Collaborative Control for Geometry-Conditioned PBR Image Generation

Shimon Vainer[1,⋆], Mark Boss[2,†], Mathias Parger[1], Konstantin Kutsy[1], Dante De Nigris[1], Ciara Rowles[1], Nicolas Perony[1], and Simon Donné[1,⋆]

Unity Technologies[1] Stability AI, work done while at Unity Technologies[2]
⋆ Equal Contributions † Core Technical Contributions
Corresponding author: shimon.vainer@unity3d.com

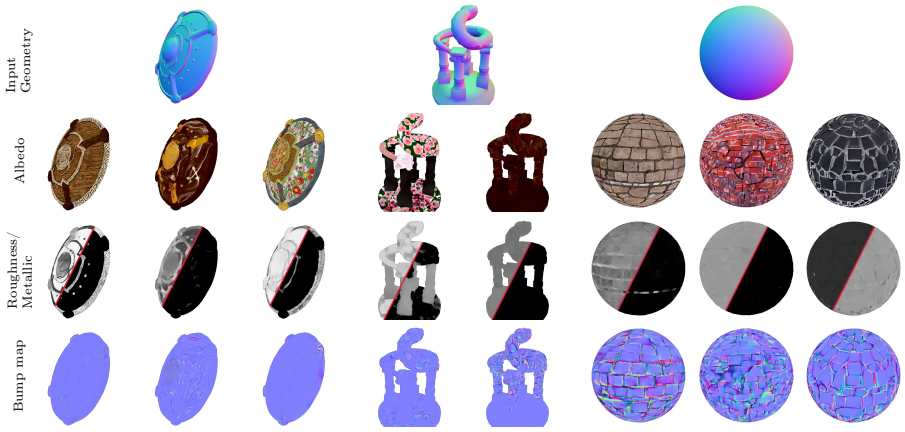


Fig. 1: Generated PBR materials. By tightly linking the PBR diffusion model with a frozen RGB model, we produce high-quality PBR images conditioned on geometry and prompts. Visit the project page at https://unity-research.github.io/holo-gen.

Abstract. Graphics pipelines require physically-based rendering (PBR) materials, yet current 3D content generation approaches are built on RGB models. We propose to model the PBR image distribution directly, avoiding photometric inaccuracies in RGB generation and the inherent ambiguity in extracting PBR from RGB. As existing paradigms for cross-modal fine-tuning are not suited for PBR generation due to both a lack of data and the high dimensionality of the output modalities, we propose to train a new PBR model that is tightly linked to a frozen RGB model using a novel cross-network communication paradigm. As the base RGB model is fully frozen, the proposed method retains its general performance and remains compatible with *e.g.* IPAdapters for that base model.

Keywords: Image Generation, Material Properties, Multi-Modal Generation, Physically-Based Rendering

1 Introduction

The recent meteoric rise of diffusion models has made at-scale generation of high-quality RGB image content more accessible than ever and Text-to-Texture and Text-to-3D approaches successfully lift this to 3D [34]. But to maximize the usefulness of the generated textures in downstream 3D workflows, generated content must be compatible with physically-based rendering (PBR) pipelines for proper shading and relighting. Current approaches rely on generated RGB images and subsequent PBR extraction through inverse rendering, suffering from the physically **in**accurate lighting in the generated RGB images as well as from significant ambiguities in the inverse rendering. We propose a solution for geometry-conditioned generation of PBR images by modeling the joint distribution directly, avoiding the issues around photometric consistency and inverse rendering.

To model the distribution of non-RGB modalities, existing approaches typically fine-tune the weights of a base RGB model [11, 29, 33, 40, 41, 58]. Applied to PBR images, this means either directly predicting the entire PBR image stack or sequentially predicting them conditioned on one another. Neither is sufficient for our use-case: jointly predicting the entire PBR image stack is problematic as the higher-dimensional modality does not compress well into the established latent spaces (as we show in Sec. 5), and sequentially predicting the elements of the PBR image stack is significantly more expensive and risks compounding errors in the sequential generation. Furthermore, while state-of-the-art RGB diffusion models are trained on billions of images [52], there is unfortunately no dataset of such size at our disposal for PBR content generation. Instead, the largest available dataset of PBR content is Objaverse [9], containing around 800,000 objects with associated PBR textures, limited to "everyday" appearances of the objects. In light of the restricted training data available, fine-tuning the base model results in catastrophic forgetting, forfeiting generalizability, as we illustrate in Sec. 5.

Instead, we keep a pre-trained RGB image model frozen and train a parallel model to generate PBR images, as shown in Fig. 2. We tightly link the PBR model to the frozen RGB model using our proposed cross-network control paradigm, in order to leverage its expressivity and rich internal state. As a result we are able to generate qualitative and diverse PBR content, even for unlikely appearances of objects (far out-of-distribution for the Objaverse dataset). Crucially, the frozen RGB model safeguards against catastrophic forgetting *and* remains compatible with techniques such as IPAdapter [73]. In summary, we:

1. Propose the novel *Collaborative Control* paradigm to tightly link the PBR generator to a fully frozen pre-trained RGB model, modeling the joint distribution of RGB and PBR images directly (see Sec. 4.1),
2. Illustrate that the proposed control mechanism is data-efficient, and generates high-quality images even from a very restricted training set,
3. Demonstrate the compatibility with IPAdapter [73] specifically, and
4. Ablate our design choices to show the improvement over existing paradigms in literature and the issues with existing paradigms.

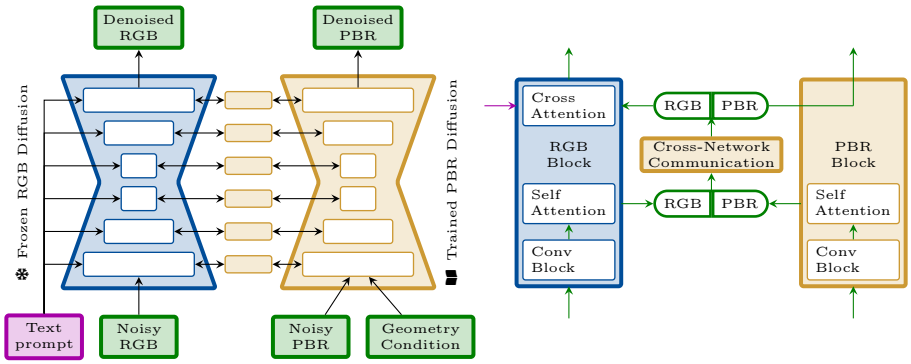


Fig. 2: Collaborative Control. Two parallel models collaborate to generate pixel-aligned outputs of different modalities. We freeze the left pre-trained RGB model and train the right PBR model with its cross-network communication layers. The cross-communication concatenates the states of both models, processes them with a small MLP, and residually distributes the result back to the respective models. As discussed in Sec. 5, prompt cross-attention in the PBR model is counter-productive.

2 Related Work

Generating natural images from text prompts. Natural image generation has a long history: from GANs [14, 25–28] and VAEs [30], to autoregressive models [45, 64]. More recently, the introduction of diffusion models [18, 56, 57] was a breakthrough in the generative field — far more stable than typical GAN training, albeit slow and computationally expensive, and easier to control and condition. Unfortunately, these approaches require billions of images to train from scratch [52], and for PBR image generation the largest commonly available dataset is Objaverse [9]. With 800,000+ objects it is still several orders of magnitude smaller than LAION-5B [52] and proves insufficient to train generative models that can generalize to unlikely semantics (as illustrated in Sec. 5). While pre-trained RGB models encode rich prior knowledge around structure, semantics, and materials [11,53,59], Sarkar *et al*. warn that the models are often still geometrically inaccurate [50]: we have found that this extends to material properties, as diffusion models prefer idealized and artistic appearances over photometric accuracy.

Generating non-RGB modalities Existing works fine-tune pre-trained RGB models to predict *e.g*. Depth [29, 58], semantics [33] or intrinsic properties [11, 40, 41], either directly or through LoRA's [19]. Sadly, this is not plausible for PBR image generation: compressing PBR images into the existing low-dimensional latent space overloads it, and the alternative of sequentially predicting channel triplets is too costly and slow. Wonder3D [41] and UniDream [40] perform joint RGB and normal diffusion using a cross-domain self-attention aligning the two

parallel branches, yet this scales poorly an increasing number of output modalities. Our proposed approach instead uses a frozen RGB model, negating the risk of catastrophic forgetting, and trains a parallel branch for all additional modalities jointly (in the latent space of a PBR VAE), reducing cost.

Image-based conditioning Existing pixel-accurate control techniques come in two flavors: re-training of the base model with modified input spaces [12, 29], and training of a parallel model that affects the base model's state [10, 20, 79]. We find in Sec. 5 that the former risks losing the base model's expressiveness and quality. In ControlNet [79] and ControlNet-XS [10], the controlling model only influences the base RGB model's output while in AnimateAnyone [20] the parallel model is only tasked with generating its own output. Instead, we leave the RGB base model's weights fully frozen and residually edit its internal states from a parallel model that is itself tasked with generating PBR images: our PBR model both *controls* the base model (to guide it towards the domain of rendered images), and *generates* its own PBR output (based on the RGB model's internal state); therefore, our proposed approach requires full bidirectional connections between both branches as shown in Fig. 5. To condition on input geometry we concatenate it to the input of the PBR branch, as Ke *et al.* [29].

Text-to-3D describes the task of generating 3D objects from text prompts, often with the aim to support downstream graphics pipelines such as game engines. Earlier methods leverage Score Distillation Sampling [46] (SDS) to iteratively optimize a 3D representation by backpropagating the diffusion model's noise predictions [15,22,35,37,42,43,55,61,62,66–68,71,80,81] through the RGB model, or building on viewpoint-aware image models [21, 38, 39, 41, 47, 54, 76] for direct fusion. Such RGB methods ignore that object appearance varies with viewing angle, often resulting in artifacts around highlights, and their RGB output is not useful in graphics pipelines. More recent work generates PBR properties so using inverse rendering with a differentiable renderer [7, 40, 70, 72, 74]: a major concern is lighting being baked into the material channels (*e.g.* HyperDreamer [70] uses an ad-hoc regularization to reduce these artifacts). **Text-to-Texture** methods restrict the Text-to-3D problem to objects with known structure by conditioning the diffusion model on the object geometry [4, 6, 31, 32, 75, 77, 78], but face similar issues by operating in the RGB domain. Paint3D [77] also discusses the lighting artifacts typical with inverse rendering and introduces a custom post-processing diffusion model to alleviate these. By directly generating PBR content, our proposed technique promises to resolve issues related to inverse rendering in the latter methods, all the while retaining the simplicity of the former methods.

Evaluation metrics for generative methods compare the output distributions with known ground-truth distributions, typically with the Inception Score [49] (IS) or the Fréchet Inception Distance [17] (FID). CMMD [23] argues that neither is well suited to modern generative models, and compare the distributions of CLIP embeddings rather than Inceptionv3 [60] internal states.

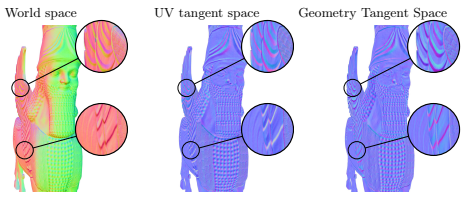


Fig. 3: Bump map. Similar surface bumps in world space (left) are dissimilar in the UV tangent space (middle) because of the arbitrary UV mapping. Representing the bump map in a tangent space solely dependent on the geometry (right) resolves this issue.

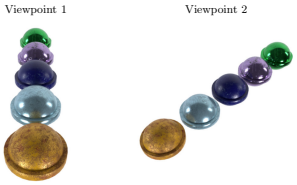


Fig. 4: Rendering function. The dataset is constructed so that the lighting remains constant with respect to the camera, simplifying the rendering function f_{RGB}: notice the similar highlight location.

Aside from comparing modelled distributions with the ground truth, we also wish to evaluate the alignment of the generated images to their text prompts. CLIPScore [16] compares the image's CLIP embedding with that of the prompt: whether all the relevant elements are represented and whether any extraneous elements were introduced. We also report the OneAlign *aesthetics* and *quality* metrics of the generated images [69], which have been shown to align well with human perception, to provide a more quantitative indication of quality.

3 Preliminaries

PBR materials are a compact representation of the bidirectional reflectance distribution function (BRDF), which describes how light is reflected from the surface of an object. We use the popular Cook-Torrance analytical BRDF model [8], using specifically the Disney BRDF Basecolor-Metallic parametrization [3] as it inherently promotes physical correctness. In this parametrization, the BRDF comprises *Albedo* ($\boldsymbol{b}_a \in \mathbb{R}^3$), *Metallic* ($\boldsymbol{b}_m \in \mathbb{R}$), and *Roughness* ($\boldsymbol{b}_r \in \mathbb{R}$) components. To increase realism during rendering beyond the resolution of the underlying geometry (often a mesh), graphics pipelines add small details such as wood grain or grout between tiles by encoding them as offsets to the surface normals in an additional *bump map* ($\boldsymbol{b}_n \in \mathbb{R}^3$). As this bump map is typically defined in a tangent space based on an arbitrary UV-unwrapping, it entangles the surface property with this arbitrary UV mapping. Instead, we propose to predict the bump map defined in a tangent space based solely on the object geometry, disentangling the texture from the UV mapping as shown in Fig. 3. To construct this geometry tangent space for a point $\boldsymbol{p} = [p_x, p_y, p_z]^T$ with geometry normal $\boldsymbol{n}$, we construct the local tangent vector as $\boldsymbol{t} = \boldsymbol{n} \times ([-p_y, p_x, 0]^T \times \boldsymbol{n})$, corresponding to Blender's *Radial Z* geometry tangent. The geometry tangent space is then constructed as $(\boldsymbol{t}/\|\boldsymbol{t}\|, \boldsymbol{n} \times \boldsymbol{t}/\|\boldsymbol{t}\|, \boldsymbol{n})^T$.

Diffusion models [18,56] iteratively invert a forward degradation process to generate high-quality images from pure noise (typically white Gaussian noise). Formally, the forward process iteratively degrades images from the data distribution $\boldsymbol{z}_0 \sim p(\boldsymbol{z})$ to standard-normal samples $\boldsymbol{z}_T \sim \mathcal{N}(0, I)$ over the course of T degradation steps as $\boldsymbol{z}_t \sim \mathcal{N}(\alpha_t \boldsymbol{z}_{t-1}, (1-\alpha_t)I)$, where α_t denotes the noise schedule for timestep t. Practically, the forward process can be condensed into the direct distribution $\boldsymbol{z}_t \sim \mathcal{N}(\sqrt{\bar{\alpha}_t}\boldsymbol{z}_0, (1-\bar{\alpha}_t)I)$ with the appropriate choice of $\bar{\alpha}_t$. The diffusion model $\mathcal{D}$ is trained to sample the stochastic reverse process $\mathcal{D}_t(\boldsymbol{z}_t) \sim p(\boldsymbol{z}_{t-1}|\boldsymbol{z}_t)$ to iteratively generate $\boldsymbol{z}_0$ from $\boldsymbol{z}_T$.

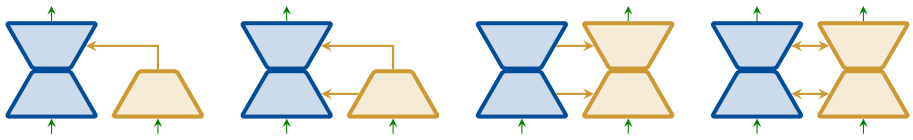
4 Method

We wish to train a PBR diffusion model $\mathcal{D}_{pbr}$ that models the reverse denoising process for PBR images as represented in the latent space of a VAE [48], representing the data distribution $p(\boldsymbol{z}_{pbr})$. We find that we lack the data required to train this model directly, and instead propose to model $p(\boldsymbol{z}'_{rgb} := f_{rgb}(\boldsymbol{z}_{pbr}), \boldsymbol{z}_{pbr})$ based on an RGB diffusion model $\mathcal{D}_{rgb}$ for the RGB data distribution $p(\boldsymbol{z}_{rgb})$; f_{rgb} is a rendering function that projects the PBR images onto the RGB domain. To motivate this, we split the joint reverse process into two separate processes using Bayes' rule:

$$\begin{aligned} &p(\boldsymbol{z}'_{rgb,t-1}, \boldsymbol{z}_{pbr,t-1} | \boldsymbol{z}'_{rgb,t}, \boldsymbol{z}_{pbr,t}) \\ &\quad \sim p(\boldsymbol{z}'_{rgb,t-1} | \boldsymbol{z}'_{rgb,t}, \boldsymbol{z}_{pbr,t}) p(\boldsymbol{z}_{pbr,t-1} | \boldsymbol{z}'_{rgb,t-1}, \boldsymbol{z}'_{rgb,t}, \boldsymbol{z}_{pbr,t}) \end{aligned} \tag{1}$$

The RGB model is implemented based on $\mathcal{D}_{rgb}(\boldsymbol{z}_{rgb,t-1}) \sim p(\boldsymbol{z}_{rgb,t-1}|\boldsymbol{z}_{rgb,t})$: we align the current RGB sample with the PBR sample and restrict it to $\mathrm{Im}(f)$ (the domain of rendered images with the fixed environment map) so that its internal states are more easily interpreted by the PBR branch[1]. To simplify this alignment problem, the rendering function f_{rgb} uses fixed camera settings and a fixed environment map as shown in Fig. 4. The PBR model now no longer models $p(\boldsymbol{z}_{pbr,t-1}|\boldsymbol{z}_{pbr,t})$: it additionally has access to the RGB context $(\boldsymbol{z}'_{rgb,t-1}, \boldsymbol{z}'_{rgb,t})$ which simplifies the problem. The RGB and PBR models are in practice much more intertwined than Eq. (1) implies: this derivation serves mostly as an intuitive indication for why the joint problem is more tractable. Note that $\boldsymbol{z}'_{rgb,t}$ is a degraded version of $\boldsymbol{z}'_{rgb,0}$, and *not* a rendered version of $\boldsymbol{z}_{pbr,t}$: the PBR model does not learn to do inverse rendering in degraded image space but rather learns to denoise PBR images given additional RGB context.

[1] Our intuition as to why a fixed environment map is beneficial is that it makes the RGB model's internal states more consistent to interpret, and makes the control problem of projecting to $\mathrm{Im}(f)$ simpler. Early in training, generated sample quality can be boosted significantly by applying the foreground mask to the RGB estimate for the first few timesteps; a rough projection to bring the estimate much closer to $\mathrm{Im}(f)$. After longer training, this is no longer necessary, as the PBR branch is capable enough to restrict the RGB branch to $\mathrm{Im}(f)$.



(a) ControlNet (b) ControlNet-Xs (c) AnimateAnyone (d) Proposed

Fig. 5: High-level overview of communication in (a) ControlNet [79], (b) ControlNet-XS [10], (c) AnimateAnyone [20] and (d) our proposed Collaborative Control approach. Blue represents frozen blocks, while orange elements are optimized during training.

4.1 Collaborative Control

In summary, our proposed approach comprises two models working in tandem: a pre-trained RGB image model and a new PBR model (see Fig. 5 for a high-level overview of our proposed control scheme). The previous section identifies two tasks for this cross-network communication: aligning the RGB model's output with both the PBR model's output and the map of the rendering function $\mathrm{Im}(f)$, and communicating knowledge in the RGB model to the PBR model. ControlNet [79] and ControlNet-XS [10] discuss solutions to the former control problem — the authors conclude that communication from the base model's encoder to the controlling model's encoder, and from the controlling model's decoder to the base model's decoder, is sufficient. AnimateAnyone [20] addresses the latter problem and concludes that, there, uni-directional communication from the left model to the right model is sufficient. We have found that full bidirectional communication is crucial for our approach: the PBR branch needs to extract relevant information from the RGB model's hidden state, while simultaneously guiding the RGB output towards render-like images (*i.e.* with a black background and compatible lighting) to ensure those hidden states are consistent with its own expectations. We dub this *Collaborative Control*.

We implement the cross-network communication as a connecting layer between the two models after every self-attention module; its inputs are the concatenation of the model states and its outputs are residually distributed to both models again. During training, we only optimize the weights of the PBR model and the cross-network communication links against both models' outputs, while the RGB model remains fully frozen. By adopting this approach, we safeguard the base RGB model's weights, and do not risk catastrophic forgetting for that base model. As we discuss in Sec. 5, we have found that a single per-pixel linear layer is sufficient, although we also evaluate the other control schemes from Fig. 5 as well as an attention-based communication layer. Notably, we have also found that disabling the text cross-attention in the PBR model is crucial to out-of-distribution performance; we attribute this to overfitting on the restricted dataset, as this problem worsens with reduced training data. Only allowing prompt attention through the frozen RGB model prevents such overfitting.

4.2 Implementation

Compressing PBR images into latent space RGB diffusion models benefit immensely from a dedicated VAE to encode the images into a lower-dimensional latent space [48]. Existing solutions that generate an alternate modality typically encode that modality with the RGB VAE, but PBR images cannot be compressed into the same latent space due to the higher dimensionality. Instead, we could select channel triplets $\boldsymbol{b}_a$, $[\boldsymbol{b}_m, \boldsymbol{b}_r, \mathbf{0}]$, and $\boldsymbol{b}_n$ and process those with the RGB VAE, but we instead choose to train a dedicated PBR VAE — our ablation studies indicate that the distribution mismatch between the PBR channels and the RGB space is too large, and performance would otherwise suffer. We adopt the VAE architecture and training code from StableDiffusion v1.5 [48], although following Vecchio et al. [65] we set the latent space channel count to 14 for the optimal balance between quality and compression when processing PBR images.

Conditioning on existing geometry We concatenate the screen-space geometry normals to the PBR model's inputs to condition the joint output. Referring to Fig. 5, Collaborative Control encapsulates the ControlNet scheme that would typically be used for this conditioning [10]: as we jointly train from scratch, this does not introduce additional cost.

Generating training data Our dataset for training both the PBR VAE and the Collaborative Control scheme is based on Objaverse [9]: a dataset containing 800,000+ 3D models with annotations for what the models represent (describing both shape and texture). After sanitizing and filtering the dataset we retain roughly 300,000 objects. Each of the objects is rendered with Blender from 16 viewpoints encircling the object using a fixed pinhole camera model and a fixed (camera colocated) environment map[2] as in Fig. 4. For the evaluations in Sec. 5, we leave out 2% randomly selected elements.

Training Collaborative Control For most of the experiments in Sec. 5, ZeroDiffusion [36, 63] is the base RGB model, a zero-terminal-SNR version fine-tuned from StableDiffusion v1.5 [48]. As Collaborative Control is agnostic to the base model, we also illustrate StableDiffusion v1.5 and v2.1 as base models in Sec. 5. We optimize the PBR model's weights as well as the cross-network communication layers to minimize the training loss for the RGB and PBR denoising jointly, while keeping the RGB model fully frozen. Unless otherwise stated, we directly train on 512×512 resolution for a total of 200,000 update steps with a batch size of 12 and a learning rate of 3×10^{-5} (on one 80 GB VRAM A100, taking roughly two days). We evaluate the effect of a larger training budget by training on 8 A100's for the same number of steps, increasing the batch size by a factor of 8 without affecting training time — for environmental and cost purposes, the training budget is kept low for the main ablation study.

[2] `https://polyhaven.com/a/studio_small_08`

5 Results

Distribution match metrics As an evaluation of how well the data distribution is modeled, distribution match is considered a proxy to both quality and diversity. The Inception Score (IS [49]), which checks the distribution match against ImageNet, is not relevant in a PBR context as it applies only to RGB images. The Fréchet Inception Distance (FID [17]), which compares the distributions of the last hidden state of the Inceptionv3 [60] network on both real and generated images, has been found to better align to perceptual quality. Finally, the recently introduced CLIP Maximum-Mean Discrepancy (CMMD [23]) compares the distribution of the CLIP embeddings of generated images to that of a reference dataset. It offers significantly improved sample efficiency, and was shown by the authors to be a better indicator of low-level image quality than FID. However, as these metrics are intended for three-channel color images, we evaluate them on PBR images following Chambon et al. [5], by averaging the relevant scores of multiple triplets. We report as PBR distribution match the average of the scores over each of the PBR channels independently, as well as over three additional triplets, as the full set of triplets is prohibitively expensive to compute (the supplementary contains all the constituting scores). The additional triplets are (grayscale albedo, roughness, metallic), (roughness, metallic, normal XY norm) and (grayscale albedo, normal X, normal Y) for a balance between the full cartesian product (which is costly) and mixing channels that are normally relatively independent.

Out-of-distribution (OOD) performance metrics indicate the level to which our generator can align to conditioning that it was not trained on. Recent work has introduced the CLIP alignment score [13, 16], which estimates the average distance between the text prompt CLIP embedding and the generated image's CLIP embedding, indicating how faithfully the prompt was followed. Additionally, OneAlign [69] is a neural model that estimates aesthetics and quality scores for images, shown to align well with human opinions, summarized in a QAlign score for both aesthetics and general quality. In order to evaluate the OOD performance, we randomly select 50 objects from Objaverse, and generate unlikely appearance prompts for them using ChatGPT4 [1]. These results, as well as a t-SNE plot of the embeddings of the original and OOD prompts, are integrally shown in the supplementary material.

5.1 Comparisons and Ablations

To the best of our knowledge, there are no published PBR generation models that generate PBR images for entire objects or scenes (only for generation of single materials [51, 65]). Therefore, we perform an extensive ablation study on our design choices, taking care to include typical approaches from techniques that generate other modalities than PBR. Please refer to Fig. 6 for the qualitative comparisons, while Tab. 1 contains quantitative results.

Comparison between control paradigms We compare the performance of the proposed bidirectional cross-network communication layer against two other paradigms: one inspired by ControlNet-XS [10], and one inspired by AnimateAnyone [20]. In the former, dubbed *one-way* communication, the communication layers receive as input only the RGB model's internal state, and they only affect the PBR model's internal state. The latter, dubbed *clockwise* communication, functions in the same way for the encoder part of the architecture, but reverses the information flow to go from the PBR model to the RGB model for the decoder half of the architecture. We see that the *one-way* attention does not perform well, with lower distribution match scores as well as OOD performance scores; the frozen RGB model cannot realign to the conditional distribution required from it in Eq. (1), and we see that the positions it generates for the objects does not align at all with the mask from the normal image. The *clockwise* attention performs significantly better, but is likely still hampered by $\boldsymbol{z}'_{rgb,t-1}$ not being easily available to the PBR model — a similar reasoning as to why the authors of ControlNet-XS included the direct communication link between the base and controlling models' encoders.

Comparison between communication types In terms of the type of communication, we compare the proposed single-layer per-pixel communication against a per-pixel MLP-based communication layer, and a global attention layer. The latter performs surprisingly well considering that it lacks pixel correspondences; it is hard to enforce pixel-wise alignment through a global attention layer, which we hypothesize to be the reason for the lower quantitative performance. As Jin et al. [24] discuss, an attention-based architecture is also less robust to resolution changes. The per-pixel MLP, containing four hidden per-pixel linear layers with normalization layers [2] in-between, does not qualitatively perform notably better than the single-layer communication layer, so that we settle for the simpler and more computationally efficient choice.

Comparison against fine-tuning We also compare Collaborative Control against the alternative where we edit the first and last layers of the pre-trained network to match the dimensionality of the PBR images (optionally with the rendered image), and then fine-tune the entire network end-to-end. Although the distribution match scores for these fine-tuning variants are similar to Collaborative Control, the fine-tuning methods strongly overfit to the training data and perform very poorly in a qualitative OOD comparison.

PBR-specific VAE vs RGB VAE We compare the performance of Collaborative Control with a PBR-specific VAE against a version that uses the triplets-based RGB VAE mentioned in Sec. 4 to encode the PBR channels (encoding albedo, roughness+metallic, and bump maps in separate triplets and concatenating their latent representations). The mismatch within the PBR domain is clear, both quantitatively through the worse distribution matching scores, and qualitatively in the produced images.

Fig. 6: Generated albedo, roughness/metallic and bump map images from the ablation studies. While significant quality differences are visible, only the fine-tuning approach and the data-sparse regime *with* PBR prompt cross-attention fail completely. The version that was trained on a smaller resolution does not break but does not result in maximum detail either. Best viewed digitally.

Table 1: Quantative results for all evaluated variants. The ablation baseline is highlighted in bold, duplicated for easier comparisons within the individual ablations.

		2% held-out evaluation data								OOD					
		CMMD ↓		FID ↓		QAlign ↑ Ae		QAlign ↑ Q		QAlign ↑ Ae		QAlign ↑ Q		CLIPScore ↑	
		PBR	Relit	PBR	Relit	Albedo	Relit	Albedo	Relit	Albedo	Relit	Albedo	Relit	Albedo	Relit
Communication	one-way	16.44	13.38	20.90	16.39	1.95	1.97	2.37	2.48	1.91	1.59	2.35	1.69	23.08	23.40
	clockwise	6.78	2.76	12.21	11.53	2.04	2.02	2.63	2.64	**2.14**	1.70	2.77	**1.74**	26.45	24.53
	bi-directional	**6.30**	**1.79**	**11.65**	**10.64**	**2.11**	2.03	**2.75**	**2.66**	2.12	1.76	**2.78**	1.73	**26.76**	**25.41**
	Pixel-wise zero-conv	6.30	**1.79**	11.65	**10.64**	**2.11**	**2.03**	**2.75**	**2.66**	2.12	1.76	2.78	1.73	26.76	25.41
	Pixel-wise MLP	**5.43**	1.87	**11.43**	10.67	2.10	2.02	2.74	**2.66**	**2.26**	1.75	**2.96**	**1.81**	**27.15**	**25.95**
	Global Attention	7.60	5.22	13.61	11.93	1.94	1.98	2.51	2.60	1.99	1.72	2.71	1.80	24.50	24.01
	Collaborative Control	6.30	**1.79**	11.65	**10.64**	**2.11**	2.03	**2.75**	**2.66**	2.12	1.76	2.78	1.73	**26.76**	**25.41**
	Fine-tuning (with RGB output)	13.40	2.78	14.42	10.79	2.05	2.02	2.60	2.61	2.10	1.76	2.62	**1.86**	25.04	22.69
	Fine-tuning (without RGB output)	**5.25**	2.88	**11.41**	11.37	2.03	1.99	2.58	2.58	**2.26**	1.71	**2.97**	1.81	25.66	23.31
	PBR VAE	**6.30**	**1.79**	**11.65**	**10.64**	2.11	**2.03**	**2.75**	**2.66**	2.12	1.76	2.78	1.73	**26.76**	**25.41**
	RGB VAE on triplets	84.66	5.99	25.81	11.63	**2.16**	1.99	2.67	2.55	**2.30**	1.71	**2.95**	**1.80**	25.27	23.98
Training budget	**1 A100, two days**	6.30	1.79	11.65	10.64	**2.11**	2.03	**2.75**	2.66	**2.12**	1.76	2.78	1.73	26.76	**25.41**
	8 A100s, two days	**2.96**	**1.12**	**9.55**	**9.76**	2.08	2.04	2.68	2.67	2.01	1.73	**2.82**	**1.82**	**26.78**	25.22
Training resolution	256×256	**2.23**	**1.44**	**9.82**	**10.20**	2.10	2.04	2.73	2.68	2.20	1.78	**3.13**	**1.80**	26.71	25.21
	512×512	6.30	1.79	11.65	10.64	**2.11**	2.03	**2.75**	2.66	**2.12**	1.76	2.78	1.73	**26.76**	**25.41**
Training data	**No PBR prompt attention** 1%	6.25	**1.43**	11.87	10.79	2.18	2.04	**2.86**	2.69	**2.35**	1.76	**3.35**	1.89	26.58	25.11
	No PBR prompt attention 5%	**5.77**	1.45	**11.49**	**10.54**	2.13	2.04	2.78	2.69	2.09	1.73	3.01	1.84	**27.28**	25.04
	No PBR prompt attention 20%	5.97	1.68	11.50	10.61	2.12	2.03	2.78	2.67	2.23	1.76	3.23	1.88	25.72	24.99
	No PBR prompt attention **98%**	6.30	1.79	11.65	10.64	2.11	2.03	2.75	2.66	2.12	1.76	2.78	1.73	26.76	25.41
	PBR prompt attention 1%	20.61	4.25	18.35	12.16	**2.19**	2.03	2.76	2.62	2.25	1.61	2.83	1.80	24.75	23.25
	PBR prompt attention 5%	12.17	2.58	14.95	10.97	2.13	2.04	2.71	2.65	2.27	1.80	2.97	**1.98**	26.89	**25.53**
	PBR prompt attention 20%	11.35	2.33	14.78	10.78	2.13	2.03	2.74	2.65	2.29	1.76	3.07	1.77	26.79	25.39
	PBR prompt attention 98%	9.18	2.57	13.25	11.02	2.10	2.03	2.68	2.64	2.17	1.80	2.84	1.87	25.80	24.83

Impact of the training budget Comparing the version training on a single A100 with the version trained with 8 A100s (for eight times the batch size), we see that the latter performs significantly better quantitatively in terms of distribution match, but not quality. Visually, the differences are less clear, although the higher-budget model appears to follow complex prompts slightly better.

Impact of the training resolution We compare the performance of Collaborative Control with two training resolutions: 256×256 and 512×512, both evaluated on 512×512 (ZeroDiffusion's native resolution). While the low-resolution model quantitatively performs better, visually it is clear that it does not capture the same level of detail as the high-resolution model — we explain this through the metrics focusing on high-level encoding of the images, and the lower resolution enables smoother training through a larger batch size (42).

Impact of training dataset size Now, we evaluate the performance of Collaborative Control under data sparsity by evaluating models trained on 98%, 20%, 5% and 1% of the full 6M training images. The proposed approach proves very data-efficient and performs well even when trained on only a few thousand images (1%). We observe that it is necessary to disable prompt cross-attention in the PBR model, and that this effect gets more pronounced with fewer data: we hypothesize that the model overfits to the training data and that forcing prompt attention to occur through the frozen RGB base model prevents this overfitting.

Fig. 7: A re-lit generated texture in Peppermint Powerplant and Pine Attic and lightfield slice under environment rotation.

Fig. 8: Interpolation on text embeddings and initial noise shows the stability of the proposed approach in both of these spaces.

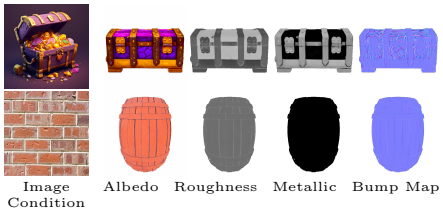


Fig. 9: Our PBR diffusion remains compatible with control techniques trained for the base frozen RGB model. We illustrate this using StableDiffusion 1.5 as the base model using a public IP-Adapter [73].

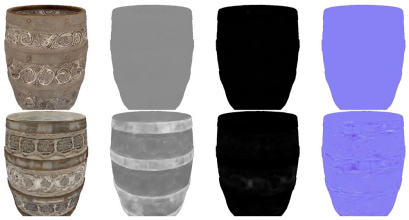


Fig. 10: Our most common failure case is the constant roughness, metallic or bump maps. Prompting a *porcelain barrel with intricate designs* for two different random seeds illustrates this behaviour.

Compatibility with other control techniques As a closing experiment, we illustrate that Collaborative Control is compatible with other control techniques [10, 44, 73], which drastically expands the practical applications of our proposed method. We demonstrate this specifically with IP-Adapter [73], which allows us to condition the final output on a style image by introducing additional style cross-attention layers within the base model. We can apply an IP-Adapter to the base model and still generate PBR content, as illustrated in Fig. 9.

Relighting For a small qualitative indication that the PBR materials we generate also look natural under different environment lighting, Fig. 7 shows a generated texture when relit in two novel environment maps. Furthermore, a slice of the lightfield under rotation of the second environment map shows that the highlights and shadows move smoothly (hinting that the bump map is meaningful).

Interpolation To illustrate the stability of our proposed approach in terms of both initial noise and the text prompt, we (separately) interpolate between two prompts and between two initial noises in Fig. 8. We perform prompt interpolation in CLIP space, and for noise interpolation we rescale the final image to have unit standard deviation and zero mean after blending. The resulting images appear natural yet meaningfully interpolate between both extremes.

5.2 Limitations and failure cases

We identify two major failure cases: lack of detail in the roughness, metallic, and bump maps, and a failure to follow OOD prompts. In the former, we see (*e.g.* Fig. 10) that the model outputs a constant (though varying per instance) roughness and metallic value, and a flat bump map. We attribute this to the training data: Objaverse contains many objects with constant roughness and metallic properties and a flat bump map — likely biasing the model towards such outputs. Anecdotally, we have found that selecting a different random seed will often succeed where the first generation disappointed — practically, the model produces very diverse results even for the same prompt and the same conditioning geometry, so that we argue that this is either not a significant issue or can be resolved with better training data.

A failure to follow out-of-distribution prompts happens mostly when structural features in the prompt are incompatible with the conditioning geometry, such as for example *a gilded lion* for a table mesh. We hypothesize that the control signal from the PBR model conflicts with the text cross-attention in the frozen RGB model, resulting in lackluster outputs. Different random seeds occasionally resolve this issue, albeit more rarely.

Finally, we note that the model does come at the cost of executing two parallel diffusion models. We note that the motivation for this was mainly to retain a frozen copy of the base RGB model: in applications where these requirements are prohibitively expensive, distilling our approach into a direct PBR model is likely to bring relief.

6 Conclusion

In this work, we have proposed Collaborative Control, a new paradigm for leveraging a pre-trained image-based RGB diffusion model for generating high-quality PBR image content conditioned on object geometry. We have shown that this bi-directional control paradigm is extremely data-efficient while retaining the high quality and expressiveness of the base RGB model, even when faced with text queries completely out of distribution for the PBR training data. The plug-and-play nature of our proposed approach is compatible with existing adaptations of the base RGB model, which we have illustrated with IP-Adapter for style guidance of the PBR content. The availability of high-quality PBR content generation as offered by our proposed approach opens up new avenues for graphics applications, specifically in Text-to-Texture.

Acknowledgements

This work was supported fully by Unity Technologies, without external funding. We would like to thank the reviewers for their valuable feedback and suggestions.

References

1. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)
2. Ba, J.L., Kiros, J.R., Hinton, G.E.: Layer normalization. stat **1050**, 21 (2016)
3. Burley, B.: Physically based shading at disney. In: ACM Transactions on Graphics (SIGGRAPH) (2012)
4. Cao, T., Kreis, K., Fidler, S., Sharp, N., Yin, K.: Texfusion: Synthesizing 3d textures with text-guided image diffusion models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4169–4181 (2023)
5. Chambon, T., Heitz, E., Belcour, L.: Passing multi-channel material textures to a 3-channel loss. In: ACM SIGGRAPH 2021 Talks, pp. 1–2 (2021)
6. Chen, D.Z., Siddiqui, Y., Lee, H.Y., Tulyakov, S., Nießner, M.: Text2tex: Text-driven texture synthesis via diffusion models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 18558–18568 (2023)
7. Chen, R., Chen, Y., Jiao, N., Jia, K.: Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 22246–22256 (2023)
8. Cook, R.L., Torrance, K.E.: A reflectance model for computer graphics. ACM Transactions on Graphics (ToG) (1982)
9. Deitke, M., Schwenk, D., Salvador, J., Weihs, L., Michel, O., VanderBilt, E., Schmidt, L., Ehsani, K., Kembhavi, A., Farhadi, A.: Objaverse: A universe of annotated 3d objects. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13142–13153 (2023)
10. Denis Zavadski, J.F.F., Rother, C.: Controlnet-xs: Designing an efficient and effective architecture for controlling text-to-image diffusion models (2023)
11. Du, X., Kolkin, N., Shakhnarovich, G., Bhattad, A.: Intrinsic lora: A generalist approach for discovering knowledge in generative models. In: Synthetic Data for Computer Vision Workshop, CVPR 2024
12. Duan, Y., Guo, X., Zhu, Z.: Diffusiondepth: Diffusion denoising approach for monocular depth estimation. arXiv preprint arXiv:2303.05021 (2023)
13. Foong, T.Y., Kotyan, S., Mao, P.Y., Vargas, D.V.: The challenges of image generation models in generating multi-component images. arXiv preprint arXiv:2311.13620 (2023)
14. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial networks. Communications of the ACM **63**(11), 139–144 (2020)
15. Guo, P., Hao, H., Caccavale, A., Ren, Z., Zhang, E., Shan, Q., Sankar, A., Schwing, A.G., Colburn, A., Ma, F.: Stabledreamer: Taming noisy score distillation sampling for text-to-3d. arXiv preprint arXiv:2312.02189 (2023)
16. Hessel, J., Holtzman, A., Forbes, M., Bras, R.L., Choi, Y.: Clipscore: A reference-free evaluation metric for image captioning. arXiv preprint arXiv:2104.08718 (2021)
17. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. Advances in neural information processing systems **30** (2017)
18. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020)
19. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: Lora: Low-rank adaptation of large language models. arXiv preprint arXiv:2106.09685 (2021)

20. Hu, L.: Animate anyone: Consistent and controllable image-to-video synthesis for character animation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8153–8163 (2024)
21. Huang, T., Zeng, Y., Zhang, Z., Xu, W., Xu, H., Xu, S., Lau, R.W., Zuo, W.: Dreamcontrol: Control-based text-to-3d generation with 3d self-prior. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5364–5373 (2024)
22. Huang, Y., Wang, J., Shi, Y., Qi, X., Zha, Z.J., Zhang, L.: Dreamtime: An improved optimization strategy for text-to-3d content creation. arXiv preprint arXiv:2306.12422 (2023)
23. Jayasumana, S., Ramalingam, S., Veit, A., Glasner, D., Chakrabarti, A., Kumar, S.: Rethinking fid: Towards a better evaluation metric for image generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9307–9315 (2024)
24. Jin, Z., Shen, X., Li, B., Xue, X.: Training-free diffusion model adaptation for variable-sized text-to-image synthesis. Advances in Neural Information Processing Systems **36** (2024)
25. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of gans for improved quality, stability, and variation. arXiv preprint arXiv:1710.10196 (2017)
26. Karras, T., Aittala, M., Laine, S., Härkönen, E., Hellsten, J., Lehtinen, J., Aila, T.: Alias-free generative adversarial networks. Advances in Neural Information Processing Systems **34**, 852–863 (2021)
27. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4401–4410 (2019)
28. Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., Aila, T.: Analyzing and improving the image quality of stylegan. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8110–8119 (2020)
29. Ke, B., Obukhov, A., Huang, S., Metzger, N., Daudt, R.C., Schindler, K.: Repurposing diffusion-based image generators for monocular depth estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9492–9502 (2024)
30. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
31. Knodt, J., Gao, X.: Consistent mesh diffusion. arXiv preprint arXiv:2312.00971 (2023)
32. Le, C., Hetang, C., Cao, A., He, Y.: Euclidreamer: Fast and high-quality texturing for 3d models with stable diffusion depth. arXiv preprint arXiv:2311.15573 (2023)
33. Lee, H.Y., Tseng, H.Y., Yang, M.H.: Exploiting diffusion prior for generalizable dense prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7861–7871 (2024)
34. Li, X., Zhang, Q., Kang, D., Cheng, W., Gao, Y., Zhang, J., Liang, Z., Liao, J., Cao, Y.P., Shan, Y.: Advances in 3d generation: A survey. arXiv preprint arXiv:2401.17807 (2024)
35. Liang, Y., Yang, X., Lin, J., Li, H., Xu, X., Chen, Y.: Luciddreamer: Towards high-fidelity text-to-3d generation via interval score matching. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6517–6526 (2024)
36. Lin, S., Liu, B., Li, J., Yang, X.: Common diffusion noise schedules and sample steps are flawed. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 5404–5411 (2024)

37. Liu, F., Wu, D., Wei, Y., Rao, Y., Duan, Y.: Sherpa3d: Boosting high-fidelity text-to-3d generation via coarse 3d prior. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20763–20774 (2024)
38. Liu, R., Wu, R., Van Hoorick, B., Tokmakov, P., Zakharov, S., Vondrick, C.: Zero-1-to-3: Zero-shot one image to 3d object. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9298–9309 (2023)
39. Liu, Y.T., Guo, Y.C., Luo, G., Sun, H., Yin, W., Zhang, S.H.: Pi3d: Efficient text-to-3d generation with pseudo-image diffusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19915–19924 (2024)
40. Liu, Z., Li, Y., Lin, Y., Yu, X., Peng, S., Cao, Y.P., Qi, X., Huang, X., Liang, D., Ouyang, W.: Unidream: Unifying diffusion priors for relightable text-to-3d generation. arXiv preprint arXiv:2312.08754 (2023)
41. Long, X., Guo, Y.C., Lin, C., Liu, Y., Dou, Z., Liu, L., Ma, Y., Zhang, S.H., Habermann, M., Theobalt, C., et al.: Wonder3d: Single image to 3d using cross-domain diffusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9970–9980 (2024)
42. Ma, B., Deng, H., Zhou, J., Liu, Y.S., Huang, T., Wang, X.: Geodream: Disentangling 2d and geometric priors for high-fidelity and consistent 3d generation. arXiv preprint arXiv:2311.17971 (2023)
43. Ma, Y., Fan, Y., Ji, J., Wang, H., Sun, X., Jiang, G., Shu, A., Ji, R.: X-dreamer: Creating high-quality 3d content by bridging the domain gap between text-to-2d and text-to-3d generation. arXiv preprint arXiv:2312.00085 (2023)
44. Mou, C., Wang, X., Xie, L., Wu, Y., Zhang, J., Qi, Z., Shan, Y.: 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. vol. 38, pp. 4296–4304 (2024)
45. Van den Oord, A., Kalchbrenner, N., Espeholt, L., Vinyals, O., Graves, A., et al.: Conditional image generation with pixelcnn decoders. Advances in neural information processing systems **29** (2016)
46. Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: Dreamfusion: Text-to-3d using 2d diffusion. In: The Eleventh International Conference on Learning Representations (2023)
47. Raj, A., Kaza, S., Poole, B., Niemeyer, M., Ruiz, N., Mildenhall, B., Zada, S., Aberman, K., Rubinstein, M., Barron, J., et al.: Dreambooth3d: Subject-driven text-to-3d generation. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 2349–2359 (2023)
48. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10684–10695 (2022)
49. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. Advances in neural information processing systems **29** (2016)
50. Sarkar, A., Mai, H., Mahapatra, A., Lazebnik, S., Forsyth, D.A., Bhattad, A.: Shadows don't lie and lines can't bend! generative models don't know projective geometry... for now. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 28140–28149 (2024)
51. Sartor, S., Peers, P.: Matfusion: A generative diffusion model for svbrdf capture. In: SIGGRAPH Asia 2023 Conference Papers. SA '23, ACM (Dec 2023). https://doi.org/10.1145/3610548.3618194, http://dx.doi.org/10.1145/3610548.3618194

52. Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., et al.: Laion-5b: An open large-scale dataset for training next generation image-text models. Advances in Neural Information Processing Systems **35**, 25278–25294 (2022)
53. Sharma, P., Jampani, V., Li, Y., Jia, X., Lagun, D., Durand, F., Freeman, B., Matthews, M.: Alchemist: Parametric control of material properties with diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 24130–24141 (2024)
54. Shi, R., Chen, H., Zhang, Z., Liu, M., Xu, C., Wei, X., Chen, L., Zeng, C., Su, H.: Zero123++: a single image to consistent multi-view diffusion base model. arXiv preprint arXiv:2310.15110 (2023)
55. Shi, Y., Wang, P., Ye, J., Mai, L., Li, K., Yang, X.: Mvdream: Multi-view diffusion for 3d generation. In: The Twelfth International Conference on Learning Representations (2024)
56. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics. In: International conference on machine learning. pp. 2256–2265. PMLR (2015)
57. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. In: International Conference on Learning Representations (2020)
58. Stan, G.B.M., Wofk, D., Fox, S., Redden, A., Saxton, W., Yu, J., Aflalo, E., Tseng, S.Y., Nonato, F., Muller, M., et al.: Ldm3d: Latent diffusion model for 3d. arXiv preprint arXiv:2305.10853 (2023)
59. Subias, J.D., Lagunas, M.: In-the-wild material appearance editing using perceptual attributes. In: Computer Graphics Forum. vol. 42, pp. 333–345. Wiley Online Library (2023)
60. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2818–2826 (2016)
61. Tang, B., Wang, J., Wu, Z., Zhang, L.: Stable score distillation for high-quality 3d generation. arXiv preprint arXiv:2312.09305 (2023)
62. Tang, J., Ren, J., Zhou, H., Liu, Z., Zeng, G.: Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. In: The Twelfth International Conference on Learning Representations (2024)
63. `https://huggingface.co/drhead`: Huggingface zerodiffusion model weights v0.9. `https://huggingface.co/drhead/ZeroDiffusion`, accessed: 2024-02-08
64. Van Den Oord, A., Kalchbrenner, N., Kavukcuoglu, K.: Pixel recurrent neural networks. In: International conference on machine learning. pp. 1747–1756. PMLR (2016)
65. Vecchio, G., Martin, R., Roullier, A., Kaiser, A., Rouffet, R., Deschaintre, V., Boubekeur, T.: Controlmat: A controlled generative approach to material capture. arXiv preprint arXiv:2309.01700 (2023)
66. Wang, P., Fan, Z., Xu, D., Wang, D., Mohan, S., Iandola, F., Ranjan, R., Li, Y., Liu, Q., Wang, Z., et al.: Steindreamer: Variance reduction for text-to-3d score distillation via stein identity. arXiv preprint arXiv:2401.00604 (2023)
67. Wang, Z., Li, M., Chen, C.: Luciddreaming: Controllable object-centric 3d generation. arXiv preprint arXiv:2312.00588 (2023)
68. Wang, Z., Lu, C., Wang, Y., Bao, F., Li, C., Su, H., Zhu, J.: Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. Advances in Neural Information Processing Systems **36** (2024)

69. Wu, H., Zhang, Z., Zhang, W., Chen, C., Liao, L., Li, C., Gao, Y., Wang, A., Zhang, E., Sun, W., et al.: Q-align: Teaching lmms for visual scoring via discrete text-defined levels. In: Forty-first International Conference on Machine Learning (2024)
70. Wu, T., Li, Z., Yang, S., Zhang, P., Pan, X., Wang, J., Lin, D., Liu, Z.: Hyperdreamer: Hyper-realistic 3d content generation and editing from a single image. In: SIGGRAPH Asia 2023 Conference Papers. pp. 1–10 (2023)
71. Wu, Z., Zhou, P., Yi, X., Yuan, X., Zhang, H.: Consistent3d: Towards consistent high-fidelity text-to-3d generation with deterministic sampling prior. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9892–9902 (2024)
72. Xu, X., Lyu, Z., Pan, X., Dai, B.: Matlaber: Material-aware text-to-3d via latent brdf auto-encoder. arXiv preprint arXiv:2308.09278 (2023)
73. Ye, H., Zhang, J., Liu, S., Han, X., Yang, W.: Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. arXiv preprint arXiv:2308.06721 (2023)
74. Yeh, Y.Y., Huang, J.B., Kim, C., Xiao, L., Nguyen-Phuoc, T., Khan, N., Zhang, C., Chandraker, M., Marshall, C.S., Dong, Z., et al.: Texturedreamer: Image-guided texture synthesis through geometry-aware diffusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4304–4314 (2024)
75. Youwang, K., Oh, T.H., Pons-Moll, G.: Paint-it: Text-to-texture synthesis via deep convolutional texture map optimization and physically-based rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4347–4356 (2024)
76. Yu, K., Liu, J., Feng, M., Cui, M., Xie, X.: Boosting3d: High-fidelity image-to-3d by boosting 2d diffusion prior to 3d prior with progressive learning. arXiv preprint arXiv:2311.13617 (2023)
77. Zeng, X., Chen, X., Qi, Z., Liu, W., Zhao, Z., Wang, Z., Fu, B., Liu, Y., Yu, G.: Paint3d: Paint anything 3d with lighting-less texture diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4252–4262 (2024)
78. Zhang, J., Tang, Z., Pang, Y., Cheng, X., Jin, P., Wei, Y., Yu, W., Ning, M., Yuan, L.: Repaint123: Fast and high-quality one image to 3d generation with progressive controllable 2d repainting. arXiv preprint arXiv:2312.13271 (2023)
79. Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3836–3847 (2023)
80. Zhou, L., Shih, A., Meng, C., Ermon, S.: Dreampropeller: Supercharge text-to-3d generation with parallel sampling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4610–4619 (2024)
81. Zhuang, J., Wang, C., Lin, L., Liu, L., Li, G.: Dreameditor: Text-driven 3d scene editing with neural fields. In: SIGGRAPH Asia 2023 Conference Papers. pp. 1–10 (2023)