SceneGaussian: Unconstrained Generation of 3D Gaussian Splatting Scenes

Hanzhe Hu, Qin Han, Haoyang He
Carnegie Mellon University

1 Introduction

3D scene generation is a crucial component for many applications, such as AR/VR asset creation and film production. This requires the ability to create diverse and realistic 3D scenes from any type of input, such as text and RGB images. Suppose we want to create an immersive 3D environment based on an input image, current 3D reconstruction techniques often struggle to produce high-fidelity scenes from limited inputs, leading to inaccuracies and artifacts that detract from the user experience. In this work, we explore how to generate consistent 3D scenes along a long-range camera trajectory based on a single image input.

Recent success in 2D generative models such as Stable Diffusion has greatly pushed the progress of 3D generation. Many works such as DreamFusion [13], Get3D [4], and Zero-123 [10] have managed to create high-fidelity 3D shapes from text or image inputs. However, these methods mainly focus on object-level data, and cannot generalize to scene-level data, which is more challenging and non-trivial. The variance of 3D scenes is much larger than a single object. On the other hand, recent works such as SceneScape [3], Text2Room [7], and RoomDreamer [15] proposed to generate 3D scenes rather than simple objects. However, these methods often suffer from slow rendering speed and limitations in indoor scenes. The most related work to ours is LucidDremaer [2], which generates point cloud in a incremental manner and fits a 3D Gaussian representation afterward, can not produce a ready-to-use 3D scene at each new camera, limiting its applications to real-world scenarios.

In this project, we propose SceneGaussian: a Gaussian Splatting [8] based image-to-3D framework to generate consistent and immersive 3D scenes while maintaining high rendering speed. We first leverage an off-the-shelf monocular depth estimation model to predict the depth of the input image, and then unproject it to 3D points, which serve as the initial 3D Gaussians. We then optimize it to have the initial 3D Gaussian representation. As we move the camera along an arbitrary trajectory, we render the existing 3D Gaussian into a new image with black content from the new camera pose. A pre-trained stable diffusion inpainting model is leveraged to inpaint the rendered image with complete content. The inpainted image is further used for updating the 3D Gaussians. In particular, we propose an efficient **online Gaussian updating** strategy to optimize 3D Gaussians in a real-time manner. We only add Gaussians to places where new contents are generated. Instead of optimizing the 3D Gaussians with all available images, we select k key frames that contribute to the new content of the 3D scene.

Our proposed method can generate more realistic 3D scenes and exhibit faster rendering speed compared with existing methods. The generated 3D scenes yield high consistency across frames and restore the identity from the single input image. The 3D scene is generated in an incremental manner along an arbitrary camera trajectory, where a consistent 3D representation is maintained. We compare with other 3D scene generation methods and achieve competitive qualitative and quantitative results.

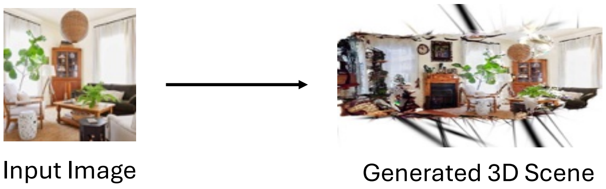

Figure 1: Given an input image, our method can generate a 3D scene along an arbitrary camera trajectory.

2 Related Work

2.1 Text-to-3D Generation

Building on the success of text-to-image generation, recent advancements in the exploration of text-to-3D generation have been significant. DreamFusion [13] first leverages pre-trained text-to-image diffusion models to optimize the NeRF [11]-based 3D representation, using Score Distillation Sampling (SDS). To overcome the drawbacks of over-saturation, over-smoothing, and low diversity associated with SDS, ProlificDreamer [18] proposes a principled partical-based variational framework along with Variational Score Distillation (VSD), while SJC [17] introduces Perturb and Averaging Scoring (PAAS) method. However, modeling high-fidelity and complex 3D scenes involving multiple objects with intricate interactions remains challenging with the implicit NeRF representation.

More recently, 3D Gaussian Splatting [8] introduces an efficient Lagrangian 3D representation composed of a collection of 3D anisotropic Gaussian Spheres. The integration of the text-to-3D framework with 3D Gaussian Splatting has led to significant advancements in the field, achieving unprecedented rendering quality and efficiency. GaussianDreamer [19] utilizes 3D diffusion priors and 2D diffusion priors for 3DGS initialization and optimization, respectively. DreamGaussian [16] designs a generative 3D Gaussian Splatting model with companioned mesh extraction and texture refinement in UV space. Additionally, GaussianDiffusion [9] proposes a variational 3D Gaussian Splatting tailored for Denoising Diffusion Probabilistic Models (DDPM) [6] models with structured noise. However, these works primarily concentrate on object-centric 3D generation, and may generate implausible 3D scenes when involving multiple objects. Instead, our project focuses on large-scale 3D scene generation, incorporating layout guidance to enhance the quality and consistency of the generated scenes.

2.2 3D Scene Generation

In the domain of 3D scene generation, the complex structure of scenes has driven researchers to develop numerous representation methods, both explicit and implicit, to capture extensive environments. Implicit methods often involve encoding scenes within neural networks, employing techniques like Signed Distance Fields (SDF), Occupancy Fields, and Neural Radiance Fields (NeRF). Notable works like Text2NeRF [21] have adopted NeRF to enhance generalizability in generation, though this approach can complicate scene control and editing.

Conversely, explicit representations such as meshes, voxels, and point clouds remain popular. Recent projects like Text2Room [7] and SceneScape [3] have innovated by integrating 2D diffusion models to progressively generate meshes; however refine the generated scene to only indoor environments. Additionally, point clouds serve as a versatile tool in efforts like WonderJourney[20], which bridges keyframes by progressively rendering and inpainting intermediate point clouds, derived from monocular depth estimation and refinement.

A novel advancement in 3D representation is the use of Gaussian Splatting [8], which has been explored in works like LucidDreamer [2] and Text2Immersion [12]. These projects use 3D Gaussians as the final 3D representation, employing pre-trained 2D diffusion models to generate images from multiple viewpoints, which are then lifted into 3D scenes using depth estimation models. This method not only achieves photorealistic renderings but also maintains high rendering speeds.

Our methodology aligns with these approaches but diverges in the utilization of 3D Gaussians. Unlike others that primarily rely on point clouds for generating multiview 2D images and only incorporate scene Gaussians in the final stages, our strategy optimizes 3D Gaussians dynamically. This online optimization ensures consistent 3D representations and allows for immediate integration of new camera viewpoints, enhancing both the speed and realism of 3D scene generation.

3 Method

Given an input RGB image describing a partial scene, SceneGaussian aims to generate the 3D Gaussians of the complete scene along an arbitrary camera trajectory. The 3D Gaussians keep getting updated during the generation process.

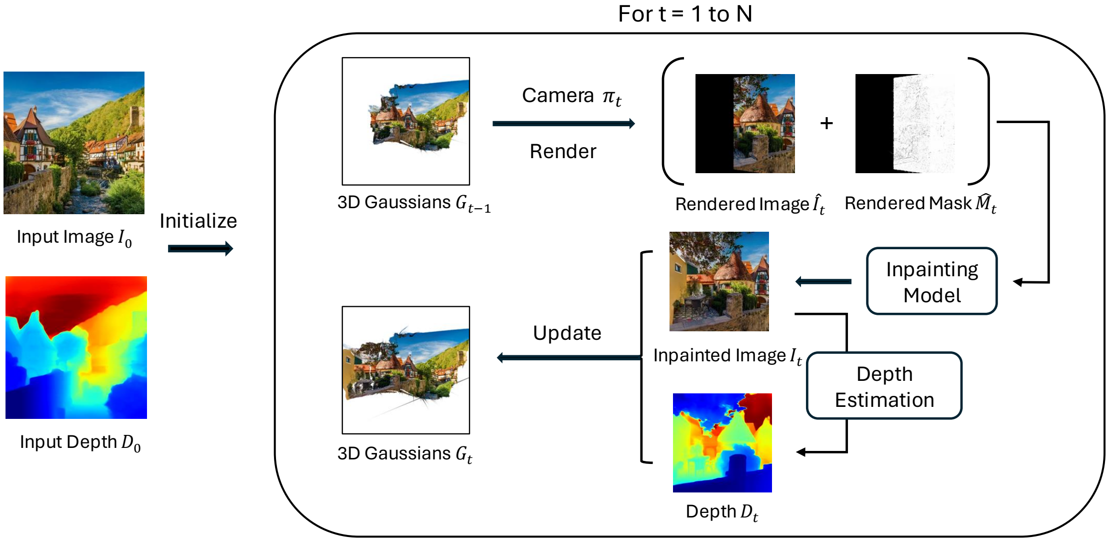

Figure 2: The overall framework of SceneGaussian. The 3D Gaussian representation is initialized by the input image and the estimated depth and gets updated along the camera trajectory. For each new camera π_t, we render the existing 3D Gaussians into $\hat{I}_t$ and $\hat{M}_t$, which are further fed into a Stable Diffusion Inpainting model to produce the inpainted image I_t. The new image with its estimated depth can be used for updating the 3D Gaussians.

3.1 3D Gaussian Splatting

3D Gaussian Representation. We represent the underlying scene using a set of 3D Gaussians. Each 3D Gaussian can be denoted as:

$$g_i(x) = exp(-\frac{1}{2}(\mathbf{x} - \mu_i)^T \Sigma_i^{-1}(\mathbf{x} - \mu_i)), \tag{1}$$

where i is the index of 3D Gaussians, $\mu_i \in \mathbb{R}^3$ is the Gaussian mean and $\Sigma_i \in \mathbb{R}^{3\times3}$ is its covariance, specifying its shape and size. Each Gaussian also has an opacity $\sigma_i \in \mathbb{R}_+$ and a view-dependent color $c_i(\mathbf{v}) \in \mathbb{R}^3$.

Rendering via Splatting. In our framework, we render the existing 3D Gaussians at each time step in a differentiable way and further update the 3D Gaussians by enforcing the RGB loss between renderings and inpainted images. Given a collection of 3D Gaussians and camera pose, we first sort all Gaussians from front-to-back. RGB images can be rendered by alpha-compositing the splatted 2D projection of each Gaussian in order in pixel space. The color of pixel $\mathbf{p}$ can be denoted as:

$$C(\mathbf{p}) = \sum_{i=1}^{N} \mathbf{c}_i g_i(\mathbf{p}) \prod_{j=1}^{i-1}(1 - g_i(\mathbf{p})). \tag{2}$$

3.2 3D Scene Generation

We first unproject the input image into a 3D point cloud using monocular depth estimation and learns an initial 3D Gaussian representation. The 3D Gaussians of the scene keep getting updated along the camera trajectory using Stable Diffusion and online 3D Gaussian updating. The generated 3D scene is consistent with the input image.

Initialization We first initialize the 3D Gaussians using a single input image $I_0 \in \mathbb{R}^{3\times H\times W}$. An off-the-shelf monocular depth estimation model is utilized to estimate the depth map $D_0 \in \mathbb{R}^{H\times W}$ of the input image. With the depth map, we can unproject the 2D image into a 3D point cloud and use that as the initialized means of 3D Gaussians. We learn an initial 3D Gaussian representation by fitting on the input image by minimizing the RGB loss function.

Inpaiting Given a new camera π_t at time step t, we render the existing 3D Gaussians G_{t-1} into an image $\hat{I}_t$ and the corresponding mask M_t that denotes whether the pixel contains information from the current 3D Gaussians, by $\hat{I}_t, M_t = \text{Render}(G_{t-1}, \pi_t)$. A pretrained Stable Diffusion inpainting model f_i is employed to generate a realistic image $I_t = f_i(\hat{I}_t, M_t)$. A corresponding depth map is obtained by the depth estimation model: $D_t = f_d(I_t)$. Based on the rendered mask M_t, we can denote which pixels contribute to new contents of the generated scene. Therefore, we unproject those pixels into 3D points which serve as the initial means of new 3D Gaussians. The updated 3D Gaussians are denoted as: $G_t = \text{Update}(I_t, D_t, M_t)$.

Gaussian Updating We update the parameters of the 3D Gaussians at each time step. The optimization process is similar to fitting a radiance field to images with known camera poses. However, in order to save the computation cost, we don't optimize over the whole set of the generated images since most of the images do not affect the existing 3D Gaussians. Instead, we select the most recent k frames to optimize the parameters of the 3D Gaussians.

4 Experiments

4.1 Implementation Details

We leverage pre-trained large-scale off-the-shelf models to build our generation framework; note that the models could be either trained using manual design or brought from off the shelf. In detail, we adopt Stable Diffusion model [14] to inpaint the partial rendered image with a rendered mask. Additionally, we utilize ZoeDepth[1] as our monocular depth estimator. For the camera trajectory, simulate a 180-degree camera rotation trajectory to generate the scene. To evaluate our generation process, we manually collect pairs of images with corresponding captions from the internet as text prompts for our models.

4.2 Results and Analysis

We evaluate the performance of our approach against the baselines of Text2Room and LucidDreamer using the average CLIPScore [5] of the input prompt used against 10 images rendered from viewing the generated mesh from cameras with 10 equal-angular horizontal rotations, summing up to 180 degrees. We manually labelled 8 prompts from 8 input images representing a diverse set of indoor and outdoor scenes, and evaluated the average CLIPScore for all rendered images in the scene, and the average CLIPScore for all scenes experimented. The results are shown in Table 1. We achieve similar CLIPScore as the baselines.

Table 1: Comparison of average CLIPScores for different inputs of prompt-image pairs.

Input	Text2Room	LucidDreamer	SceneGaussian
Kitchen-1	20.2415	23.0128	22.8589
Village-2	22.4866	19.8711	20.0014
Village-3	19.8962	20.7884	21.1376
Livingroom-1	22.9049	20.2628	19.7778
Livingroom-2	18.3841	20.7179	19.7497
Livingroom-3	22.9222	22.5316	22.6112
Sakura	18.1700	20.5635	20.2401
Japan	17.7104	19.9709	19.6931
Average	20.3395	20.9649	20.7587

We also show a qualitative comparison for our generated scenes to Text2Room and LucidDreamer in Figure 3 and 4. For each comparison, we show the input image and text prompt at the left, and display two generated views and a panographic view of the generated scenes for each model. In our approach, when we input the rendered image to the Stable Diffusion inpainting model, the rendered image from 3D Gaussians may generate some artifacts, which could confuse the pre-trained inpainting model to generate less consistent visual contents at the next timestep. This results in our model sometimes producing semantically less consistent scenes compared to the baselines.

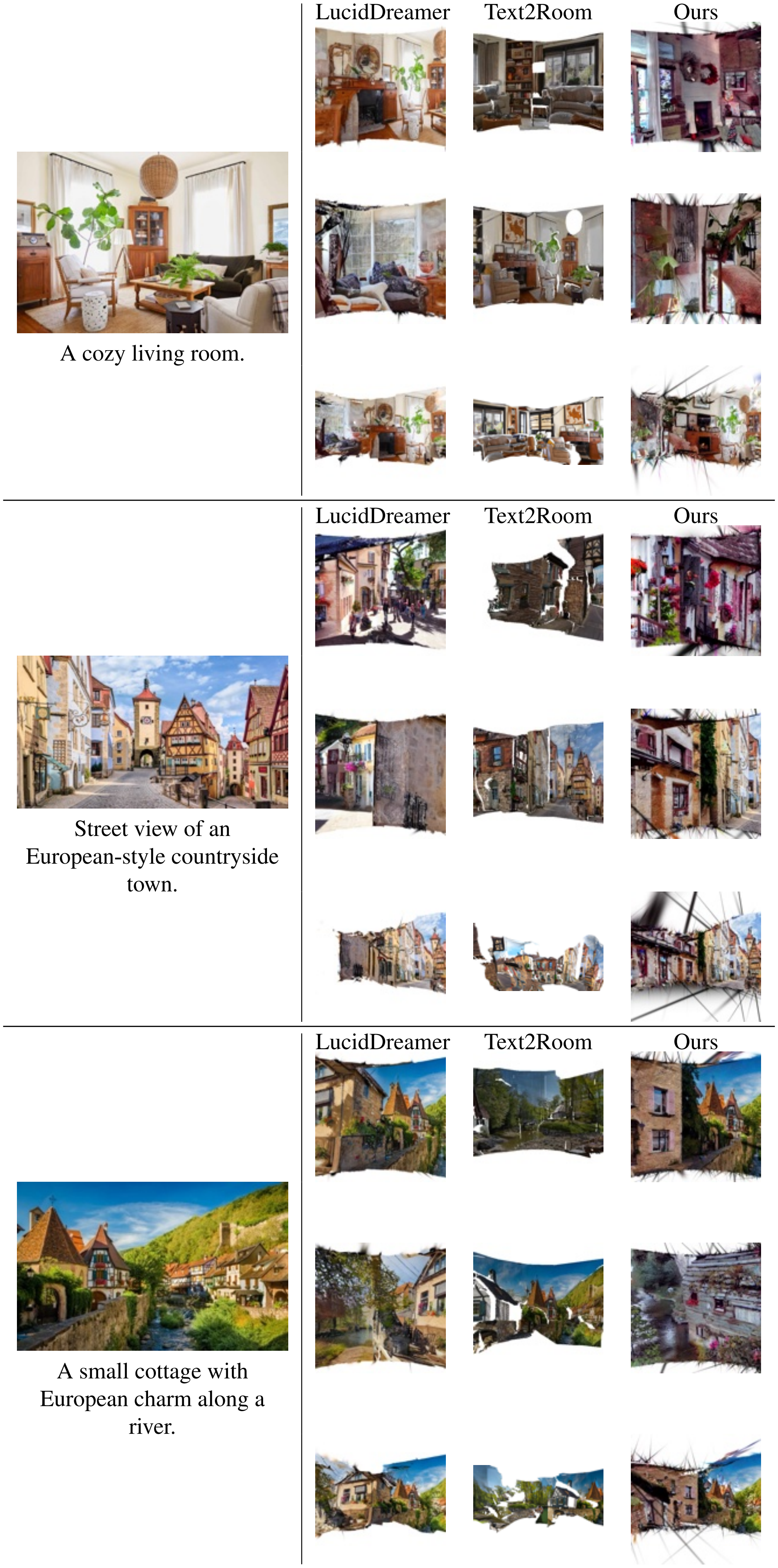
LucidDreamer
Text2Room
Ours
A cozy living room.
LucidDreamer
Text2Room
Ours
Street view of an European-style countryside town.
LucidDreamer
Text2Room
Ours
A small cottage with European charm along a river.

Figure 3: Qualitative comparison with Text2Room [7] and LucidDreamer [2].

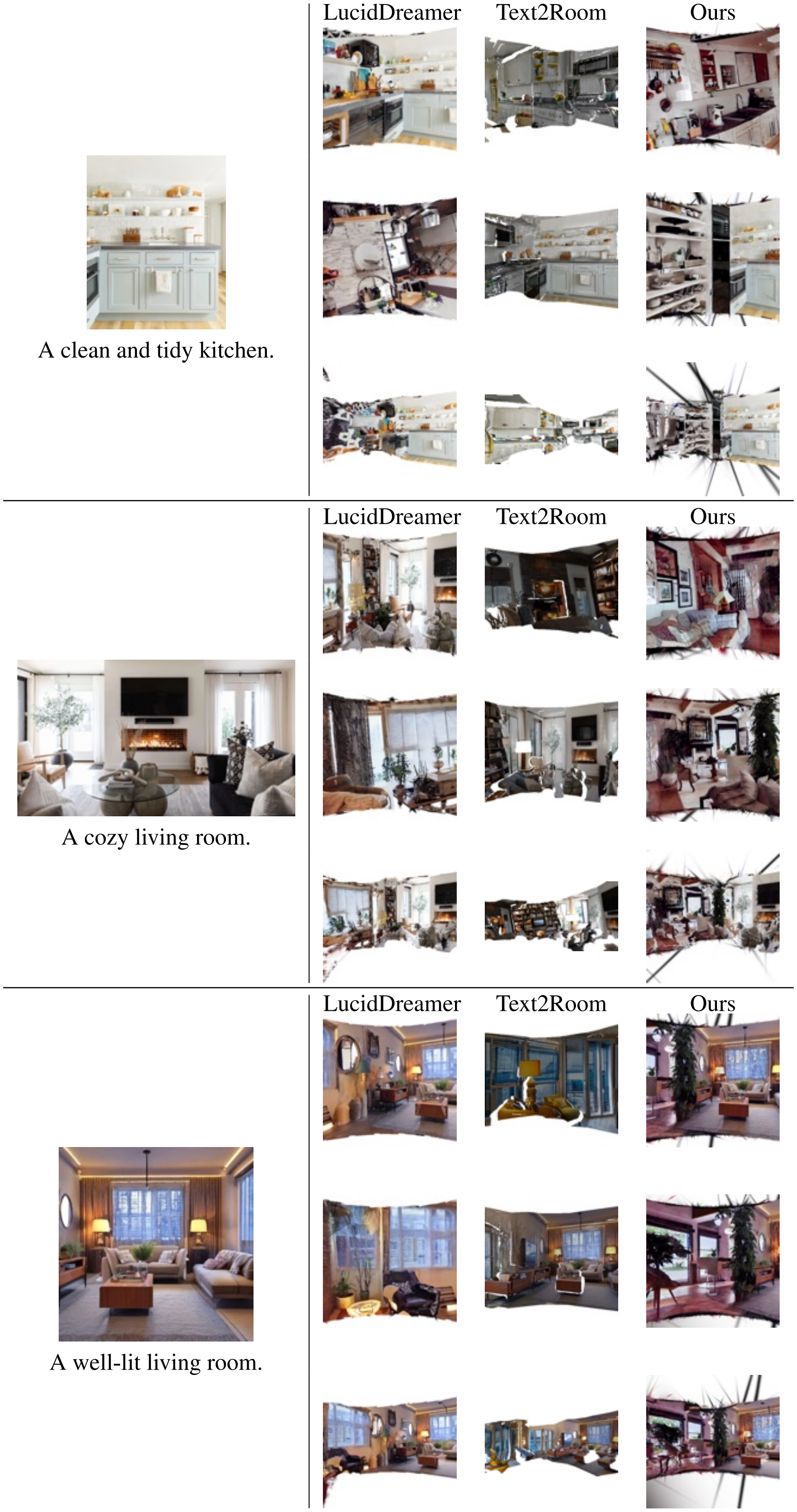

Figure 4: Qualitative comparison with Text2Room [7] and LucidDreamer [2].

5 Conclusion

In this paper, we introduced SceneGaussian, an innovative framework for generating consistent 3D scenes from a single image input along an arbitrary camera trajectory using 3D Gaussian splatting. Our method efficiently leverages off-the-shelf models for depth estimation and image inpainting to dynamically update and maintain a realistic 3D scene. Our experiments demonstrate that SceneGaussian not only produces visually appealing and realistic scenes but also performs competitively with existing methods in terms of qualitative assessments using CLIPScore metrics.

Through comparative analysis, SceneGaussian proved its capability in generating complex scenes with greater consistency and realism than other recent methods. The key to our approach is the online updating mechanism for 3D Gaussians, which significantly contributes to the efficiency and efficacy of scene generation across varied viewpoints. This feature is particularly crucial for applications in virtual reality, augmented reality, and other fields requiring rapid and reliable scene generation.

References

[1] Shariq Farooq Bhat et al. *ZoeDepth: Zero-shot Transfer by Combining Relative and Metric Depth*. 2023. DOI: 10.48550/ARXIV.2302.12288. URL: https://arxiv.org/abs/2302.12288.

[2] Jaeyoung Chung et al. "Luciddreamer: Domain-free generation of 3d gaussian splatting scenes". In: *arXiv preprint arXiv:2311.13384* (2023).

[3] Rafail Fridman et al. "SceneScape: Text-Driven Consistent Scene Generation". In: *arXiv preprint arXiv:2302.01133* (2023).

[4] Jun Gao et al. "Get3d: A generative model of high quality 3d textured shapes learned from images". In: *Advances In Neural Information Processing Systems* 35 (2022), pp. 31841–31854.

[5] Jack Hessel et al. "CLIPScore: A Reference-free Evaluation Metric for Image Captioning". In: *CoRR* abs/2104.08718 (2021). arXiv: 2104.08718. URL: https://arxiv.org/abs/2104.08718.

[6] Jonathan Ho, Ajay Jain, and Pieter Abbeel. *Denoising Diffusion Probabilistic Models*. 2020. arXiv: 2006.11239 [cs.LG].

[7] Lukas Höllein et al. "Text2room: Extracting textured 3d meshes from 2d text-to-image models". In: *arXiv preprint arXiv:2303.11989* (2023).

[8] Bernhard Kerbl et al. "3D Gaussian Splatting for Real-Time Radiance Field Rendering". In: *ACM Transactions on Graphics* 42.4 (July 2023). URL: https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/.

[9] Xinhai Li, Huaibin Wang, and Kuo-Kun Tseng. *GaussianDiffusion: 3D Gaussian Splatting for Denoising Diffusion Probabilistic Models with Structured Noise*. 2023. arXiv: 2311.11221 [cs.CV].

[10] Ruoshi Liu et al. "Zero-1-to-3: Zero-shot one image to 3d object". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 9298–9309.

[11] Ben Mildenhall et al. "NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis". In: *ECCV*. 2020.

[12] Hao Ouyang et al. "Text2immersion: Generative immersive scene with 3d gaussians". In: *arXiv preprint arXiv:2312.09242* (2023).

[13] Ben Poole et al. *DreamFusion: Text-to-3D using 2D Diffusion*. 2022. arXiv: 2209.14988 [cs.CV].

[14] Robin Rombach et al. *High-Resolution Image Synthesis with Latent Diffusion Models*. 2022. arXiv: 2112.10752 [cs.CV].

[15] Liangchen Song et al. "Roomdreamer: Text-driven 3d indoor scene synthesis with coherent geometry and texture". In: *arXiv preprint arXiv:2305.11337* (2023).

[16] Jiaxiang Tang et al. "DreamGaussian: Generative Gaussian Splatting for Efficient 3D Content Creation". In: *arXiv preprint arXiv:2309.16653* (2023).

[17] Haochen Wang et al. *Score Jacobian Chaining: Lifting Pretrained 2D Diffusion Models for 3D Generation*. 2022. arXiv: 2212.00774 [cs.CV].

[18] Zhengyi Wang et al. "ProlificDreamer: High-Fidelity and Diverse Text-to-3D Generation with Variational Score Distillation". In: *arXiv preprint arXiv:2305.16213* (2023).

[19] Taoran Yi et al. "GaussianDreamer: Fast Generation from Text to 3D Gaussians by Bridging 2D and 3D Diffusion Models". In: *CVPR* (2024).

[20] Hong-Xing Yu et al. “WonderJourney: Going from Anywhere to Everywhere”. In: *arXiv preprint arXiv:2312.03884* (2023).

[21] Jingbo Zhang et al. *Text2NeRF: Text-Driven 3D Scene Generation with Neural Radiance Fields*. 2024. arXiv: 2305.11588 [cs.CV].