Invisible Stitch: Generating Smooth 3D Scenes with Depth Inpainting

Paul Engstler Andrea Vedaldi Iro Laina Christian Rupprecht
University of Oxford
{paule,vedaldi,iro,chrisr}@robots.ox.ac.uk

Abstract

3D scene generation has quickly become a challenging new research direction, fueled by consistent improvements of 2D generative diffusion models. Current methods generate scenes by iteratively stitching newly generated images with existing geometry, using pre-trained monocular depth estimators to lift the generated images to 3D. The predicted depth is fused with the existing scene representation through various alignment operations. In this work, we make two fundamental contributions to the field of 3D scene generation. First, we note that lifting images to 3D with a monocular depth estimation model is suboptimal as it ignores the geometry of the existing scene, thus prompting the need for alignment. We introduce a depth completion model to directly learn the 3D fusion process, resulting in improved geometric coherence of generated scenes. Second, we introduce a new benchmark to evaluate the geometric accuracy of scene generation methods. We show that the commonly used CLIP score between scene prompts and images is unsuitable for measuring the geometric quality of a scene and introduce a depth-based metric. Our benchmark thus offers an additional dimension to gauge the quality of generated scenes.

1. Introduction

The advent of high-quality image generative models [19, 49, 50] has paved the way for several exciting computer vision applications. One notable example is novel-view synthesis, which has been significantly transformed by leveraging the visual priors learned by large-scale generative models. This progress has led to a new emerging direction: 3D scene generation. The goal here is to generate not just a new image or view but an entire 3D scene, starting from a single input image or text description.

Despite the 3D nature of this problem, existing work has primarily focused on the visual quality of the generated scenes and, in particular, their semantic alignment with the input text description, as evaluated by image-text models like CLIP [18, 43]. This assessment, however, overlooks the *structural* quality of the generated scenes.

Current approaches to 3D scene generation employ an iterative process, alternating between geometry estimation, moving the camera, and inpainting previously unseen regions using an image-conditional generative model until the entire scene is generated [11, 16, 20, 70, 71]. Most methods rely on general-purpose *monocular* depth estimation models to estimate the geometry of each frame. However, this approach leads to inconsistencies because these models infer depth from a single RGB image and do not consider the geometry of the already-generated scene. As a result, most methods resort to a post-hoc fusion process, such as global scale-and-shift optimization [39] or intricate pipelines to align the predicted depth with the existing scene [70]. These solutions only partially mitigate the issue, resulting in depth seams and inconsistencies. This issue has been underexplored because the currently used metrics for this task mainly evaluate the semantic and visual quality of the scene.

In this paper, we address the problem of geometric accuracy in 3D scene generation with two key contributions. First, at the system level, we introduce **DINe** (**D**epth **I**npainting **Ne**twork), a straightforward depth inpainting approach that leverages existing geometry, significantly simplifying existing frameworks. Second, we propose a new benchmark for scene generation, **SSG-3D** (**S**tructural **3D** **S**cene **G**eneration Benchmark), that decouples the image and depth generation components, making it suitable for assessing the structural quality of 3D scenes.

The goal of DINe is to leverage partial depth information during the scene generation process and directly inpaint the missing depth values. This approach leads to smooth, seamless transitions between existing geometry and newly predicted depth and obviates the need for post-hoc depth refinement or other bells and whistles. Specifically, DINe takes as input an image and an incomplete, sparse depth map, which are projections from a 3D point cloud of a scene given a novel (*i.e.*, previously unseen) viewpoint. The model predicts a complete, dense depth map for the whole image, consistent with the input geometry. DINe is trained in a self-supervised manner by fine-tuning an exist-

ing depth prediction model conditioned on partially masked depth maps. This can be done on a simple image dataset using pseudo-ground-truth depth maps, simply mimicking the typical missing depth patterns in iterative scene generation without relying on camera poses or other annotations.

We use our proposed benchmark, SSG-3D, to evaluate the geometric quality of our depth inpainting model compared to alternative geometry estimation approaches (*e.g.*, depth prediction and fusion) used in prior work. SSG-3D is designed using real and synthetic scene datasets and aims to decouple visual quality from geometric quality in scene generation. The idea is to use ground truth images and depth and evaluate the geometry of generated scenes based on depth maps. Given a view of a scene, a method is tasked with extending it based on a given novel viewpoint, for which a ground-truth depth map exists. The generated scene geometry can then be easily evaluated.

Comprehensive evaluation on SSG-3D suggests that depth prediction and fusion approaches of existing scene generation methods yield geometric inconsistencies. In contrast, our DINe, trained to retain geometric consistency across frames, drastically reduces these artefacts without compromising visual quality.

2. Related Work

3D Scene Generation. Text-to-3D or image-to-3D scene generation has seen tremendous progress in recent years, where the majority of works in this field can either be categorized as object-centric [14, 23, 30, 32, 33, 41, 42, 44, 53, 54, 56, 59, 63, 69], *i.e.*, focusing on objects without background, or holistic, generating a single 3D scene or 3D trajectories with a background.

Earlier methods in object-centric generation focus only on novel view synthesis, not considering the scene's geometry. They are often based on layer-structured representations [28, 34, 55, 60], *e.g.*, layered depth images, or more implicit ones, such as in SynSin [65]. More modern approaches [32, 33, 41, 51, 63] typically distill 2D image generation priors from models like Stable Diffusion [49] into a 3D representation, such as a NeRF [35] or 3D Gaussians [26]. Other works directly learn a 3D representation from 2D images [6, 7, 17, 37, 38, 58].

More holistic methods generate entire scenes beyond a single object. These methods generally build a scene in a sequential manner using supervision from 2D image generation models. PixelSynth [48] is the first in this line of works, learning individual depth prediction, outpainting, and refinement components from scratch, which are then queried iteratively to build a scene. Text2Room [20] leverages an image inpainting model and the depth inpainting model IronDepth[1], both pre-trained, to build a scene mesh. A fusion process is applied to attach new frames to the existing mesh. LucidDreamer [11] operates similarly, but utilizes ZoeDepth [3] instead and generates a pointcloud. Its fusion process includes depth alignment and extrapolation at the seams to eliminate discontinuities between generated frames. WonderJourney [70] follows this framework setup, relying on MiDaS [46] and a more intricate depth fusion stage, which includes depth alignment, grouping objects at similar disparity to planes, and sky depth refinement, to generate scene "journeys". Text2NeRF [71] builds a NeRF representation of a scene, utilizing LeReS [68], aligning depth and optimizing it with a separately trained refinement network. Further methods like Infinite Nature [31], SceneScape [16], and Text2Immersion [39] have similar designs and also rely on an off-the-shelf general-purpose depth estimation model to project the hallucinated 2D scene extensions into a 3D representation. PeRF [61] starts from an existing panorama as opposed to iteratively building a scene. The Denoising Diffusion Vision Model [52] has been proposed as a joint RGB and depth prediction network for scene generation. An in-depth discussion and evaluation of this idea, however, has been left for future work. Other approaches such as GAUDI [2], ZeroNVS [51], DiffDreamer [5], and InfiniteNature-Zero [29] learn implicit representations. LDM3D [57], RGBD^2 [27], and Xiang *et al*. [67] train a model for simultaneous image and depth prediction. More specialized methods introduce different representations, such as BlockFusion [66], Worldsheet [21], and Set-the-Scene [12].

Depth Completion and Inpainting. With the emergence of depth-sensing technologies, inferring a dense depth map of a 3D scene from a sparse depth representation and a given RGB image has gained significant importance. Works in this field seek to integrate cues from both modalities either in a 2D [9, 10, 40, 72] or 3D feature space [4, 8, 22, 24, 62] to produce a complete depth map.

These sparse *depth completion* methods are able to recover the depth of an entire scene from a possibly very sparse depth input but have not been designed to complete depth for regions without any depth information, which naturally occurs in a scene generation task.

For this task, *depth inpainting* approaches appear more suitable. IronDepth [1] propagates existing depth information between pixels in its iterative depth map refinement, allowing for more flexible completion. Wei et al. [64] finetuned NLSPN [40] to fill holes in depth maps after removing detected objects by a clutter segmentation network. Neither of these methods, however, is specifically designed to deal with the depth discontinuities encountered in iterative scene generation.

3. Method

In Sec. 3.1, We outline a general framework shared by current scene generation methods [11, 20, 39, 70, 71]. These

methods have three main components. An image-generative model predicts or completes new frames, a monocular depth predictor to estimate the geometry of those frames and a fusion component for merging them with the existing scene.

With DINe, we propose a depth inpainting model, which unifies depth prediction and fusion into a single neural network. This model can be conditioned on partial depth maps from an existing scene, enabling it to predict depth for new regions that seamlessly attach. In Sec. 3.2, we describe how these partial depth maps are obtained from a scene and how we generate training data and learn the model by outlining its training scheme.

3.1. Preliminaries: 3D Scene Generation

The task of 3D scene generation from a single image can be formulated as follows. Let $I_0 \in \mathbb{R}^{3\times\Omega}$ be the input image, where $\Omega = \{1,\dots,H\} \times \{1,\dots,W\}$ is a lattice representing pixels. Given an arbitrary viewpoint $V = [R|t] \in \mathbf{SE}(3)$ and an intrinsic camera matrix K, the task is to generate a new view of the scene $\hat{I}(V,K) \in \mathbb{R}^{3\times\Omega}$ that is consistent with the original images and any other views that have already been generated. Typically, scenes are generated iteratively, hallucinating unseen regions in new images $\hat{I}_i$, with $i = \{1,\dots,T\}$, and then attaching them to an existing scene representation.

Image generation. To achieve this consistency, the scene is parameterized in 3D space. In the following, we use a pointcloud but other representations may be used (such as a mesh [20] or a NeRF [71]). Let $\mathcal{P} = \{(C_j, X_j)\}_j$ be a cloud of points at 3D locations $X_j \in \mathbb{R}^3$ with color $C_j \in \mathbb{R}^3$. Generating a new view from a pointcloud can be done by projecting the 3D points to pixels into the image plane of the new view $x_j \equiv KV^{-1}X_j$ (since $x_j \in \mathbb{R}^2$, here $\equiv$ represents the mapping from homogeneous coordinates to image coordinates). However, this forward projection results in an *incomplete* image $\tilde{I}(x_i) = C_j$, leaving holes where the viewpoint captures previously unseen regions.

To complete the sparse projection $\tilde{I}$ one can leverage a large-scale generative model f (*e.g.*, Stable Diffusion [49]) which has learned a visual prior for a massive collection of visual data. In particular, we first obtain a binary mask $M = \{0,1\}^\Omega$ that indicates these holes in the image. We then use an inpainting variant of the Stable Diffusion model that has been trained to fill in missing regions in an image to obtain $\hat{I} = f(\tilde{I}, M)$.

Pointcloud generation. In 3D scene generation from a single image or text description, one cannot access a 3D pointcloud of the scene. Instead, the goal is to build the pointcloud iteratively by sampling new viewpoints V_i. By design, this process enforces consistency between each view of the scene. A natural mapping from images to pointclouds can be established via depth maps $D \in \mathbb{R}^{H\times W}$ as it allows the projection of the image pixels into the scene. Let $\mathcal{P}_i$ be the pointcloud at the i-th iteration. Each iteration expands the representation with new geometry as $\mathcal{P}_{i+1} = \mathcal{P}_i \cup \hat{\mathcal{P}}_{i+1}$.

$$\hat{\mathcal{P}}_i = \left\{ V_i K_i^{-1} \begin{bmatrix} u \\ v \\ D_i(u,v) \end{bmatrix} \right\}_{(u,v)\in\Omega} . \tag{1}$$

A new viewpoint V_i and camera matrix K_i are chosen at each iteration to expand the scene. The current image $\hat{I}_i$ is passed to a depth estimation network to obtain the corresponding depth map D_i.

Most existing work uses off-the-shelf depth prediction models that are not conditional on existing geometry. As a result, an additional fusion component is necessary to merge the new prediction with the scene. This may include heuristics such as depth interpolation and global alignment operations. These depth prediction and fusion components are thus susceptible to producing strong artifacts.

3.2. DINe: Depth Inpainting for Scene Generation

To condition the depth estimation on existing geometry, we propose to obtain D_i by *inpainting* a partial depth map. We define the depth inpainting model as $D_i = g(\hat{I}_i, M_i, \tilde{D}_i)$. At the i-th iteration of the scene generation process, the model takes as input the inpainted image $\hat{I}_i$, a mask M_i, which signifies which pixels are newly generated and thus have no existing depth estimate, and the corresponding partial depth map $\tilde{D}_i$.

Partial depth map generation. The incomplete depth map $\tilde{D}_i$ is obtained by projecting the pointcloud $\mathcal{P}_{i-1}$ into the current view (and thus contains holes). Note that since the depth map $\tilde{D}_i$ is obtained by projecting the existing point cloud $\mathcal{P}_{i-1}$ into the current view, only the holes indicated by M_i that get filled in by g contribute new points to $\mathcal{P}_i$. The other points already exist in the scene.

Our goal is to learn a model $g(I, M, \tilde{D})$ for scene generation, which provides a robust depth estimate given an image $\hat{I}$, a depth-mask M, and a partial depth map $\tilde{D}$.

Training dataset. Naturally, DINe can be trained with supervision, given a multi-view dataset with known camera poses and depth ground truth. However, these datasets are often comparatively small and usually do not provide dense ground truth depth for the entire image, leading to a weak training signal. To circumvent these issues, we utilize an off-the-shelf general-purpose monocular depth prediction model $g_T(I)$ that predicts unconditional, dense depth from a single RGB image. We thus train g in a self-supervised

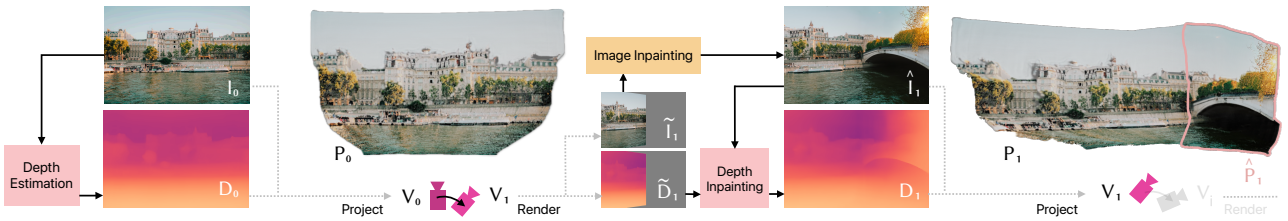

Figure 1. **Overview of a simple 3D scene generation method.** Starting from an input image I_0, we project it to a point cloud based on a depth map predicted by a depth estimation network. To extend the scene, we render it from a new viewpoint and query a generative model to hallucinate beyond the scene's boundary. Now, we condition the depth estimation network on the depth of the existing scene and the image of the scene extended by the image inpainting network to produce a geometrically consistent depth map to project the hallucinated points. This process may be repeated until a 360° scene has been generated.

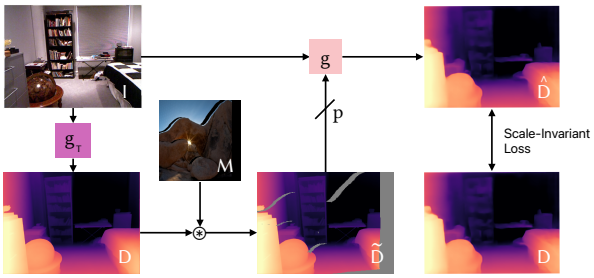

Figure 2. **Overview of our training procedure.** In this compact training scheme, a depth inpainting network g is learned by jointly training depth inpainting and depth prediction without a sparse depth input (the ratio is determined by the task probability p). A teacher network g_T is utilized to generate a pseudo-ground-truth depth map D for a given image I. This depth map is then masked with a random mask M, to obtain a sparse depth input $\tilde{D}$.

fashion using the predictions from $g_T(I)$, which takes on the role of the teacher model in a student-teacher training scheme.

Given a dataset of only images I_k, we generate a pseudo-labelled training dataset for g as follows. For each image in the dataset, we obtain a target depth map from a teacher g_T, $D_k = g_T(I_k)$. Then, for each depth map, and similar to the scene generation step, we sample one or more random viewpoints V_l and camera matrix K_l, and we warp the depth map D_k to the new viewpoint obtaining a mask $M_{k,l}$ and reprojected depth $D_{k,l}$. We collect all masks generated this way in a set $\mathcal{M} = \{M_{k,l}\}_{k,l}$ that represents the typical occlusion patterns generated by viewpoint changes.

Training scheme. Given the lack of multi-view data, during training, we sample a random mask from $M_n \in \mathcal{M}$ $(1 \leq n \leq |\mathcal{M}|)$ for each image I_k. And train g to reconstruct the pseudo depth D_k guided by the scale-invariant loss [15], where $\tilde{d} = g(I_k, M_n, D_k \odot M_n)$, $d = D_k$, and

$$\psi_i = \log \tilde{d}_i - \log d_i.$$

$$\mathcal{L}_{\text{depth}} = \sqrt{\frac{1}{T}\sum_i \psi_i^2 - \frac{\lambda}{T^2}(\sum_i \psi_i)^2} \qquad (2)$$

T is the number of pixels in D_k with valid ground-truth values. This scheme allows learning g only from pseudo-supervision. A benefit of this formulation is that g can be initialized with a depth estimation model itself, effectively fine-tuning it for depth inpainting. Moreover, we can then retain its original depth *prediction* (instead of depth *inpainting*) capabilities by choosing $M_n = 0$ with probability p, effectively masking all input depth, and recovering the depth prediction task. Finally, we can choose g_T as a large model while g can be a more lightweight architecture, which improves g via distillation.

4. Scene Geometry Evaluation Benchmark

Within the fully generative task of scene generation, evaluating the geometric properties of generated scenes is difficult due to the lack of ground-truth data. As a result, most existing work resorts to image-text similarity scores, such as the CLIP score [18], which only measures the global semantic alignment of the generation with a text description.

In Figure 3, we show that this metric does not reflect the geometric consistency and quality of the depth predictions used to build the scene.

Consequently, the CLIP score does not evaluate the depth prediction and fusion components of scene generation methods. Therefore, we propose a new evaluation benchmark that quantifies the depth extrapolation ability of these components in isolation, using a controlled environment consistent across methods and entirely independent of image extrapolation. In this benchmark, we seek to measure the deviation between the ground truth and the depth continuation on a partial scene.

Original: 27.31 / Ours: 27.36 | Original: 27.39 / Ours: 27.23 | Original: 23.39 / Ours: 23.22

Figure 3. **Qualitative comparison between LucidDreamer [11] (top) and ours (bottom).** Despite high CLIP scores, the original depth prediction and fusion component in LucidDreamer, based on ZoeDepth [3], yields scenes with distorted geometry. Our model leads to less torn structures and provides an overall geometrically more sound scene. The CLIP score does not reflect the changes in geometric quality.

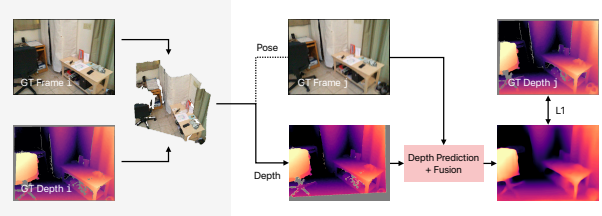

Figure 4. **Scene Geometry Evaluation overview.** For a given view pair, we use the ground-truth point cloud (*i.e.* from a depth map) of the first frame. Then, we render the representation from the second viewpoint. We feed the corresponding ground-truth image and the projected, sparse depth into a depth prediction and fusion component to extrapolate the missing depth. We calculate the mean absolute depth error only for extrapolated regions.

4.1. Approach

Starting from a scene representation constructed from the ground-truth information of one view, we seek to extrapolate the depth for another ground-truth view that overlaps the first one. As the depth is known for the second view, we compute the error between the generated depth and the ground-truth depth. We only consider the error in regions that were extrapolated. A detailed description of this approach is provided in Figure 4.

We use a point cloud as our representation of choice and base the overlap of two views $\phi(v_i, v_j)$ on the number of pixels that show part of the v_i scene from the viewpoint of v_j. Put differently, if a rendering pipeline renders an image with dimensions $H \times W$ and assigns a default value x for a pixel p that does not represent any parts of a scene, we define ϕ as:

$$\phi(v_i, v_j) = \frac{\sum_{i,j}^{H \times W} \mathbb{1}_{p(i,j) \neq x}}{H \times W} \tag{3}$$

4.2. Datasets

In our evaluation, we consider ScanNet [13] and Hypersim [47] as they provide images, dense depth, and camera poses to reconstruct scenes accurately. As the former is a real-world dataset featuring indoor scenes and the latter is a photorealistic one, they lie within the distributions of most depth estimation models.

For both datasets, we report the average absolute error on the extrapolated region across all pairs of views across all scenes.

5. Experiments

Having described the training scheme of DINe in detail, we now provide details about its implementation. First, in Section 5.2, we investigate the depth consistency of DINe as well as other depth prediction and fusion components with our SSG-3D benchmark. In Section 5.3.1, we embed DINe into existing scene generation methods, testing if the improved depth consistency of DINe leads to CLIP score improvements. This allows us to shed light on the CLIP score's limited descriptive power for the scene generation task. Finally, in Section 5.3, we show that even in a minimal scene generation pipeline, DINe is able to create scenes that are competitive with existing state-of-the-art methods.

5.1. Implementation Details

We fine-tune a pre-trained ZoeDepth model to obtain our depth completion model g, re-initializing its patch embedding layer to receive two additional channels apart from the image input. These channels provide the sparse depth input $\tilde{D}$ as well as a mask M describing the presence of sparse depth, i.e., $\tilde{D} > 0$. While we only replace this layer, we keep the entire model unfrozen to ensure the additional information can be integrated in later layers. We set $\lambda = 0.85$ in the scale-invariant loss.

We train on images from the NYU Depth v2 [36] dataset, using the monocular depth estimation network Marigold [25] as a teacher g_T to distill its prediction capabilities into our depth estimation network g.

To construct the set of warped masks $\mathcal{M}$, which contains typical masking patterns seen with view point changes, we consider the Places365 [73] dataset, generating one mask from each image. Images are projected into a three-dimensional space based on depth predicted by Marigold. We then define a look-at camera with random elevation and azimuth values (between $[0, 15]$ degrees) to render the image, where the pixel occupancy yields the mask. Masks are randomly chosen to be applied to a training sample depth D. Ensuring we retain the original depth prediction task, we set the probability to zero out the sparse depth input to 50%.

5.2. Evaluating Scene Geometry with SSG-3D

To validate the capability of our model and compare it with the depth prediction and fusion component of existing scene generation methods to faithfully extend scenes, we turn to our scene geometry evaluation benchmark (see Section 4). For comparison, we add further state-of-the-art depth prediction methods.

We turn to ZoeDepth with global scale-and-shift depth alignment (ZoeDepth†) to represent WonderJourney [70], removing the SAM-based plane construction, which would reduce geometric accuracy. For LucidDreamer [11], we utilize the same setup but add its depth extrapolation step at the seams to eliminate depth discontinuities between frames (ZoeDepth+LD†). Text2Room [20] directly utilizes IronDepth [1] for depth completion. To add more reference points, we also include CostDCNet [24], NLSPN [40], and DPT [45].

From the results in Table 1, we find that in both, a real-world and a photorealistic setting, our inpainting model produces predictions that are more faithful to the ground-truth than the other methods.

Interestingly, IronDepth, a depth completion method, and ZoeDepth†, a depth estimation method with alignment to the existing scene, appear to perform quite similarly, possibly explaining why either are used by state-of-the-art methods.

Component	DC/MDE	ScanNet	Hypersim
CostDCNet [24]	DC	0.5854	4.0149
NLSPN [40]	DC	0.1826	3.3503
IronDepth [1]	DC	0.1085	0.8241
DPT† [45]	MDE	0.1719	1.8824
ZoeDepth [3]	MDE	0.1924	1.1964
ZoeDepth†	MDE	0.1293	0.7872
ZoeDepth+LD† [11]	MDE	0.1604	0.8057
Ours	DC	**0.0816**	**0.7295**

Table 1. **SSG-3D benchmark results.** Methods are categorized as depth completion (DC) or monocular depth estimation (MDE) approaches. † indicates that the predicted depth is aligned with the existing scene through a global scale-and-shift optimization.

We generally observe a higher error for Hypersim as it features notably more fine details than ScanNet that cannot be recovered by models operating at a lower resolution than its image size, such as ZoeDepth. Second, unlike real-world depth sensors, the depth in Hypersim is exact with sharp boundaries, which makes it more difficult for models trained on real-world data.

5.3. Scene Generation Results

Based on the finding that DINe achieves better geometric consistency when extending scenes, we investigate if our depth model also leads to better CLIP scores, when used as a drop-in replacement for the depth prediction and fusion component in existing methods. Then, we embed DINe into a minimal scene generation pipeline that does not have any fusion steps nor applies further refinement to boost the visual quality. Here, we show that DINe learned both, depth prediction and fusion, as required for inpainting, and is sufficient to achieve state-of-the-art results.

5.3.1 Drop-In Replacement

Seeing the geometric improvements of DINe, we now investigate if they translate into improvements in the CLIP score, too, which is commonly used as the evaluation metric of choice for scene generation methods. By swapping out the depth prediction and fusion component of an existing method, we can isolate its effect and thus measure its individual contribution to the CLIP score. If the CLIP score captures scene geometry, we expect it to improve when a better component is used.

Using DINe as the replacement, which shows superior depth extrapolation performance, we find that its use does not degrade the quality of generated scenes for multiple existing methods (see Table 2). Notably, though, DINe does not seem to generally improve the CLIP score of generated

Scene	kyoto	nc	prague	indoor0	indoor1	indoor2	indoor3
WonderJourney [70]	26.60 ± 1.15	24.33 ± 0.78	24.34 ± 0.78	21.62 ± 1.06	21.91 ± 0.75	24.07 ± 0.74	21.95 ± 1.06
w/ ours	26.60 ± 1.15	24.34 ± 0.79	24.34 ± 0.79	21.63 ± 1.06	21.91 ± 0.73	24.06 ± 0.05	21.93 ± 1.05
LucidDreamer [11]	25.69 ± 0.63	22.19 ± 0.64	26.72 ± 0.38	23.22 ± 0.44	20.62 ± 0.40	24.24 ± 0.46	22.76 ± 0.45
w/ ours	25.70 ± 0.64	22.18 ± 0.64	26.72 ± 0.38	23.23 ± 0.45	20.62 ± 0.40	24.26 ± 0.45	22.76 ± 0.46
Text2Room [20]	27.58 ± 0.41	21.56 ± 0.92	26.75 ± 0.33	21.84 ± 0.63	21.63 ± 0.58	23.07 ± 0.62	22.12 ± 0.52
w/ ours	29.40 ± 0.58	22.86 ± 0.75	26.76 ± 0.33	21.97 ± 0.73	21.64 ± 0.54	23.00 ± 0.53	22.05 ± 0.55

Table 2. **CLIP score when using DINe in other scene-generation systems.** Both versions receive the same input image and text prompt (see supplementary materials for details), with a fixed seed, to generate a single new frame, which is then evaluated. We consider 100 samples per scene, reporting the mean and standard deviation. There is no statistically significant difference (validated with t-test) in semantic similarity, while our method does improve the geometric quality of the scenes (Tab. 1).

scenes either. This shows that the CLIP score is insufficiently sensitive to geometric (in)accuracy.

5.3.2 Minimal Pipeline

Going a step further, we show that a simple pipeline with DINe is competitive with existing methods to generate scenes with high semantic similarity to an appropriate text prompt, *i.e.*, a high CLIP score. Unlike other methods, it does not employ any depth refinement steps (*e.g.*, SAM-based grouping of objects with similar disparity and creating planes in WonderJourney [70], or depth extrapolation at frame seams in LucidDreamer [11]). Crucially, we want to demonstrate that none of these steps are required to generate scenes of similar quality. This pipeline relies on few, foundational components to generate a scene from a single image: First, in the same vein as current 3D scene generation methods, we enlist the help of a Stable Diffusion inpainting model (f) to hallucinate how a scene looks like beyond its boundaries. Second, we use our depth inpainting model (g) to produce an initial depth estimation for the original image and inpaint the depth in subsequent steps to attach the extrapolations.

Generating the Point Cloud Starting from a given single image, we obtain a depth estimation to project it to a point cloud. We use a stationary perspective camera with fixed intrinsics. With each step, we rotate the camera slightly further along its azimuth to obtain a view that provides a canvas for the Stable Diffusion model to inpaint while still partially including the existing scene.

When inpainting images with Stable Diffusion, distortion artifacts have been known to appear in those regions that are not supposed to be edited, which has been attributed to its variational autoencoder [74]. These alterations then cause a mismatch between the input image and the sparse depth input, which is difficult to resolve. To minimize these effects, we utilize an asymmetric autoencoder [74] that emphasizes the decoder, making it heavier than the encoder and providing it with additional information about the inpainting task. We find that this autoencoder leads to a significant decrease in the prevalence of these artifacts.

Once the expanded scene has been visually hallucinated by Stable Diffusion, we pass the image onto our depth inpainting model with the depth of the existing scene. We project all hallucinated pixels based on the depth prediction, which seamlessly connect to the point cloud without the need for any alignment steps. We observe that depth predictions might have a gradient instead of a hard boundary at object edges, which leads to floaters radiating around objects in the point cloud. To minimize their occurrence, we identify regions in the predicted depth map with a high gradient, mask them, and assign them new values based on their nearest neighbors. This *snaps* pixels in these gradient regions either to the object or its surrounding, creating a hard boundary.

We repeat this process until the loop is closed, yielding a 360° scene. We make sure that the final hallucination step has a wide canvas to connect both ends of the loop, assuming there has been a slight domain shift between the original image and the cascade of hallucinated views.

Qualitative Results. Using this minimal pipeline, we can generate 360° scenes given a single input image as well as a text prompt. In Figure 5, we show three example scenes generated by our approach that feature complex geometry. Our depth inpainting model is able to seamlessly extend scenes with believable geometry, creating an immersive experience. We generate these results by rotating the camera 25 degrees along its azimuth with each step, slightly tapering it towards the end to close the loop.

Visual Quality. To compare the visual quality of the scenes from our pipeline with other scene-generation methods, we utilize the CLIP score. We use the same input images and prompts shown in Figure 5 to produce similar

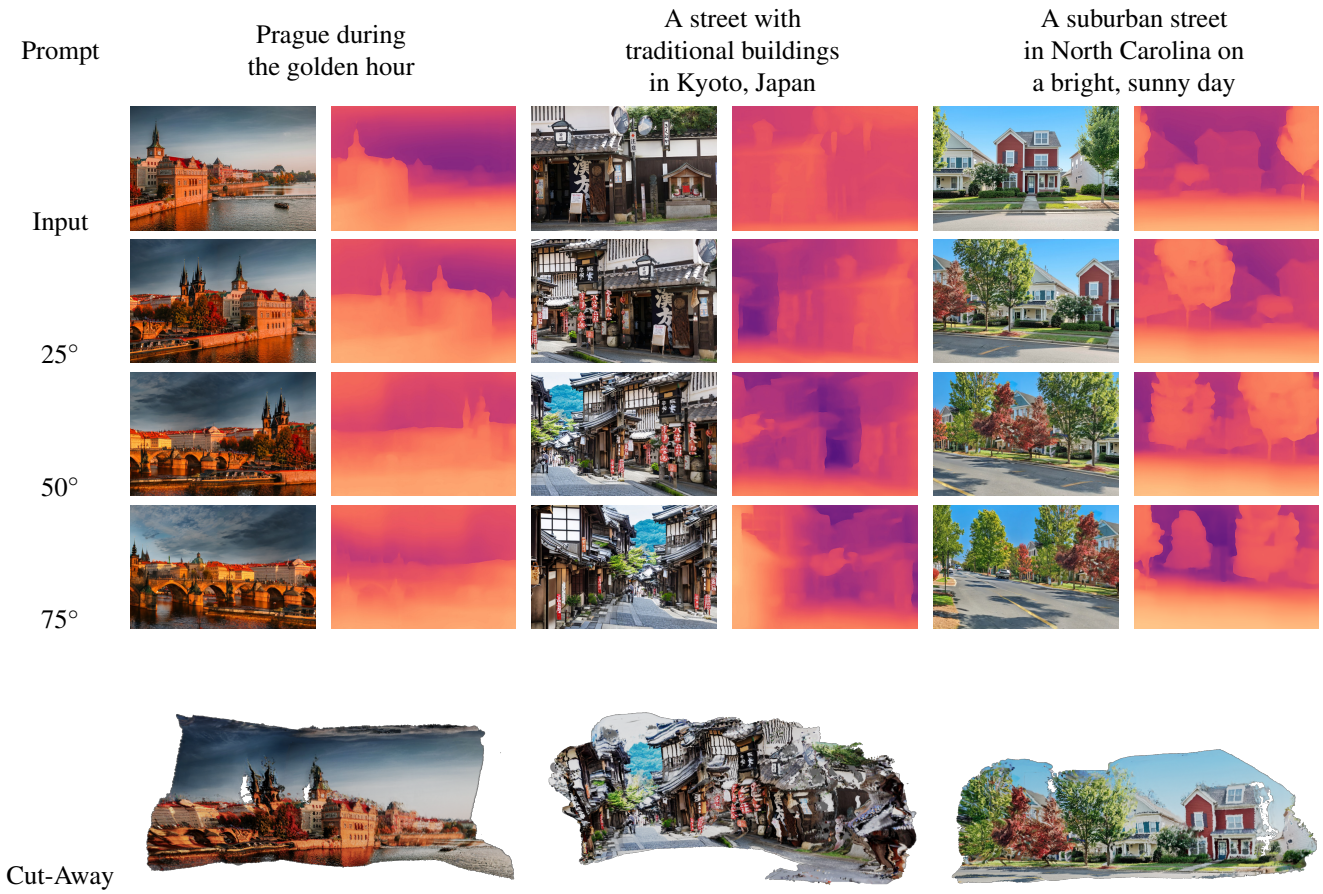

Figure 5. **Qualitative results of our method on real-world images.** We show the first few hallucinated views and the corresponding depth maps of 360° scenes. We also provide a cut-away view of the generated pointcloud.

Method	CLIP Score [18]		
	Prague	Kyoto	N. Carolina
PixelSynth [48]	12.75	12.73	14.52
Text2NeRF [71]	23.83	28.04	21.53
Text2Room [20]	26.33	28.24	24.52
LucidDreamer [11]	26.01	29.56	24.96
WonderJourney [70]	27.78	26.67	26.47
Ours	26.87	27.65	24.24

Table 3. **Quantitative results for the visual quality of generated scenes.** All methods receive the same input image and text prompt (see Figure 5) to generate 360° scenes. We use all hallucinated views to compute the average CLIP score.

scenes with each method.

In Table 3, we observe that despite using a minimal pipeline without any depth or visual refinement steps, DINe is sufficient to generate scenes that are competitive with existing state-of-the-art methods.

6. Conclusion

This paper presents DINe, a depth inpainting model for scene generation, which can be used to generate immersive 360° scenes.

We show that a straight-forward pipeline based on DINe, without any bells and whistles, is sufficient for this task and is competitive with complex, state-of-the-art scene generation methods. In contrast to prior methods, our approach eliminates the need for post-hoc depth or visual refinement steps. We further find that the commonly utilized CLIP score captures the geometric quality of scenes insufficiently, prompting the need for a new benchmark. The SSG-3D benchmark introduced in this paper is a rigorous tool to measure the depth extrapolation performance of depth prediction and fusion components in scene generation methods, fully rooted in ground-truth data.

These contributions put geometry back into the limelight, highlighting its importance for scene generation.

Ethics. For further details on ethics, data protection, and copyright, please see https://www.robots.ox.ac.uk/~vedaldi/research/union/ethics.html.

Acknowledgements. P. E., A. V., and I. L. are supported by ERC-UNION- CoG-101001212. P.E. is also supported by Meta Research. I.L. also receives support from VisualAI EP/T028572/1.

References

[1] Bae, G., Budvytis, I., Cipolla, R.: Irondepth: Iterative refinement of single-view depth using surface normal and its uncertainty. In: British Machine Vision Conference (BMVC) (2022) 2, 6

[2] Bautista, M.A., Guo, P., Abnar, S., Talbott, W., Toshev, A., Chen, Z., Dinh, L., Zhai, S., Goh, H., Ulbricht, D., et al.: Gaudi: A neural architect for immersive 3d scene generation. Advances in Neural Information Processing Systems **35**, 25102–25116 (2022) 2

[3] Bhat, S.F., Birkl, R., Wofk, D., Wonka, P., Müller, M.: Zoedepth: Zero-shot transfer by combining relative and metric depth. arXiv preprint arXiv:2302.12288 (2023) 2, 5, 6

[4] Boulch, A., Puy, G., Marlet, R.: Fkaconv: Feature-kernel alignment for point cloud convolution. In: Proceedings of the Asian Conference on Computer Vision (2020) 2

[5] Cai, S., Chan, E.R., Peng, S., Shahbazi, M., Obukhov, A., Gool, L.V., Wetzstein, G.: DiffDreamer: Towards consistent unsupervised single-view scene extrapolation with conditional diffusion models. In: ICCV (2023) 2

[6] Chan, E.R., Lin, C.Z., Chan, M.A., Nagano, K., Pan, B., De Mello, S., Gallo, O., Guibas, L.J., Tremblay, J., Khamis, S., et al.: Efficient geometry-aware 3d generative adversarial networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16123–16133 (2022) 2

[7] Chan, E.R., Monteiro, M., Kellnhofer, P., Wu, J., Wetzstein, G.: pi-gan: Periodic implicit generative adversarial networks for 3d-aware image synthesis. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5799–5809 (2021) 2

[8] Chen, Y., Yang, B., Liang, M., Urtasun, R.: Learning joint 2d-3d representations for depth completion. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10023–10032 (2019) 2

[9] Cheng, X., Wang, P., Guan, C., Yang, R.: Cspn++: Learning context and resource aware convolutional spatial propagation networks for depth completion. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 10615–10622 (2020) 2

[10] Cheng, X., Wang, P., Yang, R.: Depth estimation via affinity learned with convolutional spatial propagation network. In: Proceedings of the European conference on computer vision (ECCV). pp. 103–119 (2018) 2

[11] Chung, J., Lee, S., Nam, H., Lee, J., Lee, K.M.: Luciddreamer: Domain-free generation of 3d gaussian splatting scenes. arXiv preprint arXiv:2311.13384 (2023) 1, 2, 5, 6, 7, 8

[12] Cohen-Bar, D., Richardson, E., Metzer, G., Giryes, R., Cohen-Or, D.: Set-the-scene: Global-local training for generating controllable nerf scenes (2023) 2

[13] Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5828–5839 (2017) 5

[14] Deng, K., Liu, A., Zhu, J.Y., Ramanan, D.: Depth-supervised nerf: Fewer views and faster training for free. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12882–12891 (2022) 2

[15] Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. Advances in neural information processing systems **27** (2014) 4

[16] Fridman, R., Abecasis, A., Kasten, Y., Dekel, T.: Scenescape: Text-driven consistent scene generation. Advances in Neural Information Processing Systems **36** (2024) 1, 2

[17] Gu, J., Liu, L., Wang, P., Theobalt, C.: Stylenerf: A style-based 3d-aware generator for high-resolution image synthesis. arXiv preprint arXiv:2110.08985 (2021) 2

[18] Hessel, J., Holtzman, A., Forbes, M., Bras, R.L., Choi, Y.: CLIPScore: a reference-free evaluation metric for image captioning. In: EMNLP (2021) 1, 4, 8

[19] Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020) 1

[20] Höllein, L., Cao, A., Owens, A., Johnson, J., Nießner, M.: Text2Room: Extracting textured 3D meshes from 2D text-to-image models. In: ICCV (2023) 1, 2, 3, 6, 7, 8

[21] Hu, R., Ravi, N., Berg, A.C., Pathak, D.: Worldsheet: Wrapping the world in a 3d sheet for view synthesis from a single image. In: ICCV (2021) 2

[22] Huynh, L., Nguyen, P., Matas, J., Rahtu, E., Heikkilä, J.: Boosting monocular depth estimation with lightweight 3d point fusion. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12767–12776 (2021) 2

[23] Jain, A., Tancik, M., Abbeel, P.: Putting nerf on a diet: Semantically consistent few-shot view synthesis. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5885–5894 (2021) 2

[24] Kam, J., Kim, J., Kim, S., Park, J., Lee, S.: Costdcnet: Cost volume based depth completion for a single rgb-d image. In: European Conference on Computer Vision. pp. 257–274. Springer (2022) 2, 6

[25] Ke, B., Obukhov, A., Huang, S., Metzger, N., Daudt, R.C., Schindler, K.: Repurposing diffusion-based image generators for monocular depth estimation. arXiv preprint arXiv:2312.02145 (2023) 6

[26] Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM Transactions on Graphics **42**(4) (2023) 2

[27] Lei, J., Tang, J., Jia, K.: RGBD2: generative scene synthesis via incremental view inpainting using RGBD diffusion models. In: CVPR (2023) 2

[28] Li, J., Feng, Z., She, Q., Ding, H., Wang, C., Lee, G.H.: MINE: towards continuous depth MPI with nerf for novel view synthesis. In: ICCV (2021) 2

[29] Li, Z., Wang, Q., Snavely, N., Kanazawa, A.: Infinitenature-zero: Learning perpetual view generation of natural scenes from single images. In: European Conference on Computer Vision. pp. 515–534. Springer (2022) 2

[30] Lin, C.H., Gao, J., Tang, L., Takikawa, T., Zeng, X., Huang, X., Kreis, K., Fidler, S., Liu, M.Y., Lin, T.Y.: Magic3d: High-resolution text-to-3d content creation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 300–309 (2023) 2

[31] Liu, A., Tucker, R., Jampani, V., Makadia, A., Snavely, N., Kanazawa, A.: Infinite nature: Perpetual view generation of natural scenes from a single image. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14458–14467 (2021) 2

[32] Liu, R., Wu, R., Van Hoorick, B., Tokmakov, P., Zakharov, S., Vondrick, C.: Zero-1-to-3: Zero-shot one image to 3d object. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9298–9309 (2023) 2

[33] Melas-Kyriazi, L., Laina, I., Rupprecht, C., Vedaldi, A.: Realfusion: 360deg reconstruction of any object from a single image. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8446–8455 (2023) 2

[34] Mildenhall, B., Srinivasan, P.P., Cayon, R.O., Kalantari, N.K., Ramamoorthi, R., Ng, R., Kar, A.: Local light field fusion: practical view synthesis with prescriptive sampling guidelines **38**(4) (2019) 2

[35] Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM **65**(1), 99–106 (2021) 2

[36] Nathan Silberman, Derek Hoiem, P.K., Fergus, R.: Indoor segmentation and support inference from rgbd images. In: ECCV (2012) 6

[37] Nguyen-Phuoc, T., Li, C., Theis, L., Richardt, C., Yang, Y.L.: Hologan: Unsupervised learning of 3d representations from natural images. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7588–7597 (2019) 2

[38] Niemeyer, M., Geiger, A.: Giraffe: Representing scenes as compositional generative neural feature fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11453–11464 (2021) 2

[39] Ouyang, H., Heal, K., Lombardi, S., Sun, T.: Text2immersion: Generative immersive scene with 3d gaussians. arXiv preprint arXiv:2312.09242 (2023) 1, 2

[40] Park, J., Joo, K., Hu, Z., Liu, C.K., So Kweon, I.: Non-local spatial propagation network for depth completion. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16. pp. 120–136. Springer (2020) 2, 6

[41] Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: Dreamfusion: Text-to-3d using 2d diffusion. arXiv preprint arXiv:2209.14988 (2022) 2

[42] Qian, G., Mai, J., Hamdi, A., Ren, J., Siarohin, A., Li, B., Lee, H.Y., Skorokhodov, I., Wonka, P., Tulyakov, S., et al.: Magic123: One image to high-quality 3d object generation using both 2d and 3d diffusion priors. arXiv preprint arXiv:2306.17843 (2023) 2

[43] Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021) 1

[44] Raj, A., Kaza, S., Poole, B., Niemeyer, M., Ruiz, N., Mildenhall, B., Zada, S., Aberman, K., Rubinstein, M., Barron, J., et al.: Dreambooth3d: Subject-driven text-to-3d generation. arXiv preprint arXiv:2303.13508 (2023) 2

[45] Ranftl, R., Bochkovskiy, A., Koltun, V.: Vision transformers for dense prediction. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 12179–12188 (2021) 6

[46] Ranftl, R., Lasinger, K., Hafner, D., Schindler, K., Koltun, V.: Towards robust monocular depth estima-

tion: Mixing datasets for zero-shot cross-dataset transfer. IEEE transactions on pattern analysis and machine intelligence **44**(3), 1623–1637 (2020) 2

[47] Roberts, M., Ramapuram, J., Ranjan, A., Kumar, A., Bautista, M.A., Paczan, N., Webb, R., Susskind, J.M.: Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10912–10922 (2021) 5

[48] Rockwell, C., Fouhey, D.F., Johnson, J.: PixelSynth: Generating a 3D-consistent experience from a single image. In: ICCV (2021) 2, 8

[49] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10684–10695 (2022) 1, 2, 3

[50] Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E.L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al.: Photorealistic text-to-image diffusion models with deep language understanding. Advances in neural information processing systems **35**, 36479–36494 (2022) 1

[51] Sargent, K., Li, Z., Shah, T., Herrmann, C., Yu, H.X., Zhang, Y., Chan, E.R., Lagun, D., Fei-Fei, L., Sun, D., et al.: Zeronvs: Zero-shot 360-degree view synthesis from a single real image. arXiv preprint arXiv:2310.17994 (2023) 2

[52] Saxena, S., Herrmann, C., Hur, J., Kar, A., Norouzi, M., Sun, D., Fleet, D.J.: The surprising effectiveness of diffusion models for optical flow and monocular depth estimation. arXiv preprint arXiv:2306.01923 (2023) 2

[53] Seitz, S.M., Dyer, C.R.: Photorealistic scene reconstruction by voxel coloring. In: CVPR (1997) 2

[54] Shi, Y., Wang, P., Ye, J., Long, M., Li, K., Yang, X.: Mvdream: Multi-view diffusion for 3d generation. arXiv preprint arXiv:2308.16512 (2023) 2

[55] Shih, M., Su, S., Kopf, J., Huang, J.: 3d photography using context-aware layered depth inpainting. In: CVPR (2020) 2

[56] Sitzmann, V., Thies, J., Heide, F., Nießner, M., Wetzstein, G., Zollhöfer, M.: Deepvoxels: Learning persistent 3d feature embeddings. In: CVPR (2019) 2

[57] Stan, G.B.M., Wofk, D., Fox, S., Redden, A., Saxton, W., Yu, J., Aflalo, E., Tseng, S.Y., Nonato, F., Muller, M., et al.: Ldm3d: Latent diffusion model for 3d. arXiv preprint arXiv:2305.10853 (2023) 2

[58] Szymanowicz, S., Rupprecht, C., Vedaldi, A.: Splatter image: Ultra-fast single-view 3d reconstruction. arXiv preprint arXiv:2312.13150 (2023) 2

[59] Trevithick, A., Yang, B.: GRF: learning a general radiance field for 3d representation and rendering. In: ICCV (2021) 2

[60] Tulsiani, S., Tucker, R., Snavely, N.: Layer-structured 3d scene inference via view synthesis. In: ECCV (2018) 2

[61] Wang, G., Wang, P., Chen, Z., Wang, W., Loy, C.C., Liu, Z.: Perf: Panoramic neural radiance field from a single panorama. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) (2024) 2

[62] Wang, S., Suo, S., Ma, W.C., Pokrovsky, A., Urtasun, R.: Deep parametric continuous convolutional neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2589–2597 (2018) 2

[63] Wang, Z., Lu, C., Wang, Y., Bao, F., Li, C., Su, H., Zhu, J.: Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. Advances in Neural Information Processing Systems **36** (2024) 2

[64] Wei, F., Funkhouser, T., Rusinkiewicz, S.: Clutter detection and removal in 3d scenes with view-consistent inpainting. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 18131–18141 (2023) 2

[65] Wiles, O., Gkioxari, G., Szeliski, R., Johnson, J.: Synsin: End-to-end view synthesis from a single image. In: CVPR (2020) 2

[66] Wu, Z., Li, Y., Yan, H., Shang, T., Sun, W., Wang, S., Cui, R., Liu, W., Sato, H., Li, H., et al.: Blockfusion: Expandable 3d scene generation using latent tri-plane extrapolation. ACM Transactions on Graphics (TOG) **43**(4), 1–17 (2024) 2

[67] Xiang, J., Yang, J., Huang, B., Tong, X.: 3d-aware image generation using 2d diffusion models. In: ICCV. pp. 2383–2393 (2023) 2

[68] Yin, W., Zhang, J., Wang, O., Niklaus, S., Mai, L., Chen, S., Shen, C.: Learning to recover 3d scene shape from a single image. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 204–213 (2021) 2

[69] Yu, A., Ye, V., Tancik, M., Kanazawa, A.: pixelnerf: Neural radiance fields from one or few images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4578–4587 (2021) 2

[70] Yu, H.X., Duan, H., Hur, J., Sargent, K., Rubinstein, M., Freeman, W.T., Cole, F., Sun, D., Snavely, N., Wu, J., et al.: Wonderjourney: Going from anywhere to everywhere. In: CVPR. pp. 6658–6667 (2024) 1, 2, 6, 7, 8

[71] Zhang, J., Li, X., Wan, Z., Wang, C., Liao, J.: Text2nerf: Text-driven 3d scene generation with neu-

ral radiance fields. IEEE Transactions on Visualization and Computer Graphics (2024) 1, 2, 3, 8

[72] Zhang, Y., Guo, X., Poggi, M., Zhu, Z., Huang, G., Mattoccia, S.: Completionformer: Depth completion with convolutions and vision transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18527–18536 (2023) 2

[73] Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., Torralba, A.: Places: A 10 million image database for scene recognition. IEEE transactions on pattern analysis and machine intelligence **40**(6), 1452–1464 (2017) 6

[74] Zhu, Z., Feng, X., Chen, D., Bao, J., Wang, L., Chen, Y., Yuan, L., Hua, G.: Designing a better asymmetric vqgan for stablediffusion. arXiv preprint arXiv:2306.04632 (2023) 7