This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

COMPGS: Unleashing 2D Compositionality for Compositional Text-to-3D via Dynamically Optimizing 3D Gaussians

Chongjian Ge[1,2,3] Chenfeng Xu[1] Yuanfeng Ji[2] Chensheng Peng[1]
Masayoshi Tomizuka[1] Ping Luo[2] Mingyu Ding[1,4*] Varun Jampani[5*] Wei Zhan[1]
[1] University of California, Berkeley [2] The University of Hong Kong
[3] Adobe Research [4] UNC-Chapel Hill [5] Stability AI

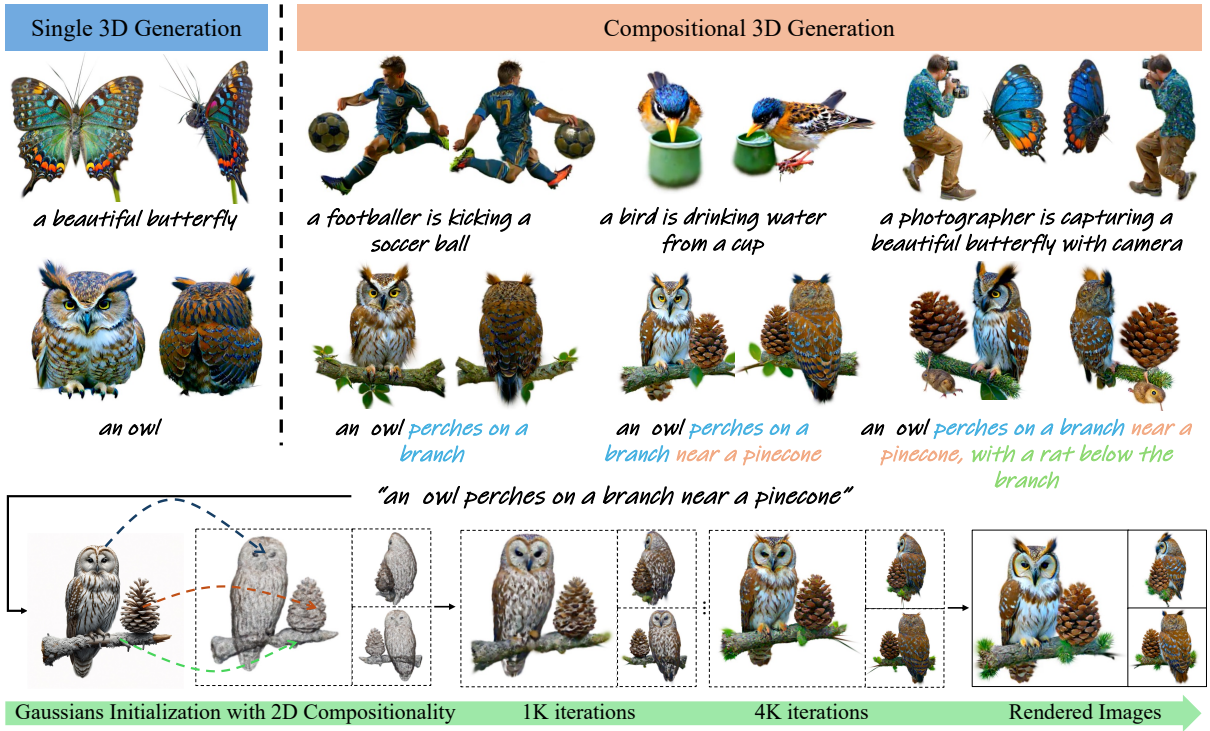


Figure 1. **Illustration of compositional 3D Generation and COMPGS.** All the contents are generated by COMPGS. *Top row:* COMPGS is capable of generating either a single object (e.g., a butterfly) or generating compositional objects with reasonable interactions (e.g., the rightmost figure in the top row). *Middle row:* Beyond text-to-3D generation, COMPGS can be easily extend to 3D editing by progressively adding objects. The colored texts (e.g., *'a branch', 'a pinecone', 'a rat'* in the rightmost figure) denote the added part compared to its previous asset. *Bottom row:* COMPGS achieves compositional text-to-3D by transferring 2D compositionality to initialize 3D Gaussians. COMPGS is further trained with dynamic SDS optimization to produce plausible results.

Abstract

Recent breakthroughs in text-guided image generation have significantly advanced the field of 3D generation. While generating a single high-quality 3D object is now feasible, generating multiple objects with reasonable interactions within a 3D space, a.k.a. compositional 3D generation, presents substantial challenges. This paper introduces COMPGS, *a novel generative framework that employs 3D Gaussian Splatting (GS) for efficient, compositional text-to-3D content generation. To achieve this goal, two core designs are proposed: (1) 3D Gaussians Initialization with 2D compositionality: We transfer the well-established 2D compositionality to initialize the Gaussian parameters on an entity-by-entity basis, ensuring both consistent 3D priors for each entity and reasonable interactions among multiple entities; (2) Dynamic Optimization: We propose a dynamic strategy to optimize 3D Gaussians using Score Distillation Sampling (SDS) loss.* COMPGS *first automatically decomposes 3D Gaussians into distinct entity parts, enabling optimization at both the entity and composition levels. Additionally,* COMPGS *optimizes across objects of varying scales by dynamically adjusting the*

spatial parameters of each entity, enhancing the generation of fine-grained details, particularly in smaller entities. Qualitative comparisons and quantitative evaluations on T^3*Bench demonstrate the effectiveness of* CompGS *in generating compositional 3D objects with superior image quality and semantic alignment over existing methods.* CompGS *can also be easily extended to progressively adding 3D objects, facilitating complex scene generation. We hope* CompGS *will provide new insights to the compositional 3D generation.*

1. Introduction

3D content creation is essential to the modern media industry, yet it has traditionally been labor-intensive and necessitated professional expertise. Typically, designing a single 3D object takes several hours for an experienced designer, and creating complex scenes with multiple 3D objects (as shown in Fig. 1) requires even more effort. Inspired by the recent success of diffusion models in the text-to-image generation [28, 51, 56, 61], much research has focused on exploring 2D diffusion models for text-to-3D generation. Previous work can be divided into two main methodologies: (1) *Feed-forward generation* [29, 39, 73], which entails training generalizable diffusion models on 3D data; and (2) *Optimization-based generation* [10, 41, 47, 52, 68, 71], which utilizes the pretrained 2D diffusion guidance to optimize 3D representations via Score Distillation Sampling (SDS) [52].

While existing work has demonstrated the feasibility of generating single 3D objects, they often struggle to produce compositional 3D content with multiple objects and complex interactions. For example, (1) Feed-forward generation methods struggle in generalizing to complex textual descriptions since the amount of 3D training data is extremely limited [17, 18], and most of the data contains only one object; (2) Optimization-based generation methods face significant challenges with current 2D diffusion guidance when optimizing compositional 3D objects. Typically, using 2D diffusion guidance to optimize a single object is feasible, as it's easy to incorporate various attributes into a single object. However, in compositional 3D generation, 2D guidance struggles to accurately compose multiple objects with different attributes and relationships into a coherent scene [30]. For example, given the prompt *'a blue bench on the left of a green car,'* distinguishing different attributes within the implicit 2D diffusion priors is challenging, resulting in misaligning attributes to different objects or generating incorrect spatial relationships. Thus, optimization-based compositional generation with standard 2D guidance often causes problems like 3D inconsistencies, multi-faced objects, semantic drift, etc [26, 58].

In this work, we propose CompGS, a generative system based on 3D Gaussian Splatting (GS) [33] for compositional text-to-3D generation. To achieve this, we introduce two core designs:

3D Gaussians Initialization with 2D Compositionality. Unlike the implicit representation, i.e., NeRF [48], CompGS uses 3D Gaussians as the representation, which helps to achieve feasible parameter initialization with a coarse 3D shape [31, 77]. As shown in the bottom row of Fig. 1, we first apply a text-to-image model [3, 7, 55] to generate an image that accurately captures the compositionality of multiple objects. Then the image is segmented into different sub-objects (*a.k.a.* entities) according to the entity information in the given prompt. These segmented entities are processed through an image-to-3D model [29, 67] to obtain coarse 3D shapes, which are used to roughly initialize the Gaussian parameters in 3D space, thereby transferring the 2D compositionality to 3D representations.

Dynamic Optimization. Current 2D guidance [41, 47, 52] struggles with optimizing multiple 3D objects simultaneously in a scene, often leading to 3D inconsistencies and semantic shifts. To address these issues, we introduce a dynamic optimization strategy based on Score Distillation Sampling (SDS) loss, consisting of two key components: (1) CompGS dynamically divides the training process to optimize different parts of 3D Gaussians. Specifically, it alternates between optimizing a single object (entity-level optimization) and the entire scene (composition-level optimization). This is achieved by labeling and filtering the Gaussian parameters for inference, and updating them through masking gradients. (2) CompGS dynamically trains the entity-level Gaussians within a normalized 3D space, crucial for compositional 3D generation where object sizes vary. This is particularly challenging when dealing with very small objects, as optimizing Gaussian parameters in a limited 3D space may not adequately capture detailed textures. To mitigate this, we first scale the subspace of each entity to a predefined, standardized volume. Following each training iteration within these standardized volumes, we rescale the Gaussian parameters back to their original sizes. This method dynamically maintains volume consistency throughout the training process, thereby facilitating the capture of fine details within smaller objects.

With these designs, CompGS is capable of generating high-quality compositional 3D objects (as shown in the first row of Fig. 1) and progressive 3D adding (as shown in the second row of Fig. 1). We demonstrate the effectiveness of CompGS both qualitatively and quantitatively. Qualitative comparisons through our user study indicate that CompGS offers superior image quality and semantic alignment compared to existing models (e.g., Fantasia3D [10], ProlificDreamer [71], VP3D [11], etc.). Besides, CompGS's performance on T^3Bench [26] quantitatively highlights its advantages in both semantic control and high-fidelity generation. We hope CompGS can provide valuable insights into

the research of compositional 3D generation.

2. Related Work

Multi-modality 3D Generation. Multi-modality 3D generation can basically categorized into the feed-forward system and optimization-based system. The former one is trained end-to-end on multi-view dataset [5, 17] for zero-shot generation, typically based on 3D representations, including NeRF [29, 67], 3D Gaussians [66, 73], tri-planes [59, 69] and feature grids [32]. Despite the excellent performance achieved, the single-object training data [5, 17, 18, 45] limits their generative abilities in scenarios involving multiple objects. Optimization-based 3D systems lift 2D diffusion priors [7, 8, 51, 56, 57] for 3D generation, and typically train 3D representations on a prompt-by-prompt basis. For example, DreamFusion [52] introduces Score Distillation Sampling (SDS) loss to transfer 2D diffusion models into the 3D domain. Magic3D [53] employs a coarse-to-fine scheme to improve both efficiency and effectiveness, while Fantasia3D [10] decouples the modelling of geometry and appearance. Despite the advancements of these methods addressing various challenges, they also usually struggle with compositional 3D generation. T^3bench's examination of ten prominent optimization-based models [10, 41, 47, 52, 58, 63, 68, 71, 77] revealed frequent issues in compositional generation. The implicit 2D diffusion guidance used in these methods [10, 52, 58, 71] often fails to maintain 3D consistency across different views, leading to significant discrepancies in compositional 3D scenes.

Compositional Generation. Compositional generation involves creating content involving multiple objects with logical interactions. These interactions include, but are not limited to, concept relations [43], attribute association with colors [6, 22, 50], and spatial relationships between objects [9, 72]. There is considerable focus on various aspects of compositional 2D generation, such as learning from human feedback [20, 37, 75, 80], enhancing image captions [3, 7, 8, 16], designing effective networks [44], and etc. T2I-CompBench [30] further proposed a comprehensive benchmark and boosted the compositionality of text-to-image models. Compared to 2D compositional generation, its 3D counterpart is under-explored due to 3D geometry and appearance complexities [1, 12, 25, 40, 64, 74, 76]. For example, Set-the-Scene [15] and Scenewiz3d [79] propose to adopt object layouts to generate compositional scenes. However, they are mainly based on the implicit NeRF representations, so the features between objects cannot be well decoupled, resulting in a relatively blurred rendering effect. VP3D [11] takes a different strategy to employ visual features for compositional generation. However, its visual features are only used as implicit supervision, which does not adequately ensure 3D consistency or address the Janus problems [58]. Besides, LucidDreaming [70] requires human-annotated layout priors for local optimization, which is labour-intensive and inaccurate; while GraphDreamer [23] optimizes all interactive objects from scratch, which is time-consuming and difficult to optimize. The most concurrent works, GALA3D [81], propose optimizing the spatial relationships of well-trained Gaussians for compositional generation. Though it achieves plausible spatial interactions, it is inefficient in generating complex mutual interactions among objects. More comparisons are in the appendix. We propose CompGS to address the above challenges, emphasizing efficient optimization and the generation of complex interactions.

3. Methodology

In this section, we revisit 3D Gaussian Splatting [33] and diffusion priors for 3D generation. We then provide an overview of CompGS, which includes initializing 3D Gaussians to incorporate compositionality priors and dynamic SDS optimization to generate high-fidelity, 3D consistent objects with complex interactions. The notations used in this section are also detailed in Appendix A.1 for clarity.

3.1. Preliminaries

3D Gaussian Splatting (GS). GS [33] has recently revolutionized novel-view synthesis of objects/scenes by its real-time rendering. Specifically, GS represents the 3D objects/scenes by N explicit anisotropic Gaussians with center positions μ_i, covariances Σ_i, opacities α_i and colors c_i, where $i \in N$. The color $\mathcal{C}(\mathbf{p})$ of image pixel $\mathbf{p}$ can be calculated through point-based volume rendering [35, 36] by integrating the color and density of the 3D Gaussians intersected by a ray, as follows:

$$\mathcal{C}(\mathbf{p}) = \sum_{i=1}^{N} c_i \sigma_i \prod_{j=1}^{i-1} (1 - \sigma_j), \tag{1}$$

$$\sigma_i = \alpha_i \exp\left[-\frac{1}{2} (\mathbf{p} - \hat{\boldsymbol{\mu}}_i)^\top \hat{\Sigma}_i^{-1} (\mathbf{p} - \hat{\boldsymbol{\mu}}_i) \right], \tag{2}$$

where $\hat{\boldsymbol{\mu}}_i$ and $\hat{\Sigma}_i$ denote the projected center positions and covariances of the 2D Gaussians, transformed from 3D space to the 2D camera's image plane. The image plane can be segmented into tiles during rendering for parallel processing. Unlike implicit representations, such as NeRF [2, 48, 49], GS offers two key advantages for compositional 3D generation: (1) 3D Gaussians facilitate localized rendering, allowing for the rendering of *object A* in one specific sub-space and *object B* in another, thus enabling global compositional generation on a divide-and-conquer basis; (2) 3D Gaussians enable direct parameter initialization, simplifying the integration of compositional priors at the initialization stage.

Diffusion Priors for 3D Generation. Diffusion-based generative models (DMs) [19, 60, 62] have been widely utilized to provide implicit priors for 3D object generation via score distillation sampling (SDS) [52]. Specifically, given a 3D

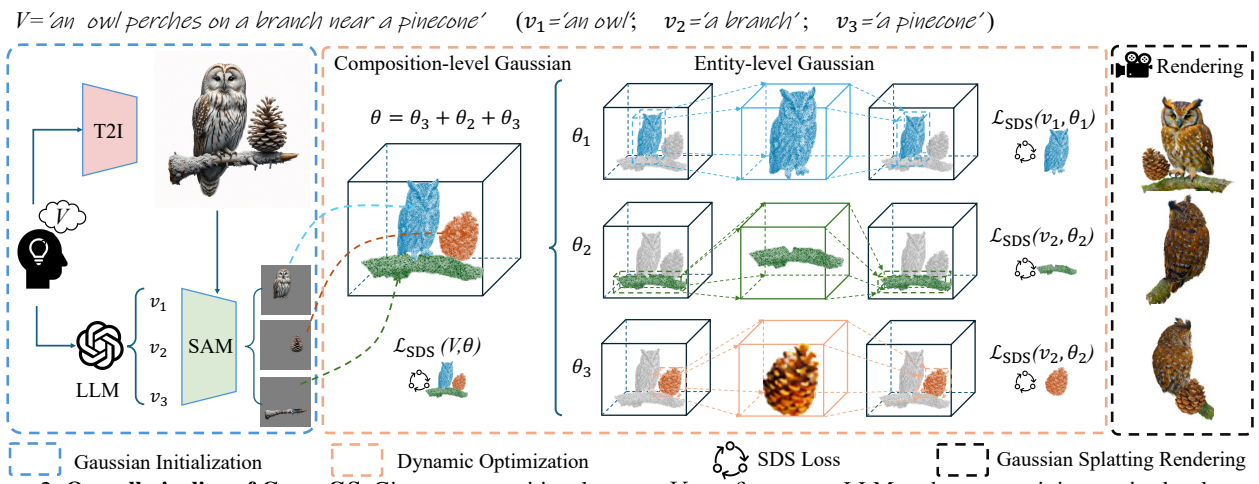


Figure 2. **Overall pipeline of CompGS.** Given a compositional prompt V, we first use an LLM to decompose it into entity-level prompts $\{v_l\}$, guiding the segmentation of each entity from the compositional image generated by T2I models. The segmented images initialize entity-level 3D Gaussians via image-to-3D models [29, 67]. CompGS employs a dynamic optimization strategy, alternating between composition-level optimization of θ and entity-level optimization of $\{\theta_l\}$. For entity-level optimization, COMPGS dynamically maintains volume consistency to refine the details of each objects, particularly the small one.

model whose parameters are θ, a differentiable rendering process g, the rendered images could be obtained via $\mathrm{x} = g(\theta)$. To ensure the rendered images resemble those generated by the DM ϕ, SDS first formulates the sampled noise $\hat{\epsilon}_\phi(\mathrm{z}_t; v, t)$ with the noisy image z_t, text embedding v, and noise level t. By comparing the difference between the added Gaussian noise ϵ and the predicted noise $\hat{\epsilon}_\phi$, SDS constructs gradients that could be back-propagated to update θ via:

$$\nabla_\theta \mathcal{L}_{\text{SDS}}(\phi, \mathrm{x} = g(\theta)) \triangleq \mathbb{E}_{t,\epsilon}\left[w(t)\left(\hat{\epsilon}_\phi(\mathbf{z}_t, v, t) - \epsilon\right)\frac{\partial \mathrm{x}}{\partial \theta}\right], \tag{3}$$

where $w(t)$ is a weighting function, and the Classifier-free guidance [27] typically amplifies the text conditioning v.

3.2. CompGS

We propose CompGS for efficiently transferring the 2D compositionality to facilitate compositional 3D generation with 3D Gaussians. The overall framework of CompGS is shown in Fig. 2. We first generate a well-composed image from the given complex prompt. After extracting the sub-object (also named entity) information within the prompt by LLM, we utilize the entity prompt to segment the composed image into different parts. Each part will be adopted to initialize a specific space of Gaussians. During the training stage, we propose a dynamic SDS optimization strategy. This strategy first automatically decomposes the training to optimize either entity-level Gaussians or composition-level Gaussians. Then, it employs a volume-adaptive strategy to dynamically optimize varying-sized entities within a consistent 3D space. We detail the initialization process and dynamic SDS optimization in the following sections.

3.2.1. 3D Gaussians Init with 2D Compositionality

Unlike the implicit NeRF [48] representations, explicit 3D Gaussians can be easily initialized with 3D shapes and colors, which facilitates introducing rough 3D priors to ensure 3D consistency [77]. Although existing text-to-3D or image-to-3D models can generate a single 3D object, they struggle with compositional 3D generation as mentioned in Sec. 2, which hinders the creation of compositional 3D priors needed for initialization. Considering this, we propose initializing the compositional Gaussians on an entity-by-entity basis.

As shown in Fig. 2 (left), given a complex prompt V, (e.g., *'an owl perches on a branch near a pinecone'*), we first adopt a 2D diffusion model [3] to generate a composed image I that faithfully captures the compositionality of multiple objects. We extract each entity information in prompt V by prompting LLM to obtain L different entity-level prompts (i.e., $L = 3$ for v_1 *'an owl'* , v_2 *'a branch'* , and v_3 *'a pinecone'*). These prompts are adopted in a text-guided segmentation model [34] to decompose image I into various parts $\{I_l\}$. Until now, each segmented image has only one entity, facilitating using existing image-based 3D generation models [67] to predict a rough triangle mesh $m_l (l \in L)$ of the corresponding 3D entity l. To initialize 3D Gaussians θ_l, we index N points from each mesh; the center positions $\mu_i^l \in \mathbb{R}^3 (i \in N, l \in L)$ are the centers of each vertex of m_l, and the texture colors $c_i^l \in \mathbb{R}^3 (i \in N, l \in L)$ are queried from each vertex of m_l. During the image-to-3D process, we did not perform any cropping or padding on the image I_l, ensuring that the spatial positions of each mesh m_l correspond to their 2D spatial positions. Additionally, we have marked the 3D layouts of each entity asset with 3D bounding boxes bbox_l for subsequent optimization. The

bounding box coordinates are determined by the outer-most center positions of the entity Gaussian.

3.2.2. Dynamic SDS Optimization

As mentioned in Sec. 2, existing SDS methods [10, 52, 58, 71] struggle in optimizing compositional 3D scenes due to the challenges of using implicit diffusion priors to maintain multi-objects consistency and interactions. To address this, we introduce dynamic SDS optimization to enable existing SDS losses to remain effective in compositional generation. Dynamic SDS optimization is performed through the following two procedures.

Automatically Decomposing Optimization to Different Levels. We adopt the Decomposed Optimization (DO) strategy that divides the entire training into $L+1$ stages, including L stages for entity-level optimization to ensure 3D consistency of each L entities, and one stage for composition-level optimization to refine inter-entity interactions. In each training iteration, we randomly render either (1) an entity Gaussian $\theta_l (l \in L)$ to obtain the entity image $\mathrm{x}_l = g(\theta_l)$ ($\theta_l \in \mathrm{bbox}_l$), or (2) the composed Gaussians θ to obtain the composed image $\mathrm{x} = g(\theta)$. Here g denotes the Gaussian Splatting rendering [33] described in Eq. 1.

To update Gaussian parameters, we apply Eq. 3 for gradients backpropagation. Specifically, for entity-level optimization, we substitute the text embedding v by v_l and parameters θ by θ_l in Eq. 3 to optimize entity Gaussian. We adopt MVDream [58] as ϕ, which provides multi-view diffusion priors that can better maintain 3D consistency. To ensure that other entity's parameters remain unchanged, we mask the gradients that are not within the corresponding 3D bounding boxes bbox_l when updating θ_l. For composition-level optimization, we integrate both 2D [56] and 3D diffusion priors [58] to jointly optimize the overall Gaussian parameters θ. The rationale is that 2D priors promote geometry exploration [53], which enhances the generation of plausible interactions between different entities.

Volume-adaptively Optimizing Entity-level Gaussians. Compared to the single object generation, compositional 3D generation using SDS presents additional challenges. Specifically, when objects vary in size, optimizing directly in the original 3D space often results in suboptimal generation for smaller objects, as shown the *'pinecone'* in Fig. 2. To mitigate this issue, we propose a Volume-adaptive Optimization (VAO) strategy that enhances optimization across objects of varying sizes by dynamically standardizing each entity's space to a standardized volumetric scale. This approach is especially beneficial for improving the generation of fine-grained texture details in smaller entities.

To scale the 3D bounding box of an entity Gaussian θ_l with a standardized volumetric space bbox_{std}, we first determine the necessary transformations for the centre positions μ. Specifically, we calculate the shift parameters β, and scale parameters λ, as follows:

$$\beta = \mathrm{Mean}(\mathrm{bbox}_l); \quad \lambda = \mathrm{bbox}_{std}/\mathrm{bbox}_l, \tag{4}$$

where Mean computes the center coordinates of the given bounding box, and bbox_{std} is the target standardized space. Then, we zoom-in the entity Gaussian on its center positions by $\hat{\mu} = \lambda(\mu - \beta)$, where $\hat{\mu}$ denotes the transformed center positions of entity Gaussian. We substitute all μ to $\hat{\mu}$ in Eq. 1 for SDS optimization. After each training iteration, we perform zooming-back via $\mu = \hat{\mu}/\lambda + \beta$. The proposed transformation ensures dynamically conducting SDS optimization on each entity Gaussian at a standardized and consistent scale. Experiments in Sec. 4.3 demonstrate its effectiveness in generating high-quality 3D assets.

3.3. Progressive Adding Objects with CompGS

The dynamic SDS optimization enables CompGS to explicitly control the generation of different parts in 3D scene. This capability can be utilized for progressive increasing 3D objects. Specifically, given well-trained 3D Gaussians θ, we begin by rendering an image from the front view, denoted as $\mathrm{x} = g(\theta)$. Unlike previous works that directly add 3D objects [13], we adopt MagicBrush [78] to modify the 2D rendered image x. The edited 2D image can then be utilized to initialize new Gaussians $\hat{\theta}$ in the original 3D space. We further train the added 3D Gaussians to finally introduce new objects into the compositional scenes. As shown in Fig. 1, given well-trained 3D Gaussians (e.g., 'an owl'), we progressively modify the 2D images to add new objects, such as the *'branch'* and *'pinecone.'* We then train the corresponding 3D Gaussians via dynamic SDS optimization to incorporate these new elements into the 3D scenes.

4. Experiments

4.1. Experimental Settings

Implementation Details. We implemented CompGS using ThreeStudio [24]. We use DALL·E [3] to generate the well-established 2D compositionality, LangSAM [46] to segment the entity image, and TripoSR [67] to generate a preliminary 3D prior (i.e., mesh) from each entity image, respectively. During the training, we employed MVDream [58] to optimize entity-level Gaussians, and both MVDream [58] and *stabilityai/stablediffusion-2-1-base* [57] to optimize composition-level Gaussians. The guidance scale for diffusion models was set as 50. Timestamps were uniformly selected from 0.02 to 0.55 for the first 1,000 iterations and then adjusted to a range from 0.02 to 0.15 for subsequent iterations. We initialized the Gaussian points at $N = 10k$, and progressively increased the density to a maximum of $1000k$ points. We optimized the overall compositional scenes through 10k iterations to achieve optimal results. However, it is worth noting that we empirically found that training with 5k iterations already produces

plausible 3D assets. The learning rates are detailed in Appendix A.3. We set the camera parameters, including radius, azimuth, elevation and FoV (field of view) to be the same as [58]. Experiments were conducted on NVIDIA A100 GPUs(40G). The prompt used for decomposing are provided in Appendix A.11. Our code will be released.

Evaluation Metrics. In addition to user studies and qualitative comparisons, we utilize T^{3}Bench [26] to render 300 prompts in order to evaluate compositional 3D generation on two criteria: (1) the visual quality of the 3D objects and (2) the alignment between the 3D objects and the input prompt. For quality evaluation, we captured multi-focal and multi-view images and sent them to text-image scoring models, CLIP [54], to obtain an average quality score of the generated 3D scene. Regarding the textual alignment, we first followed [45] to perform 3D captioning via BLIP [38] and GPT4, and then computed the recall of the original prompt within the generated caption via ROUGE-L [42].

4.2. Performance Comparisons and Analysis.

Qualitative Evaluation. In Fig. 3, we present qualitative comparisons of compositional 3D generation. Our model, CompGS, is compared with several open-source methods, including DreamFusion [52], Magic3D [53], Latent-NeRF [47], Fantasia3D [10], SJC [68], and ProlificDreamer [71], as well as methods specifically designed to handle intricate text prompts, VP3D [11] in Fig. 3. The results demonstrate that CompGS generates compositional 3D objects with superior quality, greater consistency, and more plausible interactions. DreamFusion, for example, fails to generate reasonable multi-objects from compositional prompts, revealing limitations in SDS loss for optimizing multiple objects simultaneously. Magic3D and Latent-NeRF, though guided by a 3D mesh prior, struggle with generating complex and compositional 3D geometry. Despite employing advanced SDS loss, Fantasia3D, SJC, and ProlificDreamer only improve the appearance of the content but do not fundamentally address multi-object generation. Notably, VP3D employs features from compositional images and human feedback to guide compositional 3D generation. However, it often produces unexpected artifacts (e.g., redundant dots in the top two rows of Fig. 3) or less plausible interactions (e.g., strange spatial relationships in the fourth row). This is likely due to the insufficient utilization of 2D compositionality, as VP3D adopts the implicit image features to guide optimization. In contrast, CompGS generates high-quality and composed content that strictly aligns with the given complex prompts, demonstrating its effectiveness over others. We provide more visual comparisons with both the close-source and open-source methods (including GALA3D [81] in Fig. 9, GraphDreamer [23], DreamGaussian [65] in Fig. 10, etc.) in Appendix A.5 as well as the attached video.

We also demonstrate CompGS's capabilities in generating more diverse contents in Fig. 4. CompGS accurately generates both simple spatial relationships (shown in the first row), and more complex interactions, where objects should react to each other (shown in the second row). Besides, CompGS is not limited to generating just two objects, it can generate multiple objects by expanding the number of entity-level Gaussians. For example, in the third row, CompGS generates scenes with three and four entities, each with high-quality 3D shapes and textures.

Table 1. Quantitative comparisons on T^{3}Bench [26] and user studies show CompGS outperforms feed-forward, optimization-based, and compositional generation models.

Method	T^{3}Bench (*Multiple Objects*)			User Study
	Quality↑	Alignment↑	Average↑	Ranking Score↑
OpenLRM [29]	15.2	25.5	20.4	-
TripoSR [67]	16.7	28.6	22.7	-
DreamFusion [52]	17.3	14.8	16.1	4.54
SJC [68]	17.7	5.8	11.7	3.08
LatentNeRF [47]	21.7	19.5	20.6	4.14
Fantasia3D [10]	22.7	14.3	18.5	2.43
ProlificDreamer [71]	45.7	25.8	35.8	3.56
Magic3D [41]	26.6	24.8	25.7	4.30
Set-the-Scene [15]	20.8	29.9	25.4	-
VP3D [11]	49.1	31.5	40.3	6.71
CompGS (ours)	54.2 (+5.1)	37.9(+6.4)	46.1(+5.8)	7.23

We conduct a user study for further evaluation. We randomly selected 15 prompts in T^{3}Bench, and collected the 3D assets generated by different models. These collected 3D assets were then distributed to individuals for ranking the models based on (1) 3D visual quality and (3) text-3D alignment. We average the ranking of different models as their scores. Results in Tab. 1 show that representative optimization-based models received relatively low average scores, highlighting their limitations in compositional generation. VP3D [11] outperforms its predecessors due to the incorporation of image features but still ranks lower than CompGS, which further validates our effectiveness.

Quantitative Evaluation. In Tab. 1, we benchmark the representative models on T^{3}Bench, focusing on the multi-objects track. Results indicate CompGS achieves superior performance in both quality and textual alignment. Compared with feed-forward methods in the first block, CompGS significantly improves the generation quality by a large margin. Though CompGS is initialized from the 3D priors obtained from [67], CompGS still enhances text-3D alignment, likely due to the dynamic optimization on composition-level Gaussians. Furthermore, compared with the optimization-based methods, CompGS shows significant advantages in both two metrics. Even when compared to methods specifically designed for compositional generation, e.g., VP3D [11] and Set-the-Scene [15], CompGS demonstrates clear improvements, indicating the effectiveness of our designs. Note that CompGS is not limited to compositional generation; the results in the track of single-object generation are included in Appendix A.5.

'A florist is making a bouquet with fresh flowers'

'An artist is painting on a blank canvas'

'Hot popcorn jump out from the red striped popcorn maker'

'A half-eaten sandwich sits next to a lukewarm thermos'

DreamFusion Magic3D LatentNeRF Fantasia3D SJC ProlificDreamer VP3D COMPGS (ours)

Figure 3. **Qualitative comparisons between COMPGS and other text-to-3D models on T^3Bench (multiple objects track).** Compared to others, COMPGS is better at generating highly-composed, high-quality 3D contents that strictly align with the given texts. Watch the animations by **clicking** them (Not all PDF readers support playing animations. Best viewed in Acrobat/Foxit Reader).



Figure 4. **More generated samples by COMPGS.** Four views are shown.

Runtime Comparisons. Though COMPGS is trained with 10k steps, we observed that training the model for 5k iterations already produces high-quality content with minimal loss of texture details. We show runtime comparisons with other models in Appendix A.4. In fact, compositional 3D generation (left) requires a longer training time than single-object generation (right) due to its complexity. Compositional 3D involves optimizing individual objects and ensuring the consistency of compositional scenes, making them more intricate. The runtime is generally relative to

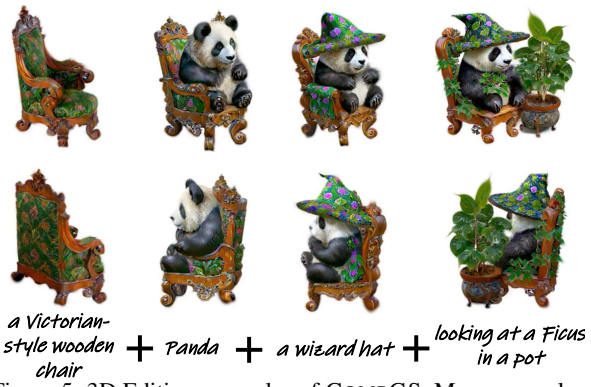


Figure 5. 3D Editing examples of CompGS. More examples could be found in Appendix A.10.



Figure 6. Visual results of the ablation studies on key designs.

the number of objects involved. Compared to open-source compositional 3D methods such as Set-the-Scene [15], Progressive3D [14], and GraphDraemer [23], our proposed CompGS is more efficient in training. For instance, given the prompt *"a parrot talks beside a perch and two bowls,"* Progressive3D takes approximately 250 minutes for 3D generation, while Set-the-Scene requires around 110 minutes. Since many compositional scene generation methods are not open-sourced, we also present the training steps listed in the papers for straightforward comparisons. In addition, CompGS demonstrates comparable and even superior efficiency over other methods in text-to-single object generation. For example, methods such as Magic3D [41] and Fantastic3D [10] require over five hours to optimize a single object. In contrast, CompGS can achieve the same task in approximately 30 minutes. Details are in Appendix A.4.

Extended Applications – Progressive 3D Adding. CompGS allows progressive 3D adding for compositional scenes. We present 3D adding examples in Fig. 5. These examples demonstrate that by transferring edited 2D compositionality and employing dynamic SDS optimization, CompGS can progressively incorporate new 3D entities into the original 3D scenes. For example, CompGS can generate the *'chair'*, *'panda'*, *'hat'* and *'plant'* step by step as shown in Fig. 5.

Limitations. Detailed limitations and future explorations are provided in Appendix A.12 and Appendix A.13.

Table 2. Ablation Studies on T^3Bench [26].

Component	Quality	Alignment	Average
Full Setting	53.8	38.0	45.9
- w/o GS Init.	22.8	18.7	20.8
- w/o DO Strategy	46.8	35.2	41.0
- w/o VAO Strategy	50.8	36.4	43.6

4.3. Ablation Study

We conduct ablation studies to validate the effectiveness of three key designs in CompGS: Gaussian parameters initialization, decomposed optimization (DO), and volume-adaptive optimization (VAO). We randomly choose 20 prompts from T^3Bench for the quantitative evaluation. Meanwhile, we use Fig. 6 to visualize the effect of each component.

Initialization. Instead of initializing Gaussian parameters using 2D compositionality, we conduct random initialization within a predefined 3D bounding box for each entity Gaussian. A significant decrease in the quality and alignment metric is observed in Tab. 2. Besides, the visualization in Fig. 6 confirms that training from random initialization with predefined layouts may result in low-quality textures (e.g., owl's face) and missing entities (e.g., the branch).

Decomposed Optimization. We analyze the decomposed training by discarding the entity-level optimization. Results in Tab. 2 and Fig 6 both indicate that removing decomposed training leads to low-fidelity generation. For example, in Fig. 6, optimizing composition-level Gaussians results in a relatively blurred generation on the owl's fur and the pinecone, validating that optimization on each single entity is necessary to effectively leverage 2D diffusion priors to guide 3D generation.

Volume-adaptive Optimization. To evaluate the effectiveness of volume-adaptive optimization, we choose to optimize the entity Gaussians in their original 3D space. We observed that optimizing different entities within varying sizes of 3D space slightly decreased in quality and alignment, which is noticeable for smaller objects in the compositional scene. For instance, the *'pinecone'* and *'branch'* trained without volume-adaptive optimization exhibited fewer fine-grained details, shown as the green leaves on the branch and the shadow on the pinecone in Fig. 6.

5. Conclusion

In this paper, we introduce CompGS, a user-friendly, optimization-based framework, that achieves compositional text-to-3D generation utilizing Gaussian Splatting. Our three core designs, including initializing 3D Gaussians with 2D compositionality, decomposed optimization for either entity-level or composition-level Gaussians, and volume-adaptive optimization, enable CompGS to generate complex interactions between multiple 3D entities. We hope this work encourages further research in compositional 3D generation.

Acknowledgment The work was supported by Berkeley DeepDrive. The authors thank Google Cloud Platform for providing the computational resources. The work was done during Chongjian Ge's visit to University of California, Berkeley.

References

[1] Haotian Bai, Yuanhuiyi Lyu, Lutao Jiang, Sijia Li, Haonan Lu, Xiaodong Lin, and Lin Wang. Componerf: Text-guided multi-object compositional nerf with editable 3d scene layout. *arXiv preprint arXiv:2303.13843*, 2023.

[2] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *IEEE/CVF International Conference on Computer Vision*, 2021.

[3] James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, 2023.

[4] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.

[5] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015.

[6] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM Transactions on Graphics (TOG)*, 2023.

[7] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, et al. Pixart-alpha: Fast training of diffusion transformer for photorealistic text-to-image synthesis. 2023.

[8] Junsong Chen, Chongjian Ge, Enze Xie, Yue Wu, Lewei Yao, Xiaozhe Ren, Zhongdao Wang, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-\sigma: Weak-to-strong training of diffusion transformer for 4k text-to-image generation. *arXiv preprint arXiv:2403.04692*, 2024.

[9] Minghao Chen, Iro Laina, and Andrea Vedaldi. Training-free layout control with cross-attention guidance. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024.

[10] Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. In *IEEE/CVF International Conference on Computer Vision*, 2023.

[11] Yang Chen, Yingwei Pan, Haibo Yang, Ting Yao, and Tao Mei. Vp3d: Unleashing 2d visual prompt for text-to-3d generation. *arXiv preprint arXiv:2403.17001*, 2024.

[12] 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.

[13] Kai Cheng, Xiaoxiao Long, Kaizhi Yang, Yao Yao, Wei Yin, Yuexin Ma, Wenping Wang, and Xuejin Chen. Gaussianpro: 3d gaussian splatting with progressive propagation. *arXiv preprint arXiv:2402.14650*, 2024.

[14] Xinhua Cheng, Tianyu Yang, Jianan Wang, Yu Li, Lei Zhang, Jian Zhang, and Li Yuan. Progressive3d: Progressively local editing for text-to-3d content creation with complex semantic prompts. *arXiv preprint arXiv:2310.11784*, 2023.

[15] Dana Cohen-Bar, Elad Richardson, Gal Metzer, Raja Giryes, and Daniel Cohen-Or. Set-the-scene: Global-local training for generating controllable nerf scenes. In *IEEE/CVF International Conference on Computer Vision*, 2023.

[16] Xiaoliang Dai, Ji Hou, Chih-Yao Ma, Sam Tsai, Jialiang Wang, Rui Wang, Peizhao Zhang, Simon Vandenhende, Xiaofang Wang, Abhimanyu Dubey, et al. Emu: Enhancing image generation models using photogenic needles in a haystack. *arXiv preprint arXiv:2309.15807*, 2023.

[17] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[18] Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, et al. Objaverse-xl: A universe of 10m+ 3d objects. *Advances in Neural Information Processing Systems*, 2024.

[19] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 2021.

[20] Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, Kashun Shum, and Tong Zhang. Raft: Reward ranked finetuning for generative foundation model alignment. *arXiv preprint arXiv:2304.06767*, 2023.

[21] Danny Driess, Zhiao Huang, Yunzhu Li, Russ Tedrake, and Marc Toussaint. Learning multi-object dynamics with compositional neural radiance fields. In *Conference on robot learning*, pages 1755–1768. PMLR, 2023.

[22] Weixi Feng, Xuehai He, Tsu-Jui Fu, Varun Jampani, Arjun Akula, Pradyumna Narayana, Sugato Basu, Xin Eric Wang, and William Yang Wang. Training-free structured diffusion guidance for compositional text-to-image synthesis. *arXiv preprint arXiv:2212.05032*, 2022.

[23] Gege Gao, Weiyang Liu, Anpei Chen, Andreas Geiger, and Bernhard Schölkopf. Graphdreamer: Compositional 3d scene synthesis from scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21295–21304, 2024.

[24] Yuan-Chen Guo, Ying-Tian Liu, Ruizhi Shao, Christian Laforte, Vikram Voleti, Guan Luo, Chia-Hao Chen, Zi-Xin Zou, Chen Wang, Yan-Pei Cao, and Song-Hai Zhang. threestudio: A unified framework for 3d content generation.

https://github.com/threestudio-project/threestudio, 2023.

[25] Haonan Han, Rui Yang, Huan Liao, Jiankai Xing, Zunnan Xu, Xiaoming Yu, Junwei Zha, Xiu Li, and Wanhua Li. Reparo: Compositional 3d assets generation with differentiable 3d layout alignment. *arXiv preprint arXiv:2405.18525*, 2024.

[26] Yuze He, Yushi Bai, Matthieu Lin, Wang Zhao, Yubin Hu, Jenny Sheng, Ran Yi, Juanzi Li, and Yong-Jin Liu. T3bench: Benchmarking current progress in text-to-3d generation. *arXiv preprint arXiv:2310.02977*, 2023.

[27] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *Advances in Neural Information Processing Systems*, 2022.

[28] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 2020.

[29] 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.

[30] Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation. *Advances in Neural Information Processing Systems*, 2023.

[31] Heewoo Jun and Alex Nichol. Shap-e: Generating conditional 3d implicit functions. *arXiv preprint arXiv:2305.02463*, 2023.

[32] Animesh Karnewar, Andrea Vedaldi, David Novotny, and Niloy J Mitra. Holodiffusion: Training a 3d diffusion model using 2d images. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[33] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 2023.

[34] 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 *IEEE/CVF International Conference on Computer Vision*, 2023.

[35] Georgios Kopanas, Julien Philip, Thomas Leimkühler, and George Drettakis. Point-based neural rendering with per-view optimization. In *Computer Graphics Forum*, 2021.

[36] Georgios Kopanas, Thomas Leimkühler, Gilles Rainer, Clément Jambon, and George Drettakis. Neural point catacaustics for novel-view synthesis of reflections. *ACM Transactions on Graphics (TOG)*, 2022.

[37] Kimin Lee, Hao Liu, Moonkyung Ryu, Olivia Watkins, Yuqing Du, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, and Shixiang Shane Gu. Aligning text-to-image models using human feedback. *arXiv preprint arXiv:2302.12192*, 2023.

[38] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, 2022.

[39] Jiahao Li, Hao Tan, Kai Zhang, Zexiang Xu, Fujun Luan, Yinghao Xu, Yicong Hong, Kalyan Sunkavalli, Greg Shakhnarovich, and Sai Bi. Instant3d: Fast text-to-3d with sparse-view generation and large reconstruction model. *arXiv preprint arXiv:2311.06214*, 2023.

[40] Runjia Li, Junlin Han, Luke Melas-Kyriazi, Chunyi Sun, Zhaochong An, Zhongrui Gui, Shuyang Sun, Philip Torr, and Tomas Jakab. Dreambeast: Distilling 3d fantastical animals with part-aware knowledge transfer. *arXiv preprint arXiv:2409.08271*, 2024.

[41] 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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[42] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, 2004.

[43] Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B Tenenbaum. Compositional visual generation with composable diffusion models. In *European Conference on Computer Vision*, 2022.

[44] Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B Tenenbaum. Compositional visual generation with composable diffusion models. In *European Conference on Computer Vision*, 2022.

[45] Tiange Luo, Chris Rockwell, Honglak Lee, and Justin Johnson. Scalable 3d captioning with pretrained models. *Advances in Neural Information Processing Systems*, 2024.

[46] Luca Medeiros. Language segment-anything: Sam with text prompt. https://github.com/luca-medeiros/lang-segment-anything, 2024. Accessed: 2024-05-21.

[47] Gal Metzer, Elad Richardson, Or Patashnik, Raja Giryes, and Daniel Cohen-Or. Latent-nerf for shape-guided generation of 3d shapes and textures. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[48] 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. *Communications of the ACM*, 2021.

[49] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM transactions on graphics (TOG)*, 2022.

[50] Dong Huk Park, Samaneh Azadi, Xihui Liu, Trevor Darrell, and Anna Rohrbach. Benchmark for compositional text-to-image synthesis. In *Advances in Neural Information Processing Systems*, 2021.

[51] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

[52] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022.

[53] Guocheng Qian, Jinjie Mai, Abdullah Hamdi, Jian Ren, Aliaksandr Siarohin, Bing Li, Hsin-Ying Lee, Ivan Skorokhodov, Peter Wonka, Sergey Tulyakov, et al. Magic123: One image to high-quality 3d object generation using both 2d and 3d diffusion priors. *arXiv preprint arXiv:2306.17843*, 2023.

[54] 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*, 2021.

[55] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, 2021.

[56] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[57] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[58] 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.

[59] J Ryan Shue, Eric Ryan Chan, Ryan Po, Zachary Ankner, Jiajun Wu, and Gordon Wetzstein. 3d neural field generation using triplane diffusion. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

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

[61] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[62] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[63] Jingxiang Sun, Bo Zhang, Ruizhi Shao, Lizhen Wang, Wen Liu, Zhenda Xie, and Yebin Liu. Dreamcraft3d: Hierarchical 3d generation with bootstrapped diffusion prior. *arXiv preprint arXiv:2310.16818*, 2023.

[64] Shaorong Sun, Shuchao Pang, Yazhou Yao, and Xiaoshui Huang. Comogen: A controllable text-to-3d multi-object generation framework. *arXiv preprint arXiv:2409.00590*, 2024.

[65] Jiaxiang 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.

[66] Jiaxiang Tang, Zhaoxi Chen, Xiaokang Chen, Tengfei Wang, Gang Zeng, and Ziwei Liu. Lgm: Large multi-view gaussian model for high-resolution 3d content creation. *arXiv preprint arXiv:2402.05054*, 2024.

[67] 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.

[68] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A Yeh, and Greg Shakhnarovich. Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[69] Tengfei Wang, Bo Zhang, Ting Zhang, Shuyang Gu, Jianmin Bao, Tadas Baltrusaitis, Jingjing Shen, Dong Chen, Fang Wen, Qifeng Chen, et al. Rodin: A generative model for sculpting 3d digital avatars using diffusion. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[70] Zhaoning Wang, Ming Li, and Chen Chen. Luciddreaming: Controllable object-centric 3d generation. *arXiv preprint arXiv:2312.00588*, 2023.

[71] 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*, 2024.

[72] Qiucheng Wu, Yujian Liu, Handong Zhao, Trung Bui, Zhe Lin, Yang Zhang, and Shiyu Chang. Harnessing the spatial-temporal attention of diffusion models for high-fidelity text-to-image synthesis. In *IEEE/CVF International Conference on Computer Vision*, 2023.

[73] Yinghao Xu, Zifan Shi, Wang Yifan, Hansheng Chen, Ceyuan Yang, Sida Peng, Yujun Shen, and Gordon Wetzstein. Grm: Large gaussian reconstruction model for efficient 3d reconstruction and generation. *arXiv preprint arXiv:2403.14621*, 2024.

[74] Han Yan, Yang Li, Zhennan 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. 2024.

[75] Ling Yang, Zhaochen Yu, Chenlin Meng, Minkai Xu, Stefano Ermon, and Bin Cui. Mastering text-to-image diffusion: Recaptioning, planning, and generating with multimodal llms. *arXiv preprint arXiv:2401.11708*, 2024.

[76] Ling Yang, Zixiang Zhang, Junlin Han, Bohan Zeng, Runjia Li, Philip Torr, and Wentao Zhang. Semantic score distillation sampling for compositional text-to-3d generation. *arXiv preprint arXiv:2410.09009*, 2024.

[77] 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. *arXiv preprint arXiv*, 2023.

[78] Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. Magicbrush: A manually annotated dataset for instruction-guided image editing. *Advances in Neural Information Processing Systems*, 2024.

[79] Qihang Zhang, Chaoyang Wang, Aliaksandr Siarohin, Peiye Zhuang, Yinghao Xu, Ceyuan Yang, Dahua Lin, Bolei Zhou, Sergey Tulyakov, and Hsin-Ying Lee. Towards text-guided 3d scene composition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6829–6838, 2024.

[80] Shu Zhang, Xinyi Yang, Yihao Feng, Can Qin, Chia-Chih Chen, Ning Yu, Zeyuan Chen, Huan Wang, Silvio Savarese,

Stefano Ermon, et al. Hive: Harnessing human feedback for instructional visual editing. *arXiv preprint arXiv:2303.09618*, 2023.

[81] Xiaoyu Zhou, Xingjian Ran, Yajiao Xiong, Jinlin He, Zhiwei Lin, Yongtao Wang, Deqing Sun, and Ming-Hsuan Yang. Gala3d: Towards text-to-3d complex scene generation via layout-guided generative gaussian splatting. *arXiv preprint arXiv:2402.07207*, 2024.