

# Scene-Conditional 3D Object Stylization and Composition

Jinghao Zhou<sup>✉</sup>, Tomas Jakab<sup>✉</sup>, Philip Torr<sup>✉</sup>, and Christian Rupprecht<sup>✉</sup>

University of Oxford  
{jinghao,tomj,chrisr}@robots.ox.ac.uk



**Fig. 1:** We present a framework that adapts a 3D object’s appearance to a location in a 2D scene. It creates an image where the 3D object is seamlessly blended (left & bottom right), with its appearance influenced by the scene’s environmental conditions and lighting effects. Moreover, the stylized object with adapted textures (top right), rendered here without the estimated lighting condition for illustrative purposes, can be further utilized as 3D assets for downstream tasks such as video games.

**Abstract.** Recently, 3D generative models have made impressive progress, enabling the generation of almost arbitrary 3D assets from text or image inputs. However, these approaches generate objects in isolation without any consideration for the scene where they will eventually be placed. In this paper, we propose a framework that allows for the stylization of an existing 3D asset to fit into a given 2D scene, and additionally produce a photorealistic composition as if the asset was placed within the environment. This not only opens up a new level of control for object stylization, for example, the same assets can be stylized to reflect changes in the environment, such as summer to winter or fantasy versus futuristic settings—but also makes the object-scene composition more controllable. We achieve this by combining modeling and optimizing the object’s texture and environmental lighting through differentiable ray

tracing with image priors from pre-trained text-to-image diffusion models. We demonstrate that our method applies to a wide variety of indoor and outdoor scenes and arbitrary objects. See also our project page.

## 1 Introduction

After the success of generative models [20, 43, 46] for images in computer vision, the community is now interested in lifting these models to 3D [40, 54] through advances in architectures, data, and training. These lifted 3D models demonstrate plausible 3D representations of objects conditioned on images or texts.

While generating whole objects—*e.g.* *an astronaut riding a horse*—is a challenging problem that generative models are becoming able to solve, in practice, these assets are difficult to adapt for use in downstream applications or existing virtual environments such as video games as they are generated without context.

This paper aims to provide a mechanism to close the appearance gap between 3D objects and environments. We formulate this problem through the lens of a creative tool. The intended goal of this paper is to align the appearance of a 3D object to a customized 2D scene as we place it into a specific location, a challenging task that requires fine-grained visual control for convincing results.

We are interested in producing a photorealistic scene-object composition where the 3D object is seamlessly blended. For instance, an object placed in a rainy night scene should appear wet and dimly lit, and if the scene is muddy and sunny, the object should also look muddy and cast a strong shadow. To achieve a high level of realism on a fine-grained level, the mud on the object must match the scene’s mud in appearance and the direction of the shadow must match where the sun is located. Instead of simply resorting to 2D blending operations [18, 50], we explicitly model an object’s appearance and lighting in 3D space, which enables more accurate and regularized control of the object’s appearance in the composed image.

We also aim to preserve the underlying object geometry and appearance factors unrelated to the scene environment. This task is highly practical; for example, for media design, when presenting a 3D asset of a product within a realistic scene, we want the product to blend naturally while remaining recognizable. Manually achieving this would require a skilled artist to modify the object’s texture to match the scene, a challenging and time-consuming process. Our paper introduces a method that automates this task. It accepts a 3D object with texture, a 2D scene, and the object’s position within the scene and outputs an adapted texture consistent with the scene.

To accomplish this, we leverage recent advances in generative diffusion models [46, 47] for images, which are trained on a large-scale dataset of internet images. We conduct an optimization procedure where we compose the 2D renderings of the 3D object with the 2D scene at a specified location and use a pre-trained diffusion model to “critique” the realism of the composed image, providing a gradient signal for optimization.

The challenge is that diffusion models, such as Stable Diffusion [46], are trained on 2D images and lack any notion of 3D geometry. Naively optimizing

or denoising in 2D space [18, 50] would result in losing the object’s 3D geometry. To prevent this, we optimize the object’s content in 3D by randomly rotating the object during optimization and rendering it from various viewpoints, which helps to align the adapted texture with the object’s geometry and avoid overfitting to a single viewpoint.

Notably, randomly rotating the 3D object stops lighting effects from being baked into the texture, which is desirable. Yet, it also prevents the object from naturally blending with the scene in a single viewpoint. To counteract this, we predict the scene’s lighting and the shadows cast by the object as separate components. For this purpose, we propose a novel technique inspired by a common physical approach from the computer graphics industry, where reflective and diffuse spheres with known material parameters are placed and photographed in a real-world scene to capture the environmental lighting conditions. We adopt this concept and place a virtual white diffuse sphere inside the scene during optimization while conditioning the generative model to expect a white sphere within the scene, thereby capturing the model’s interpretation of the scene’s lighting in an easily extracted form.

Optimizing the object’s appearance in 3D offers further significant advantages: The user can change the object’s pose without re-running the optimization, as the texture is meaningful from all viewpoints, and view-dependent effects are modeled separately by the lighting component. Additionally, the 3D object with the adapted texture can be used as a standalone asset in other downstream tasks (*e.g.* in a video game).

In summary, we present a framework that (1) enables the stylization of a 3D object, which can also be used as a standalone asset, given a 2D scene and its location within it; (2) achieves photorealistic scene-object compositing with the help of a novel mechanism to estimate scene lighting from a single image.

We further verify the effectiveness of our method in extensive experiments, demonstrating that our approach can realistically adapt objects to a diverse set of environments, laying the groundwork for practical applications.

## 2 Related Work

The purpose of this paper is to adapt the appearance of a 3D object to align with a customized 2D scene. This solves a more complicated problem than prior works, which often only tackle individual components, including but not limited to 2D scene-object compositing, light estimation, and texturing, but never as a whole. We compare with these methods next and detail their differences in Tab. 1.

**Scene-Object Compositing.** Scene-object compositing is a challenging task that requires a delicate balance between visual transfer and control. It has been extensively researched [5, 8, 9, 25]. Recent methods condition the denoising process of diffusion models [20, 49] on object images. OST [50] aims to seamlessly integrate a 2D object image into a 2D scene by modifying the keys and values in the attention blocks. CDC [18] considers composing the 2D renderings of a 3D object with a 2D scene by directly modifying denoised latent.

**Table 1: Comparison with related work.** *Rep.* denotes representation being 2D ( $\square$ ) or 3D ( $\textcircled{\text{C}}$ ). *Env. Inf.* denotes *Environmental Influence* and *I.D. Pre.* denotes *Identity Preservation*. Parameters for the lighting and the appearance model are either scene-agnostic constant ( $\textcircled{\text{C}}$ ), scene-conditioned predictable ( $\textcircled{\text{C}}$ ), or learnable ( $\textcircled{\text{C}}$ ).

| Method      | Scene       |                                  |              | Object                   |                                  |                  |                  | Compose      |
|-------------|-------------|----------------------------------|--------------|--------------------------|----------------------------------|------------------|------------------|--------------|
|             | <i>Rep.</i> | Lighting                         | Shadow       | <i>Rep.</i>              | Appearance                       | <i>Env. Inf.</i> | <i>I.D. Pre.</i> |              |
| OST [50]    | $\square$   | -                                | $\times$     | $\square$                | -                                | $\times$         | $\times$         | Blend        |
| CDC [18]    | $\square$   | Ambient $\textcircled{\text{C}}$ | $\times$     | $\textcircled{\text{C}}$ | Diffuse $\textcircled{\text{C}}$ | $\times$         | $\times$         | Blend        |
| IGAN [28]   | $\square$   | LDR $\textcircled{\text{C}}$     | $\checkmark$ | $\textcircled{\text{C}}$ | Diffuse $\textcircled{\text{C}}$ | $\times$         | $\times$         | Copy & Paste |
| EVL [27]    | $\square$   | HDR $\textcircled{\text{C}}$     | $\checkmark$ | $\textcircled{\text{C}}$ | Diffuse $\textcircled{\text{C}}$ | $\times$         | $\times$         | Copy & Paste |
| PrDr [55]   | -           | Ambient $\textcircled{\text{C}}$ | $\times$     | $\textcircled{\text{C}}$ | Diffuse $\textcircled{\text{C}}$ | $\checkmark$     | $\times$         | -            |
| Fan3D [7]   | -           | HDR $\textcircled{\text{C}}$     | $\times$     | $\textcircled{\text{C}}$ | PBR $\textcircled{\text{C}}$     | $\checkmark$     | $\times$         | -            |
| Inst3D [26] | -           | Point $\textcircled{\text{C}}$   | $\times$     | $\textcircled{\text{C}}$ | Diffuse $\textcircled{\text{C}}$ | $\times$         | $\checkmark$     | -            |
| <b>Ours</b> | $\square$   | HDR $\textcircled{\text{C}}$     | $\checkmark$ | $\textcircled{\text{C}}$ | PBR $\textcircled{\text{C}}$     | $\checkmark$     | $\checkmark$     | Copy & Paste |

However, these methods rely entirely on the 2D denoising process and do not explicitly model the object’s geometry, which often results in a significant loss of structural details and visual identity. This severely hampers these methods’ ability to adapt to the environmental and lighting conditions of the scene.

**Light Estimation from Single-View Image.** Estimating light conditions from a single-view 2D image is of practical use in the real world and has been extensively studied for both indoor [15, 16] and outdoor scenes [22, 23, 58]. The distinction between these two settings has been significantly narrowed recently [27, 28], thanks to the outpainting capabilities of diffusion models, which have contributed to a unified framework. With additional training on large high dynamic range (HDR) datasets such as SUN360 [57], these methods can approximate the location and intensity of the light source, enabling realistic shadows.

However, these methods estimate only the lighting of the scene. They are typically indifferent to how objects should be stylized to match the scene, neglecting the potential environmental influences on the textures of objects.

**Mesh Texturing.** Classic automated approaches [56] for mesh texturing are often limited and apply only simple texture patterns. In contrast, methods [14, 37, 48] based on Generative Adversarial Networks [17] (GANs) are more capable of synthesizing detailed textures. Recently, text-conditioned texture generation [34, 35] has been made possible through the optimization of CLIP-based objectives [42]. The latest works now employ powerful text-to-image diffusion models [20, 49], either through optimization [7, 29, 33, 51, 55] using Score Distillation Sampling [40] (SDS), or by wrapping the generated 2D textures onto the mesh [4, 6, 45]. Importantly, Variational Score Distillation [55] (VSD) as an improved optimization objective and the adoption of PBR model [7] have greatly boosted the quality of generated textures. Additionally, an increasing level of requirement for more visual control from users has been partially met [19, 26] through image-conditional diffusion models [3] to preserve objects’ identities.

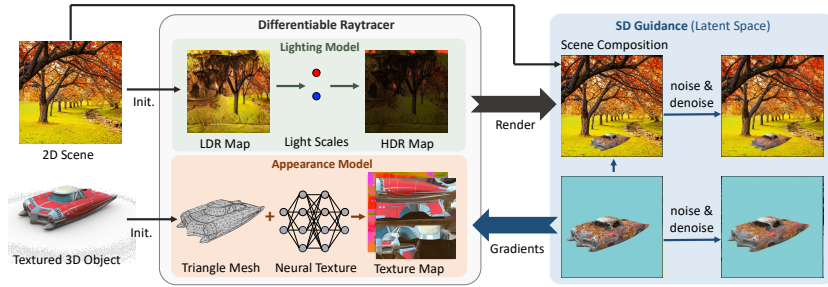


**Fig. 2: Framework.** We learn an environment map and a texture map *separately* from the 2D supervision. We initialize (*init.*) the environment map with an LDR map estimated from the 2D scene and learn light multiplying scales bright areas, yielding an HDR map. We employ the PBR material model for texture maps, encoded via MLP with positional encoding. The object is rendered through a differentiable ray tracer and further composed with the scene background, receiving gradients from Stable Diffusion (SD) in the latent space.

However, these works focus on texturing the object in isolation from the scene and do not consider the homogeneity of the composition when these assets are placed in certain environments, which limits their practical usage.

### 3 Method

The overall structure of our framework can be found in Fig. 2. Given a textured 3D object, a 2D image of a scene, and a 2D location where to place the object, we first adapt the texture of the 3D object (Secs. 3.2 and 3.3) to align with the scene. To enable a photorealistic composition of the object and the scene, we further estimate the lighting (Secs. 3.4 to 3.6) and apply it during the final rendering.

#### 3.1 Optimization

To enable end-to-end optimization, we adopt a differentiable renderer to render the object and score it using the priors from diffusion models.

**Differentiable Renderer.** We employ the differential renderer Mitsuba3 [24] (denoted as  $g$ ), due to its capabilities as a physically-based ray-tracing renderer equipped with versatile rendering options, such as diverse BSDFs, the path tracer integrator, and using an environment map as a light source. This enables us to produce photo-realistic renders complete with shadows and reflections, forming the foundation for robust texture adaptation and light estimation.

**Diffusion Model as Guidance.** To score the rendered image, we make use of a generative diffusion model [46, 47]  $\phi$ . Diffusion models have learned powerful priors of the image formation process in the real world from large-scale training

on billion-scale web data. Similar to prior work [40,54], we supervise the rendered image  $\mathbf{x}_0 = g(\boldsymbol{\theta})$  with a score-matching objective to obtain an image gradient

$$\nabla_{\boldsymbol{\theta}} \mathcal{L}(\mathbf{x}_0) \triangleq \mathbb{E}_{t,\epsilon} \left[ w_t (\epsilon_{\phi}(\mathbf{x}_t) - \hat{\epsilon}(\mathbf{x}_t)) \frac{\partial \mathbf{x}_0}{\partial \boldsymbol{\theta}} \right], \quad (1)$$

where  $\mathbf{x}_t = \sqrt{\alpha_t} \mathbf{x}_0 + \sqrt{1 - \alpha_t} \epsilon$  is the noisy rendered image,  $\epsilon_{\phi}(\mathbf{x}_t)$  is an approximated score for a noisy real image,  $t \in \mathbb{N}$  indexes the discrete diffusion steps, and  $\hat{\epsilon}$  is a scoring function for  $\mathbf{x}_t$ . Specifically,  $\hat{\epsilon}$  is set as constant  $\epsilon$  in Score Distillation Sampling [40] (SDS), and  $\epsilon_{\psi}$  is an online LoRA-tuned network initialized from  $\epsilon_{\phi}$  in Variational Score Sampling [55] (VSD). Notably, we employ their interpolated variant given by  $\hat{\epsilon} = \lambda \epsilon_{\psi} + (1 - \lambda) \epsilon$  with a  $\lambda$  annealed from 1 to  $\lambda_e$ , helping to counteract the over-texturization often observed in VSD, especially for objects with simple surfaces. With residual of noises  $\epsilon_{\phi} - \hat{\epsilon} = (\epsilon_{\phi} - \epsilon) - \lambda(\epsilon_{\psi} - \epsilon)$ , we highlight its relation with the image gradient obtained in traditional inverse rendering setup  $\nabla_{\boldsymbol{\theta}} \mathcal{L}(\mathbf{x}_0) = \mathbf{x}_0 - \mathbf{x}_{\text{gt}}$  by reformulating it as

$$\epsilon_{\phi} - \hat{\epsilon} = \sqrt{\frac{\alpha_t}{1 - \alpha_t}} [(\mathbf{x}_0 - \hat{\mathbf{x}}_{\phi}) - \lambda(\mathbf{x}_0 - \hat{\mathbf{x}}_{\psi})], \quad (2)$$

where the target used to supervise  $\mathbf{x}_0$  is predicted clean image  $\hat{\mathbf{x}}_{\phi}$  from  $\epsilon_{\phi}$  as the positive and  $\hat{\mathbf{x}}_{\psi}$  from  $\epsilon_{\psi}$  as the negative. The technique enables nominal classifier-free guidance [21] and thus bypasses mode-seeking behaviors and saturated texture. Inspired by [45], we incorporate the inner product of the camera pose and object’s surface normal  $\langle \mathbf{c}, \mathbf{n} \rangle$  into  $w_t$  to penalize faces at small grazing angles. In practice, we use Stable Diffusion [46] (SD) and absorb the encoder into  $g$ , optimizing in latent space.

### 3.2 Appearance Model

**Textured Mesh.** We represent the 3D object as a textured mesh, which is the most common format for 3D assets in the industry and is compatible with many established workflows. Textured meshes are available in abundance in online datasets and marketplaces [1, 10], and recently, one can also generate reasonable quality 3D meshes with text descriptions [7, 29, 55] or limited viewpoints [30, 41].

**Neural Texture.** We represent the appearance of objects using a texture map  $\mathcal{T} \in \mathbb{R}^{H_t \times W_t \times 5}$ , facilitating compatibility of textured results with widely-used rendering engines, such as Blender. We parameterize the texture using a coordinate neural network that maps 3D coordinates to the texture attributes. We first map the 3D coordinates of vertices using a rasterizer [36] based on their UV coordinates into the space of the texture map, resulting in  $\mathcal{C} \in \mathbb{R}^{H_t \times W_t \times 3}$ . We then use a 2-layer parameterized MLP without bias and Hash positional encoding  $\Phi: \mathcal{C} \rightarrow \mathcal{T}$  to map the coordinates to their texture attributes. For the final layer, we re-map the output activated after sigmoid according to pre-defined minimal and maximal values. The five channels in  $\mathcal{T}$  correspond to the Physically-Based

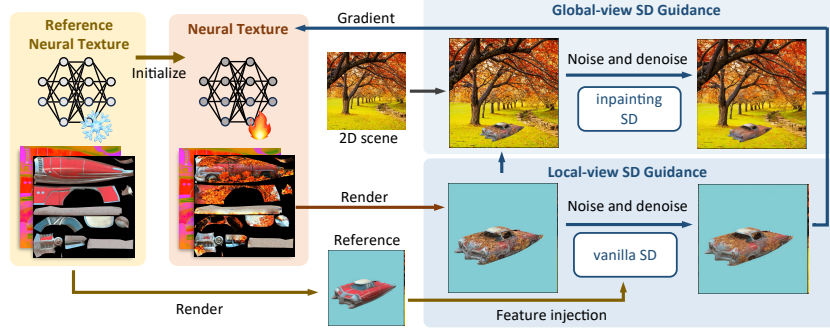


**Fig. 3: Pipeline for texture adaptation.** We initialize the neural texture from the reference object and inject the feature of reference renderings to the U-Net of SD. We use both *local-view* and *global-view* guidance.

Rendering (PBR) shading model [32], which includes a diffuse term  $k_d \in \mathbb{R}^3$  and an isotropic specular GGX lobe [53] described by  $k_{rm} = (k_r, k_m) \in \mathbb{R}^2$ , with  $k_r$  representing roughness and  $k_m$  indicating metalness. The PBR model, with  $k_r$  affecting the scattering of light and  $k_m$  influencing the reflective properties, creates realistic light interactions between surfaces.

### 3.3 Scene-Conditional Guidance

Achieving a realistic texture adaptation for 3D objects placed into 2D scenes requires three components: *Environmental Influence* involves adjusting the texture to reflect the environmental impact realistically; *Identity Preservation* aims to preserve the objects’ unique visual aspects, and *Blending* focuses on guiding the texture adaptation to match the visual characteristic of the scene, ensuring seamless integration with the surroundings and avoiding stark contrasts.

***Environmental Influence.*** To show the realistic effects of the scene environment on the object’s appearance, we guide the diffusion model with text prompts. We find that a naive text prompt, combining a simple object description with a scene description, *e.g.*, *a leather sofa in a swamp*, is insufficient to generate realistic textures that reflect the environmental impact on the object, as shown in the appendix. For instance, we would expect a sofa placed in a swamp to be dirty and possibly show signs of moss or other vegetation growth. Instead of manually crafting a detailed text prompt for each object and scene combination, we automatize the process by employing a large language model (LLM), specifically GPT-4 [38], to generate a text prompt for the image diffusion model. Given the simple description of the object (*e.g.*, *a leather sofa*) and the scene (*e.g.*, *a swamp*), we instruct the LLM first to analyze the potential impact of the scene’s environment on the object’s appearance and then to distil this information into a text prompt that describes the resulting appearance of the object (*e.g.*, *The leather sofa, partially submerged in the swamp, looks discolored and soggy, its once-polished surface marred by mud, moss, and the murky water of the swamp*



**Fig. 4: Pipeline for estimating the LDR map.** We utilize tailored pipelines for indoor and outdoor scenes. Areas masked red and blue correspond to the far light  and near light  region.

*environment*). We then use this automatically generated text prompt to guide the diffusion model in adapting the texture. The specific prompt design for this automated process is detailed in the appendix.

**Identity Preservation.** To prevent the transferred texture from drifting away from its original content, we regularize the denoising diffusion model with reference images, a technique that has been explored for controlled image generation [39, 52]. Suppose we have a rendered reference image  $\mathbf{x}_0^r = g(\boldsymbol{\theta}^r)$  and its noised version  $\mathbf{x}_t^r$  as input forwarded to  $\epsilon_\phi$  and  $\epsilon_\psi$ , for the Multi-Head Self-Attention (MHSA) layer in each Transformer block of  $\epsilon_\phi$ 's and  $\epsilon_\psi$ 's U-Nets, we concatenate the key  $K^r$  and value  $V^r$  of  $\mathbf{x}_t^r$  to the original  $K$  and  $V$  from  $\mathbf{x}_t$  yielding an image-conditional attention output

$$A(\mathbf{x}_t, \cdot, \mathbf{x}_t^r) = \text{MHSA}(Q, [K; K^r], [V; V^r]), \quad (3)$$

where  $\cdot$  can either be text or  $\emptyset$  for text conditioning. The text-unconditional attention  $A(\mathbf{x}_t, \emptyset, \mathbf{x}_t^r)$  used for classifier-free guidance [21] is further modified as  $s_c \cdot A(\mathbf{x}_t, \emptyset, \emptyset) + (1 - s_c) \cdot A(\mathbf{x}_t, \emptyset, \mathbf{x}_t^r)$  weighted by a control guidance scale  $s_c$ . With the percentage  $p$  of blocks to be injected, the intensity of preservation can be controlled:  $s_c = p = 1$  provide the most and  $s_c = p = 0$  the least preservation.

**Blending.** The two components previously mentioned directly utilize object renderings, referred to as the *local view*, as inputs for vanilla SD. To further enhance the adaptation of the object's appearance to its surroundings, we input the scene image, with the object rendered onto it, referred to as the *global view*, into an additional inpainting SD [46]. We have found that the inpainting model provides superior conditioning on the surrounding environment depicted in the image, resulting in improved blending. However, the inpainting model can sometimes attempt to undesirably modify the inserted object, removing parts of it by inpainting them with the background. To counter this, we compose the *local view* with randomly sample solid background colors, rather than the scene. This prevents the inpainting model from removing parts of the object as the object must look realistic when rendered in front of a random background. In practice, we utilize a set of *global views* created by cropping the original *global view* at various scales, ensuring that the crops encompass the entire *local view*.

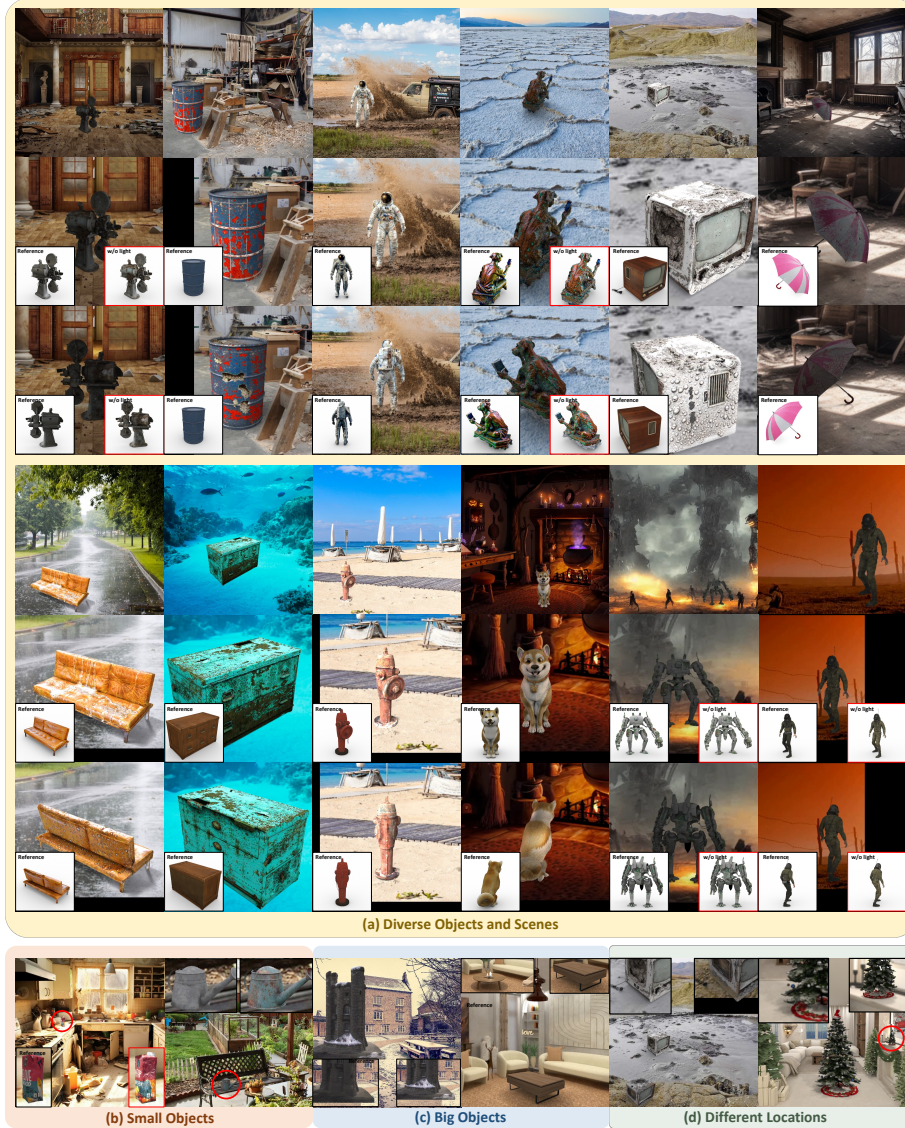


**Fig. 5: Visual Results.** (a) We showcase that our method applies to a diverse range of objects and scenes. The *global view* (top row) the overall composition quality and object-centric *local view* (bottom two rows) for the fidelity of stylized textures are demonstrated. For dim scenes, we additionally render objects without the estimated lighting condition (*w/o light*) for illustrative purposes. Additionally, we showcase that our method applies to both (b) small and (c) big objects, as well as (d) different placing locations. The texture of the *television*, for example, adjusts the texture to match its surroundings.

### 3.4 Lighting Model

To seamlessly integrate the object into the scene, we estimate the lighting conditions, as outlined in Fig. 4. To light the object, we utilize a high dynamic range (HDR) environment map  $\mathcal{E} \in \mathbb{R}^{H_e \times W_e \times 3}$ , which is well-suited for representing natural illumination. Since the given 2D scene image only captures a small angle of the full 360-degree environment map, we first use the image as input to estimate a low dynamic range (LDR) environment map. This process depends on the scene type (*i.e.*, indoor *vs.* outdoor) and is explained in the following paragraphs. Given the LDR map, we then convert it to an HDR map by estimating a light scale (*i.e.*, a scalar) for each bright region, which is thresholded from the LDR map directly (*e.g.*, where intensity  $\mathcal{I}_{i,j} \geq 0.8$ ). This scalar is then multiplied by the RGB values to approximate an HDR map. Note that we use the HDR map only to light the object. It thus does not need to be of high accuracy to make the object look plausible.

**Indoor Scenes.** The well-bounded 2D visual geometry of indoor scenes is primarily characterized by enclosed spaces and interconnected planes, facilitating straightforward (rough) 3D reconstruction. To simulate the physical lighting of indoor scenes, we lift each pixel from the 2D scene to 3D space using its estimated depth  $\mathcal{D}$  [13, 44]. We perform world-to-camera and spherical-to-equirectangular coordinate transformations in succession, unwrapping 3D points into a latitude-longitude formatted LDR map. In post-processing, we automatically fill holes and remove small isolated regions. We fill the unseen region (*i.e.* areas behind the camera) with the average RGB value of the 2D scene. Two scalars are optimized: one for the far light region ( $\textcircled{R} \triangleq \{i, j \mid \mathcal{D}_{i,j} \geq \tau_d \wedge \mathcal{I}_{i,j} \geq \tau_f\}$ ) and another for the near light region ( $\textcircled{V} \triangleq \{i, j \mid \mathcal{D}_{i,j} < \tau_d \wedge \mathcal{I}_{i,j} \geq \tau_n\}$ ).

**Outdoor Scenes.** 3D reconstruction of outdoor scenes is challenging due to their inherently unbounded geometric structure. Furthermore, in most outdoor scenes, the sun is the dominant light source, and the position change of an object has a negligible effect on the angle shift of the light. Therefore, instead of painstakingly recovering location-specific viewpoints, we use a single-view to panorama outpainting model [2], taking its generated output as the initial LDR map. We optimize one scalar for the far light area with positive elevation ( $\textcircled{R} \triangleq \{i, j \mid \frac{i}{H_e} < 0.5 \vee \mathcal{I}_{i,j} \geq \tau_o\}$ ), leaving the near light area ( $\textcircled{V} \triangleq \emptyset$ ) unoptimized. While more sophisticated designs, such as a position estimator [27] or sky modeling [22], could be incorporated, we find our approach robust enough to provide a sufficient estimation.

### 3.5 Light-Capturing Apparatus

To estimate light scales, we place 3D objects with a fixed appearance model into the 2D scene and then optimize the lighting with VSD as described in Eq. (1) to anchor the lighting impact imposed by the scene. Inspired by the traditional inverse rendering setup [11] where the environment map can almost be perfectly



**Fig. 6: Qualitative Comparison.** We compare our the generated *global view* with the scene-object compositing method: CDC [18] and the *local view* with mesh texturing methods: Prolific Dreamer [55], Fantasia3D [7], and TEXTure [45]. *s.gd.* denotes similarity guidance. *g.gd.* and *r.f.i.* denotes global-view guidance and reference feature injection, respectively. Our method achieves both seamless scene-object composition as well as texture adaptation of high fidelity. See sup. mat. for extensive comparison.

reconstructed from objects’ reflections, we introduce a novel concept incorporating a virtual light-capturing apparatus alongside the object of interest during the optimization process. We insert a white sphere made of a smooth diffuse material into the scene and guide the lighting optimization with the diffusion model using the text prompt “A gigantic diffuse white (spray-painted) sphere (ball)”. The white diffuse sphere proves advantageous in stabilizing lighting estimations for scenes with potentially strong light sources, as the reflected intensity from a white diffuse object closely approximates the intensity of the environmental lighting. In our initial experiments, we also tested spheres with mirrored and matte silver materials but found them less beneficial.

### 3.6 Light-Conditional Guidance

The absence of light conditions in text prompts makes the generated images only loosely constrained on lighting, which makes it challenging for diffusion models to score the renderings accurately. A white sphere looks gray in a darker environment, so light-agnostic prompts can engender an overestimation of the brightness for dim scenes. Therefore, we append “in a dark environment” if the average intensity of the background and light areas are below certain thresholds. We condition the LoRA-tuned model  $\epsilon_\psi$  by concatenating the light scales with the camera extrinsic [55] as class embeddings for the U-Net. This improves light estimation, especially for outdoor scenes and dimly lit scenes. For scenes with atypical lighting that significantly alters the appearance of objects, we manually append color prompts such as “in blue tint” for the seabed bathed in blue light, or “in red illumination” for a nightclub enveloped in red atmospheric light.

## 4 Experiments

In this section, we show extensive qualitative and quantitative results of our method in various settings. Please see the appendix for many more examples, animations and discussion on limitations and negative impact.



**Fig. 7: Ablation on Texture Adaptation.** We ablate global-view guidance (*g.gd.*) and reference feature injection (*r.f.i.*). *-ctrl.* and *+ctrl.* denotes less or more control in feature injection (smaller or higher  $s_f$  &  $p$ ) for *Identity Preservation* requirement.

#### 4.1 Qualitative Evaluation

We first present visual results and a case study to demonstrate the generalizability of the proposed method. We then compare our method with others that specifically focus on either scene composition or object stylization.

**Main Results.** To demonstrate the generalizability of the framework and the impact of different scenes on the inserted objects, we present visual results in Fig. 5. Our method successfully blends the objects into various environments, achieving photorealistic adaptation for both appearance and lighting. This includes scenarios drawn from both real-world and fantasy settings. For example, the dimly-lit umbrella in Fig. 5 casts a shadow that is perfectly aligned with its handle and the light direction of the scene, substantially increasing the photorealism of the composition; The Belweder TV set is wrapped with gray mud consistent with the scene while its original structural details get maintained.

**Comparison on Scene Composition.** We showcase the result from CDC [18] using LLM-prompted text prompts for a fair comparison. As shown in Fig. 6, CDC loses the structural details of the object while providing limited adaptation for texture transfer. Moreover, the composed scene appears artificial and synthetic, giving the impression that the sofa is floating above the mud. Our experiments reveal that the CDC often leads to object removal, particularly when the re-painting [31] is employed. Therefore, we further incorporate similarity guidance similar to the classifier guidance [12] by adding  $\sqrt{1-\alpha_t} \cdot \frac{\partial(\|x_0^r - \hat{x}_\phi\|^2)}{\partial x_t}$  to the predicted gradient  $\epsilon_\phi(x_t)$  each denoising step referred as **CDC w/ *s.gd.***, but find it provides limited improvement.

**Comparison on Object Stylization.** We showcase results from mesh texturing method [7, 45, 55] using our LLM-prompted text prompts for a fair comparison. As shown in Fig. 6, although achieving realistic texture adaptation, these mesh texturing methods fall short in preserving the original structural details of the object and in utilizing scene-level information. Essentially, These methods only satisfy the *Environmental Influence* requirement via text prompts and do not meet all the criteria of our intended goal.

**Table 2: Quantitative Evaluation.** (a) We measure LPIPS for controllability, CLIP<sub>dir</sub> and CLIP<sub>SI</sub> for composability. (b) We present the percentages of users who prefer our method over other methods: CDC [18] and InstructP2P [3].

| Method                             | LPIPS ↓       | CLIP <sub>dir</sub> ↑ | CLIP <sub>SI</sub> ↑ |                          |            |      |       |
|------------------------------------|---------------|-----------------------|----------------------|--------------------------|------------|------|-------|
| InstructP2P [3]                    | 0.0020        | 0.0175                | 0.1852               |                          | Ours       | Draw | Other |
| Ours ( <i>T.A.</i> )               | <b>0.0014</b> | 0.0201                | 0.2006               | Ours vs. CDC [18]        | <b>71%</b> | 20%  | 9%    |
| CDC [18]                           | 0.0053        | 0.0042                | 0.1996               | Ours vs. InstructP2P [3] | <b>74%</b> | 6%   | 20%   |
| Ours ( <i>T.A.</i> + <i>L.E.</i> ) | 0.0015        | <b>0.0206</b>         | <b>0.2013</b>        |                          |            |      |       |

(b) User study for human preference.

(a) Visual Metrics.

(b) User study for human preference.

## 4.2 Quantitative Evaluation

We collected 34 composites between diverse objects and scenes, each with 3 random views, yielding a total of 102 images. Besides CDC [18], we compare with InstructPix2Pix [3] by directly copy-pasting the object with adapted texture onto the scene. We measure LPIPS for controllability, CLIP<sub>dir</sub> and CLIP<sub>SI</sub> for composability, similar to CDC, and report the results in Tab. 2. Compared with InstructPix2Pix our method for texture adaptation (*T.A.*) alone excels in controllability and composability. Light estimation (*L.E.*) further improves results.

We also conducted a *user study* in Tab. 2 with 42 participants to compare the realism of our compositing under the condition that the object identity must be preserved. Our method is preferred over prior work in over 70% of the cases.

## 4.3 Ablation Study

**Identity Preservation.** To illustrate how the *Identity Preservation* requirement is satisfied by reference feature injection (*r.f.i.*), we showcase the leather sofa in the swamp with less and more control (Fig. 7). The setup with less control drifts to a visually different sofa with too much mud wrapped around while more control generates a less muddy and adapted sofa compared to the reference image. Both adapted textures are less visually plausible. Additionally, when comparing **Ours w/o *g.gd.* & *r.f.i.*** to the reference image, it is evident that the identity of the sofa is lost.

**Global-View Guidance.** To showcase the necessity of global-view guidance we conduct texture adaptation without it as **Ours w/o *g.gd.*** in Fig. 7. The adapted texture without seeing the full scene fails to adapt the color of the mud to that of the scene image, contrasting starkly with the scene. Experiments ablating the optimal global-view guidance set-up (*e.g.*, solid-color background augmentation, usage of inpainting model, *etc.*) are detailed in the appendix.

**Light Estimation.** In Fig. 1, the combination of accurately estimated lighting and shadows significantly enhances the photorealism of the composed image (compare the image on the left, which includes lighting, to those in the top right corner, which are without lighting). To further test the validity of the estimated light, we render images omitting the texture adaptation step. Fig. 8 demonstrates that the estimated light considers multiple factors, including its

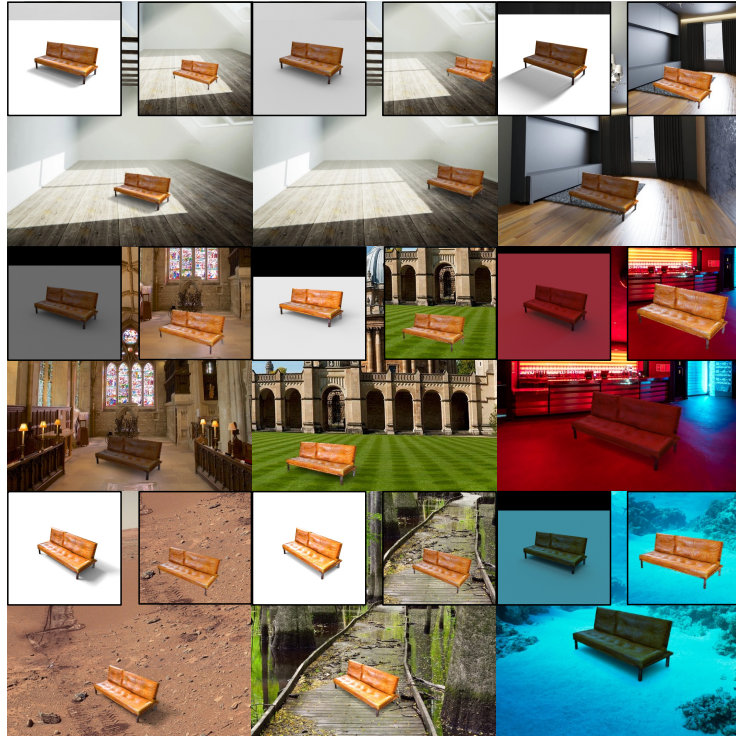


**Fig. 8: Ablation on Light Estimation.** We evaluate the effectiveness of light estimation by demonstrating the lighting effects on a sofa in various indoor and outdoor scenes. Overlaid in the left corners are objects rendered with the estimated light against a white background. In the right corners are objects before the light estimation.

direction relative to the specific locations of objects in indoor scenes and its impact on the appearance of objects under atmospheric colored lighting conditions. Importantly, the estimated HDR maps, paired with the proposed scene composition, facilitate the realistic integration of shadows.

## 5 Conclusion

In this paper, we are motivated by a practical goal—*what if a 3D object is placed into a 2D scene?*—and propose a novel framework that allows (1) stylizing the object with an adapted texture that aligns with the given scene; and (2) achieving a photorealistic scene composition with the aid of estimated light of the environment. This is enabled by physically-based differentiable ray tracing and diffusion models used as guidance. Our method allows for automatic adaptation and blending of existing objects into a variety of scenes, making it useful for both 2D and 3D downstream applications, such as visual media and video games.

**Acknowledgement.** This work is supported by the UKRI grant: Turing AI Fellowship EP/W002981/1 and EPSRC/MURI grant: EP/N019474/1. J. Zhou is also supported by the Horizon Robotics. We thank Luke Melas-Kyriazi and Fabio Pizzati for their helpful discussions.

## References

1. <https://polyhaven.com>, <https://polyhaven.com>
2. Akimoto, N., Matsuo, Y., Aoki, Y.: Diverse plausible 360-degree image outpainting for efficient 3d background creation. In: CVPR (2022)
3. Brooks, T., Holynski, A., Efros, A.A.: Instructpix2pix: Learning to follow image editing instructions. In: CVPR (2023)
4. Cao, T., Kreis, K., Fidler, S., Sharp, N., Yin, K.: Textfusion: Synthesizing 3d textures with text-guided image diffusion models. In: ICCV (2023)
5. Chen, B.C., Kae, A.: Toward realistic image compositing with adversarial learning. In: CVPR (2019)
6. Chen, D.Z., Siddiqui, Y., Lee, H.Y., Tulyakov, S., Nießner, M.: Text2tex: Text-driven texture synthesis via diffusion models. arXiv preprint arXiv:2303.11396 (2023)
7. Chen, R., Chen, Y., Jiao, N., Jia, K.: Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. In: ICCV (2023)
8. Cong, W., Niu, L., Zhang, J., Liang, J., Zhang, L.: Bargainnet: Background-guided domain translation for image harmonization. In: ICME (2021)
9. Cong, W., Zhang, J., Niu, L., Liu, L., Ling, Z., Li, W., Zhang, L.: Dovenet: Deep image harmonization via domain verification. In: CVPR (2020)
10. 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: CVPR (2023)
11. Deng, Y., Li, X., Liu, S., Yang, M.H.: Dip: Differentiable interreflection-aware physics-based inverse rendering. arXiv preprint arXiv:2212.04705 (2022)
12. Dhariwal, P., Nichol, A.: Diffusion models beat gans on image synthesis. NeurIPS (2021)
13. Eftekhari, A., Sax, A., Malik, J., Zamir, A.: Omnidata: A scalable pipeline for making multi-task mid-level vision datasets from 3d scans. In: ICCV (2021)
14. Gao, J., Shen, T., Wang, Z., Chen, W., Yin, K., Li, D., Litany, O., Gojcic, Z., Fidler, S.: Get3d: A generative model of high quality 3d textured shapes learned from images. NeurIPS (2022)
15. Gardner, M.A., Hold-Geoffroy, Y., Sunkavalli, K., Gagné, C., Lalonde, J.F.: Deep parametric indoor lighting estimation. In: ICCV (2019)
16. Gardner, M.A., Sunkavalli, K., Yumer, E., Shen, X., Gambaretto, E., Gagné, C., Lalonde, J.F.: Learning to predict indoor illumination from a single image. arXiv preprint arXiv:1704.00090 (2017)
17. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. NeurIPS (2014)
18. Hachnochi, R., Zhao, M., Orzech, N., Gal, R., Mahdavi-Amiri, A., Cohen-Or, D., Bermano, A.H.: Cross-domain compositing with pretrained diffusion models. arXiv preprint arXiv:2302.10167 (2023)
19. Haque, A., Tancik, M., Efros, A., Holynski, A., Kanazawa, A.: Instruct-nerf2nerf: Editing 3d scenes with instructions. In: CVPR (2023)

20. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. *NeurIPS* (2020)
21. Ho, J., Salimans, T.: Classifier-free diffusion guidance. *NeurIPS* (2021)
22. Hold-Geoffroy, Y., Athawale, A., Lalonde, J.F.: Deep sky modeling for single image outdoor lighting estimation. In: *CVPR* (2019)
23. Hold-Geoffroy, Y., Sunkavalli, K., Hadap, S., Gambaretto, E., Lalonde, J.F.: Deep outdoor illumination estimation. In: *CVPR* (2017)
24. Jakob, W., Speierer, S., Roussel, N., Vicini, D.: Dr.jit: A just-in-time compiler for differentiable rendering. *TOG* (2022)
25. Jiang, Y., Zhang, H., Zhang, J., Wang, Y., Lin, Z., Sunkavalli, K., Chen, S., Amirghodsi, S., Kong, S., Wang, Z.: Ssh: A self-supervised framework for image harmonization. In: *ICCV* (2021)
26. Kamata, H., Sakuma, Y., Hayakawa, A., Ishii, M., Narihira, T.: Instruct 3d-to-3d: Text instruction guided 3d-to-3d conversion. *arXiv preprint arXiv:2303.15780* (2023)
27. Karimi Dastjerdi, M.R., Eisenmann, J., Hold-Geoffroy, Y., Lalonde, J.F.: Everlight: Indoor-outdoor editable hdr lighting estimation. In: *ICCV* (2023)
28. Karimi Dastjerdi, M.R., Hold-Geoffroy, Y., Eisenmann, J., Khodadadeh, S., Lalonde, J.F.: Guided co-modulated GAN for 360 field of view extrapolation. *3DV* (2022)
29. Lin, C.H., Gao, J., Tang, L., Takikawa, T., Zeng, X., Huang, X., Kreis, K., Fidler, S., Liu, M.Y., Lin, T.Y.: Magic3d: High-resolution text-to-3d content creation. In: *CVPR* (2023)
30. Liu, M., Xu, C., Jin, H., Chen, L., Xu, Z., Su, H., et al.: One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization. *arXiv preprint arXiv:2306.16928* (2023)
31. Lugmayr, A., Danelljan, M., Romero, A., Yu, F., Timofte, R., Van Gool, L.: Repaint: Inpainting using denoising diffusion probabilistic models. In: *CVPR* (2022)
32. McAuley, S., Hill, S., Hoffman, N., Gotanda, Y., Smits, B., Burley, B., Martinez, A.: Practical physically-based shading in film and game production. In: *SIGGRAPH* (2012)
33. Metzger, G., Richardson, E., Patashnik, O., Giryas, R., Cohen-Or, D.: Latent-nerf for shape-guided generation of 3d shapes and textures. In: *CVPR* (2023)
34. Michel, O., Bar-On, R., Liu, R., Benaim, S., Hanocka, R.: Text2mesh: Text-driven neural stylization for meshes (2022)
35. Mohammad Khalid, N., Xie, T., Belilovsky, E., Popa, T.: Clip-mesh: Generating textured meshes from text using pretrained image-text models. In: *SIGGRAPH Asia* (2022)
36. Munkberg, J., Hasselgren, J., Shen, T., Gao, J., Chen, W., Evans, A., Müller, T., Fidler, S.: Extracting triangular 3d models, materials, and lighting from images. In: *CVPR* (2022)
37. Oechsle, M., Mescheder, L., Niemeyer, M., Strauss, T., Geiger, A.: Texture fields: Learning texture representations in function space. In: *ICCV* (2019)
38. OpenAI: Gpt-4 technical report (2023)
39. Parmar, G., Kumar Singh, K., Zhang, R., Li, Y., Lu, J., Zhu, J.Y.: Zero-shot image-to-image translation. In: *SIGGRAPH* (2023)
40. Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: Dreamfusion: Text-to-3d using 2d diffusion. *ICLR* (2023)

41. Qian, G., Mai, J., Hamdi, A., Ren, J., Siarohin, A., Li, B., Lee, H.Y., Skorokhodov, I., Wonka, P., Tulyakov, S., Ghanem, B.: Magic123: One image to high-quality 3d object generation using both 2d and 3d diffusion priors. arXiv preprint arXiv:2306.17843 (2023)
42. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: ICML (2021)
43. Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., Sutskever, I.: Zero-shot text-to-image generation. In: ICML (2021)
44. Ranftl, R., Lasinger, K., Hafner, D., Schindler, K., Koltun, V.: Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. PAMI (2022)
45. Richardson, E., Metzger, G., Alaluf, Y., Giryas, R., Cohen-Or, D.: Texture: Text-guided texturing of 3d shapes. SIGGRAPH (2023)
46. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models (2021)
47. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E.L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al.: Photorealistic text-to-image diffusion models with deep language understanding. NeurIPS (2022)
48. Siddiqui, Y., Thies, J., Ma, F., Shan, Q., Nießner, M., Dai, A.: Texturify: Generating textures on 3d shape surfaces. In: ECCV (2022)
49. Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S., Poole, B.: Score-based generative modeling through stochastic differential equations. In: ICLR (2021), <https://openreview.net/forum?id=PxtIG12RRHS>
50. Song, Y., Zhang, Z., Lin, Z., Cohen, S., Price, B., Zhang, J., Kim, S.Y., Aliaga, D.: Objectstitch: Generative object compositing. CVPR (2023)
51. Tsalicoglou, C., Manhardt, F., Tonioni, A., Niemeyer, M., Tombari, F.: Textmesh: Generation of realistic 3d meshes from text prompts (2024)
52. Tumanyan, N., Geyer, M., Bagon, S., Dekel, T.: Plug-and-play diffusion features for text-driven image-to-image translation. In: CVPR (2023)
53. Walter, B., Marschner, S.R., Li, H., Torrance, K.E.: Microfacet models for refraction through rough surfaces. In: EGSR (2007)
54. Wang, H., Du, X., Li, J., Yeh, R.A., Shakhnarovich, G.: Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. In: CVPR (2023)
55. 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. NeurIPS (2023)
56. Wei, L.Y., Lefebvre, S., Kwatra, V., Turk, G.: State of the art in example-based texture synthesis. EG-STAR (2009)
57. Xiao, J., Ehinger, K.A., Oliva, A., Torralba, A.: Recognizing scene viewpoint using panoramic place representation. In: ICCV (2012)
58. Zhang, J., Sunkavalli, K., Hold-Geoffroy, Y., Hadap, S., Eisenman, J., Lalonde, J.F.: All-weather deep outdoor lighting estimation. In: CVPR (2019)