Director3D: Real-world Camera Trajectory and 3D Scene Generation from Text

Xinyang Li[1][*] **Zhangyu Lai**[1] **Linning Xu**[3] **Yansong Qu**[1]
Liujuan Cao[1][†] **Shengchuan Zhang**[1] **Bo Dai**[4,2] **Rongrong Ji**[1]
[1] Key Laboratory of Multimedia Trusted Perception and Efficient Computing,
Ministry of Education of China, Xiamen University
[2] Shanghai Artificial Intelligence Laboratory,
[3] The Chinese University of Hong Kong, [4] University of Hong Kong
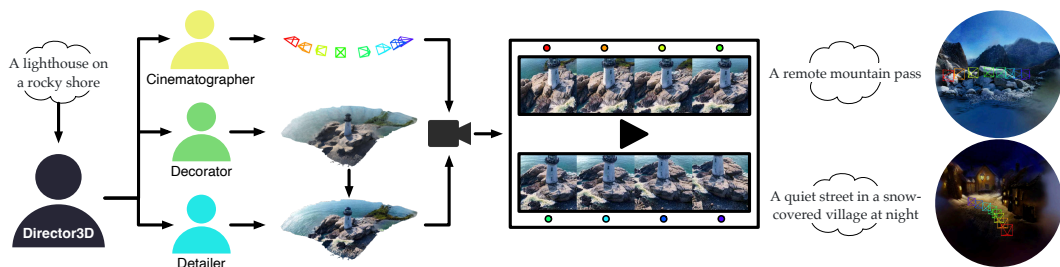Code: `https://github.com/imlixinyang/director3d`

Figure 1: Given textual descriptions, Director3D employs three key components: the Cinematographer generates the camera trajectories, the Decorator creates the initial 3D scenes, and the Detailer refines the details.

Abstract

Recent advancements in 3D generation have leveraged synthetic datasets with ground truth 3D assets and predefined camera trajectories. However, the potential of adopting real-world datasets, which can produce significantly more realistic 3D scenes, remains largely unexplored. In this work, we delve into the key challenge of the complex and scene-specific camera trajectories found in real-world captures. We introduce **Director3D**, a robust open-world text-to-3D generation framework, designed to generate both real-world 3D scenes and adaptive camera trajectories. To achieve this, (1) we first utilize a Trajectory Diffusion Transformer, acting as the *Cinematographer*, to model the distribution of camera trajectories based on textual descriptions. (2) Next, a Gaussian-driven Multi-view Latent Diffusion Model serves as the *Decorator*, modeling the image sequence distribution given the camera trajectories and texts. This model, fine-tuned from a 2D diffusion model, directly generates pixel-aligned 3D Gaussians as an immediate 3D scene representation for consistent denoising. (3) Lastly, the 3D Gaussians are further refined by a novel SDS++ loss as the *Detailer*, which incorporates the prior of the 2D diffusion model. Extensive experiments demonstrate that Director3D outperforms existing methods, offering superior performance in real-world 3D generation.

---

[*]Work done during an internship of Xinyang Li with Shanghai Artificial Intelligence Laboratory
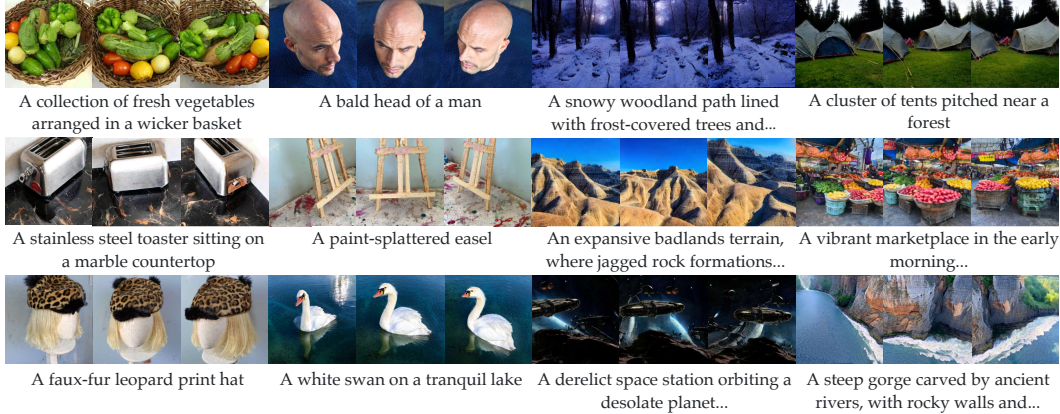[†]Corresponding Author

Figure 2: Multi-view image results rendered with the generated camera trajectories and 3D scenes.

1 Introduction

Generating 3D scenes from texts holds great promise for industries such as gaming, robotics, and VR/AR. Previous methods [1, 2, 3], which use score distillation sampling (SDS) to optimize 3D representations such as Neural Radiance Fields (NeRFs) [4, 5], involve lengthy and unstable optimization processes. In contrast, newer approaches employ feed-forward networks [6, 7], *e.g.*, diffusion and reconstruction models, to directly generate 3D representations from text or text-guided multi-view images, significantly enhancing generation speed. Moreover, advancements in 3D Gaussian Splatting [8] further accelerate training and rendering speeds, driving the next wave of progress in text-to-3D generation [9, 10, 11, 12, 13]. However, most existing methods focus solely on object-level 3D generation. Recently, preliminary works [14, 15, 16, 17, 18] have begun addressing scene-level 3D generation. Despite these efforts, visual quality, generation speed, and generalization remain suboptimal due to reliance on only 2D priors or limited few-classes 3D datasets.

In this work, we leverage real-world datasets (*e.g.*, MVImgNet [19] and DL3DV-10K [20]) to achieve realistic text-to-3D generation. However, real-world captures from in-the-wild scenes differ significantly from traditional object-level synthetic datasets, introducing new requirements for the text-to-3D generation framework. Firstly, real-world captures feature complex, unpredictable, and scene-specific camera trajectories, unlike the controlled and predefined settings in object-level synthetic datasets like Objaverse [21] shown in Fig. 3 (Left). Secondly, real-world scenes include unbounded backgrounds, complicating the use of common bounded 3D representations such as Tri-planes [22]. Lastly, the diversity and quantity of real-world captures are limited, potentially decreasing the generalization ability for open-world texts.

We address these challenges with a novel framework, Director3D, illustrated in Fig. 1 and Fig. 3 (Right). Fig. 2 shows that our framework supports generating 3D scenes across various domains. In summary, our approach includes the following three key components:

• Traj-DiT (Trajectory Diffusion Transformer) as Cinematographer: Generates dense-view camera trajectories from text. Camera parameters (intrinsics and extrinsics) are treated as temporal tokens, and a Transformer model performs conditional denoising of the camera trajectory.

• GM-LDM (Gaussian-driven Multi-view Latent Diffusion Model) as Decorator: Uses a sparse-view subset of the camera trajectory for image sequence diffusion, generating pixel-aligned and unbounded 3D Gaussians as intermediate 3D representations. This model, fine-tuned from a 2D latent diffusion model, leverages strong priors and collaborative training with multi-view and single-view data to mitigate the limited diversity and quantity of real-world captures, enhancing generalization.

• SDS++ Loss as Detailer: Enhances visual quality of the 3D Gaussians by back-propagates a novel SDS++ loss from images rendered at randomly interpolated cameras within the trajectory.
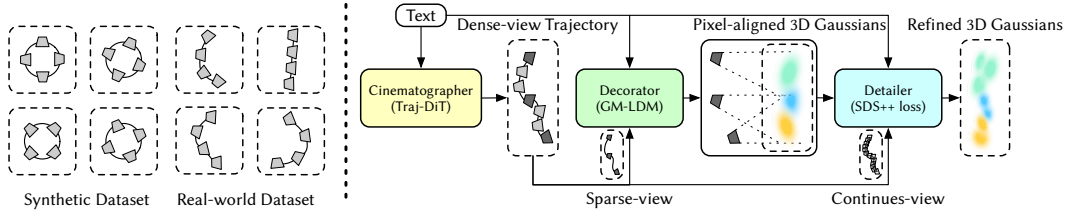
Figure 3: **Left**: Comparison of the simplified camera trajectory distributions between synthetic and real-world multi-view datasets. **Right**: Pipeline and models of Director3D.

2 Related Works

Text-to-3D Generation. Since the introduction of DreamField [23], which combines the vision-language model CLIP [24] with NeRFs [4], there has been substantial progress in the area of text-to-3D generation. DreamFusion [1] and SJC [25] employ 2D image diffusion models to refine the 3D representation through Score Distillation Sampling (SDS). Subsequent methods [26, 27, 28, 29, 30] enhance the formulations to more effectively utilize 2D diffusion models for text-to-3D generation, achieving greater stability and improved visual quality. Some methods [3, 31, 32, 33] suggest incorporating an additional 3D prior into 2D diffusion models to enhance 3D consistency. Additionally, various methods [34, 35, 36, 37, 7, 38] employ 3D diffusion models to directly produce 3D assets. However, most of these works primarily focus on object-level generation. For scene-level 3D generation, preliminary works [15, 14, 39, 40, 41, 16, 42] combine image inpainting models [43, 44] and monocular depth estimation models [45, 46] to progressively lift an image into a 3D scene with a user-defined camera trajectory. Nonetheless, the multi-view inconsistency of image inpainting and monocular depth estimation can lead to undesirable geometry and texture artifacts. Also, some works [47, 48] propose using panorama image diffusion models for 3D scene generation, which is limited to some specific types of scenes such as indoor rooms. Therefore, developing a text-to-3D scene generation method with open-world generalization capabilities remains unsolved.

3D Gaussian Splatting (3DGS) [8] introduces parameterized 3D Gaussians as 3D representation and splatting-based rasterization technique for novel view synthesis based on dense-view images, significantly reducing the rendering time compared to NeRF-based methods [4, 49, 5, 50, 51]. Current methods concentrate on improving the geometry quality [52, 53, 54], stabilizing the training [55], adapting it to dynamic scene modeling [56, 57, 58, 59] and applying it to different applications [60, 61, 62, 63, 64]. Further, to train a generalizable sparse-view reconstruction model with 3D Gaussians as the intermediate representation, some methods [65, 66, 67, 38, 12, 13] propose to convert image features into pixel-aligned 3D Gaussians and optimize reconstruction models through back-propagating losses from rendered images.

Meanwhile, DreamGaussian employs 3DGS for text-to-3D generation, with SDS loss as the optimization objective. GaussianDreamer [11] and GSGen [10] further enhance the generation quality and 3D consistency by initializing the 3D Gaussians based on point cloud diffusion models, Point-E [34]. The generalizable sparse-view reconstruction model (*e.g.*, GRM [38], LGM [12], and GS-LRM [13]) for 3D Gaussians can also facilitate text-to-3D generation, using multi-view diffusion models [3, 68] to acquire sparse-view images as inputs. Specifically, GS-LRM takes a step forward by employing a large video generation model (i.e., SORA [69]) for text-to-3D scene generation. However, these works heavily rely on the 3D consistency of the multi-view images, which the 2D-based diffusion models can not guarantee. Distinctively, our GM-LDM employs pixel-aligned 3D Gaussians as the intermediate 3D representation for rendering-based multi-view diffusion, directly enforcing 3D consistency during the diffusion process and producing 3D representations.

3 Preliminary

Latent Diffusion Models (LDMs) [44, 70] consist of two key components: an auto-encoder [71] and a latent denoising network. The autoencoder establishes a bi-directional mapping from the space of the original data to a low-resolution latent space: $z = \mathcal{E}(x), x = \mathcal{D}(z)$, where $\mathcal{E}$ and $\mathcal{D}$ are the encoder and decoder, respectively. The latent denoising network ϵ_θ is trained to denoise noisy latent

given a specific timestep t and condition y. Its training objective for ϵ-prediction is defined as:

$$L = \mathbb{E}_{x,\epsilon \sim \mathcal{N}(0,1),t}\Big[\|\epsilon - \epsilon_\theta(z_t, y, t)\|_2^2\Big], \tag{1}$$

where the noisy latent is obtained by $z_t = \sqrt{\bar{\alpha}_t}E(x) + \sqrt{1 - \bar{\alpha}_t}\epsilon$, $\bar{\alpha}_t$ is a monotonically decreasing noise schedule and $\epsilon \sim \mathcal{N}(0,1)$ is a random noise. During inference, a random noise is sampled as $z_T \sim \mathcal{N}(0,1)$. By continuously denoising the random noise z_T with condition y (*e.g.*, text embedding), we can derive a fully denoised latent $\hat{z}$. Then, the denoised latent $\hat{z}$ is fed into the latent decoder $\mathcal{D}$ to generate the high-resolution image $\hat{x} = \mathcal{D}(\hat{z})$.

Multi-view Diffusion Models aim to model the distribution of multi-view images $\mathcal{X}$ with 3D consistency, where each image is captured by a distinct camera within the same static 3D scene. Its objective for x_0-prediction can be written as:

$$L = \mathbb{E}_{\mathcal{X},\epsilon \sim \mathcal{N}(0,1),t}\Big[\|\mathcal{X} - \mathcal{X}_\theta(\mathcal{X}_t, \mathcal{C}, y, t))\|_2^2\Big], \tag{2}$$

where $\mathcal{C}$ represents the camera parameters for the different views. Early works [3, 31] in this field are based on 2D LDMs. They fine-tune the 2D LDMs by integrating cross-view connections between the multi-view images into the original single-view 2D LDMs, using multi-view data rendered from 3D datasets. These methods lack strict 3D consistency since there is no actual 3D representation during multi-view denoising. A more advanced approach, DMV3D [7], employs a 3D reconstruction model to generate noise-free 3D representations and predict multi-view images from noisy multi-view inputs by a rendering-based denoising process. This enables 3D generation tasks to be accomplished without per-asset optimization during inference.

Score Distillation Sampling (SDS) [1, 2] uses a pretrained 2D diffusion model to optimize 3D representation. Considering a differentiable 3D representation parameterized by $\mathcal{G}$ and a rendering function denoted as $\mathcal{R}$, the rendered image produced for a given camera pose c can be expressed as $x = \mathcal{R}(\mathcal{G}, c)$. SDS distills the prior of a 2D LDM to optimize 3D representation $\mathcal{G}$ as follows:

$$\nabla_\mathcal{G}\mathcal{L}_{\text{SDS}} = \mathbb{E}_{t,\epsilon,c}\left[w(t)\,(\hat{\epsilon} - \epsilon)\,\frac{\partial\mathcal{E}(\mathcal{R}(\mathcal{G}, c))}{\partial\mathcal{G}}\right] \tag{3}$$

where ϵ is the ground truth noise, $\hat{\epsilon}$ is the noise predicted by the 2D LDM with z_t as input for timestep t, and $w(t)$ represents a weighting function that varies according to the timestep t. The SDS loss can be also converted into a reconstruction-like objective [72]:

$$\mathcal{L}_{\text{SDS}} = \mathbb{E}_{t,\epsilon,c}\left[w(t)\frac{\sqrt{\bar{\alpha}_t}}{\sqrt{1 - \bar{\alpha}_t}}\|z - \hat{z}\|_2^2\right], \text{where } \hat{z} = (z_t - \sqrt{1 - \bar{\alpha}_t}\hat{\epsilon})/\sqrt{\bar{\alpha}_t} \tag{4}$$

4 Method

4.1 Problem Formulation and Overview of Director3D

We consider the multi-view dataset of real-world captures as a joint distribution of image sequences and camera trajectories conditioned on texts, denoted as $p((\mathcal{X}, \mathcal{C})|y)$. Here, $\mathcal{X} = \{x_i\}_{i=1}^M$ represents the image sequence, $\mathcal{C} = \{c_i\}_{i=1}^M$ denotes the camera trajectory, and M is the number of views. To model this joint distribution, we separately handle the conditional distributions $p(\mathcal{C}|y)$ and $p(\mathcal{X}|(\mathcal{C}, y))$ (see Appendix B for detailed discussions). Furthermore, we model each image in the sequence as a rendered view of a unified 3D scene representation $\mathcal{G}$ under the corresponding camera, expressed as $x_i = \mathcal{R}(\mathcal{G}, c_i)$, where $\mathcal{R}$ is the 3D rendering function.

Director3D addresses this by incorporating three collaborative processes analogous to roles in film production: the Cinematographer, the Decorator, and the Detailer. Firstly, the Trajectory Diffusion Transformer (Traj-DiT), serving as the Cinematographer, models the distribution of dense-view camera trajectories, as detailed in Sec. 4.2. For image sequences, directly modeling the dense-view distribution is complex and resource-intensive. To address this, we use a Gaussians-driven Multi-view Latent Diffusion Model (GM-LDM), acting as the Decorator, to model the image distribution through a sparse subset of dense views. This model utilizes pixel-aligned 3D Gaussians as the intermediate representation, described in Sec. 4.3. Finally, to improve the visual quality of the generated 3D scenes, we employ a novel SDS++ loss, functioning as the Detailer, to refine the 3D Gaussians through dense-camera interpolation rendering, as presented in Sec. 4.4.
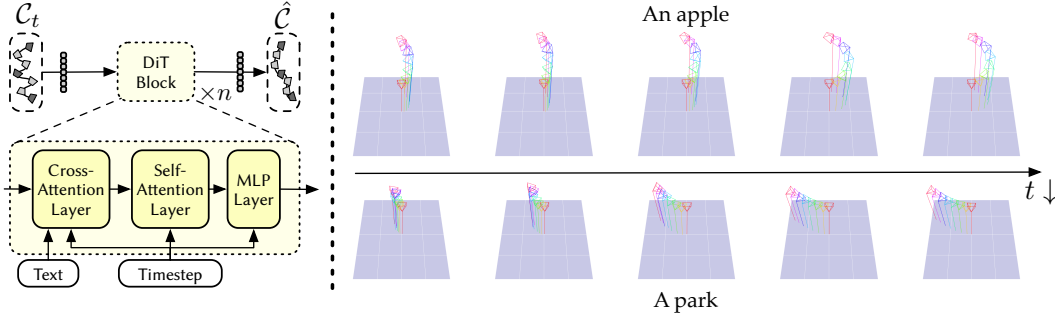
Figure 4: **Left**: Architecture of Traj-DiT. **Right**: Visualization of the predicted camera trajectory for different denoising timesteps.

4.2 Traj-DiT as Cinematographer

To model the trajectory distribution, we represent the camera trajectory $\mathcal{C}$ as a set of camera parameters $c_i = \{\mathbf{r}_i, \mathbf{t}_i, \mathbf{f}_i, \mathbf{p}_i\}$, where $\mathbf{r} \sim \mathrm{SO}(3)$ and $\mathbf{t} \sim \mathbb{R}^3$ are the rotation and translation of the camera poses, $\mathbf{f} \sim \mathbb{R}_+^2$ is the focal lengths and $\mathbf{p} \sim \mathbb{R}^2$ is the principle points. To ensure consistency and comparability across scenes, we normalize the trajectory for each scene in two steps: First, we convert all camera poses to be relative to the first one so that the first camera pose is an identity matrix; Then, we re-scale the translation to make the distance from the first to the farthest camera to 1. We adapt the architecture of the Diffusion Transformer (DiT) [73] to generate camera trajectories, as illustrated in Fig. 4 (Left). The temporal order of real-world captures, akin to video sequences, necessitates a learnable temporal embedding to differentiate between cameras of different frames. This embedding helps the model capture the sequential dependencies inherent in real-world data. Each DiT block includes a cross-attention layer to extract information from text embeddings encoded by the CLIP [24] text encoder. Additionally, the timestep t modulates the pre-layer normalization and post-layer output scalar, similar to the original DiT, allowing the model to learn temporal dynamics effectively. The model is trained to minimize the x_0-prediction diffusion objective:

$$L = \mathbb{E}_{\mathcal{C}, \epsilon \sim \mathcal{N}(0,1), t} \Big[\| \mathcal{C} - \mathcal{C}_\theta(\mathcal{C}_t, y, t) \|_2^2 \Big], \tag{5}$$

where $\mathcal{C}_\theta$ is the parameterized Traj-DiT model and $\mathcal{C}_t$ is the noisy camera trajectory.

By leveraging the strengths of the DiT architecture, we aim to enhance the fidelity and coherence of the generated trajectories, instead of relying on pre-defined ones. We showcase two examples of the predicted camera trajectories for different denoising steps t in Fig. 4 (Right), demonstrating the effectiveness of our model in generating smooth and accurate camera paths.

4.3 GM-LDM as Decorator

We propose GM-LDM to model the image sequence distribution $p(\mathcal{X}|(\mathcal{C}, y))$ and generate immediate 3D Gaussians as the joint 3D scene representation. The GM-LDM, fine-tuned from the Stable Diffusion model with a slightly modified architecture, leverages its image generation prior to enhance 3D scene generation. For efficiency, diffusion is applied to a sparse-view subset of the camera trajectory, significantly reducing computational overhead. During training, sparse-view images are processed through the frozen latent encoder $\mathcal{E}$ to obtain multi-view latents $\mathcal{Z} \sim \mathbb{R}^{N \times c \times h \times w}$, where $N \leq M$. Noise is then added to these multi-view latents $\mathcal{Z}$ to produce noisy latents $\mathcal{Z}_t$.

2D-based Denoising. The noisy multi-view latents $\mathcal{Z}_t$ are fed into the latent denoising network Z_θ in parallel. For convenience, we modify the ϵ-prediction of the original Stable Diffusion model to x_0-prediction. The denoised multi-view latents are obtained as $\{\hat{\mathcal{Z}}, \mathcal{F}\} = Z_\theta(\{\mathcal{Z}_t\}, \mathcal{C}', y, t)$, where $\mathcal{F}$ represents additional multi-view features for enhanced 3D information. Sparse-view cameras $\mathcal{C}'$ are integrated into the network by combining the ray-maps $(\boldsymbol{o} \times \boldsymbol{d}, \boldsymbol{d})$ [7] with the noisy latents, where $\boldsymbol{o}$ and $\boldsymbol{d}$ denote the origin and direction of pixel-aligned rays, respectively. We replace self-attention blocks in the original 2D latent denoising network with cross-view self-attention blocks [3] to better capture multi-view correlations. The denoised multi-view latents $\hat{\mathcal{Z}}$ are supervised using a simple
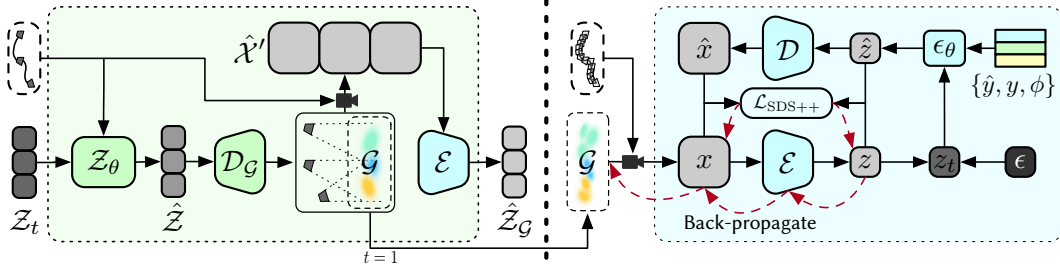
Figure 5: **Left**: Architecture of GM-LDM. The model is fine-tuned from a 2D LDM with minor modifications, performing rendering-based denoising for generating initial 3D Gaussians. **Right**: Pipeline of calculating SDS++ loss, which refines the 3D Gaussians with the original 2D LDM.

multi-view latent diffusion objective:

$$\mathcal{L}_{2d} = \mathbb{E}_{\mathcal{X},c,y,\epsilon,t}\left[\|\mathcal{Z} - \hat{\mathcal{Z}}\|_2^2\right]. \tag{6}$$

Rendering-based Denoising. To generate 3D Gaussians for rendering-based denoising, the denoised multi-view latents $\hat{\mathcal{Z}}$ and additional features $\mathcal{F}$ are input into a Gaussians decoder $\mathcal{D}_{\mathcal{G}}$. This decoder outputs Gaussian features $\{\tau_i, \boldsymbol{q}_i, \boldsymbol{s}_i, \alpha_i, \boldsymbol{c}_i\} = \mathcal{D}_{\mathcal{G}}(\hat{\mathcal{Z}}_i, \mathcal{F}_i)$, where τ_i, $\boldsymbol{q}_i$, $\boldsymbol{s}_i$, α_i, and $\boldsymbol{c}_i$ represent the depth, rotation quaternion, scale, opacity, and spherical harmonics coefficients of 256×256 3D Gaussians for view i, respectively. The Gaussians decoder $\mathcal{D}_{\mathcal{G}}$ is initialized with the weights of the original latent decoder D, with re-initialized first and last convolutional layers to handle the additional features and specific Gaussian channels. The predicted depth is then converted into pixel-aligned Gaussian positions $\boldsymbol{\mu}_i = \boldsymbol{o}_i + \tau_i \boldsymbol{d}_i$. The multi-view 3D Gaussians $\mathcal{G} = \{\boldsymbol{\mu}_i, \boldsymbol{q}_i, \boldsymbol{s}_i, \alpha_i, \boldsymbol{c}_i\}_{i=1}^N$ are concatenated to jointly represent the 3D scene. During training, views are randomly sampled from the dense-view camera trajectory to supervise the predicted 3D Gaussians in image space, ensuring consistent and accurate 3D scene representation:

$$\mathcal{L}_{3d} = \mathbb{E}_{x,c,y,\epsilon,t}\left[\ell(x, \mathcal{R}(\mathcal{G}, c))\right], \tag{7}$$

where $\mathcal{R}$ is the rendering function, $\ell(\cdot, \cdot)$ is a reconstruction loss penalizing the difference between images, and $(x, c) \in (\mathcal{X}, \mathcal{C})$ is the ground truth of an image and camera pair from the dense views. We use a combination of MSE loss and LPIPS [74] loss for the reconstruction loss ℓ, similar to the original reconstruction loss of the original auto-encoder. The total training loss is simply the sum of the above losses: $\mathcal{L} = \mathcal{L}_{2d} + \mathcal{L}_{3d}$. This approach leverages the strengths of multi-view data and pixel-aligned Gaussian representations to enhance the fidelity and coherence of the generated 3D scenes. By fine-tuning from a robust 2D LDM and using a sparse-view subset, we strike a balance between performance, efficiency, and generalizability, enabling the generation of coherent 3D scenes. As shown in Fig. 5 (Left), during inference, images are rendered with the input cameras $\mathcal{C}'$ and encoded by the latent encoder $\mathcal{E}$ to obtain Gaussian-driven denoised latents:

$$\hat{\mathcal{Z}}_{\mathcal{G}} = \mathcal{E}(\hat{\mathcal{X}}'), \text{where } \hat{\mathcal{X}}' = \mathcal{R}(\mathcal{G}, \mathcal{C}'). \tag{8}$$

Inspired by Dual3D [75], GM-LDM inference can toggle between 2D-based and rendering-based denoising. The 2D-based denoising offers better generalization, aligning closely with the original Stable Diffusion, while rendering-based denoising ensures superior 3D consistency due to its immediate joint 3D representation. The 3D Gaussians generated in the final denoising step serve as the initial 3D scene for subsequent refinement.

Collecting and annotating real-world multi-view datasets is laborious, often resulting in limited diversity and quantity, which hinders generalization for open-world texts. To address this, we follow MVDream [3] and collaboratively train the GM-LDM using both multi-view and 2D datasets to enhance generalization. We treat single-view images as a special case of multi-view images with $N = M = 1$ and apply the same rendering process and training losses, which increases the diversity of training data, thereby improving the model's ability to generalize across diverse scenarios.

4.4 SDS++ Loss as Detailer

To enhance the details and visual quality of the 3D Gaussians, we propose the SDS++ loss, leveraging the 2D diffusion prior for refinement. Our research on existing SDS-based methods identifies three key points crucial for success, this includes: (1) **Appropriate Target Distribution** [26]: This ensures that the rendered images align effectively with the textual conditions, avoiding over-smoothing and over-saturation. (2) **Adaptive Estimation of the Current Distribution** [26, 76]: This provides a counter optimization objective, pushing the rendered images away from the current distribution to enhance details. (3) **Latent-space and Image-space Objectives** [72]: Combining these objectives helps prevent noisy or over-smoothing artifacts that may arise from using only one of them. During refining, we first render the 3D Gaussians $\mathcal{G}$ with a randomly sampled camera c from the continuous interpolated camera trajectory to produce an image $x = \mathcal{R}(\mathcal{G}, c)$. This image is encoded into the latent space by $\mathcal{E}(x) = z$, then disturbed with randomly sampled noise and timestep t to produce a noisy latent z_t. The 2D diffusion model ϵ_θ then predicts the denoised latent $\hat{z}$ from z_t. As illustrated in Fig. 5 (Right), the proposed SDS++ loss can be formulated by:

$$\mathcal{L}_{\text{SDS++}} = \mathbb{E}_{t,c,\epsilon}\left[w(t)\frac{\sqrt{\bar{\alpha}_t}}{\sqrt{1 - \bar{\alpha}_t}}\left(\lambda_z\|z - \hat{z}\|_2^2 + \lambda_x\|x - \hat{x}\|_2^2\right)\right], \tag{9}$$

where λ_z and λ_x are the weights for latent-space and image-space objectives, respectively, $\hat{z}$ is the predicted latent, and $x = \mathcal{D}(\hat{z})$ is the predicted image. The predicted latent can be derived by Eq. 4. We use a compositional predictions $\hat{\epsilon}$ as follows:

$$\hat{\epsilon} = \hat{\epsilon}_{\text{trg}} - \hat{\epsilon}_{\text{src}} + \epsilon. \tag{10}$$

Instead of setting $\hat{\epsilon}_{\text{src}} = \epsilon$ as in the standard SDS loss, we introduce a learnable source prediction $\hat{\epsilon}_{\text{src}}$ for adaptive estimation of the current distribution:

$$\hat{\epsilon}_{\text{src}} = \epsilon_\theta(z_t, \hat{y}, t), \tag{11}$$

where $\epsilon_\theta(z_t, \hat{y}, t)$ uses a learnable text embedding $\hat{y}$ to efficiently estimate the current distribution. This approach leverages the original latent denoising network [76] and is trained by minimizing $\|\epsilon_\theta(z_t, \hat{y}, t) - \epsilon\|_2^2$ along with the refining process. The target prediction employs classifier-free guidance for improved text alignment:

$$\hat{\epsilon}_{\text{trg}} = \omega_{\text{cfg}} \cdot (\epsilon_\theta(z_t, y, t) - \epsilon_\theta(z_t, \phi, t)) + \epsilon_\theta(z_t, \phi, t), \tag{12}$$

where ω_{cfg} is the classifier-free guidance scale. SDS++ loss integrates the above three key points, ensuring efficient and realistic refinement of 3D Gaussians.

5 Experiments

5.1 Implementation Details

We utilize the MVImgNet [19] for object-level and DL3DV10K [20] for scene-level real-world multi-view datasets. Text prompts for each scene are generated using the multi-modal large language model InternLM-XComposer [77]. To enhance generalization, we incorporate the 2D dataset LAION [78]. For GM-LDM, we set the lengths of dense and sparse views to $M = 29$ and $N = 8$, respectively. The classifier-free guidance scale is 7.5 for 2D-based denoising and 1 for rendering-based denoising to ensure 3D consistency. Following Dual3D [75], we balance 3D consistency and generalization by using 1/10 rendering-based denoising steps. Image and latent resolutions are set to 256 and 32, respectively. For SDS++ loss, the weights for latent-space and image-space losses are $\lambda_z = 1$ and $\lambda_x = 0.01$. ω_{cfg} is set to 7.5, with refining iterations set to 1000. Generating a scene takes approximately 5 minutes. Further implementation details are provided in Appendix A.

5.2 Generation Results.

We show the generation results of camera trajectories and image sequences for various text prompts in Fig. 6. For object-level prompts, the generated camera trajectories typically circle and face the objects, aligning well with the distribution in MVImgNet. In contrast, scene-level prompts yield more diverse and complex camera trajectories, showcasing the effectiveness of our Traj-DiT model. Our method also generates realistic images across different types of prompts, demonstrating the effectiveness and generalization ability of GM-LDM and SDS++ loss. Additional generation results are available in Appendix D. These results highlight the robustness of our approach in handling both object-level and scene-level prompts for 3D scene generation.
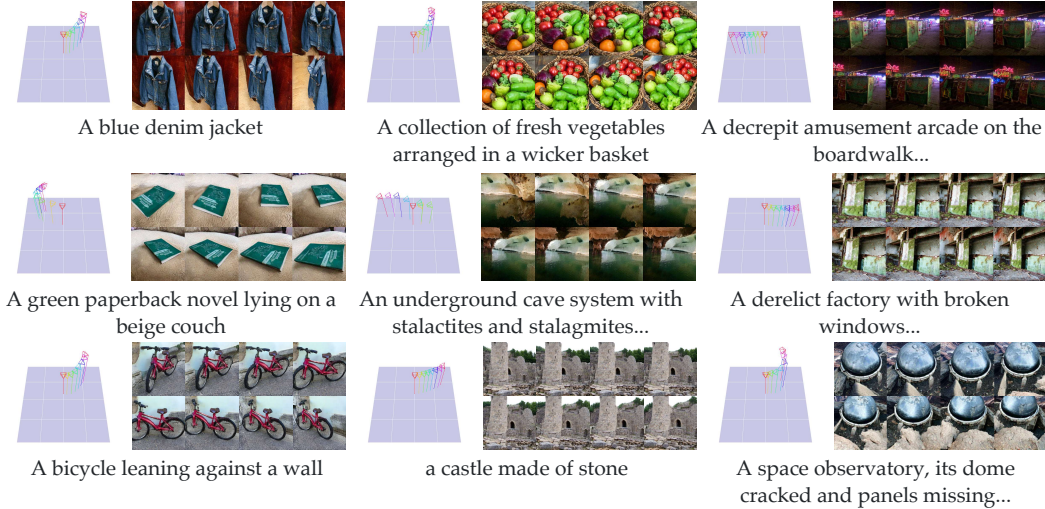
Figure 6: Generation results of Director3D for both camera trajectories and image sequences.
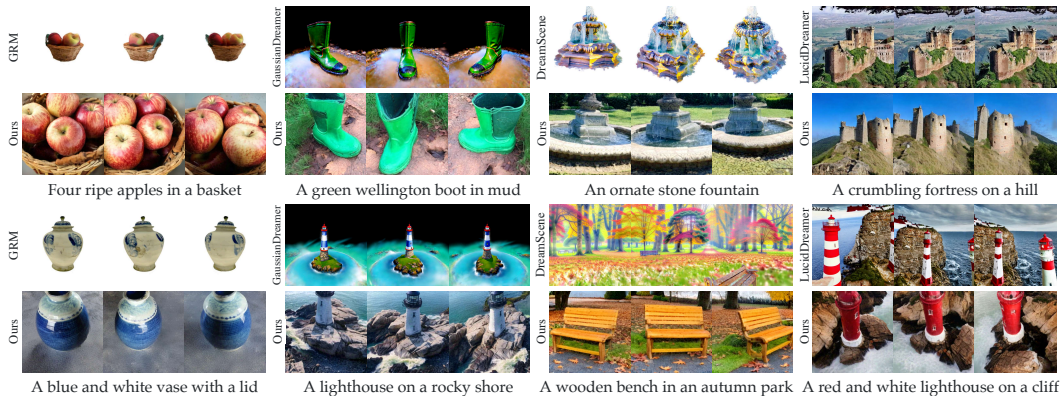

Figure 7: Qualitative comparison between Director3D and different baselines.

5.3 Qualitative Comparison.

We qualitatively compare our method with several baseline methods, as shown in Fig. 7. (1) GRM [38] is a feed-forward text-to-3D generation method for 3D Gaussians using multi-view images generated by 2D-based diffusion models. It supports only object-level 3D generation. Our comparisons using object descriptions show that GRM produces unrealistic 3D Gaussians limited to objects, due to its training on synthetic datasets. In contrast, our method generates high-quality 3D scenes with both objects and backgrounds. (2) GaussianDreamer [11] is a state-of-the-art SDS-based method for 3D Gaussians, integrating priors from both 2D and 3D diffusion models. While it can generate objects with ground layers, it tends to produce over-saturated textures. Our method, however, generates more realistic scenes with better handling of shadows, lighting, and material reflections. (3) DreamScene [17] uses Formation Pattern Sampling and strategic camera sampling for 3D Gaussians. Since it is not open-sourced, we use examples from its project page. Although it generates scene-wide consistent 3D scenes, the results are overly saturated and cartoonish. (4) LucidDreamer [16], based on Text2NeRF [14], uses 2D foundation models for 3D Gaussians. While it can generate photo-realistic textures, the multi-view consistency is poor, with visible artifacts at object edges due to inaccurate monocular depth estimation. It also struggles with excessive object generation from descriptive prompts due to reliance on single-view inpainting. These comparisons highlight the superior performance of the proposed Director3D for realistic 3D generation.

Table 1: Quantitative comparison of different object-centric models with text prompts in T3Bench.

Method	BRISQUE ↓	NIQE ↓	CLIP-Score ↑
DreamFusion [1]	90.2	10.48	67.4
Magic3D [79]	92.8	11.20	72.3
LatentNerf [80]	88.6	9.19	68.1
SJC [2]	82.0	10.15	61.5
Fantasia3D [81]	69.6	7.65	66.6
ProlificDreamer [26]	61.5	7.07	69.4
Ours *w/o* refining	37.1	6.41	80.0
Ours	**32.3**	**4.35**	**85.5**

Table 2: Quantitative comparison of scene-level models with 64 prompts

Method	NIQE ↓	CLIP-Score ↑	Inference Time (min) ↓
GaussianDreamer [11]	6.96	71.8	15
ZeroNVS [82]	9.84	67.2	90
LucidDreamer-LLFF [16]	3.53	83.3	40
LucidDreamer-HeadBang [16]	3.61	82.9	40
LucidDreamer-BackForth [16]	**3.40**	74.2	40
Ours	4.09	**83.9**	**5**

5.4 Quantitative Comparison.

We present a quantitative comparison between our framework and several baseline models in Tab. 1. For this experiment, we use the Single-Object-with-Surroundings[3] set of T3Bench [83], which contains 100 prompts closely matching the descriptions in MVImgNet. The quantitative results are evaluated using CLIP-Score [84], NIQE [85], and BRISQUE [86] metrics. For each 3D scene generated by different methods, we render a video and uniformly sample 36 frames to calculate the average score for each metric. For baselines without adaptive camera trajectories, videos are rendered by circling around the 3D representations at a fixed elevation. The BRISQUE and NIQE results indicate that our method significantly outperforms existing baseline models in terms of image quality. Additionally, the CLIP-Score shows our method's superior ability to align generated images with their textual descriptions, even without refining. These results underscore the robustness and effectiveness of our framework in generating high-quality, semantically aligned 3D scenes.

We also conduct a quantitative comparison with three scene-level baselines for 32 object-centric prompts and 32 scene-level prompts in the Tab. 2. Our method achieves the highest CLIP-Score and the second-best NIQE Score with the shortest inference time. Without adaptively generated camera trajectories (Traj-DiT) and a high-performance multi-view diffusion model with immediate 3D representation (GM-LDM), ZeroNVS exhibits deteriorated visual quality from multiple viewpoints and requires a time-consuming SDS optimization process from scratch for each scene. LucidDreamer attains the best NIQE Score; However, we observe that it is plagued by multi-view inconsistency, visible artifacts at edges, and excessive objects as shown in Fig. 7.

5.5 User-specific Camera Trajectories

Director3D supports the utilization of both pre-generation and post-generation user-specific camera trajectories. For pre-generation user-specific camera trajectories, users are capable of employing user-specific camera trajectories instead of the generated camera trajectories from Traj-DiT for 3D scene generation. For post-generation user-specific camera trajectories, users can render novel views by providing novel cameras after generating the 3D scene. We develop an interactive demo for visualizing the generated camera trajectories and 3D Gaussians, which is also capable of rendering the 3D Gaussians with novel cameras, as shown in Fig. 8.

---

[3] `https://github.com/THU-LYJ-Lab/T3Bench/blob/main/data/prompt_surr.txt`

| distance view | novel views | distance view | novel views |

A deep lake that mirrors the sky and cliffs around it...

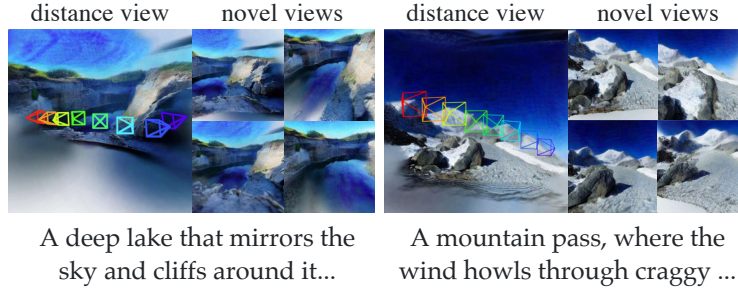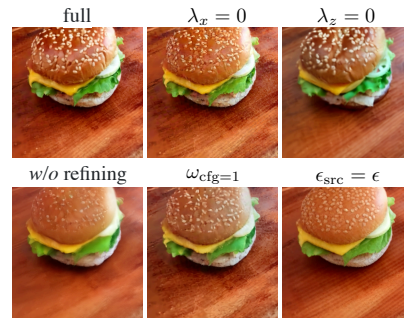A mountain pass, where the wind howls through craggy ...

Figure 8: Screenshots of the interactive demo for visualizing generated camera trajectories and 3D Gaussians of Director3D. The frames are rendered with novel cameras.

5.6 Ablation Study

Director3D is formulated to a sequential process to model the joint distribution of camera trajectories and image sequences. The Traj-DiT component generates the base camera trajectory for each scene, while the GM-LDM provides the initial 3D scene. These two models are crucial for producing meaningful 3D scenes. Our ablation study primarily focuses on the SDS++ loss to highlight its significance. As shown in Fig. 9, we conduct comprehensive experiments to analyze the impact of the SDS++ loss. First, we remove the refining process entirely. Although the initial 3D Gaussians generated by GM-LDM match the input text, the visual quality is unsatisfactory, with missing details. This is expected due to the limited diversity and quantity of the multi-view dataset used for training GM-LDM. Setting $\epsilon_{src} = \epsilon$ degrades the SDS++ loss into SDS+ loss [72]. The results show a significant decrease in visual quality and an

Figure 9: Ablation of SDS++ loss

over-smoothing issue, highlighting the importance of adaptive estimation of the current distribution. Setting $\omega_{cfg} = 1$ turns the SDS++ loss into the LODS loss [76] with an additional image-space objective. This results in noisy details, as the conditional noise prediction alone does not provide a clear optimization direction. By setting $\lambda_x = 0$ and $\lambda_z = 0$ respectively, we observe that using only the latent-space objective leads to noisy details and artifacts, while using only the image-space objective results in missing details.

The full model, incorporating the proposed SDS++ loss, achieves the best visual quality with clear and realistic details. These findings underscore the importance of each component in the SDS++ loss and its role in refining 3D Gaussians. Additional ablation studies, including qualitative and quantitative of more cases and those with randomly generated trajectories, are provided in Appendix E.

6 Conclusion

In this paper, we propose an open-world text-to-3D generation framework capable of generating real-world 3D scenes with adaptive camera trajectory, named Director3D. We first introduce a Cinematographer (*i.e.*, Traj-DiT) that can generate dense-view camera trajectories given texts. Then, a Decorator (*i.e.*, GM-LDM) and a Detailer (*i.e.*, SDS++ loss) are proposed for initial generation and further refining, respectively, with 3D Gaussians as the 3D scene representation. We demonstrate the effectiveness of our method with extensive experiments. We believe our work makes essential contributions to the text-to-3D generation community, especially in discovering the potential of leveraging real-world multi-view datasets for realistic 3D generation. Our future works include improving the generation scope, boosting model efficiency and quality, and leveraging more datasets.

7 Acknowledgements

References

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

[2] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A. Yeh, and Greg Shakhnarovich. Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. *arXiv preprint arXiv:2212.00774*, 2022.

[3] Yichun Shi, Peng Wang, Jianglong Ye, Long Mai, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. *arXiv preprint arXiv:2308.16512*, 2023.

[4] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision (ECCV)*, 2020.

[5] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)*, 41(4):1–15, 2022.

[6] Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. Lrm: Large reconstruction model for single image to 3d. *arXiv preprint arXiv:2311.04400*, 2023.

[7] Yinghao Xu, Hao Tan, Fujun Luan, Sai Bi, Peng Wang, Jiahao Li, Zifan Shi, Kalyan Sunkavalli, Gordon Wetzstein, Zexiang Xu, et al. Dmv3d: Denoising multi-view diffusion using 3d large reconstruction model. *arXiv preprint arXiv:2311.09217*, 2023.

[8] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics (SIGGRAPH)*, 42(4), July 2023.

[9] Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. *arXiv preprint arXiv:2309.16653*, 2023.

[10] Zilong Chen, Feng Wang, Yikai Wang, and Huaping Liu. Text-to-3d using gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[11] Taoran Yi, Jiemin Fang, Junjie Wang, Guanjun Wu, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Qi Tian, and Xinggang Wang. Gaussiandreamer: Fast generation from text to 3d gaussians by bridging 2d and 3d diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

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

[13] Kai Zhang, Sai Bi, Hao Tan, Yuanbo Xiangli, Nanxuan Zhao, Kalyan Sunkavalli, and Zexiang Xu. Gs-lrm: Large reconstruction model for 3d gaussian splatting. *arXiv preprint arXiv:2404.19702*, 2024.

[14] Jingbo Zhang, Xiaoyu Li, Ziyu Wan, Can Wang, and Jing Liao. Text2nerf: Text-driven 3d scene generation with neural radiance fields. *arXiv preprint arXiv:2305.11588*, 2023.

[15] Lukas Höllein, Ang Cao, Andrew Owens, Justin Johnson, and Matthias Nießner. Text2room: Extracting textured 3d meshes from 2d text-to-image models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7909–7920, October 2023.

[16] Jaeyoung Chung, Suyoung Lee, Hyeongjin Nam, Jaerin Lee, and Kyoung Mu Lee. Luciddreamer: Domain-free generation of 3d gaussian splatting scenes. *arXiv preprint arXiv:2311.13384*, 2023.

[17] Haoran Li, Haolin Shi, Wenli Zhang, Wenjun Wu, Yong Liao, Lin Wang, Lik hang Lee, and Pengyuan Zhou. Dreamscene: 3d gaussian-based text-to-3d scene generation via formation pattern sampling. *arXiv preprint arXiv:2404.03575*, 2024.

[18] Shijie Zhou, Zhiwen Fan, Dejia Xu, Haoran Chang, Pradyumna Chari, Suya Bharadwaj, Tejas You, Zhangyang Wang, and Achuta Kadambi. Dreamscene360: Unconstrained text-to-3d scene generation with panoramic gaussian splatting. *arXiv preprint arXiv:2404.06903*, 2024.

[19] Xianggang Yu, Mutian Xu, Yidan Zhang, Haolin Liu, Chongjie Ye, Yushuang Wu, Zizheng Yan, Chenming Zhu, Zhangyang Xiong, Tianyou Liang, et al. Mvimgnet: A large-scale dataset of multi-view images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9150–9161, 2023.

[20] Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22160–22169, 2024.

[21] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13142–13153, 2023.

[22] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16123–16133, 2022.

[23] Ajay Jain, Ben Mildenhall, Jonathan T Barron, Pieter Abbeel, and Ben Poole. Zero-shot text-guided object generation with dream fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 867–876, 2022.

[24] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning (ICML)*, pages 8748–8763. PMLR, 2021.

[25] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A Yeh, and Greg Shakhnarovich. Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12619–12629, 2023.

[26] Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolific-dreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *arXiv preprint arXiv:2305.16213*, 2023.

[27] Oren Katzir, Or Patashnik, Daniel Cohen-Or, and Dani Lischinski. Noise-free score distillation. *arXiv preprint arXiv:2310.17590*, 2023.

[28] Zi-Xin Zou, Weihao Cheng, Yan-Pei Cao, Shi-Sheng Huang, Ying Shan, and Song-Hai Zhang. Sparse3d: Distilling multiview-consistent diffusion for object reconstruction from sparse views. *arXiv preprint arXiv:2308.14078*, 2023.

[29] Sherwin Bahmani, Ivan Skorokhodov, Victor Rong, Gordon Wetzstein, Leonidas Guibas, Peter Wonka, Sergey Tulyakov, Jeong Joon Park, Andrea Tagliasacchi, and David B Lindell. 4d-fy: Text-to-4d generation using hybrid score distillation sampling. *arXiv preprint arXiv:2311.17984*, 2023.

[30] Jinbo Wu, Xiaobo Gao, Xing Liu, Zhengyang Shen, Chen Zhao, Haocheng Feng, Jingtuo Liu, and Errui Ding. Hd-fusion: Detailed text-to-3d generation leveraging multiple noise estimation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3202–3211, 2024.

[31] Zexiang Liu, Yangguang Li, Youtian Lin, Xin Yu, Sida Peng, Yan-Pei Cao, Xiaojuan Qi, Xiaoshui Huang, Ding Liang, and Wanli Ouyang. Unidream: Unifying diffusion priors for relightable text-to-3d generation. *arXiv preprint arXiv:2312.08754*, 2023.

[32] Xiaoxiao Long, Yuan-Chen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, et al. Wonder3d: Single image to 3d using cross-domain diffusion. *arXiv preprint arXiv:2310.15008*, 2023.

[33] Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. Syncdreamer: Generating multiview-consistent images from a single-view image. *arXiv preprint arXiv:2309.03453*, 2023.

[34] Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. Point-e: A system for generating 3d point clouds from complex prompts. *arXiv preprint arXiv:2212.08751*, 2022.

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

[36] Zhicong Tang, Shuyang Gu, Chunyu Wang, Ting Zhang, Jianmin Bao, Dong Chen, and Baining Guo. Volumediffusion: Flexible text-to-3d generation with efficient volumetric encoder. *arXiv preprint arXiv:2312.11459*, 2023.

[37] J Ryan Shue, Eric Ryan Chan, Ryan Po, Zachary Ankner, Jiajun Wu, and Gordon Wetzstein. 3d neural field generation using triplane diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20875–20886, 2023.

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

[39] Jiabao Lei, Jiapeng Tang, and Kui Jia. Rgbd2: Generative scene synthesis via incremental view inpainting using rgbd diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023.

[40] Rafail Fridman, Amit Abecasis, Yoni Kasten, and Tali Dekel. Scenescape: Text-driven consistent scene generation. *arXiv preprint arXiv:2302.01133*, 2023.

[41] Dana Cohen-Bar, Elad Richardson, Gal Metzer, Raja Giryes, and Daniel Cohen-Or. Set-the-scene: Global-local training for generating controllable nerf scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 2920–2929, October 2023.

[42] Jaidev Shriram, Alex Trevithick, Lingjie Liu, and Ravi Ramamoorthi. Realmdreamer: Text-driven 3d scene generation with inpainting and depth diffusion. *arXiv preprint arXiv:2404.07199*, 2024.

[43] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances In Neural Information Processing Systems (NeurIPS)*, 33:6840–6851, 2020.

[44] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, 2022.

[45] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[46] Rene Ranftl, Katrin Lasinger, David Hafner, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP:1–1, 08 2020.

[47] Shitao Tang, Fuyang Zhang, Jiacheng Chen, Peng Wang, and Yasutaka Furukawa. Mvdiffusion: Enabling holistic multi-view image generation with correspondence-aware diffusion. *arXiv preprint arxiv:2307.01097*, 2023.

[48] Yikun Ma, Dandan Zhan, and Zhi Jin. Fastscene: Text-driven fast 3d indoor scene generation via panoramic gaussian splatting. *arXiv preprint arXiv:2405.05768*, 2024.

[49] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. In *35th Conference on Neural Information Processing Systems*, pages 27171–27183. Curran Assoicates, Inc., 2021.

[50] Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[51] Sara Fridovich-Keil and Alex Yu, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[52] Antoine Guédon and Vincent Lepetit. Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering. *arXiv preprint arXiv:2311.12775*, 2023.

[53] Mulin Yu, Tao Lu, Linning Xu, Lihan Jiang, Yuanbo Xiangli, and Bo Dai. Gsdf: 3dgs meets sdf for improved rendering and reconstruction. *arXiv preprint arXiv:2403.16964*, 2024.

[54] Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. *ACM Transactions on Graphics (SIGGRAPH)*, 2024.

[55] Tao Lu, Mulin Yu, Linning Xu, Yuanbo Xiangli, Limin Wang, Dahua Lin, and Bo Dai. Scaffold-gs: Structured 3d gaussians for view-adaptive rendering. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[56] Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Wang Xinggang. 4d gaussian splatting for real-time dynamic scene rendering. *arXiv preprint arXiv:2310.08528*, 2023.

[57] Zhan Li, Zhang Chen, Zhong Li, and Yi Xu. Spacetime gaussian feature splatting for real-time dynamic view synthesis. *arXiv preprint arXiv:2312.16812*, 2023.

[58] Zeyu Yang, Hongye Yang, Zijie Pan, and Li Zhang. Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting. In *International Conference on Learning Representations (ICLR)*, 2024.

[59] Yuheng Jiang, Zhehao Shen, Penghao Wang, Zhuo Su, Yu Hong, Yingliang Zhang, Jingyi Yu, and Lan Xu. Hifi4g: High-fidelity human performance rendering via compact gaussian splatting. *arXiv preprint arXiv:2312.03461*, 2023.

[60] Jin-Chuan Shi, Miao Wang, Hao-Bin Duan, and Shao-Hua Guan. Language embedded 3d gaussians for open-vocabulary scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5333–5343, 2024.

[61] Minghan Qin, Wanhua Li, Jiawei Zhou, Haoqian Wang, and Hanspeter Pfister. Langsplat: 3d language gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20051–20060, 2024.

[62] Yansong Qu, Shaohui Dai, Xinyang Li, Jianghang Lin, Liujuan Cao, Shengchuan Zhang, and Rongrong Ji. Goi: Find 3d gaussians of interest with an optimizable open-vocabulary semantic-space hyperplane. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 5328–5337, 2024.

[63] Jonas Kulhanek, Songyou Peng, Zuzana Kukelova, Marc Pollefeys, and Torsten Sattler. Wildgaussians: 3d gaussian splatting in the wild. *arXiv preprint arXiv:2407.08447*, 2024.

[64] Yuze Wang, Junyi Wang, and Yue Qi. We-gs: An in-the-wild efficient 3d gaussian representation for unconstrained photo collections. *arXiv preprint arXiv:2406.02407*, 2024.

[65] Yuedong Chen, Haofei Xu, Chuanxia Zheng, Bohan Zhuang, Marc Pollefeys, Andreas Geiger, Tat-Jen Cham, and Jianfei Cai. Mvsplat: Efficient 3d gaussian splatting from sparse multi-view images. *arXiv preprint arXiv:2403.14627*, 2024.

[66] David Charatan, Sizhe Li, Andrea Tagliasacchi, and Vincent Sitzmann. pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[67] Stanislaw Szymanowicz, Christian Rupprecht, and Andrea Vedaldi. Splatter image: Ultra-fast single-view 3d reconstruction. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[68] Jiahao Li, Hao Tan, Kai Zhang, Zexiang Xu, Fujun Luan, Yinghao Xu, Yicong Hong, Kalyan Sunkavalli, Greg Shakhnarovich, and Sai Bi. Instant3d: Fast text-to-3d with sparse-view generation and large reconstruction model. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024.

[69] OpenAI. Sora: Creating video from text. *https://openai.com/sora*, 2024.

[70] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances In Neural Information Processing Systems (NeurIPS)*, 35:36479–36494, 2022.

[71] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations (ICLR) 2014, Banff, AB, Canada*, 2014.

[72] Junzhe Zhu and Peiye Zhuang. Hifa: High-fidelity text-to-3d generation with advanced diffusion guidance. *arXiv preprint arXiv:2305.18766*, 2023.

[73] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4195–4205, October 2023.

[74] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 586–595, 2018.

[75] Xinyang Li, Zhangyu Lai, Linning Xu, Jianfei Guo, Liujuan Cao, Shengchuan Zhang, Bo Dai, and Rongrong Ji. Dual3d: Efficient and consistent text-to-3d generation with dual-mode multi-view latent diffusion. *arXiv preprint arXiv:2405.09874*, 2024.

[76] Xiaofeng Yang, Yiwen Chen, Cheng Chen, Chi Zhang, Yi Xu, Xulei Yang, Fayao Liu, and Guosheng Lin. Learn to optimize denoising scores for 3d generation. *arXiv preprint arXiv:2312.04820*, 2023.

[77] Pan Zhang, Xiaoyi Dong, Bin Wang, Yuhang Cao, Chao Xu, Linke Ouyang, Zhiyuan Zhao, Shuangrui Ding, Songyang Zhang, Haodong Duan, Wenwei Zhang, Hang Yan, Xinyue Zhang, Wei Li, Jingwen Li, Kai Chen, Conghui He, Xingcheng Zhang, Yu Qiao, Dahua Lin, and Jiaqi Wang. Internlm-xcomposer: A vision-language large model for advanced text-image comprehension and composition. *arXiv preprint arXiv:2309.15112*, 2023.

[78] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances In Neural Information Processing Systems (NeurIPS)*, 35:25278–25294, 2022.

[79] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 300–309, 2023.

[80] Gal Metzer, Elad Richardson, Or Patashnik, Raja Giryes, and Daniel Cohen-Or. Latent-nerf for shape-guided generation of 3d shapes and textures. *arXiv preprint arXiv:2211.07600*, 2022.

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

[82] Kyle Sargent, Zizhang Li, Tanmay Shah, Charles Herrmann, Hong-Xing Yu, Yunzhi Zhang, Eric Ryan Chan, Dmitry Lagun, Li Fei-Fei, Deqing Sun, and Jiajun Wu. ZeroNVS: Zero-shot 360-degree view synthesis from a single real image. *CVPR, 2024*, 2023.

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

[84] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. CLIPScore: a reference-free evaluation metric for image captioning. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2021.

[85] Anish Mittal, Rajiv Soundararajan, and Alan C. Bovik. Making a "completely blind" image quality analyzer. *IEEE Signal Processing Letters*, 20(3):209–212, 2013.

[86] Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. No-reference image quality assessment in the spatial domain. *IEEE Transactions on Image Processing*, 21(12):4695–4708, 2012.

[87] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, San Diega, CA, USA, 2015.

[88] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *9th International Conference on Learning Representations (ICLR) 2021, Virtual Event, Austria*, 2021.

[89] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[90] Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *International Conference on Computer Vision (ICCV)*, 2021.

[91] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. In *ACM Transactions on Graphics (SIGGRAPH)*, 2018.

[92] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

A Implemental Details

Datasets. For real-world multi-view datasets, we leverage MVImgNet [19] and DL3DV10K [20] datasets for object-level and scene-level captures, respectively. The MVImgNet dataset is a large-scale object-level dataset, comprising 219,188 videos across 238 classes[4]. The DL3DV10K dataset consists of 10K scene-level videos with various indoor, outdoor, urban, and suburban environments[5].

Traj-DiT setup. Our Traj-DiT is trained with the Adam optimizer [87], a constant learning rate of $5e^{-5}$ and $(\beta_1, \beta_2) = (0.9, 0.95)$. The batch size is set to 128. The number of the DiT Blocks is 8 and the hidden size is 512. Training takes about 2 days with 1 NVIDIA Tesla A100 GPUs for 50K iterations. We use 1000 steps during training and reduce it to 100 steps with DDIM [88] during inference.

GM-LDM setup. Our GM-LDM is trained with the Adam optimizer, a constant learning rate of $1e^{-4}$ and $(\beta_1, \beta_2) = (0.9, 0.95)$. The batch size is also set to 128. Training takes about a week with 16 NVIDIA Tesla A100 GPUs for 150K iterations. We use Stable Diffusion v2.1[6] as our initial model. The noise schedule and other hyper-parameters are consistent with the Traj-DiT.

SDS++ loss setup. For refining, the learning rates of the Gaussian parameters $\boldsymbol{\mu}$, $\boldsymbol{q}$, $\boldsymbol{s}$, α, and $\boldsymbol{c}$ are set to be 0.0001, 0.01, 0.001, 0.01, and 0.01, respectively. The learning rate of the learnable text embedding $\hat{y}$ is set to be 0.001. The timestep t is annealing from $0.75T$ to $0.02T$ with a square root schedule, where T is the total denoising steps. We use Stable Diffusion v2.1 as our refining model as well. The rendering resolution is set to be 512 during refinement.

B Discussions

Ways to Model $p(\mathcal{X}, \mathcal{C})$**.** There are different approaches to modeling the joint distribution of image sequences and camera trajectories for real-world captures. We choose to independently model the distribution of camera trajectories and the conditional distribution of image sequences based on the camera trajectories. The image sequence is directly rendered with a joint 3D representation by the proposed GM-LDM, ensuring 3D consistency. Another alternative is to first directly model the image sequences and then infer the camera trajectories from the image sequences. Concurrent work, GS-LRM [13], adopts this approach. Specifically, it uses pose-free image sequences generated by SORA, then recovers camera information through COLMAP [89], and finally performs sparse-view 3D reconstruction. However, this approach heavily relies on the 3D consistency of the pose-free image sequences and the accurate estimation of the camera parameters. One can also choose to directly model the joint distribution of camera trajectories and image sequences. However, this requires further exploration of network architectures and camera-differentiable 3DGS operator. We will explore the potential of this approach in future work.

Why Director3D Needs a Detailer? The core reason remains the limitations of multi-view datasets in terms of diversity and quantity, leading to biased and sparse distributions. Although MVImgNet has a large quantity of scenes, it only consists of 238 categories, with the majority being simple objects staged in indoor scenes. While DL3DV-10K exhibits good diversity, it only consists of 10,000 scenes in total. Therefore, even under the 2D data and model priors, it is still challenging for GM-LDM to meet the quality requirements for directly generating realistic 3D scenes with details. This limitation can be alleviated by introducing more diverse and larger multi-view datasets.

Limitations. Although the proposed Director3D possesses the capability to generate real-world 3D scenes with adaptive camera trajectories, it still has some limitations. First, the supported views of our GM-LDM are currently limited, which restricts the range of perspectives in the generated 3D scenes. Further, because the GM-LDM is currently trained with only two real-world datasets, the reliance on an additional refining process for open-world generalization limits the efficiency of our framework, which may be alleviated by introducing a wider variety of datasets [90, 91, 92]. Similar to text-to-image diffusion models, the success rate of our model decreases when generating with complicated and compositional prompts, objects with exact numbers, and articulated objects, as shown in Fig. 10.

---

[4] https://github.com/GAP-LAB-CUHK-SZ/MVImgNet, under the MVImgNet Terms of Use
[5] https://github.com/DL3DV-10K/Dataset, under the DL3DV-10K Terms of Use
[6] https://huggingface.co/stabilityai/stable-diffusion-2-1-base

Stone statue of a knight
wearing a diamond crown and
holding a golden sword.

Six oranges in
a wooden basket.

A Tyrannosaurus Rex skeleton
in an exhibition hall.

Figure 10: Failure cases.

Table 3: Quantitative ablation study of SDS++ loss

Setting	NIQE ↓	CLIP-Score ↑
full	**4.10**	**83.7**
w/o refining	5.95	79.3
$\lambda_x = 0$	4.12	83.1
$\lambda_z = 0$	7.18	81.3
$\omega_{cfg} = 1$	4.26	79.6
$\epsilon_{src} = \epsilon$	4.27	80.4

Broader Impacts. This paper presents a framework whose goal is to advance the field of text-to-3D generation for efficiently generating realistic 3D scene assets. Since it is a 3D generative framework, it has the potential to create harmful 3D content if the user provides harmful text prompts as inputs. This harmful behavior can be avoided by integrating harmful content detection models as used in 2D generative models.

C   Diversity and Fine-grained Control

We showcase the diverse generation results with the same prompts in Fig. 11. The results show that Director3D is able to generate diverse camera trajectories and 3D scenes even with a single prompt.

We showcase the body generation results with fine-grained control (*i.e.*, gender and clothing) in Fig. 12. Note that there are no categories for human body in the multi-view datasets used for training the GM-LDM. The results show that Director3D is able to finely control the 3D scenes with text modifications.

D   More Generation Results

We further provide 40 cases of multi-view image results in Fig. 13 generated from wide-range text prompts to demonstrate the visual quality and generalization ability of our method.

E   More Ablation Studies

We provide a qualitative ablation study with more cases in Fig. 14. The corresponding quantitative ablation study is shown in the Tab. 3. Although the ablation with $\lambda_x = 0$ incurs a slight deterioration in metrics, the qualitative results are noisy in terms of visual quality. These results can further demonstrate the effectiveness of SDS++ loss.

Additionally, we showcase the generation results with randomly generated camera trajectories for some object-level prompts in Fig 15. The results are generated by the GM-LDM without refining. It can be observed that without scene-specific camera trajectories, the generated image sequences are unsatisfying, with weird perspectives or limited camera range. This ablation study further demonstrate the importance of scene-specific camera trajectories for our Director3D.
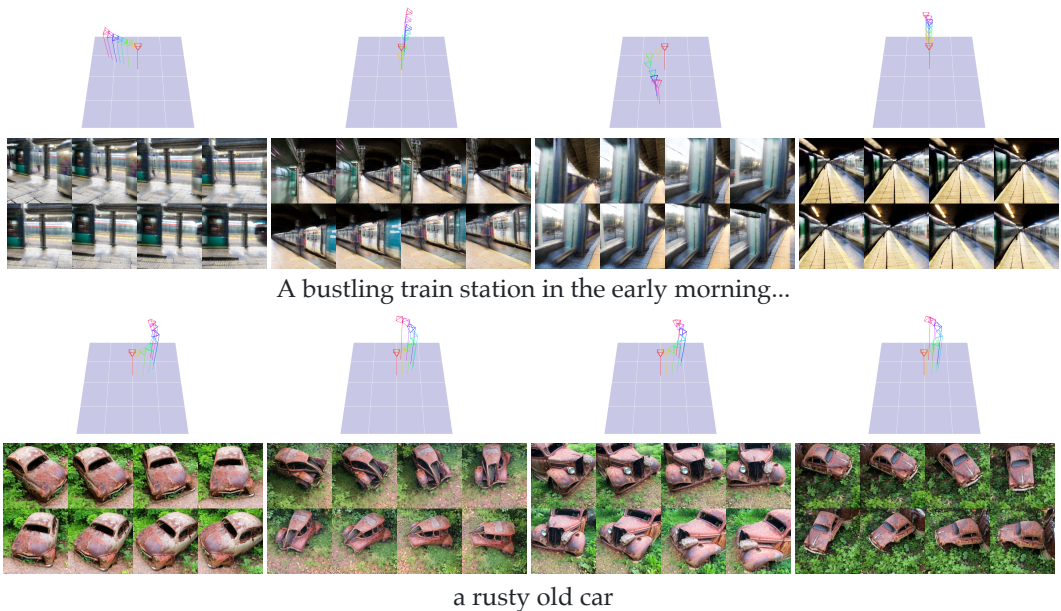
A bustling train station in the early morning...

a rusty old car

Figure 11: Generation results with diversity.

A man wearing a graphic T-shirt and cargo shorts

A woman wearing a floral blouse and denim shorts

A woman wearing a black blazer and a pencil skirt

A man wearing a checkered shirt and tan chinos

A woman wearing a leather jacket and black skinny jeans

Figure 12: Generation results with fine-grained control.

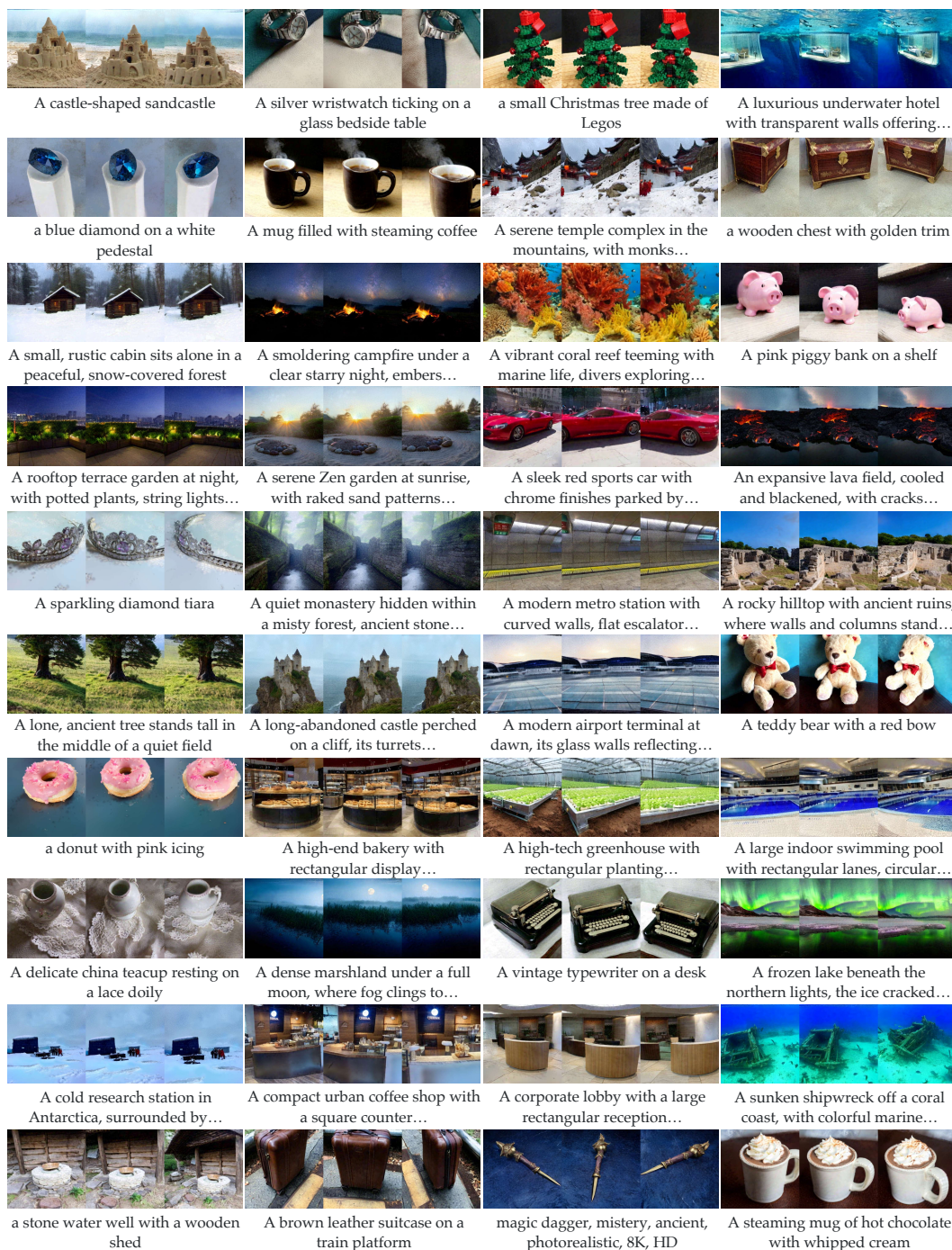| A castle-shaped sandcastle | A silver wristwatch ticking on a glass bedside table | a small Christmas tree made of Legos | A luxurious underwater hotel with transparent walls offering… |
a blue diamond on a white pedestal	A mug filled with steaming coffee	A serene temple complex in the mountains, with monks…	a wooden chest with golden trim
A small, rustic cabin sits alone in a peaceful, snow-covered forest	A smoldering campfire under a clear starry night, embers…	A vibrant coral reef teeming with marine life, divers exploring…	A pink piggy bank on a shelf
A rooftop terrace garden at night, with potted plants, string lights…	A serene Zen garden at sunrise, with raked sand patterns…	A sleek red sports car with chrome finishes parked by…	An expansive lava field, cooled and blackened, with cracks…
A sparkling diamond tiara	A quiet monastery hidden within a misty forest, ancient stone…	A modern metro station with curved walls, flat escalator…	A rocky hilltop with ancient ruins, where walls and columns stand…
A lone, ancient tree stands tall in the middle of a quiet field	A long-abandoned castle perched on a cliff, its turrets…	A modern airport terminal at dawn, its glass walls reflecting…	A teddy bear with a red bow
a donut with pink icing	A high-end bakery with rectangular display…	A high-tech greenhouse with rectangular planting…	A large indoor swimming pool with rectangular lanes, circular…
A delicate china teacup resting on a lace doily	A dense marshland under a full moon, where fog clings to…	A vintage typewriter on a desk	A frozen lake beneath the northern lights, the ice cracked…
A cold research station in Antarctica, surrounded by…	A compact urban coffee shop with a square counter…	A corporate lobby with a large rectangular reception…	A sunken shipwreck off a coral coast, with colorful marine…
a stone water well with a wooden shed	A brown leather suitcase on a train platform	magic dagger, mistery, ancient, photorealistic, 8K, HD	A steaming mug of hot chocolate with whipped cream

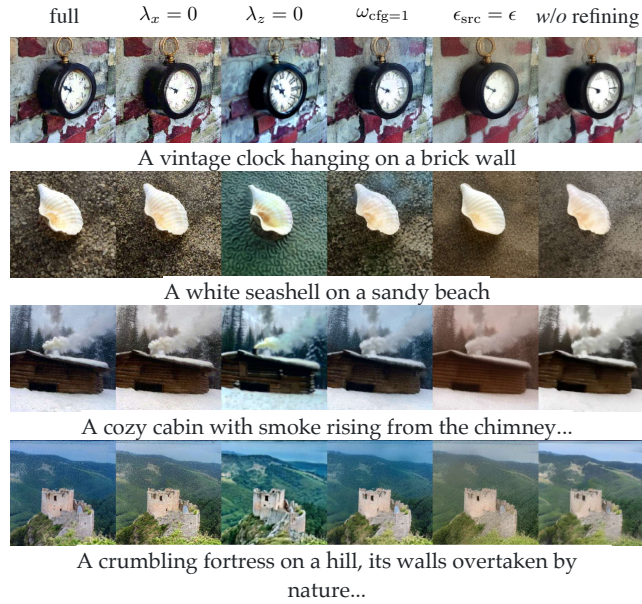Figure 13: More multi-view image results.

Figure 14: Additional qualitative ablation study of SDS++ loss for more cases.

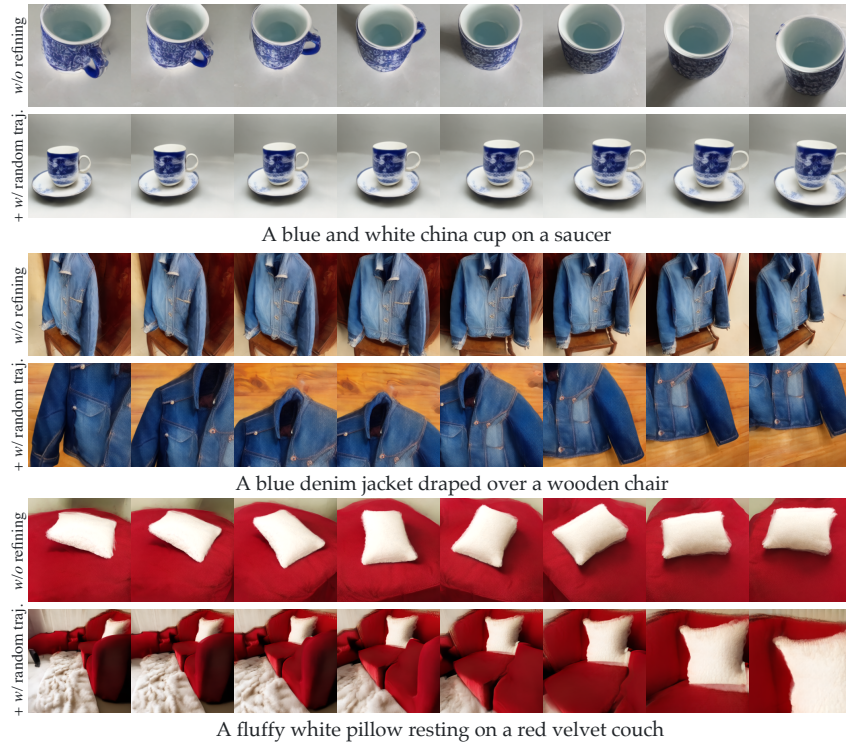
Figure 15: Generation results with randomly generated camera trajectories.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We claim clearly in Abstract and Sec. 1 about contributions and scope made in the paper.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See Appendix B

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: Our paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: See Sec. 5.1 and Appendix A

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: All datasets we use are public data, and our code will be open source in the future.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: See Sec. 5.1 and Appendix A

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: Since most of the experiments are visually qualitative evaluations, they cannot be converted into reports with quantitative error bars.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: See Appendix A

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: We have read the NeurIPS Code of Ethics in detail to make sure that our study meets the standards.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: See Appendix B.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [Yes]

Justification: See Appendix B.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All existing assets used in the paper are properly credited.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: All urls are anonymized.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: Our paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: Our paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.