VideoMatGen: PBR Materials through Joint Generative Modeling

Jon Hasselgren NVIDIA
Miloš Hašan NVIDIA
Zheng Zeng NVIDIA
Jacob Munkberg NVIDIA

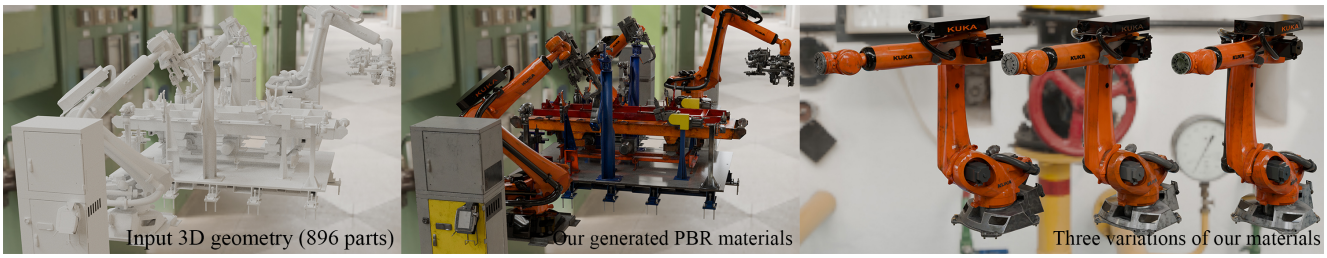

Figure 1. Given 3D models and text prompts, we generate unique high quality PBR materials for each 3D part using a finetuned video diffusion model. Our generated materials are directly applicable in content creation applications. Here we show a Physical AI training application, applying the generated materials to a virtual factory setting. On the right, we show three variations of generated materials (from the same detailed text prompts and different random seeds) for an industrial robot asset with 19 parts.

Abstract

We present a method for generating physically-based materials for 3D shapes based on a video diffusion transformer architecture. Our method is conditioned on input geometry and a text description, and jointly models multiple material properties (base color, roughness, metallicity, height map) to form physically plausible materials. We further introduce a custom variational auto-encoder which encodes multiple material modalities into a compact latent space, which enables joint generation of multiple modalities without increasing the number of tokens. Our pipeline generates high-quality materials for 3D shapes given a text prompt, compatible with common content creation tools.

1. Introduction

Manually authoring 3D assets is time-consuming and requires expert skills; using generative models to produce 3D assets is a promising alternative. A new research field of leveraging diffusion models to generate 3D models from text prompts has recently emerged [16, 39, 58, 71]. Another line of work assumes an input untextured 3D shape and generates texture through multi-view applications of image diffusion models [13, 15, 18, 46, 61]. While the results look impressive for novel view synthesis, the methods bake final RGB colors (under some lighting) into the asset and cannot extract materials for physically-based rendering (PBR) [4, 54], critical in more advanced content creation workflows. Fitting these properties through differentiable rendering is possible, but in addition to unknown lighting, image diffusion models typically lack perfect view-consistency, introducing blur and washing out material details. This is particularly obvious when optimizing parameters for PBR material models, which rely on consistency of specular reflections.

Video diffusion models provide improved view consistency and exceed image models in handling specular highlights. This greatly helps when estimating per-pixel material parameters and for intrinsic decomposition [31]. This decomposition approach is utilized in recent work, VideoMat [36], to generate a video orbit around a given 3D shape with synthesized final RGB appearance, and finally extract material parameters from this video using intrinsic decomposition. While the results are promising, their quality is limited by unnecessarily solving two hard problems that cancel each other out: synthesizing final appearance under natural lighting, and then removing the lighting to estimate clean material parameters.

We present VideoMatGen, a video diffusion method for direct text-to-material generation. Our work extends VideoMat [36] to generate higher quality materials using a more efficient fused architecture based on joint generative modeling, without relying on an intermediate RGB appearance. We start from a known untextured 3D geometry and a text prompt describing the desired material. We condition a video diffusion model on multiple views of geometry guides (G-buffers): surface normals and world space positions.

By fine-tuning a recent video model, Cosmos Predict 1-7B [37], with a custom dataset mapping these conditions and text prompts to material parameters, we generate video sequences of synthesized intrinsic material channels (G-buffers): *base color*, *roughness*, *metallicity*, and *height*. Finally, the resulting views are projected into traditional texture maps, optionally turning height into normal variation (though the height could also be used as displacement). As shown in Fig. 1 we produce spatially varying, detailed materials that adapt to the underlying geometry. In comparison to related work, we show higher quality results and improved separation of lighting and materials. Our main contributions are:

- A video diffusion method for generating physically-based materials for 3D shapes based on text prompts, jointly predicting base color, roughness, metallicity, and height.
- A unified variational auto-encoder and latent space, jointly encoding base color, roughness, metallicity and height. This enables improved joint prediction without increasing the number of tokens.

2. Related Work

Diffusion Models. Image diffusion models add random noise to an image through a sequence of diffusion steps. They are trained to reverse this process, enabling sample generation by iterative denoising starting from Gaussian noise. Many generative models have been developed based on similar principles [12, 20, 48]. Recently, video diffusion models [1, 2, 21, 37, 62] extend image-based diffusion approaches to the temporal domain, enabling video generation from inputs such as text or an initial frame. Diffusion transformer (DiT) models have become the standard architecture of choice for both image and video diffusion [38] due to their performance and flexible finetuning opportunities. In this work, we build upon the Cosmos [37] DiT-based video diffusion model.

Differentiable Rendering. In this paper, we focus on mesh-based surface geometry with PBR materials [4]. Previous work includes differentiable rasterization [29], which has low run-time cost and has been successfully applied to photogrammetry [35]. Differentiable path tracing [22, 70] approaches are considerably more costly, and introduce Monte-Carlo noise in the training process, which can make gradient-based optimization more challenging. However, path tracing accurately simulates global illumination effects, and has higher potential reconstruction quality. Fuzzy scene representations such as NeRFs [34] and Gaussian splatting [25] are commonly used in optimization setups, and generate impressive novel-view synthesis results. However, disentangling materials and lighting remains non-trivial. We assume known mesh geometry, but our approach can be extended to generate materials on other geometry representations (e.g. Gaussians, SDFs, etc.).

Texture and material extraction using diffusion. Various hybrid approaches combine image diffusion models with inpainting, or coarse-to-fine texture refinement, such as TEXTure [42], Text2tex [7], and Paint3D [68]. Paintit [64] proposes representing material texture maps with randomly initialized convolution-based neural kernels. This regularizes the optimization landscape, improving material quality. TextureDreamer [63] finetunes the diffusion model using Dreambooth [43] with a few images of a 3D object, and uses variational score distillation [55] to optimize the material maps. DreamMat [75] and FlashTex [11] improve on light and material disentanglement by finetuning image diffusion models to condition on geometry and lighting, allowing for optimization over many known lighting conditions.

MaPa [74], MatAtlas [5], and Make-it-Real [14] start from a database of known high-quality materials, and learn to project the input (image or text) onto the known representation. MaPa relies on material graphs and optimize parameters of known graphs, while Make-it-Real uses a database of PBR-textures, and MatAtlas a database of procedural materials. These methods are limited by the expressiveness of their material databases, but benefit from much improved regularization.

Diffusion-based 3D asset generation. Many methods build on image diffusion models to produce full 3D assets, with either RGB colors or PBR material maps. DreamFusion [39] introduces a *score distillation sampling* (SDS) loss, and generates 3D assets from pre-trained text-to-image diffusion models. This approach has since been refined [55, 77, 77]. SDS-based methods require slow optimization, prompting the development of methods like Instant3D [30] and GS-LRM [71] that instead reconstruct in a forward pass using a single pretrained transformer model.

A common limitation for most image models is lack of view consistency, which may show up as blur in the extracted textures. SV3D [53] and Hi3D [60] improve on this aspect by finetuning video models for object rotations, and extract 3D models from the generated views. However, these approaches have limited resolution and do not provide PBR materials. Trellis [57] and TEXGen [65] avoid the view consistency problem altogether by having the diffusion model operate directly in 3D space and texture space respectively. These methods show great promise, but they do not focus on material parameter generation. CLAY [72] and SF3D [3] also generate 3D geometry and materials from text or image inputs. CLAY's material generation models uses a finetuned multi-view image diffusion model [47] conditioned on normal maps. The material model generates

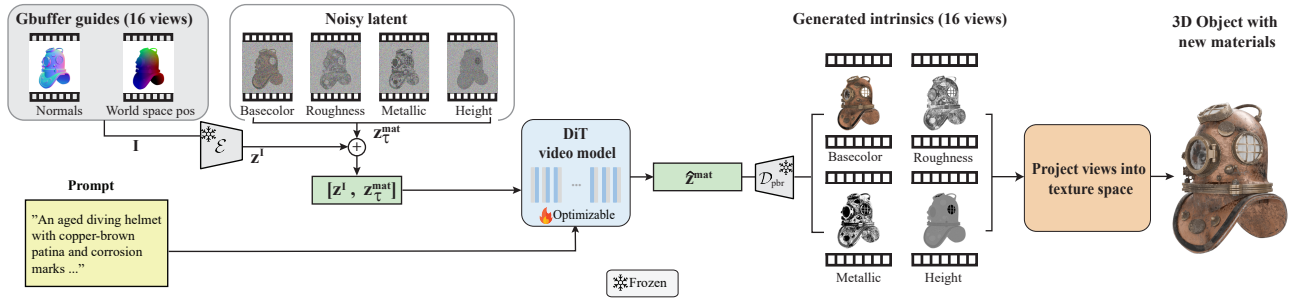

Figure 2. Our method starts from a known 3D model and a text prompt. We first render videos of normal maps and world space positions. Next, these conditions are encoded into latent space, using a pretrained encoder, $\mathcal{E}$, to produce latent conditions, $\mathbf{z}^{\mathbf{I}}$. These are concatenated with noisy latents, $\mathbf{z}_{\tau}^{\mathbf{mat}}$, representing material modalities, along the channel dimension. The latents and text prompt are then passed to our finetuned video model, which generates a denoised latent, $\hat{\mathbf{z}}^{\mathbf{mat}}$. The denoised latent is decoded into videos of the intrinsic material channels: base color, roughness, metallicity, and height, using a custom VAE decoder $\mathcal{D}_{\text{pbr}}$ which decodes all material properties jointly. Finally, we project the generated views into texture space to extract high quality, standard PBR materials.

four canonical views of the PBR texture maps (base color, roughness, metallicity), which are then projected into texture space. Several recent methods [13, 15, 18, 44, 46, 61] extends this approach with additional input conditioning (normal, depth and/or world space positions). 3DTopia-XL [8] proposes a novel 3D representation, which encodes the 3D shape, textures, and materials in volumetric primitives anchored to the surface of the object. Their denoising process jointly generates shape and PBR materials.

Intrinsic decomposition of images/videos. Another related line of research is intrinsic decomposition of images, which is closely related to per-pixel material parameter estimation. IntrinsicAnything [56] decomposes images into diffuse and specular components, and leverages these components as priors using physically-based inverse rendering to extract material maps. MaterialFusion [32] introduces a 2D diffusion model prior to help estimate material parameters in an multi-view reconstruction pipeline. RGB↔X [69] uses finetuned diffusion models for both intrinsic decomposition of images into G-buffers and the neural rendering of images from G-buffers. DiffusionRenderer [31] extends RGB↔X to videos, and also supports relighting. NeuralGaffer [24] and DiLightNet [67] leverage diffusion models for relighting single views. IllumiNerf [76] relights each view in a multi-view dataset, then reconstructs a NeRF model with these relit images. IntrinsiX [27] combines intrinsic predictions for PBR G-buffers for a single view from text (using image diffusion models) with a rendering loss. MCMat [78] leverages Diffusion Transformers (DiT) to extract multi-view images of PBR material maps, combined with a second DiT to enhance details in UV space.

VideoMat [36], the closest related work to ours, generates materials for 3D shapes by first generating an RGB video of a textured and lit 3D model conditioned on untextured geometry, and then extracting the material parameters by combining video intrinsic decomposition and differentiable rendering to project the material parameters into texture space.

Joint generative modeling approaches enable diffusion models to predict multiple modalities. Matrix3D [33] predicts pose estimation, depth, and novel view synthesis using a single DiT [38] model. VideoJAM [6] extends this by predicting both generated pixels and their corresponding motion from a single DiT. UniRelight [17] leverages this approach to jointly predict relit and base color videos.

3. Method

Our pipeline, as shown in Fig. 2, uses join generative modeling with video diffusion models to produce PBR material textures. We assume a given 3D model with a valid texture parameterization (but no textures) as input. We generate multiple views of material intrinsics: G-buffers of *base color*, *roughness*, *metallicity*, and *height* values, conditioned on corresponding input geometry (views of surface normals and world space positions). Finally, we project the intrinsic views into texture space to obtain standard PBR materials directly compatible with common 3D authoring tools: Blender, Unreal Engine, etc. Below, we describe each step in detail.

3.1. Base Video Model Architecture

In a first step, we produce a synthetic dataset consisting of multiple views of material intrinsics, conditioned on geometry (surface normals and world space positions for each view) and a text prompt describing each object's material. We use this data to finetune a recent Diffusion Transformer (DiT) video model, Cosmos [37], for this task. We use the `Cosmos-1.0-Diffusion-7BVideo2World`[1] model

[1]https://github.com/NVIDIA/Cosmos

which is trained in a latent space with $8\times$ compression in the spatial and temporal domain. This model supports text- and image guided video generation at a resolution of 1280×704 pixels and 121 frames. The base model leverages the pretrained `Cosmos-1.0-Tokenizer-CV8x8x8` to encode and decode RGB videos to and from latent space. We directly use this encoder to encode our input conditions, but introduce a novel tokenizer to jointly compress the material modalities.

3.2. Per-frame encoding

The temporal compression of the Cosmos Tokenizer [37] encoder, $\mathcal{E}$, introduces some motion blur in the reconstructed frames. To avoid this, we use the image (keyframe) mode, which encodes each frame individually, so our latents only have $8\times$ spatial compression. In other words, we opted for encoded videos with fewer, but higher quality, frames. Specifically, we encode an input video with F frames, $C = 6$ channels, and spatial resolution $H \times W$, represented a tensor $F \times C \times H \times W$ into a latent space with dimensions $F \times 16 \times H/8 \times W/8$. Furthermore, a typical video VAE is trained on mostly coherent videos with limited motion between frames; we encode each frame individually, so we do not need to adhere to this constraint, and we pick a random camera view for each frame in each training example.

3.3. Joint generative modeling

Our goal is to jointly predict spatially varying *base color*, *roughness*, *metallicity*, and *height* material parameters, conditioned on positions and normals of the input 3D model. Unlike recent neural inverse renderers [31, 69] which predict one modality at a time in separate inference passes, we instead follow the approach in recent joint generative modeling approaches [6, 17, 33] to predict multiple modalities in a single inference pass.

UniRelight [17] jointly predicts a relit video and base color by concatenating latents for the two modalities along the *frame* dimension. In contrast, we leverage a custom variational auto-encoder (VAE), which encodes all material modalities into a shared latent space. This way we obtain a VAE specialized for the material domain, while avoiding the increased token length from frame concatenation.

Recent work in neural texture compression [51] shows that multiple material maps can be efficiently compressed together as the maps often contain correlated details. We explore if this is also applicable to VAEs. More precisely, we leverage the pretrained Cosmos Tokenizer [37], which bidirectionally maps between RGB images ($3 \times H \times W$ tensors) and a latent representation using an encoder-decoder pair, $(\mathcal{E}, \mathcal{D})$. We use the image (keyframe) VAE encoding mode. We make minimal changes to the base model, only updating the channel count for the input layer of the encoder and output layer of the decoder, and perform finetuning to create our VAE_{pbr} which maps a $6 \times H \times W$ tensor (*base color*, *roughness*, *metallicity*, and *height*) to latents of the same size as the basemodel. We leverage the latent space produced by VAE_{pbr} in the diffusion process to jointly predict frames of material parameters for all views, as is shown in Fig. 2.

3.4. Finetuning

We finetune the embedding layer (extended from the base model to support our input conditions) and all DiT layers for 20k iterations on 64 A100 GPUs.

Given an input video $\mathbf{I}$ consisting of normals and world space positions for N views of a 3D model, our goal is to train a model $\mathbf{f}_\theta$ that *jointly* denoises views of PBR material maps conditioned on $\mathbf{I}$.

The model comprises a VAE encoder-decoder pair (the Cosmos Tokenizer), $(\mathcal{E}, \mathcal{D})$, and a transformer-based denoising function, $\mathbf{f}_\theta$. We use the encoder $\mathcal{E}$ to encode the input conditions, $\mathbf{I}$, into a latent tensor, $\mathbf{z}^{\mathbf{I}}$.

Our model is finetuned on a synthetic video dataset. Each data sample consists of 16 random object-centric camera views of a 3D objects. Each view includes G-buffers of normals, depth, base color, roughness, metallicity, height values, and the camera pose. We use the depth and camera pose to compute a world space position buffer in the data loader.

The target latent variable, $\mathbf{z}_0^{\mathbf{mat}}$, for this dataset is constructed by encoding the base color, roughness, metallicity, and height values *jointly* (six channels) using our VAE_{pbr} encoder, $\mathcal{E}_{\text{pbr}}$. Noise, ϵ, is introduced to our latent, $\mathbf{z}_0^{\mathbf{mat}}$, representing the material parameters, to produce $\mathbf{z}_\tau^{\mathbf{mat}}$. The model parameters, θ, of the diffusion model, $\mathbf{f}_\theta$, are optimized by minimizing the objective function:

$$\begin{aligned} \hat{\mathbf{z}}^{\mathbf{mat}}(\theta) &= \mathbf{f}_\theta([\mathbf{z}_\tau^{\mathbf{mat}}, \mathbf{z}^{\mathbf{I}}]; \mathbf{c}_{\text{prompt}}, \tau) \\ \mathcal{L}(\theta) &= \mathbb{E}_{\mathbf{z}_0^{\mathbf{mat}} \sim p_{\text{data}}, \epsilon \sim \mathcal{N}(0, \sigma^2 I)} \left\| \hat{\mathbf{z}}^{\mathbf{mat}}(\theta) - \mathbf{z}_0^{\mathbf{mat}} \right\|_2^2, \end{aligned}$$

where $[\cdot]$ denotes concatenation in the channel dimension and $\mathbf{c}_{\text{prompt}}$ is the encoded text prompt (encoded using T5-XXL [40]). We increase the input feature count of the input embedding layer of $\mathbf{f}_\theta$ to account for our additional input conditions, $\mathbf{z}^{\mathbf{I}}$.

We use the denoising score matching loss from Cosmos [37] unmodified, applied to the predicted latent $\hat{\mathbf{z}}^{\mathbf{mat}}(\theta)$ and the corresponding target latent $\mathbf{z}_0^{\mathbf{mat}}$.

Dataset Our dataset consists of 60k videos of object-centric renderings of 3D models from Objaverse [10], BlenderVault [32], ABO [9], and HSSD [26]. For each object, we render a video with 16 frames at a resolution of 1024×1024, using a path tracer with three bounces and

Blender AgX tonemapping. We use black background, and for each frame the view is randomized. For lighting, we use the "BoilerRoom" light probe from Poly Haven [66], providing constant, neutral lighting for all objects. We only use the shaded video to automatically generate captions using Qwen2.5-VL-7B [49], and want to avoid prompt noise due to variation in lighting. We also render intrinsic maps (normals, world space positions, base color, roughness, metallicity, height). The height map is not available for most assets, and we reconstruct it from the normal map using standard conversion tools when available. We augment the dataset by randomly reversing the video, and randomly offsetting the video start frame in each training iteration.

We additionally use this dataset to finetune our VAE. To avoid biasing too heavily towards objects on a black background, we additionally use the MatSynth [52] training set (which contains all material channels expected by our model) and randomly pick samples using a $60/40$ distribution.

3.5. Transfer multi-view intrinsics to texture space

At inference, we generate 16 views of the material intrinsics from known cameras. To extract material maps in texture space, which is the standard format in content creation tools, we project the intrinsic views into texture space using a splatting approach. We upscale the generated views to a resolution of 16k $\times$ 16k pixels and render a texture coordinate guide using the 3D asset, assuming a known, non-overlapping UV-mapping. Each pixel is splatted to the corresponding (nearest neighbor) texel of a 2048 $\times$ 2048 texture with a weight inversely proportional to the screen space texture derivatives [19] to suppress areas with high perspective distortion. More formally, given texture coordinates (u, v) for a pixel (x, y) the weight is computed as:

$$w = \frac{1}{\max\left(\left|\left(\partial u/\partial x, \partial v/\partial x\right)\right|, \left|\left(\partial u/\partial y, \partial v/\partial y\right)\right|\right)}.$$

We normalize the final texture by the total weight per texel, and perform basic inpainting [50] for all texels with zero weight to reduce texture atlas seams.

3.6. Image-conditioned video generation

While our primary focus is on material generation from text, our pipeline can be straightforwardly extended to add image conditioning. We adopt an approach similar to Gen3C [41] where the video model is conditioned on a single shaded input image, which is warped (using a provided depth buffer) according to the known camera matrix and intrinsics for each view. As in our text-to-video setting, we condition the video model on normals and world space positions, and simply concatenate the warped shaded images to the condition, $\mathbf{I}$, with no further changes needed. We argue that both forms of conditioning are useful in production workflows, as reference images are not always available.

4. Results

We evaluate our method against VideoMat [36], a material generation method which also leverages DiT video models. To make comparisons easier, we use the same pretrained base video model as VideoMat throughout this paper. However, we note that our method will directly benefit from a stronger base model. As a representative example of recent multi-view diffusion material generation techniques, we chose Hunyuan3D-Paint 2.1 [18] and MVPainter [46], which both are image-guided material generation methods. We also note that image-conditioned models can be repurposed for text conditioning by an additional text-to-image step. Therefore, we also constructed a text-guided version of Hunyuan3D-Paint and MVPainter by first generating an image from the text prompt using a depth-guided Flux ControlNet [45], and feeding it as an image condition into Hunyuan3D-Paint and MVPainter. We include TRELLIS.2 [59] as an image-conditioned method generating materials directly in 3D space (using their PBR texture generation mode with known geometry). There is a plethora of recent multi-view diffusion methods, and we refer the reader to the concurrent commercial approach Seed3D [44] for extensive comparisons; however, their model has not been released.

4.1. Quantitative evaluation

We report quantitative results on material generation in Tab. 1. There are no established metrics for text-to-material generation quality; therefore, we repurposed image-based metrics as follows. We choose 32 test assets, render images of the assets with their original material assignments, and annotate them with text captions using Qwen2.5-VL-7B (similarly to training assets). We generate materials using the estimated text prompts using all methods for these 32 test models. For the image-guided methods, we used a reference rendering per assets with original material assignments as guidance. Finally, we render four views, each in four different lighting conditions (four different HDR probes), resulting in 512 images each for both original and generated materials. The resulting renderings can be compared using image metrics. Note that this comparison goes through a "text bottleneck": the achievable similarity of the corresponding image pairs is limited by this, and the resulting numbers are not directly comparable to image-conditioned models.

We report CLIP-based Fréchet Inception Distance (CLIP-FID) [28], Learned Perceptual Image Patch Similarity (LPIPS) [73], and CLIP Maximum-Mean Discrepancy (CMMD) [23]. We refer to VideoMat [36] for additional comparisons against Paint-it [64], DreamMat [75], and Make-it-Real [14].

Among the text-guided variants, our method has the best scores. We encourage the reader to closely inspect the vi-

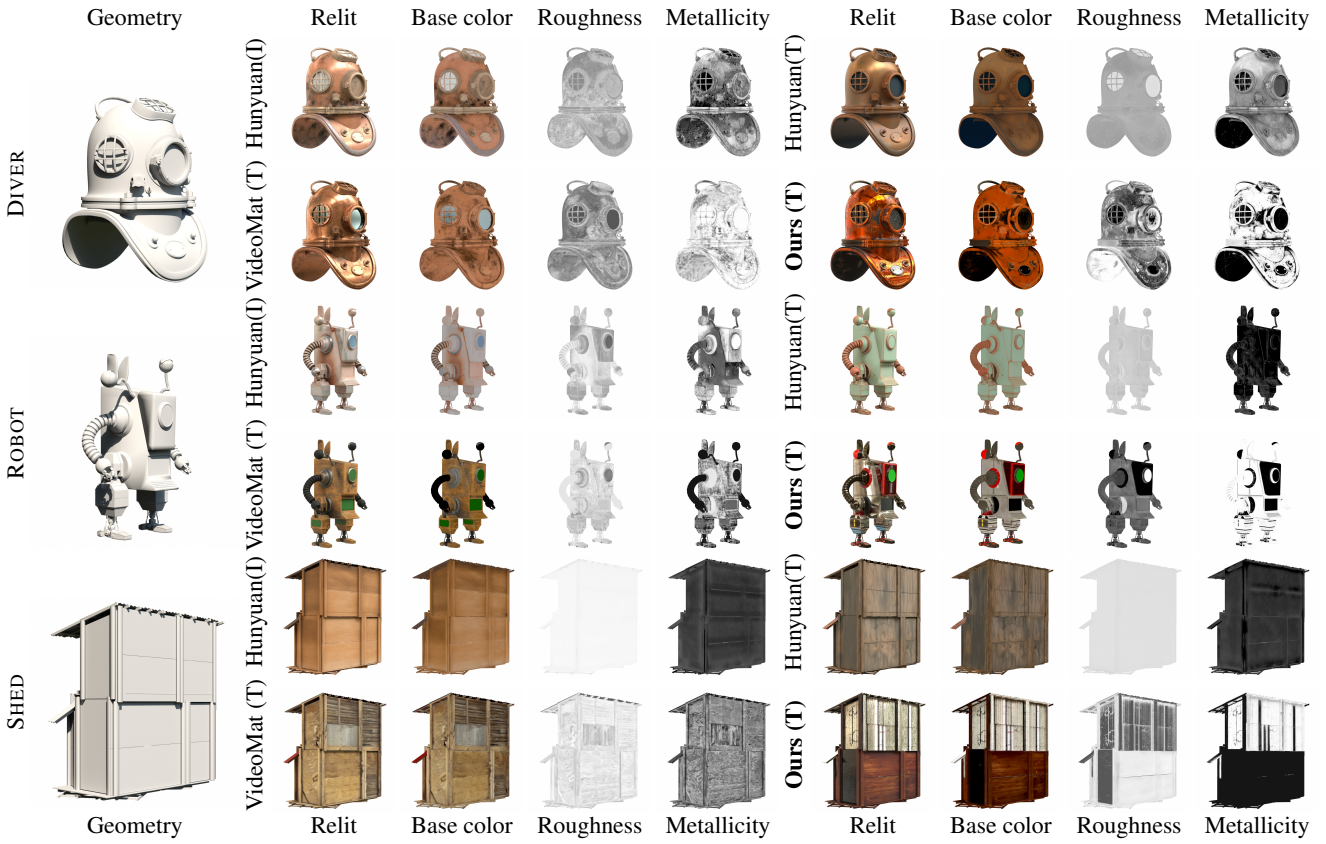

Figure 3. Material generation. We compare against Hunyuan3D Paint 2.1 [18] (image and text guided versions) and VideoMat [36] (text) on three example meshes from the BlenderVault [32] dataset. We encourage the reader to zoom in and compare the quality of the intrinsics (base color, roughness, metallicity), as well as to see the supplementary materials.

sual results, where we argue that VideoMatGen produces sharper results, with more definition, particularly in the roughness and metallicity maps; furthermore, VideoMatGen is the only method producing a height map. For completeness, we also report image-guided results where we have extended our model to accept both a prompt and a single image as guides. While not our primary design goal, we note that our method still performs competitively compared to the state of the art.

4.2. Qualitative evaluation

In Fig. 3, we show visual comparisons against VideoMat [36] and two variants of Hunyuan3D-Paint [18] (image- and text-guided). Overall, the visual results are compelling for all methods, but we notice that the strong prior from the video model helps us generate fine scale detail, and more interesting spatial texture variations, which are coherent across the different material maps thanks to our joint modeling. Unlike the competing methods, we predict a height map, which improves fine scale material detail, as highlighted in Fig. 4. We can create subtle material variations from a single prompt by changing the seed, as shown in Figs. 1 and 5. This can be a helpful artistic tool in creating unique instances for the same base geometry in larger scenes. In Fig. 6 we show our generated materials rendered with three different lighting conditions. Finally, our image conditioned pipeline generates materials which are visually more similar to the test set examples, as shown in Fig. 7.

Table 1. Quantitative metrics for material generation. The mode column indicates if the method is image or text guided.

Method	Mode	CLIP-FID (↓)	CMMD (↓)	LPIPS (↓)
TRELLIS.2 [59]	image	3.913	0.030	0.101
Hunyuan3D 2.1 [18]	image	4.197	0.039	0.102
MVPainter [46]	image	6.583	0.112	0.136
VideoMatGen (ours)	image	4.032	0.028	0.109
Hunyuan3D 2.1 [18]	text	6.694	0.046	0.137
MVPainter [46]	text	7.313	0.096	0.149
VideoMat [36]	text	5.640	0.070	0.130
VideoMatGen (ours)	text	5.575	0.035	0.124

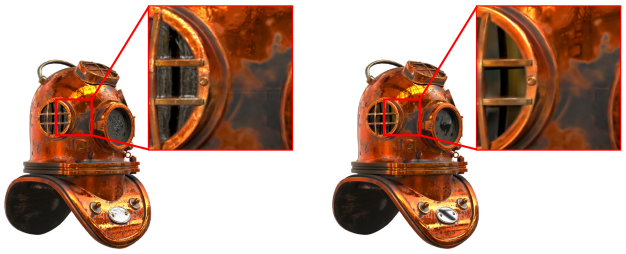

Figure 4. **Left**: Our method predicts a height (bump) map, which improves the visual richness of the generated material. **Right**: corresponding rendering without bump map.

Figure 5. We generate three materials from the same text prompt (see supplemental), each with a unique random seed. This results in subtle variations of materials for the two examples.

Figure 6. We show relit results, using three HDR probes [66], of the generated materials for Hunyuan3D-Paint (image-guided), VideoMat, and our method (both text-guided). Our generated materials produce convincing details in three different lighting scenarios.

Table 2. VAE finetuning evaluation on material maps from our test set and the MatSynth test set. We report PSNR (dB) and LPIPS scores for base color and only PSNR (dB) scores for HRM (height, roughness, metallicity), as perceptual metrics are not applicable. The VAE_{pbr} offers $2\times$ additional compression but has similar visual quality as the Cosmos Tokenizer.

	Base color		HRM
Method	PSNR (↑)	LPIPS (↓)	PSNR (↑)
Cosmos Tokenizer	**38.8**	0.046	**35.1**
VAE_{pbr}	38.3	**0.043**	33.8

4.3. Evaluation of joint prediction

VAE Quality We compare our finetuned VAE with the Cosmos Tokenizer (Cosmos-0.1-Tokenizer-CV8x8x8, applied to single frames). Quality is evaluated using image metrics after encoding and decoding each image. For the Cosmos Tokenizer, base color and HRM (a packed 3-triplet with height, roughness, metallicity) are encoded separately as RGB images, while we jointly encode all six channels using VAE_{pbr}. Our test set consists of 4 views of each of our 32 test assets (128 samples), with their original material assignments. For each view, we render material intrinsics maps for base color, height, roughness and metallicity. Additionally, we use the material textures from the MatSynth [52] test set (89 samples). As shown in Tab. 2, when applying our VAE on material maps, we have similar quality as the Cosmos Tokenizer, while achieving $2\times$ higher compression rate in latent space.

5. Limitations and Future Work

Our method is currently unoptimized and made with no regards to runtime performance. Inference is costly, approximately 2-3 minutes for a single asset on 8×A100 GPUs. We see large potential for optimizing inference with recent video model acceleration and distillation techniques.

Our image (keyframe) VAE approach allows for random camera views at inference time, but we still note that the video model produces best results with a reasonably coherent camera trajectory. Incoherent view-patterns can lead to ghosting or blurring due to misaligned details, and for this reason we chose a object-centric 360° camera orbit during inference. In future work we hope consistency can be improved by better image guides or 3D positional encoding.

While not the primary focus of this paper, our texture baking step is a relatively simple projection of the generated video frames. Recent works have shown that quality can be improved by applying image diffusion models in texture space [44, 78] to in-paint or sharpen details.

We would also like to upgrade our base model to a more recent video diffusion model. In this paper, we deliberately

Figure 7. We compare our text- and image-guided models for six examples, and note that the image-guided version more closely resembles the materials of the dataset entry. We deliberately chose a view with 45° rotation from the conditioning view.

used Cosmos-1.0 for fair comparison with VideoMat, but more recent models can likely improve quality.

6. Conclusion

We present a video diffusion method for joint prediction of material parameters for 3D shapes. We also show the benefits of our new joint material modeling VAE. Our model produces high-quality PBR materials with coherent detail between the material channels and meaningful correlation to geometry parts, and outperforms previous text-to-material approaches. We believe that our text-based material generation can be a useful tool for artists to quickly prototype materials for large sets of 3D objects. Unique material variation for instances of the same geometry can be obtained by simply changing the seed of the noise passed to the diffusion process.

References

[1] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, Varun Jampani, and Robin Rombach. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv:2311.15127*, 2023. 2

[2] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[3] Mark Boss, Zixuan Huang, Aaryaman Vasishta, and Varun Jampani. SF3D: Stable fast 3D mesh reconstruction with uv-unwrapping and illumination disentanglement. *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 2

[4] Brent Burley. Physically Based Shading at Disney. In *SIGGRAPH Courses: Practical Physically Based Shading in Film and Game Production*, 2012. 1, 2

[5] Duygu Ceylan, Valentin Deschaintre, Thibault Groueix, Rosalie Martin, Chun-Hao Huang, Romain Rouffet, Vladimir Kim, and Gaëtan Lassagne. MatAtlas: Text-driven Consistent Geometry Texturing and Material Assignment, 2024. 2

[6] Hila Chefer, Uriel Singer, Amit Zohar, Yuval Kirstain, Adam Polyak, Yaniv Taigman, Lior Wolf, and Shelly Sheynin. VideoJAM: Joint appearance-motion representations for enhanced motion generation in video models. *arXiv: 2502.02492*, 2025. 3, 4

[7] Dave Zhenyu Chen, Yawar Siddiqui, Hsin-Ying Lee, Sergey Tulyakov, and Matthias Nießner. Text2tex: Text-driven texture synthesis via diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 18558–18568, 2023. 2

[8] Zhaoxi Chen, Jiaxiang Tang, Yuhao Dong, Ziang Cao, Fangzhou Hong, Yushi Lan, Tengfei Wang, Haozhe Xie, Tong Wu, Shunsuke Saito, Liang Pan, Dahua Lin, and Ziwei Liu. 3DTopia-XL: High-Quality 3D PBR Asset Generation via Primitive Diffusion. *arXiv preprint arXiv:2409.12957*, 2024. 3

[9] Jasmine Collins, Shubham Goel, Kenan Deng, Achleshwar Luthra, Leon Xu, Erhan Gundogdu, Xi Zhang,

Tomas F Yago Vicente, Thomas Dideriksen, Himanshu Arora, Matthieu Guillaumin, and Jitendra Malik. ABO: Dataset and Benchmarks for Real-World 3D Object Understanding. *CVPR*, 2022. 4

[10] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A Universe of Annotated 3D Objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13142–13153, 2023. 4

[11] Kangle Deng, Timothy Omernick, Alexander Weiss, Deva Ramanan, Jun-Yan Zhu, Tinghui Zhou, and Maneesh Agrawala. FlashTex: fast relightable mesh texturing with LightControlNet. In *European Conference on Computer Vision (ECCV)*, 2024. 2

[12] Prafulla Dhariwal and Alexander Quinn Nichol. Diffusion models beat GANs on image synthesis. In *Advances in Neural Information Processing Systems*, 2021. 2

[13] Andreas Engelhardt, Mark Boss, Vikram Voleti, Chun-Han Yao, Hendrik P. Lensch, and Varun Jampani. Svim3d: Stable video material diffusion for single image 3d generation. *International Conference on Computer Vision*, 2025. 1, 3

[14] Ye Fang, Zeyi Sun, Tong Wu, Jiaqi Wang, Ziwei Liu, Gordon Wetzstein, and Dahua Lin. Make-it-Real: Unleashing Large Multimodal Model for Painting 3D Objects with Realistic Materials, 2024. 2, 5

[15] Yifei Feng, Mingxin Yang, Shuhui Yang, Sheng Zhang, Jiaao Yu, Zibo Zhao, Yuhong Liu, Jie Jiang, and Chunchao Guo. RomanTex: Decoupling 3D-aware Rotary Positional Embedded Multi-Attention Network for Texture Synthesis. *arXiv preprint arXiv:2503.19011*, 2025. 1, 3

[16] Ruiqi Gao*, Aleksander Holynski*, Philipp Henzler, Arthur Brussee, Ricardo Martin-Brualla, Pratul P. Srinivasan, Jonathan T. Barron, and Ben Poole*. CAT3D: Create Anything in 3D with Multi-View Diffusion Models. *Advances in Neural Information Processing Systems*, 2024. 1

[17] Kai He, Ruofan Liang, Jacob Munkberg, Jon Hasselgren, Nandita Vijaykumar, Alexander Keller, Sanja Fidler, Igor Gilitschenski, Zan Gojcic, and Zian Wang. UniRelight: Learning Joint Decomposition and Synthesis for Video Relighting, 2025. 3, 4

[18] Zebin He, Mingxin Yang, Shuhui Yang, Yixuan Tang, Tao Wang, Kaihao Zhang, Guanying Chen, Yuhong Liu, Jie Jiang, Chunchao Guo, and Wenhan Luo. MaterialMVP: Illumination-Invariant Material Generation via Multi-view PBR Diffusion. *arXiv preprint arXiv:2503.10289*, 2025. 1, 3, 5, 6

[19] Paul S. Heckbert. Fundamentals of texture mapping and image warping. Technical report, 1989. 5

[20] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 2

[21] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. CogVideo: Large-scale Pretraining for Text-to-Video Generation via Transformers. In *The Eleventh International Conference on Learning Representations*, 2023. 2

[22] Wenzel Jakob, Sébastien Speierer, Nicolas Roussel, Merlin Nimier-David, Delio Vicini, Tizian Zeltner, Baptiste Nicolet, Miguel Crespo, Vincent Leroy, and Ziyi Zhang. Mitsuba 3 renderer, 2022. https://mitsuba-renderer.org. 2

[23] Sadeep Jayasumana, Srikumar Ramalingam, Andreas Veit, Daniel Glasner, Ayan Chakrabarti, and Sanjiv Kumar. Rethinking fid: Towards a better evaluation metric for image generation, 2024. 5

[24] Haian Jin, Yuan Li, Fujun Luan, Yuanbo Xiangli, Sai Bi, Kai Zhang, Zexiang Xu, Jin Sun, and Noah Snavely. Neural gaffer: Relighting any object via diffusion. In *Advances in Neural Information Processing Systems*, 2024. 3

[25] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D Gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42 (4), 2023. 2

[26] Mukul Khanna*, Yongsen Mao*, Hanxiao Jiang, Sanjay Haresh, Brennan Shacklett, Dhruv Batra, Alexander Clegg, Eric Undersander, Angel X. Chang, and Manolis Savva. Habitat Synthetic Scenes Dataset (HSSD-200): An Analysis of 3D Scene Scale and Realism Tradeoffs for ObjectGoal Navigation. *arXiv preprint*, 2023. 4

[27] Peter Kocsis, Lukas Höllein, and Matthias Nießner. IntrinsiX: High-Quality PBR Generation using Image Priors, 2025. 3

[28] Tuomas Kynkäänniemi, Tero Karras, Miika Aittala, Timo Aila, and Jaakko Lehtinen. The Role of ImageNet Classes in Fréchet Inception Distance. In *Proc. ICLR*, 2023. 5

[29] 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. 2

[30] Jiahao Li, Hao Tan, Kai Zhang, Zexiang Xu, Fujun Luan, Yinghao Xu, Yicong Hong, Kalyan Sunkavalli, Greg Shakhnarovich, and Sai Bi. Instant3d: Fast text-to-3d with sparse-view generation and large reconstruction model, 2023. 2

[31] Ruofan Liang, Zan Gojcic, Huan Ling, Jacob Munkberg, Jon Hasselgren, Zhi-Hao Lin, Jun Gao, Alexander Keller, Nandita Vijaykumar, Sanja Fidler, and Zian Wang. DiffusionRenderer: Neural Inverse and Forward Rendering with Video Diffusion Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. 1, 3, 4

[32] Yehonathan Litman, Or Patashnik, Kangle Deng, Aviral Agrawal, Rushikesh Zawar, Fernando De la Torre, and Shubham Tulsiani. MaterialFusion: Enhancing Inverse Rendering with Material Diffusion Priors. In *3DV*, 2025. 3, 4, 6

[33] Yuanxun Lu, Jingyang Zhang, Tian Fang, Jean-Daniel Nahmias, Yanghai Tsin, Long Quan, Xun Cao, Yao Yao, and Shiwei Li. Matrix3D: Large Photogrammetry Model All-in-One, 2025. 3, 4

[34] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *ECCV*, 2020. 2

[35] Jacob Munkberg, Jon Hasselgren, Tianchang Shen, Jun Gao, Wenzheng Chen, Alex Evans, Thomas Müller, and Sanja Fidler. Extracting Triangular 3D Models, Materials, and Light-

ing From Images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8280–8290, 2022. 2

[36] Jacob Munkberg, Zian Wang, Ruofan Liang, Tianchang Shen, and Jon Hasselgren. VideoMat: Extracting PBR Materials from Video Diffusion Models. In *Eurographics Symposium on Rendering - CGF Track*, 2025. 1, 3, 5, 6

[37] NVIDIA. Cosmos World Foundation Model Platform for Physical AI. *arXiv preprint arXiv:2501.03575*, 2025. 2, 3, 4

[38] William Peebles and Saining Xie. Scalable diffusion models with transformers. *arXiv preprint arXiv:2212.09748*, 2022. 2, 3

[39] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. DreamFusion: Text-to-3D using 2D Diffusion. *arXiv*, 2022. 1, 2

[40] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2023. 4

[41] Xuanchi Ren, Tianchang Shen, Jiahui Huang, Huan Ling, Yifan Lu, Merlin Nimier-David, Thomas Müller, Alexander Keller, Sanja Fidler, and Jun Gao. GEN3C: 3D-Informed World-Consistent Video Generation with Precise Camera Control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. 5

[42] Elad Richardson, Gal Metzer, Yuval Alaluf, Raja Giryes, and Daniel Cohen-Or. TEXTure: Text-guided texturing of 3d shapes. In *ACM SIGGRAPH 2023 conference proceedings*, pages 1–11, 2023. 2

[43] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. DreamBooth: Fine Tuning Text-to-image Diffusion Models for Subject-Driven Generation. *arXiv preprint arxiv:2208.12242*, 2022. 2

[44] ByteDance Seed. *Seed3D 1.0: From Images to High-Fidelity Simulation-Ready 3D Assets*, 2025. 3, 5, 7

[45] InstantX Team Shakker Labs. *FLUX.1-dev-ControlNet-Depth*, 2024. 5

[46] Mingqi Shao, Feng Xiong, Zhaoxu Sun, and Mu Xu. MV-Painter: Accurate and Detailed 3D Texture Generation via Multi-View Diffusion with Geometric Control. *arXiv preprint arXiv:2505.12635*, 2025. 1, 3, 5, 6

[47] Yichun Shi, Peng Wang, Jianglong Ye, Long Mai, Kejie Li, and Xiao Yang. MVDream: Multi-view Diffusion for 3D Generation. *arXiv:2308.16512*, 2023. 2

[48] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, 2015. 2

[49] Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024. 5

[50] Alexandru Telea. An image inpainting technique based on the fast marching method. *Journal of Graphics Tools*, 9(1): 23–34, 2004. 5

[51] Karthik Vaidyanathan, Marco Salvi, Bartlomiej Wronski, Tomas Akenine-Moller, Pontus Ebelin, and Aaron Lefohn. Random-access neural compression of material textures. *ACM Trans. Graph.*, 42(4), 2023. 4

[52] Giuseppe Vecchio and Valentin Deschaintre. Matsynth: A modern pbr materials dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22109–22118, 2024. 5, 7

[53] Vikram Voleti, Chun-Han Yao, Mark Boss, Adam Letts, David Pankratz, Dmitrii Tochilkin, Christian Laforte, Robin Rombach, and Varun Jampani. SV3D: Novel multi-view synthesis and 3D generation from a single image using latent video diffusion. In *European Conference on Computer Vision (ECCV)*, 2024. 2

[54] Bruce Walter, Stephen R. Marschner, Hongsong Li, and Kenneth E. Torrance. Microfacet Models for Refraction through Rough Surfaces. In *Proceedings of the 18th Eurographics Conference on Rendering Techniques*, page 195–206, 2007. 1

[55] 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. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 2

[56] Chen Xi, Peng Sida, Yang Dongchen, Liu Yuan, Pan Bowen, Lv Chengfei, and Zhou. Xiaowei. IntrinsicAnything: learning diffusion priors for inverse rendering under unknown illumination. *arxiv: 2404.11593*, 2024. 3

[57] Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3D Latents for Scalable and Versatile 3D Generation. *arXiv preprint arXiv:2412.01506*, 2024. 2

[58] Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation. *arXiv preprint arXiv:2412.01506*, 2024. 1

[59] Jianfeng Xiang, Xiaoxue Chen, Sicheng Xu, Ruicheng Wang, Zelong Lv, Yu Deng, Hongyuan Zhu, Yue Dong, Hao Zhao, Nicholas Jing Yuan, and Jiaolong Yang. Native and Compact Structured Latents for 3D Generation. *Tech report*, 2025. 5, 6

[60] Haibo Yang, Yang Chen, Yingwei Pan, Ting Yao, Zhineng Chen, Chong-Wah Ngo, and Tao Mei. Hi3D: Pursuing High-Resolution Image-to-3D Generation with Video Diffusion Models. In *ACM MM*, 2024. 2

[61] Jiayu Yang, Taizhang Shang, Weixuan Sun, Xibin Song, Ziang Chen, Senbo Wang, Shenzhou Chen, Weizhe Liu, Hongdong Li, and Pan Ji. Pandora3D: A Comprehensive Framework for High-Quality 3D Shape and Texture Generation. *arXiv preprint arXiv:2502.14247*, 2025. 1, 3

[62] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, Da Yin, Xiaotao Gu, Yuxuan Zhang, Weihan Wang, Yean Cheng, Ting Liu, Bin Xu, Yuxiao Dong, and Jie Tang. CogVideoX: Text-to-Video Diffusion Models with An Expert Transformer, 2024. 2

[63] Yu-Ying Yeh, Jia-Bin Huang, Changil Kim, Lei Xiao, Thu Nguyen-Phuoc, Numair Khan, Cheng Zhang, Manmohan Chandraker, Carl S Marshall, Zhao Dong, et al. TextureDreamer: Image-guided Texture Synthesis through Geometry-aware Diffusion. *arXiv preprint arXiv:2401.09416*, 2024. 2

[64] Kim Youwang, Tae-Hyun Oh, and Gerard Pons-Moll. Paint-it: Text-to-Texture Synthesis via Deep Convolutional Texture Map Optimization and Physically-Based Rendering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2, 5

[65] Xin Yu, Ze Yuan, Yuan-Chen Guo, Ying-Tian Liu, Jianhui Liu, Yangguang Li, Yan-Pei Cao, Ding Liang, and Xiaojuan Qi. TEXGen: a Generative Diffusion Model for Mesh Textures. *ACM Trans. Graph.*, 43(6), 2024. 2

[66] Greg Zaal and et al. *Poly Haven - The Public 3D Asset Library*, 2024. 5, 7

[67] Chong Zeng, Yue Dong, Pieter Peers, Youkang Kong, Hongzhi Wu, and Xin Tong. DiLightNet: fine-grained lighting control for diffusion-based image generation. In *ACM SIGGRAPH 2024 Conference Papers*, 2024. 3

[68] Xianfang Zeng, Xin Chen, Zhongqi Qi, Wen Liu, Zibo Zhao, Zhibin Wang, Bin Fu, Yong Liu, and Gang Yu. Paint3d: Paint anything 3d with lighting-less texture diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4252–4262, 2024. 2

[69] Zheng Zeng, Valentin Deschaintre, Iliyan Georgiev, Yannick Hold-Geoffroy, Yiwei Hu, Fujun Luan, Ling-Qi Yan, and Miloš Hašan. RGB↔X: image decomposition and synthesis using material-and lighting-aware diffusion models. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 3, 4

[70] Cheng Zhang, Bailey Miller, Kai Yan, Ioannis Gkioulekas, and Shuang Zhao. Path-space differentiable rendering. *ACM Trans. Graph.*, 39(4):143:1–143:19, 2020. 2

[71] Kai Zhang, Sai Bi, Hao Tan, Yuanbo Xiangli, Nanxuan Zhao, Kalyan Sunkavalli, and Zexiang Xu. Gs-lrm: Large reconstruction model for 3d gaussian splatting. *European Conference on Computer Vision*, 2024. 1, 2

[72] Longwen Zhang, Ziyu Wang, Qixuan Zhang, Qiwei Qiu, Anqi Pang, Haoran Jiang, Wei Yang, Lan Xu, and Jingyi Yu. CLAY: A Controllable Large-scale Generative Model for Creating High-quality 3D Assets. *ACM Transactions on Graphics (TOG)*, 43(4):1–20, 2024. 2

[73] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 5

[74] Shangzhan Zhang, Sida Peng, Tao Xu, Yuanbo Yang, Tianrun Chen, Nan Xue, Yujun Shen, Hujun Bao, Ruizhen Hu, and Xiaowei Zhou. MaPa: Text-driven Photorealistic Material Painting for 3D Shapes. In *ACM SIGGRAPH 2024 Conference Papers*, 2024. 2

[75] Yuqing Zhang, Yuan Liu, Zhiyu Xie, Lei Yang, Zhongyuan Liu, Mengzhou Yang, Runze Zhang, Qilong Kou, Cheng Lin, Wenping Wang, and Xiaogang Jin. DreamMat: High-quality PBR Material Generation with Geometry- and Light-aware Diffusion Models. *ACM Trans. Graph.*, 43(4), 2024. 2, 5

[76] Xiaoming Zhao, Pratul P. Srinivasan, Dor Verbin, Keunhong Park, Ricardo Martin Brualla, and Philipp Henzler. IllumiNeRF: 3D Relighting Without Inverse Rendering. In *NeurIPS*, 2024. 3

[77] Junzhe Zhu and Peiye Zhuang. HiFA: High-fidelity Text-to-3D Generation with Advanced Diffusion Guidance, 2023. 2

[78] Shenhao Zhu, Lingteng Qiu, Xiaodong Gu, Zhengyi Zhao, Chao Xu, Yuxiao He, Zhe Li, Xiaoguang Han, Yao Yao, Xun Cao, Siyu Zhu, Weihao Yuan, Zilong Dong, and Hao Zhu. Mcmat: Multiview-consistent and physically accurate pbr material generation, 2024. 3, 7