IntrinsiX:
High-Quality PBR Generation using Image Priors

Peter Kocsis **Lukas Höllein** **Matthias Nießner**
Technical University of Munich

peter-kocsis.github.io/IntrinsiX/

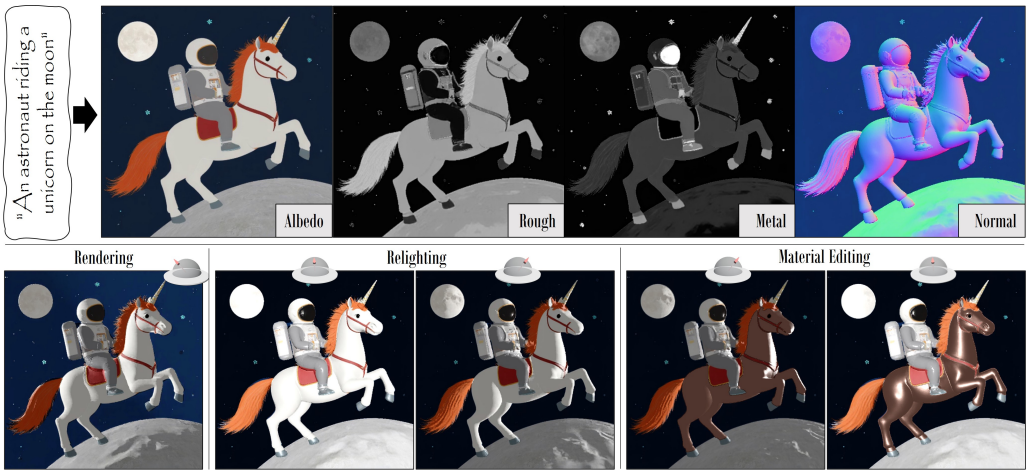

Figure 1: **IntrinsiX.** We present a text-guided intrinsic image generator. Given a text prompt, our method produces high-quality albedo, roughness, metallic, and normal maps which can be rerendered under any lighting conditions. Our model enables downstream applications, such as relightable object or scene generation, and material or lighting editing.

Abstract

We introduce IntrinsiX, a novel method that generates high-quality intrinsic images from text description. In contrast to existing text-to-image models whose outputs contain baked-in scene lighting, our approach predicts physically-based rendering (PBR) maps. This enables the generated outputs to be used for content creation scenarios in core graphics applications that facilitate re-lighting, editing, and texture generation tasks. In order to train our generator, we exploit strong image priors, and pre-train separate models for each PBR material component (albedo, roughness, metallic, normals). We then align these models with a new cross-intrinsic attention formulation that concatenates key and value features in a consistent fashion. This allows us to exchange information between each output modality and to obtain semantically coherent PBR predictions. To ground each intrinsic component, we propose a rendering loss which provides image-space signals to constrain the model, thus facilitating sharp details also in the output BRDF properties. Our results demonstrate detailed intrinsic generation with strong generalization capabilities that outperforms existing intrinsic image decomposition methods used with generated images by a significant margin. Finally, we show a series of applications, including re-lighting, editing, and for the first time text-conditioned room-scale PBR texture generation. We will release the code and the pre-trained model weights.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

1 Introduction

Text-to-image (T2I) models have revolutionized 2D content creation, by generating high-quality RGB images from just a text description [50, 54, 48]. They are used in widespread applications, including extensions for controllable generation beyond text [74, 71, 44], personalization and stylization of generated images [52, 25], and 3D asset or scene generation [46, 7, 22]. However, in all cases the content is typically generated in shaded RGB space, that contains baked-in lighting effects (e.g., reflections, shadows, specular highlights). This limits the usability of T2I models for many content creation scenarios such as gaming or VR applications, that requires PBR maps (albedo, roughness, metallic, normal) to render or relight scenes realistically.

Existing methods perform intrinsic image decomposition on RGB images [76, 31, 72, 5]. However, finding the correct decomposition to a given input image is a constrained task, usually causing over-smoothed or simplified predictions on out-of-domain samples. These methods are trained with synthetic conditioning input [76, 35], leading to low-quality decompositions for out-of-distribution inputs, limiting their effectiveness on diverse real-world images. Similarly, methods that generate 3D PBR content from T2I models [58, 61, 47, 26] are trained on object-scale datasets [10, 9], making them unsuitable for large-scale 3D scenes.

We take a different approach for PBR map generation. For the first time, we *directly* generate PBR maps from text as input in a probabilistic diffusion process. Since our method does not rely on an input image, it is more self-contained, enabling better generalization capabilities. We can use the generated PBR maps for downstream tasks, such as physically-based rendering, relighting, or material editing (Figure 1). We also showcase that our method can generate PBR textures for entire 3D scenes, for the first time to the best of our knowledge, making it directly usable for gaming/VR applications (Figure 5). Our method leverages the strong image prior of pretrained T2I models and converts it into a PBR map generator. This way, our model can generate PBR content from diverse, out-of-distribution text prompts, similar to existing T2I models that operate in RGB space. Concretely, we first train intrinsic priors for each material property and for normal map generation separately (Section 3.1). We leverage small, curated datasets and the established LoRA [25] extension for T2I models. Then, we fine-tune all priors jointly by employing cross-intrinsic attention in the diffusion transformer network (Section 3.2). This allows intrinsic properties to interact, enabling their joint and coherent generation. We also introduce a rendering objective with importance-based lighting sampling to ground the intrinsic components. This image-space signal encourages sharp and semantically meaningful decompositions. In summary, our contributions:

- We introduce the first method, that *directly* generates PBR images from text as input in a probabilistic diffusion process. In comparison to baselines, our PBR maps are of higher quality and can be used for various downstream tasks, including physically-based rendering, editing/relighting, and room-scale 3D scene PBR texturing.
- We decompose the strong image prior of pretrained T2I models into intrinsic components in a two-stage training process. This allows us to generate PBR maps from diverse text prompts, that are not limited to the distribution of existing, synthetic datasets.
- We combine cross-intrinsic attention with a novel rendering objective using importance-based light sampling to jointly generate semantically coherent PBR maps.

2 Related Work

Text-to-Image Models Text-to-image (T2I) models have emerged as powerful tools for 2D content creation; they create high-quality, diverse images from only text as input [50, 54, 48]. Since their inception, several models further increased the visual quality of generated images [45, 32, 67, 73]. These models are trained on datasets consisting of billions of images, like [55]. This makes them a strong 2D prior for arbitrary content generation. They typically model the diffusion process following Ho et al. [21] or Lipman et al. [37] with U-Net [51] or diffusion transformer (DiT) [41, 62] architectures. Many downstream applications leverage T2I models, including controllable content generation [74, 71, 44, 30, 56] as well as personalization and stylization of generated images [52, 25, 63, 59]. We leverage pretrained T2I models as prior for our task, the generation of PBR maps from text.

Task-specific Finetuning of Text-to-Image Models In order to use T2I models for downstream tasks, different modifications to the model architecture exist and can be applied [74, 44, 71, 38, 30]. In particular, LoRA layers [25] can be used to teach T2I models about specific "styles" (e.g., artistic paintings). Additional low-rank linear layers are trained in every attention block, which keeps the generalized prior of the T2I model, while finetuning on smaller-scale datasets.

We similarly finetune multiple LoRAs to teach a T2I model about the distribution of intrinsic images.

Other tasks generate multi-view image outputs, such as video generation [65, 70] multi-view image generation [23, 39, 60] or multi-modal generation [68]. They augment the attention operation in the transformer architecture to jointly process multiple images in a batch with the same model. Related to these tasks, we perform cross-intrinsic attention to generate aligned PBR maps in a single denoising forward pass with our finetuned model.

T2I models are also applied to 3D tasks, like object generation [6] or scene generation [22, 7]. Some methods finetune T2I models on synthetic 3D objects datasets, like [10, 9], to generate object-scale 3D assets [3, 58, 13]. In contrast, we utilize score distillation [46, 18] to generate PBR textures of entire 3D scenes following Chen et al. [7].

Material Reconstruction Completely decoupling lighting from material properties requires multiple surface observations under different lighting conditions. Pret-trained models can enable material acquisition from sparse observations. [11] uses a feed-forward model to predict the texture of a single material and similarly use lighting sampling with the reflected view direction. Orthogonal to this, we aim to generate the PBR properties of complex scenes in image space to enable downstream applications. We introduce a rendering loss to the diffusion framework using a roughness-weighted importance sampling for the lighting direction.

The field of intrinsic image decomposition focuses on obtaining PBR maps from a single RGB image. Early approaches focus on separating the reflectance from shading [33, 24, 64] using various heuristics, such as sparsity in reflectance properties [14, 57, 17, 75], or smoothness [2]. Later, deep-learning methods [34, 12, 31, 72, 61, 36] train decomposition networks on synthetic datasets, such as [76]. However, the decomposition of an RGB image into its intrinsic properties is a constrained task, making it hard to generalize to out-of-distribution input images. In contrast, we directly generate all PBR components from text as input. This drastically improves the performance on in-the-wild settings.

Material Generation Recent works use diffusion models for single material generation, conditioned on text or image inputs [40, 42]. The work of [69] concurrently addresses text-conditional complex PBR generation. They train a shared ControlNet [74] and use a diffusion renderer for editability. In contrast, our method first learns a prior over the PBR properties independently, then aligns them with cross-intrinsic attention using a fixed renderer to ensure compatibility with standard rendering engines.

3 Method

Our method generates the intrinsic properties of an image given a text prompt as input (Figure 1 top). Specifically, we leverage the strong prior of a pretrained text-to-image model and turn it into a PBR map generator. First, we learn the distribution of intrinsic properties (albedo, roughness, metallic, normal) by finetuning LoRA layers on each modality separately (Section 3.1). Then, we learn the joint distribution by leveraging cross-intrinsic attention and by minimizing a novel rendering objective (Section 3.2). Our method generates multiple images corresponding to the different PBR maps, allowing for various downstream applications (Figure 4, Figure 5). We summarize our method in Figure 2.

3.1 PBR Prior Training

In order to generate PBR maps of an image, we model the distribution of the individual intrinsic image properties. Specifically, we model the probability distribution $p_\theta(\mathbf{X}_0)$ over data $\mathbf{X}_0 \sim q(\mathbf{X}_0)$, where $\mathbf{X}_0 = \{\mathbf{x}_a \in \mathbb{R}^{3\times P}, \mathbf{x}_r \in \mathbb{R}^P, \mathbf{x}_m \in \mathbb{R}^P, \mathbf{x}_n \in \mathbb{R}^{3\times P}\}$, $P := H \times W$ is shorthand for the image size, and the suffixes a, r, m, n refer to the albedo, roughness, metallic, and normal intrinsic properties, respectively. In other words, we learn the *joint* probability distribution of all intrinsic properties through the parameters θ of a neural network.

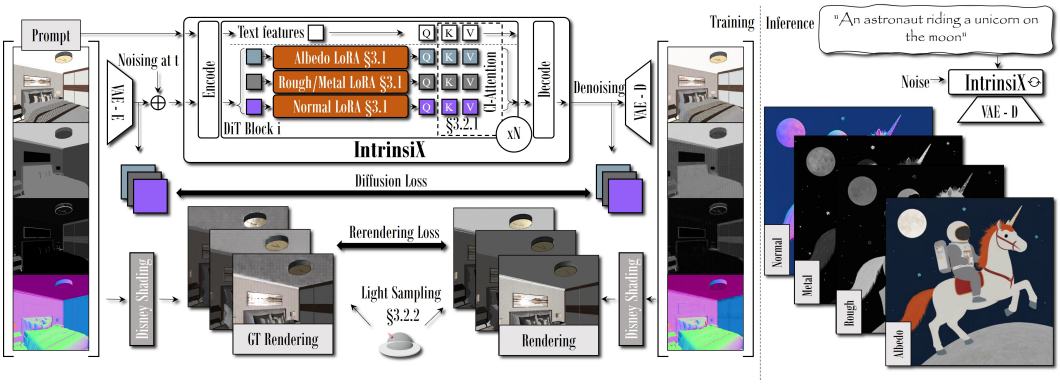

Figure 2: **Method Overview**. We generate the intrinsic properties of an image given text as input. **Left:** we train 3 different LoRAs for a pretrained, latent text-to-image model, corresponding to the intrinsic properties (albedo, normal, and roughness + metallic) on curated synthetic datasets (Section 3.1). We facilitate communication between all 4 modalities through cross-intrinsic attention to predict PBR maps corresponding to the same image (Section 3.2.1). A novel rendering loss using importance-based light sampling ensures that we can render high-quality RGB images from physically realistic PBR maps (Section 3.2.2). **Right:** after training, we jointly denoise and decode all 4 PBR maps and can prompt our model with diverse, out-of-distribution descriptions.

Unfortunately, existing datasets, such as Openrooms [35], InteriorVerse [76] or Hypersim [49], contain either only synthetic examples of intrinsic decompositions or are limited in size. Thus, models trained on such datasets exhibit limited generalization to arbitrary, real-world examples. On the other side, recent text-to-image diffusion models [50, 45, 67] are able to generate high-quality and diverse image samples. These models learn the probability distribution $p_\phi(\mathbf{x}_0|\mathbf{c}) = \int p_\phi(\mathbf{x}_{0:T}|\mathbf{c}) d\mathbf{x}_{1:T}$ where $\mathbf{c}$ is a text condition, $\mathbf{x}_0 \in \mathbb{R}^{3\times P} \sim q_{rgb}(\mathbf{x}_0)$ is sampled from billions of RGB images [55], and the latent variables $\mathbf{x}_{1:T} = \mathbf{x}_1, \ldots, \mathbf{x}_T$ gradually add more Gaussian noise to the data, following [21]. We leverage this strong image prior by turning pretrained diffusion models into PBR map generators.

In the first stage, we model the intrinsic image properties separately. That is, we learn $p_{\phi,\theta_a}(\mathbf{x}_a)$ and $p_{\phi,\theta_n}(\mathbf{x}_n)$ corresponding to the albedo and normal maps, respectively. Since roughness and metallic are both 1-channel properties, we concatenate them together with an additional 0-channel and learn $p_{\phi,\theta_{r,m}}(\mathbf{x}_r, \mathbf{x}_m)$. This concatenation makes our samples compatible with the VAE, similarly as in [31]. Here, ϕ are the pretrained weights of the Flux.1-dev [1] model [32] and θ are the parameters of LoRA layers [25] injected into the MLP layers before and after the attention (to_q, to_k, to_v, to_out) module of all DiT blocks of the diffusion transformer model architecture [41]. This is an established way to teach large text-to-image models about new concepts (e.g., our PBR map distribution), while retaining the ability to generate diverse samples [19]. To this end, we curate a paired dataset of prompts and intrinsic properties and train the LoRA layers, while keeping the rest of the pretrained model frozen. Precisely, we minimize the conditional flow matching loss [37]:

$$\mathcal{L}_{\text{CFM}}(\theta_a) = \mathbb{E}_{t\sim\mathcal{U}(0,1),\epsilon\sim\mathcal{N}(\mathbf{0},\mathbf{I})}\left[||\hat{\mathbf{u}}_t(\mathbf{z}_t;t) - \mathbf{u}_t(\mathbf{x}_a;\epsilon)||_2^2\right] \tag{1}$$

where $\mathbf{x}_a \sim q(\mathbf{x}_a)$, $\mathbf{z}_t = (1-t)\mathbf{x}_a + t\epsilon$ the noisy data at timestep t, $\mathbf{u}_t = \epsilon - \mathbf{x}_a$ the ground-truth vector field, and $\hat{\mathbf{u}}_t = \hat{\epsilon} - \hat{\mathbf{x}}_a$ its network prediction.

Dataset for albedo and normals Thanks to utilizing a pre-trained image prior, our method does not require extensive PBR datasets, which are generally not available. We collect as little as 20 synthetic examples of albedo and normal maps from the InteriorVerse dataset [76]. Then, we generate captions for each image with the Florence-2 model [66] using the respective rgb renderings. We train the LoRAs θ_a and θ_n on these text-image pairs and obtain high-quality results for diverse, out-of-distribution prompts. This follows previous works, in which text-to-image models learn a new "style" of generated images given only a few example images [63, 59, 53]. We refer to the supplementary material for more details.

[1] `https://huggingface.co/black-forest-labs/FLUX.1-dev`

Dataset for roughness and metallic Similarly, we collect and caption samples for roughness and metallic properties. However, we observe that training on a small dataset does not teach the model intricate details about the distribution of these PBR maps. We hypothesize that this is because the data distribution of roughness/metallic is drastically different from RGB images and therefore requires more observations to learn. To this end, we curate a large dataset of 20K roughness/metallic samples using the InteriorVerse dataset [76]. The resulting LoRA $\theta_{r,m}$ exhibits worse generalization capabilities than θ_a and θ_n, i.e., it overfits to the indoor scene setup. However, in Section 3.2, we show how we can still turn $\theta_{r,m}$ into a generalized PBR generator by combining it with θ_a and θ_n.

3.2 PBR Prior Alignment

After training the LoRAs separately in the first stage, we finetune all LoRA parameters together to learn the *joint* distribution $p_{\phi,\theta_a,\theta_r,\theta_m,\theta_n}(\mathbf{X}_0)$. At inference time, this allows us to sample aligned PBR maps across all modalities. First, we replace self-attention with cross-intrinsic attention in every DiT block to facilitate communication between the different PBR maps. Second, we propose a novel rendering objective that uses all generated PBR maps to create an RGB output image. In the following, we detail both components.

3.2.1 Cross-Intrinsic Attention

Inspired by multi-view diffusion methods [39, 23, 60, 16], we leverage cross-attention in the DiT blocks to facilitate communication between batch elements. We employ a batch-size of 3 and use one of the intrinsic LoRAs from the first stage training for each of the images, while sharing weights for all the other parts of the model. We denote $\mathbf{q}_a^i, \mathbf{k}_a^i, \mathbf{v}_a^i$ as the query, key, and value features of the i-th DiT block for the batch element corresponding to the albedo image and similarly for the other intrinsic properties. Then, we calculate cross-intrinsic attention as:

$$\mathbf{h}_a^i = \text{softmax}\left(\frac{\mathbf{q}_a^i[\mathbf{k}_{r,m}^i, \mathbf{k}_n^i, \mathbf{k}_a^i]^T}{\sqrt{d}}\right)[\mathbf{v}_{r,m}^i, \mathbf{v}_n^i, \mathbf{v}_a^i] \tag{2}$$

where $[\cdot,\cdot]$ denotes concatenation along the sequence dimension and we omit the text feature for clarity. We similarly calculate $\mathbf{h}_{r,m}^i$ and $\mathbf{h}_n^i$. Finetuning all LoRA layers *jointly* with cross-intrinsic attention allows us to generate aligned PBR maps of the same image.

Additionally, we employ dropout regularization to preserve the learned prior of the intrinsic LoRAs. That is, with probability $p_i = 0.25$ we calculate regular self-attention instead of cross-intrinsic attention in the i-th DiT block during training. We show in Figure 8, that this yields PBR maps of higher quality with sharper details.

3.2.2 RGB Rendering Loss

Cross-intrinsic attention allows us to generate aligned PBR maps of the same image. However, individual intrinsics can still be of low quality (see Figure 8). This is because all LoRAs are finetuned jointly, which encourages similar feature distributions during attention, i.e., the differences between the PBR maps are "averaged out". To this end, we incorporate a novel rendering loss in the finetuning stage. Its goal is to provide semantic guidance to the intrinsic properties, that is, it teaches how the PBR maps are combined, encouraging their distinct feature distributions.

Concretely, we render an RGB image from the predicted PBR maps. First, we obtain the clean data samples as $\hat{\mathbf{z}}_0 = \mathbf{z}_t - t\hat{\mathbf{u}}(\mathbf{z}_t; t)$, where $\mathbf{z}$ denotes the batched data of all PBR maps. Then, we decode them from the latent space with the

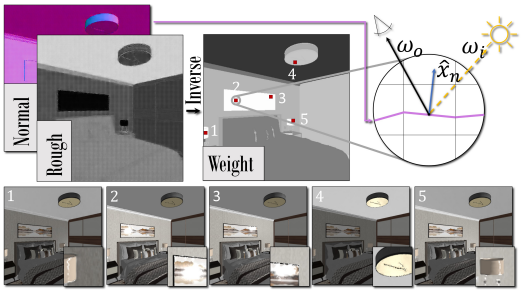

Figure 3: **Importance-based Light Sampling**. We render RGB images (bottom) from our generated PBR maps and a sampled light source as input (top). We employ multinomial importance sampling based using the inverse roughness to select *less* rough pixels more often (red squares). The light direction is then the viewing direction to the pixel reflected by its normal. The rendered images thus contain more specular effects, which provides better gradients during training.

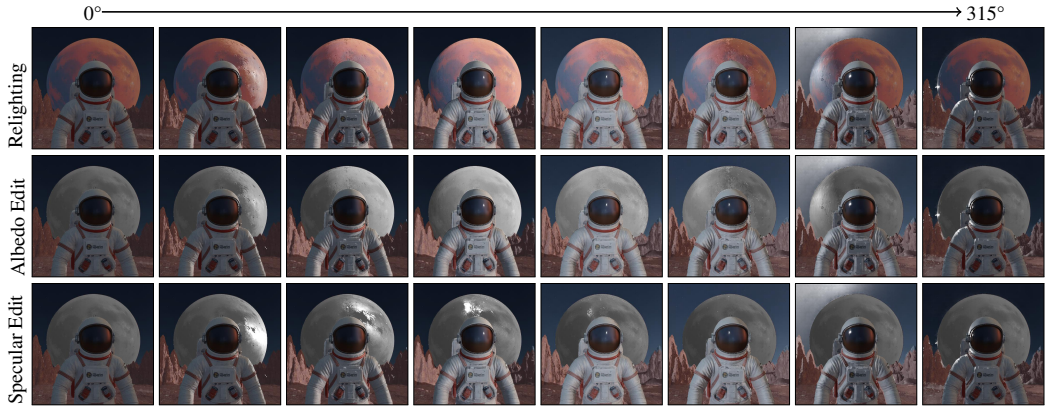

Figure 4: **Editable Image Generation**. Our generated PBR maps can be edited and utilized in standard physically-based rendering frameworks to produce RGB renderings. Here, we place a light source on top of the scene at constant elevation and rotate it around the vertical axis. From top to bottom we show, **(1):** RGB renderings with different light source positions; **(2):** manual edit of the albedo (desaturate the moon color); **(4):** lower roughness and higher metallic value (more glossy, mirror-like reflections).

VAE to obtain $\hat{\mathbf{X}}_0$. We use the simplified Disney BRDF model [4] and interpret $\hat{\mathbf{X}}_0$ as the screen-space buffers of albedo, roughness, metallic, and normal properties. Assuming a single directional light source, we can use deferred shading to obtain an RGB image as:

$$\hat{\mathbf{I}} = f(\omega_o; \omega_i; \hat{\mathbf{X}}_0) \cdot L_i \cdot (\hat{x}_n^T \omega_i) \tag{3}$$

where f is the BRDF evaluation value, ω_o the viewing direction, and $\{L_i, \omega_i\}$ the intensity and direction of a single light source. We determine the viewing direction ω_0 using the camera intrinsics of the dataset (we find this still yields good results during inference). Similarly, we obtain the ground-truth RGB image $\mathbf{I}$ by using the same light, but the PBR maps of the dataset. Then, we calculate the rendering loss:

$$\mathcal{L}_{\text{rgb}}(\hat{\mathbf{I}}, \mathbf{I}) = ||\hat{\mathbf{I}} - \mathbf{I}||_2^2 + 0.1 \cdot \text{LPIPS}(\hat{\mathbf{I}}, \mathbf{I}) \tag{4}$$

where LPIPS denotes the perceptual loss [28].

We require light samples $\{L_i, \omega_i\}$ to render RGB images, following Equation (3). In practice, we sample a single directional light source per image and always use constant intensity $L_i{=}e^2$. We employ importance sampling to obtain the direction of the light ω_i (see Figure 3). That is, we invert the generated roughness $\hat{\mathbf{x}}_\mathbf{r}{\in}[0, 1]$ and use it as the weights for multinomial sampling of a pixel in the image. Thus, pixels with *lower* roughness are selected more often. Then, we obtain the light direction as the reflected view direction $\omega_i{=}2\hat{\mathbf{x}}_n\langle\, \hat{\mathbf{x}}_n, \omega_o\rangle{-}\omega_o$, where ω_o is the viewing direction and $\hat{\mathbf{x}}_n$ the normal vector corresponding to the sampled pixel. This way, we produce RGB images that contain specular highlights and therefore we obtain better gradients for the roughness and metallic LoRAs. This helps to increase the quality of those PBR maps (Figure 8).

During the second finetuning stage, we sample 5 directional light sources in every iteration and render a separate RGB image with each of them. The final loss then becomes $\mathcal{L}{=}\mathcal{L}_{\text{CFM}}{+}\sum_{i=1}^{5}\mathcal{L}_{\text{rgb}}(\hat{\mathbf{I}}_i, \mathbf{I}_i)$. We do not backpropagate $\mathcal{L}_{\text{rgb}}$ to the parameters θ_n of the normal LoRA, as we find it stabilizes the rendering quality by avoiding ambiguities between material and geometry.

4 Applications

Since we directly generate PBR maps, we can utilize standard computer graphics pipelines for physically-based rendering to produce RGB renderings. This allows for various downstream tasks.

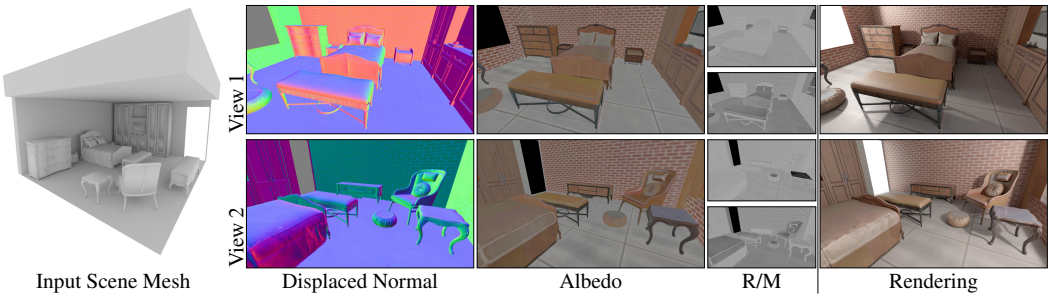

Figure 5: **Scene Texturing**. We can use our method for scene texturing using score distillation [7]. Given a scene geometry, first, we condition our method on the rendered normal maps to produce the remaining PBR maps. Through iterative optimization, we obtain realistic PBR textures for the whole scene. Then, we similarly optimize for normal map textures to obtain fine geometric details, conditioned on rendered material maps. This showcases the potential of *direct* PBR map generation to democratize scene texturing from only text as input.

Editable Image Generation We select a directional light source during rendering of an RGB image from our PBR maps (see Equation (3)). Since our model produces PBR maps, we can vary the direction of the light source arbitrarily and render them under numerous lighting conditions. Similarly, we can manually edit the individual PBR maps, e.g., by changing the albedo color of individual objects or by making them more specular. We show two examples in Figure 4 and Figure 1. Note that our PBR maps are not restricted to a single lighting direction. This can enable artists to precisely tune the appearance of our generated images to their individual needs and therefore make the generations more useful for practical applications.

PBR Scene Texturing We can use our method to perform 3D scene PBR texturing (Figure 5). Recently, pretrained text-to-image models have been used as prior to distill information in 3D [46, 58, 7]. We apply the SceneTex approach [7], but use our fine-tuned PBR model instead of an RGB model. This enables us to distill *uv*-textures for a given geometry corresponding to the individual intrinsic components. We can then render, relight, and edit an entire 3D scene according to physically-based rendering frameworks (see Figure 5). This shows the potential of *direct* PBR map generation using AI-generated environments for games or VR applications.

SceneTex [7] requires a conditional generator. To this end, we finetune our model for 4K iterations after the first stage as described in Section 3.2. Additionally, we randomly (with probability p=0.25) set one of the PBR maps to the ground truth and the corresponding timestep to t=0. This enables our model to be conditioned on any PBR input, similar to [23]. We render normal maps of the geometry from different viewpoints to condition our PBR generation. In the first stage, we optimize for the material properties, conditioned on the rendered normal. Since our model is based on Flow Matching [32], we also modify the distillation in SceneTex [7]. Concretely, we use the VFDS loss [18] in image space. To avoid over-smoothed results, we use CFG=10 and normalize the flow direction. We backpropagate to separate *uv*-textures for each property and follow the weighting scheme of [27]. We weight the loss with the observation frequency and represent the textures with a regularized multi-resolution Laplacian-pyramid to stabilize the updates for sparsely observed regions. In the second stage, we similarly optimize normal textures for fine geometric details, conditioned on the already obtained material properties. We represent the normal map in tangent space and regularize with the original geometry. For more samples and details, see the supplement.

5 Experiments

Training and Testing Details In the first stage, we train the LoRAs separately for 2K iterations with a batch size of 10, which takes 5h on a single NVIDIA A100 (80GB) GPU. In the second stage, we finetune for another 2.5K iterations with a batch size of 30 (10 aligned PBR maps), which takes 21h. We employ the Prodigy optimizer [43] in both stages. The LoRA layers use a rank of 64, which gives a total of 224M additional parameters. For inference, we use a single NVIDIA A6000 (48GB) GPU. Sampling a single image takes around 12 seconds.

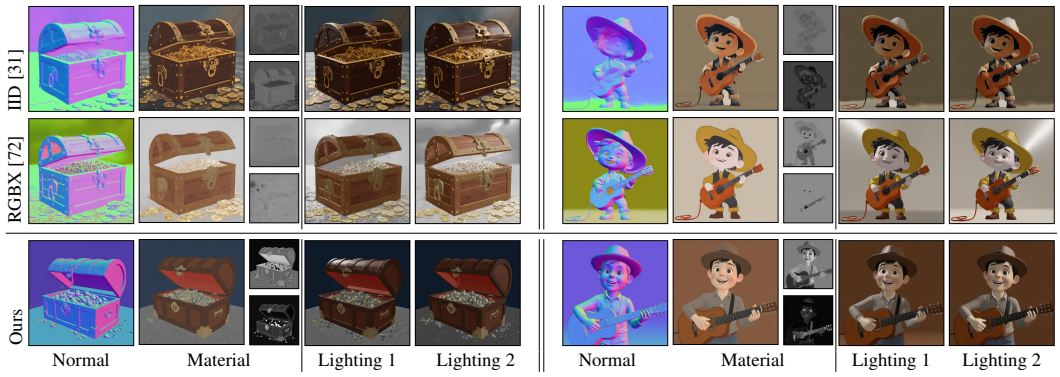

Figure 6: **Rendering comparisons**. We show sample PBR maps of our method and baselines as well as rendered RGB images under two different lighting conditions. We use a diverse set of text prompts to produce our PBR maps, as well as the input RGB images for the baseline methods. This highlights our models' capability to retain the generalized prior of the pretrained text-to-image model. Our method better captures the semantic meaning of the individual intrinsic properties. For example, there are no baked-in lighting effects in the albedo, and the metallic/roughness maps are sharper with more intricate details. This leads to more realistic renderings and lighting effects.

Table 1: **Baseline comparisons.** We compare the albedo quality for in-distribution (A-ID-FID) and out-of-distribution (A-OOD-FID) settings as well as perceptually with a user study (A-PQ). We evaluate the material quality with a user study focusing on the rendering quality (R-PQ), specularity quality (S-PQ), and prompt coherence (PC). Our method produces the best quality and it is preferred by most of the participants.

	A-ID-FID↓	A-OOD-FID↓	A-PQ↑	R-PQ↑	S-PQ↑	PC↑
IID [31]	78.77	98.77	14.24%	$2.95_{\pm 1.03}$	$2.82_{\pm 1.13}$	$4.47_{\pm 0.89}$
RGBX [72]	**61.36**	90.12	15.63%	$2.96_{\pm 0.98}$	$2.57_{\pm 1.07}$	$4.33_{\pm 0.93}$
ColorfulShading [5]	91.10	86.48	2.77%	N/A	N/A	N/A
IID[31] w/ FLUX-LoRA	103.36	79.29	N/A	N/A	N/A	N/A
w/o Rendering	78.77	72.23	N/A	$3.42_{\pm 0.92}$	$2.73_{\pm 0.93}$	$4.52_{\pm 0.78}$
w/o CIA-Dropout	71.47	75.54	N/A	$3.68_{\pm 0.87}$	$3.21_{\pm 1.18}$	$4.52_{\pm 0.76}$
Ours	72.09	**71.39**	**67.36%**	$\mathbf{3.93}_{\pm \mathbf{0.88}}$	$\mathbf{3.62}_{\pm \mathbf{0.96}}$	$\mathbf{4.62}_{\pm \mathbf{0.67}}$

Rendering Images During inference, we render RGB images following Equation (3) to obtain $\hat{\mathbf{I}}$. We use a slightly higher lighting intensity ($L_i{=}e^3$) than during training. Then we add an ambient color term: $\hat{\mathbf{I}}_{\text{amb}} = (1{-}\alpha)\hat{\mathbf{I}} + \alpha\hat{\mathbf{x}}_a$ with $\alpha{=}0.2$. Afterwards, we apply the tone mapping from [29]: $\hat{\mathbf{I}}_{\text{tone}}{=}\log(1+\mu\hat{\mathbf{I}}_{\text{amb}})/\log(1+\mu)$ with $\mu{=}64$. We empirically find this creates visually more pleasing RGB images. This also demonstrates the advantages of generating intrinsic image properties, i.e., we can arbitrarily render them post-generation. We list the used text prompts in the supplemental.

Baselines To the best of our knowledge, we are the first method to perform *direct* PBR map generation (from only text as input). Therefore, we compare our method against recent methods that perform intrinsic image decomposition, namely *IID* [31], *RGBX* [72], and *ColorfulShading* [5]. In contrast to our method, these works require an RGB image as input from which the PBR maps are generated. Unless noted otherwise, we generate the RGB image for the baselines by prompting our pretrained text-to-image model [32]. We only compare albedo quality against *ColorfulShading* [5], since they are decomposing an image into albedo and shading components, which does not allow for complete relighting (including specular effects) or editing effects.

Metrics We measure the quality of generated PBR maps through various metrics. First, we calculate the FID score [20] on in-distribution and out-of-distribution albedo images. For in-distribution (A-ID-FID), we use all 2595 albedo images from the InteriorVerse [76] test set and caption them based on the corresponding renderings with Florence-2 [66]. For each caption, we generate an albedo image, creating a total of 2595 generated albedo images. For out-of-distribution (A-OOD-FID), we evaluate on the pre-processed G-Buffer renderings [47] of ObjaVerse [10] (GObjaVerse). We take 1000 samples from the diverse "Daily-Used" category. As before, we generate an albedo map for each of the prompts, creating a total of 1000 generated albedo images. In both cases, we calculate FID against the respective ground-truth albedos.

Evaluating generated PBR maps remains a hard problem. To this end, we also conduct a user study and ask to rate the quality of albedo (A-PQ), specularity (S-PQ), rendered images (R-PQ), and the prompt coherence (PC). In total, we collect 2,274 data points from 36 participants and report averaged results (we refer to the supplementary material for more details).

5.1 Intrinsic Image Generation

We show qualitative comparisons against IID [31], RGBX [72] and [5] in Figure 6 using text prompts from [16], LLM-generated ones, and our own prompts. The baselines receive an RGB image as input, which was created with our pretrained text-to-image model, whereas we directly generate the PBR maps from only text as input. For a fair comparison, we also train another variant of IID on the same small dataset and same architecture as ours, i.e. we use LoRA fine-tuning of FLUX [32]. All methods showcase similar diversity, i.e., the generated images align well with the out-of-distribution text prompts. This showcases that our finetuned model still retains the generalized prior, which is also confirmed in the user study (Table 1, PC). Furthermore, our generated PBR maps are of higher quality, semantically more meaningful, and they closer resemble the expected distribution for physically-based rendering. This is because the baseline methods are trained on synthetic, indoor scenes [76] and are not designed to generalize their decomposition to out-of-distribution setups. Furthermore, intrinsic image decomposition is constrained to match the appearance of the input image, making it difficult to rather focus on the PBR distribution for out-of-domain samples. Additional albedo comparisons in Figure 7 as well as the quantitative comparisons in Table 1 confirm this observation. Our generated albedos are not oversmoothed, showing sharp details with flat colors. We provide more samples in the supplemental.

Figure 7: **Albedo comparisons**. We show albedo images of our method and baselines corresponding to the same text prompt in each column. Our albedo images have less baked-in shadows and reflections, which is desirable for downstream tasks, such as physically-based rendering. We provide more samples in the supplemental.

5.2 Ablations

The main technical contributions of our method are the cross-intrinsic attention (Section 3.2.1) and the rendering loss (Section 3.2.2). In the following, we highlight the importance of each component. We provide additional ablations in our supplementary material.

How important is the rendering loss? The rendering loss improves the quality of *all* PBR maps (see Figure 8 and Table 1). The additional supervision of Equation (4) provides more diverse gradients to the LoRA weights than the L2 loss of Equation (1). This way, the influence of the loss on the individual PBR maps is different and becomes grounded in image space through the rendering function, Equation (3). This leads to a better separation of the intrinsic properties, giving meaningful normal maps, detailed albedos without baked-in lighting effects, and sharper roughness/metallic maps without undesired texture or lighting patterns. Our importance-based light sampling strategy further improves the sharpness of roughness and metallic maps. In comparison, sampling light directions uniformly renders specular effects less often. This results in less pronounced PBR maps in Figure 8.

How important is the dropout in Cross-Intrinsic Attention? Without cross-intrinsic attention, we cannot create aligned PBR maps, because then there is no communication between batch elements during inference (see supplementary material). Additionally, we utilize dropout regularization on our cross-intrinsic attention. This technique motivates the model to preserve the prior of each intrinsic component during the 2nd stage alignment training. As can be seen in Figure 8 and Table 1, this increases the quality of both the rendered images and the PBR maps. The generated samples are sharper and do not suffer from noisy artifacts.

Can a generalized PBR prior help intrinsic image decomposition? We believe that intrinsic image decomposition methods are constrained to match the appearance of the input image. Thus, the model can learn to rely more on the input, instead of focusing on staying in the PBR distribution; therefore, facing an out-of-distribution input image poses a challenge to these models to produce faithful PBR maps, corresponding to the respective distributions. On the contrary, our model is trained with the key constraint to produce faithful PBR maps enabling better generalization. To verify this hypothesis, we fine-tune our model to make it image-conditional. We extend the set of input modalities with the rgb rendering and apply dropout similarly how we achieve a normal conditional variant for the room-scale scene texturing section 4. The resulting model can be conditioned on an input image during inference, similar to the baselines. This variant achieves 70.51 FID on the out-of-domain GObjaverse evaluation set (A-OOD-FID), outperforming direct intrinsic image decomposition training (IID w/ FLUX-LoRA), and even our base model. This shows that a pre-trained prior that directly models the underlying distribution is beneficial for generalization.

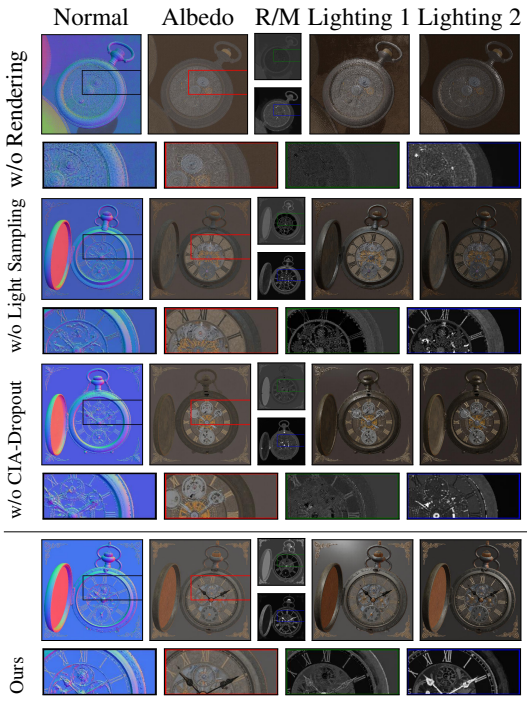

Figure 8: **Ablations**. We compare our full method against ablations that do not use the rendering loss (w/o Rendering), use uniform light sampling instead of importance-based light sampling (w/o Light Sampling), and do not use dropout in the cross-intrinsic attention (w/o CIA-Dropout). Without the rendering loss (Section 3.2.2), the PBR maps lose their semantic meaning, e.g., there are baked-in shadows in the albedo and the generated images appear "averaged out". Importance-based light sampling (Section 3.2.2) and CIA dropout (Section 3.2.1) both increase the sharpness of individual PBR maps, e.g., the roughness/metallic images have realistic details *without* baked-in textures. Overall, all components improve the quality of rendered images under varied lighting conditions. We provide more samples in the supplement.

6 Conclusion

We have presented IntrinsiX, the first method for *direct* generation of intrinsic image properties from text as input. We leverage the strong image prior of pretrained text-to-image models and convert it into a PBR map generator. We have introduced cross-intrinsic attention to produce semantically aligned PBR maps. Furthermore, we have shown that using our novel rendering loss with tailored light sampling provides important signal for the model to better ground each intrinsic component. Our approach allows us to generate high-quality, diverse results that go beyond the distribution of existing, synthetic datasets. Our method enables several downstream applications, such as physically-based rendering, material editing, relighting, and for the first time 3D scene PBR texture generation. We believe this showcases the potential that text-to-image models like ours can have on gaming and VR applications. Instead of generating content in shaded RGB space, we produce the PBR maps that can be directly used in standard computer graphics pipelines.

Acknowledgments and Disclosure of Funding

This work was supported by the ERC Consolidator Grant Gen3D (101171131) of Matthias Nießner, the German Research Foundation (DFG) Grant "Making Machine Learning on Static and Dynamic 3D Data Practical", and the German Research Foundation (DFG) Research Unit "Learning and Simulation in Visual Computing". We thank Angela Dai for the video voice-over.

References

[1] Maria Teresa Baldassarre, Danilo Caivano, Berenice Fernández Nieto, Domenico Gigante, and Azzurra Ragone. The social impact of generative AI: an analysis on chatgpt. *CoRR*, abs/2403.04667, 2024.

[2] Sean Bell, Kavita Bala, and Noah Snavely. Intrinsic images in the wild. *ACM Trans. Graph.*, 33 (4):159:1–159:12, 2014.

[3] Raphael Bensadoun, Tom Monnier, Yanir Kleiman, Filippos Kokkinos, Yawar Siddiqui, Mahendra Kariya, Omri Harosh, Roman Shapovalov, Benjamin Graham, Emilien Garreau, Animesh Karnewar, Ang Cao, Idan Azuri, Iurii Makarov, Eric-Tuan Le, Antoine Toisoul, David Novotný, Oran Gafni, Natalia Neverova, and Andrea Vedaldi. Meta 3d gen. *CoRR*, abs/2407.02599, 2024.

[4] Brent Burley and Walt Disney Animation Studios. Physically-based shading at disney. In *Acm Siggraph*, pages 1–7. vol. 2012, 2012.

[5] Chris Careaga and Yagiz Aksoy. Colorful diffuse intrinsic image decomposition in the wild. *ACM Trans. Graph.*, 43(6):178:1–178:12, 2024.

[6] 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.

[7] Dave Zhenyu Chen, Haoxuan Li, Hsin-Ying Lee, Sergey Tulyakov, and Matthias Nießner. Scenetex: High-quality texture synthesis for indoor scenes via diffusion priors. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 21081–21091. IEEE, 2024.

[8] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018.

[9] Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, Eli VanderBilt, Aniruddha Kembhavi, Carl Vondrick, Georgia Gkioxari, Kiana Ehsani, Ludwig Schmidt, and Ali Farhadi. Objaverse-xl: A universe of 10m+ 3d objects. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 13142–13153. IEEE, 2023.

[11] Valentin Deschaintre, Miika Aittala, Frédo Durand, George Drettakis, and Adrien Bousseau. Single-image SVBRDF capture with a rendering-aware deep network. *ACM Trans. Graph.*, 37 (4):128, 2018.

[12] Xiaodan Du, Nicholas I. Kolkin, Greg Shakhnarovich, and Anand Bhattad. Generative models: What do they know? do they know things? let's find out! *CoRR*, abs/2311.17137, 2023.

[13] Xiang Feng, Chang Yu, Zoubin Bi, Yintong Shang, Feng Gao, Hongzhi Wu, Kun Zhou, Chenfanfu Jiang, and Yin Yang. ARM: appearance reconstruction model for relightable 3d generation. *CoRR*, abs/2411.10825, 2024.

[14] Graham D. Finlayson, Mark S. Drew, and Cheng Lu. Intrinsic images by entropy minimization. In *Computer Vision - ECCV 2004, 8th European Conference on Computer Vision, Prague, Czech Republic, May 11-14, 2004. Proceedings, Part III*, pages 582–595. Springer, 2004.

[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*, pages 1–25, 2021.

[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. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024.

[17] Roger B. Grosse, Micah K. Johnson, Edward H. Adelson, and William T. Freeman. Ground truth dataset and baseline evaluations for intrinsic image algorithms. In *IEEE 12th International Conference on Computer Vision, ICCV 2009, Kyoto, Japan, September 27 - October 4, 2009*, pages 2335–2342. IEEE Computer Society, 2009.

[18] Jun Guo, Xiaojian Ma, Yikai Wang, Min Yang, Huaping Liu, and Qing Li. Flowdreamer: A RGB-D world model with flow-based motion representations for robot manipulation. *CoRR*, abs/2505.10075, 2025.

[19] Zeyu Han, Chao Gao, Jinyang Liu, Jeff Zhang, and Sai Qian Zhang. Parameter-efficient fine-tuning for large models: A comprehensive survey. *Trans. Mach. Learn. Res.*, 2024, 2024.

[20] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6626–6637, 2017.

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

[22] Lukas Höllein, Ang Cao, Andrew Owens, Justin Johnson, and Matthias Nießner. Text2room: Extracting textured 3d meshes from 2d text-to-image models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7909–7920, 2023.

[23] Lukas Höllein, Aljaž Božič, Norman Müller, David Novotny, Hung-Yu Tseng, Christian Richardt, Michael Zollhöfer, and Matthias Nießner. Viewdiff: 3d-consistent image generation with text-to-image models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5043–5052, 2024.

[24] Berthold KP Horn. Determining lightness from an image. *Computer graphics and image processing*, 3(4):277–299, 1974.

[25] 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. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

[26] Xin Huang, Tengfei Wang, Ziwei Liu, and Qing Wang. Material anything: Generating materials for any 3d object via diffusion. *CoRR*, abs/2411.15138, 2024.

[27] Yukun Huang, Jianan Wang, Yukai Shi, Xianbiao Qi, Zheng-Jun Zha, and Lei Zhang. Dreamtime: An improved optimization strategy for text-to-3d content creation. *CoRR*, abs/2306.12422, 2023.

[28] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 694–711. Springer, 2016.

[29] Nima Khademi Kalantari and Ravi Ramamoorthi. Deep high dynamic range imaging of dynamic scenes. *ACM Trans. Graph.*, 36(4):144:1–144:12, 2017.

[30] Peter Kocsis, Julien Philip, Kalyan Sunkavalli, Matthias Nießner, and Yannick Hold-Geoffroy. Lightit: Illumination modeling and control for diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 9359–9369. IEEE, 2024.

[31] Peter Kocsis, Vincent Sitzmann, and Matthias Nießner. Intrinsic image diffusion for indoor single-view material estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 5198–5208. IEEE, 2024.

[32] Black Forest Labs. Flux. `https://github.com/black-forest-labs/flux`, 2023.

[33] Edwin H Land and John J McCann. Lightness and retinex theory. *Josa*, 61(1):1–11, 1971.

[34] Zhengqin Li, Mohammad Shafiei, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. Inverse rendering for complex indoor scenes: Shape, spatially-varying lighting and SVBRDF from a single image. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 2472–2481. Computer Vision Foundation / IEEE, 2020.

[35] Zhengqin Li, Ting-Wei Yu, Shen Sang, Sarah Wang, Meng Song, Yuhan Liu, Yu-Ying Yeh, Rui Zhu, Nitesh B. Gundavarapu, Jia Shi, Sai Bi, Hong-Xing Yu, Zexiang Xu, Kalyan Sunkavalli, Milos Hasan, Ravi Ramamoorthi, and Manmohan Chandraker. Openrooms: An open framework for photorealistic indoor scene datasets. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 7190–7199. Computer Vision Foundation / IEEE, 2021.

[36] 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. *arXiv preprint arXiv: 2501.18590*, 2025.

[37] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023.

[38] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 9264–9275. IEEE, 2023.

[39] Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. Syncdreamer: Generating multiview-consistent images from a single-view image. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024.

[40] Ivan Lopes, Fabio Pizzati, and Raoul de Charette. Material palette: Extraction of materials from a single image. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 4379–4388. IEEE, 2024.

[41] Nanye Ma, Mark Goldstein, Michael S. Albergo, Nicholas M. Boffi, Eric Vanden-Eijnden, and Saining Xie. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part LXXVII*, pages 23–40. Springer, 2024.

[42] Xiaohe Ma, Valentin Deschaintre, Milos Hasan, Fujun Luan, Kun Zhou, Hongzhi Wu, and Yiwei Hu. Materialpicker: Multi-modal dit-based material generation. *ACM Trans. Graph.*, 44 (4):133:1–133:12, 2025.

[43] Konstantin Mishchenko and Aaron Defazio. Prodigy: An expeditiously adaptive parameter-free learner. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024.

[44] Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, and Ying Shan. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4296–4304, 2024.

[45] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: improving latent diffusion models for high-resolution image synthesis. *CoRR*, abs/2307.01952, 2023.

[46] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023.

[47] Lingteng Qiu, Guanying Chen, Xiaodong Gu, Qi Zuo, Mutian Xu, Yushuang Wu, Weihao Yuan, Zilong Dong, Liefeng Bo, and Xiaoguang Han. Richdreamer: A generalizable normal-depth diffusion model for detail richness in text-to-3d. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 9914–9925. IEEE, 2024.

[48] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with CLIP latents. *CoRR*, abs/2204.06125, 2022.

[49] Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Angel Bautista, Nathan Paczan, Russ Webb, and Joshua M Susskind. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10912–10922, 2021.

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

[51] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015.

[52] 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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 22500–22510. IEEE, 2023.

[53] 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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 22500–22510. IEEE, 2023.

[54] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L. Denton, Seyed Kamyar Seyed Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022.

[55] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5B: an open large-scale dataset for training next generation image-text models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022.

[56] Prafull Sharma, Varun Jampani, Yuanzhen Li, Xuhui Jia, Dmitry Lagun, Frédo Durand, Bill Freeman, and Mark J. Matthews. Alchemist: Parametric control of material properties with diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 24130–24141. IEEE, 2024.

[57] Li Shen, Ping Tan, and Stephen Lin. Intrinsic image decomposition with non-local texture cues. In *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2008), 24-26 June 2008, Anchorage, Alaska, USA*. IEEE Computer Society, 2008.

[58] Yawar Siddiqui, Tom Monnier, Filippos Kokkinos, Mahendra Kariya, Yanir Kleiman, Emilien Garreau, Oran Gafni, Natalia Neverova, Andrea Vedaldi, Roman Shapovalov, and David Novotný. Meta 3d assetgen: Text-to-mesh generation with high-quality geometry, texture, and PBR materials. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024.

[59] Kihyuk Sohn, Nataniel Ruiz, Kimin Lee, Daniel Castro Chin, Irina Blok, Huiwen Chang, Jarred Barber, Lu Jiang, Glenn Entis, Yuanzhen Li, Yuan Hao, Irfan Essa, Michael Rubinstein, and Dilip Krishnan. Styledrop: Text-to-image generation in any style. *CoRR*, abs/2306.00983, 2023.

[60] Shitao Tang, Fuyang Zhang, Jiacheng Chen, Peng Wang, and Yasutaka Furukawa. Mvdiffusion: Enabling holistic multi-view image generation with correspondence-aware diffusion. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[61] Shimon Vainer, Mark Boss, Mathias Parger, Konstantin Kutsy, Dante De Nigris, Ciara Rowles, Nicolas Perony, and Simon Donné. Collaborative control for geometry-conditioned PBR image generation. In *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part XIII*, pages 127–145. Springer, 2024.

[62] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008, 2017.

[63] Zhouxia Wang, Xintao Wang, Liangbin Xie, Zhongang Qi, Ying Shan, Wenping Wang, and Ping Luo. Styleadapter: A single-pass lora-free model for stylized image generation. *CoRR*, abs/2309.01770, 2023.

[64] Jiaye Wu, Sanjoy Chowdhury, Hariharmano Shanmugaraja, David Jacobs, and Soumyadip Sengupta. Measured albedo in the wild: Filling the gap in intrinsics evaluation. In *IEEE International Conference on Computational Photography, ICCP 2023, Madison, WI, USA, July 28-30, 2023*, pages 1–12. IEEE, 2023.

[65] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan Weixian Lei, Yuchao Gu, Yufei Shi, Wynne Hsu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 7589–7599. IEEE, 2023.

[66] Bin Xiao, Haiping Wu, Weijian Xu, Xiyang Dai, Houdong Hu, Yumao Lu, Michael Zeng, Ce Liu, and Lu Yuan. Florence-2: Advancing a unified representation for a variety of vision tasks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 4818–4829. IEEE, 2024.

[67] Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Haotian Tang, Yujun Lin, Zhekai Zhang, Muyang Li, Ligeng Zhu, Yao Lu, and Song Han. SANA: efficient high-resolution text-to-image synthesis with linear diffusion transformers. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025.

[68] Xiaogang Xu, Hengshuang Zhao, Vibhav Vineet, Ser-Nam Lim, and Antonio Torralba. Mtformer: Multi-task learning via transformer and cross-task reasoning. In *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part XXVII*, pages 304–321. Springer, 2022.

[69] Bowen Xue, Giuseppe Claudio Guarnera, Shuang Zhao, and Zahra Montazeri. Diffusion-based g-buffer generation and rendering. *CoRR*, abs/2503.15147, 2025.

[70] Shuai Yang, Yifan Zhou, Ziwei Liu, and Chen Change Loy. Rerender A video: Zero-shot text-guided video-to-video translation. In *SIGGRAPH Asia 2023 Conference Papers, SA 2023, Sydney, NSW, Australia, December 12-15, 2023*, pages 95:1–95:11. ACM, 2023.

[71] Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *CoRR*, abs/2308.06721, 2023.

[72] Zheng Zeng, Valentin Deschaintre, Iliyan Georgiev, Yannick Hold-Geoffroy, Yiwei Hu, Fujun Luan, Ling-Qi Yan, and Milos Hasan. Rgb↔x: Image decomposition and synthesis using material- and lighting-aware diffusion models. In *ACM SIGGRAPH 2024 Conference Papers, SIGGRAPH 2024, Denver, CO, USA, 27 July 2024- 1 August 2024*, page 75. ACM, 2024.

[73] Chenshuang Zhang, Chaoning Zhang, Mengchun Zhang, and In So Kweon. Text-to-image diffusion models in generative AI: A survey. *CoRR*, abs/2303.07909, 2023.

[74] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 3813–3824. IEEE, 2023.

[75] Qing Zhang, Jin Zhou, Lei Zhu, Wei Sun, Chunxia Xiao, and Wei-Shi Zheng. Unsupervised intrinsic image decomposition using internal self-similarity cues. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(12):9669–9686, 2022.

[76] Jingsen Zhu, Fujun Luan, Yuchi Huo, Zihao Lin, Zhihua Zhong, Dianbing Xi, Rui Wang, Hujun Bao, Jiaxiang Zheng, and Rui Tang. Learning-based inverse rendering of complex indoor scenes with differentiable monte carlo raytracing. In *SIGGRAPH Asia 2022 Conference Papers, SA 2022, Daegu, Republic of Korea, December 6-9, 2022*, pages 6:1–6:8. ACM, 2022.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The key contributions are summarized in the abstract as well as in the last paragraph of the introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Appendix C describes the limitations of our method.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: The paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We will release the training and testing codes along with our trained model weights upon acceptance. It should also be possible to reproduce the model based on our description in the paper and the supplementary.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We will release the training and testing codes along with our trained model weights upon acceptance.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Sections 3 and 5 and Appendices F and G describe the the necessary training and testing details.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We report the standard deviation for our user study results (table 1). The FID metrics are evaluating a statistical similarity between two distributions (between a large amount of generated samples), thus already providing information about the statistical significance of the experiments. Additionally, we show multiple generated samples with different seeds in Figure 11.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The first section of Section 5 describes the required resources.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Our work is in accordance with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We provide a discussion about the societal impact in Appendix D.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: We will add a section about ethical and out-of-scope usage in our code release similarly to [32] to ensure responsible usage.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We describe the licenses of all the assets in Appendix E.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We will release the code and the trained model weights with training and testing scripts. We provide details about the training in Section 5. We describe the licenses of all the assets in Appendix E.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

    Justification: We conduct a user-study with human participants. We describe the details in Appendix F and provide one example of the provided instructions in Figure 18.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [Yes]

    Justification: We conduct a user-study with human participants. We describe the details in Appendix F. Our user-study is anonymous and does not raise any risks for the participants.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

IntrinsiX:
High-Quality PBR Generation using Image Priors
— Supplementary material —

Peter Kocsis **Lukas Höllein** **Matthias Nießner**
Technical University of Munich

peter-kocsis.github.io/IntrinsiX/

A Additional Ablations

How important is the dataset size and diversity? In the first stage of training, we train 3 separate LoRAs, corresponding to the different intrinsic properties. We curate synthetic indoor scene examples from the InteriorVerse dataset [76]. We empirically find that we need a large dataset size for the roughness/metallic PBR maps to achieve reasonable understanding of the corresponding intrinsic distribution. In contrast, the albedo/normal maps can be learned from a much smaller dataset of only 20 samples. This is important to retain the generalizable prior of the pretrained text-to-image model (see Appendix A). We confirm this with additional experiments in Table 2, that compare the quality and diversity of generated albedo images for different dataset sizes. The in-distribution FID (A-ID-FID) measures the quality of the albedo (calculated on a subset of 100 test images of InteriorVerse [76], similar as in the main paper). The diversity metric (A-Diversity) compares the FID between the generated set of all images and the mean of the generated set. This measures if the distribution is collapsed and therefore signals how diverse the generated samples are. We can see that a dataset consisting of 20 samples does the best in terms of diversity, while still having reasonable albedo quality. Importantly, albedos trained on larger datasets also start to include baked-in lighting effects (see Figure 10). This motivates our choice to not increase the dataset size further. The final dataset consists of sampled images from the InteriorVerse dataset [76]. We sample images from the following room-types to curate 20 samples: 5 bedrooms, 5 kitchens, 5 livingrooms, 1 kidroom, 2 offices, 1 cabinet, 1 bathroom.

LoRA Rank We ablate the rank of the LoRA [25] modules in fig. 9. The rank determines the total number of trainable parameters. Therefore, with a too low rank, the number of trainable parameters are too low in order to achieve the domain shift. On the other hand, with a too high rank, we are introducing too many parameters, which will lead to forgetting; thus, negatively effecting the generalization. As a middle-ground, we chose rank 64.

Rank	16	32	64 (Ours)	128	256
A-OOD-FID	68.52	70.07	**67.25**	68.48	71.13

Figure 9: **Qualitative and quantitative comparison of albedo quality for different LoRA [25] ranks.** Too low rank fails to change the domain from rgb to the target albedo modality. On the other hand, a too high rank negatively impacts the generalization, resulting in unrealistic composition.

Table 2: **Quantitative comparison of albedo quality for different dataset sizes.** We observe that training with larger dataset might lead to slightly better albedo quality (A-ID-FID); however, the diversity (A-Diversity) and thus the generalization capabilities degrade. This motivates our choice for a small, curated dataset of 20 samples for the first stage finetuning of the albedo/normal LoRAs.

Dataset size	A-ID-FID ↓	A-Diversity ↑
10	220.28	284.93
20 (Ours)	187.83	**398.36**
100	161.51	369.43
1k	**154.58**	366.35
20k	155.64	352.04

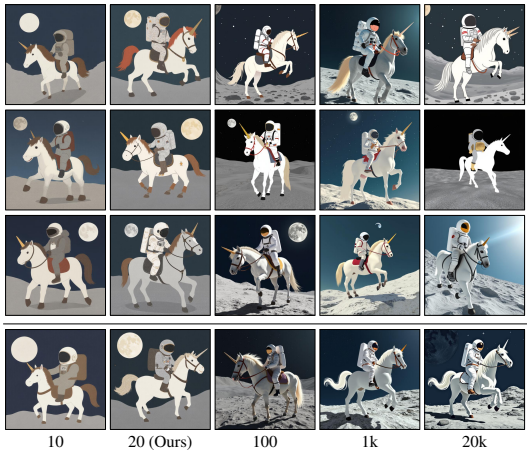

Figure 10: **Qualitative comparison of albedo quality for different dataset sizes.** Dataset sizes of 100 or more images tend to generate albedos with baked-in lighting effects, which is undesirable for physically-based rendering. A dataset that only consists of 10 images shows less details in generated albedos. This motivates our usage of 20 curated samples in the albedo/normal LoRA training, which balances both extrema. We show multiple samples per column, corresponding to different generations from the same text prompts. This highlights, that our model creates diverse images.

Can we maintain sample diversity? We show multiple samples using the same text prompt in Figure 11. Our method manages to maintain the generalization capabilities of the T2I model and generates diverse samples even for out-of-distribution prompts (see also Figure 6 and the supplementary material).

More samples We show additional qualitative comparisons in Figure 12.

Individual PBR Priors In the first stage of training, we train 3 separate LoRAs, corresponding to the different intrinsic properties. We curate synthetic indoor scene examples from the InteriorVerse dataset [76]. We show in Figure 13 (top) that this leads to high-quality and diverse albedo and normal map generations. This confirms our choice of training these PBR maps on small-scale datasets, i.e., we retain the generalized prior of the pretrained text-to-image model during the first stage finetuning.

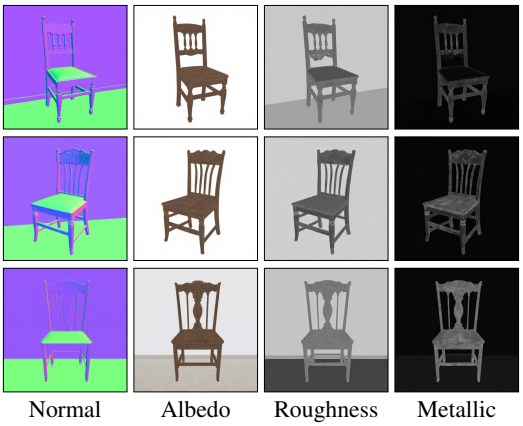

Figure 11: **Sample diversity**. We show 3 generated samples using the same text prompt. Our model predicts different samples and maintains the diversity of the T2I backbone (numerous chairs were not seen during training).

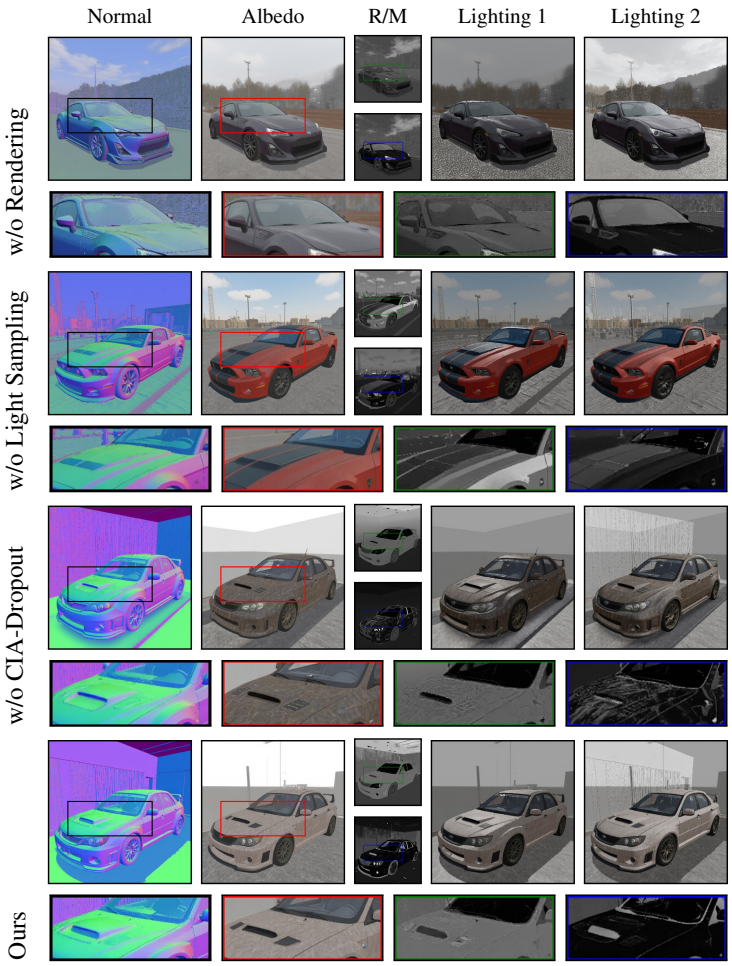

Figure 12: **Additonal ablations**. We compare our full method against ablations that do not use the rendering loss (w/o Rendering), use uniform light sampling instead of importance-based light sampling (w/o Light Sampling), and do not use dropout in the cross-intrinsic attention (w/o CIA-Dropout). Without the rendering loss (Section 3.2.2), the PBR maps lose their semantic meaning, e.g., there are baked-in shadows in the albedo and the generated images appear "averaged out". Importance-based light sampling (Section 3.2.2) and CIA dropout (Section 3.2.1) both increase the sharpness of individual PBR maps, e.g., the roughness/metallic images have realistic details *without* baked-in textures. Overall, all components improve the quality of rendered images under varied lighting conditions.

In contrast, the roughness/metallic LoRAs fail to generalize to out-of-distribution scenarios. This is because we use a larger dataset for training this LoRA. However, Figure 13 (bottom) shows that the second stage alignment training turns this LoRA to an equally-well generalizable PBR map generator. In other words, the generalizability of the albedo/normal LoRAs can be combined with the understanding of the intrinsic distribution of the roughness/metallic LoRA. Together, we can still produce high-quality, diverse PBR maps.

B Additional Results

Lighting Direction Sampling We sample light directions that maximize specular highlights to provide strong gradients about reflectance. As an alternative, we train a BRDF-sampled variant (Disney model) for more diverse lighting directions. This yields slightly worse results on the in-domain dataset (75.26 A-ID-FID), but further improves generalization (68.87 A-OOD-FID), showing

Figure 13: **Comparison between stage 1 and stage 2 samples**. In the first stage, we train 3 LoRAs separately corresponding to the different PBR maps (albedo, normal, roughness+metallic) on synthetic indoor-scene examples. In the second stage, we align these PBR priors through cross-intrinsic attention and the rendering loss. **Top:** generated images in the first stage (independently for each modality) show good quality for the albedo and normal maps. However, the roughness/metallic predictions are only reasonable for in-distribution scenarios (e.g. the 4th column) and become less detailed for out-of-distribution prompts. **Bottom:** after alignment training, all PBR maps have meaningful structure and exhibit sharp, high-quality content.

that lighting direction sampling is crucial for our task. Exploring other sampling strategies is a great avenue for future research.

Baseline comparisons We show additional comparisons to the baselines in Figure 14.

Albedo comparisons We show additional albedo comparisons to the baselines in Figure 15.

Scene Texturing Results We show more scene texturing results in Figure 16. We used Blender [8] to render the scene with uniform white environment map lighting and a single spherical light source. To enhance geometric details, we used an approximation of the displacement map by thresholding the normal textures.

To achieve room-scale scene texturing, we apply the SceneTex method [7] in a two-stage manner with the conditional variant of our model. First, we render normal maps from multiple views and generate material (albedo, roughness and metallic) textures, conditioned on the rendered normals. Then, we render the material properties for the given views and generate fine-grained normal details, conditioned on the rendered material maps. We use VFDS [18] [18] loss in image space. To balance the updates between over- and under-sampled texels, we weight the lostt with the inverse obervations frequency. As a pre-processing step, we create a texture, which stores the texel observation frequency. During the optimization, we render this texture together with the other components apply the weighting pixel-wise. We found that too low CFG value causes over-smoothed results, while too high

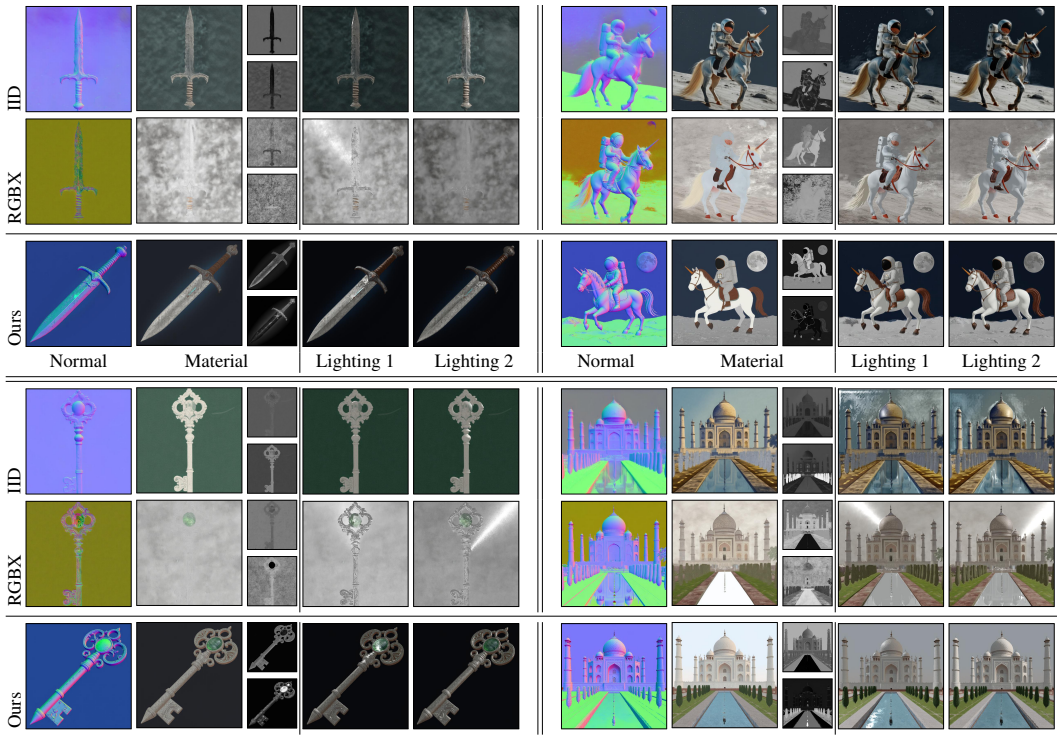

Figure 14: **Additional rendering comparisons**. We show sample PBR maps of our method and baselines as well as rendered RGB images under two different lighting conditions. We use a diverse set of text prompts to produce our PBR maps, as well as the input RGB images for the baseline methods. This highlights our models' capability to retain the generalized prior of the pretrained text-to-image model. Our method better captures the semantic meaning of the individual intrinsic properties. For example, there are no baked-in lighting effects in the albedo, and the metallic/roughness maps are sharper with more intricate details. This leads to more realistic renderings and lighting effects.

values can break the generated images in case of Flux [32]. To solve this issue, we normalize the flow direction to keep the norm of the text-conditional prediction, but use the direction towards the extrapolated flow direction.

C Limitations

We use a screen-space renderer similar to [31, 76]. For better 2D results, a neural/diffusion renderer can optionally be trained on top of our method. Our method maintains generalization far beyond the minimal training set, thanks to formulating the task directly in PBR space. Since FLUX does not inherently know about intrinsic components, we sacrifice some compositional diversity to enable our task using LoRA modules (see Figure 17). Ultimately, training with a diverse billion-scale PBR dataset would further improve generalization, but it does not exist.

Figure 17: **Limitations.** Since FLUX [32] does not inherently know about the intrinsic properties and we cannot train on a similarly large dataset as the model was originally trained, we sacrifice details during the fine-tuning. Therefore, our PBR maps do not contain as much details as an image generate by FLUX and sometimes the generated properties can be incorrect (e.g. normals).

Figure 15: **Additional albedo comparisons**. We show albedo images of our method and baselines corresponding to the same text prompt in each column. Our albedo images have less baked-in shadows and reflections, which is desirable for downstream tasks, such as physically-based rendering.

D Societal Impact

Generative models impact the society in general. Next to accelerating and democratizing creative content creation, they can also raise ethical concerns. Potential misuse for generating misinformation or deepfakes can become a major threat for naive users. These risks needs to be discussed and made public as soon as possible with open-sourcing and publishing results in the field to show the limits of current state-of-the-art. These challenges have been widely discussed in the recent years, such as in [1]. Our method enables decomposed generation, which coupled with a photo-realistic rendering can produce realistic-looking results.

E Licenses

Table 3 shows a summary about the licenses of the used components.

Table 3: **Licenses.** We provide a summary about the licenses of the used resources.

Type	Source	License
Code	`Flux 1.0 Dev`	Apache v2.0
Code	`SceneTex`	CC BY-NC-SA 3.0
Data	`InteriorVerse`	MIT License
Data	`3D-FRONT`	Custom, research-only
Data	`GObjaVerse`	Apache v2.0

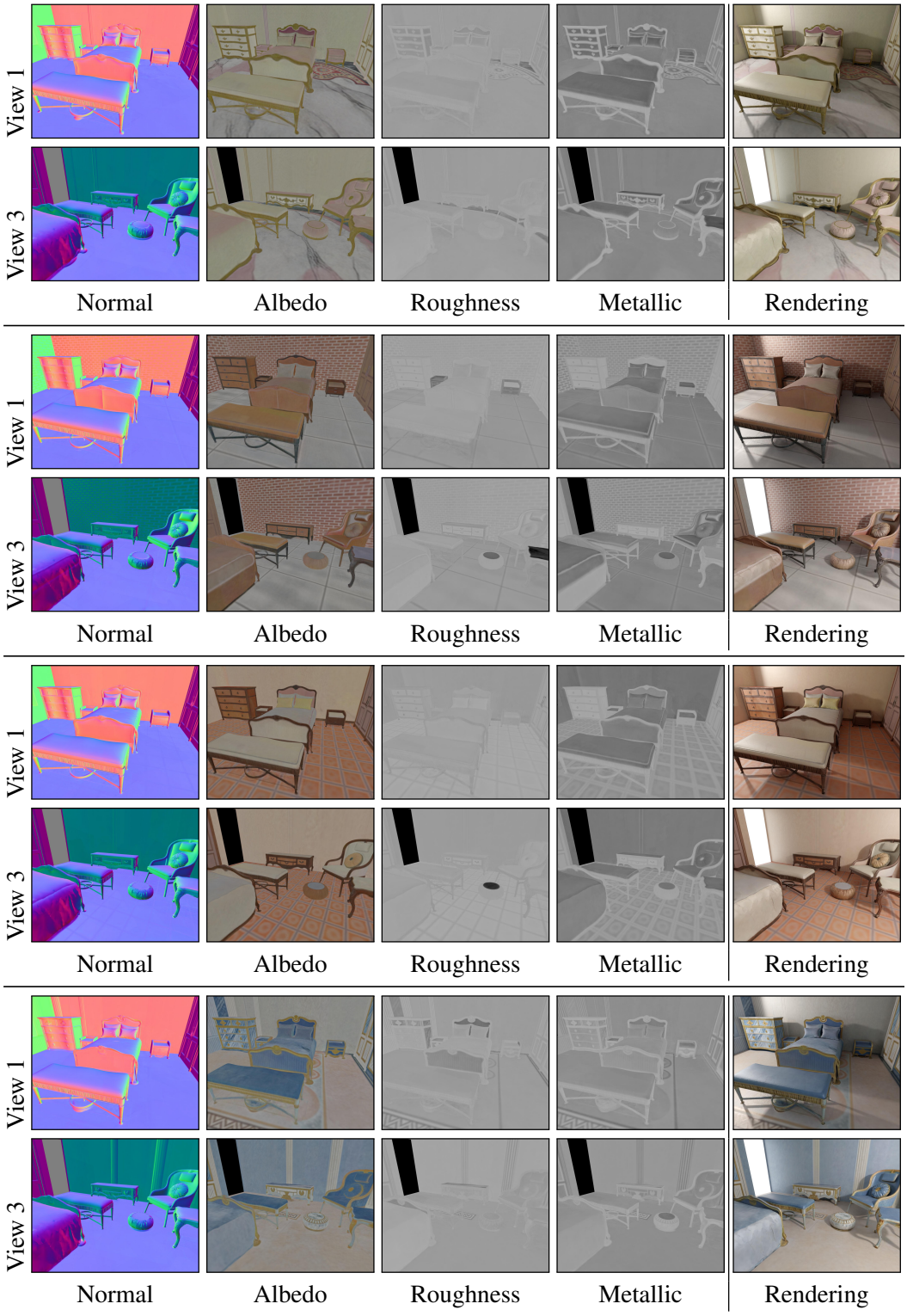

Figure 16: **Scene Texturing**. We show more scene texturing results on multiple 3D-Front scenes [15] with multiple prompts. Continues on the next page.

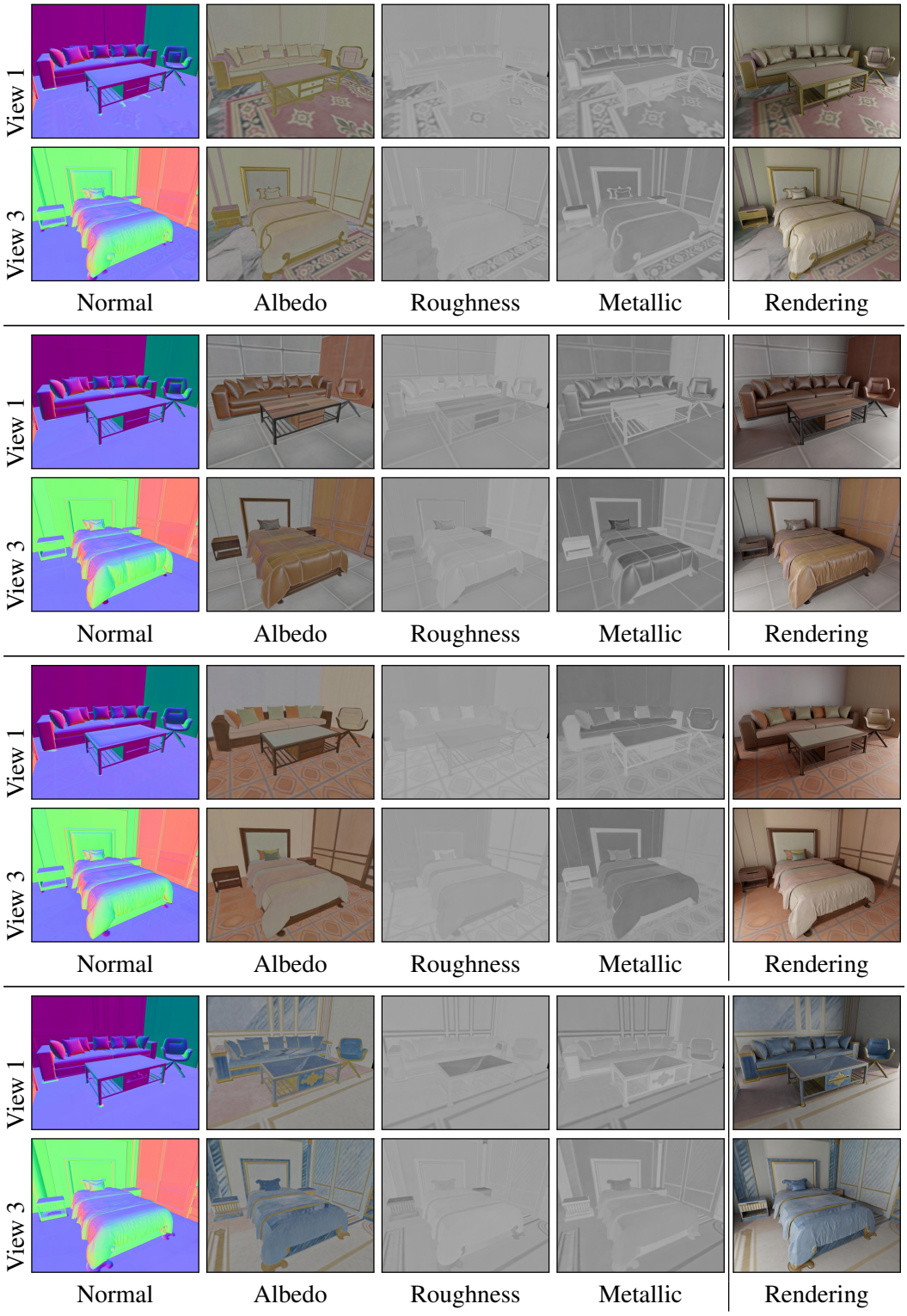

Figure 16: **Scene Texturing**. We show more scene texturing results on multiple 3D-Front scenes [15] with multiple prompts. Continues on the next page.

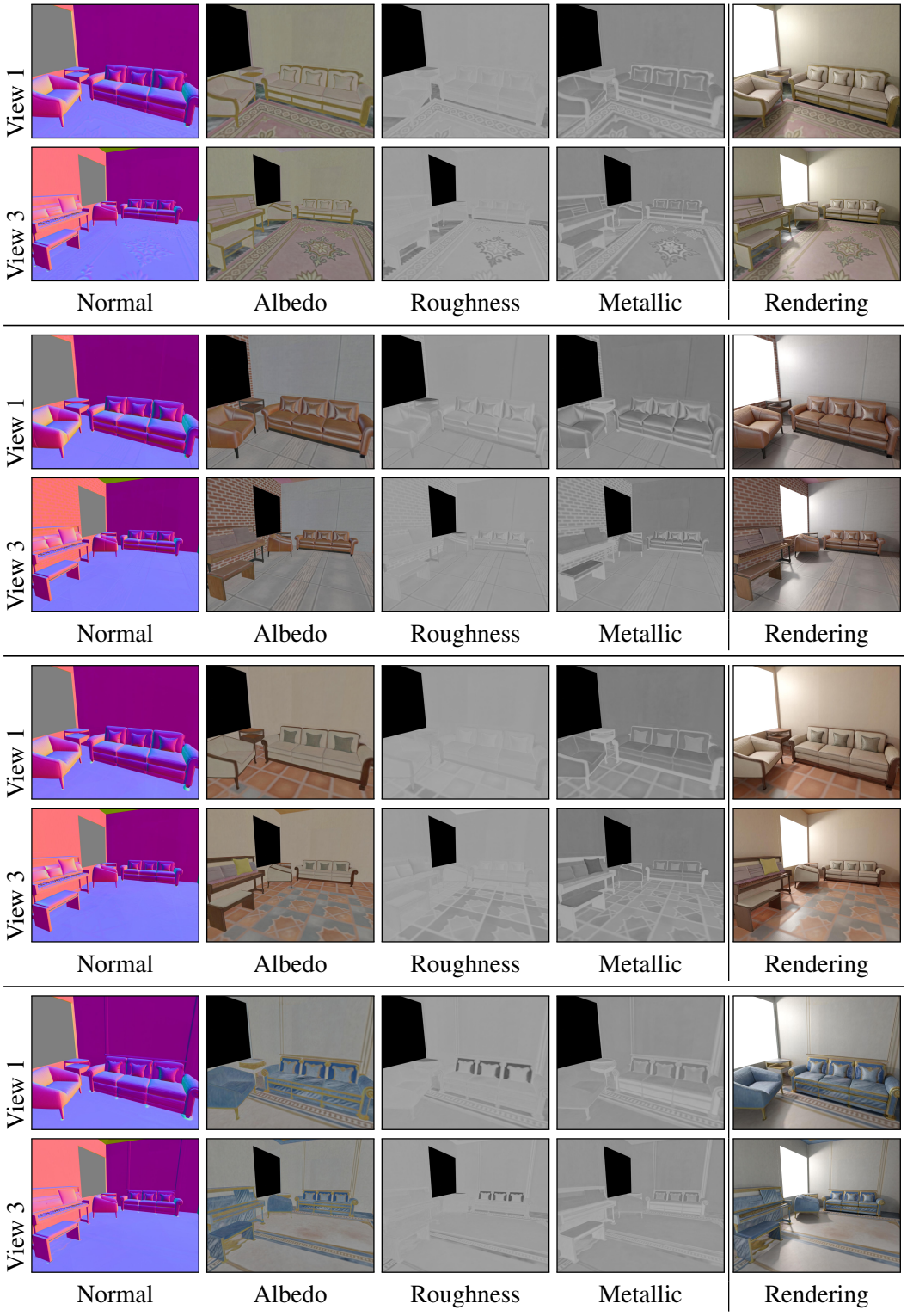

Figure 16: **Scene Texturing**. We show more scene texturing results on multiple 3D-Front scenes [15] with multiple prompts. Continues on the next page.

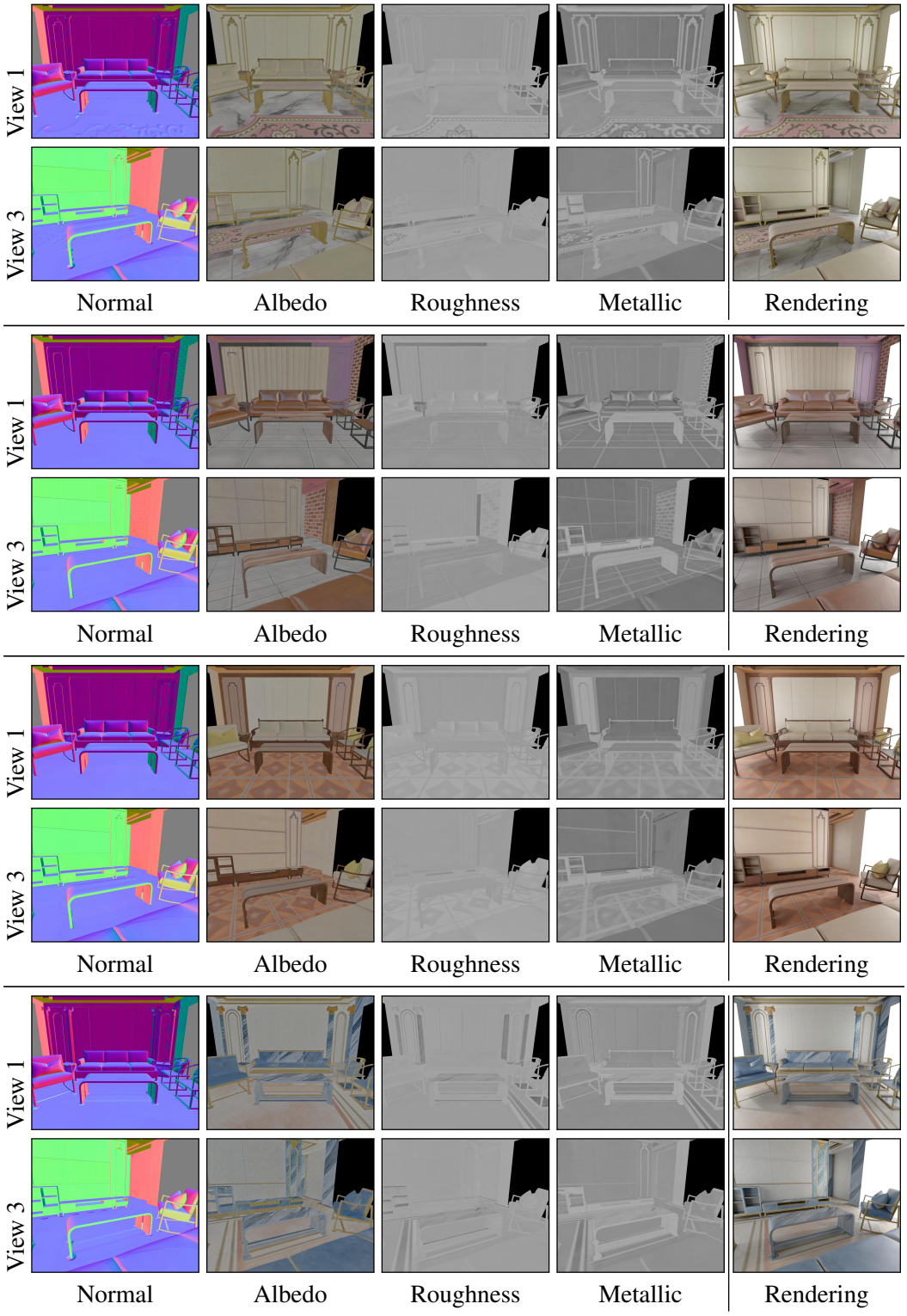

Figure 16: **Scene Texturing**. We show more scene texturing results on multiple 3D-Front scenes [15] with multiple prompts.

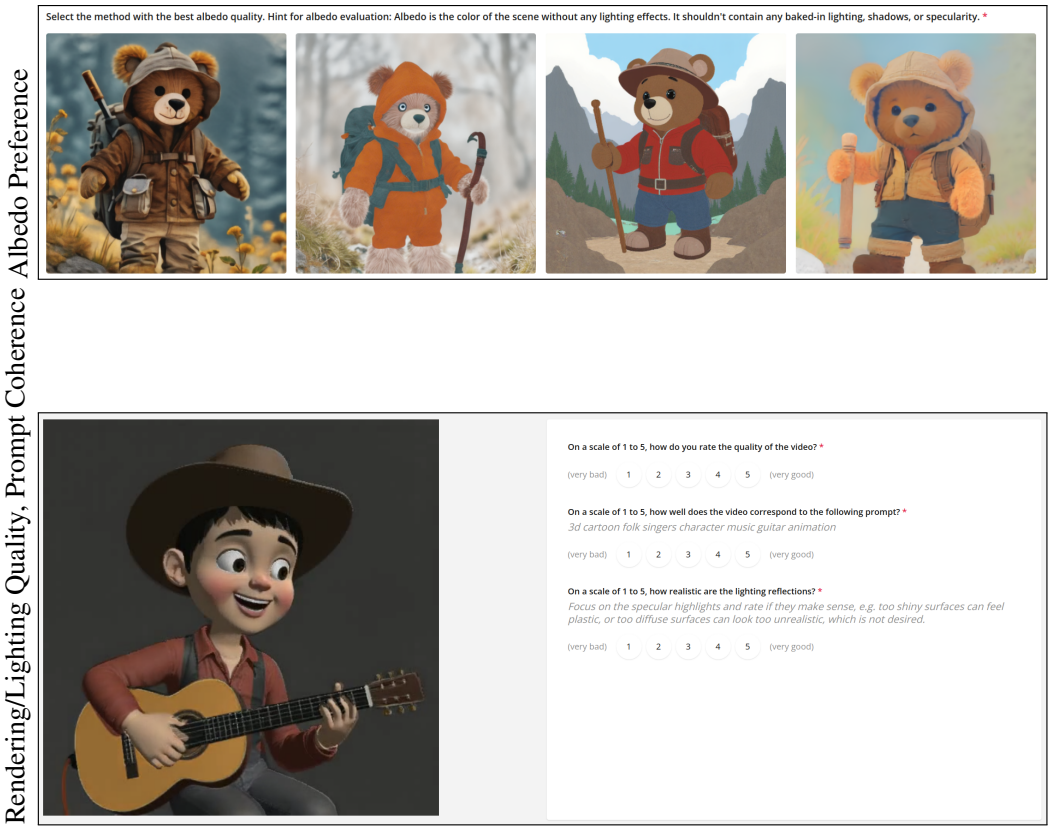

Figure 18: **Sample questions in the user study**. Users are presented with two types of questions. **Top:** users select the best albedo among all methods. **Bottom:** users rate the specular and rendered quality as well as the prompt coherence on a scale of 1-5 for a rendered video example.

F User Study

To better evaluate the quality of our generated PBR maps, we conduct a user study. The participation is anonymous and no personal data is collected. We summarize in Figure 18 the questions we asked by the participants. In the following, we explain how each metric is calculated.

- A-PP: we calculate the perceptual preference of albedo images (see Figure 18 top). Users choose one of the images and we calculate in percentage how often each method was preferred.
- S-PQ: we calculate the quality of specularity of the rendered video under varying lighting conditions (see Figure 18 bottom). Users rate on a scale of 1-5 how good the specular quality is.
- R-PQ: we calculate the general quality of the rendered video under varying lighting conditions (see Figure 18 bottom). Users rate on a scale of 1-5 how good the general quality is.
- PC: we calculate the prompt coherence, i.e, how well the text prompt matches the rendered video (see Figure 18 bottom). Users rate on a scale of 1-5 how good the coherence is.

G Prompts

We used the following prompts in our main results. We used our own, LLM-generated prompts, and prompts from Gao et al. [16]:

- Figure 1: “An astronaut riding a unicorn on the moon”
- Figure 2: “An astronaut riding a unicorn on the moon”
- Figure 4: “Astronaut in front of landscape space alien planet”
- Figure 5: “An industrial-style room with exposed brick walls, and reclaimed wood furniture, The room features a leather sofa, a coffee table made from a metal frame, and modern decor that complements its raw, edgy vibe”

- Figure 6 from left to right and top to bottom: “A wooden treasure chest reinforced with golden bands, its lid slightly ajar to reveal glittering jewels and coins, with faint beams of light spilling out from inside”, “3d cartoon folk singers character music guitar animation”,
- Figure 7 from left to right: “3d cartoon boy character animation”, “Adventurer standing in forest exploration nature trees hiking woodland outdoor”, “Adventurous teddy bear explorer travel outdoor”, “Alpaca wearing a suit animal clothing formal wool”, “Anime character in lab coat scientist cartoon drawing japanese style”,
- Figure 8: “A vintage pocket watch with its cover open, revealing a complex arrangement of gears and springs, some of which are glowing faintly, surrounded by engraved floral patterns.”
- Figure 9: “An astronaut riding a unicorn on the moon”
- Figure 10: “An astronaut riding a unicorn on the moon”
- Figure 11: “A wooden chair”
- Figure 12: “A sportcar”
- Figure 13 from left to right: “An astronaut riding a unicorn on the moon”, “3d cartoon folk singers character music guitar animation”, “Alien merchant extraterrestrial market fantasy science fiction”, “Alley city urban narrow passage architecture outdoor”, “Astronaut in front of landscape space alien planet”, “A majestic castle made entirely of ice, perched atop a snowy hill with shimmering pink and golden light reflecting off its towers. Below, a frozen lake mirrors the grandeur of the scene”, “A sprawling library with towering bookshelves reaching to the ceiling, glowing orbs floating mid-air to provide light, books that seem to fly on their own, and a spiral staircase made of golden wood.”, “A house in a forest”, “New York”, “A wooden chair”
- Figure 14 from left to right and top to bottom: “A rusted sword with a glowing blue rune etched into the blade, its hilt wrapped in weathered leather, and a faint aura of light surrounding it as if imbued with ancient magic”, “An astronaut riding a unicorn on the moon”, “A large, ornate key made of silver, with intricate vine-like patterns etched along the shaft and a glowing emerald embedded in the handle”, “Taj Mahal”
- Figure 15 from left to right: “Arches national park nature rock formations desert travel outdoor”, “Astronaut in colorful cave exploration adventure discovery geology outdoor”, “An epic battlefield where knights in shining armor clash with dragon-riding warriors under a stormy sky. A massive fire-breathing dragon is mid-flight, casting shadows over the chaos below”, “A massive sea turtle with a forest on its back swims through crystal-clear waters, accompanied by schools of colorful fish. A small sailing ship navigates beside it, dwarfed by the turtle’s size”, “A sleek, metallic helmet with a reflective visor that glows neon blue, featuring angular designs and small vents that emit a soft, white mist”
- Figure 17: “A very strange burger food unusual creative”
- Figure 16 from top to bottom: “An opulent Baroque-style room with intricate details, Walls are decorated with elaborate molding, in shades of cream, gold, and soft pastels, A plush velvet sofa, A richly patterned Persian rug covers the marble floor”, “An industrial-style room with exposed brick walls, and reclaimed wood furniture, The room features a leather sofa, a coffee table made from a metal frame, and modern decor that complements its raw, edgy vibe”, “A Tuscan-style room with warm earthy tones, terracotta tiles, and wrought iron details, The furniture features rich wood frames and soft cushions, complemented by Mediterranean-inspired decor”, “A breathtaking Greek-style room with intricate details, featuring a serene blue-and-white color scheme, Majestic marble columns with ornate Corinthian capitals support a high, coffered ceiling adorned with classical frescoes, The walls showcase elegant friezes and gold-accented moldings, reflecting the grandeur of ancient Greece, Large arched windows allow soft, natural light to flood the space, enhancing the contrast between crisp white walls and rich blue decorative elements, A luxurious chaise lounge with blue upholstery sits, accompanied by a marble-topped table with delicate carvings, The floor is adorned with intricate mosaic patterns”