Penetration-Free Compositional 3D Generation via Gaussian Surface Offset

Yilin Shao[1], Licheng Jiao[1] (✉), Lingling Li[1], Xu Liu[1], Fang Liu[1], Wenping Ma[1], Long Sun[1], and Jiaxuan Zhao[2]

[1] School of Artificial Intelligence, Xidian University, Xi'an, China
yilinshao@stu.xidian.edu.cn, lchjiao@mail.xidian.edu.cn
[2] Tianjin Research Institute for Water Transport Engineering, M.O.T

Abstract. Compositional text-to-3D generation aims to create multi-object scenes from text descriptions, yet existing approaches often suffer from severe inter-object penetrations during joint optimization, where entangled object representations lead to physically implausible scenes. To address this, we propose CompSAG to enable penetration-free and complete compositional 3D generation. Our key insight is a Gaussian surface offset (GSO) mechanism that mimics real-world repulsive forces to effectively resolve inter-object penetrations. The proposed GSO calculates offset values by aggregating repulsive forces from an object proxy surface, pushing apart the Gaussians in collision regions. Furthermore, during appearance optimization, we design a Gaussian surface rectification module to suppress the spiky geometric artifacts introduced by SDS. These are built upon a layout-guided scene initialization driven by a fine-tuned LLM, which provides precise spatial grounding from the text. Extensive experiments across diverse multi-object scenarios demonstrate that CompSAG produces complete, spatially coherent, and penetration-free 3D scenes, outperforming prior compositional generation baselines.

Keywords: 3D Compositional Generation · 3DGS

1 Introduction

Compositional text-to-3D scene generation plays a crucial role in advancing fields such as virtual reality [12, 55, 56], robotics [9], and spatial intelligence [49]. It challenges text-to-3D models to generate semantically coherent and spatially plausible multi-object scenes from complex prompts. However, this area remains underexplored, and existing methods commonly exhibit inter-object penetrations, causing entangled scene entities.

Although recent progress in text-to-3D has enabled efficient and realistic object-level generation [17, 47, 48, 50], these methods still struggle to generate compositional 3D scenes effectively [11]. Therefore, efforts have been made toward compositional text-to-3D generation, which requires parsing textual descriptions and inferring the underlying spatial relationships, *e.g.*, object interactions and positional cues, to ground semantically coherent and spatially plausible 3D scenes [7]. Existing approaches generally fall into two paradigms. The

first, guidance-space spatialization, conditions the diffusion model's latent space to encode more structured spatial priors [22, 26, 33]. These methods then optimize the 3D scene by a guidance framework, *e.g.* Score Distillation Sampling (SDS) [34, 45]. While efficient, such methods often struggle with guidance collapse when handling complex scenes. The second paradigm, object-compositional modeling, parses text into discrete semantic units and jointly optimizes these components by leveraging both individual and relational guidance distilled from diffusion models [1,7,11,12,21,52,55,56]. These methods mitigate common issues in compositional generation, such as missing objects or attribute confusion.

Among these methods, NeRF-based approaches have been making efforts to build semantically consistent 3D scenes through relational modeling [7, 11, 33]. However, the emergence of 3D Gaussian Splatting (3DGS) has reshaped this landscape [20]. Beyond its real-time rendering capability, 3DGS offers an explicit spatial representation that naturally supports flexible object manipulation [56]. Consequently, recent research in compositional text-to-3D generation has increasingly focused on representing scenes as collections of 3D Gaussians to support fine-grained compositional control [12, 44, 55, 56].

Nevertheless, 3DGS parameterizes the scene through volumetric density and color distributions. Therefore, without a clear boundary, surface interpenetration between objects is common in compositional generation, and existing methods fail to address this issue at the object-entity level, often resulting in entangled object representations. These limitations significantly hinder the development of high-fidelity compositional text-to-3D generation.

To address these challenges, we propose CompSAG (Compositional 3D generation with Surface-Aware Gaussians), a novel framework for penetration-free compositional 3D scene generation. CompSAG introduces a Gaussian Surface Offset (GSO) mechanism to calculate the repulsive force based on a maintained a proxy surface, pushing penetrated Gaussians toward a collision-free setting. Furthermore, to resolve the semantic entanglement and geometric artifacts common in SDS optimization, we introduce a geometry-constrained appearance refinement module. This module decomposes the scene into interacting subject-relation-object pairs optimized via an SDS loss, while simultaneously enforcing Gaussian surface alignment to suppress artifacts. Coupled with a layout-guided initialization driven by a fine-tuned Large Language Model (LLM), our framework ensures both semantic fidelity and spatial coherence. In summary, the main contributions of this paper are threefold:

- A novel Gaussian surface offset mechanism that enables penetration-free compositional generation. By leveraging a proxy surface to compute repulsive forces, it effectively resolves inter-object collisions and ensures physically plausible object arrangements.
- A geometry-constrained appearance refinement module that enhances visual and geometric fidelity. We exploit SDS supervision on decomposed subject-relation-object pairs to disentangle appearance and propose a surface rectification mechanism that optimizes the Gaussian properties to suppress coarse and spiky artifacts.

– A layout-guided initialization framework driven by a fine-tuned LLM on the Synthetic Visual Genome (SVG) dataset. This module provides precise 2D spatial grounding and depth-calibrated 3D priors, serving as a robust foundation for the subsequent optimization.

2 Related Work

2.1 Optimization-Based 3D Lifting

Early optimization-based text-to-3D methods such as DreamFusion [34] established the foundation of SDS-based generation, followed by extensions that improved geometric consistency [40], optimization stability [57], and visual fidelity [3,24,43,47]. Recently, 3D Gaussian Splatting (3DGS) [6,20,41,42,50] has emerged as an efficient explicit representation, enabling real-time rendering and compact geometry modeling. Subsequent works improved density control, relational reasoning, and feed-forward text-to-3D inference [8,17]. However, existing pipelines mainly focus on single-object generation and often fail to capture inter-object relations in complex scenes, resulting in object omissions and semantic leakage [11]. Our proposed CompSAG is a compositional 3D generation framework that produces spatially coherent and semantically consistent multi-object scenes.

2.2 Compositional 3D Generation

Compositional text-to-3D generation aims to synthesize scenes with multiple interacting objects that align with textual semantics. Existing methods fall into two major categories: guidance-space spatialization, which structures diffusion latents via hierarchical refinement, grounded attention, or bounding-box conditioning [22,26,33]; and object-compositional modeling, which assembles object-level representations using local–global NeRFs, hybrid implicit–explicit models, or scene-graph guidance [1,7,11,21,52]. Recent efforts extend compositional generation to 3DGS, leveraging its explicit and manipulable structure. LLM-based approaches predict object layouts [56], while CompGS and Layout-Your-3D integrate 2D-to-3D spatial priors for joint Gaussian optimization [12,55].

However, current methods largely overlook surface-level geometry, which is crucial for modeling proximity, contact, and collision. Our CompSAG addresses this gap by maintaining a surface for each object, providing smooth explicit surfaces that guide Gaussian arrangement and enable penetration-free compositional scene generation.

2.3 Neural Surface Representation

Surface representations provide accurate and continuous geometry modeling. Classical SDF-based methods such as NeuS [46] and UNISURF [30] reconstruct surfaces via differentiable rendering, while later works like Neuralangelo [23] and PermutoSDF [38] enhance fidelity with multi-resolution encodings.

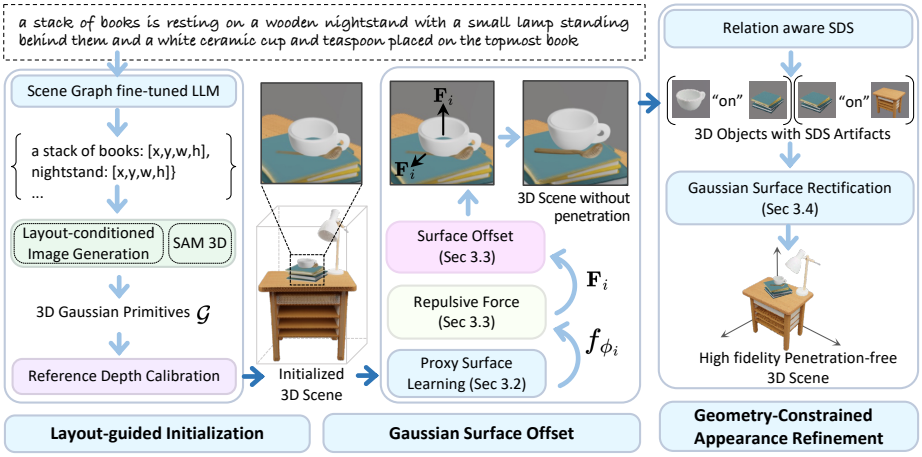

Fig. 1: Overall pipeline of CompSAG. Given a scene-level text prompt, our framework generates a penetration-free 3D scene in three main stages: (1) layout-guided initialization, where bounding-box-conditioned images are lifted into 3D Gaussian primitives; (2) Gaussian surface offset, which identifies colliding Gaussians via a learned proxy surface and applies aggregated repulsive forces to effectively resolve inter-object penetrations; and (3) geometry-constrained appearance refinement, which optimizes object appearance using a relation SDS loss on interacting pairs while enforcing surface alignment for geometric regularity.

Recent efforts combine SDFs with 3DGS to improve geometric regularity. SuGaR [14] aligns Gaussians to surfaces for mesh extraction, GSDF [51] jointly optimizes Gaussian and SDF fields, and GS-Pull [53] dynamically attracts Gaussians to SDF zero-level sets for consistent reconstruction.

Motivated by these advances, CompSAG incorporates per-object surface constraints into compositional text-to-3D generation. This regularizes Gaussian arrangements into smooth surfaces and enables surface-aware interaction constraints that prevent interpenetration, yielding physically plausible multi-object scenes.

3 Method

We present CompSAG, a framework for generating penetration-free compositional 3D scenes from textual descriptions. Our core contribution lies in a novel mechanism that mimics physical repulsive forces to resolve inter-object penetrations, thereby ensuring visually accurate and disentangled 3D object representations.

The pipeline proceeds in three stages. First, we initialize each object o_i with a set of 3D Gaussian primitives $\mathcal{G}_i = \{\boldsymbol{\theta}_i^{(k)}\}$ (Sec. 3.1) by lifting a bounding-box-conditioned generated image into 3D using SAM3D. Second, to resolve physical

conflicts, we identify the set of colliding Gaussians $\mathcal{C}_{i\to j}$ that penetrate adjacent objects o_j (Sec. 3.2) and aggregate a total repulsive force derived from a learned proxy surface to offset these geometric intersections (Sec. 3.3). Finally, we propose a geometry-constrained appearance refinement strategy, which distills relation-aware features from interacting pairs via SDS loss, while simultaneously constraining the 3D Gaussians to align with the proxy surface to suppress artifacts. An overview of the proposed framework is illustrated in Fig. 1.

3.1 Layout-Guided 3D Scene Initialization

Given a prompt describing M objects $\{o_i\}_{i=1}^{M}$, this step grounds them into accurately arranged 3D Gaussian primitives $\mathcal{G} = \{\mathcal{G}_i\}_{i=1}^{M}$.

Generate 2D layout and reference image. Optimizing a 3D scene directly from text with SDS is prone to multi-view inconsistency and Janus artifacts [25, 27]; following common practice in recent works [12,55], we anchor the generation on a concrete 2D visual blueprint rather than distilling from text alone.

We first synthesize a layout-conditioned 2D reference image that serves as the blueprint for 3D lifting. The 2D coordinates are parsed from the prompt by an LLM. However, off-the-shelf models often struggle with precise spatial grounding. To bridge this gap, we fine-tune a Qwen2.5-3B model [35] on the SVG dataset [31], which contains rich scene graphs with object bounding boxes. We prompt the LLM to output layouts in a structured CSS format, which facilitates the model's explicit reasoning on spatial relations and object scales [10]. Conditioned on these inferred object 2D bounding boxes and corresponding reference objects, box-conditioned image generation models [32, 54] synthesize a high-fidelity reference image $\mathcal{I}_{ref}$.

Initialize 3D Gaussian primitives. Then, we proceed to lift the 2D content into 3D Gaussians. The naive 3D scene is obtained via SAM3D [4]. Formally, the initialized 3D Gaussian primitives for each object o_i are denoted as:

$$\mathcal{G}_i = \{\mathcal{G}_i^{(k)}\}_{k=1}^{N_i} = \{\boldsymbol{\theta}_i^{(k)}\}_{k=1}^{N_i}, \quad \boldsymbol{\theta}_i^{(k)} = \{\mathbf{p}_k, \mathbf{s}_k, \mathbf{q}_k, \mathbf{c}_k, \alpha_k\},$$

where $\mathbf{p}_k \in \mathbb{R}^3$ denotes the mean position, $\mathbf{s}_k \in \mathbb{R}^3$ the scaling factors, and $\mathbf{q}_k \in \mathbb{R}^4$ the rotation quaternion, alongside the spherical harmonic coefficients $\mathbf{c}_k$ and opacity α_k for appearance modeling.

Reference depth calibration. However, a naive lifting often results in depth overlap. To resolve this, we perform a preliminary depth alignment to enforce consistency with the reference image $\mathcal{I}_{ref}$. While SAM3D places all objects in a shared camera frame, providing an analytic camera for $\mathcal{I}_{ref}$, this estimate is only approximate due to per-object pose errors. To refine the viewpoint, we render the coarse 3D scene from views sampled around the analytic camera and employ SuperGlue [39] to match feature points with $\mathcal{I}_{ref}$. Using the view with maximum matches as initialization, we establish correspondences between 2D keypoints and the projected 3D Gaussians, and refine the camera extrinsics $(\mathbf{R}, \mathbf{t})$ via Perspective-n-Point (PnP).

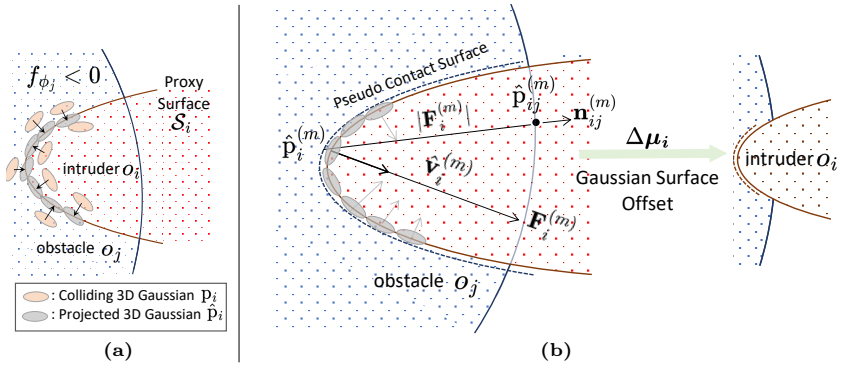

Fig. 2: Illustration of the Gaussian Surface Offset process. (a) Locating penetrating Gaussians: Colliding Gaussians (orange) of the intruder O_i are mapped to their projections (gray) onto the proxy surface $\mathcal{S}$. (b) Gaussian surface offset: Repulsive forces are aggregated along the pseudo-contact surface to drive a global position update $\Delta\boldsymbol{\mu}_i$.

With the refined pose, we render a depth map D_{ren} and align it with the monocular metric depth D_{ref} estimated from $\mathcal{I}_{ref}$. Prior to calibration, we rescale D_{ref} into the rendered unit by a per-object factor $s_{d,i}$, given by their median depth ratio within the masks. We then perform an object-wise depth alignment: utilizing the segmentation masks $\mathcal{M}_i^s$ obtained from SAM3D and the rendered object masks $\mathcal{M}_i^r$, we compute for each object o_i the depth discrepancy δ_i as the difference between their spatially-averaged depths:

$$\delta_i = \frac{1}{|\mathcal{M}_i^s|} \sum_{\mathbf{u}\in\mathcal{M}_i^s} s_{d,i} \cdot D_{ref}(\mathbf{u}) - \frac{1}{|\mathcal{M}_i^r|} \sum_{\mathbf{u}\in\mathcal{M}_i^r} D_{ren}(\mathbf{u}). \tag{1}$$

The Gaussian primitives belonging to object o_i are translated along the camera's viewing direction by δ_i, and uniformly scaled around their centroid by $s_{b,i} = \sqrt{|\mathcal{M}_i^s|/|\mathcal{M}_i^r|}$. Finally, to establish a canonical world frame, we designate one structural object as the mount and rotate the entire scene so that its canonical $+Z$ axis aligns with the world $+Z$ axis; the scene is then translated to be centered at the origin and uniformly rescaled into the $[-1, 1]$ cube. This explicit depth calibration ensures that the initialized 3D layout respects the visual depth order of the reference blueprint, providing a collision-reduced starting point for subsequent optimization.

3.2 Locating Penetrating Gaussians via Proxy Surface

Prior collision losses reduce inter-object overlap by constraining the distances between the interior points of objects [11, 55], which only decreases the number of intersecting points without locating where the collision actually occurs. Explicitly localizing the colliding regions from the object surface is instead more

effective [2]. Following this insight, in this step we localize the penetrating Gaussian primitives $\mathcal{C}_i$ within the object $\mathcal{G}_i$ and map them to their rectified locations on the collision proxy surface.

Proxy surface learning. Since 3D Gaussian primitives represent a discrete volumetric field without explicit boundaries, directly computing intersection forces is ill-posed. To address this, we propose to maintain a continuous Signed Distance Function (SDF) as a surface proxy, which facilitates precise collision detection and contact point estimation.

For each object o_i, we characterize its geometry by learning an object-specific SDF, parameterized by a lightweight MLP $f_{\phi_i} : \mathbb{R}^3 \to \mathbb{R}$. To supervise this network efficiently, we utilize the mesh extracted from SAM3D as a geometric prior, which undergoes the same transformation as in Sec. 3.1 to align with the object's 3D Gaussian surface in the scene coordinate frame. Specifically, we sample point clouds $\mathbf{x}$ near the mesh surface and train f_{ϕ_i} to minimize the Eikonal-regularized [13] geometric loss:

$$\mathcal{L}_{\text{SDF}} = \sum_{\mathbf{x}\in\Omega} |f_{\phi_i}(\mathbf{x}) - d_{GT}(\mathbf{x})| + \lambda(\|\nabla_{\mathbf{x}} f_{\phi_i}(\mathbf{x})\|_2 - 1)^2, \tag{2}$$

where d_{GT} denotes the ground-truth signed distance from the mesh. The zero-level set $\mathcal{S}_i = \{\mathbf{x} \mid f_{\phi_i}(\mathbf{x}) = 0\}$ implicitly defines the collision proxy surface.

Colliding Gaussian identification. With the learned proxy surfaces, we can now mathematically define the inter-object penetration. For a pair of objects, intruder o_i and obstacle o_j, we identify the colliding Gaussian set $\mathcal{C}_{i\to j}$, which consists of the primitives from o_i that have intruded into the interior of o_j. Formally, this is defined as:

$$\mathcal{C}_{i\to j} = \left\{ \mathbf{p}_i^{(m)} \in \mathcal{G}_i \mid f_{\phi_j}(\mathbf{p}_i^{(m)}) < 0 \right\}, \tag{3}$$

where $\mathbf{p}_i^{(m)}$ represents the mean position of the m-th penetrating Gaussian primitive in object o_i. The condition $f_{\phi_j} < 0$ explicitly identifies the Gaussians located inside the boundary of object o_j.

Projecting onto proxy surface. Due to the unstructured distribution and inherent high-frequency noise of 3D Gaussians, calculating forces directly on the raw positions $\mathbf{p}_i^{(m)}$ can be unstable. Therefore, to establish a reliable geometric foundation for force computation, we map each colliding Gaussian to its orthogonal projection on the proxy surface, as conceptually illustrated in Fig. 2(a). Specifically, we compute the projected colliding point $\hat{\mathbf{p}}_i^{(m)}$ by moving the Gaussian center along the surface normal direction [53]:

$$\hat{\mathbf{p}}_i^{(m)} = \mathbf{p}_i^{(m)} - f_{\phi_i}(\mathbf{p}_i^{(m)}) \cdot \frac{\nabla f_{\phi_i}(\mathbf{p}_i^{(m)})}{\|\nabla f_{\phi_i}(\mathbf{p}_i^{(m)})\|_2}. \tag{4}$$

This projection rectifies the noisy Gaussian onto the smooth zero-level set of the collision target, as illustrated in Fig. 2(a), ensuring that the subsequent repulsive force is geometrically stable.

3.3 Gaussian Surface Offset (GSO)

Having located the penetrating Gaussians and projected them onto the proxy surface, we now resolve the penetration by emulating a repulsion force. We first aggregate the per-primitive repulsive forces within the penetration region into a single object-level force, and then apply it as a momentum-based offset that pushes the intruder out of the obstacle.

Repulsive force aggregation. We first quantify the magnitude of the repulsive force for each colliding primitive. As illustrated by the dashed boundary in Fig. 2(b), we treat the penetration region as a pseudo-contact surface. For each projected colliding point $\mathbf{p}_i^{(m)} \in \mathcal{C}_{i\to j}$ with projection $\hat{\mathbf{p}}_i^{(m)}$, let $\hat{\mathbf{p}}_{ij}^{(m)}$ denote its closest corresponding point on the proxy surface of the obstacle o_j. The force magnitude is calculated as the projection of the penetration vector $\left(\hat{\mathbf{p}}_i^{(m)} - \hat{\mathbf{p}}_{ij}^{(m)}\right)$ onto the surface normal:

$$|\mathbf{F}_i^{(m)}| = -\left(\hat{\mathbf{p}}_i^{(m)} - \hat{\mathbf{p}}_{ij}^{(m)}\right) \cdot \mathbf{n}_{ij}^{(m)} = \left| f_{\phi_j}\left(\hat{\mathbf{p}}_i^{(m)}\right) \right|. \tag{5}$$

where $\mathbf{n}_{ij}^{(m)}$ denotes the outward surface normal at $\hat{\mathbf{p}}_{ij}^{(m)}$, computed as $\mathbf{n}_{ij}^{(m)} = \nabla f_{\phi_j}(\hat{\mathbf{p}}_{ij}^{(m)}) / \|\nabla f_{\phi_j}(\hat{\mathbf{p}}_{ij}^{(m)})\|$. Conveniently, by leveraging the learned SDF f_{ϕ_j} of the obstacle, this geometric projection is directly equivalent to the absolute SDF value. This formulation also ensures that deeper penetrations intuitively induce stronger repulsive forces.

Then, the force direction for $\hat{\mathbf{p}}_i^{(m)}$ is derived with the inward normal of the intruder object o_i, computed as $\mathbf{v}_i^{(m)} = -\mathbf{n}_i^{(m)}$. To filter out forces that would push the intruder deeper, we retain only those aligned with the inter-object separation axis $\mathbf{d}_{ij} = \boldsymbol{\mu}_i - \boldsymbol{\mu}_j$, where $\boldsymbol{\mu}_i$ and $\boldsymbol{\mu}_j$ denote the centroids of the intruder and obstacle. The total repulsive force $\mathbf{F}_i$ acting on object o_i from the obstacle o_j is then obtained by averaging these directionally-filtered per-primitive forces:

$$\mathbf{F}_i = \frac{1}{|\mathcal{C}_{i\to j}|} \sum_{\mathbf{p}_i^{(m)} \in \mathcal{C}_{i\to j}} \mathbb{1}\left(\mathbf{v}_i^{(m)} \cdot \mathbf{d}_{ij} > 0\right) |\mathbf{F}_i^{(m)}|\, \mathbf{v}_i^{(m)}, \tag{6}$$

where $\mathbb{1}(\cdot)$ is the indicator function that zeros out forces inconsistent with the separation axis.

Momentum-based Surface Offset. Finally, we apply the aggregated force to offset the Gaussian surface of o_i. To ensure optimization stability, we emulate a momentum-based optimizer coupled with an ease-in-out learning rate schedule $\gamma(t)$. Specifically, we maintain a momentum buffer to compute the update vector $\Delta\boldsymbol{\mu}_i$ by tracking the previous force direction: $\Delta\boldsymbol{\mu}_i = \beta\, \mathbf{F}_i^{\text{prev}} + (1-\beta)\, \mathbf{F}_i$, where β controls the momentum contribution. The Gaussian positions are subsequently updated as $\boldsymbol{\mu}_i \leftarrow \boldsymbol{\mu}_i + \gamma(t)\, \Delta\boldsymbol{\mu}_i$. As the depth-proportional force decays to zero once penetration is resolved, the step is inherently bounded and stays stable under $\gamma(t)$ without extra clipping. Additionally, we explicitly zero out the z-component of forces if an object contacts the ground plane and suppress boundary-normal forces near scene borders.

3.4 Geometry-constrained Appearance Refinement

This stage aims to optimize the disentangled 3D Gaussians $\{\mathcal{G}_i^{(k)}\}$ for high-fidelity rendering while regularizing geometric parameters (position and rotation) to mitigate artifacts. To this end, we propose a geometry-constrained appearance refinement strategy. This approach first employs relation-aware SDS distillation on interacting object pairs to enhance appearance. However, directly optimizing Gaussian with SDS can produce concavities and spiky artifacts [44]. Therefore, we explicitly rectify the Gaussian primitives to suppress the irregular geometry.

Relation-aware appearance optimization. We employ the fine-tuned LLM (from Sec. 3.1) to parse the scene prompt into a scene graph, extracting interacting subject-relation-object triplets. For each interacting pair, we isolate it, center it at the origin, and render from uniformly sampled viewpoints π to synthesize 2D images $\mathcal{I}_\pi$. We then optimize the Gaussian parameters $\boldsymbol{\theta}$ of these isolated pairs via a relation-aware SDS loss using a pre-trained 2D diffusion model:

$$\nabla_{\boldsymbol{\theta}} \mathcal{L}_{\text{SDS}_{rel}} = \mathbb{E}_{t,\epsilon,\pi} \left[w(t) \left(\epsilon_\phi(\mathbf{z}_t; y_{rel}, t) - \epsilon \right) \frac{\partial \mathcal{I}_\pi}{\partial \boldsymbol{\theta}} \right], \tag{7}$$

where y_{rel} is the text prompt of the interacting pair; the remaining terms $\mathbf{z}_t$, ϵ_ϕ, and $w(t)$ follow the standard SDS formulation [34]. Empirically, we observe that distilling interacting pairs yields a more stable appearance compared to optimizing the entire scene simultaneously.

Gaussian surface rectification (GSR) builds on the surface-alignment techniques of SuGaR [14] and GS-Pull [53], enforcing a tight coupling between the explicit 3D Gaussians and the implicit proxy surface to alleviate the artifacts brought by SDS. We align these distilled Gaussians to the proxy surface in two complementary ways. First, we optimize the Gaussian positions to tightly adhere to the implicit boundary: Eq. 4 is repurposed as an optimization objective, where we freeze f_{ϕ_i} and minimize the Gaussian centers with respect to the proxy surface $f_{\phi_i}(\mathbf{p}_i^{(k)})$, denoted as $\mathcal{L}_{GS}^i$. Second, we mitigate the spike-like artifacts by preventing the major axes of 3D Gaussians from protruding from the object boundaries, encouraging consistency between the Gaussian normal $\mathbf{n}_{p_i}^{(k)}$ and the surface normal. $\mathbf{n}_{p_i}^{(k)}$ is defined as the column of the rotation matrix $R(\mathbf{q})$ associated with the smallest scaling factor, *i.e.*, the direction in which the Gaussian is thinnest. We then optimize the rotation $\mathbf{q}$ via:

$$\mathcal{L}_{\text{normal}}^i = \sum_{k=1}^{N_i} \left(\left\| \frac{\nabla f_{\phi_i}(\mathbf{p}_i^{(k)})}{\|\nabla f_{\phi_i}(\mathbf{p}_i^{(k)})\|} \cdot \mathbf{n}_{p_i}^{(k)} \right\|_2^2 - 1 \right). \tag{8}$$

Since the alignment is measured by the squared cosine, the loss is invariant to whether $\mathbf{n}_{p_i}^{(k)}$ points inward or outward.

Both terms rely on a proxy surface that faithfully tracks the current Gaussians. We keep f_{ϕ_i} (from Sec. 3.3) tight to the evolving Gaussians using the surface-tracking loss $\mathcal{L}_{\text{SDF}_g}^i$ from [14, 53], which minimizes the discrepancy between its zero-level set and the explicit Gaussian centers.

In practice, we employ an alternating optimization scheme between the relation-aware SDS and the Gaussian surface rectification. This iterative process not only progressively refines the object geometry to achieve finer surface details, but also provides a strong structural regularization that inherently stabilizes the SDS-driven appearance distillation.

4 Experiment

4.1 Implementation Details

Training setup. At initialization, we constrain the total number of 3D Gaussians based on the scene complexity, downsampling them to a maximum of 100k, 150k, and 200k points for scenes with 2, 3, and 5 objects, respectively. During the densification process in subsequent optimization, the overall Gaussian count is strictly capped at 800k. The surface offset runs for 1,600 steps and terminates the repulsive force computation once the fraction of colliding Gaussians drops below a threshold of 0.03. We set the momentum coefficient to $\beta = 0.3$.

During the appearance refinement, we adopt Stable Diffusion v2.1 [37] as the generative prior with a guidance scale of 100. The maximum diffusion timestep progressively anneals from 0.98 to 0.5. The loss weights are empirically set as $\mathcal{L}_{\text{SDS}_{rel}}$: 0.01, $\mathcal{L}_{\text{SDF}_g}$: 10.0, $\mathcal{L}_{\text{normal}}$: 0.1, and $\mathcal{L}_{\text{GS}}$: 10.0. The refinement phase is executed for 3 alternating cycles, with each cycle consisting of 400 steps of relation-aware SDS, 200 steps of SDF refinement, and 400 steps of Gaussian rectification. In the first cycle, the relation-aware SDS is preceded by a 500-step object-level optimization. Our CompSAG framework is built upon ThreeStudio [15], and all models are trained on a multi-GPU server equipped with four NVIDIA RTX 3090 GPUs.

Table 1: Quantitative comparison between CompSAG and baseline methods. The best scores are shown in **bold**, and the second-best are underlined.

Method	3D Rep.	Time	CLIP	B-VQA	GPT-CoT	CVR
ProlificDreamer [47]	NeRF	240min†	28.91	42.16	27.27	N/A
MVDream [40]	NeRF	30min†	31.05	28.29	30.98	N/A
GaussianDreamer [50]	3DGS	15min†	28.24	31.87	31.06	N/A
Set-the-Scene [7]	NeRF	50min	29.43	41.20	37.74	0.265
GraphDreamer [11]	NeRF	120min	32.11	40.29	45.69	0.278
CG3D [44]	3DGS	60min†	31.56	46.19	43.25	0.147
GALA3D [56]	3DGS	60min	32.63	50.48	52.03	0.205
CompGS [12]	3DGS	70min	31.67	53.62	51.23	0.186
Layout-your-3D [55]	3DGS	30min	31.79	52.57	50.93	0.103
CompSAG (Q)	3DGS	30min	32.72	56.17	53.69	0.042
CompSAG (Q^*)	3DGS	30min	<u>33.64</u>	<u>57.28</u>	<u>54.38</u>	**0.034**
CompSAG ($Q^* + U$)	3DGS	30min	**33.87**	**58.11**	**56.64**	<u>0.038</u>

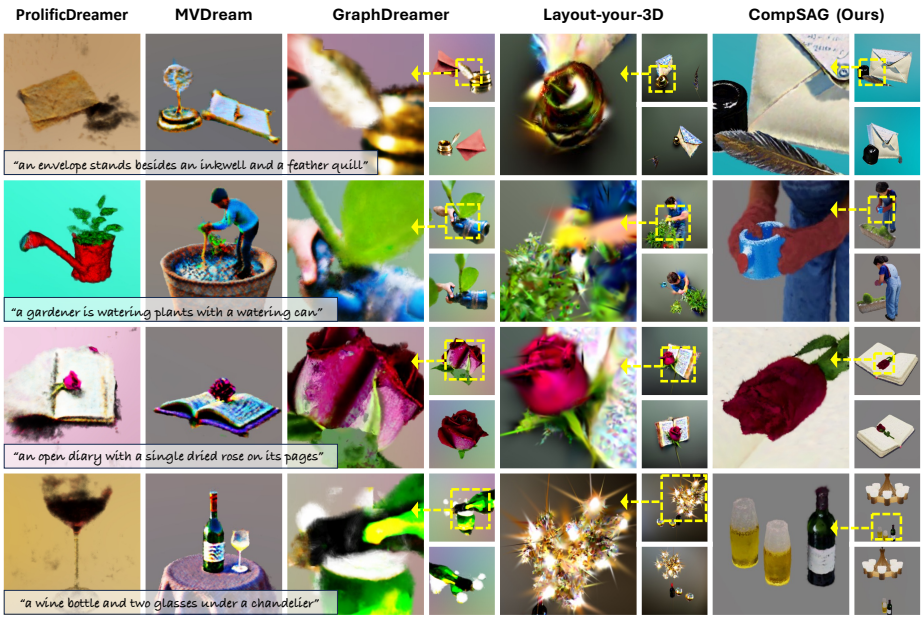

Fig. 3: Qualitative comparison with ProlificDreamer, MVDream, GraphDreamer, Layout-your-3D and our CompSAG.

Layout LLM fine-tuning. The Qwen2.5-3B model is fine-tuned using the parameter-efficient LoRA ($rank = 64$) implemented via the HuggingFace PEFT library [28]. The training is conducted for 3 epochs with a batch size of 4. To construct a high-quality training corpus from the SVG dataset, we explicitly filter out background entities and invalid semantic labels, strictly retaining relation data that solely involves tangible objects with explicit physical shapes. By utilizing these refined subject-predicate-object triplets, we compose scene descriptions containing 2 to 5 objects, yielding approximately 5k high-quality synthetic scene instructions.

Evaluation Metrics. Our evaluation incorporates a CLIP Score [36], a B-VQA and a GPT-CoT [18] that specifically evaluate the compositional correspondence and a Colliding Volume Ratio (CVR) to quantify the ability to resolve inter-object penetrations. CVR measures penetration by assigning each object an SDF, discretizing the scene into a 256^3 voxel grid, and reporting the fraction of the occupied volume that falls inside two or more objects. Note that a small residual CVR reflects legitimate surface contact rather than penetration, as adjacent objects share boundary voxels at finite resolution; thus "penetration-free" denotes the absence of *perceptible* interpenetration.

Table 2: Comparison result on T^{3}Bench Multiple Objects. We highlight the **best**, second best, and third best results with background colors (dark to light).

Method	Quality$^\uparrow$	Alignment$^\uparrow$	Average$^\uparrow$	User Study$^\uparrow$
Fantasia3D [3]	22.7	14.3	18.5	2.42
LatentNeRF [29]	21.7	19.5	20.6	3.17
Magic3D [24]	26.6	24.8	25.7	3.46
ProlificDreamer [47]	45.7	25.8	35.8	3.65
DreamGaussian [42]	12.3	9.5	10.9	2.54
GaussianDreamer [50]	38.8	30.2	34.5	3.21
VP3D [5]	49.1	31.5	40.3	2.93
Set-the-Scene [7]	20.8	29.9	25.4	3.82
CompGS [12]	54.2	37.9	46.1	4.15
Ours	**55.5**	**41.9**	**48.7**	**4.42**

4.2 Comparison with State-of-the-art

Quantitative Comparison. Table 1 presents the quantitative comparisons against state-of-the-art methods, categorizing them into non-compositional text-to-3D approaches (ProlificDreamer, MVDream, and GaussianDreamer) and compositional frameworks (Set-the-Scene, GraphDreamer, CG3D, GALA3D, CompGS, and Layout-your-3D). For a fair comparison, the Layout-your-3D baseline is re-initialized with the same SAM3D lifter as ours. CompSAG consistently achieves state-of-the-art performance across all quality metrics.

While high CLIP scores indicate strong text-to-latent alignment, B-VQA more reliably assesses object integrity and appearance. CompSAG surpasses all baselines in B-VQA because our disentangled 3D representation ensures an interference-free optimization process. For GPT-CoT, our precise layout initialization and relation-aware optimization secure the top score.

Notably, in terms of the CVR metric, CompSAG outperforms all methods. Unlike Layout-your-3D, which relies on coarse bounding spheres that fail to resolve fine-grained penetrations, our approach strictly resolves boundary collisions. The other compositional 3DGS baselines (CG3D, GALA3D, and CompGS) likewise lack surface-level penetration handling and thus retain a higher CVR. Furthermore, our fine-tuned layout LLM (Q^*) consistently improves upon the baseline (Q) and approaches human-adjusted layout performance ($Q^* + U$). Finally, evaluations on the T^{3}Bench multi-object dataset [16] and user studies (Table 2) confirm CompSAG's improved quality and alignment in grounding spatially coherent, semantically accurate compositions.

Qualitative Results. As shown in Fig. 3, non-compositional methods (ProlificDreamer, MVDream) frequently suffer from object omission, semantic drift, and unrealistic scaling—such as ProlificDreamer blending the plant into the watering can, or MVDream generating a disproportionately large flower pot.

While compositional approaches (GraphDreamer, Layout-your-3D) better preserve semantic completeness via individual object optimization, they strug-

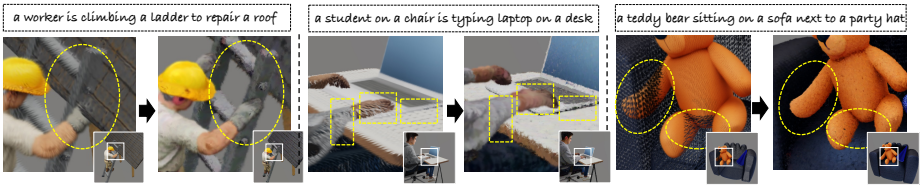

Fig. 4: Visualization of Gaussian surface offset for penetration-free scene composition.

gle with severe inter-object penetrations. For instance, GraphDreamer produces collisions between the quill/inkwell and plant/watering can, whereas Layout-your-3D buries the rose into the diary. Beyond collisions, these baselines lack explicit geometric constraints, leading to intrinsic flaws. GraphDreamer often yields fragmented or disproportionate structures (*e.g.*, a fragmented gardener, an ill-defined chandelier). Layout-your-3D relies on coarse bounding spheres, causing distorted spatial arrangements that push objects to unnaturally distant positions or introduce excessive rotation (*e.g.*, the tilted chandelier), without fully resolving intersections.

CompSAG leverages precise layout initialization to guarantee the distinct presence of every entity, while our surface-aware mechanism ensures a physically plausible, penetration-free optimization process.

Runtime Analysis. Table 1 reports the average generation time per scene, where † marks timings cited from the original papers or the T^{3}Bench report. Among compositional methods, CompSAG is the most efficient while maintaining high fidelity. The pairwise collision detection is $O(M^2)$, but its overhead is negligible for typical scenes ($M \leq 5$) given our forward-only operations; and since the total number of Gaussians is capped, the SDS stage does not grow linearly with M. Increasing the object count from 2 to 5 thus raises the runtime only moderately, from 27 to 34 minutes.

A stage-wise breakdown shows that collision resolution is cheap: the text-to-image, SAM3D lifting, RDC, and GSO stages take ∼8 minutes in total, leaving ∼22 minutes for SDS-based appearance refinement. This yields two operating points: a *fast mode* that stops after GSO yet already produces a penetration-free layout, and a *high-quality mode* that adds the refinement to remove the

Table 3: Quantitative ablation study on Gaussian surface offset.

Config.	M=2	M=3	M=5	Ave.
Baseline	0.214	0.291	0.317	0.276
+ RDC	0.172	0.260	0.278	0.239
+ TCL [55]	0.085	0.109	0.113	0.103
+ GSO	**0.028**	**0.033**	**0.040**	**0.034**

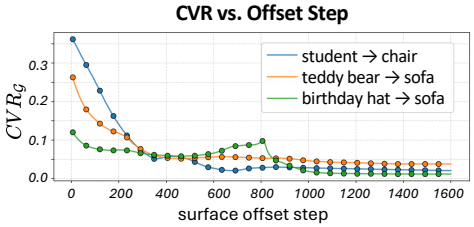

Fig. 5: Variation of the CVR$_\mathcal{G}$ with respect to the surface offset steps.

toy-like appearance of the feedforward initialization. It also clarifies our relation to feedforward single-image-to-scene generators: taking either SAM3D [4] (CVR 0.276) or MIDI [19] (CVR 0.321) as the feedforward initialization, CompSAG drives the CVR down to 0.038, a penetration-free gain we analyze in Sec. 4.3.

4.3 Components Analysis

Resolving interpenetration with GSO. As shown in Fig. 4, scenes initialized solely with layout priors exhibit severe collisions (*e.g.*, overlapping teddy bear and sofa, hands sinking into the keyboard, and a ladder intersecting the roof), which GSO effectively resolves into penetration-free layouts. Quantitatively (Table 3), the CVR confirms that while RDC mitigates early-stage spatial conflicts, GSO provides the most substantial penetration reduction. Furthermore, GSO consistently outperforms the Tolerant Collision Loss (TCL) [55], whose reliance on coarse bounding spheres fails to resolve fine-grained boundary collisions.

Fig. 5 plots the collision ratio during training, computed directly on the Gaussian point clouds ($\text{CVR}_{\mathcal{G}}$) for efficiency rather than on dense voxel grids. The steady decline confirms that GSO consistently drives objects apart; minor fluctuations reflect temporary intersections during joint optimization or the traversal of local geometric bumps, which our momentum-based optimization swiftly overcomes.

Smoother object surface with GSR. This experiment analyzes the effectiveness of GSR in resolving the geometric defects and spiky artifacts introduced by the SDS process. As illustrated in Fig. 6, GSR significantly regularizes the Gaussian primitives during optimization. The optimized Gaussian surfaces become noticeably more structured and well-aligned, alleviating the concavities and spiky artifacts.

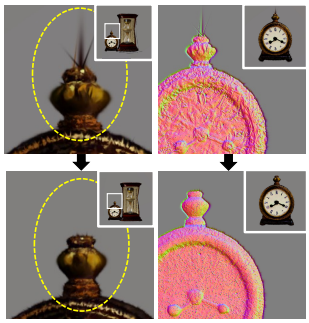

Fig. 6: Results visualization, w/o GSR (up) and w/ GSR (down).

Relation-aware SDS vs. Unstructured SDS. This experiment evaluates the contribution of the proposed relation-aware SDS to the appearance optimization of multi-object scenes. As shown in Fig. 7, we compare optimization with SDS_{rel} and the conventional unstructured SDS (SDS_u). SDS_u suffers from severe semantic leakage, where the wooden attribute of the desk leaks into the globe. In contrast, SDS_{rel} ensures disentangled appearance distillation, preserving the semantic completeness of each entity.

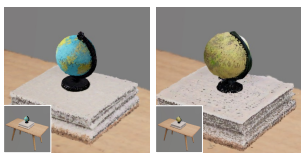

Fig. 7: Result w/ SDS_{rel} (left) and w/ SDS_u (right).

Ablation Summary. Finally, we evaluate the quantitative impact of each component on the B-VQA metric, as summarized in Table 4. The results confirm that: (1) the removal of GSO causes the largest drop, as it not only resolves the entangled scene but also provides clean

Table 4: Ablation results.

Setting	B-VQA$^{\uparrow}$
Full setting	**57.28**
– GSO	50.33
– GSR	52.19
– SDS$_{rel}$	54.47

object representations that would otherwise interfere during joint optimization; (2) the absence of GSR introduces surface irregularities and spiky artifacts, which adversely affect the perceptual fidelity; and (3) substituting SDS$_{rel}$ with unstructured supervision leads to semantic leakage and attribute confusion, resulting in decreased appearance accuracy.

5 Conclusion

In this paper, we present CompSAG, a novel framework that fundamentally addresses the persistent challenge of inter-object penetration in compositional text-to-3D generation. With the help of explicit surface-aware modeling, our approach introduces a Gaussian Surface Offset (GSO) mechanism. This mechanism mimics real-world repulsive forces to resolve spatial collisions. Furthermore, our pipeline is reinforced by a fine-tuned layout LLM and explicit depth calibration, which jointly ensure accurate spatial grounding and a collision-reduced 3D initialization. Coupled with Gaussian Surface Rectification (GSR) to suppress geometric artifacts caused by SDS, CompSAG demonstrates state-of-the-art semantic fidelity and scene integrity.

6 Limitations and Future Work

While CompSAG significantly advances the generation of multi-object scenes, several limitations remain and point toward promising future directions.

Scalability in complex scenes. To bound memory and computation, we cap the total number of 3D Gaussian primitives; yet this hard cap limits per-object fidelity in densely populated scenes. Multi-scale or adaptive sparse Gaussian representations are a promising remedy for dense, large-scale compositions.

Degenerate collision configurations. Since GSO aggregates repulsive directions from the intruder's surface normals, certain symmetric configurations become ill-posed. For instance, when two symmetric objects penetrate a shared middle object from opposite sides, their forces on the middle object cancel out, leaving its penetration unresolved. Such boundary cases lie beyond the reach of our normal-based offset.

Rigid-body assumption. CompSAG treats all entities as rigid objects, so it may leave objects floating without support and cannot model non-rigid deformations (*e.g.*, a soft object compressing under squeezing). A natural remedy is to hand the collision-free objects to a differentiable physics solver, which would settle them under gravity and simulate elastoplastic deformations. This further positions CompSAG's penetration-free geometry as a strong initialization for dynamic 4D generation, since initial interpenetrations would otherwise destabilize Lagrangian solvers such as the Material Point Method (MPM) and Finite Element Method (FEM).

References

1. Bai, H., Lyu, Y., Jiang, L., Li, S., Lu, H., Lin, X., Wang, L.: Componerf: Text-guided multi-object compositional nerf with editable 3d scene layout. arXiv preprint arXiv:2303.13843 (2023)
2. Chen, A.H., Hsu, J., Liu, Z., Macklin, M., Yang, Y., Yuksel, C.: Offset geometric contact. ACM Transactions on Graphics **44**(4), 1–21 (2025)
3. Chen, R., Chen, Y., Jiao, N., Jia, K.: Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 22246–22256 (2023)
4. Chen, X., Chu, F.J., Gleize, P., Liang, K.J., Sax, A., Tang, H., Wang, W., Guo, M., Hardin, T., Li, X., et al.: Sam 3d: 3dfy anything in images. arXiv preprint arXiv:2511.16624 (2025)
5. Chen, Y., Pan, Y., Yang, H., Yao, T., Mei, T.: Vp3d: Unleashing 2d visual prompt for text-to-3d generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4896–4905 (2024)
6. Chen, Z., Wang, F., Wang, Y., Liu, H.: Text-to-3d using gaussian splatting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21401–21412 (2024)
7. Cohen-Bar, D., Richardson, E., Metzer, G., Giryes, R., Cohen-Or, D.: Set-the-scene: Global-local training for generating controllable nerf scenes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2920–2929 (2023)
8. Di, D., Yang, J., Luo, C., Xue, Z., Chen, W., Yang, X., Gao, Y.: Hyper-3dg: Text-to-3d gaussian generation via hypergraph. International Journal of Computer Vision **133**(5), 2886–2909 (2025)
9. Driess, D., Huang, Z., Li, Y., Tedrake, R., Toussaint, M.: Learning multi-object dynamics with compositional neural radiance fields. In: Proceedings of the Conference on Robot Learning. pp. 1755–1768 (2023)
10. Feng, W., Zhu, W., Fu, T.j., Jampani, V., Akula, A., He, X., Basu, S., Wang, X.E., Wang, W.Y.: Layoutgpt: Compositional visual planning and generation with large language models. In: Advances in Neural Information Processing Systems. vol. 36, pp. 18225–18250 (2023)
11. Gao, G., Liu, W., Chen, A., Geiger, A., Schölkopf, B.: Graphdreamer: Compositional 3d scene synthesis from scene graphs. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21295–21304 (2024)
12. Ge, C., Xu, C., Ji, Y., Peng, C., Tomizuka, M., Luo, P., Ding, M., Jampani, V., Zhan, W.: Compgs: Unleashing 2d compositionality for compositional text-to-3d via dynamically optimizing 3d gaussians. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18509–18520 (2025)
13. Gropp, A., Yariv, L., Haim, N., Atzmon, M., Lipman, Y.: Implicit geometric regularization for learning shapes. In: Proceedings of the International Conference on Machine Learning. pp. 3789–3799 (2020)
14. Guédon, A., Lepetit, V.: Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5354–5363 (2024)
15. Guo, Y.C., Liu, Y.T., Shao, R., Laforte, C., Voleti, V., Luo, G., Chen, C.H., Zou, Z.X., Wang, C., Cao, Y.P., Zhang, S.H.: threestudio: A unified framework for 3d content generation. `https://github.com/threestudio-project/threestudio` (2023)

16. He, Y., Bai, Y., Lin, M., Zhao, W., Hu, Y., Sheng, J., Yi, R., Li, J., Liu, Y.J.: T^3bench: Benchmarking current progress in text-to-3d generation. arXiv preprint arXiv:2310.02977 (2023)
17. Hu, H., Yin, T., Luan, F., Hu, Y., Tan, H., Xu, Z., Bi, S., Tulsiani, S., Zhang, K.: Turbo3d: Ultra-fast text-to-3d generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 23668–23678 (2025)
18. Huang, K., Sun, K., Xie, E., Li, Z., Liu, X.: T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation. In: Advances in Neural Information Processing Systems. vol. 36, pp. 78723–78747 (2023)
19. Huang, Z., Guo, Y.C., An, X., Yang, Y., Li, Y., Zou, Z.X., Liang, D., Liu, X., Cao, Y.P., Sheng, L.: Midi: Multi-instance diffusion for single image to 3d scene generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 23646–23657 (2025)
20. Kerbl, B., Kopanas, G., Leimkuehler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM Transactions on Graphics **42**(4), 1–14 (2023)
21. Li, H., Shi, H., Zhang, W., Wu, W., Liao, Y., Wang, L., Lee, L.h., Zhou, P.Y.: Dreamscene: 3d gaussian-based text-to-3d scene generation via formation pattern sampling. In: European Conference on Computer Vision. pp. 214–230 (2024)
22. Li, X., Mo, J., Wang, Y., Parameshwara, C., Fei, X., Swaminathan, A., Taylor, C., Tu, Z., Favaro, P., Soatto, S.: Grounded compositional and diverse text-to-3d with pretrained multi-view diffusion model. arXiv preprint arXiv:2404.18065 (2024)
23. Li, Z., Müller, T., Evans, A., Taylor, R.H., Unberath, M., Liu, M.Y., Lin, C.H.: Neuralangelo: High-fidelity neural surface reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8456–8465 (2023)
24. Lin, C.H., Gao, J., Tang, L., Takikawa, T., Zeng, X., Huang, X., Kreis, K., Fidler, S., Liu, M.Y., Lin, T.Y.: Magic3d: High-resolution text-to-3d content creation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 300–309 (2023)
25. Liu, M., Shi, R., Chen, L., Zhang, Z., Xu, C., Wei, X., Chen, H., Zeng, C., Gu, J., Su, H.: One-2-3-45++: Fast single image to 3d objects with consistent multi-view generation and 3d diffusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10072–10083 (2024)
26. Liu, Y., Li, X., Li, X., Qi, L., Li, C., Yang, M.H.: Pyramid diffusion for fine 3d large scene generation. In: European Conference on Computer Vision. pp. 71–87. Springer (2024)
27. Long, X., Guo, Y.C., Lin, C., Liu, Y., Dou, Z., Liu, L., Ma, Y., Zhang, S.H., Habermann, M., Theobalt, C., et al.: Wonder3d: Single image to 3d using cross-domain diffusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9970–9980 (2024)
28. Mangrulkar, S., Gugger, S., Debut, L., Belkada, Y., Paul, S., Bossan, B.: PEFT: State-of-the-art parameter-efficient fine-tuning methods. `https://github.com/huggingface/peft` (2022)
29. Metzer, G., Richardson, E., Patashnik, O., Giryes, R., Cohen-Or, D.: Latent-nerf for shape-guided generation of 3d shapes and textures. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12663–12673 (2023)
30. Oechsle, M., Peng, S., Geiger, A.: Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5589–5599 (2021)

31. Park, J.S., Ma, Z., Li, L., Zheng, C., Hsieh, C.Y., Lu, X., Chandu, K., Kong, Q., Kobori, N., Farhadi, A., et al.: Synthetic visual genome. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9073–9086 (2025)
32. Peng, F., Wu, J., Li, Y., Gao, T., Zhang, D., Fu, H.: Muse: Multi-subject unified synthesis via explicit layout semantic expansion. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15885–15895 (2025)
33. Po, R., Wetzstein, G.: Compositional 3d scene generation using locally conditioned diffusion. In: Proceedings of the International Conference on 3D Vision. pp. 651–663. IEEE (2024)
34. Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: Dreamfusion: Text-to-3d using 2d diffusion. In: International Conference on Learning Representations. Kigali, Rwanda (2023)
35. Qwen, :, Yang, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Li, C., Liu, D., Huang, F., Wei, H., Lin, H., Yang, J., Tu, J., Zhang, J., Yang, J., Yang, J., Zhou, J., Lin, J., Dang, K., Lu, K., Bao, K., Yang, K., Yu, L., Li, M., Xue, M., Zhang, P., Zhu, Q., Men, R., Lin, R., Li, T., Tang, T., Xia, T., Ren, X., Ren, X., Fan, Y., Su, Y., Zhang, Y., Wan, Y., Liu, Y., Cui, Z., Zhang, Z., Qiu, Z.: Qwen2.5 technical report (2025), https://arxiv.org/abs/2412.15115
36. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: Proceedings of the International Conference on Machine Learning. pp. 8748–8763 (2021)
37. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10684–10695 (2022)
38. Rosu, R.A., Behnke, S.: Permutosdf: Fast multi-view reconstruction with implicit surfaces using permutohedral lattices. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8466–8475 (2023)
39. Sarlin, P.E., DeTone, D., Malisiewicz, T., Rabinovich, A.: Superglue: Learning feature matching with graph neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4938–4947 (2020)
40. Shi, Y., Wang, P., Ye, J., Mai, L., Li, K., Yang, X.: Mvdream: Multi-view diffusion for 3d generation. In: International Conference on Learning Representations (2024)
41. Tang, J., Chen, Z., Chen, X., Wang, T., Zeng, G., Liu, Z.: Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In: European Conference on Computer Vision (2024)
42. Tang, J., Ren, J., Zhou, H., Liu, Z., Zeng, G.: Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. In: International Conference on Learning Representations (2024)
43. Tsalicoglou, C., Manhardt, F., Tonioni, A., Niemeyer, M., Tombari, F.: Textmesh: Generation of realistic 3d meshes from text prompts. In: Proceedings of the International Conference on 3D Vision. pp. 1554–1563 (2024)
44. Vilesov, A., Chari, P., Kadambi, A.: Cg3d: Compositional generation for text-to-3d via gaussian splatting. arXiv preprint arXiv:2311.17907 (2023)
45. Wang, H., Du, X., Li, J., Yeh, R.A., Shakhnarovich, G.: Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12619–12629 (2023)

46. Wang, P., Liu, L., Liu, Y., Theobalt, C., Komura, T., Wang, W.: Neus: learning neural implicit surfaces by volume rendering for multi-view reconstruction. In: Advances in Neural Information Processing Systems. pp. 27171–27183 (2021)
47. Wang, Z., Lu, C., Wang, Y., Bao, F., Li, C., Su, H., Zhu, J.: Prolificdreamer: high-fidelity and diverse text-to-3d generation with variational score distillation. In: Advances in Neural Information Processing Systems. pp. 8406–8441 (2023)
48. Xiang, J., Lv, Z., Xu, S., Deng, Y., Wang, R., Zhang, B., Chen, D., Tong, X., Yang, J.: Structured 3d latents for scalable and versatile 3d generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21469–21480 (2025)
49. Yang, J., Yang, S., Gupta, A.W., Han, R., Fei-Fei, L., Xie, S.: Thinking in space: How multimodal large language models see, remember, and recall spaces. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10632–10643 (2025)
50. Yi, T., Fang, J., Wang, J., Wu, G., Xie, L., Zhang, X., Liu, W., Tian, Q., Wang, X.: Gaussiandreamer: Fast generation from text to 3d gaussians by bridging 2d and 3d diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6796–6807 (2024)
51. Yu, M., Lu, T., Xu, L., Jiang, L., Xiangli, Y., Dai, B.: Gsdf: 3dgs meets sdf for improved neural rendering and reconstruction. In: Advances in Neural Information Processing Systems. vol. 37, pp. 129507–129530 (2024)
52. Zhang, Q., Wang, C., Siarohin, A., Zhuang, P., Xu, Y., Yang, C., Lin, D., Zhou, B., Tulyakov, S., Lee, H.Y.: Towards text-guided 3d scene composition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6829–6838 (2024)
53. Zhang, W., Liu, Y.S., Han, Z.: Neural signed distance function inference through splatting 3d gaussians pulled on zero-level set. In: Advances in Neural Information Processing Systems. vol. 37, pp. 101856–101879 (2024)
54. Zhou, D., Li, Y., Ma, F., Zhang, X., Yang, Y.: Migc: Multi-instance generation controller for text-to-image synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6818–6828 (2024)
55. Zhou, J., Li, X., Qi, L., Yang, M.H.: Layout-your-3d: Controllable and precise 3d generation with 2d blueprint. In: International Conference on Learning Representations (2025)
56. Zhou, X., Ran, X., Xiong, Y., He, J., Lin, Z., Wang, Y., Sun, D., Yang, M.H.: Gala3d: towards text-to-3d complex scene generation via layout-guided generative gaussian splatting. In: Proceedings of the International Conference on Machine Learning. pp. 62108–62118 (2024)
57. Zhu, J., Zhuang, P., Koyejo, S.: Hifa: High-fidelity text-to-3d generation with advanced diffusion guidance. In: International Conference on Learning Representations (2024)