

REPARO: Compositional 3D Assets Generation with Differentiable 3D Layout Alignment

Haonan Han^{1*} Rui Yang^{2*} Huan Liao^{1*} Jiankai Xing¹ Zunnan Xu¹
Xiaoming Yu³ Junwei Zha³ Xiu Li^{1†} Wanhua Li^{4†}

¹ Tsinghua University ² The University of Hong Kong ³ Tencent ⁴ Harvard University

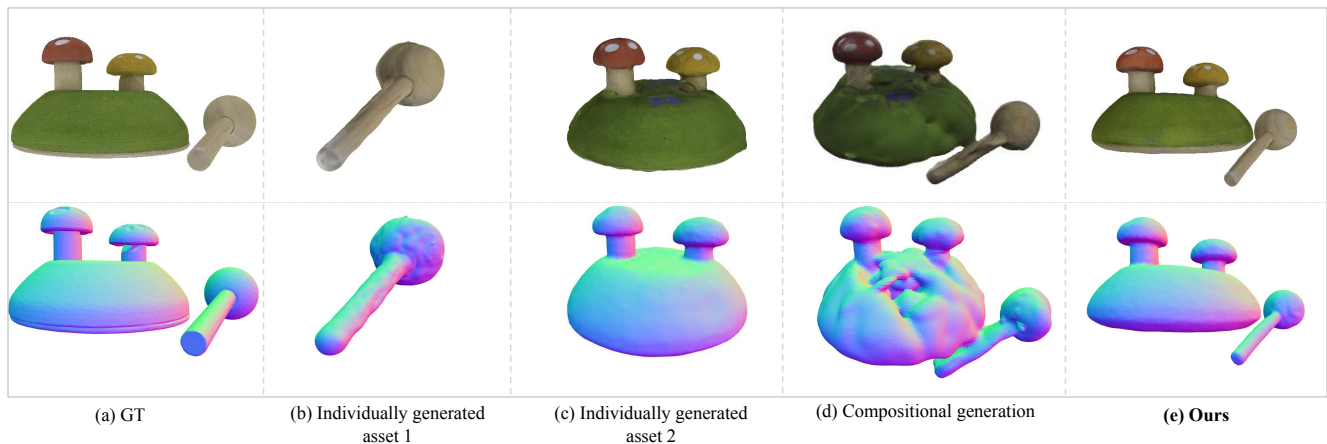


Figure 1. Qualitative comparison of generated 3D assets of the single object and multiple objects. (b) and (c) are generated 3D assets of the single object assets using DreamGaussian [51]; (d) is a 3D asset of multiple objects generated by DreamGaussian [51]; (e) is our result.

Abstract

Traditional image-to-3D models often struggle with scenes containing multiple objects due to biases and occlusion complexities. To address this challenge, we present REPARO, a novel approach for compositional 3D asset generation from single images. REPARO employs a two-step process: first, it extracts individual objects from the scene and reconstructs their 3D meshes using image-to-3D models; then, it optimizes the layout of these meshes through differentiable rendering techniques, ensuring coherent scene composition. By integrating optimal transport-based long-range appearance loss term and high-level semantic loss term in the differentiable rendering, REPARO can effectively recover the layout of 3D assets. The proposed method can significantly enhance object independence, detail accuracy, and overall scene coherence. Extensive evaluation of multi-object scenes demonstrates that our REPARO offers a comprehensive approach to address the complexities of multi-object 3D scene generation from single images. The demo have been available at <https://reparo-3d.github.io/>.

1. Introduction

The field of 3D content creation has advanced rapidly due to progress in reconstruction methods [16, 35] and generative frameworks [40]. Automated 3D content generation enables creators in augmented reality (AR), virtual reality (VR), gaming, and filmmaking [20, 25, 50] by significantly reducing manual labor requirements.

Recent research in 3D generation primarily focuses on optimization-based 2D approaches, particularly text-to-3D [26, 40, 60] and image-to-3D [27, 31, 34, 52] models. For example, Dreamfusion [40] overcomes 3D data scarcity by distilling geometry and appearance from 2D diffusion models; and DreamGaussian [51] integrates 3D Gaussian Splatting into generative pipelines with mesh extraction and texture refinement. However, despite their effectiveness in single-object scenes, these methods face challenges in reconstructing multi-object scenes. This limitation stems from dataset biases: most 3D training samples feature centrally aligned single objects, and preprocessing pipelines often re-center inputs, thereby introducing inherent positional bias. Consequently, occluded objects are frequently misrepresented as fused entities, leading to erroneously merged assets in rendered outputs, as shown in Figure 1 (d). Fur-

* Equal contribution. † Corresponding authors.

thermore, typical 3D generation models output assets as monolithic mesh representations. This forces users to rely on error-prone post-processing steps to segment individual object mesh, which does not ensure the quality of the segmented assets.

To address this challenge, we propose **REPARO**, a compositional 3D generation pipeline capable of generating multiple objects from a single input image. Our approach decomposes the scene into discrete assets, then systematically reassembles and globally optimizes their spatial arrangement. This methodology not only leverages the strengths of off-the-shelf models (the ability to generate single objects with high fidelity) but also addresses inherent challenges in the multi-object scene with complex occlusions and interactions.

Specifically, our pipeline comprises two core stages. First, individual objects are isolated from the input image through cropping and inpainting to generate context-complete single-object images, which are processed using off-the-shelf image-to-3D models [33, 41, 54] to produce high-fidelity 3D assets. Second, the 3D assets are assembled into a cohesive scene and geometrically aligned with the input image. After placing all objects within a unified coordinate system, we employ differentiable rendering to optimize the spatial arrangement of each asset. During rendering, the pose parameters (translation, rotation, and scale) of each 3D asset are treated as optimizable variables, which are iteratively refined via gradient descent using a loss function. Unlike conventional pixel-wise loss function [3], we introduce an optimal transport (OT)-based long-range appearance loss that integrates RGB color, depth, and positional information to constrain global structural relationships between objects. Therefore, this loss can match the rendered image and the reference image. Additionally, a high-level semantic loss is introduced to ensure semantic alignment and coherence between the synthesized scene and the original image context using high-level visual features.

Through these two stages, individual-object reconstruction and combined-scene optimization, our **REPARO** enhances object independence and geometric detail accuracy, particularly for occluded regions. Simultaneously, it preserves both spatial arrangement fidelity and semantic-geometric consistency across the synthesized scene. Critically, our method obviates the need for post hoc mesh segmentation or manual joining operations by generating instance-level meshes directly within the scene context, streamlining the 3D asset creation workflow and improving end-user usability.

To evaluate our method, we conducted experiments on 20 multi-object scenes from the Google Scanned Objects (GSO) dataset [9] (containing 2–6 objects per scene) and 20 additional in-the-wild images for qualitative analysis. Quantitative and qualitative comparisons demonstrate that

REPARO offers significant improvements over previous techniques in ensuring the quality of assets, managing multiple assets, and processing occlusion. Our contributions are summarized as follows:

1. A two-stage compositional generation Pipeline: We propose **REPARO**, a framework to decouple multi-object 3D generation into two stages: (1) per-object high-fidelity reconstruction via inpainting-aware cropping and off-the-shelf models, and (2) global scene optimization using novel differentiable rendering. This structure ensures object-level geometric independence while preserving scene-level contextual integrity.
2. Optimal transport-driven layout loss: We design a long-range appearance loss based on optimal transport (OT) theory, which enforces multi-modal constraints (RGB, depth, positional channels) to align 3D asset layouts with the reference image. By solving for global pixel correspondences, OT mitigates local minima issues inherent in the pixel-wise loss.
3. Semantic loss: We further introduce a semantic alignment loss that leverages high-level visual features to refine object placements. This loss ensures synthesized scenes adhere to the original image’s semantic context.

2. Related work

Text-to-3D generation. Early exploration in 3D generation based on text [15, 35–37, 44, 56, 63] relied on a large-scale pre-trained model of text and image, CLIP [44], to optimize representation as the prior. In light of the promising capability of diffusion models, some works [1, 21, 26, 40, 45, 46, 57, 60] introduce Score Distillation Sampling (SDS) to narrow the gap between novel view images rendered from 3D representation and diffusion prior. Recent works [4, 24, 51, 65] have used 3D Gaussian Splatting [16] instead of NeRF [35] for 3D generation, due to the fact that this efficient 3D representation shows excellent quality [23, 43, 66] in reconstruction tasks.

Image-to-3D generation. Differing from the ambiguity and diversity associated with textual description, 3D generation based on a single image predominantly concentrates on maintaining geometric and textural consistency between the generated assets and the input images. Some early works [34, 52] used 2D diffusion models as prior knowledge to guide the training of 3D representation of the object in input images. However, with the enrichment of 3D data [6, 7], a series of adapting fine-tuned diffusion models [27, 29, 30, 42, 47, 61, 68] represented by Zero-1-to-3 [31] have been continuously developed. Subsequently, further advancements [32, 33, 48, 49] have been proposed, which offer improved quality and multi-view consistency.

Compositional 3D generation. Unlike single-object reconstruction, multi-object 3D generation requires attention to both local details and global spatial relation-

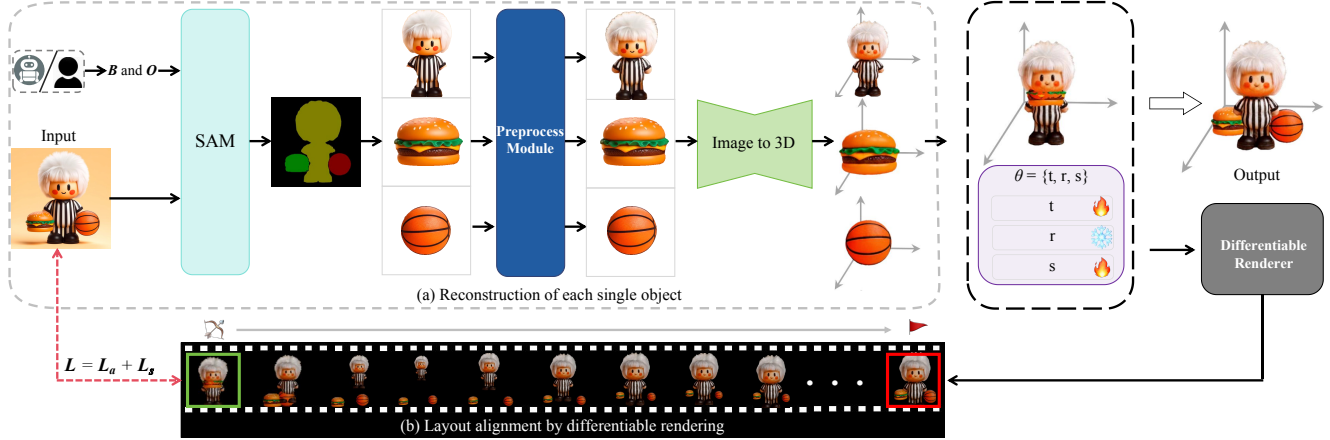


Figure 2. The diagram of the proposed REPARO. (a) is the pipeline to reconstruct the 3D asset of each object in the reference image. B and O denote bounding boxes and occlusion information of each object, respectively. If an object is occluded, the preprocessing module will complement it using the inpainting model. (b) is the process of layout alignment based on differentiable rendering. The parameters of reconstructed meshes are optimized by gradient descent. The loss function L (Eq. 9) consists of the long-range appearance loss L_a and the high-level semantic loss L_s .

ships. Existing approaches address this through two main paradigms: (1) Single-object construction using geometric surface models [10] and editable generative models [53]; (2) Scene composition employing bounding boxes [39], text prompts [59], Gaussian radiance fields [55], and scene graphs [11]. Emerging approaches integrate LLMs for layout interpretation [28], leverage 2D semantic maps [64], or combine Gaussian Splatting with LLMs [67]. Distinct from text/layout-based methods, our approach directly generates multiple high-quality 3D assets from a single compositional image while preserving input-aligned spatial arrangements.

3. Method

The field of 3D generation is advancing rapidly, with numerous image-to-3D models [51, 54] capable of generating 3D assets from a single image. However, these models typically focus on individual objects, making it challenging to apply them to scenes containing multiple objects. To address this limitation, we propose **REPARO** for compositional 3D object generation from a single image. Our approach consists of two steps: (1) Extracting each target object from the given image and reconstructing their 3D meshes using off-the-shelf 3D reconstruction models. (2) Optimizing the parameters of each mesh through differentiable rendering to align their layout. In the following section, we provide a detailed explanation of these two steps.

3.1. Reconstruction of Each Single Object

Since most data samples in 3D datasets only have one object located centrally within the image, and most image-to-3D models recenter the object in the preprocessing step, there is an inherent center bias in these models. As a result, the generated 3D assets for single objects are better

than for multiple objects. As shown in Figure 1, DreamGaussian [51] can generate the sound 3D asset for a single object, but it fails in generation for multiple objects. Instead of reconstructing 3D assets of multiple objects at the same time, our **REPARO** first focuses on extracting individual objects from images containing several objects and then generating their 3D assets using image-to-3D models.

Object extraction. As illustrated in Figure 2 (a), given an image I^{ref} with the bounding boxes B and occlusion information O , we use the foundation model SAM [18] to segment each object and obtain their binary masks M . For occluded objects, we utilize a Stable-Diffusion-based inpainting model to reconstruct the occluded parts in the preprocessing module, thereby obtaining complete RGB priors for each object. Plus, we resize and crop the images to centralize the involved objects, which adapts to the center bias of existing image-to-3D models.

Single object reconstruction. After that, we employ off-the-shelf image-to-3D models, such as Dreamgaussian [51] and TripoSR [54], to generate the corresponding 3D assets for each obtained object.

3.2. Layout Alignment by Differentiable Rendering

After obtaining every single object, we need a layout. To achieve the goal, first, we put every object together in one coordinate system as demonstrated in Figure 2 (b), then we use differentiable rendering techniques [19, 22, 62] to optimize the layout of each object. Differentiable rendering aims to recover the scene parameters θ from reference image(s) I^{ref} through analysis by synthesis process. Given an initial estimation of the scene, differentiable rendering can produce a rendered image I together with the gradients with respect to arbitrary scene parameters θ . Briefly, the gradient

can be represented as:

$$\frac{\partial I}{\partial \theta} = \left[\frac{\partial I(p_1)}{\partial \theta}, \dots, \frac{\partial I(p_N)}{\partial \theta} \right] = \left[\frac{\partial I_1}{\partial \theta}, \dots, \frac{\partial I_N}{\partial \theta} \right], \quad (1)$$

where p_i refers to the position of i -th pixel, $I(p_i)$ is the RGB color of i -th pixel, and N is the total number of pixels. We denote $I(p_i) = I_i$ for simplicity in the following. Combining with the loss function between rendered and reference image, we can leverage a gradient-decent-based method like Adam [17] to optimize the scene parameters of interest, like geometry, material, or lights. According to the chain rule, the derivative of the loss function $L(I, I^{ref})$ with respect to scene parameters θ is:

$$\frac{\partial L}{\partial \theta} = \frac{\partial L}{\partial I} \cdot \frac{\partial I}{\partial \theta}. \quad (2)$$

Specifically, for our task, since we have put every generated object in the same coordinate system, the layout of objects in the initial rendered image and reference image may have quite large differences. Such a case is not suitable for the commonly used pixel-wised L_2 loss function and differentiable rendering methods that only compute color derivatives. Imagine that when an object in the reference image has no overlapped area with it in the rendered image, the L_2 loss will not change when we slightly move the object, the gradients of the loss function could get stuck in an undesired local minima and not indicate the right way for optimization. To mitigate this issue, we propose to introduce a loss function that could find the global correspondences. For one thing, inspired by previous work [12, 62], we use the optimal transport algorithm to find a match between the rendered image and the reference image, thus obtaining long-range correspondences. For another thing, we utilize the feature embedding from the visual backbone to get semantic correspondences.

3.2.1. Long-range Appearance Loss Term

Optimal transport theory describes the following problem: suppose there are N suppliers and M demanders within a region. The i -th supplier holds s_i units of goods, and the j -th demander needs d_j units of goods. The transportation cost per unit of goods from i -th supplier to j -th demander is c_{ij} . The goal of the optimal transport algorithm is to find a transportation matrix $T = \{T_{ij} > 0 \mid i = 1, \dots, N; j = 1, \dots, M\}$ that minimizes the total transportation cost:

$$\begin{aligned} \min_T \sum_{i=1}^N \sum_{j=1}^M T_{ij} c_{ij}, \quad T_{ij} > 0, \\ \text{s.t.} \sum_{i=1}^N T_{ij} = d_j, \quad \sum_{j=1}^M T_{ij} = s_i, \quad \sum_{i=1}^N s_i = \sum_{j=1}^M d_j. \end{aligned} \quad (3)$$

In our layout alignment, we consider all the pixels in the rendered image I as suppliers and all the pixels in the

reference image I^{ref} as demanders, where $N = M$ and $s_i = d_j = 1$. For the cost, instead of only considering RGB color distance between i -th pixel and j -th pixel, we define it based on RGB color, depth value, and position distance at the same time:

$$c_{ij} = \alpha \cdot \|I_i - I_j^{ref}\|_2 + \beta \cdot \|D_i - D_j^{ref}\|_2 + \gamma \cdot \|p_i - p_j\|_2, \quad (4)$$

where p_i and p_j refer to the screen space position of i -th pixel and j -th pixel, respectively; D is the depth map predicted from the rendered image I using a frozen depth foundation model $F_D(\cdot)$; D_i denotes the depth value of i -th pixel; and α , β , and γ are hyper-parameters.

Based on Eq. 3 and Eq. 4, we can obtain a transport matrix T using Sinkhorn divergences [5]. T recording a one-to-one mapping between the rendered image I and the reference image I^{ref} . In other words, we can find a target pixel for each pixel in the rendered image. Formally, we define $I_{\sigma(i)}^{ref}$ is the target pixel of I_i , where $\sigma(\cdot)$ is the one-to-one mapping function from the transport matrix T . In addition, rather than taking only RGB color distance as the differentiable rendering loss, we also consider RGB color, depth value, and position distance together as Eq. 4. Accordingly, our appearance loss function can be expressed as:

$$L_a(I, I^{ref}) = \frac{1}{N} \sum_i c_{i\sigma(i)}. \quad (5)$$

Here, N denotes the number of pixels of the rendered image, and the settings of all hyper-parameters are consistent with Eq. 4. To propagate the gradient of the pixel position to the scene parameter, we can define the pixel position as the projection of the shading point used for this pixel in the rasterization process as DROT [62]. According to Eq. 2 and the chain rule, the derivative of L_a with respect to scene parameters θ is:

$$\frac{\partial L_a}{\partial \theta} = \frac{\partial L_a}{\partial I} \cdot \frac{\partial I}{\partial \theta} + \frac{\partial L_a}{\partial D} \cdot \frac{\partial F_D}{\partial I} \cdot \frac{\partial I}{\partial \theta} + \frac{\partial L_a}{\partial p} \cdot \frac{\partial p}{\partial \theta} \quad (6)$$

Note that since we freeze the depth model $F_D(\cdot)$, the gradients will totally contribute to scene parameters. As the mapping function $\sigma(\cdot)$ considers the long-range correspondence, our appearance loss term can also leverage long-range information during differentiable rendering.

3.2.2. High-level Semantic Loss Term

To enhance the semantic information during the alignment process, we further propose incorporating high-level features into the loss function of differentiable rendering. Our semantic loss term is defined as:

$$L_s(I, I^{ref}) = \frac{1}{K} \sum_i^K \|f_i - f_i^{ref}\|_2, \quad (7)$$

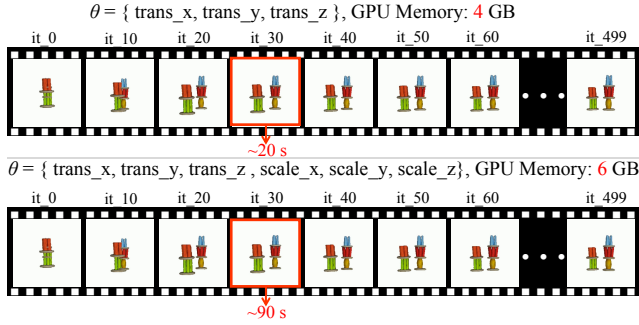


Figure 3. GPU memory usage and approximate elapsed time for optimization of compositional assets layout under different parameter settings.

where f_i and f_i^{ref} are the i -th embedding in the feature map f and f^{ref} , respectively; f and f^{ref} are extracted using a frozen DINO-v2 [38] backbone $F(\cdot)$; and K is the total number of embedding from the last hidden state. Based on this term, we can align the semantic relation between the rendered image and reference image from part to part. According to Eq. 2, the derivative of L_s with respect to layout parameters θ is:

$$\frac{\partial L_s}{\partial \theta} = \frac{\partial L_s}{\partial F} \cdot \frac{\partial F}{\partial I} \cdot \frac{\partial I}{\partial \theta} \quad (8)$$

So far, we have integrated the proposed loss terms together to align the layout of multiple reconstructed 3D objects:

$$L(I, I^{ref}) = \lambda L_a(I, I^{ref}) + (1 - \lambda) L_s(I, I^{ref}), \quad (9)$$

where λ is a hyper-parameter to adjust the weight between appearance and semantic loss term. In practice, we align the layout by optimizing the object’s translation parameter $t = \{\text{trans}_x, \text{trans}_y, \text{trans}_z\}$ and scale parameter $s = \{\text{scale}_x, \text{scale}_y, \text{scale}_z\}$. We exclude the rotation parameter r of assets from the optimization process, as assets generated via image-to-3D models are loaded into the scene with the same orientation as in the input image, eliminating the need for rotation optimization. Thanks to the proposed loss function, we can get the layout of the compositional 3D assets through differentiable rendering.

4. Experiments

4.1. Implementation details

In the reconstruction of 3D assets from a single object, we utilize SAM-ViT-H [18] as the segmentation model to obtain masks corresponding to each bounding box. In the pre-processing module, we extract images of the objects using these masks and subsequently resize and crop the images to center the objects. We then apply the Stable Diffusion-based inpainting model [14] to address occlusions in the objects. During the generation of 3D assets from a single object, we employ the DreamGaussian [51] and TripoSR [54].

Table 1. Performance of different 3D generation models for compositional 3D assets generation. REPARO♣ and REPARO♠ separately denote utilizing DreamGaussian and TripoSR as the image-to-3d model based on our framework.

Method	CLIP ↑	PSNR ↑	SSIM ↑	LPIPS ↓
DreamGaussian [51]	0.807	13.280	0.802	0.240
TripoSR [54]	0.795	17.248	0.863	0.218
Wonder3D [33]	0.801	13.689	0.807	0.238
LRM [13]	0.812	13.664	0.806	0.237
REPARO♣ (ours)	0.833	17.279	0.826	0.234
REPARO♠ (ours)	0.822	17.751	0.865	0.216

Table 2. Runtime and memory usage of each stage. Table 3. Comparison with representative methods on perceptual quality and resource usage.

Stage	VRAM	Time	Method	CLIP ↑	GPU	Time
SAM Segmentation	6 GB	< 1s	DreamGaussian	0.807	8 GB	120 s
Inpainting	8 GB	20 s				
Individual Generation♠	6 GB	< 1s	Wonder3D	0.801	16 GB	180 s
Individual Generation♣	8 GB	120 s	LRM	0.812	14 GB	20 s
Layout Alignment	6 GB	90 s				
Total♠	≤8GB	120 s	REPARO♣ (ours)	0.833	≤8GB	240 s
Total♣	≤8GB	240 s	REPARO♠ (ours)	0.822	≤8GB	120 s

For layout alignment, we use Nvdiffrast [19] as our differentiable rendering framework. During the differentiable rendering process, we optimize the translation x, y, z , and rotation r parameters of each 3D asset using gradient descent. Specifically, we employ the Adam optimizer [17] with a learning rate of 0.02 and a weight decay of 0.999. The total number of iterations is set to 500. All experiments were conducted on RTX 3090 GPUs, and as shown in Figure 3, optimizing parameter t alone takes approximately 20 seconds and utilizes 4GB of memory, while optimizing both parameters t and s to their optimal states requires about 90 seconds and utilizes 6GB of memory. We report per-stage runtime and memory in Table 2. All modules run on ≤8GB VRAM, covering SAM, inpainting, 3D generation, and layout alignment. Table 3 compares our method with baselines in runtime and memory. REPARO achieves better performance under similar or lower resource budgets.

To validate the effectiveness of our method, we selected 20 samples containing multiple objects from the Google Scanned Objects (GSO) dataset [8] as our test set. We used images from 18 views to compute quantitative metrics, including CLIP score [44], Peak signal-to-noise ratio (PSNR), the structural similarity (SSIM) [58], and Perceptual Similarity (LPIPS) [69]. Moreover, we also conducted a subjective evaluation of the generation quality for different models. We collected opinions from 40 participants. Participants selected the visually best subjective quality option among the generation results of 4 models for the same input image. We computed the preference score for each model by dividing the selection count of a specific model by the total selections, reflecting the human preference distribution for each model.

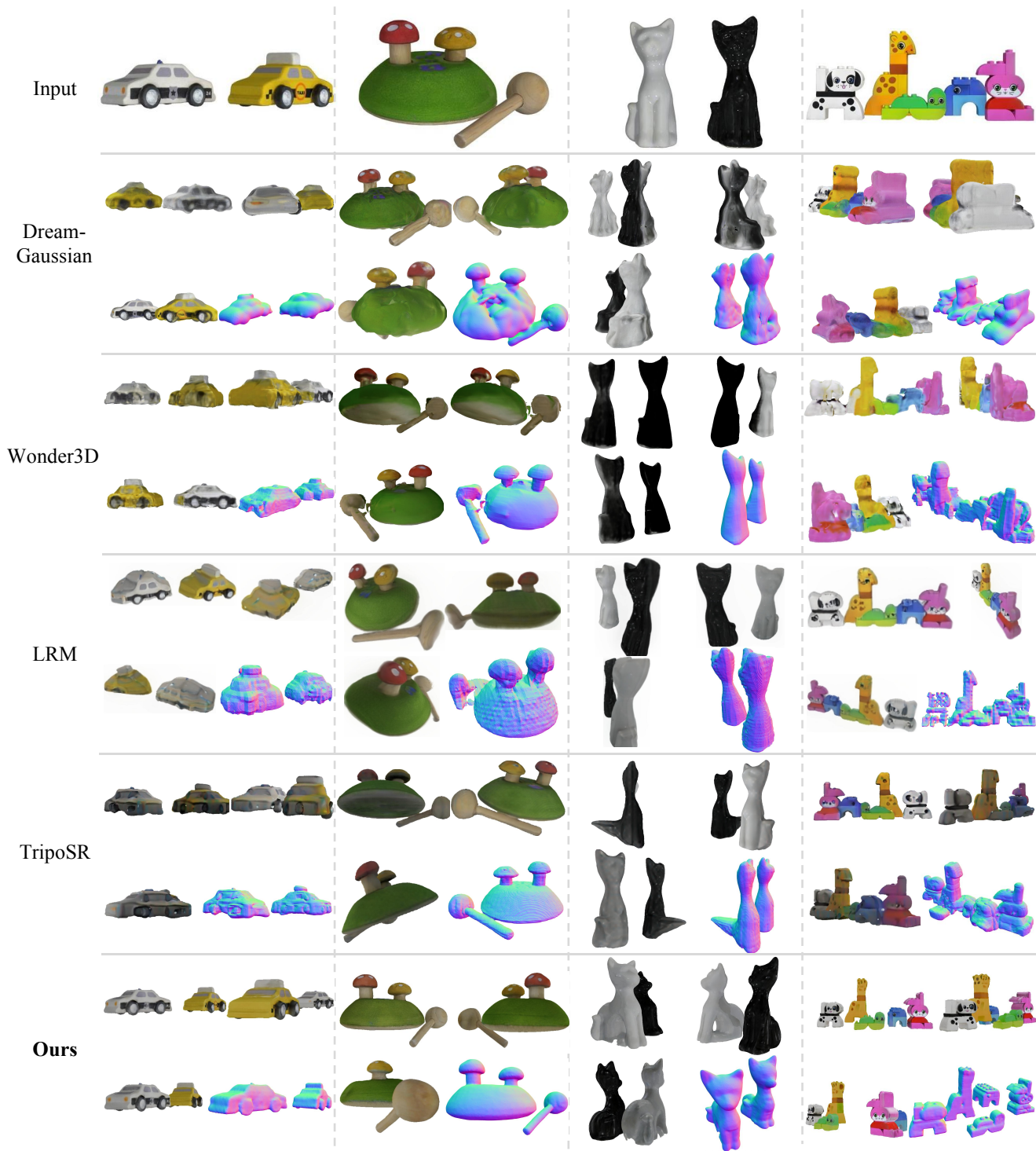


Figure 4. Qualitative comparison with different image-to-3D generation models. Given an input image, previous methods produce inaccurate textures and geometry with noticeable artifacts. Our method generates high-quality, high-fidelity compositional assets with the correct spatial layout.

Table 4. Comparison of different loss functions.

(a) DreamGaussian-based REPARO.					
L_a	L_s	CLIP \uparrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
✓		0.833	17.296	0.826	0.233
	✓	0.809	17.820	0.849	0.210
✓	✓	0.833	17.279	0.826	0.234

(b) TripoSR-based REPARO.					
L_a	L_s	CLIP \uparrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
✓		0.822	17.765	0.865	0.216
	✓	0.813	17.906	0.867	0.212
✓	✓	0.822	17.751	0.865	0.216

Table 5. Comparison of optimization parameters.

(a) DreamGaussian-based REPARO.						
T	R	S	CLIP \uparrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
✓			0.832	17.369	0.826	0.232
✓	✓		0.826	16.906	0.820	0.243
✓	✓	✓	0.833	17.279	0.826	0.234
✓	✓	✓	0.831	17.122	0.823	0.241

(b) TripoSR-based REPARO.						
T	R	S	CLIP \uparrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
✓			0.819	17.797	0.865	0.216
✓	✓		0.822	17.768	0.864	0.219
✓	✓	✓	0.822	17.751	0.865	0.216
✓	✓	✓	0.826	17.752	0.864	0.219

Table 6. Comparison of information in appearance loss L_a .

(a) DreamGaussian-based REPARO.					
Info in L_a	OT	CLIP \uparrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
RGB	×	0.800	18.054	0.853	0.206
RGB	✓	0.828	17.009	0.821	0.245
RGBXY	✓	0.828	17.069	0.821	0.243
RGBDXY	✓	0.833	17.279	0.826	0.234

(b) TripoSR-based REPARO.					
Info in L_a	OT	CLIP \uparrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
RGB	×	0.774	17.918	0.868	0.216
RGB	✓	0.819	17.837	0.865	0.215
RGBXY	✓	0.823	17.847	0.865	0.213
RGBDXY	✓	0.822	17.751	0.865	0.216

4.2. Main results

Quantitative experiment. Table 7 compares the performance (CLIP score, PSNR, SSIM, and LPIPS) of various methods for reconstructing 3D assets of multiple objects using one reference image. DreamGaussian-based REPARO and TripoSR-based REPARO, our proposed methods, show significant improvements over baseline methods (DreamGaussian [51], Wonder3D [33], and LRM [13]). DreamGaussian-based REPARO achieves 83.3% CLIP score, 17.279 PSNR, 0.826 SSIM, and 0.234 LPIPS, indicating better alignment between reconstructed 3D assets and ground-truth 3D assets. TripoSR-based REPARO exhibits superior performance in PSNR (17.751) and SSIM (0.865) and the lowest LPIPS (0.216), highlighting its exceptional quality in image reconstruction and perceptual similarity. These results show that better reconstruction quality on single objects is beneficial for structural similarity and perceptual similarity during differentiable rendering.

Qualitative experiment. Some qualitative comparison results are shown in Figure 4, where our TripoSR-based REPARO can produce high-quality 3D assets with multiple objects. By contrast, DreamGaussian [51], Wonder3D [33], and LRM [13] have problems in layout and completeness, since they possess preferences on an individual object.

User study. In addition to numerical metrics, we conducted a user study to compare our method with others, gathering responses from participants who assessed the realism of different 3D assets. Our approach received 61% approval, while DreamGaussian received 19% approval, LRM received 12% approval, and Wonder 3D received 8% approval. The results consistently showed that our approach was favored over previous methods. For more detailed results, please refer to the supplementary material.

4.3. Ablation study

Ablation for different loss functions in differentiable rendering. We conducted ablation experiments on different

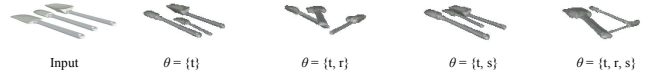


Figure 5. Qualitative comparison for different options of optimization parameters θ .

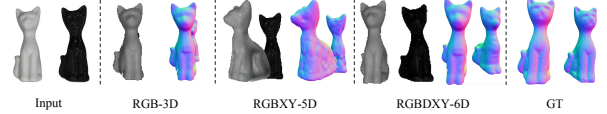


Figure 6. Qualitative comparison for different terms in the appearance loss L_a (Eq. 5).

loss components in Eq. 9, with the results presented in Table 4. The results indicate that incorporating the high-level semantic loss function L_s can bring improvements in both SSIM and LPIPS. However, the enhancement in CLIP score is not as pronounced. Additionally, when only L_s is used, there is a notable decrease in CLIP score. For instance, in the DreamGaussian-based model, the CLIP score decreased by 2.1%. This suggests that relying solely on feature supervision can reduce the effectiveness of layout alignment.

Ablation for different options of scene parameters θ . Table 5 provides an ablation study on the impact of various parameter optimization settings within θ across both DreamGaussian-based and TripoSR-based REPARO models. Specifically, we compare combinations of translation (t), rotation (r), and scale (s) optimizations. For DreamGaussian-based REPARO, the combination of parameters t and s achieves the highest CLIP score (0.833), competitive PSNR (17.279), and the lowest LPIPS (0.234), indicating a high level of perceptual quality and fidelity. Similarly, for TripoSR-based REPARO, optimizing t and s yields a balance between perceptual similarity (CLIP 0.822) and structural fidelity, with the lowest LPIPS (0.216) and strong SSIM (0.865), aligning closely with the input image.

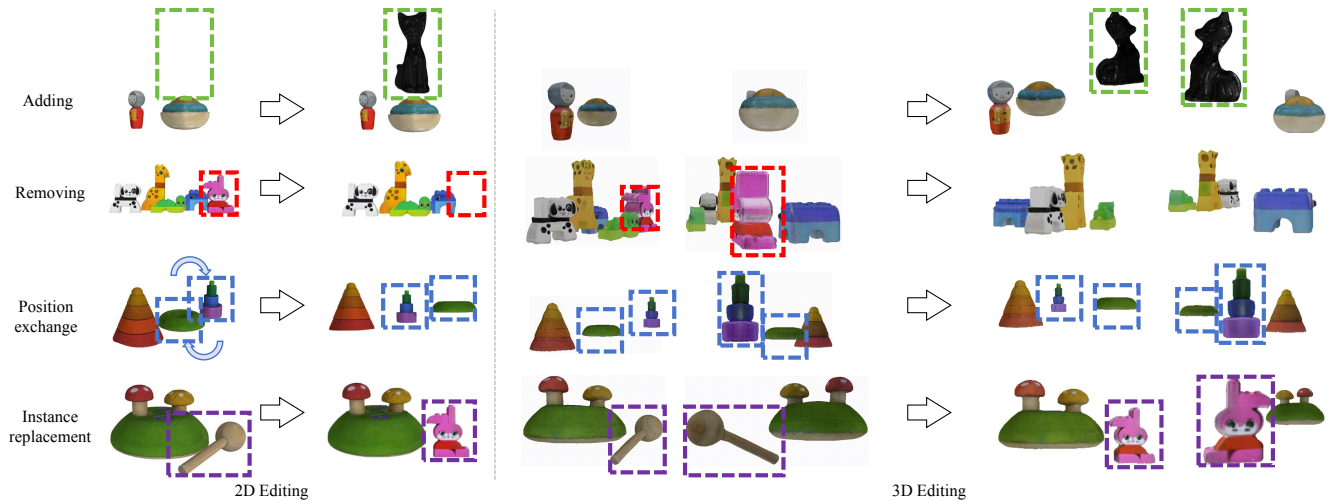


Figure 7. 3D scene editing based on the implementation of 2D image editing. Based on our REPARO framework, it is possible to implement four types of 3D editing guided by 2D image editing, namely adding, removing, position exchanging and instance replacement.

Table 7. DreamGaussian-based (♣), TripoSR-based (♠), and GT-based (♢) results.

Method	CLIP _s ↑	PSNR ↑	SSIM ↑	LPIPS ↓
REPARO♣	0.833	17.279	0.826	0.234
REPARO♠	0.822	17.751	0.865	0.216
REPARO♢	0.906	15.708	0.789	0.321

Notably, as shown in Figure 5, including the rotation parameter r in the optimization tends to degrade alignment with the input image orientation, as shown by the rotation-inclusive configurations, which generally exhibit slight reductions in CLIP scores and increased LPIPS. As a result, we select the combination of t and s for the final configuration, as it achieves optimal perceptual alignment without sacrificing the fidelity to the original image orientation.

Ablation for long-range appearance loss term. As shown in Eq. 5, long-range appearance loss L_a incorporates RGB color distance, position distance, and depth value distance. More importantly, we use the optimal transport to compute pixel matching from a global perspective, thereby obtaining long-range correspondences. Quantitative results in Table 6 demonstrate that in the DreamGaussian-based model, introducing long-range correspondence results in a 2.8% improvement in CLIP score; in the TripoSR-based model, it results in a 4.5% improvement in CLIP score. Additionally, when position distance and depth distance are introduced separately, there is no significant change in quantitative results, but there is a noticeable improvement in qualitative results, as illustrated in Figure 6.

The impact of 3D asset quality. As shown in Tab. 7, the experimental results indicate that the process of generating individual assets in the first stage introduces significant errors. Therefore, we still need a model capable of generating high-quality 3D assets.

4.4. Extension: Using 2D editing to guide 3D editing

We further explored potential applications of our combinatorial generation framework. As depicted in the Figure 7, the entire framework relies on a single input image as guidance, enabling both the reconstruction of individual objects and layout alignment through differentiable rendering. Leveraging this capability, we integrated pre-existing image editing techniques to facilitate editing of the generated 3D scenes. Specifically, we demonstrate four types of editing operations: adding, removing, position exchange, and instance replacement. The 2D editing process is implemented using the AnyDoor [2] model. By comparing the 3D scenes before and after editing, it is evident that the REPARO framework exhibits robust spatial arrangement capabilities and maintains consistency for each assets.

5. Conclusion

In conclusion, we introduce REPARO, a comprehensive approach for generating compositional 3D scenes from single images. Our REPARO addresses the multi-object issue by decomposing the scene into individual objects, reconstructing their 3D meshes, and optimizing their layout through differentiable rendering. During differentiable rendering, we incorporate an optimal transport-based long-range appearance loss which considers RGBDXY information, and a high-level semantic loss which aligns the feature correspondence. Consequently, REPARO is able to obtain multi-object 3D assets with visual-spatial arrangement and contextual consistency. One limitation of our method is that severe occlusion may lead to hallucinated parts inconsistent with the image, and in semantically implausible scenes, layout optimization tends to favor common-sense interpretations over the actual image. We hope our method can pave the way for image to 3D generation in multi-object scenes.

References

- [1] Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 2
- [2] Xi Chen, Lianghua Huang, Yu Liu, Yujun Shen, Deli Zhao, and Hengshuang Zhao. Anydoor: Zero-shot object-level image customization, 2024. 8
- [3] Yongwei Chen, Tengfei Wang, Tong Wu, Xingang Pan, Kui Jia, and Ziwei Liu. Comboverse: Compositional 3d assets creation using spatially-aware diffusion guidance. *arXiv preprint arXiv:2403.12409*, 2024. 2
- [4] Zilong Chen, Feng Wang, Yikai Wang, and Huaping Liu. Text-to-3d using gaussian splatting, 2024. 2
- [5] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013. 4
- [6] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects, 2022. 2
- [7] Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, Eli VanderBilt, Aniruddha Kembhavi, Carl Vondrick, Georgia Gkioxari, Kiana Ehsani, Ludwig Schmidt, and Ali Farhadi. Objaverse-xl: A universe of 10m+ 3d objects. *arXiv preprint arXiv:2307.05663*, 2023. 2
- [8] Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 2553–2560. IEEE, 2022. 5
- [9] Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 2553–2560. IEEE, 2022. 2
- [10] Thomas Funkhouser, Michael Kazhdan, Philip Shilane, Patrick Min, William Kiefer, Ayellet Tal, Szymon Rusinkiewicz, and David Dobkin. Modeling by example. *23(3):652–663*, 2004. 3
- [11] Gege Gao, Weiyang Liu, Anpei Chen, Andreas Geiger, and Bernhard Schölkopf. Graphdreamer: Compositional 3d scene synthesis from scene graphs. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 3
- [12] Zheng Ge, Songtao Liu, Zeming Li, Osamu Yoshie, and Jian Sun. Ota: Optimal transport assignment for object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 303–312, 2021. 4
- [13] Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. Lrm: Large reconstruction model for single image to 3d. *arXiv preprint arXiv:2311.04400*, 2023. 5, 7
- [14] Stable Diffusion Infinity. Stable diffusion infinity. <https://github.com/lkwq007/stablediffusion-infinity>, 2023. Accessed: 2023-10-01. 5
- [15] Ajay Jain, Ben Mildenhall, Jonathan T Barron, Pieter Abbeel, and Ben Poole. Zero-shot text-guided object generation with dream fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 867–876, 2022. 2
- [16] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), 2023. 1, 2
- [17] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations, ICLR*, 2015. 4, 5
- [18] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023. 3, 5
- [19] Samuli Laine, Janne Hellsten, Tero Karras, Yeongho Seol, Jaakko Lehtinen, and Timo Aila. Modular primitives for high-performance differentiable rendering. *ACM Transactions on Graphics*, 39(6), 2020. 3, 5
- [20] Chenghao Li, Chaoning Zhang, Atish Waghvase, Lik-Hang Lee, Francois Rameau, Yang Yang, Sung-Ho Bae, and Choong Seon Hong. Generative ai meets 3d: A survey on text-to-3d in aigc era. *arXiv preprint arXiv:2305.06131*, 2023. 1
- [21] Ming Li, Pan Zhou, Jia-Wei Liu, Jussi Keppo, Min Lin, Shuicheng Yan, and Xiangyu Xu. Instant3d: Instant text-to-3d generation. *International Journal of Computer Vision*, pages 1–17, 2024. 2
- [22] Tzu-Mao Li, Miika Aittala, Frédo Durand, and Jaakko Lehtinen. Differentiable monte carlo ray tracing through edge sampling. *ACM Transactions on Graphics (TOG)*, 37(6):1–11, 2018. 3
- [23] Wanhua Li, Renping Zhou, Jiawei Zhou, Yingwei Song, Johannes Herter, Minghan Qin, Gao Huang, and Hanspeter Pfister. 4d langsplat: 4d language gaussian splatting via multimodal large language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 22001–22011, 2025. 2
- [24] Zhiqi Li, Yiming Chen, Lingzhe Zhao, and Peidong Liu. Controllable text-to-3d generation via surface-aligned gaussian splatting, 2024. 2
- [25] Jingbo Zhang, Zhihao Liang, Jing Liao, Yan-Pei Cao, and Ying Shan. Advances in 3d generation: A survey. *arXiv preprint arXiv:2401.17807*, 2024. 1
- [26] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 300–309, 2023. 1, 2
- [27] Yukang Lin, Haonan Han, Chaoqun Gong, Zunnan Xu, Yachao Zhang, and Xiu Li. Consistent123: One image to

- highly consistent 3d asset using case-aware diffusion priors. *arXiv preprint arXiv:2309.17261*, 2023. 1, 2
- [28] Yiqi Lin, Hao Wu, Ruichen Wang, Haonan Lu, Xiaodong Lin, Hui Xiong, and Lin Wang. Towards language-guided interactive 3d generation: Llms as layout interpreter with generative feedback. *arXiv preprint arXiv:2305.15808*, 2023. 3
- [29] Minghua Liu, Ruoxi Shi, Linghao Chen, Zhuoyang Zhang, Chao Xu, Xinyue Wei, Hansheng Chen, Chong Zeng, Jiayuan Gu, and Hao Su. One-2-3-45++: Fast single image to 3d objects with consistent multi-view generation and 3d diffusion. *arXiv preprint arXiv:2311.07885*, 2023. 2
- [30] Minghua Liu, Chao Xu, Haiyan Jin, Linghao Chen, Mukund Varma T, Zexiang Xu, and Hao Su. One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization. *Advances in Neural Information Processing Systems*, 36, 2024. 2
- [31] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9298–9309, 2023. 1, 2
- [32] Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. Syncdreamer: Generating multiview-consistent images from a single-view image. *arXiv preprint arXiv:2309.03453*, 2023. 2
- [33] Xiaoxiao Long, Yuan-Chen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, et al. Wonder3d: Single image to 3d using cross-domain diffusion. *arXiv preprint arXiv:2310.15008*, 2023. 2, 5, 7
- [34] Luke Melas-Kyriazi, Christian Rupprecht, Iro Laina, and Andrea Vedaldi. Realfusion: 360 reconstruction of any object from a single image. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 2
- [35] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *Proceedings of the European conference on computer vision (ECCV)*, pages 405–421, Cham, 2020. Springer International Publishing. 1, 2
- [36] Nasir Mohammad Khalid, Tianhao Xie, Eugene Belilovsky, and Tiberiu Popa. Clip-mesh: Generating textured meshes from text using pretrained image-text models. In *SIGGRAPH Asia 2022 Conference Papers*, 2022.
- [37] Nasir Mohammad Khalid, Tianhao Xie, Eugene Belilovsky, and Tiberiu Popa. Clip-mesh: Generating textured meshes from text using pretrained image-text models. In *SIGGRAPH Asia 2022 conference papers*, pages 1–8, 2022. 2
- [38] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 5
- [39] Ryan Po and Gordon Wetzstein. Compositional 3d scene generation using locally conditioned diffusion. *arXiv preprint arXiv:2303.12218*, 2023. 3
- [40] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022. 1, 2
- [41] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *The Eleventh International Conference on Learning Representations (ICLR)*, 2023. 2
- [42] Guocheng Qian, Jinjie Mai, Abdullah Hamdi, Jian Ren, Aliaksandr Siarohin, Bing Li, Hsin-Ying Lee, Ivan Skokhodov, Peter Wonka, Sergey Tulyakov, and Bernard Ghanem. Magic123: One image to high-quality 3d object generation using both 2d and 3d diffusion priors. *arXiv preprint arXiv:2306.17843*, 2023. 2
- [43] Minghan Qin, Wanhua Li, Jiawei Zhou, Haoqian Wang, and Hanspeter Pfister. Langsplat: 3d language gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20051–20060, 2024. 2
- [44] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2, 5
- [45] Amit Raj, Srinivas Kaza, Ben Poole, Michael Niemeyer, Nataniel Ruiz, Ben Mildenhall, Shiran Zada, Kfir Aberman, Michael Rubinstein, Jonathan Barron, et al. Dreambooth3d: Subject-driven text-to-3d generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2349–2359, 2023. 2
- [46] Junyoung Seo, Wooseok Jang, Min-Seop Kwak, Hyeonsu Kim, Jaehoon Ko, Junho Kim, Jin-Hwa Kim, Jiyoung Lee, and Seungryong Kim. Let 2d diffusion model know 3d-consistency for robust text-to-3d generation, 2024. 2
- [47] Ruoxi Shi, Hansheng Chen, Zhuoyang Zhang, Minghua Liu, Chao Xu, Xinyue Wei, Linghao Chen, Chong Zeng, and Hao Su. Zero123++: a single image to consistent multi-view diffusion base model. *arXiv preprint arXiv:2310.15110*, 2023. 2
- [48] Yukai Shi, Jianan Wang, He Cao, Boshi Tang, Xianbiao Qi, Tianyu Yang, Yukun Huang, Shilong Liu, Lei Zhang, and Heung-Yeung Shum. Toss: High-quality text-guided novel view synthesis from a single image. *arXiv preprint arXiv:2310.10644*, 2023. 2
- [49] Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. *arXiv preprint arXiv:2308.16512*, 2023. 2
- [50] Ryo Suzuki, Adnan Karim, Tian Xia, Hooman Hedayati, and Nicolai Marquardt. Augmented reality and robotics: A survey and taxonomy for ar-enhanced human-robot interaction and robotic interfaces. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, pages 1–33, 2022. 1
- [51] Jiayang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. *arXiv preprint arXiv:2309.16653*, 2023. 1, 2, 3, 5, 7

- [52] Junshu Tang, Tengfei Wang, Bo Zhang, Ting Zhang, Ran Yi, Lizhuang Ma, and Dong Chen. Make-it-3d: High-fidelity 3d creation from a single image with diffusion prior. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 22819–22829, 2023. 1, 2
- [53] Konstantinos Tertikas, Despoina Paschalidou, Boxiao Pan, Jeong Joon Park, Mikaela Angelina Uy, Ioannis Emiris, Yannis Avrithis, and Leonidas Guibas. Generating part-aware editable 3d shapes without 3d supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4466–4478, 2023. 3
- [54] Dmitry Tochilkin, David Pankratz, Zexiang Liu, Zixuan Huang, , Adam Letts, Yangguang Li, Ding Liang, Christian Laforte, Varun Jampani, and Yan-Pei Cao. Triposr: Fast 3d object reconstruction from a single image. *arXiv preprint arXiv:2403.02151*, 2024. 2, 3, 5
- [55] Alexander Vilesov, Pradyumna Chari, and Achuta Kadambi. Cg3d: Compositional generation for text-to-3d via gaussian splatting. *arXiv preprint arXiv:2311.17907*, 2023. 3
- [56] Can Wang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. Clip-nerf: Text-and-image driven manipulation of neural radiance fields. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3825–3834, 2022. 2
- [57] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A. Yeh, and Greg Shakhnarovich. Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation, 2022. 2
- [58] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 5
- [59] Zhaoning Wang, Ming Li, and Chen Chen. Luciddreaming: Controllable object-centric 3d generation. *arXiv preprint arXiv:2312.00588*, 2023. 3
- [60] Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *Advances in Neural Information Processing Systems*, 36, 2024. 1, 2
- [61] Haohan Weng, Tianyu Yang, Jianan Wang, Yu Li, Tong Zhang, CL Chen, and Lei Zhang. Consistent123: Improve consistency for one image to 3d object synthesis. *arXiv preprint arXiv:2310.08092*, 2023. 2
- [62] Jiankai Xing, Fujun Luan, Ling-Qi Yan, Xuejun Hu, Houde Qian, and Kun Xu. Differentiable rendering using rgbxy derivatives and optimal transport. *ACM Transactions on Graphics (TOG)*, 41(6):1–13, 2022. 3, 4
- [63] Jiale Xu, Xintao Wang, Weihao Cheng, Yan-Pei Cao, Ying Shan, Xiaohu Qie, and Shenghua Gao. Dream3d: Zero-shot text-to-3d synthesis using 3d shape prior and text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20908–20918, 2023. 2
- [64] Han Yan, Yang Li, Zhenan Wu, Shenzhou Chen, Weixuan Sun, Taizhang Shang, Weizhe Liu, Tian Chen, Xiaqiang Dai, Chao Ma, et al. Frankenstein: Generating semantic-compositional 3d scenes in one tri-plane. *arXiv preprint arXiv:2403.16210*, 2024. 3
- [65] Taoran Yi, Jiemin Fang, Junjie Wang, Guanjun Wu, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Qi Tian, and Xinggang Wang. Gaussiandreamer: Fast generation from text to 3d gaussians by bridging 2d and 3d diffusion models. In *CVPR*, 2024. 2
- [66] Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. Mip-splatting: Alias-free 3d gaussian splatting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19447–19456, 2024. 2
- [67] Xuening Yuan, Hongyu Yang, Yueming Zhao, and Di Huang. Dreamscape: 3d scene creation via gaussian splatting joint correlation modeling. *arXiv preprint arXiv:2404.09227*, 2024. 3
- [68] Junwu Zhang, Zhenyu Tang, Yatian Pang, Xinhua Cheng, Peng Jin, Yida Wei, Wangbo Yu, Munan Ning, and Li Yuan. Repaint123: Fast and high-quality one image to 3d generation with progressive controllable 2d repainting. *arXiv preprint arXiv:2312.13271*, 2023. 2
- [69] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 5