ZERO-1-TO-G: TAMING PRETRAINED 2D DIFFUSION MODEL FOR DIRECT 3D GENERATION

Anonymous authors
Paper under double-blind review

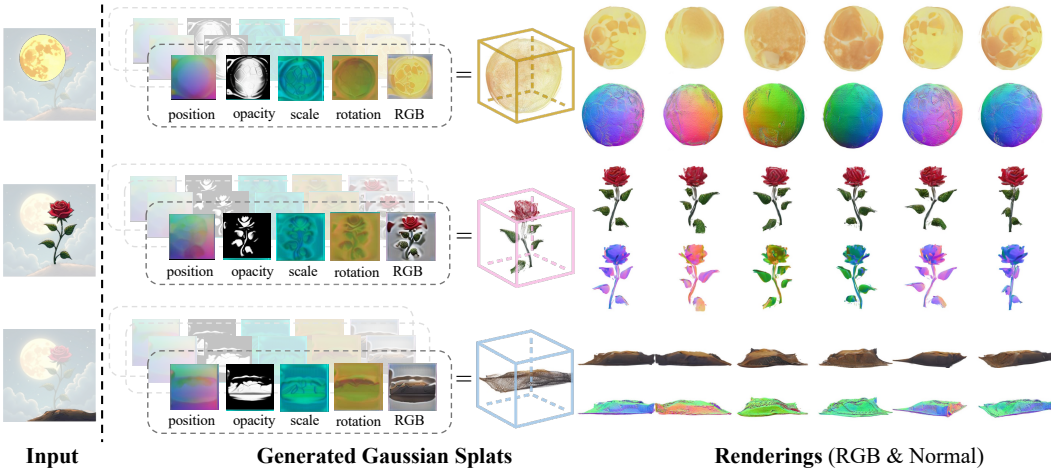

Figure 1: **Zero-1-to-G** tackles direct Gaussian splat generation from single images. By using pretrained 2D diffusion models, we are able to generalize to in-the-wild objects.

ABSTRACT

Recent advances in 2D image generation have achieved remarkable quality, largely driven by the capacity of diffusion models and the availability of large-scale datasets. However, direct 3D generation is still constrained by the scarcity and lower fidelity of 3D datasets. In this paper, we introduce *Zero-1-to-G*, a novel approach that addresses this problem by enabling direct single-view **image-to-3D** generation on Gaussian splats using **pretrained** 2D diffusion models. Our key insight is that Gaussian splats, a 3D representation, can be decomposed into multi-view images encoding different attributes. This reframes the challenging task of direct 3D generation within a 2D diffusion framework, allowing us to leverage the rich priors of pretrained 2D diffusion models. To incorporate 3D awareness, we introduce cross-view and cross-attribute attention layers, which capture complex correlations and enforce 3D consistency across generated splats. This makes *Zero-1-to-G* the first direct 3D generative model to effectively utilize 2D pretrained diffusion priors, enabling efficient training and improved generalization to unseen objects. Extensive experiments on both synthetic and in-the-wild datasets demonstrate superior performance in 3D object generation, offering a new approach to high-quality 3D generation.

1 INTRODUCTION

Single image to 3D generation is a pivotal challenge in computer vision and graphics, supporting various downstream applications such as virtual reality and gaming technologies. A primary difficulty lies in managing the uncertainty of unseen regions, as these areas represent a conditional distribution based on the visible portions of a 3D object. Recent advancements in diffusion models (Ho et al., 2020; Rombach et al., 2022) have demonstrated significant efficacy in capturing complex data distributions within images and videos, prompting researchers to harness these models for single image to 3D generation. Early efforts distilled 3D neural fields from pretrained 2D diffusion models via score distillation (Poole et al., 2022; Wang et al., 2023). However, these approaches necessitate

per-scene optimization, which is time-consuming and susceptible to multi-faced Janus problems. Subsequent research achieved feed-forward generation by fine-tuning pretrained models to generate multi-view images of the same object (Liu et al., 2023b; Shi et al., 2023a; Long et al., 2023; Liu et al., 2023c) and enabling indirect 3D generation through sparse-view reconstruction models (Li et al., 2023; Tang et al., 2024; Xu et al., 2024), and Cat3D Gao et al. (2024) further extends sparse view generation to dense view generation for better reconstruction. Although these two-stage methods enhance quality and efficiency, they often yield poor geometric fidelity and blurry renderings due to inconsistencies in multi-view images. To circumvent these limitations, recent methodologies have trained diffusion models directly on 3D representations (Liu et al., 2023d; Chen et al., 2023a; Zhang et al., 2024a; He et al., 2024; Nichol et al., 2022), thereby eliminating the reliance on multi-view images. However, direct 3D generation techniques necessitate training from scratch, requiring substantial computational resources and large 3D datasets, which remain scarce—three orders of magnitude less prevalent than 2D data.

In this paper, we propose a novel approach for direct 3D generation that unites the strengths of both worlds: it leverages the expressive power of 2D diffusion networks while maintaining the 3D structural consistency required for accurate 3D generation. Our key contribution is bridging the gap between Gaussian splats and natural images typically used in 2D generation tasks. While the original Gaussian splats consist of 14-channel images encoding various attributes, we decompose each of them into multiple 3-channel attribute images while preserving its 3D information (Sec. 3.1) This decomposition also enables efficient latent diffusion training by projecting the splatter images into the latent space of a pretrained VAE, making our method directly generate 3D structures within a pretrained 2D diffusion framework. To ensure strong 3D consistency, we introduce cross-view attention layers in Stable Diffusion to enable information exchange between different viewpoints and cross-attribute attention mechanisms to maintain coherence across Gaussian attributes within a splatter image (Sec. 3.2). Additionally, we fine-tune the VAE decoder to address the domain gap between splatter images and natural images, as we observed that splatter image quality is highly sensitive to pixel-level variations (Sec. 3.3). By leveraging pretrained priors, our method not only generalizes better to unseen objects but also improves training efficiency compared to existing 3D generation methods.

It is important to note that, although we generate multiview splatter images, our method produces more structurally consistent and higher-quality results compared to traditional two-stage multiview 3D generation approaches. In two-stage methods (Xu et al., 2024; Tang et al., 2024), strong pixel-level consistency is required in the first stage to ensure accurate reconstruction in the second stage, which is often difficult to achieve. Moreover, the first stage operates independently of the second, lacking coordination between the two, thereby exacerbating inconsistencies. In contrast, our approach directly generates the final 3D representation in a single stage, eliminating the need for pixel-level correspondence between multiview splatter images. Since splatter images can contain redundant information, and their spatial positions do not necessarily map to the final 3D positions, this single-stage process offers greater flexibility and robustness, ensuring a more consistent final 3D structure.

Overall, our contributions can be summarized as below:

- We present *Zero-1-to-G*, a novel direct 3D generative model for Gaussian splats that achieves excellent 3D consistency and superior rendering quality.
- We observe that Gaussian splats, as a 3D representation, can be decomposed into multi-view images representing different attributes, making them inherently compatible with backbones designed for 2D image generation.
- Through splatter image decomposition, we unleash the power of pretrained 2D diffusion models for direct 3D generation, enabling us to complete training more efficiently and have better generalization towards in-the-wild data.

2 Related Works

Optimization-based 3D Generation Dreamfusion (Poole et al., 2022) and subsequent works (Lin et al., 2023; Chen et al., 2023b; Wang et al., 2023) utilize a pretrained text-to-image diffusion model to optimize a 3D representation through score distillation. DreamGaussian (Tang et al., 2023) sig-

nificantly reduces training time by optimizing Gaussian splats. However, score distillation-based methods still require minutes of optimization per scene, as they must compare renderings with diffusion outputs from various viewpoints, which limits their generation speed. Additionally, these methods lack a clear understanding of geometry and viewpoint, resulting in multi-face problems.

Direct 3D Generation Significant efforts have been made to directly train diffusion models on various 3D representations, including point clouds (Luo & Hu, 2021; Zhou et al., 2021; Nichol et al., 2022; Jun & Nichol, 2023), meshes (Liu et al., 2023d), and neural fields (Chen et al., 2023a; Shue et al., 2023; Müller et al., 2023). However, these methods are typically constrained to category-level datasets and often struggle to generate high-quality assets. More recent approaches have begun encoding 3D assets into more compact latent representations (Zhang et al., 2023a; Lan et al., 2024; Zhao et al., 2023; Zhang et al., 2024b; Hong et al., 2024; Dong et al., 2024; **?**), enabling diffusion models to be trained more efficiently and enhancing generalization capabilities. Despite these advancements, direct 3D generative models are still primarily trained on synthetic 3D datasets like Objaverse (Deitke et al., 2024), which may hinder their ability to effectively handle more in-the-wild inputs.

More closely related to our work are GVGen (He et al., 2024) and GaussianCube (Zhang et al., 2024a), which investigate training diffusion models on Gaussian splats. Both approaches acknowledge the challenges of directly learning diffusion models from Gaussian splats and propose organizing the Gaussian points into a more structured volume. Different from their methods, we adopt a multi-view splatter image representation, enabling us to train diffusion models on Gaussian splats directly with high efficiency.

Finetuning Pretrained Diffusion models To enhance the 3D awareness of 2D pretrained diffusion models, MVDream (Shi et al., 2023b)integrates cross-view attention layers and fine-tunes them to produce multi-view renderings. As an application, LGM (Tang et al., 2024) and InstantMesh (Xu et al., 2024) utilize a pre-trained single-view to multi-view 2D diffusion model, transforming the single view generation problem to a multi-view reconstruction task. These methods adopt various compact 3D representations to adapt to the sparse view 2D input. This has greatly pushed the 3D generation to more complex data. However, this approach is constrained by the performance of the multi-view diffusion model, which often lacks strict multi-view consistency. This inconsistency can result in poor geometry and blurry textures in the final 3D reconstructions.

Some other works further leverage the power of pretrained diffusion models to generate data beyond the domain of natural images. For instance, JointNet (Zhang et al., 2023b) uses two diffusion models for joint RGB and depth prediction. Marigold (Ke et al., 2023) fine-tunes pretrained diffusion models for monocular depth estimation from single image input. Wonder3D (Long et al., 2023) generates both multi-view RGB and normal maps from a unified diffusion model equipped with a domain switcher. Similarly, GeoWizard (Fu et al., 2024) predicts depth and normals from a single image using cross-domain geometric self-attention to maintain geometric consistency. Following this line of work to adapting diffusion prior to images of other domains, our method aims to generate Gaussian splats represented as splatter images using pretrained 2D diffusion models, thereby enhancing the ability of direct 3D generation to tackle in-the-wild images.

3D Generation with Reconstruction-based methods Researchers have opted to train reconstruction-based models for highly efficient 3D generation Hong et al. (2023); Tochilkin et al. (2024); Woo et al. (2024); Zou et al. (2023); Xu et al. (2023). LRM (Hong et al., 2023) and TripoSR (Tochilkin et al., 2024) introduced a transformer-based model that directly output a triplane from a single image. The model is trained on million-scale data by comparing the renderings of the triplane with ground truth using regression-based loss. TriplaneGaussian (Zou et al., 2023) further used a hybrid triplane-Gaussian representation to greatly accelerate the rendering of the generated 3D assets. However, the main drawback of regression-based methods is their failure to account for the uncertainty in single-view to 3D generation. Although GECO (Wang et al., 2024) attempts to address this issue by distilling knowledge from multi-view diffusion models into a feedforward model, its results remain limited by the quality of the generated multi-view images.

Direct 3D Generation with 2D Diffusion Concurrent works Yan et al. (2024) and Elizarov et al. (2024) also propose methods of generating 3D with 2D diffusion models. They use UV atlas to encode the geometry and texture of 3D mesh, while we use Splatter Image Szymanowicz et al. (2023) as 3D representation. Since Omage Yan et al. (2024) uses a 12-channel UV atlas, its model

cannot utilize the pretraiend 2D diffusion prior and has to be trained from scratch, limiting its ability within the distribution of the training dataset. On the other hand, GIMDiffusion Elizarov et al. (2024) decomposes the UV atlas into separate geometry maps and albedo textures to match the 3-channel output of pretraiend 2D diffusion model. Still, they tackle the task of text-to-3D generation while we focus more on single-view image-to-3D reconstruction.

Direct 3D Generation with 2D Diffusion Concurrent works, Omage Yan et al. (2024) and GIMDiffusion Elizarov et al. (2024), also explore the utilization of 2D diffusion models for 3D generation but differ from ours in the representation and model architecture. Omage Yan et al. (2024) employs a 12-channel UV atlas to encode the whole 3D meshes, thus precluding the use of pretrained 2D diffusion models. As a result, Omage requires training from scratch on category-level datasets, confining its generative capability to the distribution of its training data. In contrast, GIMDiffusion Elizarov et al. (2024) adopts a similar idea of using pretrained 2D diffusion models through decomposition, but the utilization of pretrained 2D prior is only limited to the generation of albedo texture: it consists of a frozen pretrained model for albedo texture generation, and another architectural clone trained from scratch for geometry generation.

In our work, we leverage Splatter Image Szymanowicz et al. (2023) as the 3D representation, with our insight that each attribute of the Splatter Image is modeled within the distribution of pretrained 2D diffusion, therefore using the diffusion prior to generate the entire 3D representation.

3 METHODS

Our method Zero-1-to-G is a single stage direct 3D generation: given single view input $\mathbf{I}$, Zero-1-to-G generates the corresponding 3D representation $\mathbf{z}$, where $z = \{z_i | i = 1, ..., N\}$ multiple splatter images under $\mathbf{N}$ camera views.

To harness the power of large-scale pretrained 2D diffusion models for direct 3D generation, we represent each 3D object as a set of multi-view splatter images (Szymanowicz et al., 2023). In Sec. 3.1, we detail our decomposition process, converting each multi-view splatter image into five 2D attribute images corresponding to RGB color, scale, rotation, opacity, and position. This decomposition allows us to effectively leverage the priors of 2D pretrained diffusion models to learn the underlying 3D object distribution (Sec. 3.2). Furthermore, we fine-tune the VAE decoder to enhance the rendering quality of the decoded splatter images (Sec. 3.3).

3.1 REPRESENTING 3D OBJECTS AS 2D IMAGES

A splatter image is a rearrangement of Gaussian splat into regular grids of $H \times W$, with 14 channels stacking 5 Gaussian splat attributes, including 3-channel RGB, 4-channel rotation, 3-channel scale, 3-channel position, and 1-channel opacity. By applying the data transformation in A.1, each attribute can be represented as a 2D image that explicitly models the object's appearance or geometry. Our key observation is that, each attribute image is well modeled within the distribution of pretrained 2D diffusion model, which means the pretrained 2D diffusion prior might be helpful in the generation of all these attribute images. Motivated by previous works that leverage pretrained diffusion priors for domain-specific images that contain rich 3D information such as normal or depth (Ke et al., 2023; Long et al., 2023; Fu et al., 2024), we take a step further and generate five attribute images of Gaussian splats. Since StableDiffusion is trained on image data in the range $[-1, 1]$, we normalize the values of different attributes to ensure compatibility. Detailed operation is included in the appendix A.1. After applying the data transformations, we convert the 14-channel splatter into a set of 5 multi-view, 3-channel attribute images that are normalized to the RGB space. Example visualization of multi-view splatters are shown in Figure 2.

To obtain our splatter image ground truth, we fine-tune LGM (Tang et al., 2024) to generate Gaussian splats using six views of 3D object renderings as input. A straightforward way would be to directly fit Gaussian splats from multi-view renderings of each object. However, we found that this direct fitting results in attribute images with excessive high-frequency signals compared to LGM output (Figure 3), which makes it challenging for the pretrained VAE to decode effectively.

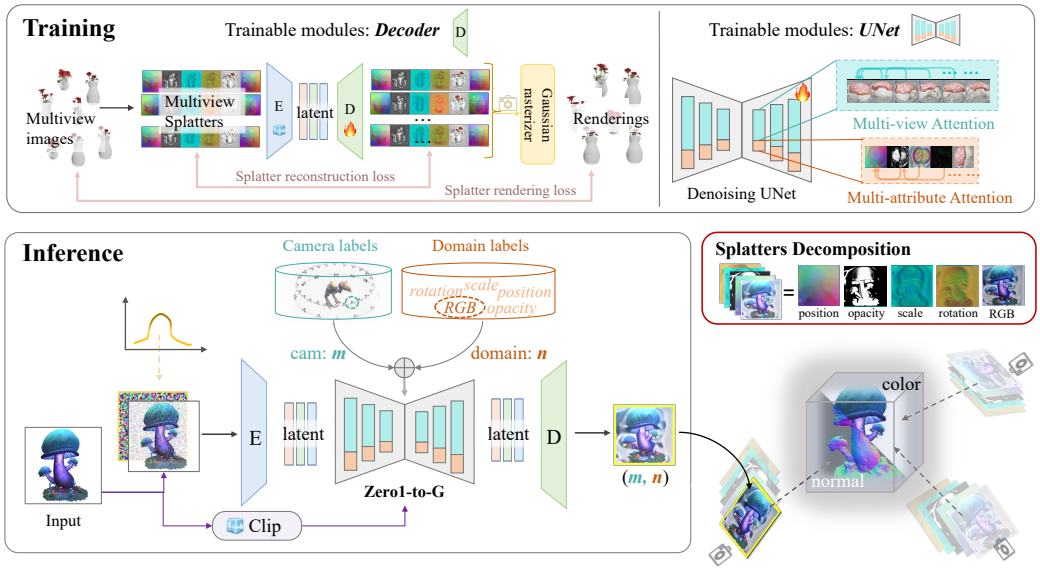

Figure 2: The pipeline of Zero-1-to-G. During training, we fine-tune both the VAE decoder and the denoising UNet of Stable Diffusion. Decoder fine-tuning is required for high-quality splatter image rendering because the renderings of splatter images are sensitive to pixel value changes; the attention mechanism of denoising UNet is extended to model the multi-view and multi-attribute correlation. At inference time, given a single view input of the target object, each component in the splatter image is generated by conditioning the camera view and attribute switcher. The generated set of splatter image components can be directly composed into Gaussian splats. Note that here we only show 3 views of splatter images for better visualization, but we use 6 views in our experiment.

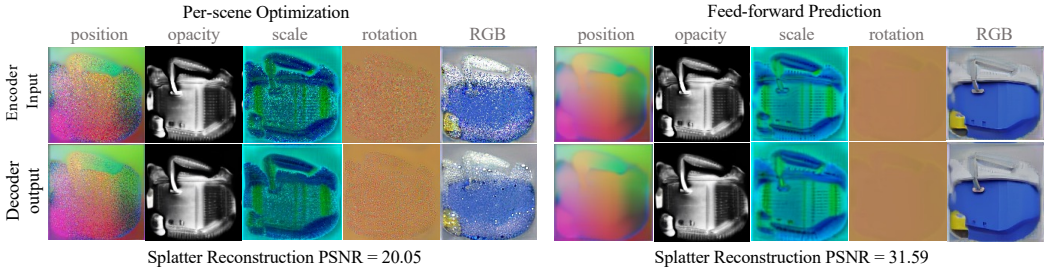

Figure 3: VAE encoding and decoding comparison with per-scene optimized splatters and feed-forward predicted splatters.

3.2 Direct 3D Generation via 2D Pretrained Diffusion

With the decomposition discussed in Sec 3.1, we are ready to directly learn a 3D generative model on Gaussian splats represented by a set of multi-view attribute images. The distribution of our objects represented as splatter images, denoted as $p(\mathbf{z})$, is modeled as a joint distribution of 6 fixed camera views and 5 splatter attributes. Given a set of fixed camera viewpoints $\{\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \cdots, \boldsymbol{\pi}_K\}$ and condition input image y:

$$p(\mathbf{z}) = p(\mathbf{z}^{(1:K,1:N)}|y) = p_{\text{pos, op, sc, rot, rgb}}\left(\{\mathbf{z}_{\text{pos}}^{1:K}, \mathbf{z}_{\text{op}}^{1:K}, \mathbf{z}_{\text{sc}}^{1:K}, \mathbf{z}_{\text{rot}}^{1:K}, \mathbf{z}_{\text{rgb}}^{1:K}\} \mid y\right) \tag{1}$$

where $\{\text{pos, op, sc, rot, rgb}\}$ are the 5 attributes of the splatter image.

Namely, we can naturally utilize the pretrained 2D diffusion models to generate these attribute images since they are already formatted as 3-channel images. However, original Stable Diffusion can only generate single-view images, while our goal is to learn the joint distribution across multiple views and attributes. To ensure that the generated 5 multi-view attribute images are coherent and represent the same object, we insert additional attention layers into the pretrained diffusion UNet to jointly model both the cross-view and cross-attribute distribution of our decomposed splatter images.

Modeling Multi-view Distribution Prior works have approached multi-view diffusion either by reshaping the batch dimension into a token dimension and applying self-attention (Shi et al., 2023b; Liu et al., 2023c; Long et al., 2023; Liu et al., 2024), or by spatially concatenating multi-view images to form a larger image, which directly maps the latent distribution to a multi-view distribution (Shi et al., 2023a). We choose the former approach for its flexibility in reshaping data for both cross-view and cross-attribute attention mechanisms. This design allows for efficient information exchange among different views, where tokens corresponding to the same attribute from different views are concatenated for self-attention. This facilitates our model's ability to learn a consistent multi-view distribution for each Gaussian attribute.

Modeling Multi-Attribute Distribution Building on the work of (Long et al., 2023), we utilize an attribute switcher to specify which attribute the network should generate. To maintain consistency across generated images that represent different attributes of the same object, we employ an attention mechanism to capture the interactions between images taken from the same viewpoint but corresponding to different attributes.

Specifically, we introduce additional self-attention modules to model the cross-attribute correlations, where tokens representing all attributes from the same viewpoint are combined and processed using standard scaled dot-product attention.

Training Loss During training, we organize each view and attribute of a splatter image within the batch dimension and apply independently sampled Gaussian noise. In each attention block, we alternately apply multi-view attention and multi-attribute attention to enhance the model's ability to learn complex correlations.

The forward process of our diffusion model is directly extended from the original DDPM (Ho et al., 2020), which is

$$q(\mathbf{z}_{1:T}^{(1:K,1:N)}|\mathbf{z}_0^{(1:K,1:N)}) = \prod_{t=1}^{T} q(\mathbf{z}_t^{(1:K,1:N)}|\mathbf{z}_{t-1}^{(1:K,1:N)}) = \prod_{t=1}^{T}\prod_{k=1}^{K}\prod_{n=1}^{N} q(\mathbf{z}_t^{(k,n)}|\mathbf{z}_{t-1}^{(k,n)}), \tag{2}$$

And the reverse process will be

$$p_\theta(\mathbf{z}_{0:T}^{(1:K,1:N)}) = p(\mathbf{z}_T^{(1:K,1:N)}) \prod_{t=1}^{T} p_\theta(\mathbf{z}_{t-1}^{(1:K,1:N)}|\mathbf{z}_t^{(1:K,1:N)}) \tag{3}$$

$$= p(\mathbf{z}_T^{(1:K,1:N)}) \prod_{t=1}^{T}\prod_{k=1}^{K}\prod_{n=1}^{N} p_\theta(\mathbf{z}_{t-1}^{(k,n)}|\mathbf{z}_t^{(1:K,1:N)}), \tag{4}$$

where $p_\theta(\mathbf{z}_{t-1}^{(k,n)}|\mathbf{z}_t^{(1:K,1:N)}) = \mathcal{N}(\mathbf{z}_{t-1}^{(k,n)}; \mu_\theta^{(k,n)}(\mathbf{z}_t^{(1:K,1:N)}, t), \sigma_t^2\mathbf{I})$.. The definition of the Gaussian mean for the reverse process is defined as:

$$\mu_\theta^{(k,n)}(\mathbf{z}_t^{(1:K,1:N)}, t) = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{z}_t^{(k,n)} - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta^{(k,n)}(\mathbf{z}_t^{(1:K,1:N)}, t)\right), \tag{5}$$

The corresponding loss function for multi-view and multi-domain modeling is as follows:

$$\ell = \mathbb{E}_{t,\mathbf{x}_0^{(1:K,1:N)},k,n,\epsilon^{(1:K,1:N)}}\left[\|\epsilon^{(k,n)} - \epsilon_\theta^{(k,n)}(\mathbf{z}_t^{(1:K,1:N)}, t)\|_2^2\right], \tag{6}$$

where $\epsilon^{(k,n)}$ is the Gaussian noise added to attribute n for the k-th view, and $\epsilon_\theta^{(k,n)}$ is the model's noise prediction for attribute n in the k-th view.

3.3 VAE Decoder Fine-tuning

The pretrained VAE of Stable Diffusion is initially trained on natural images. While directly utilizing this VAE can produce visually appealing splatter images, the 2D renderings often exhibit noticeable artifacts. This issue arises from two main factors: (1) each pixel in the splatter image corresponds to a Gaussian splat, meaning that even minor changes in pixel values can significantly impact the final rendering, and (2) splatter images contain high-frequency details that are challenging for the VAE to recover accurately.

To leverage the pretrained knowledge while enhancing rendering quality, we freeze the VAE encoder and only fine-tune the decoder. This strategy preserves the latent space of the pretrained model. In addition to using the standard reconstruction loss based on the direct decoder output, we also evaluate the rendering quality of the decoded splatter by comparing its renderings with ground truth images. This rendering-based loss is essential, as a low reconstruction error in the splatter image does not guarantee high-quality renderings. The rendering loss is composed of MSE loss and LPIPS loss, defined as follows:

$$\mathcal{L}_{\text{rgb}} = \mathcal{L}_{\text{MSE}} + \mathcal{L}_{\text{LPIPS}} \tag{7}$$

The overall objective function of decoder finetuning is defined as:

$$\mathcal{L}_{\text{decoder}} = \mathcal{L}_{\text{splatter}} + \mathcal{L}_{\text{normal}} + \mathcal{L}_{\text{rgb}} + \mathcal{L}_{\text{mask}} \tag{8}$$

where $\mathcal{L}_{\text{splatter}}$ denotes the reconstruction loss of splatter image itself, while $\mathcal{L}_{\text{normal}}$, $\mathcal{L}_{\text{rgb}}$ and $\mathcal{L}_{\text{mask}}$ are all about the renderings of the reconstructed splatter, denoting cosine similarity loss of rendered normals, the sum of all losses of rendered images, and the binary cross-entropy loss of the rendered masks.

4 Experiments

4.1 Implementation Details

Dataset We train on the G-buffer Objaverse (Qiu et al., 2024) dataset, which consists of approximately 262,000 objects sourced from Objaverse (Deitke et al., 2024). Each object in the dataset is rendered from 38 viewpoints, with additional normal and depth renderings provided. For generating the ground truth splatter images, we use the first viewpoint as the input condition, along with five additional views at the same elevation and azimuth angles of 30°, 90°, 180°, 270°, and 330° to comprehensively cover the full 360 degrees. We only use the RGB and normal renderings for the supervision of decoder finetuning and Gaussian Splats prediction model.

Model Training and Inference We initialize our model from Stable Diffusion Image Variations. Following Wonder3D (Long et al., 2023), our training includes two stages. In the first stage, we only train multi-view attention, and in the second stage, we add one more cross-domain attention layer for training, and together fine-tune the multi-view attention layer learned in the first stage. For the first stage, we use a batch size of 64 on 4 NVIDIA L40 GPUs for 13k iterations, which takes about 1 day. For the second stage, we use a batch size of 64 on 8 NVIDIA L40 GPUs for 30k iterations, which takes about 2 days. For decoder fine-tuning, we use a total batch size of 64 on 8 NVIDIA L40 GPUs for 20k iterations. The second stage of training takes about 2 days. During inference, we use cfg $= 3.5$ and our method can generate Gaussian splats per object in 8.7 seconds on a single NVIDIA L40 GPU.

4.2 Evaluation Protocol

Dataset and Metrics Following prior works (Liu et al., 2023c;a; Wang et al., 2024), we conduct our quantitative comparisons using the Google Scanned Object (GSO) dataset (Downs et al., 2022). We utilize a randomly selected subset of 30 objects from the GSO, encompassing a range of daily items and animals, as in SyncDreamer (Liu et al., 2023c). For each object, we render a condition image at a resolution of 512×512 and an elevation of $10°$. Additionally, we generate evenly sampled evaluation images at $30°$ azimuthal intervals around the object, maintaining the same elevation. We report commonly used metrics for novel view synthesis, including PSNR, SSIM (Wang et al., 2004), and LPIPS (Zhang et al., 2018). Beyond the GSO dataset, we also evaluate our methods on in-the-wild images.

Baselines We compare our methods against several recent approaches across different categories. For reconstruction-based methods, we include TriplaneGaussian (Zou et al., 2023) and TripoSR (Tochilkin et al., 2024). In the realm of direct 3D generation, we compare with LN3Diff (Lan et al., 2024). Finally, for two-stage methods transitioning from single-image to multi-view to 3D, we include InstantMesh (Xu et al., 2024) and LGM (Tang et al., 2024).

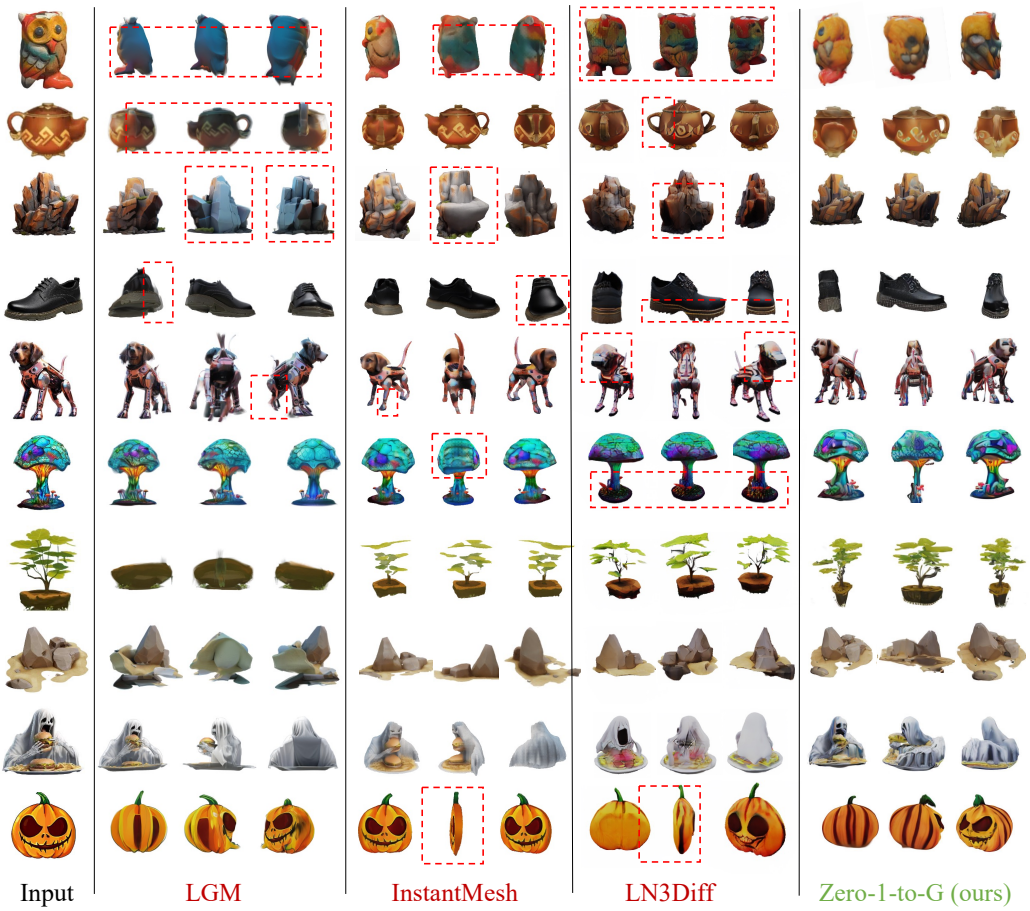

Figure 4: Qualitative comparison with, LGM, InstantMesh, LN3Diff on in-the-wild data.

Table 1: Quantitative comparison between our methods and other baselines on the GSO dataset.

Methods	PSNR ↑	SSIM ↑	LPIPS ↓	CD ↓
TriplaneGaussian (Tang et al., 2023)	17.80	0.811	0.216	0.0440
TripoSR (Tochilkin et al., 2024)	17.32	0.804	0.217	0.0423
LGM (Tang et al., 2024)	17.01	0.793	0.199	0.0621
InstantMesh (Xu et al., 2024)	18.15	0.810	0.179	0.0419
LN3Diff (Lan et al., 2024)	16.30	0.786	0.241	0.0637
Ours-fast(10 steps)	19.03	0.812	0.182	0.0396,
Ours (50 steps)	**19.40**	**0.818**	**0.178**	**0.0390**

4.3 RESULTS

Qualitative Comparison Figure 4 showcases the renderings produced by Zero-1-to-G alongside various baselines for in-the-wild image inputs. We find that two-stage methods relying on multi-view images are often limited by the quality of the generated views. For example, LGM frequently exhibits blue and overly smooth textures in the background (first and third examples in Figure 4), a result of artifacts in the multi-view images generated by ImageDream (Wang & Shi, 2023). Furthermore, their reconstruction model struggles to maintain view-consistent 3D Gaussians, leading to "floaters" around the object (fourth and fifth examples in Figure 4). While InstantMesh provides more consistent renderings, its outputs tend to be overly smooth, as it fuses inconsistent multi-view images into a triplane representation. And its texture also suffers from grid-like artifacts brought by its mesh generation process. Both LGM and InstantMesh are constrained by the limitations of upstream 2D multi-view image generation models, resulting in unreasonable geometry and view-inconsistency features (last example in Figure 4). In contrast, our method incorporates inherent 3D representations, yielding accurate geometry and view-consistent renderings.

Compared to previous direct 3D generation methods, LN3Diff (Lan et al., 2024) also faces challenges with complex textures and fails to produce high-quality reconstructions from single-view inputs. Their textures often lack fine-grained details, and the geometry tends to be oversmoothed. This issue arises because their model is trained from scratch, making it difficult to achieve high-quality 3D generation given the available computational resources and datasets. Consequently, this limits their generalization ability on unseen objects.

In contrast, by leveraging pretrained 2D diffusion priors, our approach delivers high-fidelity 3D renderings with accurate geometry and texture, effectively handling in-the-wild input images.

Quantitative Comparison The results of the quantitative comparison on the GSO dataset are presented in Table 1. Zero-1-to-G consistently outperforms all baselines across all metrics. Reconstruction-based methods, such as TriplaneGaussian and TripoSR, struggle to deliver sharp predictions for unseen regions due to their deterministic nature. Two-stage methods, like InstantMesh, demonstrate relatively good performance; however, they still fall short compared to our approach due to the limitations imposed by sparse multi-view images. In contrast, direct 3D methods such as LN3Diff fail to yield satisfactory results as they lack pretrained priors.

Training Efficiency By leveraging pretrained diffusion priors, our method reduces training time and resource requirements. We complete training in just 3 days using only 8 NVIDIA L40 GPUs, which is more efficient compared to previous two-stage and direct 3D generation methods, as detailed in Table 2. This efficiency highlights the advantage of integrating 2D priors for direct 3D generation, reducing the need for extensive computational resources.

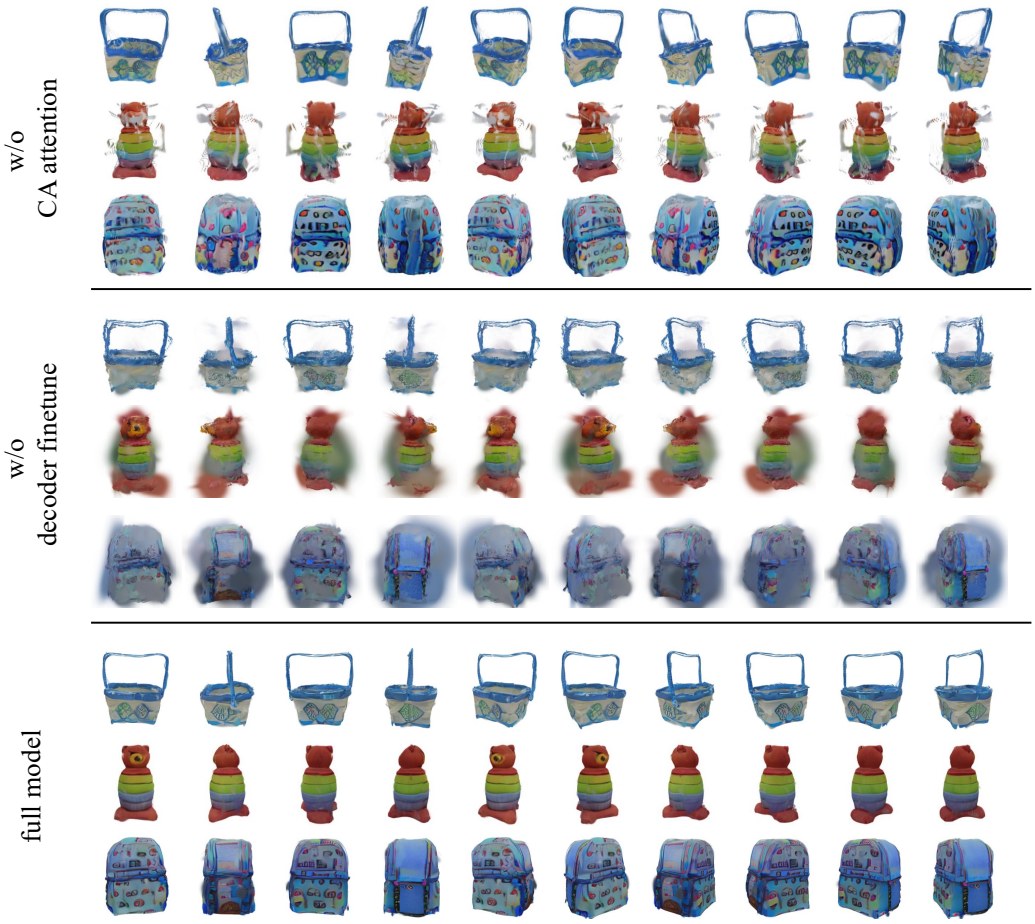

Figure 5: Ablation study on GSO dataset.

4.4 Ablation Study

VAE Decoder Finetuning Without fine-tuning VAE decoder, although the decoded splatter image visually looks fine, the renderings exhibit noticeable artifacts. Since each pixel represents a Gaussian splat and the decoder cannot capture high-frequency areas, well-reconstructed splatter images don't necessarily ensure good renderings.

Cross-attribute Attention We can see from Figure 5, if we don't use cross-attribute attention, the renderings of the Gaussian splats have many floaters and the textures become blurry, this is because that different attributes of the same Gaussian splat are not well aligned.

No Diffusion Prior The use of diffusion prior is essential in our model. To verify this, we conducted training with the same StableDiffusion UNet architecture but using random initialization, and we also added the same cross-view attention layers to the UNet as in our method. We can see that without using the prior, the training of the model cannot converge to meaningful results using the same data and training iterations.

Table 2: Comparison of training efficiency with other baseline methods. For LGM and InstantMesh, we only count the reconstruction part. More time and resources are needed to train their multi-view generation model.

Methods	Training Time ↓	GPUs ↓
LGM (reconstruction module)	4 days	32 * A100 (80G)
InstantMesh (reconstruction module)	12 days	16 * H800 (80G)
LN3Diff	7 days	8 * A100 (80G)
Ours	3 days	8 * L40 (48G)

Table 3: Abalation study on module design, inference with GSO dataset.

Model Components	PSNR ↑	SSIM ↑	LPIPS ↓
w/o diffusion prior	9.39	0.592	0.722
w/o decoder fine-tuning	17.13	0.775	0.272
w/o cross-attribute attention	17.26	0.767	0.237
Full Model	**19.40**	**0.818**	**0.178**

5 Conclusion

In this work, we introduce a novel framework that leverages 2D diffusion priors for direct 3D generation by decomposing Gaussian splats into multi-view attribute images. This decomposition preserves the full 3D structure while efficiently mapping it to 2D images, enabling fine-tuning of pretrained Stable Diffusion models with cross-view and cross-attribute attention layers. Our approach significantly reduces computational costs compared to other direct 3D generation methods. By bypassing the stringent requirement for multi-view image consistency in two-stage approaches, we generate more accurate 3D geometry and produce higher-quality renderings through a single-stage diffusion process. Furthermore, our method exhibits stronger generalization capabilities than existing direct 3D generation techniques due to the use of diffusion priors, offering a more efficient and scalable solution for 3D content creation.

In future work, we plan to extend our approach to 4D generation by fine-tuning video models with dynamic Gaussian splats mapped to multi-view attribute images. While our method produces high-quality 3D results, it currently faces limitations in speed due to the multi-step denoising process, which results in slower inference speed. To address this, future work will explore consistency models as a potential solution to improve inference speed while maintaining the quality of 3D generation.

References

Hansheng Chen, Jiatao Gu, Anpei Chen, Wei Tian, Zhuowen Tu, Lingjie Liu, and Hao Su. Single-stage diffusion nerf: A unified approach to 3d generation and reconstruction. *arXiv preprint arXiv:2304.06714*, 2023a.

Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. *arXiv preprint arXiv:2303.13873*, 2023b.

Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, et al. Objaverse-xl: A universe of 10m+ 3d objects. *Advances in Neural Information Processing Systems*, 36, 2024.

Yuan Dong, Qi Zuo, Xiaodong Gu, Weihao Yuan, Zhengyi Zhao, Zilong Dong, Liefeng Bo, and Qixing Huang. Gpld3d: Latent diffusion of 3d shape generative models by enforcing geometric and physical priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 56–66, 2024.

Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items. In *2022 International Conference on Robotics and Automation (ICRA)*, pp. 2553–2560. IEEE, 2022.

Slava Elizarov, Ciara Rowles, and Simon Donné. Geometry image diffusion: Fast and data-efficient text-to-3d with image-based surface representation. *arXiv preprint arXiv:2409.03718*, 2024.

Xiao Fu, Wei Yin, Mu Hu, Kaixuan Wang, Yuexin Ma, Ping Tan, Shaojie Shen, Dahua Lin, and Xiaoxiao Long. Geowizard: Unleashing the diffusion priors for 3d geometry estimation from a single image. *arXiv preprint arXiv:2403.12013*, 2024.

Ruiqi Gao, Aleksander Holynski, Philipp Henzler, Arthur Brussee, Ricardo Martin-Brualla, Pratul Srinivasan, Jonathan T Barron, and Ben Poole. Cat3d: Create anything in 3d with multi-view diffusion models. *arXiv preprint arXiv:2405.10314*, 2024.

Xianglong He, Junyi Chen, Sida Peng, Di Huang, Yangguang Li, Xiaoshui Huang, Chun Yuan, Wanli Ouyang, and Tong He. Gvgen: Text-to-3d generation with volumetric representation. *arXiv preprint arXiv:2403.12957*, 2024.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 33: 6840–6851, 2020.

Fangzhou Hong, Jiaxiang Tang, Ziang Cao, Min Shi, Tong Wu, Zhaoxi Chen, Tengfei Wang, Liang Pan, Dahua Lin, and Ziwei Liu. 3dtopia: Large text-to-3d generation model with hybrid diffusion priors. *arXiv preprint arXiv:2403.02234*, 2024.

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.

Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *ACM SIGGRAPH 2024 Conference Papers*, pp. 1–11, 2024.

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

Bingxin Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. *arXiv preprint arXiv:2312.02145*, 2023.

Yushi Lan, Fangzhou Hong, Shuai Yang, Shangchen Zhou, Xuyi Meng, Bo Dai, Xingang Pan, and Chen Change Loy. Ln3diff: Scalable latent neural fields diffusion for speedy 3d generation. *arXiv preprint arXiv:2403.12019*, 2024.

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

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 *CVPR*, pp. 300–309, 2023.

Anran Liu, Cheng Lin, Yuan Liu, Xiaoxiao Long, Zhiyang Dou, Hao-Xiang Guo, Ping Luo, and Wenping Wang. Part123: Part-aware 3d reconstruction from a single-view image. In *ACM SIGGRAPH 2024 Conference Papers*, pp. 1–12, 2024.

Minghua Liu, Ruoxi Shi, Linghao Chen, Zhuoyang Zhang, Chao Xu, Xinyue Wei, Hansheng Chen, Chong Zeng, Jiayuan Gu, and Hao Su. One-2-3-45++: Fast single image to 3d objects with consistent multi-view generation and 3d diffusion. *arXiv preprint arXiv:2311.07885*, 2023a.

Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. *arXiv preprint arXiv:2303.11328*, 2023b.

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*, 2023c.

Zhen Liu, Yao Feng, Michael J Black, Derek Nowrouzezahrai, Liam Paull, and Weiyang Liu. Meshdiffusion: Score-based generative 3d mesh modeling. *arXiv preprint arXiv:2303.08133*, 2023d.

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.

Shitong Luo and Wei Hu. Diffusion probabilistic models for 3d point cloud generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2837–2845, 2021.

Norman Müller, Yawar Siddiqui, Lorenzo Porzi, Samuel Rota Bulo, Peter Kontschieder, and Matthias Nießner. Diffrf: Rendering-guided 3d radiance field diffusion. In *CVPR*, pp. 4328–4338, 2023.

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.

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

Lingteng Qiu, Guanying Chen, Xiaodong Gu, Qi Zuo, Mutian Xu, Yushuang Wu, Weihao Yuan, Zilong Dong, Liefeng Bo, and Xiaoguang Han. Richdreamer: A generalizable normal-depth diffusion model for detail richness in text-to-3d. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9914–9925, 2024.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pp. 10684–10695, 2022.

Ruoxi Shi, Hansheng Chen, Zhuoyang Zhang, Minghua Liu, Chao Xu, Xinyue Wei, Linghao Chen, Chong Zeng, and Hao Su. Zero123++: a single image to consistent multi-view diffusion base model, 2023a.

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

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*, pp. 20875–20886, 2023.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. *arXiv preprint arXiv:2303.01469*, 2023.

Stanislaw Szymanowicz, Christian Rupprecht, and Andrea Vedaldi. Splatter image: Ultra-fast single-view 3d reconstruction. In *arXiv*, 2023.

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.

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.

Dmitry Tochilkin, David Pankratz, Zexiang Liu, Zixuan Huang, Adam Letts, Yangguang Li, Ding Liang, Christian Laforte, Varun Jampani, and Yan-Pei Cao. Triposr: Fast 3d object reconstruction from a single image. *arXiv preprint arXiv:2403.02151*, 2024.

Chen Wang, Jiatao Gu, Xiaoxiao Long, Yuan Liu, and Lingjie Liu. Geco: Generative image-to-3d within a second. *arXiv preprint arXiv:2405.20327*, 2024.

Peng Wang and Yichun Shi. Imagedream: Image-prompt multi-view diffusion for 3d generation. *arXiv preprint arXiv:2312.02201*, 2023.

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

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

Sangmin Woo, Byeongjun Park, Hyojun Go, Jin-Young Kim, and Changick Kim. Harmonyview: Harmonizing consistency and diversity in one-image-to-3d. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10574–10584, 2024.

Jiale Xu, Weihao Cheng, Yiming Gao, Xintao Wang, Shenghua Gao, and Ying Shan. Instantmesh: Efficient 3d mesh generation from a single image with sparse-view large reconstruction models. *arXiv preprint arXiv:2404.07191*, 2024.

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.

Xingguang Yan, Han-Hung Lee, Ziyu Wan, and Angel X Chang. An object is worth 64x64 pixels: Generating 3d object via image diffusion. *arXiv preprint arXiv:2408.03178*, 2024.

Biao Zhang, Jiapeng Tang, Matthias Niessner, and Peter Wonka. 3dshape2vecset: A 3d shape representation for neural fields and generative diffusion models. *arXiv preprint arXiv:2301.11445*, 2023a.

Bowen Zhang, Yiji Cheng, Jiaolong Yang, Chunyu Wang, Feng Zhao, Yansong Tang, Dong Chen, and Baining Guo. Gaussiancube: Structuring gaussian splatting using optimal transport for 3d generative modeling. *arXiv preprint arXiv:2403.19655*, 2024a.

Jingyang Zhang, Shiwei Li, Yuanxun Lu, Tian Fang, David McKinnon, Yanghai Tsin, Long Quan, and Yao Yao. Jointnet: Extending text-to-image diffusion for dense distribution modeling. *arXiv preprint arXiv:2310.06347*, 2023b.

Longwen Zhang, Ziyu Wang, Qixuan Zhang, Qiwei Qiu, Anqi Pang, Haoran Jiang, Wei Yang, Lan Xu, and Jingyi Yu. Clay: A controllable large-scale generative model for creating high-quality 3d assets. *ACM Transactions on Graphics (TOG)*, 43(4):1–20, 2024b.

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*, pp. 586–595, 2018.

Zibo Zhao, Wen Liu, Xin Chen, Xianfang Zeng, Rui Wang, Pei Cheng, Bin Fu, Tao Chen, Gang Yu, and Shenghua Gao. Michelangelo: Conditional 3d shape generation based on shape-image-text aligned latent representation. *arXiv preprint arXiv:2306.17115*, 2023.

Linqi Zhou, Yilun Du, and Jiajun Wu. 3d shape generation and completion through point-voxel diffusion. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 5826–5835, 2021.

Zi-Xin Zou, Zhipeng Yu, Yuan-Chen Guo, Yangguang Li, Ding Liang, Yan-Pei Cao, and Song-Hai Zhang. Triplane meets gaussian splatting: Fast and generalizable single-view 3d reconstruction with transformers. *arXiv preprint arXiv:2312.09147*, 2023.

A Appendix

A.1 Implementation Details

Splatter Image Transformation
Each attribute, except opacity, possesses three degree-of-freedoms, which align gracefully with the 3 channels of the RGB space. The following illustrates the detailed operations to convert each attribute into an RGB image

- RGB: the RGB attribute naturally lies in the RGB space, no conversion is needed.
- Position: We normalize the 3D object in the bounding box $[-1, 1]$ and use the 3D coordinates of each Gaussian as position attribute.
- Scale: The raw scale value spans from $1e{-}15$ to $1e{-}2$, so directly converting the 3D scale to RGB space using the min-max value for the whole dataset will result in most regions being zeros due to the significant difference in power. The value distribution also does not match the normal distribution, making it difficult for diffusion models to learn effectively. We thus convert the raw scale values to log-space and clamp the minimum values to -10, as we found Gaussian splats with scales smaller $1e{-}10$ will have negligible effects on the final rendering.
- Rotation: We first convert the 4-dimensional quaternion to 3-dimensional axis angle, then normalize it to $[-1, 1]$.
- Opacity: We directly duplicate the single channel to 3 channels, and average the predicted 3-channel image to get the final opacity prediction.

UNet Fine-tuning When fine-tuning the StableDiffusion UNet, for both stages, we use a constant learning rate of $1e{-}4$ with a warmup of the first 100 steps. We use the Adam optimizer for both stages and the betas are set to $(0.9, 0.999)$. For classifier-free guidance, we drop the condition image with a probability of 0.1.

LGM Fine-tuning To obtain the splatter image ground truth for our training, as mentioned in Sec. 3.1, we fine-tune LGM (Tang et al., 2024) to take as input 6 multi-view renderings of the G-Objaverse dataset and output splatter images of 2D Gaussian splatting (Huang et al., 2024). The training objective is to compare the splatter renderings with ground truth images using MSE and LPIPS loss. We also use cosine similarity loss between ground truth normals and rendered normals. We fine-tune LGM for 30k iterations with a batch size of 32 on 8 NVIDIA L40 GPUS, which takes about 1 day.

A.2 Limitations

Despite achieving superior reconstruction metrics and strong generalization to in-the-wild data, our method has some limitations. First, our inference speed is not as fast as regression models, as each splatter must be generated through diffusion. A potential improvement would be to integrate a consistency model Song et al. (2023) to reduce denoising steps and enhance inference speed. Second, we do not currently disentangle material and lighting conditions, leading to highlights and reflections being baked into the Gaussian splat texture. Future work could address this by incorporating inverse rendering to better predict non-Lambertian surfaces.

A.3 More Results

Chamfer Distance We have further evaluated Chamfer Distance on the GSO datasets to show the superiority of our generated geometry. The improvement in Chamfer Distance further validates our claim that our approach generates superior 3D representations, demonstrating the effectiveness of our method.

Table 4: Comparison of Chamfer Distance across methods.

Methods	Chamfer Distance ↓
LGM Tang et al. (2024)	0.0621
InstantMesh Xu et al. (2024)	0.0419
LN3Diff Lan et al. (2024)	0.0637
Ours	**0.0390**

More visualizations on ablation Figure 6 shows the splatter image visualization of ablation study. We can see that without cross-attribute attention, there are obvious misalignments of different domains of the splatter images. Without decoder fine-tuning, although the splatter image is visually good, the rendering is not satisfying because Gaussian splats are sensitive to the value changes in the pixels. Fine-tuning the decoder can greatly improve the rendering quality.

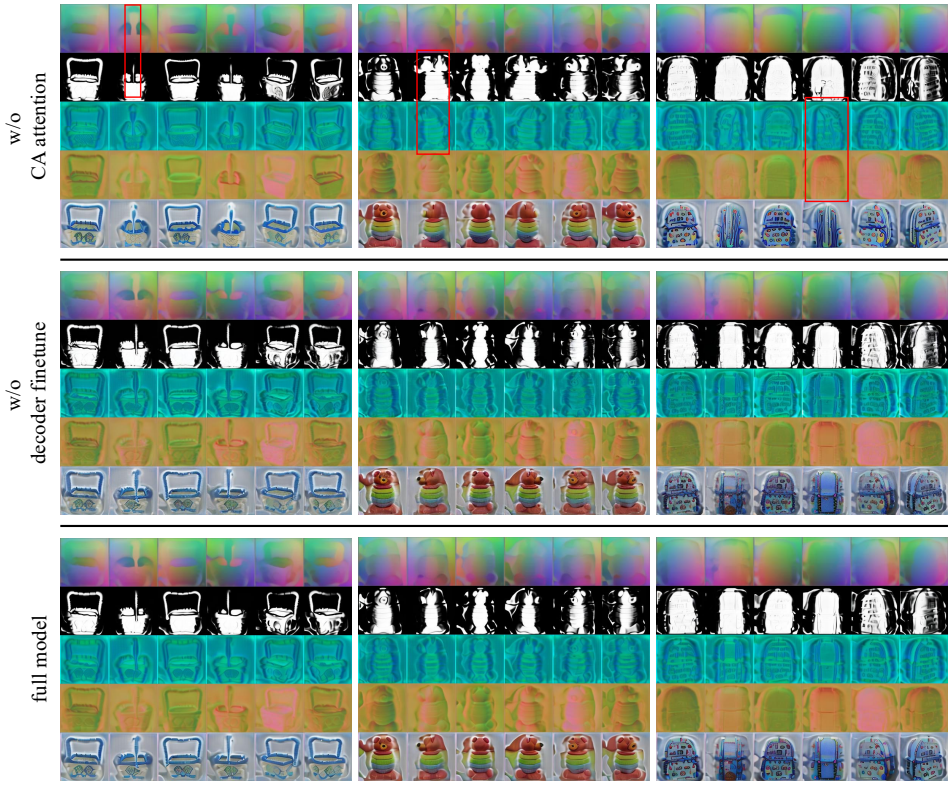

Figure 6: Splatter visualization of ablation study.

More examples More RGB and normal renderings can be found in Figure 7.

Diversity Since we model the unseen viewpoints with diffusion models, our results can generate diverse results given the same input Figure 8.

Figure 7: RGB and normal renderings of more examples on in-the-wild and GSO datasets.

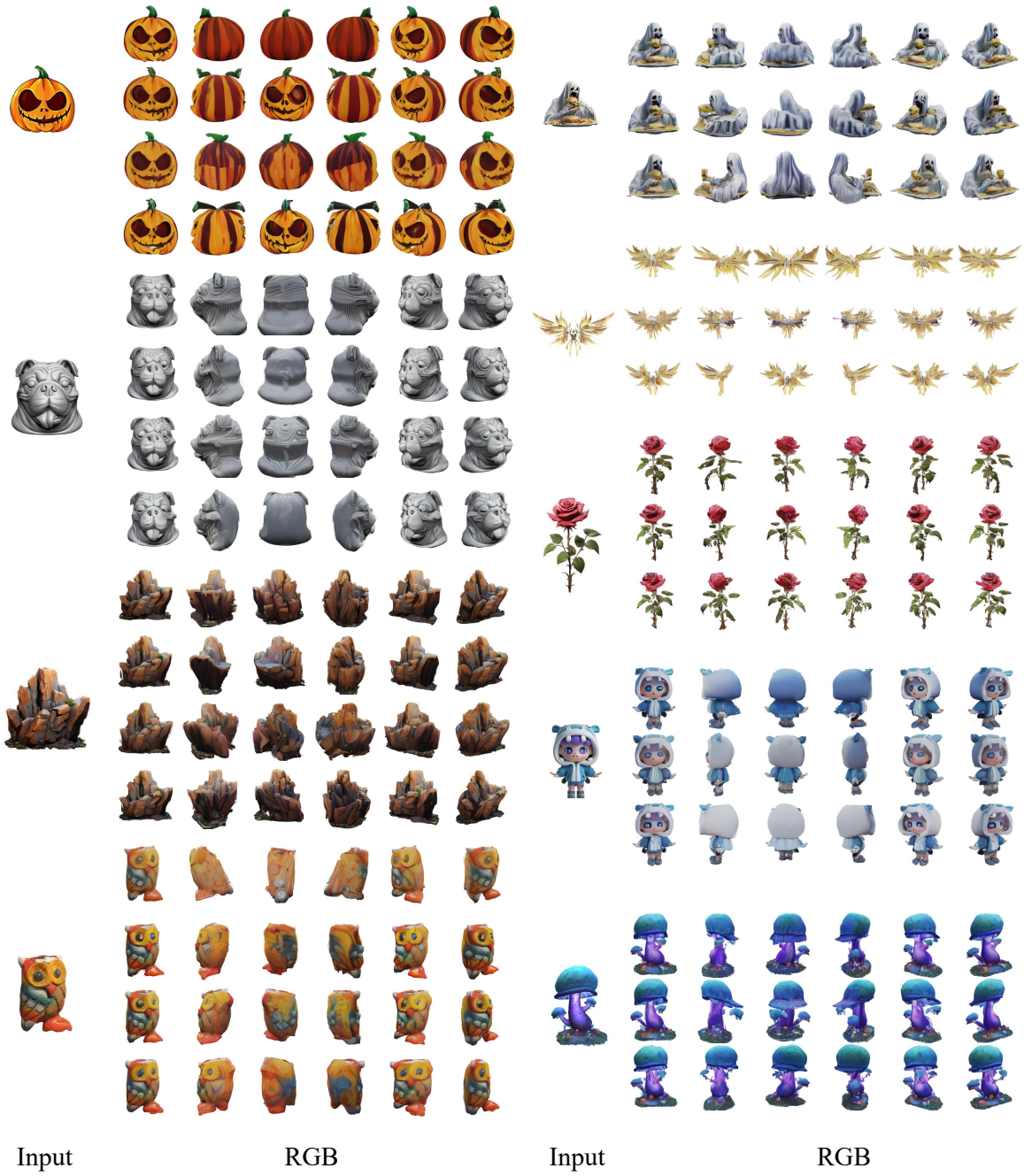

Figure 8: Generative 3D model with various geometry and texture given the same condition image, which shows strong generative ability of our model.