DiffDreamer: Towards Consistent Unsupervised Single-view Scene Extrapolation with Conditional Diffusion Models

Shengqu Cai\textsuperscript{1,2,*} Eric Ryan Chan\textsuperscript{1} Songyou Peng\textsuperscript{2,3} Mohamad Shahbazi\textsuperscript{2} Anton Obukhov\textsuperscript{2} Luc Van Gool\textsuperscript{2,4} Gordon Wetzstein\textsuperscript{1}

\textsuperscript{1}Stanford University \textsuperscript{2}ETH Zürich \textsuperscript{3}MPI for Intelligent Systems, Tübingen \textsuperscript{4}KU Leuven

Figure 1: \textbf{DiffDreamer} (Top) is a novel diffusion-based approach for scene extrapolation. It exhibits high spatio-temporal consistency, a desired property missing in prior art, such as InfNat-0 \textsuperscript{[40]} (Bottom). We check for consistency by extracting keypoints from the sequences with COLMAP, resulting in point clouds of vastly different sizes and sparsity (Right).

Abstract

Scene extrapolation—the idea of generating novel views by flying into a given image—is a promising, yet challenging task. For each predicted frame, a joint inpainting and 3D refinement problem has to be solved, which is ill posed and includes a high level of ambiguity. Moreover, training data for long-range scenes is difficult to obtain and usually lacks sufficient views to infer accurate camera poses. We introduce DiffDreamer, an unsupervised framework capable of synthesizing novel views depicting a long camera trajectory while training solely on internet-collected images of nature scenes. Utilizing the stochastic nature of the guided denoising steps, we train the diffusion models to refine projected RGBD images but condition the denoising steps on multiple past and future frames for inference. We demonstrate that image-conditioned diffusion models can effectively perform long-range scene extrapolation while preserving consistency significantly better than prior GAN-based methods. DiffDreamer is a powerful and efficient solution for scene extrapolation, producing impressive results despite limited supervision. Project page: https://primecai.github.io/diffdreamer.

1. Introduction

3D content creation tools are the foundation of emerging metaverse applications, among many others. Current approaches primarily rely on heavy manual labor, making the process expensive and inefficient. We set out to make 3D content creation automated and accessible. More specifically, an important downstream task we approach is \textit{consistent scene extrapolation}. Given a single image and a long camera trajectory flying into the scene, the goal of \textit{consistent scene extrapolation} is to synthesize a multiview-consistent 3D scene along the camera trajectory. In other words, we want to teach a machine to hallucinate content when flying into the image while maintaining multiview consistency, thereby extrapolating the scene realistically. Successfully addressing this task opens up a wide range of potential applications in virtual reality, 3D content creation, synthetic data creation, and 3D viewing platforms.

Consistent scene extrapolation is extremely challenging as it tries to tackle two difficult tasks simultaneously: consistent single-view novel view synthesis (NVS) and long-range extrapolation. Consistent single-view novel view synthesis has been studied for a long time. Many meth-
ods [67, 49] propose utilizing multi-view data to infer the correspondences between frames, but they generally do not scale down to single- or few-view settings. Recently, there have also been attempts at single-view novel view synthesis. These methods mostly rely on learning a prior [101, 8] or utilizing geometry information [99, 97, 68]. However, they do not generalize to long-range camera movement, as the content of the original image is quickly lost when taking large camera movements. Current methods of long-range extrapolation [42, 40, 65, 36, 70] employ per-frame generation protocols, where the frames are generated in an autoregressive feed-forward manner. The common downside of these methods is the lack of consistency between subsequent frames due to the per-frame refinement. A few recent methods [20, 5] attempt to generate a whole scene directly using implicit representations. However, this setting is computationally expensive, causing them to fail to achieve photo-realism even on low-resolution synthetic data.

Very recently, efforts have been made to perform scene extrapolation using pre-trained large-scale text-to-image diffusion models [61, 69]. This line of methods relies on prompt engineering and produces results with jittering between frames due to the lack of consistency enforcement. However, image-conditioned diffusion models are naturally suitable for the task of scene extrapolation, as the guided denoising process can be interpreted as a search in the latent space. Compared to feed-forward GAN-based methods [42, 40, 36], this allows the model to preserve high-level semantic meaning and low-frequency features while adding in high-frequency details and in-painting the missing parts. We hence utilize these strengths of diffusion models for scene exploration and further improve its 3D consistency.

In this paper, we propose DiffDreamer, a fully unsupervised method capable of consistent scene extrapolation given only a single image as input, and only internet photo collections as training data. Inspired by recent success in diffusion-based image refinement [44, 71], we formulate consistent scene extrapolation as learning a conditional diffusion model from images only. We train the conditional diffusion model to generate the frames in an iterative refinement manner, showing that this allows convergence towards a harmonic set of frames with high fidelity. The consistency achieved by our conditional diffusion model potentially enables one to fuse the outputs as a 3D model, e.g., a NeRF [49] with high consistency score [96]. A key advantage of diffusion models is the flexibility of modifying their sampling behaviors at inference. By stochastic conditioning at inference, we can condition the generation on multiple past and future frames and form a bidirectional pipeline, despite having only single images during training.

Experiments demonstrate that our framework allows one to synthesize a long-range fly-through sequence into an RGB image. We believe our framework not only serves as a starting point for consistent scene extrapolation and diffusion-model-based novel view synthesis but also scene extrapolation on more complex large-scale scenarios, such as autonomous driving scenes.

Our contributions include

- We introduce DiffDreamer, the first single-view scene extrapolation framework based on diffusion models for large-scale scenes.
- We propose an anchored sampling strategy and a lookahead mechanism for long-range scene extrapolation. Combined with diffusion models, we significantly alleviate the well-known domain drifting issue [42, 40] of scene extrapolation.
- We demonstrate a fully automated scene-level novel view synthesis pipeline using conditional diffusion models.

2. Related works

**Novel view synthesis from multi-view images** Research related to Novel View Synthesis (NVS) has a long history. Traditional multi-view NVS relies on inferring underlying geometry and interpolating the input images [58, 12, 18, 19, 22, 37, 15, 25, 39, 78, 109]. Recent successful attempts utilize deep learning methods to construct scene representations from multi-view data. These scene representations include but are not limited to: depth images [1, 47, 66, 100, 92, 81], multi-plane images [90, 107], voxels [43, 84], and implicit functions [85, 55, 57]. Among these representations, major progress has been made on radiance field approaches. Neural Radiance Fields (NeRFs) [49] have demonstrated encouraging progress for view synthesis by encoding color and transmittance in a multilayer perceptron, hence encoding a scene as an implicit representation. Using volumetric rendering, NeRF can perform photo-realistic novel view synthesis from only multi-view captured images and their poses. The outstanding performance of NeRF attracts tremendous efforts to improve its performance [3, 4, 93, 16], accelerate its training [50, 74, 88, 54], speed up rendering [41, 83], extend or generalize it towards other downstream tasks [45, 48, 9, 10, 77, 51, 26, 108], etc. Unlike these multi-view methods, we assume a single input image at the inference stage, where no geometry or interpolation can be inferred easily. Even with a single input image, our method can produce novel views faithfully.

**Novel view synthesis from a single image** There has been a vein of research on single-shot NVS. Some of them rely on geometry information or annotations [52, 82, 91]. However, geometry information and annotations are usually expensive to obtain for in-the-wild images. Other methods [17, 101, 8, 92, 85, 33, 89, 2] relax this constraint by learning a prior filling in the missing information. These
methods typically either only work well on simple objects (e.g., ShapeNet [11]) or restrict camera motions to small regions around the reference view. In contrast, we aim to relax such constraints and target long-range view extrapolation.

**Scene extrapolation** Long-range view extrapolation requires going beyond observations. With recent progress in generative modeling, several view extrapolation methods have emerged [7, 14, 38, 32, 35, 56, 79, 94, 104, 70]. Earlier methods such as SynSin [97] perform inpainting after reprojection, which struggles after a very limited range. A follow-up work, PixelSynth [68], works similarly to our DiffDreamer; it performs large-step image outpainting and accumulates a 3D point cloud for intermediate refinement and rendering. However, PixelSynth does not generalize to larger camera movements and requires a refinement module on top of the point cloud to enhance and inpaint, causing severely inconsistent intermediate view synthesis results.

**Long-term path synthesis** State-of-the-art methods such as InfNat [42], InfNat-0 [40], PathDreamer [36] and LOTR [65] deploy iterative training protocols and achieve perpetual view extrapolation for extremely long camera trajectories. However, these methods work in an autoregressive per-frame generation framework. As a consequence, severe inconsistency can be observed from their rendered frames. Solving such inconsistency potentially requires generating an entire 3D world model, which is extremely computationally expensive, as shown by previous works [20, 5], whilst feed-forward per-frame generation methods [70, 42, 40] suffer from content drifting and both local and global inconsistency. Therefore, we attempt to benefit from a diffusion-based iterative refinement method to generate consistent content.

**Diffusion models** The recent development of diffusion models [28, 87] has pushed AI-driven content creation to another level. These methods learn to transform any data distribution into a prior distribution, then sample new data by first sampling a random latent vector from the prior distribution, followed by ancestral sampling with the reverse Markov chain, parameterized by deep neural networks. The powerful diffusion/denoising mechanism enables various traditional image-based tasks, including super-resolution [72, 29], inpainting [44, 71], and editing [46]. Their well-defined steady training protocols enhance the diffusion models’ performance for large-scale training [69, 61]. Very recently, success has been made to lift the strength of diffusion models to the 3D domain [39, 96], further demonstrating the potential of 3D-based diffusion models.

We formulate our task similarly to InfNat [42] and InfNat-0 [40], but use a conditional diffusion model instead of GANs and take focus on achieving consistency.

### 3. DiffDreamer

Given a single input image, the aim of DiffDreamer is to generate a consistent and harmonic 3D camera trajectory that represents flying into the given image. DiffDreamer addresses this task by training a conditional diffusion model to perform image inpainting and refinement concurrently.

---

**Figure 2: Overview of our pipeline.** We train an image-conditional diffusion model to perform image-to-image refinement and inpainting given a corrupted image and its missing region mask. At inference, we perform stochastic conditioning on three conditionings: naive forward warping from the previous frame (black arrow), anchored conditioning by warping a further frame (blue arrow), and lookahead conditioning by warping a virtual future frame (red arrow). We repeat this render-refine-repeat pipeline to get sequences extrapolating a given image.
The overview of our pipeline is illustrated in Fig. 2.

We synthesize frames of a fly-through video with three steps: render, refine and repeat. In detail, given an RGB image \( I_0 \) with its monocular predicted disparity \( D_0 \) located at camera pose \( c_0 \), we can unproject the colored pixels into 3D space and render the projected view at the next camera pose \( c_{i+1} \) by a 3D renderer \([63]\) \( \pi: (I_{i+1}, D_{i+1}) = \pi(I_0, D_0, c_i, c_{i+1}) \). With a refinement network \( F_\theta \), the warped RGBD image \( (I_{i+1}^{\text{warped}}, D_{i+1}^{\text{warped}}) \) can be inpainted and refined to get a fine next frame \( (I_{i+1}, D_{i+1}) = F_\theta(I_{i+1}^{\text{warped}}, D_{i+1}^{\text{warped}}) \), shown by the black arrow flow in Fig. 2. We then treat \( (I_{i+1}, D_{i+1}) \) as the starting view of the next step and perform the warping and refinement stages repeatedly, yielding a set of frames extrapolating the scene.

### 3.1. Training

Prior works \([42, 40]\) model the training process exactly as the render-refine-repeat pipeline since the process is fully differentiable. However, this naive approach is not generalizable to diffusion models for two reasons: 1) the training process of diffusion models is split into different noise levels, and 2) sampling from a diffusion model requires up to thousands of denoising steps. This means we need to store thousands of intermediate steps and gradients to perform back-propagation, which is computationally infeasible.

The main function of the “repeat” step during training is to feed the network with its own outputs to ameliorate distribution drifting \([42]\). Therefore, it becomes critical to replace this step, especially when diffusion models are known to be sensitive to input distribution. Firstly, we create training pairs similar to \([40]\) by projecting a ground truth RGBD image \( (I_{gt}, D_{gt}) \) at initial camera pose \( c_0 \) to a pseudo previous camera pose \( c_{\text{pseudo}}: (I_{\text{pseudo}}, D_{\text{pseudo}}) = \pi(I_0, D_0, c_0, c_{\text{pseudo}}) \), then project back to \( c_0 \) to create a corresponding corrupted RGBD: \( (I_{\text{corrupted}}, D_{\text{corrupted}}) = \pi(I_{\text{pseudo}}, D_{\text{pseudo}}, c_{\text{pseudo}}, c_0) \). We thus obtain a pair of ground truth RGBD images \( (I_{gt}, D_{gt}) \) and its corrupted version \( (I_{\text{corrupted}}, D_{\text{corrupted}}) \), which involves missing parts and warping artifacts simulating a forward motion.

Having these paired data enables training an image-conditioned diffusion model \( p(y|x, m) \), where \( x = (I_{\text{corrupted}}, D_{\text{corrupted}}) \), a corrupted version of ground truth image \( y = (I_{gt}, D_{gt}) \) while \( m \) denotes the missing region mask from warping. We train the model with the following objective \([28, 71]\):

\[
L(\theta) = \mathbb{E}_{(x, m, y)} \mathbb{E}_{\epsilon, \gamma} \left[ ||f_\theta(x, m, \tilde{y}, \gamma) - \epsilon||_2^2 \right],
\]

where \( \tilde{y} = \sqrt{\gamma} y + \sqrt{1-\gamma} \epsilon, \epsilon \sim \mathcal{N}(0, I) \) , and \( \gamma \) indicates the noise level. Note that since the diffusion model needs to concurrently learn inpainting and refinement, we additionally condition the neural network on the missing region mask to provide stronger guidance, following prior works \([42, 40]\). Similar to previous work \([40]\), we assume that the sky region lies infinitely far away and does not change. Therefore, we inject noise only to the ground region to obtain \( \tilde{y} \).

### 3.2. Inference

Diffusion models trained as described above perform well for a single forward step but do not generalize to long-term due to severe domain drifting after only a few iterations. This causes the extrapolated results to gradually drift away after only a handful of steps (see Sec. 4.2). We propose two strategies at the inference stage to counter this issue and preserve both local and global consistency.

#### 3.2.1 Anchored conditioning

We introduce anchored conditioning, which conditions the diffusion model on long-range camera movements in order to enhance consistency over larger distances. As shown in prior work \([96]\), it is feasible to naively approximate true auto-regressive sampling via stochastic conditioning for conditional diffusion models. While moving forward from camera pose \( c_0 \) to \( c_{i+1} \), instead of strictly conditioning on the warped previous image \( (I_{\text{warped}}, D_{\text{warped}}) = \pi(I_i, D_i, c_i, c_{i+1}) \) and mask during the inference denoising stage, we additionally select a frame \( (I_{far}, D_{far}) \) at a previous camera pose \( c_{far} \) further away from the current camera position. In practice, we empirically select the current frame \( (I_i, D_i) \) every 5 steps. We then perform stochastic conditioning on the warped previous frame \( (I_{\text{warped}}, D_{\text{warped}}) \) and an “anchored” frame by warping \( (I_{far}, D_{far}) \) to the desired camera pose: \( (I_{\text{anchored}}, D_{\text{anchored}}) = \pi(I_{far}, D_{far}, c_{far}, c_{i+1}) \). We perform this conditioning without specifying the missing region mask, as anchored conditioning requires longer-range warping, which may introduce more regions as missing, thus undermining the goal of long-term consistency. Conditioning on the warped previous frame naively encourages frame-to-frame consistency, while conditioning on a far-away frame offers long-term consistency. Stochastic conditioning on a largely warped image is also helpful with respect to domain drifting, as it is easier to simulate the same artifacts and blurriness during training by simply warping ground truth images equally further away. Thanks to the diffusion models’ steady training protocol, refining and inpainting largely warped images with massive missing areas can be learned jointly during training.

#### 3.2.2 Virtual lookahead conditioning

Prior works \([42, 40]\) deploy per-frame generation; therefore, they suffer from severe inconsistency. A straightforward approach to solve this is to generate a scene representation directly, but this is extremely challenging and expen-
sive to perform. While anchored conditioning solves parts of the global consistency issue, warping an image distorts and stretches the texture, and blurs out fine details.

We notice that compared with flying into an image and refining the artifacts and blurriness, it is significantly easier to zoom out of an image and outpaint the missing regions without suffering from domain drift. This is due to the available regions preserving high-frequency details, which confer a strong signal for filling in missing regions.

Therefore, adding in a “lookahead” mechanism in diffusion models is helpful for both achieving long-term consistency and preventing domain drifting, as we can benefit from conditioning the generation on a future image, whose fine details preserve after warping to the current pose. While flying deep into an image, we observe that the generated content shares little overlap with the input image. Utilizing this fact, we can create a virtual view lying ahead for a sequence of views. With stochastic conditioning, this is as simple as additionally conditioning the generation on \((I_{\text{future}}, D_{\text{future}})\), acquired by warping a shared virtual view \((I_n, D_n)\) lying ahead at a shared virtual future camera pose \(c_n\) warped to each camera pose \(c_i\): \((I_{\text{future}}, D_{\text{future}}) = \pi(I_n, D_n, c_n, c_i)\). In practice, we empirically generate \(c_n\) by taking a forward motion 10 times larger than a single step. We update the shared \((I_n, D_n)\) every 10 steps and condition the future 10 frames on it. The shared virtual view can be flexibly generated by refining an available view to a future camera pose, a randomly generated view, or even another real image. To preserve consistency, we find it is suitable to warp the current frame \((I_i, D_i)\) to a future camera pose \(c_n\) significantly beyond a single forward motion so that there is enough ambiguity, then refine to get virtual lookahead conditioning \((I_{\text{future}}, D_{\text{future}})\).

With the proposed anchored and virtual lookahead conditioning, we can formulate each denoising step going from camera pose \(c_i\) to \(c_{i+1}\) as:

\[
y_{t-1} \leftarrow \frac{1}{\sqrt{\alpha_t}} \left( y_t - \frac{1 - \alpha_t}{\sqrt{1 - \gamma_t}} f_0(\mathbf{x}_i, m_i, y_t, \gamma_t) \right) + \sqrt{1 - \alpha_t} \mathbf{e}_t
\]

at the inference stage, where \(\mathbf{x}_i\) is a weighted selection of \((0.5, 0.25, 0.25)\) among \((I_{\text{warped}}, D_{\text{warped}}), (I_{\text{anchored}}, D_{\text{anchored}})\) and \((I_{\text{future}}, D_{\text{future}})\), with \(m_i\) being the missing region mask. We apply classifier-free guidance [30] during inference as it encourages the denoising process to take more signals from the conditioning.

### 3.3. Training details

We carefully design the training protocols according to our inference strategies. Firstly, instead of assuming a fixed step size like previous works [42, 40], we randomly choose step size from a range \((-s, s)\) while training the model to generalize to long-range conditioning, where we empirically choose \(s = 20\). Note that we also train the model to fly out of the image for the purpose of lookahead conditioning. We find injecting random Gaussian noise into the missing regions rather than preserving the stretched details or masking out the missing regions to be very helpful, as it serves as an additional latent space that encourages diversity and effectively reduces the domain drifting between
forward motions and circular motions we used for creating pseudo training pairs. To add support for classifier-free guidance [31] at inference, we zero out all conditioning inputs with 10% probability during training.

4. Experiments

4.1. Evaluation

Datasets We report results on the LHQ [86] dataset, a collection of 90K nature landscape photos. Following prior work [40], we use the full data for training and 100 images provided by [40]’s authors, generated from a pre-trained StyleGAN2 [34], as the test set. Following [42], we also supply quantitative results on the ACID [42] dataset, with evaluation on 50 input images from its test set.

Evaluation metrics Evaluation of scene extrapolation frameworks is non-trivial as there is no single evaluation metric that covers every aspect of the generation quality. We follow the evaluation protocol of prior works [42, 40] on the rendered sequences. For that, we report Inception Score (IS) [73], Frechet Inception Distance (FID) [27] and Kernel Inception Distance (KID) [6] with scaling factor $\times 10$ computed using the torch-fidelity package [53]. We evaluate the models in two settings: a shorter range of 20 refinement steps, a middle range of 50 steps, and a longer range of 100 refinement steps. We additionally report 3D consistency scoring [96], a recent metric for evaluating 3D consistency. We compute this metric by generating a sequence of frames from an input, training a neural field [49] with a fraction of the generated frames, and calculating PSNR (Peak Signal to Noise Ratio), SSIM (Structural Similarity Index Measure) [95] and Perceptual Similarity (LPIPS) [105] against the held-out generated frames. We evaluate 3D consistency scoring over 10 sequences of 30 frames from 10 randomly selected input images. As our prior works [42, 40] also output disparity maps, we use a disparity supervised DVOv2 [88] as the underlying neural field model. Following [96], we zero out the viewing direction conditioning to avoid overfitting to view-dependencies. To further assess local consistency, we compare the number of points from COLMAP [75, 76] reconstruction on the same 10 rendered video sequences, using the default automatic reconstruction without specifying the camera poses.

Qualitative results Qualitative comparisons are shown in Fig. 3, where we ask the rendering models to fly toward a target. We compare the intermediate frames rendered by the model. Interestingly, we find that even with identical input depth and step size, it takes significantly more refinement steps for InfNat-0 [40] to reach a target due to a tendency to render the far plane to be further than the mesh projection. In contrast, our DiffDreamer maintains high 3D consistence, and because it takes fewer steps to reach a target, it exhibits less drift. However, to ensure a fair comparison with [40] and to evaluate our model’s drift under more refinement steps, we manually insert additional intermediate camera poses between each pair of nearby autocruise [42] poses so that the models conduct exactly the same number of refinement steps to reach the final frame.

We compare DiffDreamer’s consistency against InfNat-0 in Fig. 5, along with a reference mesh projection created by warping the initial image into the final view according to the estimated disparity map. With identical disparity map and step size provided by InfNat-0 [40]’s authors, our model shows significantly better alignment with the projected mesh. While InfNat-0 [40] renders decent intermediate frames, it demonstrates only limited consistency with the mesh. Although it accurately models the expected movement of the foreground, the hill in the mid-distance remains static, whereas we expect it to fill the frame as the camera flies forward. This artifact may be due to a bias that encourages maintaining useful distant content while training scene extrapolation models. By enforcing the look-ahead mechanism, which enforces future frames to be consistent with the mesh projection, our DiffDreamer does not suffer from this issue. We show an example of the detail-preserving ability of DiffDreamer in Fig 6.

We additionally visualize examples of scene extrapolation of 50 steps in Fig. 4. Despite only seeing single images during training, the learned generative prior enables DiffDreamer to perform long-range extrapolation.

Table 1: Quantitative results on LHQ [86] for 20 steps, 50 steps, and 100 steps generation. To our knowledge, InfNat-0 [40] is the only prior work capable of long-range view synthesis without supervision from sequential data or accurate ground truth depth.

<table>
<thead>
<tr>
<th>Method</th>
<th>first 20 steps</th>
<th>first 50 steps</th>
<th>full 100 steps</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FID</td>
<td>KID</td>
<td>IS</td>
</tr>
<tr>
<td>InfNat-0 [40]</td>
<td>39.45</td>
<td>0.12</td>
<td>2.80</td>
</tr>
<tr>
<td>DiffDreamer</td>
<td>34.49</td>
<td>0.08</td>
<td>2.82</td>
</tr>
</tbody>
</table>

Table 2: Quantitative results on ACID [42] for 20 steps, 50 steps, and 100 steps generation. Note that all prior works require posed multi-view sequences for training, while our DiffDreamer is trained from single image collections.

<table>
<thead>
<tr>
<th>Method</th>
<th>first 20 steps</th>
<th>first 50 steps</th>
<th>full 100 steps</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FID</td>
<td>KID</td>
<td>IS</td>
</tr>
<tr>
<td>SynSin [97]</td>
<td>79.58</td>
<td>0.63</td>
<td>1.90</td>
</tr>
<tr>
<td>PixelSynth [68]</td>
<td>89.63</td>
<td>1.10</td>
<td>1.23</td>
</tr>
<tr>
<td>3D Photos [81]</td>
<td>99.79</td>
<td>0.80</td>
<td>1.65</td>
</tr>
<tr>
<td>InfNat [42]</td>
<td>59.93</td>
<td>0.22</td>
<td>2.36</td>
</tr>
<tr>
<td>DiffDreamer</td>
<td>52.81</td>
<td>0.12</td>
<td>2.69</td>
</tr>
</tbody>
</table>
Quantitative results  We show the quantitative evaluation on LHQ \cite{86} in Tab. 1 and ACID \cite{42} in Tab. 2. We observe that our 20-step generation outperforms prior works on all metrics by a relatively large margin. Our 50-step generation also has a significant advantage over prior works except for InfNat \cite{42} and InfNat-0 \cite{40}, which are on par with our model. Our DiffDreamer’s 100-step generation is not as good as \cite{42} and \cite{40} on FID \cite{27} and KID \cite{6} while achieving higher IS \cite{73}. However, we achieve significantly better 3D consistency metrics, as shown in Tab. 3 and Tab. 4. Our model performs scene extrapolation based on the presented content from the input image very well, but we do not enforce it to generate diverse content. Therefore, the model may output blander frames after it goes significantly beyond the input. We also notice that our better 3D consistency makes autocruise \cite{42} fail and hit the ground/hills.

**Table 3**: 3D consistency scoring \cite{96}, where we train disparity supervised DVGOv2 \cite{88} using 10 sequences generated by the models and report the mean±std novel view synthesis metrics.

<table>
<thead>
<tr>
<th>Method</th>
<th>PSNR↑</th>
<th>SSIM↑</th>
<th>LPIPS↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>InfNat \cite{42}</td>
<td>19.94±1.63</td>
<td>0.55±0.07</td>
<td>0.18±0.04</td>
</tr>
<tr>
<td>InfNat-0 \cite{40}</td>
<td>18.92±1.42</td>
<td>0.41±0.08</td>
<td>0.20±0.02</td>
</tr>
<tr>
<td>DiffDreamer</td>
<td><strong>23.56±3.30</strong></td>
<td><strong>0.68±0.04</strong></td>
<td><strong>0.12±0.02</strong></td>
</tr>
</tbody>
</table>

**Table 4**: Number of reconstructed points via COLMAP \cite{75,76}, where we run COLMAP on 10 generated sequences, count the number of reconstructed points and report mean±std.

<table>
<thead>
<tr>
<th>Method</th>
<th>Avg. points reconstructed</th>
</tr>
</thead>
<tbody>
<tr>
<td>InfNat \cite{42}</td>
<td>1476±477</td>
</tr>
<tr>
<td>InfNat-0 \cite{40}</td>
<td>612±104</td>
</tr>
<tr>
<td>DiffDreamer</td>
<td><strong>3124±622</strong></td>
</tr>
</tbody>
</table>
Figure 6: **Detail preservation** of over 30 steps. DiffDreamer can preserve details upon long-range extrapolation.

Figure 7: **Ablation study**, where we disabled our key building blocks. We observe clear artifacts (naive, no lookahead) or inconsistency (no anchored) in all ablations.

More often, constituting a large portion of the failed scenes. This is due to our model preserving geometry cues—it does not refine regions to be further away from their actual positions. Note that though InfNat [42] and InfNat-0 [40] can synthesize sequences with hundreds of frames, they resemble complete trade-offs with consistency. We argue that without accurate geometry preservation, scene extrapolation models [42, 40] will converge towards random latent space walk using pretrained GANs.

### 4.2. Ablation studies

We perform a thorough ablation study to verify our design choices, shown in Fig. 7. The setups are: 1) Naive auto-regressive: we first set up a baseline by performing the simplest per-step generation. The naive method fails after only a few refinement steps. This is due to the input domain drifting as observed in prior works [42, 40]. 2) Without anchored conditioning: next, we disable anchored conditioning: we observe more severe 3D inconsistency while moving forward, as the conditioning signal is purely from the past frame. 3) Without lookahead conditioning: we proceed with removing the lookahead mechanism. We observe significant domain drifting and artifacts as the model no longer takes advantage of the easier flying-out task. We supply the corresponding quantitative ablations in Appendix Sec. A.

### 5. Discussion

In this paper, we introduced DiffDreamer, a novel unsupervised pipeline based on conditional diffusion models for scene extrapolation. DiffDreamer can conduct scene extrapolation capable of “flying” into the image while training only from internet-collected single images. The key idea of DiffDreamer is to utilize a conditional diffusion model to simultaneously inpaint and refine a corrupted image obtained by warping a previous image. This is accomplished by training a conditional diffusion model that is capable of performing image-to-image translation under various corruption augmentations and utilizing stochastic conditioning to refine a corrupted image given multiple conditioning. Our model demonstrated comparable generation quality with GAN-based methods while maintaining significantly better consistency than prior works.

**Limitation and future work**  
DiffDreamer cannot synthesize novel views in real time due to the heavy inference of diffusion models. However, speeding up diffusion models’ inference has been a very active area, and we expect advances in this area to speed up our approach directly. In addition, we do not enforce the diversity of content while going significantly beyond the input image, causing degradation for a true perpetual generation. We believe adding CLIP [60] conditioning to be a very exciting extension.

**Conclusion**  
Diffusion models are emerging as state-of-the-art generative 2D methods. DiffDreamer is the first approach to apply them to 3D scene extrapolation, demonstrating a high amount of view consistency that is crucial for many downstream tasks.
Acknowledgement We thank Zhengqi Li for providing all the technical details and data samples related to InfNat-0 [40]. Gordon Wetzstein was supported by Samsung, Stanford HAI, and a PECASE from the ARO.

References

[22] Andrew Fitzgibbon, Yonatan Wexler, and Andrew Zisserman. Image-based rendering using image-based priors. IJCV, 2005. 2
[40] Zhengqi Li, Qianqian Wang, Noah Snavely, and Angjoo Kanazawa. Infinitenature-zero: Learning perpetual view generation of natural scenes from single images. In ECCV, 2022. 1, 2, 3, 4, 5, 6, 7, 8, 9, 13, 14
[50] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. SIGGRAPH, 2022. 2
[52] Simon Niklaus, Long Mai, Jimie Yang, and Feng Liu. 3d ken burns effect from a single image. TOG, 2019. 2
[58] Eric Penner and Li Zhang. Soft 3d reconstruction for view synthesis. TOG, 2017. 2


[82] Daeyun Shin, Zhile Ren, Erik B Sudderth, and Charless C Fowlkes. 3d scene reconstruction with multi-layer depth and epipolar transformers. In CVPR, 2019. 2


[86] Ivan Skorokhodov, Grigoriі Sotnikov, and Mohamed Elhoseiny. Aligning latent and image spaces to connect the unconnectable. arXiv, 2021. 6, 7, 13, 14


2149
[101] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelNeRF: Neural radiance fields from one or few images. In CVPR, 2021. 2