

ViewCrafter: Taming Video Diffusion Models for High-fidelity Novel View Synthesis

Wangbo Yu*, Jinbo Xing*, Li Yuan*, Wenbo Hu†, Xiaoyu Li, Zhipeng Huang, Xiangjun Gao, Tien-Tsin Wong, Ying Shan, and Yonghong Tian†, *Fellow, IEEE*

Abstract—Despite recent advancements in neural 3D reconstruction, the dependence on dense multi-view captures restricts their broader applicability. In this work, we propose **ViewCrafter**, a novel method for synthesizing high-fidelity novel views of generic scenes from single or sparse images with the prior of video diffusion model. Our method takes advantage of the powerful generation capabilities of video diffusion model and the coarse 3D clues offered by point-based representation to generate high-quality video frames with precise camera pose control. To further enlarge the generation range of novel views, we tailored an iterative view synthesis strategy together with a camera trajectory planning algorithm to progressively extend the 3D clues and the areas covered by the novel views. With ViewCrafter, we can facilitate various applications, such as immersive experiences with real-time rendering by efficiently optimizing a 3D-GS representation using the reconstructed 3D points and the generated novel views, and scene-level text-to-3D generation for more imaginative content creation. Extensive experiments on diverse datasets demonstrate the strong generalization capability and superior performance of our method in synthesizing high-fidelity and consistent novel views. Our project webpage and code are available at <https://drexubery.github.io/ViewCrafter/>.

Index Terms—Novel View Synthesis, Video Diffusion Models, 3D Scene Generation.

1 INTRODUCTION

NOVEL view synthesis plays a crucial role in computer vision and graphics for creating immersive experiences in games, mixed reality, and visual effects. Despite the significant success of 3D neural reconstruction techniques such as NeRF [1] and 3D-GS [2], their dependence on dense multi-view observations restricts their broader applicability in situations where only limited views are available.

A more desirable problem scenario in practice involves synthesizing novel views of generic scenes from sparse observations or even a single image. This task is considerably challenging as it necessitates a comprehensive understanding of the 3D world, including structures, appearance, semantics, and occlusions. Early researches [3], [4], [5], [6], [7], [8], [9], [10] focused on training regression-based models to synthesize novel views from sparse or single input. However, due to their limited representation capabilities, these methods are mostly category-specific and only handle certain domains such as indoor scenes. Recent advancements in powerful diffusion models have made zero-shot novel view synthesis [11], [12], [13] from single view approachable. Nevertheless, these methods are either restricted to handling object-level images or lack precise control of the camera pose due to their dependency on high-level pose prompts to guide the view synthesis process. Some works [14], [15] also attempt to synthesize novel views from a single image using depth-based warping and diffusion-based inpainting. Yet, these methods often produce inconsistent

content in occlusion regions due to the per-view inpainting mechanism.

In this work, we focus on high-fidelity novel view synthesis of *generic scenes* from single or sparse images, maintaining *precise control of the camera pose* and *consistency* of the generated novel views. To achieve this, we investigate leveraging the generative capabilities of video diffusion models alongside the explicit 3D information provided by point cloud representations. On one hand, video diffusion models [16], [17], [18], trained on web-scale video datasets, develop a reasonable understanding of the world, which facilitates the generation of plausible video content from a single image or text prompt. However, they lack the underlying 3D information of the scene and fall short in achieving precise camera view control. On the other hand, recent dense stereo methods [19], [20] have made fast point cloud reconstruction from single or sparse images accessible. Point cloud representation provides valuable coarse 3D scene information, enabling precise pose control for free-view rendering. Yet, due to its poor representation capability and the limited 3D cues offered by extremely sparse reference images, it suffers from occlusions, missing areas, and geometry distortion, hindering its utility in novel view synthesis. With these in mind, we propose integrating the generative power of video diffusion models with the coarse 3D prior provided by point-based representation, aiming to facilitate higher-fidelity novel view synthesis of generic scenes.

*: Equal contribution. †: Corresponding authors. Wangbo Yu, Li Yuan, Zhipeng Huang, Yonghong Tian are with Peking University and Peng Cheng Laboratory. Jinbo Xing is with The Chinese University of Hong Kong. Wenbo Hu, Xiaoyu Li are with Tencent AI Lab. Ying Shan is with ARC Lab, Tencent PCG and Tencent AI Lab. Xiangjun Gao is with Hong Kong University of Science and Technology. Tien-Tsin Wong is with Monash University.

Our method, **ViewCrafter**, accomplishes novel view synthesis by a *point-conditioned video diffusion model* that generates high-fidelity and consistent videos under a novel view trajectory, conditioned on corresponding frames rendered from point cloud reconstructed from single or sparse images. Leveraging the explicit 3D information from the

point cloud and the generative capabilities of video diffusion models, our method enables precise control of 6 DoF camera poses and generates high-fidelity, consistent novel views. Furthermore, video diffusion models face challenges in generating long videos because of unacceptably increased memory and computation costs. To tackle this challenge, we propose an *iterative view synthesis* strategy along with a *content-adaptive camera trajectory planning* algorithm to progressively extend the reconstructed point cloud and the areas covered by the novel views. Starting from the initial point cloud derived from the input image(s), we first employ the camera trajectory planning algorithm to predict a content-adaptive camera pose sequence from the current point cloud to effectively reveal occlusions. Next, we render the point cloud according to the predicted pose sequence and synthesize novel views by ViewCrafter with the conditions of the rendered point cloud. Subsequently, the point cloud is updated from the synthesized novel views to extend the global point cloud representation. Through iteratively conducting these steps, we can ultimately obtain high-fidelity novel views that cover a large view range and an extended point cloud.

In addition to novel view synthesis, we explore several applications of our method. For instance, we can efficiently optimize a 3D-GS representation based on the constructed point cloud and the synthesized novel views within minutes, enabling real-time rendering for immersive experiences. Furthermore, our method shows the potential to adapt to scene-level text-to-3D generation, which can foster more imaginative artistic creations.

We extensively evaluate our method for zero-shot novel view synthesis and sparse view 3D-GS reconstruction on various datasets, including Tanks-and-Temples [21], RealEstate10K [7], and CO3D [22]. For zero-shot novel view synthesis, our method outperforms the baselines in both image quality and pose accuracy metrics. This demonstrates its superior ability to synthesize high-fidelity novel views and achieve precise pose control. In 3D-GS reconstruction, our approach consistently surpasses previous state-of-the-art. This further validates its effectiveness in scene reconstruction from sparse views.

Our contributions can be summarized as follows:

- We propose **ViewCrafter**, a **novel view synthesis framework tailored for synthesizing high-fidelity novel view sequences of generic scenes from single or sparse images while maintaining precise control of camera poses**.
- We present an **iterative view synthesis strategy in conjunction with a content-adaptive camera trajectory planning algorithm to progressively expand the covered areas of novel views and the reconstructed point cloud, enabling long-range and large-area novel view synthesis**.
- Our method achieves superior performance on various challenging datasets in terms of both the quality of synthesized novel views and the accuracy of camera pose control. It facilitates various applications beyond novel view synthesis, such as real-time rendering for immersive experiences by efficiently optimizing a 3D-GS representation from our results,

and scene-level text-to-3D generation for more imaginative artistic creations.

2 RELATED WORK

2.1 Regression-based Novel View Synthesis

Regression-based methods aim to train a feed-forward model to generate novel views from sparse or single image inputs. This is often achieved using CNN/Transformer-based [23] architectures to establish a 3D representation of the input image(s). For instance, several works [24], [25] have applied this idea to specific modalities, such as human faces, by generating tri-plane representations for novel view synthesis. LRM [26] extends this strategy to generic objects, while other methods like [7], [8], [9] adopt the multi-plane representation, and PixelNeRF [10] employs NeRF [1] as 3D representation for novel view synthesis. Inspired by the success of 3D-GS [2], recent approaches such as PixelSplat [27] and MVSplat [28] explore training regression-based models to produce 3D Gaussian representations for real-time rendering capabilities. Additionally, some methods like [3], [4], [5], [6] combine monocular depth estimation and image inpainting modules in a unified framework for novel view synthesis. However, these methods are limited to category-specific domains, such as objects and indoor scenes, and are prone to artifacts due to their limited model representation capabilities. In contrast, our method can synthesize high-fidelity novel views of generic scenes.

2.2 Diffusion-based Novel View Synthesis

The rapid advancement of diffusion models [29], [30], [31] have demonstrated exceptional proficiency in synthesizing high-quality images and shows the potential to be adapted in synthesizing novel views from single or sparse inputs. While some optimization-based approaches [32], [33], [34] directly train a 3D representation under the supervision of text-to-image (T2I) diffusion models [31], they require scene-specific optimization, which compromising their generalization capabilities. To address this, GeNVS [35] proposes a generalized novel view synthesis framework by training a 3D feature-conditioned diffusion model on a large-scale multiview dataset [22]. Similarly, Zero-1-to-3 [11] develops camera pose-conditioned diffusion models trained on synthetic datasets [36], [37], enabling novel view synthesis from more diverse inputs. However, these models are either category-specific [35], [38] or limited to handling toy-level objects with simple backgrounds. Recently, ZeroNVS [12] improves the generation capability of Zero-1-to-3 [11] by training it on a mixed dataset containing both synthetic [36] and real data [7], [39], [40], enabling zero-shot novel view synthesis of generic scenes from a single input image. Nonetheless, it still struggles to synthesize consistent novel views and lacks precise pose control, as it treats camera pose conditions as high-level text embeddings. Reconfusion [41] proposes a PixelNeRF [10] feature-conditioned diffusion model to achieve relatively accurate pose control in novel view synthesis. However, it fails to synthesize consistent novel views due to its inability to model the correlations among sampled views. Additionally, it requires multiple images as input and cannot process a single image. Several

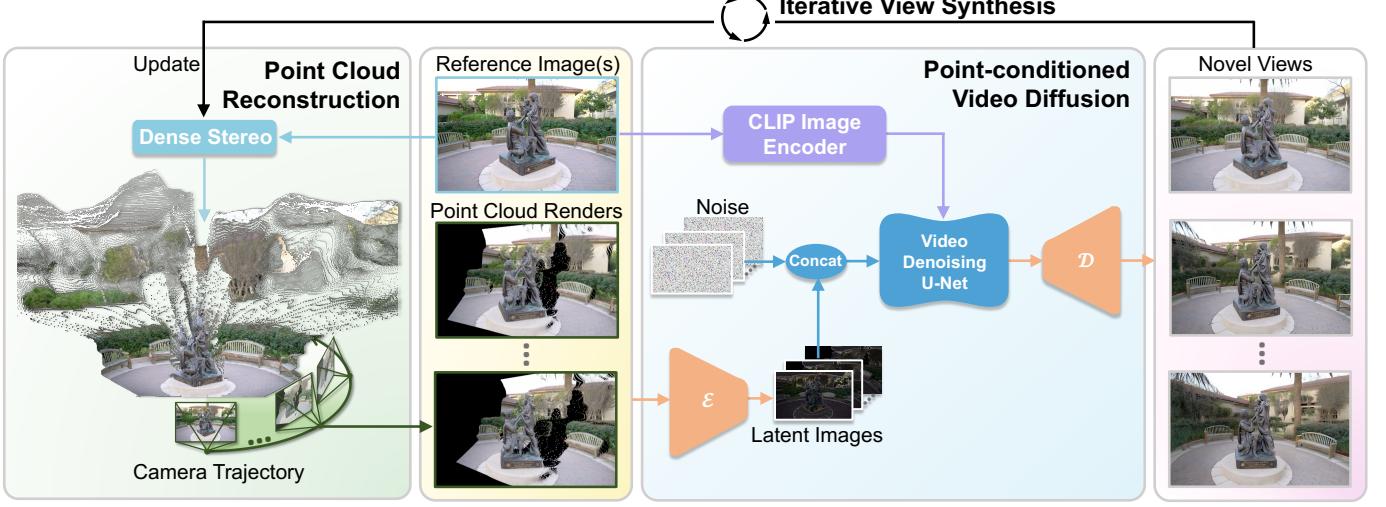


Fig. 1. Overview of our ViewCrafter. Given a single reference image or sparse image sets, we first build its point cloud representation using a dense stereo model, which enables accurately moving cameras for free-view rendering. Subsequently, to address the large missing regions, geometric distortions, and point cloud artifacts exhibited in the point cloud render results, we train a point-conditioned video diffusion model to serve as an enhanced renderer, facilitating the generation of high-fidelity and consistent novel views based on the coarse point cloud renders. To achieve long-range novel view synthesis, we adopt an iterative view synthesis strategy that involves iteratively moving cameras, generating novel views, and updating the point cloud, which enables a more complete point cloud reconstruction and benefits downstream tasks such as 3D-GS optimization.

works [14], [15], [42] utilize depth-based warping to synthesize novel views and employ a pre-trained T2I diffusion model [43] to refine the warped images. However, the novel views generated by these methods often suffer from artifacts and unrealistic contexts in the inpainted regions, which limits their applicability.

2.3 Conditional Video Diffusion Models

As generative models that produce content from diverse input modalities evolve rapidly, enhancing user control over generation has garnered significant interest. ControlNet [44], T2I-adapter [45], and GLIGEN [46] pioneered the introduction of condition signals for T2I generation. Similar strategies have also been employed in video generation, enabling controls like RGB images [17], [18], [47], depth [48], [49], trajectory [50], [51], and semantic maps [52]. However, camera motion control has received comparatively less attention. AnimateDiff [53] and SVD [17] investigate class-conditioned video generation, grouping camera movements and utilizing LoRA [54] modules to create specific camera motions. MotionCtrl [13] improves control by using camera extrinsic matrices as conditioning signals. Although effective for simple trajectories, their dependence on 1D numeric values leads to imprecise control in complex real-world situations. MultiDiff [55] leverage depth-based warping to produce warped images, and condition the video diffusion model on the warped images to provide explicit 3D prior. Nevertheless, it trains the video diffusion model on class-specific datasets [7], [56], thereby lacking the generalization ability to handle generic scenes. Recently, CamCo [57] and CameraCtrl [58] introduced Plücker coordinates [59] in video diffusion models for camera motion control. Nevertheless, these methods still cannot precisely control the camera motion due to the complicated mapping from numeric camera parameters to videos. In this paper, we

propose to leverage explicit point cloud representations for precise camera control in video generation, thereby fulfilling our needs for consistent and accurate novel view synthesis.

3 METHOD

In the following, we start with a brief introduction to video diffusion models in Section. 3.1, followed by an explanation of the point cloud reconstruction pipeline in Section. 3.2 and an illustration of the point-conditioned video diffusion model in Section. 3.3. Subsequently, we elaborate on the iterative view synthesis and camera trajectory planning strategy in Section. 3.4, and demonstrate how to apply our approach for efficient 3D-GS optimization and text-to-3D generation in Section. 3.5.

3.1 Preliminary: Video Diffusion Models

A diffusion model [30] consists of two primary components: a forward process q and a reverse process p_θ . The forward process initiates with clean data $\mathbf{x}_0 \sim q_0(\mathbf{x}_0)$ and gradually introduces noise to \mathbf{x}_0 , creating a noisy state across different time steps. This is mathematically represented as $\mathbf{x}_t = \alpha_t \mathbf{x}_0 + \sigma_t \epsilon$, where $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The hyper-parameters α_t and σ_t satisfy the constraint $\alpha_t^2 + \sigma_t^2 = 1$. The reverse process p_θ focuses on removing noise from the clean data utilizing a noise predictor ϵ_θ , which is optimized by the objective:

$$\min_{\theta} = \mathbb{E}_{t \sim \mathcal{U}(0,1), \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})} [\|\epsilon_\theta(\mathbf{x}_t, t) - \epsilon\|_2^2]. \quad (1)$$

In diffusion-based video generation, Latent Diffusion Models (LDMs) [60] are frequently employed to mitigate the computational cost. In LDMs, the video data $\mathbf{x} \in \mathbb{R}^{L \times 3 \times H \times W}$ are encoded into the latent space using a pre-trained VAE encoder frame-by-frame, expressed as $\mathbf{z} = \mathcal{E}(\mathbf{x})$, $\mathbf{z} \in \mathbb{R}^{L \times C \times h \times w}$. Then, both the forward process q

and the reverse process p_θ are performed in the latent space. The final generated videos are obtained through the VAE decoder $\hat{x} = \mathcal{D}(z)$. In this work, we build our model based on an open-sourced Image-to-Video (I2V) diffusion model DynamiCrafter [18], which is capable of creating dynamic videos from a single input image. This aligns naturally with our goal of synthesizing novel views from sparse or single inputs.

3.2 Point Cloud Reconstruction from Single or Sparse Images

To achieve accurate pose control in our novel view synthesis framework, we first establish the point cloud representation from the reference image(s). Specifically, we employ a dense stereo model, *e.g.* DUST3R [19], to reconstruct the point cloud and estimate camera parameters simultaneously. It takes a pair of RGB images $\mathbf{I}^0, \mathbf{I}^1 \in \mathbb{R}^{H \times W \times 3}$ as input and generates corresponding point maps $\mathbf{O}^{0,0}, \mathbf{O}^{1,0} \in \mathbb{R}^{H \times W \times 3}$ along with their respective confidence maps $\mathbf{D}^{0,0}, \mathbf{D}^{1,0} \in \mathbb{R}^{H \times W}$, reflecting the level of confidence in their accuracy. The subscript of $\mathbf{O}^{0,0}, \mathbf{O}^{1,0}$ denote that they are expressed in the same camera coordinate system of the anchor view \mathbf{I}^0 . To recover the camera’s intrinsic parameters, it is assumed that the principal point is centered and the pixels are square. Consequently, only the focal length f_0^* remains unknown, which can be solved through a few optimization steps using Weiszfeld algorithm [61]:

$$f_0^* = \arg \min_{f_0} \sum_{u=0}^W \sum_{v=0}^H \mathbf{D}_{u,v}^{0,0} \left\| (u', v') - f_0 \frac{(\mathbf{O}_{u,v,0}^{0,0}, \mathbf{O}_{u,v,1}^{0,0})}{\mathbf{O}_{u,v,2}^{0,0}} \right\|, \quad (2)$$

where $u' = u - \frac{W}{2}$ and $v' = v - \frac{H}{2}$. In the case where only a single input image is available, we duplicate the input image to create a paired input and then estimate its point map and camera intrinsic. When there are more than two input images, it can also perform global point map alignment with a few optimization iterations.

The colored point cloud, which provides coarse 3D information of the scene, can be obtained by integrating the point maps with their corresponding RGB images. However, the limited representation capabilities of the point cloud and the insufficient 3D cues provided by sparse or single inputs can result in significant missing regions, occlusions, and artifacts in the reconstructed point cloud, leading to low-quality render results. Therefore, we propose incorporating video diffusion models to achieve high-fidelity novel view synthesis based on the imperfect point cloud.

3.3 Rendering High-fidelity Novel Views with Video Diffusion Models

As shown in Fig. 1, taking a single reference image \mathbf{I}^{ref} as an example, we first obtain its point cloud, camera intrinsics and camera pose \mathbf{C}^{ref} through the dense stereo model [19]. Subsequently, we can navigate the camera along a camera pose sequence $\mathbf{C} = \{\mathbf{C}^0, \dots, \mathbf{C}^{L-1}\}$ that contains \mathbf{C}^{ref} to render the point cloud and obtain a sequence of render results, denote as $\mathbf{P} = \{\mathbf{P}^0, \dots, \mathbf{P}^{L-1}\}$. While the point cloud renders accurately represent view relationships, they are plagued by substantial occlusions, missing areas, and reduced visual fidelity. To enhance the quality of novel view

rendering, our objective is to learn a conditional distribution $\mathbf{x} \sim p(\mathbf{x} | \mathbf{I}^{\text{ref}}, \mathbf{P})$ that can produce high-quality novel views $\mathbf{x} = \{\mathbf{x}^0, \dots, \mathbf{x}^{L-1}\}$ based on the point cloud renders \mathbf{P} and the reference image(s) \mathbf{I}^{ref} . Motivated by the efficacy of video diffusion models [16], [17], [18] in synthesizing high-quality and consistent videos, we learn this conditional distribution by training a video diffusion model conditioned on the point cloud renders and the reference image(s). As a result, the novel view synthesis process can be naturally modeled as the reverse process of a point-conditioned video diffusion model, expressed as $\mathbf{x} \sim p_\theta(\mathbf{x} | \mathbf{I}^{\text{ref}}, \mathbf{P})$, where θ denotes the model parameters.

The architecture of the point-conditioned video diffusion model is illustrated in Fig. 1. It inherits the LDM [60] architecture, which primarily comprises a pair of VAE encoder \mathcal{E} and decoder \mathcal{D} for image compression, a video denoising U-Net with spatial layers followed by temporal layers for temporal-aware noise estimation, as well as a CLIP [62] image encoder for reference image understanding. **We incorporate point cloud renders as conditional signals in the video denoising U-Net by encoding them using \mathcal{E} and concatenating the resulting latent images with noise across the channel dimension.**

To train the model, we create paired training data that includes both point cloud renders $\mathbf{P} = \{\mathbf{P}^0, \dots, \mathbf{P}^{L-1}\}$ and the corresponding ground-truth reference images $\mathbf{I} = \{\mathbf{I}^0, \dots, \mathbf{I}^{L-1}\}$. The point cloud renders are forced to traverse at least one ground-truth view, *i.e.*, to include at least one ground-truth reference image at a random location among the L frames. It helps the model better learn to transfer fine details from the reference image(s) to the point cloud renders and enables our model to flexibly handle arbitrary number of reference image(s). Following the approach of LDMs [60], we freeze the parameters of the VAE encoder \mathcal{E} and decoder \mathcal{D} , and conduct the training process in the latent space. Specifically, we encode the training data pair $\mathbf{I} = \{\mathbf{I}^0, \dots, \mathbf{I}^{L-1}\}$ and $\mathbf{P} = \{\mathbf{P}^0, \dots, \mathbf{P}^{L-1}\}$ into the latent space, yielding the ground-truth latents $\mathbf{z} = \{\mathbf{z}^0, \dots, \mathbf{z}^{L-1}\}$ and the condition signals $\hat{\mathbf{z}} = \{\hat{\mathbf{z}}^0, \dots, \hat{\mathbf{z}}^{L-1}\}$ that will be concatenated channel-wise with the sampled noise. Subsequently, the video denoising U-Net is optimized by the diffusion loss:

$$\min_{\theta} = \mathbb{E}_{t \sim \mathcal{U}(0,1), \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})} [\|\epsilon_\theta(\mathbf{z}_t, t, \hat{\mathbf{z}}, \mathbf{I}^{\text{ref}}) - \epsilon\|_2^2], \quad (3)$$

where $\mathbf{z}_t = \alpha_t \mathbf{z}_0 + \sigma_t \epsilon$.

During the inference process, we render a sequence of point cloud renders $\mathbf{P} = \{\mathbf{P}^0, \dots, \mathbf{P}^{L-1}\}$ and replace the reference view render results with the corresponding reference image(s). Subsequently, we encode them into the latent space to obtain the latent images $\hat{\mathbf{z}} = \{\hat{\mathbf{z}}^0, \dots, \hat{\mathbf{z}}^{L-1}\}$, sample noise $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, then concatenate them channel-wise to construct the noisy latent. In addition to the latent space condition, we also pass the reference image(s) into the CLIP image [62] encoder, which will modulate the U-Net features through cross-attention for better 3D understanding. With the trained U-Net, the noisy latents are iteratively denoised into clean latents and then decoded into high-quality novel views $\mathbf{x} = \{\mathbf{x}^0, \dots, \mathbf{x}^{L-1}\}$ using the VAE decoder \mathcal{D} .

Algorithm 1 Camera trajectory planning

Input: Reference image(s) \mathcal{I}_{ref} , dense stereo model $\mathcal{D}(\cdot)$, point-conditioned video diffusion model $\mathcal{V}(\cdot)$, initial point cloud \mathcal{P}_{ref} , searching space \mathcal{S} , initial pose \mathcal{C}_{ref} , maximum predicted poses N , number of candidate poses K , utility function $\mathcal{F}(\cdot)$

- 1: **Initialize** Current point cloud $\mathcal{P}_{\text{curr}} \leftarrow \mathcal{P}_{\text{ref}}$, current camera pose $\mathcal{C}_{\text{curr}} \leftarrow \mathcal{C}_{\text{ref}}$, $\text{step} \leftarrow 0$
- 2: **while** $\text{step} \leq N$ **do**
- 3: Uniformly sample K candidate poses $\mathcal{C}_{\text{can}} = \{\mathcal{C}_{\text{can}}^1, \dots, \mathcal{C}_{\text{can}}^K\}$ from the searching space \mathcal{S} around the current pose $\mathcal{C}_{\text{curr}}$, initialize candidate mask set $\mathcal{M}_{\text{can}} = \{\}$
- 4: **for** \mathcal{C} in $\{\mathcal{C}_{\text{can}}^1, \dots, \mathcal{C}_{\text{can}}^K\}$ **do**
- 5: $\mathcal{M}_{\mathcal{C}} = \text{Render}(\mathcal{P}_{\text{curr}}, \mathcal{C})$
- 6: $\mathcal{M}_{\text{can}}.\text{append}(\mathcal{M}_{\mathcal{C}})$
- 7: **end for**
- 8: $\mathcal{C}_{\text{nkv}} = \arg \max_{\mathcal{C} \in \mathcal{C}_{\text{can}}} \mathcal{F}(\mathcal{C})$
- 9: $\mathcal{I}_{\text{nkv}} = \mathcal{V}(\text{interpolate}(\mathcal{C}_{\text{curr}}, \mathcal{C}_{\text{nkv}}), \mathcal{P}_{\text{curr}})$
- 10: $\mathcal{P}_{\text{curr}} \leftarrow \mathcal{D}(\mathcal{I}_{\text{nkv}}, \mathcal{P}_{\text{curr}})$
- 11: $\mathcal{C}_{\text{curr}} \leftarrow \mathcal{C}_{\text{nkv}}$
- 12: $\text{step} \leftarrow \text{step} + 1$
- 13: **end while**
- 14: **return**

3.4 Iterative View Synthesis and Camera Trajectory Planning

Existing video diffusion models encounter challenges in generating long videos with numerous frames. As the video length increases during inference, it results in decreased video stability and increased computational costs. Therefore, the trained ViewCrafter model may face challenges in generating longer videos to produce a larger view range. To address this challenge, we adopt an iterative view synthesis approach. Specifically, given an initial point cloud established from the reference image(s), we navigate the camera from one of the reference views to a target camera pose to reveal occlusion and missing regions of the current point cloud. Subsequently, we can generate high-fidelity novel views using ViewCrafter and back-project the generated views to complete the point cloud. Through iteratively moving the camera, generating novel views, and updating the point cloud, we can ultimately obtain novel views with an extended view range and a more complete point cloud representation of the scene.

In the iterative view synthesis process, the design of the camera trajectory significantly impacts the synthesis results. Methods like [14], [63] use predefined camera trajectories for scene generation, which overlooks the diverse geometry relationships presented in different scenes, resulting in significant occlusions. To effectively reveal occlusions in the iterative view synthesis process and facilitate more complete scene generation, we designed a Next-Best-View (NBV) [64], [65], [66]-based camera trajectory planning algorithm, which enables the adaptive generation of camera trajectories tailored to handle various scene types. The camera trajectory planning algorithm is illustrated in Algorithm 1. Starting with the input reference image(s) \mathcal{I}_{ref} , we construct an

initial point cloud \mathcal{P}_{ref} using the dense stereo model [19]. Referring [64], [66], [67], [68], we opt for a forward-facing quarter-sphere with evenly distributed camera poses as the searching space, denoted as \mathcal{S} , and position it centrally at the origin of the point cloud's world coordinate system, setting the radius to the depth of the center pixel in the reference image. The camera trajectory is initialized from one of the reference camera poses \mathcal{C}_{ref} . To predict the subsequent pose, we uniformly sample K candidate camera poses $\mathcal{C}_{\text{can}} = \{\mathcal{C}_{\text{can}}^1, \dots, \mathcal{C}_{\text{can}}^K\}$ from the searching space surrounding the current camera pose $\mathcal{C}_{\text{curr}} = \mathcal{C}_{\text{ref}}$, then render a set of candidate masks \mathcal{M}_{can} (where 1 signifies occlusion and missing regions, while 0 represents filled regions) from the current point cloud $\mathcal{P}_{\text{curr}}$. We then establish a utility function [64] $\mathcal{F}(\cdot)$ to determine the optimal camera pose for the subsequent step, defined as:

$$\mathcal{F}(\mathcal{C}) = \begin{cases} \frac{\sum(\mathcal{M}_{\mathcal{C}})}{W \times H}, \frac{\sum(\mathcal{M}_{\mathcal{C}})}{W \times H} < \Theta \\ 1 - \frac{\sum(\mathcal{M}_{\mathcal{C}})}{W \times H}, \frac{\sum(\mathcal{M}_{\mathcal{C}})}{W \times H} > \Theta, \end{cases} \quad (4)$$

where $\mathcal{C} \in \mathcal{C}_{\text{can}}$, $\mathcal{M}_{\mathcal{C}} \in \mathcal{M}_{\text{can}}$, and $\sum(\mathcal{M}_{\mathcal{C}}) = \sum_{u=0}^W \sum_{v=0}^H \mathcal{M}_{\mathcal{C}}(u, v)$. The utility function helps identify a suitable camera pose that reveals an adequate area of occlusion and missing regions while avoiding poses that reveal excessively large holes deviating significantly from a threshold Θ , which may affect ViewCrafter's generation capability. Once the next best camera pose \mathcal{C}_{nkv} is predicted, we interpolate a camera path between $\mathcal{C}_{\text{curr}}$ and \mathcal{C}_{nkv} , and then apply ViewCrafter along the path to generate a sequence of high-fidelity novel views. Subsequently, we back-project and align the generated novel view \mathcal{I}_{nkv} onto the current point cloud $\mathcal{P}_{\text{curr}}$, and designate \mathcal{C}_{nkv} as the new $\mathcal{C}_{\text{curr}}$, then repeat the aforementioned process until the predicted poses reach the predefined limitation N . Through iteratively predicting camera poses, synthesizing novel views, and updating the point cloud, we can ultimately obtain a more complete point cloud representation of the scene.

3.5 Applications

ViewCrafter can effectively produce accurate, consistent, and high-fidelity novel views from single or sparse inputs. Nevertheless, it faces challenges in providing immersive experiences due to the slow multi-step denoising process. To achieve real-time rendering, we further delve into optimizing a 3D-GS [2] representation from the results of our ViewCrafter. To that aim, a direct approach involves concurrently running ViewCrafter multiple times on the initially built point cloud to generate multiple novel views and optimizing a 3D-GS from them. However, this will lead to suboptimal optimization results, since the initial point cloud is incomplete and will introduce inconsistencies in occlusion regions among the generated view sequences at different times.

As illustrated in Fig. 2, to facilitate more consistent 3D-GS optimization, we leverage the aforementioned iterative view synthesis strategy to iteratively complete the initial point cloud and synthesize novel views using ViewCrafter, which not only provides consistent novel views as training data but also offers a strong geometry initialization for the

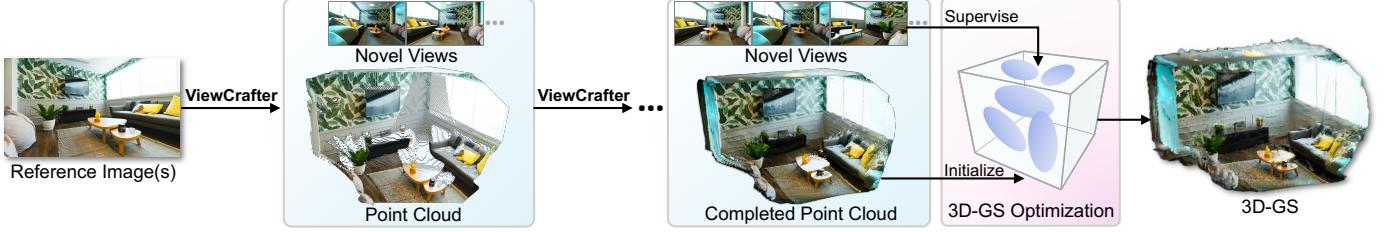


Fig. 2. Application of 3D-GS optimization. To facilitate more consistent 3D-GS optimization, we leverage the iterative view synthesis strategy to progressively complete the initial point cloud and synthesize novel views using ViewCrafter. We then use the completed dense point cloud to initialize 3D-GS and employ the synthesized novel views to supervise 3D-GS training.

3D-GS [2]. During training, the center of each 3D Gaussian is initialized from the completed dense point cloud, and the attributes of each 3D Gaussian are optimized under the supervision of the synthesized novel views. We simplify the 3D-GS optimization process by deprecating the densification, splitting, and opacity reset tricks [69], and reduce the overall optimization time into 2,000 iterations, which is considerably faster than the original 3D-GS training.

In addition to synthesizing novel views of real-world images, we also explore the application of combining ViewCrafter with creative text-to-image diffusion models for text-to-3D generation. This involves using a text-to-image diffusion model to generate a reference image from the provided text prompt, followed by employing ViewCrafter for novel view synthesis and 3D reconstruction.

4 EXPERIMENTS

In this section, we begin with an illustration of the implementation details in Section. 4.1, followed by a comparison of zero-shot novel view synthesis in Section. 4.2 and scene reconstruction in Section. 4.3. Subsequently, we conduct ablation studies to evaluate design choices and training strategies in Section. 4.4. Finally, we present the results of text-to-3D generation in Section. 4.5.

4.1 Implementation Details

We employ a progressive training strategy. In the first stage, we train the ViewCrafter model at a resolution of 320×512 , with the frame length set to 25. The entire video denoising U-Net is trained for 50,000 iterations using a learning rate of 5×10^{-5} and a mini-batch size of 16. In the second stage, we fine-tune the spatial layers (*i.e.*, 2D Conv and spatial attention layers) of the video denoising U-Net at a resolution of 576×1024 for high-resolution adaptation, with 5,000 iterations on a learning rate of 1×10^{-5} and a valid mini-batch size of 16. Our model was trained on a mixed dataset consisting of DL3DV [70] and RealEstate10K [7]. We divide the video data into video clips, each containing 25 frames. To generate the conditional signals, specifically the point cloud renders, we process the video clips using DUST3R [19] to obtain the camera trajectory of the video clips and the globally aligned point clouds of each video frame. Then, we randomly select the constructed point cloud of the video frames and render it along the estimated camera trajectory using Pytorch3D [71]. In total, we generate 632,152 video pairs as training

data. During inference, we adopt DDIM sampler [30] with classifier-free guidance [72].

4.2 Zero-shot Novel View Synthesis Comparison

Datasets and evaluation metrics. In our study, we employ three real-world datasets of different scales as our zero-shot novel view synthesis evaluation benchmark, which includes the CO3D [39] dataset, the RealEstate10K [7] dataset, and the Tanks-and-Temples [21] dataset. For CO3D [39] consisting of object-centric scenes, we evaluate on 10 scenes. RealEstate10K [7] comprises video clips of indoor scenes, we adopt 10 scenes from its test set for evaluation. For Tanks-and-Temples [21] containing large-scale outdoor and indoor scenes, we use all of the 9 scenes. For all benchmarks, we extract frames from the original captured videos and create two types of test sets by applying different sampling rates to the original video. The easy test set is generated using a small frame sampling stride, characterized by slow camera motions and limited view ranges. In contrast, the hard test set is produced with a large sampling stride, featuring rapid camera motions and large view ranges.

We employ PSNR, SSIM [73], LPIPS [74], and FID [75] as the evaluation metrics for assessing image quality. Among these, PSNR is a traditional metric used to compare image similarity. SSIM [73] and LPIPS [74] are designed to evaluate the structural and perceptual similarity between the generated images and the ground truth images, as these metrics are specifically designed to align more closely with human perceptual judgment. Referring to [3], [4], [5], we further integrate FID into our evaluation process for assessing the quality of synthesized views, which proves particularly efficacious when evaluating the hard test set that contains a significant number of missing and occlusion regions. Additionally, to evaluate the pose accuracy of the generated novel views, we estimate the camera poses of the generated novel views to compare with the ground truth camera poses. Following [58], we transform the camera coordinate of the estimated poses to be relative to the first frame, and normalize the translation scale using the furthest frame. We then calculate the rotation distance (R_{dist}) in comparison to the ground truth rotation matrices of each generated novel view sequence, expressed as:

$$R_{\text{dist}} = \sum_{i=1}^n \arccos\left(\frac{\text{tr}(\mathbf{R}_{\text{gen}}^i \mathbf{R}_{\text{gt}}^{iT}) - 1}{2}\right), \quad (5)$$

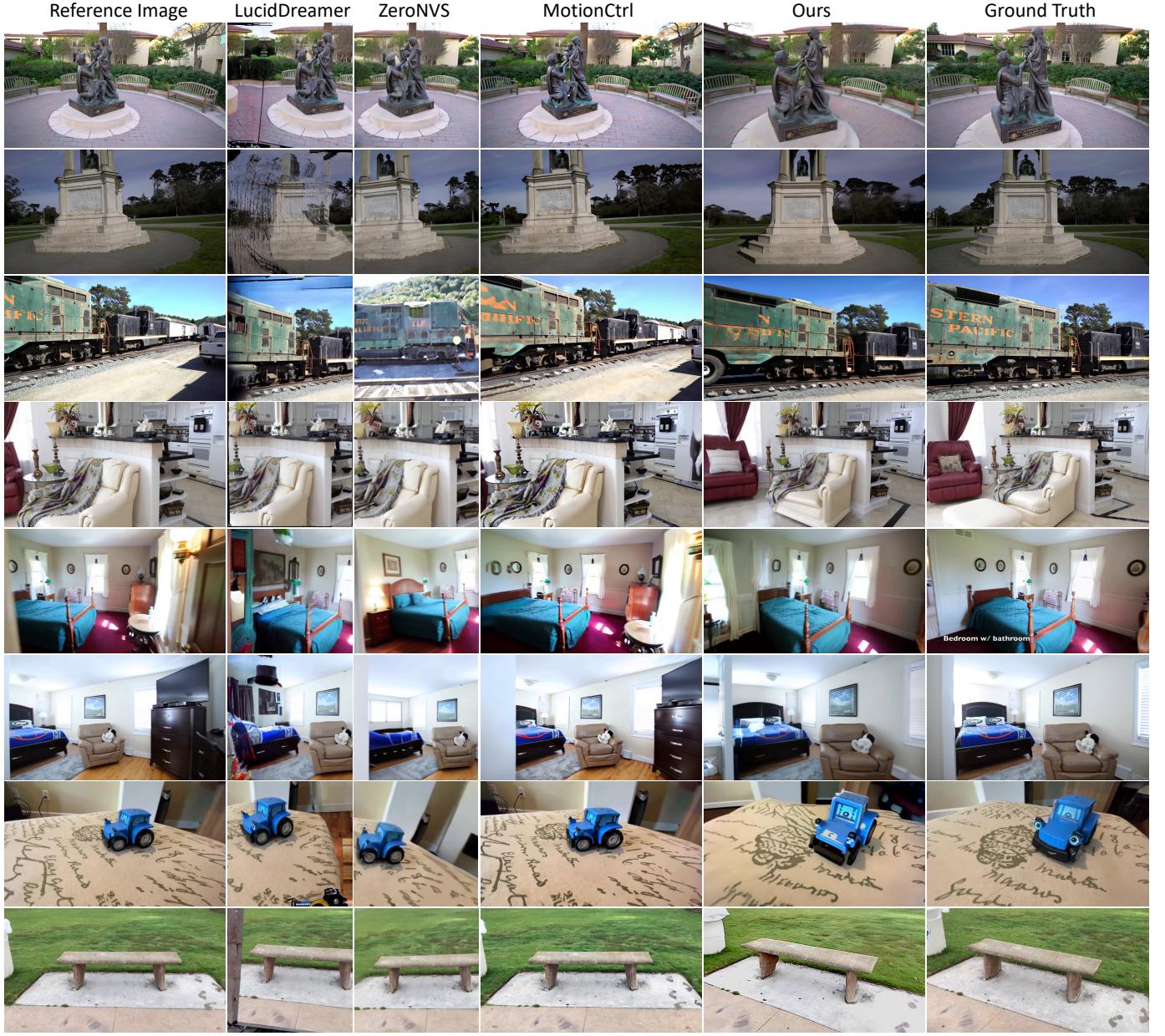


Fig. 3. Qualitative comparison of zero-shot novel view synthesis on Tanks-and-Temples [21], RealEstate10K [7] and CO3D [39] dataset. The reference images are displayed in the left-most column, and the ground truth novel views are located in the right-most column.

where \mathbf{R}_{gt}^i and $\mathbf{R}_{\text{gen}}^i$ denote the ground truth rotation matrix and generated rotation matrix, and we sum the distance of all frames as the final results. We also compute the translation distance (T_{dist}), expressed as:

$$T_{\text{dist}} = \sum_{i=1}^n \|\mathbf{T}_{\text{gt}}^i - \mathbf{T}_{\text{gen}}^i\|_2. \quad (6)$$

Notably, since COLMAP [76] is sensitive to inconsistent features and prone to fail to extract poses from the generated novel views, we instead use DUS3R [19] for more robust pose estimation.

Comparison baselines. As a diffusion-based generalizable novel view synthesis framework, we compare our method with three diffusion-based baselines: ZeroNVS [12], MotionCtrl [13] and LucidDreamer [14]. Specifically, ZeroNVS

[12] is finetuned from Zero-1-to-3 [11] and can generate novel views conditioned on a reference image and the relative camera pose. The camera pose is processed as CLIP [62] text embedding and injected into the diffusion U-Net via cross-attention. MotionCtrl [13] is a camera-conditioned video diffusion model finetuned from SVD [17]. It can generate consistent novel views from the conditioned reference image and the relative camera pose sequences, which are also processed as high-level embedding and injected into the video diffusion U-Net through cross-attention. Lucid-Dreamer [14] utilizes depth-based warping to synthesize novel views, and employs a pretrained diffusion-based inpainting model [43] to inpaint missing regions in the novel views. For the zero-shot novel view synthesis comparison, we use a single reference image as input for all baselines

TABLE 1

Quantitative comparison of zero-shot novel view synthesis on Tanks-and-Temples [21], RealEstate10K [7] and CO3D [39] dataset. Since ZeroNVS [12] and LucidDreamer [14] can only handle square images, we crop the generated novel views from our method and MotionCtrl [13] to align with them when computing the quantitative metrics.

Dataset	Easy set						Hard set					
	LPIPS ↓	PSNR ↑	SSIM ↑	FID ↓	$R_{\text{dist}} \downarrow$	$T_{\text{dist}} \downarrow$	LPIPS ↓	PSNR ↑	SSIM ↑	FID ↓	$R_{\text{dist}} \downarrow$	$T_{\text{dist}} \downarrow$
Tanks-and-Temples												
LucidDreamer [14]	0.413	14.53	0.362	42.32	6.137	5.695	0.558	11.69	0.267	200.8	8.998	9.305
ZeroNVS [12]	0.482	14.71	0.380	74.60	8.810	6.348	0.569	12.05	0.309	131.0	8.860	8.557
MotionCtrl [13]	0.400	15.34	0.427	70.3	7.299	8.039	0.473	13.29	0.384	196.8	9.801	9.112
ViewCrafter (ours)	0.194	21.26	0.655	27.18	0.471	1.009	0.283	18.07	0.563	38.92	1.109	0.910
RealEstate10K												
LucidDreamer [14]	0.315	16.35	0.579	56.77	5.821	10.02	0.400	14.13	0.511	71.43	7.990	10.85
ZeroNVS [12]	0.364	16.50	0.577	96.18	6.370	9.817	0.431	14.24	0.535	105.8	8.562	10.31
MotionCtrl [13]	0.341	16.31	0.604	89.90	4.236	9.091	0.386	16.29	0.587	70.02	8.084	9.295
ViewCrafter (ours)	0.145	21.81	0.796	33.09	0.380	2.888	0.178	22.04	0.798	24.89	1.098	2.867
CO3D												
LucidDreamer [14]	0.429	15.11	0.451	78.87	12.90	6.665	0.517	12.69	0.374	157.8	16.43	8.301
ZeroNVS [12]	0.467	15.15	0.463	93.84	15.44	8.872	0.524	13.31	0.426	143.2	15.02	10.22
MotionCtrl [13]	0.393	16.87	0.529	69.18	16.87	5.131	0.443	15.46	0.502	112.7	18.81	5.575
ViewCrafter (ours)	0.243	21.38	0.687	24.63	2.175	1.033	0.324	18.96	0.641	36.96	2.849	1.480

and our method, since the baselines are only capable of performing single-view novel view synthesis.

Qualitative comparison. The qualitative results are presented in Fig. 3, where the reference images are displayed in the left-most column, and the ground truth novel views are located in the right-most column. The results of LucidDreamer [14] exhibit severe artifacts, since it uses depth-based warping for generating novel views, which is particularly problematic when handling in-the-wild images with unknown camera intrinsic, leading to inaccurate novel views. Moreover, it employs an off-the-shelf inpainting model [43] to refine the warped results, which tends to introduce inconsistencies between the original and inpainted content. Novel views generated by ZeroNVS [12] also exhibit relatively low quality and poor accuracy; the primary reason is that ZeroNVS introduces the camera pose condition into diffusion models through text embedding, which fails to provide precise control over the generation of novel views, leading to sub-optimal results. Similarly, although MotionCtrl [13] can produce novel views with better fidelity, it falls short in generating novel views that precisely align with the given camera conditions. This is because MotionCtrl also adopts a high-level camera embedding to control camera pose, leading to less accurate novel view synthesis. In comparison, our method incorporates explicit point cloud prior and video diffusion model, the results demonstrate the superiority of our method in terms of both pose control accuracy and the overall quality of the generated novel views.

Quantitative comparison. The quantitative comparison results are reported in Table. 1. Since ZeroNVS [12] and LucidDreamer [14] can only handle squared images, we crop the generated novel views of our method and MotionCtrl [13] to align with ZeroNVS and LucidDreamer when computing the quantitative metrics. In terms of image quality, it can be observed that our approach consistently outperforms the baselines in all the metrics. Specifically, the higher PSNR and SSIM values indicate that our method

maintains better image quality and similarity to the ground truth. The lower LPIPS score further demonstrates that our approach generates more perceptually accurate images, while the significantly improved FID score suggests that our method captures the underlying distribution of the data more effectively. In terms of pose accuracy, the reduced R_{dist} and T_{dist} demonstrate the effectiveness of our model design, which enables more accurate pose control in novel view synthesis.

4.3 Scene Reconstruction Comparison

Datasets and evaluation metrics. In the scene reconstruction comparison, we use 6 scenes from the Tanks-and-Temples dataset [21] for evaluation. We create a challenging sparse-view benchmark that contains only 2 ground truth training images for each scene, and use 12 views for evaluation. We employ PSNR, SSIM [73], and LPIPS [74] as the evaluation metrics for image quality assessment.

Comparison baselines. We compare our method with three 3D-GS representation-based sparse view reconstruction methods: DNGaussian [77], FSGS [78] and InstantSplat [79]. Specifically, DNGaussian [12] and FSGS [78] utilize point cloud produced by COLAMP [76] as initialization, and leverage both image supervision and depth regularization for sparse view reconstruction. InstantSplat [79] explores utilizing point cloud produced by DUSt3R [19] as initialization, which enables efficient 3D-GS training from sparse images.

Qualitative comparison. The qualitative comparison results are presented in Fig. 4. It can be observed the results from DNGaussian [77] exhibit significant artifacts. Similarly, results from FSGS [78] show artifacts when viewed from novel views that deviate from the ground truth training images. Although InstantSplat [79] utilizes DUSt3R [19] for point cloud initialization, which better preserves details from the ground truth training images, it fails to recover occlusion regions due to its omission of the densification

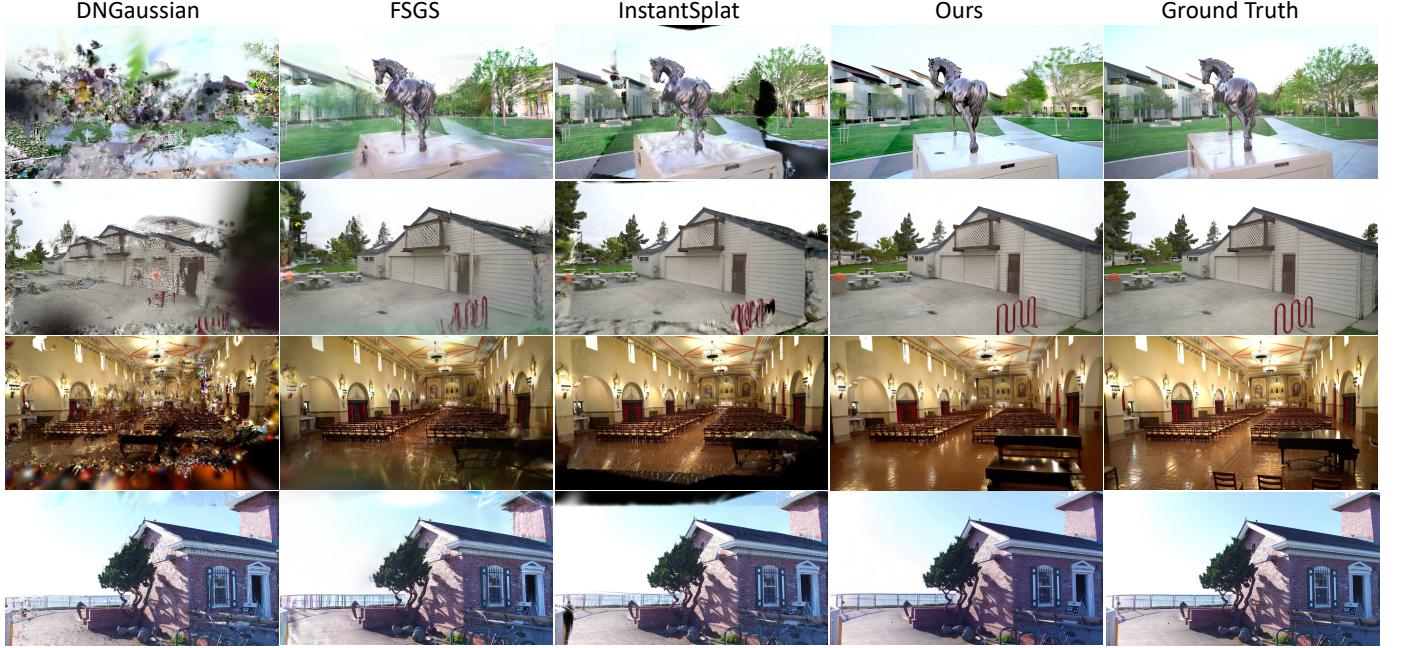


Fig. 4. **Qualitative comparison of scene reconstruction on Tanks-and-Temples [21] dataset.** We train each scene using 2 ground truth training images, and render novel views to compare with the ground truth novel view (right-most column).

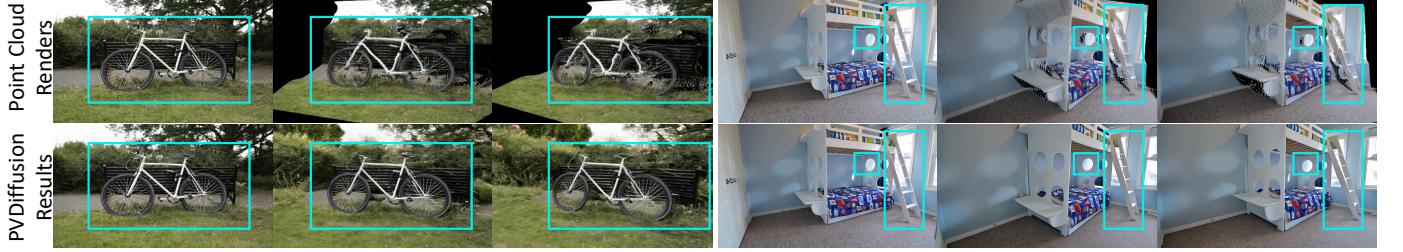


Fig. 5. **Robustness for point cloud condition.** We show the point cloud render results and the corresponding novel views generated by ViewCrafter in the top and bottom rows, respectively (Best viewed with zoom-in).

TABLE 2
Quantitative comparison of scene reconstruction on
Tanks-and-Temples [21]. We use 2 ground truth training images for
each scene, and adopt 12 views for evaluation.

Method	LPIPS ↓	PSNR ↑	SSIM ↑
DNGaussian [77]	0.331	15.47	0.541
FSGS [78]	0.364	17.53	0.558
InstantSplat [79]	0.275	18.61	0.614
ViewCrafter (ours)	0.245	21.50	0.692

TABLE 3
Ablation study of the pose condition strategy. Except for the
condition signal, the architecture and training strategy of the Plücker
model are identical to those of ViewCrafter.

Method	LPIPS ↓	PSNR ↑	SSIM ↑	FID ↓	$R_{\text{dist}} \downarrow$	$T_{\text{dist}} \downarrow$
Plücker model	0.370	17.51	0.546	49.33	2.688	2.570
Ours	0.270	20.25	0.649	38.17	0.552	0.983

4.4 Ablation Study

process [2], resulting in severe holes under novel views. In comparison, our method leverages the priors from video diffusion models, enabling the generation of high-fidelity novel views given only 2 ground truth training images.

Quantitative comparison. The quantitative comparison results are presented in Table. 2. It can be observed that our approach consistently outperforms the comparison baselines in all the metrics, further validating the effectiveness of our method in scene reconstruction from sparse views.

Discussion on pose condition strategy. In our method, we utilize point cloud renders as an explicit condition for the video diffusion model, enabling highly accurate pose control for novel view synthesis. Some concurrent works [57], [80] adopt Plücker coordinates [81] as pose condition for pose-controllable video generation. To compare the pose accuracy between our point cloud-based pose condition strategy and the Plücker coordinate-based pose condition strategy, we train a Plücker coordinate-conditioned video diffusion model (Plücker model for short) that accepts



Fig. 6. **Qualitative ablation of pose condition strategy.** We make the architecture and training strategy of the Plücker model to be identical to those of ViewCrafter, with the exception of the condition signal.

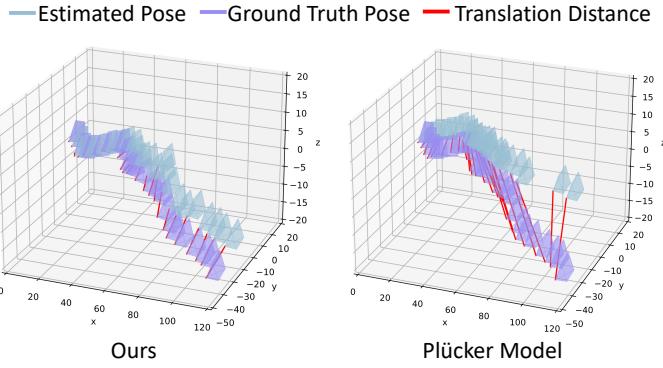


Fig. 7. **Visualization of pose accuracy.** We compare the alignment level between the ground truth camera poses and the poses estimated from the generated novel views of ViewCrafter and the Plücker model.

TABLE 4

Quantitative comparison of different training paradigms. We analyze the effectiveness of training both the spatial and temporal layers of the video denoising U-Net, as well as the benefits of the progressive training strategy and inference with more frames.)

Traing paradigm	LPIPS ↓	PSNR ↑	SSIM ↑	FID ↓
Only train spatial layers	0.301	18.82	0.595	42.30
Directly train on 576×1024	0.314	18.55	0.582	41.01
16 frames model	0.289	19.07	0.610	38.43
Ours	0.280	19.52	0.615	37.77

Plücker coordinates as conditions for synthesizing novel views. The Plücker coordinate describes per-pixel motion; For a given RGB frame and its camera pose, its Plücker coordinate shares the same spatial size with the RGB frame and comprises 6 channels for each pixel location. Given a pose sequence, we resize the corresponding Plücker coordinates to align with the size of the latent space, and concatenate them with noise along the channel dimension. Except for the pose condition strategy, we maintain the rest of the architecture of the Plücker model to be identical to ViewCrafter, and train the model at 320×512 resolution, following the training details reported in Section. 4.1. We conduct a zero-shot novel view synthesis comparison between ViewCrafter (320×512 resolution) and the Plücker model. The qualitative and quantitative results are shown in Fig. 6 and Table. 3, which demonstrates that the point cloud-based pose condition strategy employed in ViewCrafter achieves more accurate pose control in novel view synthesis. We also observed that the Plücker model prone to ignore the

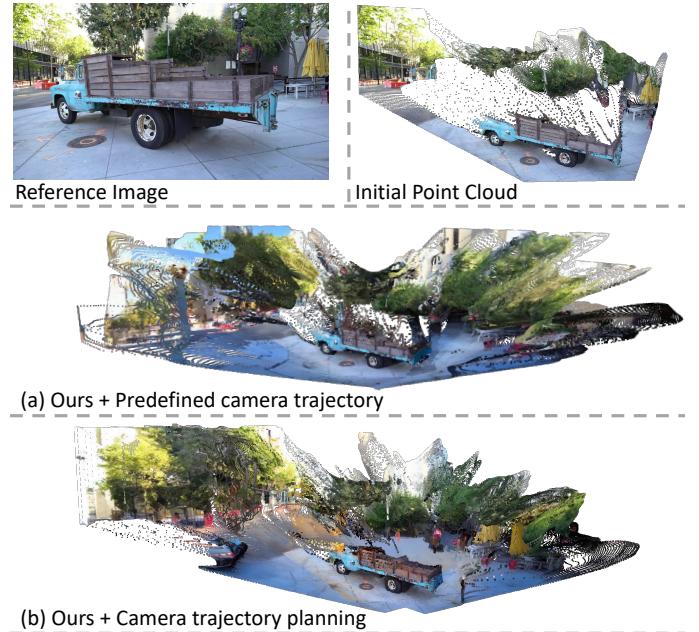


Fig. 8. **Ablation study on the camera trajectory planning.** The point cloud reconstructed using the predefined camera trajectory fails to effectively complete the occlusion region. In contrast, the point cloud generated through our camera trajectory planning algorithm reveals the occlusion region of the scene more effectively, enhancing the overall reconstruction quality of the point cloud.

high-frequency movements of the camera. Fig. 7 presents an example, where we compare the alignment level between the ground truth camera poses and the poses estimated from the generated novel views. The results show that the poses estimated from the novel views generated by ViewCrafter align more closely with the ground truth poses, further demonstrating the effectiveness of our point cloud-based pose condition strategy.

Robustness for point cloud condition. ViewCrafter utilizes point cloud render results as conditions, enabling highly accurate pose control. However, these renders may contain artifacts and geometric distortions. Fig. 5 presents an example: the first row shows that the conditioned point cloud renders exhibit occlusions and missing regions, as well as geometric distortions along the boundary of the foreground. The second row displays the corresponding novel views produced by ViewCrafter, demonstrating its ability to fill in the holes and correct the inaccurate geometry. This demonstrate that ViewCrafter has developed a comprehensive understanding

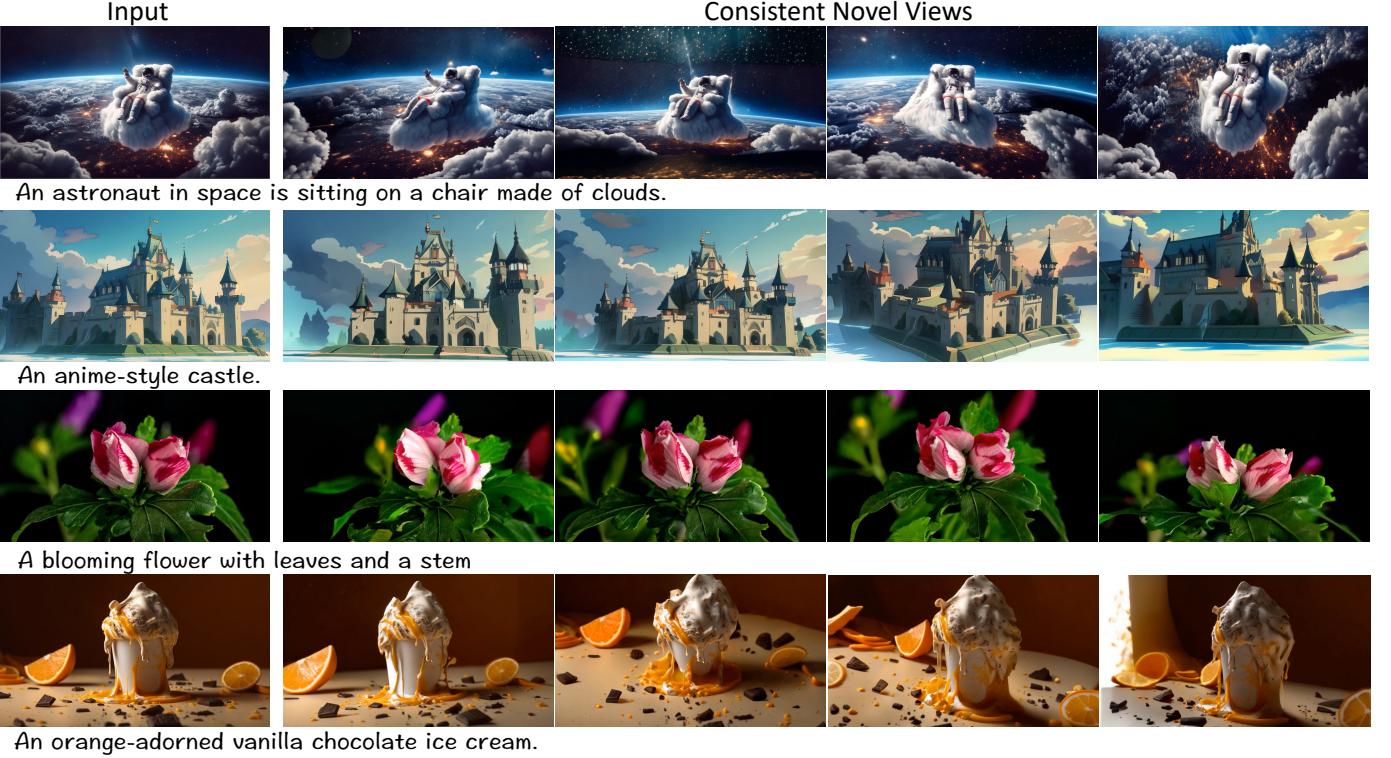


Fig. 9. **Text-to-3D generation results.** The leftmost column displays the input text prompt and the corresponding image generated by T2I model, while the subsequent columns show the consistent novel views produced by our ViewCrafter based on the generated image.

of the 3D world, allowing it to generate high-quality novel views from imperfect conditional information and exhibit robustness to the point cloud conditions.

Ablation on training paradigm. We examine the effectiveness of the adopted training paradigm in this ablation study. To evaluate the choice of training which module of the video denoising U-Net, we compare the novel view synthesis quality of training only the spatial layers and training both the spatial and temporal layers (Ours), the results are shown in the first row of Table. 4. To assess the importance of progressive training, we provide a comparison between directly training the model at 576×1024 resolution and training the model using the progressive training strategy (Ours), the results are reported in the second row of Table. 4. Additionally, we have observed that the inference length of ViewCrafter influences the quality of novel view synthesis. Specifically, for the same range of view change, inference with more frames improves the temporal consistency of the generated frames. To balance the computational cost and synthesis quality, we train two models: a base model that infers 16 frames and a stronger model (Ours) that infers 25 frames. We present a comparison between the two models in the third row of Table. 4. The results above showcase the effectiveness of the implemented training paradigm.

Ablation on camera trajectory planning. In this ablation study, we assess the effectiveness of our proposed camera trajectory planning algorithm for revealing occlusions and completing point clouds. An example is presented in Fig. 8, where we compare the point cloud generated by iterative view synthesis using a predefined circular camera trajec-

tory (similar as [14]) with that produced using our camera trajectory planning algorithm. Given a reference image and an initial point cloud, we adopt a quarter-sphere searching space centered at the origin of the initial point cloud's world coordinate system, with the radius set to the depth of the center pixel of the reference image. We begin with exploring the left half area of the searching space, with the parameters for camera trajectory planning set to $N = 3$, $K = 5$, and $\Theta = 0.6$. Accordingly, the circular camera trajectory is set as uniformly moving the camera three times from the reference pose to the left direction of the searching space, with each movement measuring 20 degrees. Subsequently, we apply the same parameters to explore the right half of the searching space. The final generated point cloud are shown in Fig. 8(a) and Fig. 8(b). In Fig. 8(a), It can be observed that the point cloud reconstructed using the predefined circular camera trajectory results in ineffective completion of the occlusion region. In comparison, Fig. 8(b) presents the reconstruction using our camera trajectory planning algorithm. The more complete reconstruction results demonstrate that it can effectively reveal occlusion regions of the scene, improving the overall scene reconstruction quality.

4.5 Text-to-3D Generation

In addition to synthesizing novel views of real-world images, we also explore the application of combining our framework with creative text-to-image (T2I) diffusion models for text-to-3D generation. To accomplish this, given a text prompt, we first adopt T2I models to generate a correspond-

ing image, then apply ViewCrafter to synthesize consistent novel views. The results are shown in Fig. 9.

5 CONCLUSION AND LIMITATION

This work presents ViewCrafter, a novel view synthesis framework that combines video diffusion models and point cloud priors for high-fidelity and accurate novel view synthesis. Our method overcomes the limitations of existing approaches by providing generalization ability for various scene types and adaptability for both single and sparse image inputs, while maintaining consistency and accuracy in the quality of novel views. Additionally, we introduce an iterative view synthesis method and an adaptive camera trajectory planning procedure that facilitate long-range novel view synthesis and automatic camera trajectory generation for diverse scenes. Beyond novel view synthesis, we explore the efficient optimization of a 3D-GS representation for real-time, high frame-rate novel view rendering, and adapting our framework for text-to-3D generation.

Limitations. Despite its advantages, our method still has several limitations. Firstly, it may encounter challenges in synthesizing novel views with a very large view range given limited 3D clues, such as generating a front-view image from only a back-view image. Additionally, we leverage point clouds as an explicit prior and have validated the robustness of our method for low-quality point clouds. However, challenges may still persist in scenes where the conditioned point clouds are significantly inaccurate. Furthermore, as a video diffusion model, our method necessitates multi-step denoising during the inference process, which requires a relatively higher computing cost.

REFERENCES

- [1] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, “Nerf: Representing scenes as neural radiance fields for view synthesis,” in *ECCV*, 2020.
- [2] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, “3d gaussian splatting for real-time radiance field rendering,” *ACM TOG*, 2023.
- [3] O. Wiles, G. Gkioxari, R. Szeliski, and J. Johnson, “Synsin: End-to-end view synthesis from a single image,” in *CVPR*, 2020.
- [4] R. Rombach, P. Esser, and B. Ommer, “Geometry-free view synthesis: Transformers and no 3d priors,” in *ICCV*, 2021.
- [5] C. Rockwell, D. F. Fouhey, and J. Johnson, “Pixelsynth: Generating a 3d-consistent experience from a single image,” in *ICCV*, 2021.
- [6] B. Park, H. Go, and C. Kim, “Bridging implicit and explicit geometric transformation for single-image view synthesis,” *IEEE TPAMI*, 2024.
- [7] T. Zhou, R. Tucker, J. Flynn, G. Fyffe, and N. Snavely, “Stereo magnification: Learning view synthesis using multiplane images,” *ACM TOG*, 2018.
- [8] Y. Han, R. Wang, and J. Yang, “Single-view view synthesis in the wild with learned adaptive multiplane images,” in *SIGGRAPH Conference*, 2022.
- [9] R. Tucker and N. Snavely, “Single-view view synthesis with multiplane images,” in *CVPR*, 2020.
- [10] A. Yu, V. Ye, M. Tancik, and A. Kanazawa, “pixelnerf: Neural radiance fields from one or few images,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 4578–4587.
- [11] R. Liu, R. Wu, B. Van Hoorick, P. Tokmakov, S. Zakharov, and C. Vondrick, “Zero-1-to-3: Zero-shot one image to 3d object,” in *ICCV*, 2023.
- [12] K. Sargent, Z. Li, T. Shah, C. Herrmann, H.-X. Yu, Y. Zhang, E. R. Chan, D. Lagun, L. Fei-Fei, D. Sun, and J. Wu, “ZeroNVS: Zero-shot 360-degree view synthesis from a single real image,” in *CVPR*, 2024.
- [13] Z. Wang, Z. Yuan, X. Wang, T. Chen, M. Xia, P. Luo, and Y. Shan, “Motionctrl: A unified and flexible motion controller for video generation,” in *SIGGRAPH Conference*, 2024.
- [14] J. Chung, S. Lee, H. Nam, J. Lee, and K. M. Lee, “Luciddreamer: Domain-free generation of 3d gaussian splatting scenes,” *arXiv preprint arXiv:2311.13384*, 2023.
- [15] J. Shriram, A. Trevithick, L. Liu, and R. Ramamoorthi, “Realm-dreamer: Text-driven 3d scene generation with inpainting and depth diffusion,” *arXiv preprint arXiv:2404.07199*, 2024.
- [16] P.-Y. Lab and T. A. etc., “Open-sora-plan,” <https://github.com/PKU-YuanGroup/Open-Sora-Plan>, 2024. [Online]. Available: <https://doi.org/10.5281/zenodo.10948109>
- [17] A. Blattmann, T. Dockhorn, S. Kulal, D. Mendelevitch, M. Kilian, D. Lorenz, Y. Levi, Z. English, V. Voleti, A. Letts *et al.*, “Stable video diffusion: Scaling latent video diffusion models to large datasets,” *arXiv preprint arXiv:2311.15127*, 2023.
- [18] J. Xing, M. Xia, Y. Zhang, H. Chen, X. Wang, T.-T. Wong, and Y. Shan, “Dynamicrafter: Animating open-domain images with video diffusion priors,” *arXiv preprint arXiv:2310.12190*, 2023.
- [19] S. Wang, V. Leroy, Y. Cabon, B. Chidlovskii, and J. Revaud, “Dust3r: Geometric 3d vision made easy,” in *CVPR*, 2024.
- [20] V. Leroy, Y. Cabon, and J. Revaud, “Grounding image matching in 3d with mast3r,” *arXiv:2406.09756*, 2024.
- [21] A. Knapitsch, J. Park, Q.-Y. Zhou, and V. Koltun, “Tanks and temples: Benchmarking large-scale scene reconstruction,” *ACM TOG*, 2017.
- [22] J. Reizenstein, R. Shapovalov, P. Henzler, L. Sbordone, P. Labatut, and D. Novotny, “Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction,” in *ICCV*, 2021.
- [23] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in *NeurIPS*, 2017.
- [24] A. Trevithick, M. Chan, M. Stengel, E. Chan, C. Liu, Z. Yu, S. Khamis, M. Chandraker, R. Ramamoorthi, and K. Nagano, “Real-time radiance fields for single-image portrait view synthesis,” *ACM TOG*, 2023.
- [25] W. Yu, Y. Fan, Y. Zhang, X. Wang, F. Yin, Y. Bai, Y.-P. Cao, Y. Shan, Y. Wu, Z. Sun *et al.*, “Nofa: Nerf-based one-shot facial avatar reconstruction,” in *SIGGRAPH Conference*, 2023.
- [26] Y. Hong, K. Zhang, J. Gu, S. Bi, Y. Zhou, D. Liu, F. Liu, K. Sunkavalli, T. Bui, and H. Tan, “Lrm: Large reconstruction model for single image to 3d,” in *ICLR*, 2024.
- [27] D. Charatan, S. L. Li, A. Tagliasacchi, and V. Sitzmann, “pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction,” in *CVPR*, 2024.
- [28] Y. Chen, H. Xu, C. Zheng, B. Zhuang, M. Pollefeys, A. Geiger, T.-J. Cham, and J. Cai, “Mvsplat: Efficient 3d gaussian splatting from sparse multi-view images,” in *ECCV*, 2024.
- [29] J. Ho, A. Jain, and P. Abbeel, “Denoising diffusion probabilistic models,” in *NeurIPS*, 2020.
- [30] J. Song, C. Meng, and S. Ermon, “Denoising diffusion implicit models,” in *ICLR*, 2021.
- [31] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, “High-resolution image synthesis with latent diffusion models,” in *CVPR*, 2022.
- [32] W. Yu, L. Yuan, Y.-P. Cao, X. Gao, X. Li, L. Quan, Y. Shan, and Y. Tian, “Hifi-123: Towards high-fidelity one image to 3d content generation,” in *ECCV*, 2024.
- [33] J. Tang, T. Wang, B. Zhang, T. Zhang, R. Yi, L. Ma, and D. Chen, “Make-it-3d: High-fidelity 3d creation from a single image with diffusion prior,” in *ICCV*, 2023.
- [34] J. Sun, B. Zhang, R. Shao, L. Wang, W. Liu, Z. Xie, and Y. Liu, “Dreamcraft3d: Hierarchical 3d generation with bootstrapped diffusion prior,” in *ICLR*, 2024.
- [35] E. R. Chan, K. Nagano, M. A. Chan, A. W. Bergman, J. J. Park, A. Levy, M. Aittala, S. De Mello, T. Karras, and G. Wetzstein, “Generative novel view synthesis with 3d-aware diffusion models,” in *ICCV*, 2023.
- [36] M. Deitke, D. Schwenk, J. Salvador, L. Weihs, O. Michel, E. VanDerBilt, L. Schmidt, K. Ehsani, A. Kembhavi, and A. Farhadi, “Objaverse: A universe of annotated 3d objects,” in *CVPR*, 2023.
- [37] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su *et al.*, “Shapenet: An information-rich 3d model repository,” *arXiv preprint arXiv:1512.03012*, 2015.

- [38] D. Watson, W. Chan, R. M. Brualla, J. Ho, A. Tagliasacchi, and M. Norouzi, "Novel view synthesis with diffusion models," in *ICLR*, 2023.
- [39] J. Reizenstein, R. Shapovalov, P. Henzler, L. Sbordone, P. Labatut, and D. Novotny, "Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction," in *ICCV*, 2021.
- [40] X. Yu, M. Xu, Y. Zhang, H. Liu, C. Ye, Y. Wu, Z. Yan, C. Zhu, Z. Xiong, T. Liang *et al.*, "Mvimgnet: A large-scale dataset of multi-view images," in *CVPR*, 2023.
- [41] R. Wu, B. Mildenhall, P. Henzler, K. Park, R. Gao, D. Watson, P. P. Srinivasan, D. Verbin, J. T. Barron, B. Poole *et al.*, "Reconfusion: 3d reconstruction with diffusion priors," in *CVPR*, 2024.
- [42] J. Zhang, X. Li, Z. Wan, C. Wang, and J. Liao, "Text2nerf: Text-driven 3d scene generation with neural radiance fields," *IEEE TVCG*, 2024.
- [43] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *CVPR*, 2022.
- [44] L. Zhang, A. Rao, and M. Agrawala, "Adding conditional control to text-to-image diffusion models," in *ICCV*, 2023.
- [45] C. Mou, X. Wang, L. Xie, Y. Wu, J. Zhang, Z. Qi, and Y. Shan, "T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models," in *AAAI*, 2024.
- [46] Y. Li, H. Liu, Q. Wu, F. Mu, J. Yang, J. Gao, C. Li, and Y. J. Lee, "Gligen: Open-set grounded text-to-image generation," in *CVPR*, 2023.
- [47] J. Xing, H. Liu, M. Xia, Y. Zhang, X. Wang, Y. Shan, and T.-T. Wong, "Tooncrafter: Generative cartoon interpolation," *arXiv preprint arXiv:2405.17933*, 2024.
- [48] J. Xing, M. Xia, Y. Liu, Y. Zhang, Y. Zhang, Y. He, H. Liu, H. Chen, X. Cun, X. Wang *et al.*, "Make-your-video: Customized video generation using textual and structural guidance," *IEEE TVCG*, 2024.
- [49] P. Esser, J. Chiu, P. Atighehchian, J. Granskog, and A. Germanidis, "Structure and content-guided video synthesis with diffusion models," in *ICCV*, 2023.
- [50] S. Yin, C. Wu, J. Liang, J. Shi, H. Li, G. Ming, and N. Duan, "Drag-nuwa: Fine-grained control in video generation by integrating text, image, and trajectory," *arXiv preprint arXiv:2308.08089*, 2023.
- [51] M. Niu, X. Cun, X. Wang, Y. Zhang, Y. Shan, and Y. Zheng, "Mofa-video: Controllable image animation via generative motion field adaptions in frozen image-to-video diffusion model," *arXiv preprint arXiv:2405.20222*, 2024.
- [52] E. Peruzzo, V. Goel, D. Xu, X. Xu, Y. Jiang, Z. Wang, H. Shi, and N. Sebe, "Vase: Object-centric appearance and shape manipulation of real videos," *arXiv preprint arXiv:2401.02473*, 2024.
- [53] Y. Guo, C. Yang, A. Rao, Y. Wang, Y. Qiao, D. Lin, and B. Dai, "Animatediff: Animate your personalized text-to-image diffusion models without specific tuning," *arXiv preprint arXiv:2307.04725*, 2023.
- [54] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," in *ICLR*, 2022.
- [55] N. Müller, K. Schwarz, B. Rössle, L. Porzi, S. R. Bulò, M. Nießner, and P. Kontschieder, "Multidiff: Consistent novel view synthesis from a single image," in *CVPR*, 2024.
- [56] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "Scannet: Richly-annotated 3d reconstructions of indoor scenes," in *CVPR*, 2017.
- [57] D. Xu, W. Nie, C. Liu, S. Liu, J. Kautz, Z. Wang, and A. Vahdat, "Camco: Camera-controllable 3d-consistent image-to-video generation," *arXiv preprint arXiv:2406.02509*, 2024.
- [58] H. He, Y. Xu, Y. Guo, G. Wetstein, B. Dai, H. Li, and C. Yang, "Cameractrl: Enabling camera control for text-to-video generation," *arXiv preprint arXiv:2404.02101*, 2024.
- [59] V. Sitzmann, S. Rezhikov, B. Freeman, J. Tenenbaum, and F. Durand, "Light field networks: Neural scene representations with single-evaluation rendering," in *NeurIPS*, 2021.
- [60] G. Metzer, E. Richardson, O. Patashnik, R. Giryes, and D. Cohen-Or, "Latent-nerf for shape-guided generation of 3d shapes and textures," in *CVPR*, 2023.
- [61] F. Plastria, "The weiszfeld algorithm: proof, amendments and extensions, ha eiselt and v. marianov (eds.) foundations of location analysis, international series in operations research and management science," 2011.
- [62] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *ICML*, 2021.
- [63] R. Gao, A. Holynski, P. Henzler, A. Brussee, R. Martin-Brualla, P. Srinivasan, J. T. Barron, and B. Poole, "Cat3d: Create anything in 3d with multi-view diffusion models," *arXiv preprint arXiv:2405.10314*, 2024.
- [64] R. Zeng, Y. Wen, W. Zhao, and Y.-J. Liu, "View planning in robot active vision: A survey of systems, algorithms, and applications," *Computational Visual Media*, vol. 6, 2020.
- [65] H. Dhami, V. D. Sharma, and P. Tokekhar, "Pred-nbv: Prediction-guided next-best-view planning for 3d object reconstruction," in *IROS*, 2023.
- [66] L. Jin, X. Chen, J. Rückin, and M. Popović, "Neu-nbv: Next best view planning using uncertainty estimation in image-based neural rendering," in *IROS*, 2023.
- [67] S. Isler, R. Sabzevari, J. Delmerico, and D. Scaramuzza, "An information gain formulation for active volumetric 3d reconstruction," in *ICRA*, 2016.
- [68] D. Peralta, J. Casimiro, A. M. Nilles, J. A. Aguilar, R. Atienza, and R. Cajote, "Next-best view policy for 3d reconstruction," in *ECCVW*, 2020.
- [69] M.-L. Shih, S.-Y. Su, J. Kopf, and J.-B. Huang, "3d photography using context-aware layered depth inpainting," in *CVPR*, 2020.
- [70] L. Ling, Y. Sheng, Z. Tu, W. Zhao, C. Xin, K. Wan, L. Yu, Q. Guo, Z. Yu, Y. Lu *et al.*, "Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision," in *CVPR*, 2024.
- [71] N. Ravi, J. Reizenstein, D. Novotny, T. Gordon, W.-Y. Lo, J. Johnson, and G. Gkioxari, "Accelerating 3d deep learning with pytorch3d," *arXiv:2007.08501*, 2020.
- [72] J. Ho and T. Salimans, "Classifier-free diffusion guidance," *arXiv preprint arXiv:2207.12598*, 2022.
- [73] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE TIP*, vol. 13, no. 4, pp. 600–612, 2004.
- [74] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *CVPR*, 2018.
- [75] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," in *NeurIPS*, 2017.
- [76] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *CVPR*, 2016.
- [77] J. Li, J. Zhang, X. Bai, J. Zheng, X. Ning, J. Zhou, and L. Gu, "Dngaussian: Optimizing sparse-view 3d gaussian radiance fields with global-local depth normalization," in *CVPR*, 2024.
- [78] Z. Zhu, Z. Fan, Y. Jiang, and Z. Wang, "Fsgs: Real-time few-shot view synthesis using gaussian splatting," in *ECCV*, 2024.
- [79] Z. Fan, W. Cong, K. Wen, K. Wang, J. Zhang, X. Ding, D. Xu, B. Ivanovic, M. Pavone, G. Pavlakos *et al.*, "Instantsplat: Unbounded sparse-view pose-free gaussian splatting in 40 seconds," *arXiv:2403.20309*, 2024.
- [80] S. Bahmani, I. Skorokhodov, A. Siarohin, W. Menapace, G. Qian, M. Vasilkovsky, H.-Y. Lee, C. Wang, J. Zou, A. Tagliasacchi *et al.*, "Vd3d: Taming large video diffusion transformers for 3d camera control," *arXiv preprint arXiv:2407.12781*, 2024.
- [81] Y.-B. Jia, "Plücker coordinates for lines in the space," *Problem Solver Techniques for Applied Computer Science, Com-S-477/577 Course Handout*, 2020.