GALA3D: Towards Text-to-3D Complex Scene Generation via Layout-guided Generative Gaussian Splatting

Xiaoyu Zhou [1] **Xingjian Ran** [1] **Yajiao Xiong** [1] **Jinlin He** [1] **Zhiwei Lin** [1]
Yongtao Wang [1 2] **Deqing Sun** [3] **Ming-Hsuan Yang** [3 4]

Figure 1: **GALA3D** generates high-quality complex 3D scenes and supports interactive controllable editing. Existing methods either produce low-quality textures, visual artifacts, and geometric distortions or fail to accurately generate multiple objects and their interactions according to the text.

[1]Wangxuan Institute of Computer Technology, Peking University [2]National Key Laboratory for Multimedia Information Processing [3]Google DeepMind [4]University of California, Merced. Correspondence to: Yongtao Wang <wyt@pku.edu.cn>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

Abstract

We present GALA3D, generative 3D GAussians with LAyout-guided control, for effective compositional text-to-3D generation. We first utilize large language models (LLMs) to generate the initial layout and introduce a layout-guided 3D Gaussian representation for 3D content generation with adaptive geometric constraints. We

then propose an instance-scene compositional optimization mechanism with conditioned diffusion to collaboratively generate realistic 3D scenes with consistent geometry, texture, scale, and accurate interactions among multiple objects while simultaneously adjusting the coarse layout priors extracted from the LLMs to align with the generated scene. Experiments show that GALA3D is a user-friendly, end-to-end framework for state-of-the-art scene-level 3D content generation and controllable editing while ensuring the high fidelity of object-level entities within the scene. The source codes and models will be available at gala3d.github.io.

1. Introduction

Crafting 3D content has been labor-intensive for domain specialists (e.g., 3D artists and interior designers), particularly for complex 3D scenes. Furthermore, the diversity of the generated scenes remains limited, and ordinary users usually find it challenging to customize scenes or edit them.

These issues have prompted the recent emergence of text-to-3D generation models (Chang et al., 2015; Poole et al., 2022; Lin et al., 2023a; Raj et al., 2023). Given a textual description as input, text-to-3D methods optimize the 3D representations under the supervision of pre-trained 2D diffusion priors, producing object-centric 3D contents (Poole et al., 2022; Chen et al., 2023a; Xu et al., 2023; Wang et al., 2023b; Tang et al., 2023).

However, existing text-to-3D generative models struggle to generate complex 3D scenes with multiple objects and intricate interactions because they optimize a shared 3D representation. They lack guidance on interactions and spatial positions of objects and generate low-quality 3D scenes, including distorted geometry, 3D inconsistency, multi-face objects, and content drift across different rendering views.

One recent trend is to introduce manually designed layouts to enforce geometric constraints and capture interactions among multiple objects in the scenes (Po & Wetzstein, 2023; Lin et al., 2023b; Cohen-Bar et al., 2023). However, the implicit NeRF representation (Mildenhall et al., 2020) often cannot satisfy all the constraints from the layout, resulting in textural blurring and geometric distortions. Further, the layout creation requires manual work, which may be time consuming and not friendly for ordinary users.

In this paper, we propose GALA3D, a generative layout-guided Gaussian Splatting framework for complex text-to-3D generation. Instead of handcrafted layouts, GALA3D utilizes large language models (LLMs) to extract instance relationships from textual descriptions and translate them into coarse layouts. We introduce a layout-guided Gaussian representation and adaptively optimize the shape and distribution of Gaussians for high-quality geometry. Further, we integrate a compositional optimization strategy combined with diffusion priors to update the parameters of layout-guided Gaussians, which enforces semantic and spatial consistency among multiple objects. To address the misalignment between layouts generated by LLMs and the generated scene, we iteratively optimize the spatial position and scale of the layouts.

GALA3D presents a user-friendly, end-to-end framework for high-quality scene-level 3D content generation and controllable editing. Extensive qualitative and quantitative studies show that GALA3D attains impressive results on compositional text-to-3D scene generation while ensuring high fidelity of object-level entities within the scene.

We make the following contributions in this paper:

- We introduce GALA3D, a scene-level text-to-3D framework based on generative 3D Gaussian Splatting, which generates high-fidelity, coherent, complex 3D scenes with multiple objects and precise interactions.
- GALA3D bridges text description and compositional scene generation through layout priors obtained from LLMs and a layout refinement module that optimizes the coarse layout interpreted by LLMs.
- GALA3D introduces a layout-guided Gaussian representation with adaptive geometry control to model complex 3D scenes and utilizes a compositional optimization mechanism to tackle the challenge of maintaining 3D consistency in geometry and texture, obtaining accurate interactions among multiple objects.
- GALA3D outperforms existing methods in text-to-3D scene generation and provides a user-friendly, end-to-end framework for high-quality complex 3D content generation and controllable editing conversationally.

2. Related Work

Text-to-3D generation by Neural Radiance Field. The success of text-to-image methods has been extended to text-to-3D generation, resulting in rapid progress. DreamFusion (Poole et al., 2022) first introduces the Score Distillation Sampling (SDS) to optimize NeRF representations from a pre-trained 2D diffusion model. Magic3D (Lin et al., 2023a) improves Dreamfusion with a coarse-to-fine optimization scheme. In contrast, Fantasia3D (Chen et al., 2023a) disentangles the modeling of geometry and appearance. To deal with issues of over-smoothing and out-of-distribution that arise in the diffusion process, ProlificDreamer (Wang et al., 2023b) introduces a principled particle-based variational framework named Variational-Score-Distillation, while SJC (Wang et al., 2023a) proposes the Perturb-and-Average Scoring. Recent works (Xu et al.,

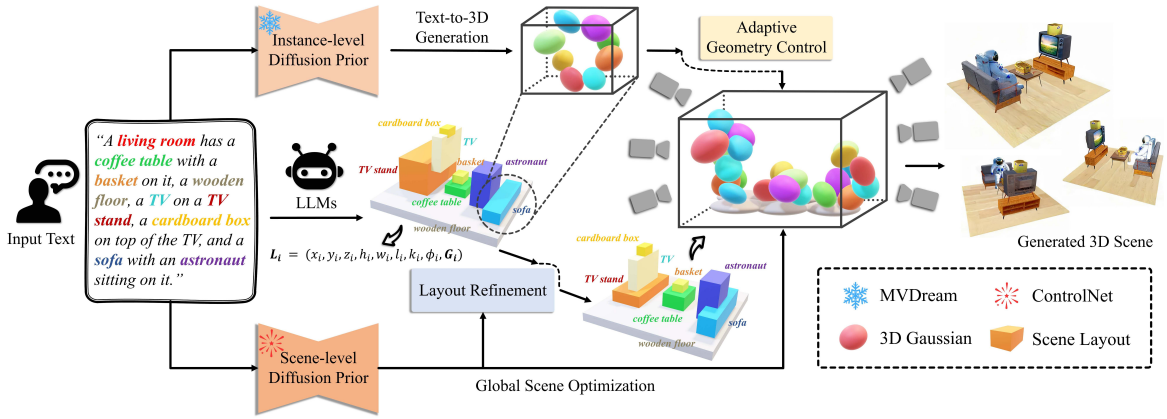

Figure 2: **Overview of our method.** Given a textual description, GALA3D first creates a coarse layout using LLMs. The layout is then utilized to construct the Layout-guided Gaussian Representation, incorporating Adaptive Geometry Control to constrain the Gaussians' geometric shape and spatial distribution. Subsequently, Compositional Diffusions are employed to optimize the 3D Guassians using text-to-image priors compositionally. Simultaneously, the Layout Refinement module refines the initial layout provided by LLMs, enabling better adherence to real-world scene constraints.

2023; Metzer et al., 2023) incorporate additional explicit 3D shape priors to assist in generating high-quality 3D geometric structures and assets. However, the implicit NeRF representation is often insufficient to generate complex scenes that involve multiple objects with intricate interactions.

One promising approach to address these issues is to use layout to constrain the NeRF representation for compositional 3D generation. For example, given the user-defined bounding boxes with corresponding texts, Comp3d (Po & Wetzstein, 2023) blends multiple objects into a scene. Similarly, Set-the-scene (Cohen-Bar et al., 2023) and CompoNeRF (Lin et al., 2023b) generate 3D scenes with compositional NeRFs using pre-defined customizable layouts as object proxies. However, the layout is manually designed to align with text descriptions, which is time-consuming. LI3D (Lin et al., 2023c) and SceneWiz3D (Zhang et al., 2023b) propose using LLMs as a layout interpreter and connect them to off-the-shelf NeRF-based layout-to-3D generative models (Lin et al., 2023b) to generate 3D scenes. However, layouts interpreted by LLMs are often not precise, resulting in misalignment between the layout and the desired scene (e.g., a floating hat, as shown in Figure 8). Besides, compositional NeRF models tend to suffer from degradations in visual quality and geometric deformation because they cannot effectively handle the constraints imposed by layout during the NeRF optimization process, as shown in Figure 1. Here, we adaptively refine the layout interpreted by LLMs to resolve spatial ambiguities and introduce layout-guided Gaussians to model complex 3D scenes.

Text-to-3D generation by 3D Gaussian Splatting. More recently, 3D Gaussian Splatting (Kerbl et al., 2023) (3DGS) provides an efficient point-based representation by optimizing a collection of 3D Gaussian spheres to characterize the 3D space. Recent advances have shown promise in merging 3DGS with diffusion models for text-to-3D generation. Yi et al. (2023) and Liang et al. (2023) utilize 3D text-to-point generative models to generate the initialized point clouds with human priors for 3DGS. In contrast, (Chen et al., 2023b; Tang et al., 2023) adopt a two-stage optimization process for 3DGS involving geometry optimization and texture refinement. To maintain multi-view geometric consistency, GaussianDiffusion (Li et al., 2023) proposes a variational 3DGS combined with structured noise.

However, these object-centric methods optimize a single set of 3DGS and cannot effectively generate complex composite scenes with multiple objects. Further, as there is no constraint on the distribution and shape of Gaussians, these methods may generate distorted geometry, multi-face, and content drift across different rendered views. To address these issues, we introduce layout priors and adaptive geometry control to make 3DGS more controllable. Our method expands the capabilities of 3DGS for representing complex multi-object scenes in a compositional construction manner, resulting in high-quality and consistent 3D scene content.

3D generation with Large Language Models. LLMs possess rich knowledge of large text corpus and can interpret and extract object relationships according to the prompts. However, this capability has not been extensively explored in the field of 3D generation. Some efforts have attempted to leverage LLMs for procedural 3D modeling (Sun et al., 2023), avatars simulation (Ren et al., 2023), text-to-3D benchmark (He et al., 2023), and 3D editing (Fang et al., 2023). The recent combination (Wen et al., 2023; Yang et al., 2024; Feng et al., 2024) of 3D asset retrieval and LLMs has

enabled the creation of restricted indoor scenarios. However, these methods have not explored the capability of LLMs in zero-shot 3D generation and end-to-end complex scene structuring. Furthermore, the aforementioned methods assume that the outputs by LLMs are reliable, leading to potential error propagation with the generated layouts diverging significantly from real-world scenes and textual descriptions. GALA3D addresses this by refining LLM-generated layouts to better align with the generated scenes in 3D space, integrating the 3D generation process with layout optimization.

3. Method

As shown in Figure 2, given a text input, GALA3D first obtains coarse layout prior interpreted by LLMs and constructs Layout-guided Gaussian Representation based on the layout (Section 3.1). Adaptive Geometry Control is introduced to optimize the geometry and distribution of Gaussian ellipsoids, making them more regularized and closely adherent to the geometric surface (Section 3.2). Subsequently, GALA3D utilizes a Compositional Optimization strategy with Diffusion Priors (Section 3.3) for layout-guided Gaussians, combined with Layout Refinement module (Section 3.4) to refine the coarse layout from LLMs. Our method ultimately employs an aggregated loss function to jointly optimize the entire pipeline (Section 3.5).

3.1. Layout-guided Gaussian Representation

A few generative models use geometric priors (e.g., layout) to learn 3D representations and ensure shape and consistency. However, existing methods (Lin et al., 2023b; Po & Wetzstein, 2023; Cohen-Bar et al., 2023) face two challenges: (i) how to obtain relatively reasonable layout priors without manual design, and (ii) how to mitigate the interference of layout constraints in optimizing 3D representations, minimizing visual artifacts and geometric distortions.

Coarse Layout prior interpreted by LLMs. To deal with the first challenge, we introduce LLMs (e.g., GPT-3.5) as coarse layout interpreters. LLMs have showcased remarkable language understanding and relationship extraction capabilities, making layout extraction more efficient and cost-effective than manual crafting. We utilize LLMs to extract instances from textual descriptions and generate their corresponding coarse layout priors. Notably, the layout interpreted by LLMs still deviates from the textual descriptions and actual scenes. Therefore, we introduce the Layout Refinement module to address this issue in Section 3.4.

Layout-guided Gaussian Representation. For the second challenge, we introduce layout constraints into 3DGS representation for the first time and propose Layout-guided Gaussian Representation. At a macro level, Layout-guided Gaussian Representation is a collection of scene Gaussians formed by multiple instance Gaussians corresponding to each instance layout. At a micro level, we employ the Adaptive Geometry Control (Section 3.2) to better constrain the geometry and distribution of Gaussians. Each set of layout-guided Gaussians can be parameterized as:

$$L_i = \{(x_i, y_i, z_i, h_i, w_i, l_i, k_i, \phi_i, G_i), i \in [1, \dots, N]\}, \quad (1)$$

where x_i, y_i, z_i are the position of layout center for the i-th object; h_i, w_i, l_i represent the length, width, and height of the layout boundary, respectively; k_i is the scaling factor; ϕ_i is the rotation angle; G_i denotes instance Gaussians within the layout; and N is the total number of instances in the scene enumerated by LLMs.

Instance Guassians represent the 3D instance through a set of anisotropic Gaussians, defined by center position $\mathbf{p} = (p_x, p_y, p_z) \in \mathbb{R}^3$, color c, opacity α, and covariance $\mathbf{\Sigma_{obj}} = \mathbf{RSS}^\top\mathbf{R}^\top$, where $\mathbf{S}$ is the scale matrix and $\mathbf{R}$ is the rotation matrix. The scene Gaussians can be then defined as a set of layout-guided Guassians $L_{\text{scene}} = \{L_i, i \in [1, \dots, N]\}$ within the entire scene.

Layout-guided Gaussians Rendering at Scene-level. To render the entire scene from the Layout-guided Gaussian Representation, we first transform the Gaussians of each layout L_i into a uniform global scene coordinate system:

$$\mathbf{p}_{\text{scene}} = k_i \mathbf{R_z}(\phi_\mathbf{i})\mathbf{p_i} + (x_i, y_i, z_i)^\top, \quad (2)$$

where $\mathbf{p}_i$ denotes the center position of instance Gaussians $\mathbf{G_i}$, k_i is the scaling factor, and $\mathbf{R_z}(\phi_\mathbf{i})$ is the rotation matrix for rotating ϕ_i degrees around the z-axis. The global scene parameters comprise the covariance matrix of Gaussian collections from all layouts in the scene under transformations:

$$\mathbf{\Sigma}_{\text{scene}} = k_i^2 \mathbf{R_z}(\phi_\mathbf{i})\mathbf{\Sigma_{obj}}\mathbf{R_z}^\top(\phi_\mathbf{i}), \quad (3)$$

where $\mathbf{\Sigma}_{\text{scene}}, \mathbf{\Sigma}_{\text{obj}}$ represent the covariance of the scene Gaussians and instance Gaussians; $\mathbf{R_z}$ is the rotation matrix and $\mathbf{R_z}^\top$ is its transpose. The corresponding 2D covariance can be projected by:

$$\mathbf{\Sigma}'_{\text{scene}} = \mathbf{JW\Sigma}_{\text{scene}}\mathbf{W}^\top\mathbf{J}^\top, \quad (4)$$

where $\mathbf{W}$ is the viewing transformation matrix and $\mathbf{J}$ denotes the Jacobian of the affine approximation of the projective transformation. We further utilize global Gaussian splatting to render the entire scene containing multiple objects:

$$C = \sum_{i \in N} c_i \alpha'_i \prod_{j=1}^{i-1}(1 - \alpha'_j), \quad (5)$$

where C is the color of the rendering pixel; c_i is the rendering color of each Gaussian; and α'_i is the final opacity of the Gaussian. The final opacity α'_i is queried by $\mathbf{Q}$, the rendering pixel's coordinate in the projection space:

$$\alpha'_i = \alpha_i e^{-\frac{1}{2}(\mathbf{Q}-\mathbf{P_i})^\top \mathbf{\Sigma_i^{-1}}(\mathbf{Q}-\mathbf{P_i})}, \quad (6)$$

where $\mathbf{\Sigma_i^{-1}}$ is equivalent to the axes of the ellipsoid; α_i denotes the learned opacity; and $\mathbf{P_i}$ is the spatial position of the Gaussian in the projected plane.

3.2. Adaptive Geometry Control for Gaussians

The raw 3DGS representation adopts the densification scheme for Gaussians, providing good control over the total number of Gaussians. However, this strategy fails to constrain the distribution of Gaussian ellipsoids, resulting in numerous unused invisible Gaussians. It is also incapable of controlling the generation of Gaussians with a uniform regular shape, which shares similar covariances and normal vectors. As a comparison, we propose Adaptive Geometry Control for Gaussians, which achieves adaptive geometric control of Layout-guided Gaussians through distribution constraint and shape optimization. Similar to (de Queiroz & Chou, 2016; Liu et al., 2023; Low & Lee, 2023), given an initialized set of Gaussians, the distribution constraint can be implemented by a density distribution function:

$$\frac{1}{\|\mathbf{p_i} - \zeta_\mathbf{i}\|} \sim \hat{\mathcal{N}}(\mu, \sigma^2), \tag{7}$$

where $\zeta_\mathbf{i} = (x_i, y_i, z_i)$ is the center coordinate of the corresponding layout prior; $\|\mathbf{p_i} - \zeta_\mathbf{i}\|$ is the Euclidean distance between the Gaussian center and $\zeta_\mathbf{i}$; μ is the mean of the Gaussians' distributions; and σ is the standard deviation. Both are changeable parameters. Here, $\hat{\mathcal{N}}$ represents the folded normal distribution, with a truncation range from the layout center to the boundary. We then sample Gaussians near the layouts' surface according to the distribution.

To obtain Gaussian shapes with more regular geometry and scale, we introduce a regularization term:

$$\mathcal{L}_{reg} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{S_i}\,\|\mathbf{q} - \mathbf{p_i}\|\,, \tag{8}$$

where $\mathbf{S_i}$ is a 3D vector along three axes and denotes the scale matrix for the i-th Gaussian; and $\mathbf{q} - \mathbf{p_i}$ denotes the flatness of the Gaussian ellipsoid and will be compressed if too long. As shown in Figure 3, Adaptive Geometry Control adaptively optimizes the distribution and shape of the layout-guided Gaussians, achieving more refined geometric structures and highly detailed textures.

3.3. Compositional Optimization with Diffusion Priors

In pursuit of generating scenes with a consistent style and multiple instances, our method leverages a compositional optimization strategy with diffusion priors to update the parameters of Layout-guided Gaussians. We initially utilize a multi-view diffusion model to optimize instance Gaussian, followed by a scene-conditioned diffusion to align and optimize multiple objects in the scene along with their interactive relationships. Layout loss is further employed

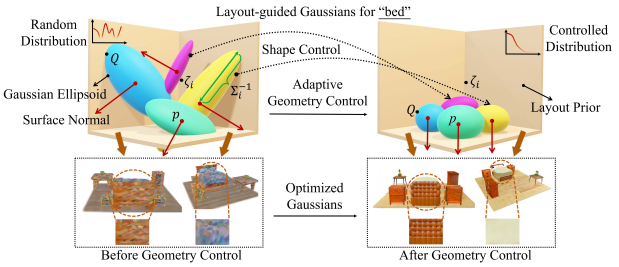

Figure 3: **Adaptive Geometry Control for instance Gaussians.** Note that the improved Gaussian distribution results in enhanced texture and geometry, as the colors of Gaussians on the surface become more aligned.

to ensure the semantic and spatial consistency between the generated 3D scene and the layout prior.

Text-to-3D generation by multi-view diffusion. To optimize instance Gaussian for each instance in the scene, we utilize MVDream (Shi et al., 2023) as a multi-view diffusion prior combined with Score Distillation Sampling (SDS). The gradient for the i-th instance Gaussian can be formulated as:

$$\nabla_{G_i}\mathcal{L}_{SDS}^{(i)} = \mathbb{E}_{\epsilon,\eta}\left[w(\eta)(\epsilon_\varphi(I_i; t_i, M_i, \eta) - \epsilon)\frac{\partial I_i}{\partial G_i}\right], \tag{9}$$

where ϵ is the added noise; t_i is the text prompt corresponding to the i-th instance; η is the time step for optimization; $w(\eta)$ is a weighting function from DDPM (Ho et al., 2020); I_i denotes the sampled image from diffusion prior; M_i is the extrinsic matrix of the camera; G_i denotes the instance Gaussians within the layout, and ϵ_φ is the denoising function for the diffusion process of 3DGS. We embed a virtual camera model to render multi-view images from diffusion prior, with a camera radius of $\frac{3}{4}\|(h_i, w_i, l_i)\|_2$, a horizontal angle of 360°, and uniform sampling of viewing poses.

Global scene optimization by conditioned diffusion. We then introduce conditioned diffusion to optimize the global scene, generating interactions between multiple instances while adhering to the layout prior. Unlike single object generation, we use ControlNet (Zhang et al., 2023a) for compositional optimization, ensuring that the generated scene aligns with the layout. Concretely, we fine-tuned ControlNet to support rendering layouts from multiple viewpoints as input and generate 2D diffusion supervision with layout-text consistency. The gradients of SDS for scene parameters can be formulated as:

$$\nabla_{G_{\text{scene}}}\mathcal{L}_{SDS} = \mathbb{E}_{\epsilon,\eta}\left[w(\eta)(\epsilon_\phi(I; t, \delta, \eta) - \epsilon)\frac{\partial I}{\partial G_{\text{scene}}}\right], \tag{10}$$

where δ is the condition input for the ControlNet, obtained by rendering the 2D images from the layouts. During the diffusion process, the instance-level and scene-level optimization share the same time step η to ensure synchronous

and collaborative learning. t is the textual description of the whole scene encompassing multiple instances; I is the rendered global scene from conditioned diffusion, and G_{scene} denotes the parameters of scene Gaussians.

Global scene optimization by Layout loss. To constrain the generated instances in 3D space to maintain scale, position, and geometric consistency with the provided layout priors, we introduce the layout loss:

$$\mathcal{L}_{layout}^{(i)} = \mathbb{1}_{bbox}(\mathbf{p})[d^x(p_x, x_i, h_i) + d^y(p_y, y_i, w_i) + d^z(p_z, z_i, l_i)], \quad (11)$$

where distance function $d^x(p_x, x_i, h_i)$ calculates the Manhattan distance from each center point outside the 3D layout boundaries to the nearest point on the x-axis and similarly for the other two axes:

$$d^x(p_x, x_i, h_i) = \min(|p_x - (x_i + \frac{h_i}{2})|, |p_x - (x_i - \frac{h_i}{2})|), \quad (12)$$

where p_x is the center coordinate of the instance Gaussian on the x-axis, x_i is the position of the layout center on the x-axis, and h is the height of the layout prior to the i-th instance. The indicator function $\mathbb{1}_{bbox}(\mathbf{p})$ checks whether a point $\mathbf{p}$ is in the bounding box and is 1 if $p_x \in [x_i - \frac{h_i}{2}, x_i + \frac{h_i}{2}]$ and $p_y \in [y_i - \frac{w_i}{2}, y_i + \frac{w_i}{2}]$ and $p_z \in [z_i - \frac{l_i}{2}, z_i + \frac{l_i}{2}]$, and 0 otherwise.

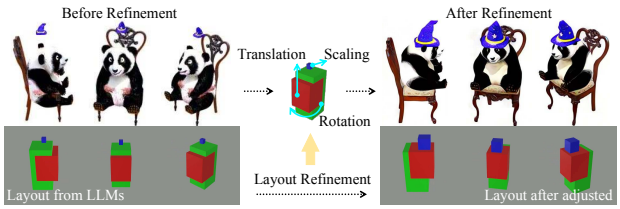

Figure 4: **Layout Refinement.** The LLM-generated layouts exhibit spatial misalignment and abnormal scale. We employ Layout Refinement to optimize the layout, resulting in a more aligned layout with the text and the 3D scene.

3.4. Layout Refinement

Although LLMs possess the ability to extract textual-instance relationships, they may still exhibit significant errors due to the lack of 3D understanding of scenes. The layout priors interpreted by LLMs may deviate from the actual scene and text description, leading to issues like object drift and size discrepancies (Figure 4). To solve this issue, we propose the Layout Refinement module to adaptively adjust the coarse layout generated from LLMs, making it more consistent with scene constraints. The gradient of the layout can be formulated as:

$$\nabla_{(\zeta_i, \alpha_i, k_i, \phi_i)} \mathcal{L}_{Def}^{(i)} = \mathbb{E}_{\epsilon,\eta}[w(\eta)(\epsilon_\phi(I; t, \delta, \eta) - \epsilon)\frac{\partial I}{\partial(\zeta_i, \alpha_i, k_i, \phi_i)}], \quad (13)$$

where $\zeta_i = (x_i, y_i, z_i)$ is the center coordinate of the layout corresponding to the i-th instance; α_i is the opacity; k_i denotes the scale scaling factor, and ϕ_i is the rotation matrix for the layout. All of the above are learnable parameters, continuously updated during the optimization. t is the text prompt, and I denotes the rendered scene image from the conditioned diffusion priors.

3.5. Total Loss

The total loss function can be summarized as

$$\mathcal{L} = \sum_{i=1}^{N}(\beta_1 \mathcal{L}_{SDS}^{(i)} + \beta_2 \mathcal{L}_{layout}^{(i)} + \beta_3 \mathcal{L}_{Def}^{(i)}) + \beta_4 \mathcal{L}_{global} + \beta_5 \mathcal{L}_{reg}, \quad (14)$$

where $\mathcal{L}_{SDS}^{(i)}$ optimize the i-th layout-guided instance Gaussian, $\mathcal{L}_{layout}^{(i)}$ optimize the corresponding i-th layout. $\mathcal{L}_{Def}^{(i)}$ denotes the Layout Refinement for coarse layout priors. $\mathcal{L}_{global}$ denotes the global optimization by conditioned diffusion for the whole scene, and $\mathcal{L}_{reg}$ is applied to supervise the shape control for Gaussians.

4. Experimental Results

Implementation details. We utilize MVDream (Shi et al., 2023) as the multi-view diffusion model, with a guidance scale of 50. The guidance scale of ControlNet is set to 100 to optimize the scene and decrease the timestep linearly during training. For the 3DGS, the learning rates of opacity and position are 5×10^{-2} and 1.6×10^{-4}. The color of 3D Gaussians is represented by the spherical harmonic coefficient, with the degree set to 0 and the learning rate set to 5×10^{-3}. The covariance of the 3D Gaussians is converted into scaling and rotation for optimization, with learning rates of 5×10^{-3} and 10^{-3}, respectively. We set coefficients $\beta_1, \beta_2, \beta_3, \beta_4$ as $\beta_1 = 1, \beta_2 = 10^3, \beta_3 = 10^{-1}, \beta_4 = 10^{-1}$, and $\beta_5 = 10^3$ to balance the magnitude of the losses. For each instance, we initialize the 3D Gaussians with 100,000 particles and discard adaptive density control in 3D Gaussian Splatting to save memory and speed up training. The sampling radius of the camera is set to the scene range in the spherical coordinate system, while horizontal angles are uniformly sampled at 360°. All the experiments are carried out on a single A800 with 80 GB memory.

4.1. Quantitative Comparison

To evaluate our method on the Text-to-3D task, we conduct benchmarking against the state-of-the-art (SOTA) approaches, including NeRF-based methods (Metzer et al., 2023; Wang et al., 2023b), Voxel-based method (Wang et al., 2023a), 3DGS-based methods (Tang et al., 2023; Yi et al., 2023; Chen et al., 2023b; Liang et al., 2023), and compositional NeRF-based generation with layout (Cohen-Bar et al., 2023). Given the absence of ground truth for zero-shot text-to-3D generation, we follow previous works (Jain et al., 2022; Huang et al., 2023) to employ CLIP Score as the evaluation metric to assess the quality and consistency of generated 3D scenes in relation to textual descriptions. As

Table 1: **Overall performance of GALA3D with existing state-of-the-art Text-to-3D approaches using single-object and multi-object text prompts. T** denotes using text prompt and **TL** denotes using text prompt combined with layout. ■, ■, ■, ■ refer to the number of instances in the scene as 1, 3, 7, and 10, respectively. Average represents the average score of multiple generated scenes used for evaluation, including 22 scenes with varying numbers of objects, ranging from 1 to 10.

Methods	Representation	Input	Average	■Case 1	■Case 2	■Case 3	■Case 4
Latent-NeRF (Metzer et al., 2023)	NeRF	T	27.772	22.135	27.482	22.203	19.606
ProlificDreamer (Wang et al., 2023b)	NeRF	T	28.401	30.237	21.913	19.219	25.587
MVDream (Shi et al., 2023)	NeRF	T	30.856	28.756	32.636	26.015	27.417
SJC (Wang et al., 2023a)	Voxel Grid	T	28.775	29.100	31.764	21.154	26.352
DreamGaussian (Tang et al., 2023)	3DGS	T	25.117	26.281	23.051	18.595	25.739
GaussianDreamer (Yi et al., 2023)	3DGS	T	28.351	29.469	31.237	25.727	24.143
GSGEN (Chen et al., 2023b)	3DGS	T	30.293	28.932	29.578	29.959	23.927
LucidDreamer (Liang et al., 2023)	3DGS	T	31.174	28.720	26.533	27.768	26.895
Set-the-scene (Cohen-Bar et al., 2023)	Comp NeRF	TL	29.628	28.129	19.135	29.003	25.899
Ours	Comp 3DGS	T	34.573	31.637	37.658	31.459	35.052

shown in Table 1, text prompts containing varying numbers of objects are chosen to assess the performance of text-to-3D generative models under different settings. Our method excels over all competitors in generating complex 3D scenes with multiple interacting objects.

Compared with NeRF-based and voxel-based methods. To ensure a fair comparison, we employ the vanilla form of Latent-NeRF (Metzer et al., 2023), which employs a text-guided NeRF model to optimize the spatial radiance field in latent space. Our method outperforms Latent-NeRF by a large margin across all evaluated metrics. ProlificDreamer (Wang et al., 2023b) presents Variational Score Distillation for 3D scene generation, maintaining a set of parameters as particles to represent the 3D distribution. However, it fails to model complex scenes with multiple interacting objects using this scheme. GALA3D also boosts the performance of our baseline method MVDream (Shi et al., 2023) in both object-level and scene-level generation and achieves optimal results. SJC (Wang et al., 2023a) regards the 3D diffusion process as an optimization of a vector field. Instead, our method integrates conditioned diffusion with compositional optimization of Gaussians, proven to be more effective for 3D scene generation.

Compared with compositional NeRFs with scene layout. Recent works (Cohen-Bar et al., 2023; Lin et al., 2023b; Po & Wetzstein, 2023) utilize manually designed layouts as priors for compositional NeRF to assist in generating more controllable and intricate scenes. We compare our method with Set-the-scene (Cohen-Bar et al., 2023) and provide the rendered layout interpreted by LLMs as its prior input. Under the same specified layout input, scenes generated by Set-the-Scene exhibit unpleasant blurriness and artifacts. Conversely, our method demonstrates superior scene consistency, spatial geometry, and overall quality, especially in scenes with multiple instances (e.g., ten objects).

Compared with 3DGS-based methods. For Gaussian-based approaches (Tang et al., 2023; Yi et al., 2023; Chen et al., 2023b; Liang et al., 2023), our method exhibits superior performance in 3D generation for both single-object and complex scenes. Our proposed Adaptive Geometry Control for 3DGS ensures the generation of 3D models with high-resolution geometry and texture, thereby avoiding the distortions and blurring observed in existing approaches.

4.2. Qualitative Comparison

We report qualitative comparisons on text-to-3D generation in Figure 1 and Figure 5, including the generation of single-object, interactive multi-object, and complex composite scenes. Visually, our method enables the generation of highly realistic 3D objects and multi-object scenes, surpassing other methods in terms of generated texture, geometric shapes, and semantic consistency. Notably, the NeRF-based generative models produce noticeable artifacts, distortions, and multi-view inconsistencies. The 3DGS-based methods often exhibit multi-face issues and rough geometric shapes. Additionally, these methods show significant deficiencies in scene-text alignment, struggling to accurately generate specified instances, interaction relationships, and correct spatial positions. GALA3D not only precisely generates the desired multiple objects and their interaction relationships but also maintains the consistency between text and multiple objects in the scene, ensuring a unified style.

Compared with compositional scene generation methods. We further compare our approach with recent works (Lin et al., 2023b; Vilesov et al., 2023; Po & Wetzstein, 2023; Lin et al., 2023c) in compositional scene generation, which use the layout as an additional constraint for 3D representation (e.g., NeRF). Since most of these works are not open-sourced, we use the results provided in their papers for comparison, applying the same prompts as input to generate 3D scenes. As shown in Figure 6, these compositional 3D scene generation approaches exhibit unpleasant floating objects, visual artifacts, and geometric distortions in the generated 3D scenes. They also face significant challenges in texture degradation and transition smoothness. In contrast, the 3D scenes generated by our method obtain higher realistic visual effects. Our method also supports more user-friendly and controllable editing in a convenient interactive manner.

Table 2: **User study results.** Human evaluation results comparing GALA3D with other SOTA text-to-3D approaches. Participants scored on the following four metrics, rating from 1 to 10, with higher scores indicating stronger preference.

Methods	Scene Quality	Geometric Fidelity	Text Alignment	Scene Consistency
SJC (Wang et al., 2023a)	5.98	5.04	6.76	4.61
DreamGaussian (Tang et al., 2023)	5.22	4.18	4.30	5.46
GaussianDreamer (Yi et al., 2023)	6.09	5.71	5.23	4.37
GSGEN (Chen et al., 2023b)	6.54	4.23	5.41	6.25
LucidDreamer (Liang et al., 2023)	4.78	5.62	5.03	4.77
Set-the-scene (Cohen-Bar et al., 2023)	6.36	5.03	7.12	6.12
Ours	8.42	8.37	8.55	9.68

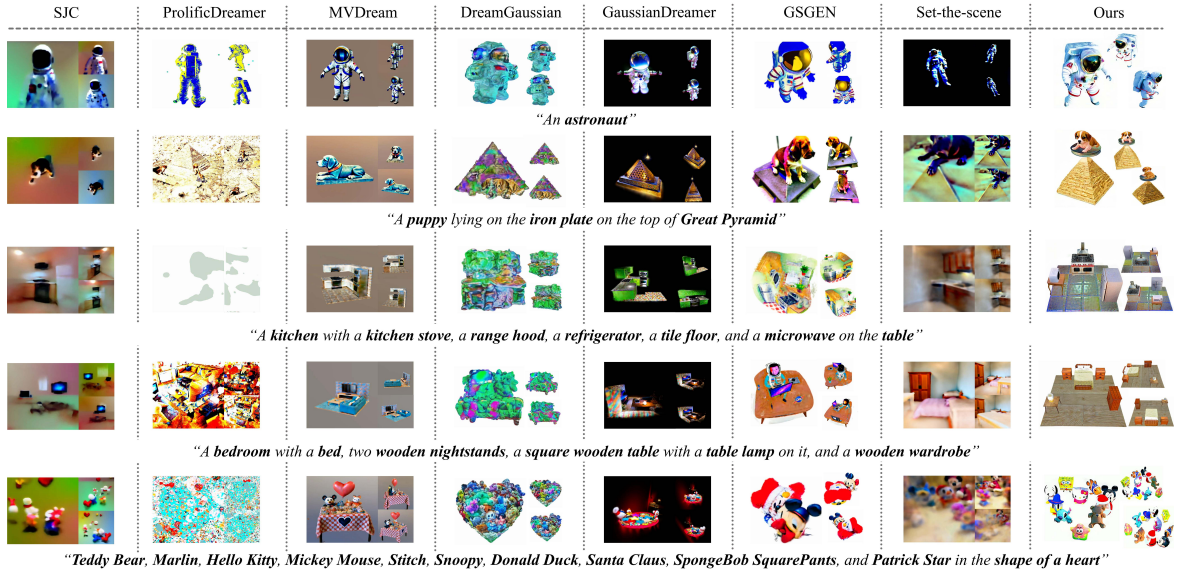

Figure 5: **Qualitative comparisons of text-to-3D generation approaches.** Our method is capable of generating high-quality single-object, interactive multi-object, and complex composite scenes with high consistency in textual descriptions.

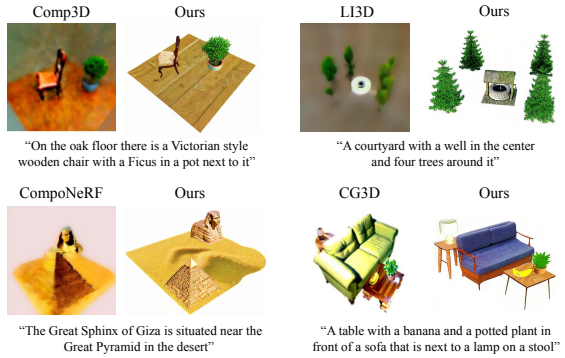

Figure 6: **Comparisons with compositional scene generation methods.** Our method ensures superior coherence and consistency in generated content compared to competitors.

4.3. User Study

We conduct a user study to further evaluate the effectiveness of our method in generating high-quality, text-consistent 3D assets. Specifically, we engage human evaluators to compare 3D models generated by our method and competitive approaches from 8 text descriptions. A total of 125 participants were asked to rank based on four dimensions: (a) Scene Quality, (b) Geometric Fidelity, (c) Text Alignment, and (d) Scene Consistency. Each round of comparison requires participants to rate the four assessment options on a scale from 1 to 10 (10 being the best and vice versa). Among these users, 39.2% are professionals in the fields of art design and 3D modeling.

We report the average score of the trial, reflecting user preferences for generated 3D assets. As shown in Table 2, the results demonstrate a clear preference for our method, receiving consistently positive reviews. Compared with SOTA approaches, GALA3D excels across all four assessments. Our approach also garners preferences from domain experts, demonstrating its potential in practical applications.

4.4. Conversational Interactive Editing

Our method allows conversational interactive editing. Users can freely and controllably edit the generated scene based on textual conversations. Specifically, editing instructions are initially interpreted by LLMs into corresponding layout transformation operations (e.g., adding/removing objects,

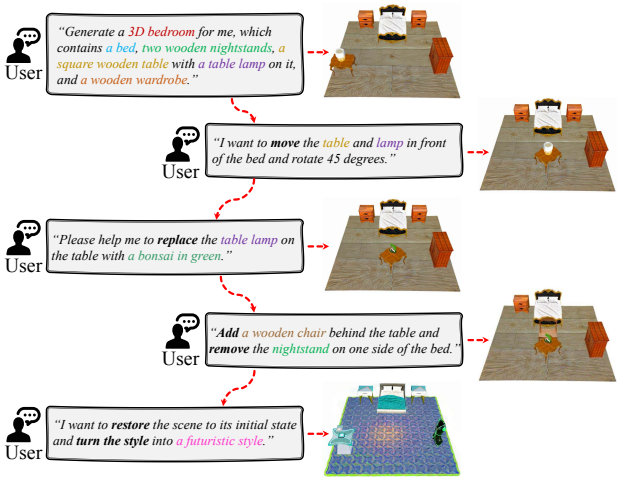

Figure 7: **Conversational Interactive Editing.** Our method facilitates user-friendly and controlled editing of 3D scenes.

moving positions, rotating angles, etc.). We then optimize the Layout-guided Gaussian Representation in the edited local layout areas while maintaining the stability of other regions. Our approach guarantees highly controllable and personalized scene editing, including the addition or removal of objects, spatial adjustments, style transfer, and object interactions. This paradigm of conversational interactive editing combined with LLMs achieves real-world applications, providing a user-friendly 3D assets generation and customized editing pipeline.

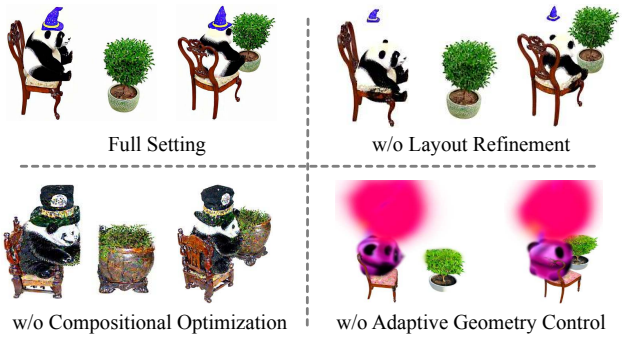

Figure 8: **Visual results of the ablation studies.** Experiments validate the effectiveness of each proposed module, highlighting the crucial role of Layout-guided Gaussian representation coupled with Adaptive Geometry Control in producing high-quality scene geometry and texture.

4.5. Ablation Studies

Adaptive Geometry Control for Gaussians. We replace the Adaptive Geometry Control with the density control scheme employed by the raw 3DGS (Kerbl et al., 2023) and observe a significant decrease in the realism of the generated scene, as shown in Figure 8. The Gaussian densification fails to constrain the distribution and shape of Gaussian ellipsoids, resulting in unpleasant artifacts and blurs. In contrast, our method continuously optimizes the geometric shapes and spatial distributions of 3D Gaussians during the training process. The ablation confirms the effectiveness of Adaptive Geometry Control, which finely improves the complex topological structures and results in enhanced texture and geometry within the global optimization space.

Table 3: **Effect of each module in our proposed method.** AGC is short for Adaptive Geometry Control, LRM denotes the Layout Refinement Module, and COS denotes the Compositional Optimization Scheme.

Model	CLIP Score	Model	CLIP Score
w/o AGC	32.198	w/o LRM	34.293
w/o COS	32.213	w/o $\mathcal{L}_{global}$	34.342
w/o $\mathcal{L}_{layout}$	33.297	Ours-Full	34.885

Layout Refinement Module for LLMs interpreted coarse layout. Directly using the layout interpreted by LLM without refinement results in 3D scenes not well aligned, as shown in Figure 4 and Figure 8. By contrast, the Layout Refinement module enables the optimizing of layouts, continuously adjusting them throughout the denoising process to achieve more intricately aligned interactions among instances, adhering closely to real-world constraints.

Compositional Optimization Scheme. Figure 8 shows ablations to assess the efficacy of the proposed compositional optimization scheme. Specifically, we remove the Global Scene Optimization module, retaining only SDS supervision for instances (with MVDream), and concatenate each object into the scene according to the adjusted layouts. Due to the absence of comprehensive global scene optimization, the generated 3D scenes exhibit impoverished textures and lack scene coherence. Furthermore, the generated geometry only adheres to local layout supervisions, resulting in the emergence of "over-constrained" boundaries.

Effect of Loss Functions. We analyze how each proposed loss function contributes to the final performance. As shown in Table 3, results indicate that both L_{layout} and L_{scene} improve the generating quality, enhancing texture details and maintaining text-3D alignment.

5. Conclusion

In this paper, we present GALA3D, a scene-level text-to-3D framework based on generative layout-guided 3D Gaussian representation, which generates high-fidelity, 3D consistent scenes with multiple objects. Experiments demonstrate that our method surpasses existing methods in text-to-3D generation, showcasing the ability to generate complex scenes with multiple objects and interactions, achieving outstanding texture and geometry. Our method also facilitates interactive and controllable scene editing, achieving an efficient and user-friendly 3D scene generation and editing framework.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grant 62176007 and China National Petroleum Corporation-Peking University Strategic Cooperation Project of Fundamental Research. This work was also a research achievement of Key Laboratory of Science, Technology, and Standard in Press Industry (Key Laboratory of Intelligent Press Media Technology).

Impact Statement

This paper presents work aimed at advancing the fields of Deep Learning and 3D Vision. While AI-generated 3D content offers numerous advantages, it also introduces adverse social impacts. The automation of 3D modeling and scene creation could pose potential risks to the labor market. Additionally, similar to other generative models, our approach has the capacity to generate deceptive and malicious 3D content, highlighting the importance of exercising caution in its application.

References

Chang, A., Monroe, W., Savva, M., Potts, C., and Manning, C. D. Text to 3d scene generation with rich lexical grounding. *arXiv preprint arXiv:1505.06289*, 2015.

Chen, R., Chen, Y., Jiao, N., and Jia, K. Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. In *ICCV*, 2023a.

Chen, Z., Wang, F., and Liu, H. Text-to-3d using gaussian splatting. *arXiv preprint arXiv:2309.16585*, 2023b.

Cohen-Bar, D., Richardson, E., Metzer, G., Giryes, R., and Cohen-Or, D. Set-the-scene: Global-local training for generating controllable nerf scenes. *arXiv preprint arXiv:2303.13450*, 2023.

de Queiroz, R. L. and Chou, P. A. Compression of 3d point clouds using a region-adaptive hierarchical transform. *IEEE Transactions on Image Processing*, 25(8):3947–3956, 2016.

Fang, J., Wang, J., Zhang, X., Xie, L., and Tian, Q. Gaussianeditor: Editing 3d gaussians delicately with text instructions. *arXiv preprint arXiv:2311.16037*, 2023.

Feng, W., Zhu, W., Fu, T.-j., Jampani, V., Akula, A., He, X., Basu, S., Wang, X. E., and Wang, W. Y. Layoutgpt: Compositional visual planning and generation with large language models. *NIPS*, 36, 2024.

He, Y., Bai, Y., Lin, M., Zhao, W., Hu, Y., Sheng, J., Yi, R., Li, J., and Liu, Y.-J. T3bench: Benchmarking current progress in text-to-3d generation. *arXiv preprint arXiv:2310.02977*, 2023.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Huang, Y., Wang, J., Shi, Y., Qi, X., Zha, Z.-J., and Zhang, L. Dreamtime: An improved optimization strategy for text-to-3d content creation. *arXiv preprint arXiv:2306.12422*, 2023.

Jain, A., Mildenhall, B., Barron, J. T., Abbeel, P., and Poole, B. Zero-shot text-guided object generation with dream fields. In *CVPR*, pp. 867–876, 2022.

Kerbl, B., Kopanas, G., Leimkühler, T., and Drettakis, G. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), 2023.

Li, X., Wang, H., and Tseng, K.-K. Gaussiandiffusion: 3d gaussian splatting for denoising diffusion probabilistic models with structured noise. *arXiv preprint arXiv:2311.11221*, 2023.

Liang, Y., Yang, X., Lin, J., Li, H., Xu, X., and Chen, Y. Luciddreamer: Towards high-fidelity text-to-3d generation via interval score matching. *arXiv preprint arXiv:2311.11284*, 2023.

Lin, C.-H., Gao, J., Tang, L., Takikawa, T., Zeng, X., Huang, X., Kreis, K., Fidler, S., Liu, M.-Y., and Lin, T.-Y. Magic3d: High-resolution text-to-3d content creation. In *CVPR*, 2023a.

Lin, Y., Bai, H., Li, S., Lu, H., Lin, X., Xiong, H., and Wang, L. Componerf: Text-guided multi-object compositional nerf with editable 3d scene layout. *arXiv preprint arXiv:2303.13843*, 2023b.

Lin, Y., Wu, H., Wang, R., Lu, H., Lin, X., Xiong, H., and Wang, L. Towards language-guided interactive 3d generation: Llms as layout interpreter with generative feedback. *arXiv preprint arXiv:2305.15808*, 2023c.

Liu, X., Zhan, X., Tang, J., Shan, Y., Zeng, G., Lin, D., Liu, X., and Liu, Z. Humangaussian: Text-driven 3d human generation with gaussian splatting. *arXiv preprint arXiv:2311.17061*, 2023.

Low, W. F. and Lee, G. H. Robust e-nerf: Nerf from sparse & noisy events under non-uniform motion. In *ICCV*, pp. 18335–18346, 2023.

Metzer, G., Richardson, E., Patashnik, O., Giryes, R., and Cohen-Or, D. Latent-nerf for shape-guided generation of 3d shapes and textures. In *CVPR*, pp. 12663–12673, 2023.

Mildenhall, B., Srinivasan, P. P., Tancik, M., Barron, J. T., Ramamoorthi, R., and Ng, R. Nerf: Representing scenes

as neural radiance fields for view synthesis. In *ECCV*, 2020.

Po, R. and Wetzstein, G. Compositional 3d scene generation using locally conditioned diffusion. *arXiv preprint arXiv:2303.12218*, 2023.

Poole, B., Jain, A., Barron, J. T., and Mildenhall, B. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv*, 2022.

Raj, A., Kaza, S., Poole, B., Niemeyer, M., Ruiz, N., Mildenhall, B., Zada, S., Aberman, K., Rubinstein, M., Barron, J., et al. Dreambooth3d: Subject-driven text-to-3d generation. *arXiv preprint arXiv:2303.13508*, 2023.

Ren, J., He, C., Liu, L., Chen, J., Wang, Y., Song, Y., Li, J., Xue, T., Hu, S., Chen, T., et al. Make-a-character: High quality text-to-3d character generation within minutes. *arXiv preprint arXiv:2312.15430*, 2023.

Shi, Y., Wang, P., Ye, J., Long, M., Li, K., and Yang, X. Mvdream: Multi-view diffusion for 3d generation. *arXiv preprint arXiv:2308.16512*, 2023.

Sun, C., Han, J., Deng, W., Wang, X., Qin, Z., and Gould, S. 3d-gpt: Procedural 3d modeling with large language models. *arXiv preprint arXiv:2310.12945*, 2023.

Tang, J., Ren, J., Zhou, H., Liu, Z., and Zeng, G. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. *arXiv preprint arXiv:2309.16653*, 2023.

Vilesov, A., Chari, P., and Kadambi, A. Cg3d: Compositional generation for text-to-3d via gaussian splatting. *arXiv preprint arXiv:2311.17907*, 2023.

Wang, H., Du, X., Li, J., Yeh, R. A., and Shakhnarovich, G. Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. In *CVPR*, pp. 12619–12629, 2023a.

Wang, Z., Lu, C., Wang, Y., Bao, F., Li, C., Su, H., and Zhu, J. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *arXiv preprint arXiv:2305.16213*, 2023b.

Wen, Z., Liu, Z., Sridhar, S., and Fu, R. Anyhome: Open-vocabulary generation of structured and textured 3d homes. *arXiv preprint arXiv:2312.06644*, 2023.

Xu, J., Wang, X., Cheng, W., Cao, Y.-P., Shan, Y., Qie, X., and Gao, S. 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*, pp. 20908–20918, 2023.

Yang, Y., Sun, F.-Y., Weihs, L., VanderBilt, E., Herrasti, A., Han, W., Wu, J., Haber, N., Krishna, R., Liu, L., et al. Holodeck: Language guided generation of 3d embodied ai environments. In *CVPR*, volume 30, pp. 20–25, 2024.

Yi, T., Fang, J., Wu, G., Xie, L., Zhang, X., Liu, W., Tian, Q., and Wang, X. Gaussiandreamer: Fast generation from text to 3d gaussian splatting with point cloud priors. *arXiv preprint arXiv:2310.08529*, 2023.

Zhang, L., Rao, A., and Agrawala, M. Adding conditional control to text-to-image diffusion models. In *ICCV*, pp. 3836–3847, 2023a.

Zhang, Q., Wang, C., Siarohin, A., Zhuang, P., Xu, Y., Yang, C., Lin, D., Zhou, B., Tulyakov, S., and Lee, H.-Y. Scenewiz3d: Towards text-guided 3d scene composition. *arXiv preprint arXiv:2312.08885*, 2023b.