \mathcal{L}_{\text{SDS}}(\theta) \triangleq \mathbb{E}_{t,\epsilon}\left[w(t)(\epsilon_\psi\left(\mathbf{x}_t, y^c, t\right) - \epsilon)\,\frac{\partial g(\theta, c)}{\partial \theta}\right],$$

where $\omega(t)$ is the weight coefficient, y is the text input, y^c denotes view-dependent conditioning, t is the time step, c is the camera pose, $g(\cdot)$ is a differentiable renderer, $\mathbf{x}_t$ is the noisy image computed by adding noise to the rendered image $\mathbf{x} = g(\theta, c)$ with variance dependent on time t. Despite its wide usage, SDS requires a much higher weight than normal classifier-free guidance [22] to converge, oversmoothed and oversaturated appearance. To overcome this issue, Wang et al. [66] propose an improved version, called variational score distillation (VSD), which can converge with standard classifier-free guidance. VSD treats the whole

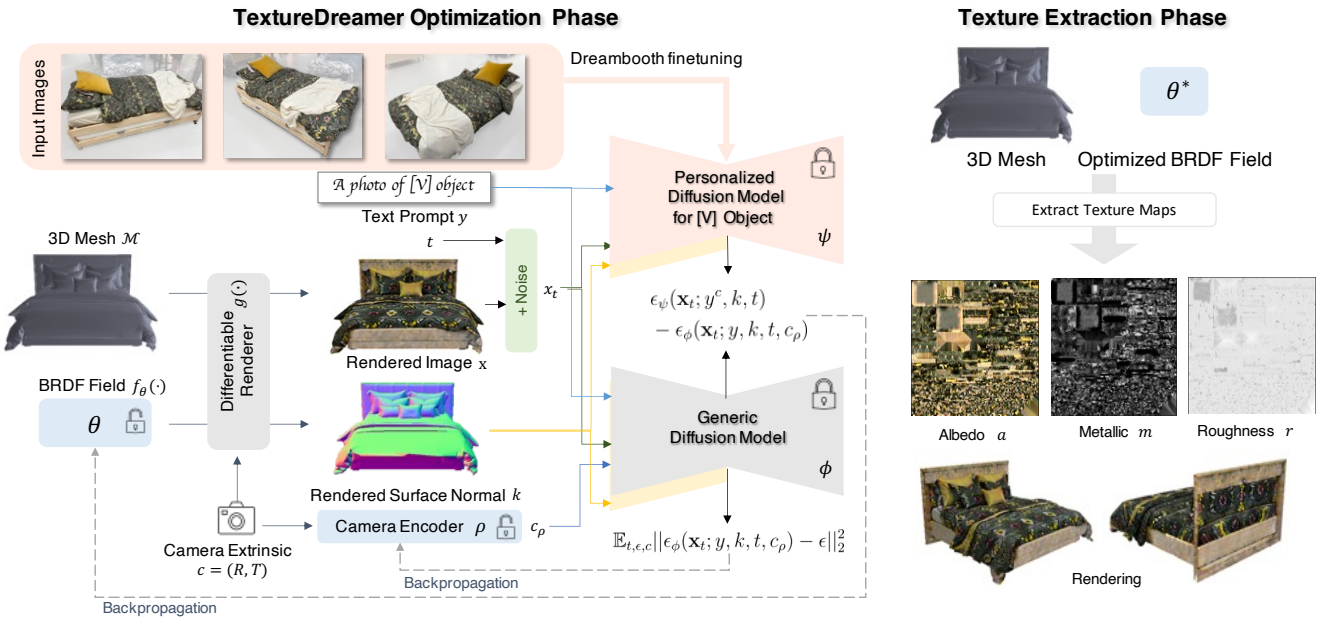

Figure 3. **Overview of TextureDreamer**, a framework which synthesizes texture for a given mesh with appearance similar to 3-5 input images of an object. We first obtain personalized diffusion model ψ with Dreambooth [57] finetuning on input images. The spatially-varying bidirectional reflectance distribution (BRDF) field f_θ for the 3D mesh $\mathcal{M}$ is then optimized through personalized geometric-aware score distillation (PGSD) (detailed in Section 3.2). After optimization finished, high-resolution texture maps corresponding to albedo, metallic, and roughness can be extracted from the optimized BRDF field.

3D representation θ as a random variable and minimizes the KL divergence between θ and the distribution defined by the pre-trained diffusion model. It involves fine-tuning a LoRA [24] network ϵ_ϕ (and a camera encoder ρ which embeds camera pose c as an condition input to ϵ_ϕ) to denoise the noisy images generated from 3D representation θ

$$\min_\phi \mathbb{E}_{t,\epsilon,c}\left[||\epsilon_\phi(\mathbf{x}_t, y, t, c) - \epsilon||_2^2\right] \tag{1}$$

The gradient to the 3D representation θ is then computed as

$$\mathbb{E}_{t,\epsilon,c}\left[w(t)(\epsilon_\psi\left(\mathbf{x}_t, y^c, t\right) - \epsilon_\phi(\mathbf{x}_t, y, t, c))\frac{\partial g(\theta, c)}{\partial \theta}\right]. \tag{2}$$

While VSD significantly improves both visual quality and diversity of generated 3D contents, it cannot address the 3D consistency issue due to the inherent lack of 3D knowledge, leading to multi-face errors and mismatches between geometry and textures. We address this challenge by explicitly injecting geometry information to make our diffusion model geometry aware.

3.2. Personalized Geometry-aware Score Distillation (PGSD)

Problem setup. We illustrate our method in Figure 3. The inputs to our framework include a small set of images (3 to 5) casually captured from different views $\{I\}_{k=1}^K$ and a target 3D mesh $\mathcal{M}$. The outputs of our framework are relightable textures transferred from image set $\{I\}_{k=1}^K$ to $\mathcal{M}$ in a semantically meaningful and visually pleasing manner. Our relightable textures are parameterized as standard microfacet bidirectional reflectance distribution (BRDF) model [25], which consists of 3 parameters, diffuse albedo a, roughness r, and metallic m. We deliberately *do not* optimize normal maps as it encourages the pipeline to fake details that are inconsistent with mesh $\mathcal{M}$. Following the recent trend of neural implicit representation [20, 44, 45], during optimization, we represent our texture as a neural BRDF field $f_\theta(v) : v \in \mathbb{R}^3 \rightarrow a, r, m \in \mathbb{R}^5$, where v is an arbitrary point sampled on the surface of $\mathcal{M}$ and f_θ consists of a multi-scale hash encoding and a small MLP. We find such an implicit representation can better regularize the optimization process, leading to smoother textures. However, given the UV mapping of $\mathcal{M}$, our representation can also be converted to standard 2D texture maps that are compatible with standard graphics pipelines, by querying every 3D point corresponding to each texel, as shown on the right-hand side of Figure 3.

Personalized texture information extraction. We follow Dreambooth [57] to extract texture information from sparse images. To be specific, we fine-tune a personalized diffusion model on input images with a text prompt y, *"A photo of [V] object"*, where *"[V]"* is a unique identifier to de-

scribe the input object. Compared to the alternative textual inversion method [16], we observe that Dreambooth converges faster and can better preserve intricate texture patterns, possibly due to its larger capacity. We first mask out the background of the target object with a white color. For the reconstruction loss, we resize the shorter edge of input images to 512 and randomly crop 512x512 patches for training. We do not apply class-specific prior preservation loss, as we hope our Dreambooth finetuning model can generalize to other categories. We also experiment with different variations, including jointly fine-tuning the text encoder and replacing the diffusion denoising network with a pre-trained ControlNet, but do not observe any improvements.

Geometry-aware score distillation Once we finish extracting texture information with Dreambooth, we transfer the information to mesh $\mathcal{M}$ by adopting the fine-tuned Dreambooth model as the denoising network ϵ_ψ for score distillation sampling. Specifically, we choose VSD instead of the original SDS because of its superior ability to generate highly realistic and diverse appearances. To render images $\mathbf{x}$ for VSD gradient computation, we follow Fantasia3D [10] to pre-select a fixed HDR environment map E as illumination and use Nvdiffrast [30] as our differentiable renderer. We set the object background to be a constant white color to match the input images for Dreambooth training. We observe this can help achieve better color fidelity compared to random color or neutral background.

However, simply replacing SDS with VSD cannot address the limitation of lacking 3D knowledge in 2D diffusion models. We thus propose geometry-aware score distillation, where we inject geometry information extracted from mesh $\mathcal{M}$ into our personalized diffusion model ϵ_ψ through a pre-trained ControlNet conditioned on normal maps k rendered from $\mathcal{M}$. This augmentation significantly boosts 3D consistency of generated textures (see Figure 7). With the ControlNet conditioning, the pillow texture from the input images can be accurately matched to the target shape, despite the shape mismatch. We experiment with different ControlNet conditions and show that normal conditions can best prevent texture-geometry mismatch.

Let $\mathbf{x} = g(\theta, c)$ be the rendered image under a fixed environment map from camera pose c with the extracted BRDF maps $a_\theta, r_\theta, m_\theta$. The gradient of proposed Personalized Geometry-aware Score Distillation (PGSD) to optimize the MLP parameter θ of BRDF field is:

$$\nabla_\theta \mathcal{L}_{\text{PGSD}}(\theta) \triangleq \mathbb{E}_{t,\epsilon,c}[w(t)(\epsilon_\psi(\mathbf{x}_t; y^c, k, t) - \epsilon_\phi(\mathbf{x}_t; y, k, t, c_\rho))\frac{\partial \mathbf{x}}{\partial \theta}],$$

where $\mathbf{x}_t = \alpha_t \mathbf{x} + \sigma_t \epsilon$ is the rendered image $\mathbf{x}$ perturbed by noise $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ at time t, c_ρ is the embedding of the camera extrinsic c encoded by a learnable camera encoder ρ, ϵ_ψ and ϵ_ϕ are the fine-tuned personalized diffusion model and the generic diffusion model pretrained on a large-scale dataset, respectively. Both models are augmented with ControlNet conditioned on normal map k, as shown in the yellow part underneath the diffusion model in Figure 3.

We found that our method does not benefit from classifier-free guidance (CFG) [22], probably because the personalized model ϵ_ψ has been fine-tuned on a small number of images. Since our goal is to faithfully transfer input appearance to target shape, it is not necessary to have CFG to increase the diversity. Similar observation can be found in recent literature [58].

We additionally identify several important design choices through extensive experiments. First, it is important to initialize the ϵ_ϕ in Eq. 1 with original pre-trained diffusion model weights while the Dreambooth weight will remove texture details. This is probably because the Dreambooth fine-tuning process makes the diffusion model overfit to a small training set, as pointed out by previous work [53]. Moreover, we find that removing the LoRA weights can substantially improve texture fidelity. Similar difficulties in training LoRA were also reported in [59]. We therefore implement our personalized geometry-aware score distillation loss $\mathcal{L}_{PGSD}$ by removing the LoRA structure in ϵ_ϕ and only keeping the camera embedding, achieving the best quality. We show more comparisons in Figure 7.

4. Experiment

4.1. Experimental setup

Dataset. We conduct our experiments on 4 categories of objects: sofa, bed, mug/bowl, and plush toy. For each category, we select 8 instances of objects and create a small image set by casually sampling 3 to 5 views surrounding the object, resulting in 32 image sets in total. For every image in the 32 image sets, we apply U2-Net [51] to obtain the foreground mask automatically or use a semi-auto background removal application[1] to obtain more accurate masks. We perform texture transfer for each image set to diverse meshes including but not limited to same category shapes, different category shapes, or even geometry with different genus numbers. To test our texture-transferring framework, we select 3 meshes for each of the 4 categories that are dissimilar to the captured image sets. We acquire these 3D meshes from 3D-FUTURE [15] and online repositories.[2][3]. We run intra-class texture transfer for all 4 categories of objects and also run inter-class texture transfer between bed and chair, to test our method's generalization ability.

Implementation details. We implement our framework based on PyTorch [47] and Threestudio [19]. We use latent diffusion and ControlNet v1.1 as our pre-trained diffusion

[1] https://www.remove.bg/upload
[2] https://www.cgtrader.com/
[3] https://sketchfab.com/

Figure 4. **Image-guided transfer results** from four categories (beds, sofas, plush toys, and mugs) of image sets to diverse objects. Our method can be applied to a wide range of object types and transfer the textures to diverse object shapes.

model and ControlNet respectively. In all our experiments, we set the classifier-free guidance weight of $\mathcal{L}_{PGSD}$ as 1.0 (equivalent to setting $\omega = 0$ in the original CFG formulation). Following DreamFusion [49], we also apply view-dependent conditioning to the input text prompt. The BRDF field is parameterized with an MLP using hash-grid positional encoding [44], following prior works [10, 66]. Our camera encoder consists of two linear layers that project the camera extrinsic to a latent vector of $1,280$ dimensions to be fused with time and text embedding in U-Net. We empirically set the learning rate to 0.01 for encoding, 0.001 for MLP, and 0.0001 for camera encoder for all experiments.

4.2. Baseline methods

Latent-paint [38] and TEXTure [55] are two recent text-guided texturing methods with 2D diffusion prior. They also demonstrate the capability of texturing meshes from images. Latent-paint leverages the Texture Inversion [16] to extract image information into text embedding and distills the texture with SDS. TEXTure first finetunes the pre-trained diffusion model by combining Texture Inversion and Dreambooth [57] and use this fine-tuned model to synthesize texture with an iterative mesh painting algorithm. Following TEXTure, we augment the input images with a random color background. We closely follow the original implementation of baseline methods to run the experiments.

Figure 5. **Example of cross-category texture synthesis results.** Input images (top row) can guide the texture synthesis (bottom row) for shapes which is not in the same category.

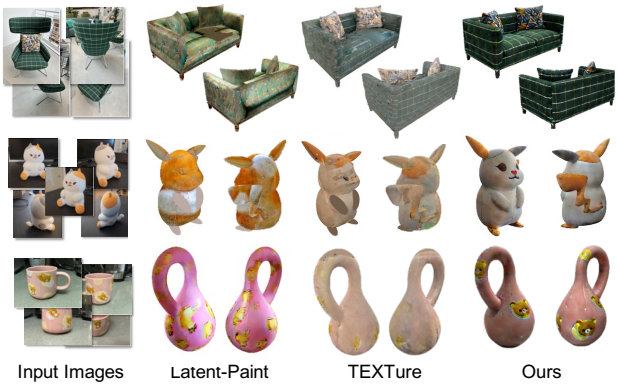

Figure 6. **Comparison between baseline methods.** Compared with Latent-Paint [38] and TEXTure [55], our method can synthesize seamless and geometry-aware textures which are compatible with the target mesh geometry.

Table 1. **User study** on image-guided texture transfer.

	Ours preferred over	
	Latent-Paint	TEXTure
Image Fidelity	71.82%	69.43%
Texture Photorealism	77.03%	85.52%
Shape-Texture Consistency	78.49%	85.16%

Table 2. **Quantitative evaluation** on image-guided texturing.

	CLIP similarity ↑
Latent-Paint [38]	0.7969
TEXTure [55]	0.7988
Ours	**0.8296**

4.3. Image-guided texture transfer

Qualitative evaluation Our method can synthesize geometry-aware and seamless textures that has similar patterns and styles as the input for diverse object geometry. In Figure 4, textures can be synthesized for the *same* category. It can be diverse under different seeds as shown in Figure 10 of supplementary material. We also demonstrate that our method can synthesize textures *across different categories*. In Figure 1, we show texture synthesis results from images of sofa to bed shapes, and vice versa. Our method is also capable of performing texture synthesis across a broader range of different categories. In Figure 5, high-quality and realistic textures can be synthesized across chair, mug, plush toy, or even non-rigid objects such as bags or clothes. It can also be used to synthesize texture for shapes captured from 3D scanner, as shown in Figure 13 of supplementary material.

In Figure 6, we qualitatively compare our method with baseline methods. Two views are shown in each example. Latent-Paint tends to generate textures with colors and patterns that are different from input images. TEXTure can preserve the color and texture better than Latent-Paint, but the texture contains visible seams (possibly due to the iterative painting). Our results method can reason the semantics of the geometry (*e.g.* the positions of eyes) and demonstrate higher quality, seamless, and geometry-aware texturing results with higher fidelity from the input images.

Quantitative evaluation It is non-trivial to perform quantitative comparisons for texture transfer due to the shape difference between geometry and photos. We perform a user study to evaluate transfer fidelity, texture photorealism, and texture-geometry compatibility across baselines by asking users the following questions: 1) Which one has the texture that looks more similar to input images? 2) Which one has a texture which looks more like a real object? 3) Which one has the texture which is more compatible with the meshes? (Which texture painted more fitted to the geometry?) We conduct a user study with Amazon Turk with three separate tasks. For each task, we ask each user 24 questions. Each question is a forced single-choice selection with two options among our and one baseline result with the rendered images from the same 4 sampled views and is evaluated by 20 different users. We only show input photos for the first similarity question, and hide the input photos for the other two questions to make the user focus on texture quality. We summarize the results in Table 1. Our results show significant preference by the users in terms of image fidelity, texture photorealism, and shape-texture consistency.

We also evaluate the similarity via image-based CLIP feature [42] between reference and the rendered images of synthesized textures. The CLIP similarity has been ap-

Figure 7. **Ablation study.** With ControlNet conditioned on normal maps, the result has the best texture-geometry consistency. Without ControlNet or with depth-based ControlNet, the results suffer from texture-geometry misalignment. Using SDS loss leads to blurry textures. Without the LoRA module removed, the results tend to remove the existing texture from the personalized diffusion model. Our full method can synthesize accurate texture which is similar to input appearances.

plied to material matching [68] and stylization [39]. A good transfer should transfer only the texture from images and should take into account the target shape geometry and transfer the texture semantically. For example, the transfer should be painted with respect to each part of the shape. We use our evaluation set to compute the comparison. For each image set and target 3D mesh pair, we compute the average of the metric among each reference image and each of rendered image from 4 sampled views (*i.e.* left front, right front, left back, and right back). We average the CLIP similarity across all (image set, mesh) pairs. Table 2 shows our method has the highest CLIP similarity.

Ablation study We qualitatively perform an ablation study in Figure 7. The results suffer from geometry-texture misalignment without ControlNet or the depth-based ControlNet. Only normal-based ControlNet can accurately control the synthesized texture to be consistent with the input mesh geometry. We validate the importance of score distillation sampling. Only using SDS in our framework cannot achieve enough fidelity and the result tends to be blurry. Without LoRA removed (which is usually optimized with vanilla VSD), the optimization tends to make the distribution diverge from the personalized diffusion model. It makes the output contain less original texture but more irrelevant patterns from the input. We hypothesize that optimizing LoRA weights with a text condition containing a rare identifier tends to drive the distribution of rendered images to have a rare appearance. Additional ablation study is provided in the supplementary material in Figure 14 and Table 3.

Number of images We evaluate various number of input images. In Figure 8, a single image cannot provide sufficient information for texturing, while using 11 images doesn't show advantages in terms of texture quality.

Figure 8. **Number of images.** Input images are from Figure 6.

Figure 9. **Limitations.** Our method may bake-in lighting into texture, have Janus problem when lacking enough input viewpoints, and ignore special and non-repeated patterns from the input.

5. Discussions

We proposed a framework to transfer high-quality texture from input images to an arbitrary shape. There are still some limitations, as shown in Figure 9. Our method may not be able to transfer special and non-repeated texture to the target shapes. Our method tends to bake in lighting to texture when there are strong specular highlights in the input images. Janus problem might appear when the viewpoints of input images do not cover the entire object. Nevertheless, we believe that our method can be the first step to tackling this challenging problem and will make an impact on the 3D content creation community.

Acknowledgement We thank Google PhD Fellowship supports the research.

References

[1] Adobe substance 3d. https://docs.substance3d.com/sat. 2

[2] Badour AlBahar, Shunsuke Saito, Hung-Yu Tseng, Changil Kim, Johannes Kopf, and Jia-Bin Huang. Single-image 3d human digitization with shape-guided diffusion. In *SIGGRAPH Asia*, 2023. 3

[3] Sai Bi, Nima Khademi Kalantari, and Ravi Ramamoorthi. Patch-based optimization for image-based texture mapping. *ACM Trans. Graph.*, 36(4):106–1, 2017. 2

[4] Alexey Bokhovkin, Shubham Tulsiani, and Angela Dai. Mesh2tex: Generating mesh textures from image queries. *arXiv preprint arXiv:2304.05868*, 2023. 2

[5] G. Cai, K. Yan, Z. Dong, I. Gkioulekas, and S. Zhao. Physics-based inverse rendering using combined implicit and explicit geometries. *Computer Graphics Forum*, 41(4): 129–138, 2022. 3

[6] Tianshi Cao, Karsten Kreis, Sanja Fidler, Nicholas Sharp, and Kangxue Yin. Texfusion: Synthesizing 3d textures with text-guided image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4169–4181, 2023. 3

[7] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16123–16133, 2022. 3

[8] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 1

[9] Dave Zhenyu Chen, Yawar Siddiqui, Hsin-Ying Lee, Sergey Tulyakov, and Matthias Nießner. Text2tex: Text-driven texture synthesis via diffusion models. *arXiv preprint arXiv:2303.11396*, 2023. 3

[10] Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. *arXiv preprint arXiv:2303.13873*, 2023. 3, 5, 6

[11] Zhiqin Chen, Kangxue Yin, and Sanja Fidler. Auv-net: Learning aligned uv maps for texture transfer and synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1465–1474, 2022. 2

[12] Christopher B Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII 14*, pages 628–644. Springer, 2016. 1

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

[14] Ziya Erkoç, Fangchang Ma, Qi Shan, Matthias Nießner, and Angela Dai. Hyperdiffusion: Generating implicit neural fields with weight-space diffusion. *arXiv preprint arXiv:2303.17015*, 2023. 3

[15] Huan Fu, Rongfei Jia, Lin Gao, Mingming Gong, Binqiang Zhao, Steve Maybank, and Dacheng Tao. 3d-future: 3d furniture shape with texture. *International Journal of Computer Vision*, 129:3313–3337, 2021. 5

[16] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit Haim Bermano, Gal Chechik, and Daniel Cohen-or. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *The Eleventh International Conference on Learning Representations*, 2022. 2, 5, 6

[17] Jun Gao, Tianchang Shen, Zian Wang, Wenzheng Chen, Kangxue Yin, Daiqing Li, Or Litany, Zan Gojcic, and Sanja Fidler. Get3d: A generative model of high quality 3d textured shapes learned from images. *Advances In Neural Information Processing Systems*, 35:31841–31854, 2022. 3

[18] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 1

[19] Yuan-Chen Guo, Ying-Tian Liu, Ruizhi Shao, Christian Laforte, Vikram Voleti, Guan Luo, Chia-Hao Chen, Zi-Xin Zou, Chen Wang, Yan-Pei Cao, and Song-Hai Zhang. threestudio: A unified framework for 3d content generation. https://github.com/threestudio-project/threestudio, 2023. 5

[20] Jon Hasselgren, Nikolai Hofmann, and Jacob Munkberg. Shape, light, and material decomposition from images using monte carlo rendering and denoising. *Advances in Neural Information Processing Systems*, 35:22856–22869, 2022. 4

[21] Paul Henderson, Vagia Tsiminaki, and Christoph H Lampert. Leveraging 2d data to learn textured 3d mesh generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7498–7507, 2020. 2

[22] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 3, 5

[23] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 2, 3

[24] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 4

[25] Brian Karis and Epic Games. Real shading in unreal engine 4. *Proc. Physically Based Shading Theory Practice*, 4(3):1, 2013. 4

[26] Animesh Karnewar, Andrea Vedaldi, David Novotny, and Niloy J Mitra. Holodiffusion: Training a 3d diffusion model using 2d images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18423–18433, 2023. 3

[27] Oren Katzir, Or Patashnik, Daniel Cohen-Or, and Dani Lischinski. Noise-free score distillation. *arXiv preprint arXiv:2310.17590*, 2023. 3

[28] Johannes Kopf, Chi-Wing Fu, Daniel Cohen-Or, Oliver Deussen, Dani Lischinski, and Tien-Tsin Wong. Solid tex-

ture synthesis from 2d exemplars. In *ACM SIGGRAPH 2007 papers*, pages 2–es. 2007. 2

[29] Vivek Kwatra, Arno Schödl, Irfan Essa, Greg Turk, and Aaron Bobick. Graphcut textures: Image and video synthesis using graph cuts. *Acm transactions on graphics (tog)*, 22 (3):277–286, 2003. 2

[30] Samuli Laine, Janne Hellsten, Tero Karras, Yeongho Seol, Jaakko Lehtinen, and Timo Aila. Modular primitives for high-performance differentiable rendering. *ACM Transactions on Graphics*, 39(6), 2020. 5

[31] Jiabao Lei, Yabin Zhang, Kui Jia, et al. Tango: Text-driven photorealistic and robust 3d stylization via lighting decomposition. *Advances in Neural Information Processing Systems*, 35:30923–30936, 2022. 2

[32] Marc Levoy, Kari Pulli, Brian Curless, Szymon Rusinkiewicz, David Koller, Lucas Pereira, Matt Ginzton, Sean Anderson, James Davis, Jeremy Ginsberg, et al. The digital michelangelo project: 3d scanning of large statues. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 131–144, 2000. 2

[33] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation, 2022. 2

[34] Zhengqin Li, Yu-Ying Yeh, and Manmohan Chandraker. Through the looking glass: Neural 3d reconstruction of transparent shapes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1262–1271, 2020. 1

[35] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 300–309, 2023. 2, 3

[36] Shitong Luo and Wei Hu. Diffusion probabilistic models for 3d point cloud generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2837–2845, 2021. 3

[37] Yiwei Ma, Xiaoqing Zhang, Xiaoshuai Sun, Jiayi Ji, Haowei Wang, Guannan Jiang, Weilin Zhuang, and Rongrong Ji. X-mesh: Towards fast and accurate text-driven 3d stylization via dynamic textual guidance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2749–2760, 2023. 2

[38] Gal Metzer, Elad Richardson, Or Patashnik, Raja Giryes, and Daniel Cohen-Or. Latent-nerf for shape-guided generation of 3d shapes and textures. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12663–12673, 2023. 2, 3, 6, 7

[39] Oscar Michel, Roi Bar-On, Richard Liu, Sagie Benaim, and Rana Hanocka. Text2mesh: Text-driven neural stylization for meshes. *arXiv preprint arXiv:2112.03221*, 2021. 8

[40] Oscar Michel, Roi Bar-On, Richard Liu, Sagie Benaim, and Rana Hanocka. Text2mesh: Text-driven neural stylization for meshes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13492–13502, 2022. 2

[41] 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. *Communications of the ACM*, 65(1):99–106, 2021. 1

[42] Nasir Mohammad Khalid, Tianhao Xie, Eugene Belilovsky, and Tiberiu Popa. Clip-mesh: Generating textured meshes from text using pretrained image-text models. In *SIGGRAPH Asia 2022 conference papers*, pages 1–8, 2022. 2, 7

[43] Joep Moritz, Stuart James, Tom SF Haines, Tobias Ritschel, and Tim Weyrich. Texture stationarization: Turning photos into tileable textures. In *Computer graphics forum*, pages 177–188. Wiley Online Library, 2017. 2

[44] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)*, 41(4):1–15, 2022. 1, 4, 6

[45] Jacob Munkberg, Jon Hasselgren, Tianchang Shen, Jun Gao, Wenzheng Chen, Alex Evans, Thomas Müller, and Sanja Fidler. Extracting triangular 3d models, materials, and lighting from images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8280–8290, 2022. 4

[46] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 165–174, 2019. 1

[47] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. 5

[48] Dario Pavllo, Jonas Kohler, Thomas Hofmann, and Aurelien Lucchi. Learning generative models of textured 3d meshes from real-world images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13879–13889, 2021. 2

[49] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022. 2, 3, 6

[50] Guocheng Qian, Jinjie Mai, Abdullah Hamdi, Jian Ren, Aliaksandr Siarohin, Bing Li, Hsin-Ying Lee, Ivan Skorokhodov, Peter Wonka, Sergey Tulyakov, et al. Magic123: One image to high-quality 3d object generation using both 2d and 3d diffusion priors. *arXiv preprint arXiv:2306.17843*, 2023. 3

[51] Xuebin Qin, Zichen Zhang, Chenyang Huang, Masood Dehghan, Osmar Zaiane, and Martin Jagersand. U2-net: Going deeper with nested u-structure for salient object detection. page 107404, 2020. 5

[52] 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. 2

[53] Amit Raj, Srinivas Kaza, Ben Poole, Michael Niemeyer, Nataniel Ruiz, Ben Mildenhall, Shiran Zada, Kfir Aberman, Michael Rubinstein, Jonathan Barron, et al. Dreambooth3d: Subject-driven text-to-3d generation. *arXiv preprint arXiv:2303.13508*, 2023. 2, 3, 5

[54] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1 (2):3, 2022. 2, 3

[55] 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*, New York, NY, USA, 2023. Association for Computing Machinery. 2, 3, 6, 7

[56] Carlos Rodriguez-Pardo and Elena Garces. Seamless-gan: Self-supervised synthesis of tileable texture maps. *IEEE Transactions on Visualization and Computer Graphics*, 2022. 2

[57] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22500–22510, 2023. 2, 3, 4, 6

[58] Prafull Sharma, Varun Jampani, Yuanzhen Li, Xuhui Jia, Dmitry Lagun, Fredo Durand, William T Freeman, and Mark Matthews. Alchemist: Parametric control of material properties with diffusion models. *arXiv preprint arXiv:2312.02970*, 2023. 5

[59] Ruoxi Shi, Hansheng Chen, Zhuoyang Zhang, Minghua Liu, Chao Xu, Xinyue Wei, Linghao Chen, Chong Zeng, and Hao Su. Zero123++: a single image to consistent multi-view diffusion base model, 2023. 5

[60] 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. 3

[61] Yawar Siddiqui, Justus Thies, Fangchang Ma, Qi Shan, Matthias Nießner, and Angela Dai. Texturify: Generating textures on 3d shape surfaces. In *European Conference on Computer Vision*, pages 72–88. Springer, 2022. 2

[62] 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. 1, 2, 3

[63] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. 2, 3

[64] Cheng Sun, Guangyan Cai, Zhengqin Li, Kai Yan, Cheng Zhang, Carl Marshall, Jia-Bin Huang, Shuang Zhao, and Zhao Dong. Neural-pbir reconstruction of shape, material, and illumination. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 3

[65] 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, 3

[66] Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *arXiv preprint arXiv:2305.16213*, 2023. 2, 3, 6

[67] Jiajun Wu, Chengkai Zhang, Tianfan Xue, Bill Freeman, and Josh Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. *Advances in neural information processing systems*, 29, 2016. 1

[68] K. Yan, F. Luan, M. Hašan, T. Groueix, V. Deschaintre, and S. Zhao. Psdr-room: Single photo to scene using differentiable rendering. In *ACM SIGGRAPH Asia 2023 Conference Proceedings*, 2023. 8

[69] Xin Yu, Peng Dai, Wenbo Li, Lan Ma, Zhengzhe Liu, and Xiaojuan Qi. Texture generation on 3d meshes with point-uv diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4206–4216, 2023. 3

[70] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023. 2, 3

[71] Qian-Yi Zhou and Vladlen Koltun. Color map optimization for 3d reconstruction with consumer depth cameras. *ACM Transactions on Graphics (ToG)*, 33(4):1–10, 2014. 2

[72] 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. 2