

# iControl3D: An Interactive System for Controllable 3D Scene Generation

Xingyi Li

School of AIA, Huazhong University  
of Science and Technology  
Wuhan, China  
S-Lab, Nanyang Technological  
University  
Singapore, Singapore  
xingyi\_li@hust.edu.cn

Juewen Peng

College of Computing and Data  
Science, Nanyang Technological  
University  
Singapore, Singapore  
juewen.peng@ntu.edu.sg

Zhe Wang

SenseTime Research  
Hong Kong SAR, China  
wangzhe@sensetime.com

Yizheng Wu

School of AIA, Huazhong University  
of Science and Technology  
Wuhan, China  
S-Lab, Nanyang Technological  
University  
Singapore, Singapore  
yzwu21@hust.edu.cn

Kewei Wang

School of AIA, Huazhong University  
of Science and Technology  
Wuhan, China  
S-Lab, Nanyang Technological  
University  
Singapore, Singapore  
wangkewei@hust.edu.cn

Zhiguo Cao\*

School of AIA, Huazhong University  
of Science and Technology  
Wuhan, China  
zgcao@hust.edu.cn

Jun Cen

S-Lab, Nanyang Technological  
University  
Singapore, Singapore  
jcenaa@connect.ust.hk

Ke Xian

School of EIC, Huazhong University  
of Science and Technology  
Wuhan, China  
kxian@hust.edu.cn

Guosheng Lin

S-Lab, Nanyang Technological  
University  
Singapore, Singapore  
gslin@ntu.edu.sg

## Abstract

3D content creation has long been a complex and time-consuming process, often requiring specialized skills and resources. While recent advancements have allowed for text-guided 3D object and scene generation, they still fall short of providing sufficient control over the generation process, leading to a gap between the user's creative vision and the generated results. In this paper, we present iControl3D, a novel interactive system that empowers users to generate and render customizable 3D scenes with precise control. To this end, a 3D creator interface has been developed to provide users with fine-grained control over the creation process. Technically, we leverage 3D meshes as an intermediary proxy to iteratively merge individual 2D diffusion-generated images into a cohesive and unified 3D scene representation. To ensure seamless integration of 3D meshes, we propose to perform boundary-aware depth alignment before fusing the newly generated mesh with the existing one in 3D space. Additionally, to effectively manage depth discrepancies

\*Corresponding author.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

MM '24, October 28–November 1, 2024, Melbourne, VIC, Australia

© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.  
ACM ISBN 979-8-4007-0686-8/24/10  
<https://doi.org/10.1145/3664647.3680557>

between remote content and foreground, we propose to model remote content separately with an environment map instead of 3D meshes. Finally, our neural rendering interface enables users to build a radiance field of their scene online and navigate the entire scene. Extensive experiments have been conducted to demonstrate the effectiveness of our system. The code will be made available at <https://github.com/xingyi-li/iControl3D>.

## CCS Concepts

- Human-centered computing → Interactive systems and tools;
- Computing methodologies → Computer vision.

## Keywords

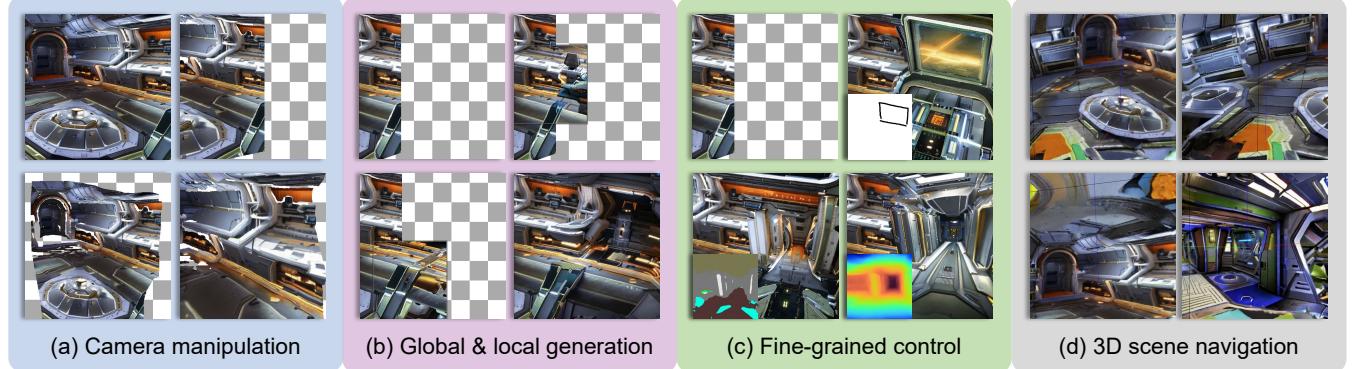
Interactive User Interface, 3D Scene Generation, Controllable Generation, Mesh, Neural Rendering

## ACM Reference Format:

Xingyi Li, Yizheng Wu, Jun Cen, Juewen Peng, Kewei Wang, Ke Xian, Zhe Wang, Zhiguo Cao, and Guosheng Lin. 2024. iControl3D: An Interactive System for Controllable 3D Scene Generation. In *Proceedings of the 32nd ACM International Conference on Multimedia (MM '24), October 28–November 1, 2024, Melbourne, VIC, Australia*. ACM, New York, NY, USA, 17 pages. <https://doi.org/10.1145/3664647.3680557>

## 1 Introduction

Recent years have witnessed explosive growth in the development of generative image and video models. In particular, diffusion models [14, 16, 39, 46] have pushed the boundaries of image generation,



**Figure 1: Our system empowers users to generate and render customizable 3D scenes with precise control over the 3D scene generation process. With our system, users can actively participate in the 3D scene creation process. For example, they can (a) manipulate the virtual camera to any viewpoint, (b) adjust the size of the selection box to generate global and local content, and try different random seeds to generate various results. (c) Besides text prompts, users can achieve fine-grained control over the output by adding extra conditions such as scribbles, semantic segmentation maps, and depth. (d) After generating 3D scenes, they can navigate the entire scene and create camera trajectories to render videos according to their preferences.**

or AI-Generated Content (AIGC) to an unprecedented level of realism, with their outputs often indistinguishable from real images. Despite the success in the 2D domain, generating 3D assets and realistic 3D scenes remains a complex process that requires a significant amount of expertise and specialized software. It can take years of practice to master the necessary skills and techniques involved in the 3D content creation.

In light of this, many researchers are eager to extend the power of 2D diffusion models to the field of 3D generation. Existing works [26, 28, 29, 33, 51] have demonstrated the potential of text-guided 3D object generation using 2D diffusion. Yet, these methods present challenges when it comes to generating 3D structures and textures on a scene-scale level. Inspired by previous studies [5, 25, 27], Fridman et al. [15] introduce SceneScape, a novel method for text-driven perpetual view generation. While SceneScape enables the synthesis of flying-out trajectories of scenes from text, it struggles with generating complete 3D scenes. Concurrently, Text2Room [17] proposes to create room-scale textured 3D meshes by using pre-trained 2D text-to-image diffusion models. However, it is restricted to indoor scene generation and offers limited control over the synthesis process, since only text and predefined camera trajectory are available. This can be frustrating for users who have specific creative visions for their 3D scene generation, as they cannot directly manipulate the scene’s features or details to match their preferences.

In this paper, we present a novel system that can generate 3D scenes while providing users with fine-grained control over the creation process (see Fig. 1). Despite the existence of 3D generative models [4, 10, 57], the availability of large-scale 3D datasets required for their training is still limited. Motivated by prior works [15, 26], we instead rely on 2D diffusion models [39] that have been pre-trained on a large number of 2D images. For 3D scene generation, we use 3D meshes as an intermediary proxy to merge individual 2D images into a unified representation.

Our system builds upon a generative RGB-D fusion method. Specifically, we begin by obtaining an input image from the user or generating one using 2D diffusion. We then utilize a monocular depth estimator [3] to estimate the underlying geometry of the image and unproject it into 3D space to generate an initial mesh. After transforming the virtual camera to a new viewpoint, we render the mesh and apply 2D diffusion to inpaint holes and outpaint for new content. To ensure seamless integration of the generated content with the existing mesh, we estimate the depth of the image from that viewpoint and perform boundary-aware depth alignment. We then fuse the new mesh with the existing one in 3D space. The above process is repeated iteratively until we obtain a satisfactory complete 3D structure. However, outdoor scenes often pose challenges as 3D meshes cannot handle dramatic depth discontinuities well. To address this issue, we propose to model remote content (e.g., sky) separately with an environment map. This leads to more realistic outdoor scene representation.

To provide users with fine-grained control over the creation process, we develop a 3D creator interface that enables users to actively participate in the 3D scene creation process. Our interface offers several advantages. First, users can manipulate the virtual camera to any viewpoint and customize camera trajectories to create personalized 3D scenes. Second, users can adjust the size of the selection box to generate local content, and try different random seeds to generate various results. Third, inspired by ControlNet [54], we adopt a neural network structure to control diffusion models by adding extra conditions such as user scribbles, semantic segmentation maps, depth, and other information to achieve fine-grained control over the generation process. Finally, we introduce a neural rendering interface and incorporate Neural Radiance Fields (NeRFs) [31, 48] into our system, allowing users to create a radiance field of their scene online and navigate the entire scene. Users can also create camera trajectories to render videos according to their preferences.

In summary, our main contributions are:

- We present a new interactive system to generate and render customizable 3D scenes with user control. To this end, we introduce a 3D creator interface and a neural rendering interface.
- Our proposed boundary-aware depth alignment allows for the seamless integration of 3D meshes. To better handle outdoor scenes, we propose to model remote content with an environment map rather than 3D meshes.
- We achieve interactive 3D scene generation with precise controllability.

## 2 Related Work

*3D-aware image synthesis.* Various 3D-GAN based methods [7, 8, 32, 42] have been proposed to combine neural scene representations with 2D generative models for 3D-aware image synthesis, enabling direct camera control. While these methods have demonstrated impressive results on the problem of generating single objects such as cars or faces, they are challenging to apply to large and diverse scenes. To extend 3D-aware image synthesis from single objects to completely unconstrained 3D scenes, several recent works [1, 13, 19, 45, 58] have been proposed. For example, GSN [13] proposes to break the radiance field into a grid of local radiance fields and collectively represent a scene by conditioning it on a 2D grid of floorplan latent codes. Bautista et al. [1] present GAUDI, where they first optimize a latent representation that disentangles radiance fields and camera poses, and then use the disentangled latent representation to learn a generative model. This allows for both unconditional and conditional generation of 3D scenes. However, these methods usually have a significant demand for extensive training and large-scale training data, limiting their generalization to only specific domains. Instead, our objective is to generate diverse 3D scenes.

*Perpetual view generation.* Perpetual view generation [20, 27] refers to the process of generating a continuous video sequence that corresponds to an arbitrary camera trajectory, using only a single image of the scene as input. Different kinds of methods have been explored in the literature. One line of research [23, 37, 49, 52] has focused on synthesizing indoor scenes with controllable camera trajectories. Motivated by Liu et al. [27], recent works such as InfNat-Zero [25] and DiffDreamer [5] aim at synthesizing fly-through videos of natural landscapes along long camera trajectories. Yet, due to their per-frame generation framework and the lack of underlying scene representations, these methods may suffer from issues such as domain drifting and inconsistent novel views. Recent studies [6, 12] learn a generative model for unconditional synthesis of unbounded 3D nature scenes with a persistent 3D scene representation. Although these methods are capable of producing view-consistent flythrough videos, they necessitate significant training on large-scale datasets and are restricted to a specific domain, e.g., landscapes. On the contrary, our system can generate diverse 3D scenes without the need for large-scale training.

*3D content generation.* Diffusion models [16, 35, 39, 40, 46, 47] have demonstrated remarkable success in generating highly realistic images and videos. By iteratively applying a series of steps, these models can transform a simple noise distribution into a

complex, high-dimensional data distribution, resulting in images and videos that are virtually indistinguishable from real-world data. As diffusion models continue to advance and gain popularity in the 2D domain, researchers are exploring the possibility of using 2D diffusion priors to generate 3D content. Recent works [9, 18, 24, 26, 28, 29, 33, 44, 50, 51] have shown promise in text-guided 3D object generation, but challenges remain in generating large-scale 3D structures and textures for entire scenes. Motivated by previous studies in perpetual view generation [5, 25, 27], Fridman et al. [15] propose SceneScape, a text-driven approach that synthesizes flying-out trajectories of scenes using 2D diffusion. However, SceneScape struggles with generating complete 3D scenes. Text2Light [11] introduces a zero-shot text-driven HDR panorama generation framework for creating 3D scenes but fails to impress users with freely moving cameras. Concurrently, Text2Room [17] uses pre-trained 2D text-to-image diffusion models to create textured 3D meshes of indoor scenes but offers limited control over the output. To bridge this gap, we present a novel system that can generate and render customizable 3D scenes with user control.

## 3 Method

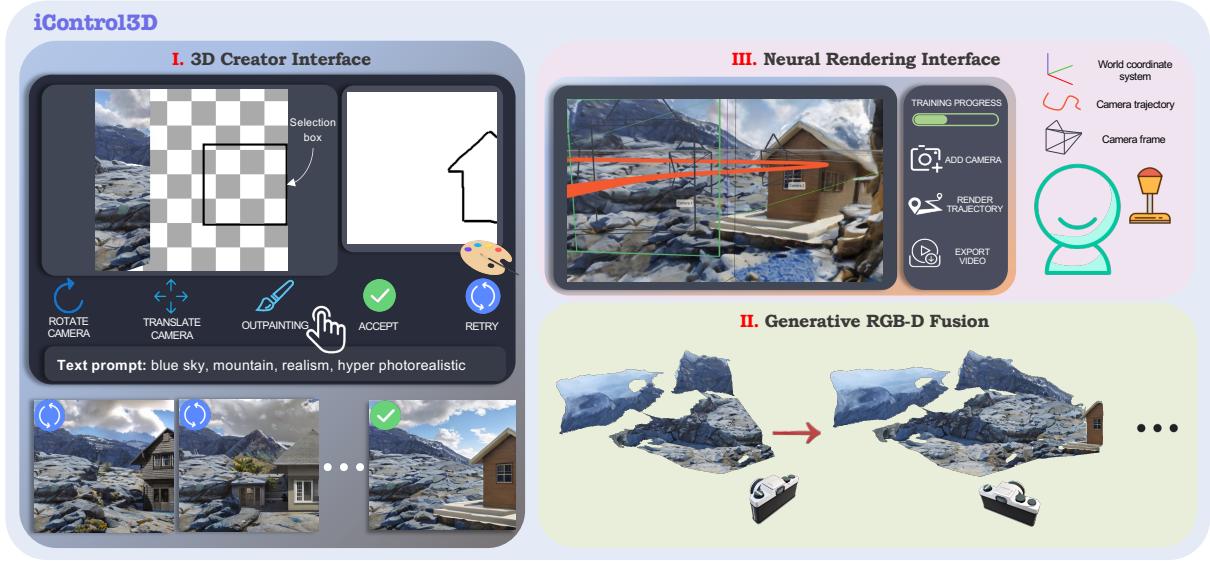
### 3.1 System Overview

Our goal is to generate diverse 3D scenes while providing users with fine-grained control over the creation process. This entails tackling two challenges, i.e., leveraging 2D diffusion priors for consistent 3D scene generation and providing users with controllability over the creation process. To achieve our goal, we present iControl3D, an interactive system for 3D scene generation with user control. We schematically illustrate our system in Fig. 2.

Our system mainly consists of a generative RGB-D fusion module, a 3D creator interface, and a neural rendering interface. Our system begins by obtaining an input image from users or generating one using 2D diffusion [39], estimating its geometry via a depth estimator [3], and generating an initial 3D mesh. We then render the mesh from different viewpoints, apply inpainting, perform boundary-aware depth alignment, and fuse it with the existing mesh, iteratively refining it until a satisfactory 3D structure is obtained. To handle outdoor scenes with depth discontinuities, we model remote content separately with an environment map, resulting in a more realistic representation. Unlike previous methods that offer a limited degree of user control, our system presents a 3D creator interface that enables users to actively participate in the 3D scene creation process. We also incorporate ControlNet [54], which can control diffusion models by adding extra conditions, into our interface to provide users with fine-grained control over the synthesized outputs. After generating 3D scenes, our neural rendering interface then builds a radiance field online and enables users to navigate the entire scene and create camera trajectories to render videos according to their preferences.

### 3.2 Generative RGB-D Fusion

**Initialization.** Motivated by previous works [15, 26], we leverage 2D diffusion models [39] that have been pre-trained on a large number of 2D images. Our system starts by obtaining an input image  $I_0$  from users or generating one using 2D diffusion. Formally, let  $\mathcal{G}$  be a pre-trained 2D diffusion model. We then can generate



**Figure 2: System overview.** (I) Within our 3D creator interface, users are allowed to manipulate the camera to any viewpoint, adjust the size of the selection box to generate local content, and try different random seeds to create a variety of results. Moreover, users can achieve fine-grained control over the generation process by adding extra conditions such as user scribbles; (II) Once the generated result in (I) is accepted by the users, our generative RGB-D fusion module fuses it with the existing mesh. This alternating process between (I) and (II) continues until a satisfactory 3D structure is obtained; (III) After generating 3D scenes, our neural rendering interface then builds a radiance field online and enables users to navigate the entire scene. By recording their virtual journey through the scene, users can also produce high-quality videos that showcase the intricacies and beauty of their designs.

the input image  $I_0$  using 2D diffusion model  $\mathcal{G}$ :

$$I_0 = \mathcal{G}(T, z), \quad (1)$$

where  $T$  is a text prompt and  $z$  represents additional conditions, e.g., user scribbles, semantic segmentation maps, and depth maps. It is worth noting that 2D diffusion models can only generate independent 2D images without any 3D structural relationship between them. Hence, relying solely on 2D diffusion models is insufficient to create a unified 3D scene. Inspired by prior works [17], we leverage 3D meshes as an intermediary proxy to merge individual 2D images generated by 2D diffusion models into a unified 3D scene representation. To this end, we utilize an off-the-shelf monocular depth estimator [3] to estimate the underlying geometry of the input image. After that, we proceed to unproject the input image into an initial 3D mesh  $\mathcal{M}_0 = (\mathcal{V}, \mathcal{F}, \mathcal{C})$  using depth values, where  $\mathcal{V} = \{v_i\}_{i=1}^N$  is the set of  $N$  vertices,  $\mathcal{F} = \{f_i\}_{i=1}^F$  is the set of  $F$  faces with each connecting three vertices, and  $\mathcal{C} = \{c_i\}_{i=1}^N$  are the color vectors attached on vertices.

**Mesh projection and inpainting.** We now have the initial 3D mesh  $\mathcal{M}_0$ . Our next step is to build up the scene iteratively. To do this, we generate new content from previously unobserved viewpoints. Specifically, we first render the mesh in the target camera pose  $\mathbf{P}_{t+1}$ :

$$\hat{I}_{t+1}, \hat{D}_{t+1}, \hat{\mathbf{m}}_{t+1} = \Pi(\mathcal{M}_t, \mathbf{P}_{t+1}). \quad (2)$$

The mesh renderer  $\Pi$  [36] produces the rendered image  $\hat{I}_{t+1}$ , the rendered depth  $\hat{D}_{t+1}$  and the rendered mask  $\hat{\mathbf{m}}_{t+1}$  indicating the visible regions of the mesh in the rendered image, where pixels

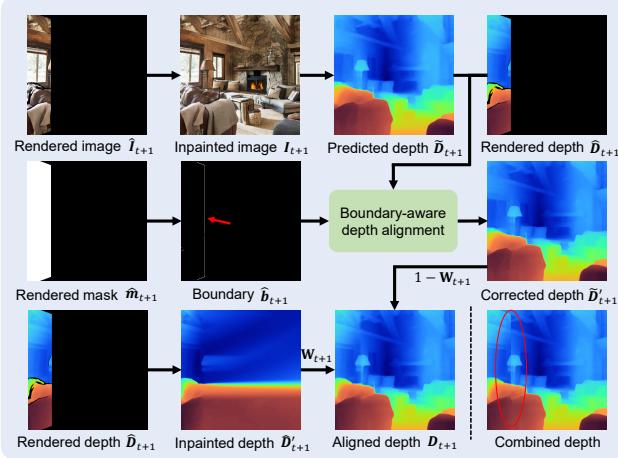
corresponding to visible and invisible parts of the mesh are set to 1 and 0, respectively. To create new content, the 2D diffusion model  $\mathcal{G}$  is employed to inpaint missing pixels via

$$I_{t+1} = \mathcal{G}\left(\hat{I}_{t+1}, \sim\hat{\mathbf{m}}_{t+1}, T, z\right), \quad (3)$$

where  $\sim\hat{\mathbf{m}}_{t+1}$  is the inverted mask used to guide the diffusion model by highlighting the areas of the image that should be inpainted.

**Boundary-aware depth alignment.** Likewise, we then employ the depth estimator to predict the underlying geometry of  $I_{t+1}$ , denoted as  $\tilde{D}_{t+1}$ . It should be noted that the depth of shared regions between the predicted depth map  $\tilde{D}_{t+1}$  and the rendered depth map  $\hat{D}_{t+1}$  may differ. To ensure seamless integration of the generated content with the existing mesh, it is intuitive to align the depth such that similar regions in a scene are placed at a similar depth as much as possible. This can help to avoid abrupt transitions at the boundaries between the generated content and the existing mesh. SceneScape [15] utilizes an online test-time training technique to promote the predicted depth map of the current frame to be in line with the geometric structure of the synthesized scene. However, this technique requires a certain amount of time to achieve depth alignment, making it unsuitable for real-time applications.

To this end, we propose boundary-aware depth alignment. The rationale behind incorporating boundary-aware depth alignment is to minimize the time needed for depth alignment, rendering it suitable for real-time 3D scene generation. This eliminates the necessity of prolonged waiting periods before progressing to the



**Figure 3: Boundary-aware depth alignment.** Directly combining the rendered depth  $\hat{D}_{t+1}$  and the predicted depth  $\hat{D}_{t+1}$  leads to abrupt transitions in the combined depth while our boundary-aware depth alignment ensures a more seamless depth fusion.

next step in scene creation. As shown in Fig. 3, we first obtain the boundary  $\hat{b}_{t+1}$  of the rendered mask  $\hat{m}_{t+1}$  via:

$$\hat{b}_{t+1} = (\sim \hat{m}_{t+1} \oplus B) \cap \hat{m}_{t+1}, \quad (4)$$

where  $\oplus$  and  $\cap$  respectively denote the dilation and intersection operation, while  $B$  represents a structuring element used for the dilation operation. Inspired by Liu et al. [27], we then conduct our boundary-aware depth alignment by solving the following least squares problem:

$$\min_{\alpha, \beta} \left\| \hat{b}_{t+1} \odot \left( \frac{\alpha}{\hat{D}_{t+1}} + \beta - \frac{1}{\hat{D}_{t+1}} \right) \right\|^2, \quad (5)$$

where  $\odot$  denotes the Hadamard (i.e., element-wise) product. Intuitively, this optimization attempts to scale and shift the predicted disparity using  $\alpha$  and  $\beta$ , so that the aligned disparity of the boundary region matches the rendered disparity. After obtaining the optimal scale and shift parameters, we can use them to compute the corrected depth  $\tilde{D}'_{t+1}$  as follows:

$$\tilde{D}'_{t+1} = \frac{1}{\alpha/\tilde{D}_{t+1} + \beta}. \quad (6)$$

To ensure a smoother depth transition, we further propose a depth blending technique. Specifically, we first inpaint the rendered depth  $\hat{D}_{t+1}$  with the Navier-Stokes inpainting algorithm [2], resulting in the inpainted depth  $\hat{D}'_{t+1}$ . Next, we blend the corrected depth  $\tilde{D}'_{t+1}$  with the inpainted depth  $\hat{D}'_{t+1}$  to compute the aligned depth  $D_{t+1}$  as follows:

$$D_{t+1} = W_{t+1} \cdot \tilde{D}'_{t+1} + (1 - W_{t+1}) \cdot \hat{D}'_{t+1}, \quad (7)$$

where the weight map  $W_{t+1}$  is obtained by applying Gaussian blur to  $\hat{m}_{t+1}$ .

**Mesh fusion.** Given the aligned depth map  $D_{t+1}$ , our next step is to generate a new 3D mesh representation of the scene  $\hat{M}_{t+1}$  from

the 2D image  $I_{t+1}$  and fuse it with the existing mesh  $M_t$ . We first unproject the image pixels into 3D space using the camera intrinsic matrix and target camera pose  $P_{t+1}$ . Once we have a set of 3D points representing the scene, we follow Höller et al. [17] and use a triangulation scheme to construct a mesh representation  $\hat{M}_{t+1}$ . This scheme involves connecting each set of four neighboring points in a regular grid pattern to form two triangles.

To fuse the new 3D mesh  $\hat{M}_{t+1}$  with the existing mesh  $M_t$ , we extend the triangulation scheme at the edges of the inpainting mask  $\sim \hat{m}_{t+1}$  to connect these faces with their neighboring faces from the existing mesh  $M_t$ . This process results in the fusion of the two meshes, producing the final mesh  $M_{t+1}$ .

**Environment map modeling.** We can repeat the aforementioned process iteratively until we achieve a satisfactory 3D structure. However, outdoor scenes may pose a challenge as 3D meshes struggle to handle dramatic depth discontinuities, e.g., between the sky and ground. These discontinuities often lead to flawed structures or large holes in the reconstructed mesh, leading to visible artifacts in the fusion of the new 3D mesh with the existing one. To this end, we propose to model remote content separately with an environment map. Specifically, we assume that remote content has an infinite depth and can be represented as a texture on a sphere surrounding the scene. To embed the remote region into the environment map, for each  $I_{t+1}$ , we use SAM [22] to segment the remote region in the image  $I_{t+1}$  and map each pixel in the segmented region to a point on the surface of a sphere using inverse equirectangular projection. When we change to the next viewpoint, we first obtain the remote content from the environment map, followed by the mesh projection. This allows for accurate rendering of remote regions in subsequent steps, bypassing the issue of depth discontinuities between remote content and foreground.

### 3.3 3D Creator Interface

Current methods [6, 11, 12, 15, 17, 37] provide limited control over the synthesis process as they only allow for text and predefined camera trajectories as input. This can be frustrating for users who have specific creative visions or requirements for their 3D scene generation, as they cannot directly manipulate the scene's features or details to match their preferences. To overcome this limitation, we introduce a 3D creator interface as a key component of our system. The interface provides a user-friendly and intuitive way for users to actively participate in the 3D scene creation process. Our 3D creator interface offers several advantages (see Fig. 1). One of the most notable features of our interface is the ability for users to adjust the size of the selection box, allowing them to generate local content and try different random seeds to create a variety of results. This feature gives users the ability to select the best output that matches their creative vision. The virtual camera module is another highlight of our interface. It allows users to manipulate the camera to any viewpoint and customize camera trajectories, providing a personalized experience for creating 3D scenes.

**Fine-grained control.** Our objective is to provide users with not only full control over the 3D scene creation process but also fine-grained control to achieve their desired level of detail and customization. Inspired by ControlNet [54], we adopt a neural network structure to control diffusion models, which allows users to achieve

**Table 1: Quantitative comparisons.** We show that our system outperforms all baselines in terms of both the Inception Score (IS) [41] and CLIP Score (CS) [34].

Method	IS ↑	CS ↑
LOR [37]	1.77	20.96
SceneDreamer [12]	1.35	21.35
Persistent Nature [6]	1.33	28.20
Text2Room [17]	2.57	28.50
Ours	<b>2.63</b>	<b>29.77</b>

fine-grained control over the generation process by adding extra conditions such as user scribbles, semantic segmentation maps, depth, and other information. This feature enables users to create more complex and detailed 3D scenes.

### 3.4 Neural Rendering Interface

Our generative RGB-D fusion module uses 3D meshes as an intermediary proxy to merge individual 2D images into a unified 3D scene representation. However, we do not employ hole-filling and smoothing techniques as used in previous works [17]. This is because we empirically find that iterative mesh reconstruction often leads to unavoidable artifacts in 3D meshes. We instead propose to leverage the 2D diffusion-generated images, which are usually visually pleasing. These images have shared a 3D structural relationship due to our generative RGB-D fusion module. We, therefore, introduce a neural rendering interface and integrate Neural Radiance Fields [31, 48] into our system to further smooth the artifacts shown in 3D meshes. We train a neural radiance field using the 2D diffusion-generated images and their corresponding poses. This enables users to create a radiance field of their scene online and navigate the entire scene during training and after training. Our neural rendering interface offers users an immersive way to explore their 3D creations and produce customized videos.

## 4 Experiments

In this section, we present a comprehensive evaluation of our system on a diverse range of indoor and outdoor scenes and compare its performance with state-of-the-art methods both quantitatively and qualitatively. Additionally, we conduct a user study to better evaluate the effectiveness of our system. Finally, we perform an ablation study to justify our design choices.

### 4.1 Baselines

In our experiments, we primarily compare ours against four representative works including LOR [37], SceneDreamer [12], Persistent Nature [6], and Text2Room [17]. Specifically, LOR [37] is an autoregressive method that can generate long-term 3D indoor scene video from a single image but presents challenges when it comes to generating consistent 3D structures and textures on a scene-scale level. SceneDreamer [12] and Persistent Nature [6] learn a generative model for unconditional synthesis of unbounded 3D nature scenes with a persistent 3D scene representation, but necessitate significant training on large-scale datasets and are restricted to a specific domain. Text2Room [17] uses pre-trained 2D text-to-image

**Table 2: User study.** We conduct a user study to compare our system against competitive methods. All methods are evaluated on the perceptual quality (PQ) of the imagery and scene diversity (SD). Here we only present pairwise comparison results between ours and baselines.

Comparison	PQ ↑	SD ↑
LOR [37] / Ours	7.6% / <b>92.4%</b>	1.7% / <b>98.3%</b>
SceneDreamer [12] / Ours	29.5% / <b>70.5%</b>	11.9% / <b>88.1%</b>
Persistent Nature [6] / Ours	17.7% / <b>82.3%</b>	13.8% / <b>86.2%</b>
Text2Room [17] / Ours	18.6% / <b>81.4%</b>	25.0% / <b>75.0%</b>

diffusion models to create textured 3D meshes of indoor scenes but lacks fine-grained control over the synthesis process.

## 4.2 Results

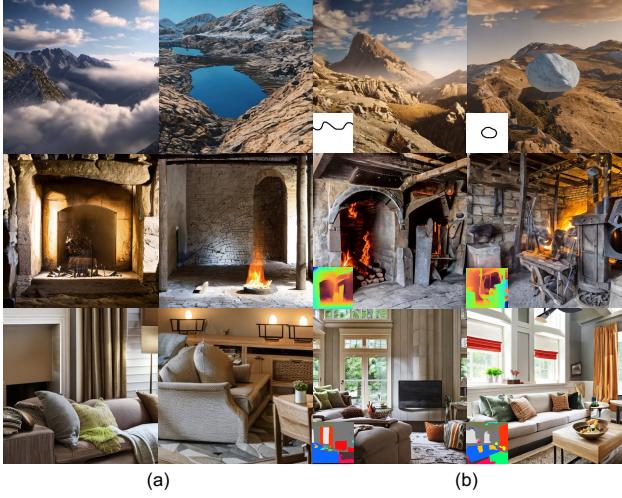
**Evaluation metrics.** To evaluate our system, we utilize Inception Score [41] and CLIP Score [34] as our evaluation metrics. A higher Inception Score indicates that the generated images have both high quality and diversity, whereas a higher CLIP Score signifies a greater similarity between the generated image and the given text prompt.

**Quantitative comparisons.** We adopt 21 scene settings, including 6 challenging outdoor settings such as “mountain” and “garden”, and 15 indoor settings such as “living room” and “spaceship”, and randomly generate outdoor scenes twice and indoor scenes once, resulting in 12 outdoor scenes and 15 indoor scenes. Since our focus is not on achieving complete mesh reconstruction, we instead render 200 images to compute both Inception Score and CLIP Score for each scene. In our evaluation, we closely follow Text2Room. It employs 20 different trajectories for method evaluation, generating 60 images from novel viewpoints for each scene to calculate 2D metrics. Likewise, we adopt 21 scene settings, totaling 27 scenes for each method, and generate 200 images for each scene to compute both the IS and CS. Therefore, our evaluation scale aligns with that of Text2Room. As shown in Table 1, our system outperforms existing baselines, which indicates that our system produces high-quality and diverse images across different scene settings.

**Qualitative comparisons.** Fig. 7 and Fig. 8 present a qualitative comparison between our system and baselines. We showcase randomly extracted novel views of generated scenes. We find that LOR [37] exhibits the tendency to produce inconsistent novel views and susceptibility to error accumulation. These limitations can lead to domain drifting and a decline in output quality. While SceneDreamer [12] and Persistent Nature [6] can synthesize large camera trajectories consistently, they require extensive training and are limited to specific domains such as landscapes. On the other hand, Text2Room [17] performs well in indoor scenarios but faces challenges when dealing with outdoor scenes. It also often produces over-smoothed regions in the reconstructions. In contrast, our system can generate high-quality novel views in both indoor and outdoor scenes. In addition, we show in Fig. 4 that our system can achieve fine-grained control.

### 4.3 User Study

To further evaluate the performance of our system, we conduct a user study involving 65 participants with diverse backgrounds and



**Figure 4: Fine-grained control.** Compared to (a) current text-driven methods [15, 17], (b) our system can achieve fine-grained control over the output by adding extra conditions such as scribbles, depth, and semantic segmentation maps.

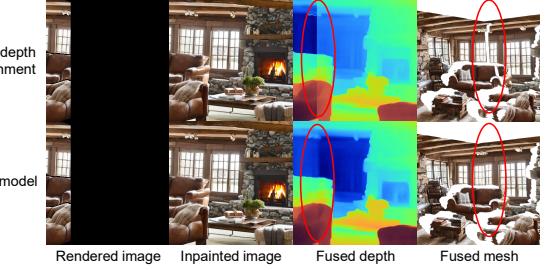
**Table 3: Ablation study.** We perform an ablation study on different components of our model to investigate their influence. Each component of our model contributes to the overall performance.

Method	IS ↑	CS ↑
w/o boundary-aware depth alignment	2.19	28.37
w/o environment map	2.53	29.02
Full model	<b>2.63</b>	<b>29.77</b>

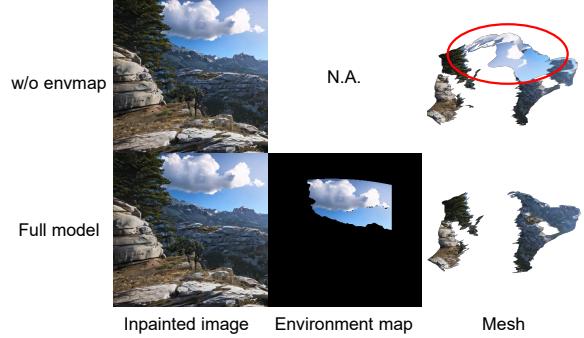
expertise in the field. We use different approaches to generate 60 free-navigating videos of various scenes, respectively. To prevent participants from guessing which results are generated by our system during the user study, we randomly present two sets of three videos each time. Both sets consist of three videos generated by randomly selected methods, rather than having one set generated exclusively by our system and the other set by another method. Participants are asked to compare two key aspects: the perceptual quality of the imagery and scene diversity. They are invited to choose the method with better perceptual quality and scene diversity, or none if difficult to judge. We report the results in Table 5, which points out that our system achieves higher perceptual quality and scene diversity compared to the alternative methods.

#### 4.4 Ablation Study

To validate the effectiveness of each component of our system, we also conduct an ablation study. We design two variants of our system by removing boundary-aware depth alignment and environment map modeling while keeping the rest of the pipeline intact. As shown in Fig. 5 and Fig. 6, both boundary-aware depth alignment and environment map modeling contribute to the overall performance. Table 3 also confirms the effect of these components.



**Figure 5: Effectiveness of boundary-aware depth alignment.** Without boundary-aware depth alignment, the generated mesh may exhibit abrupt transitions at the boundaries between the newly generated content and the existing mesh.



**Figure 6: Effectiveness of environment map modeling.** Without environment map modeling, handling outdoor scenes with 3D meshes becomes challenging due to dramatic depth discontinuities, leading to visible artifacts.

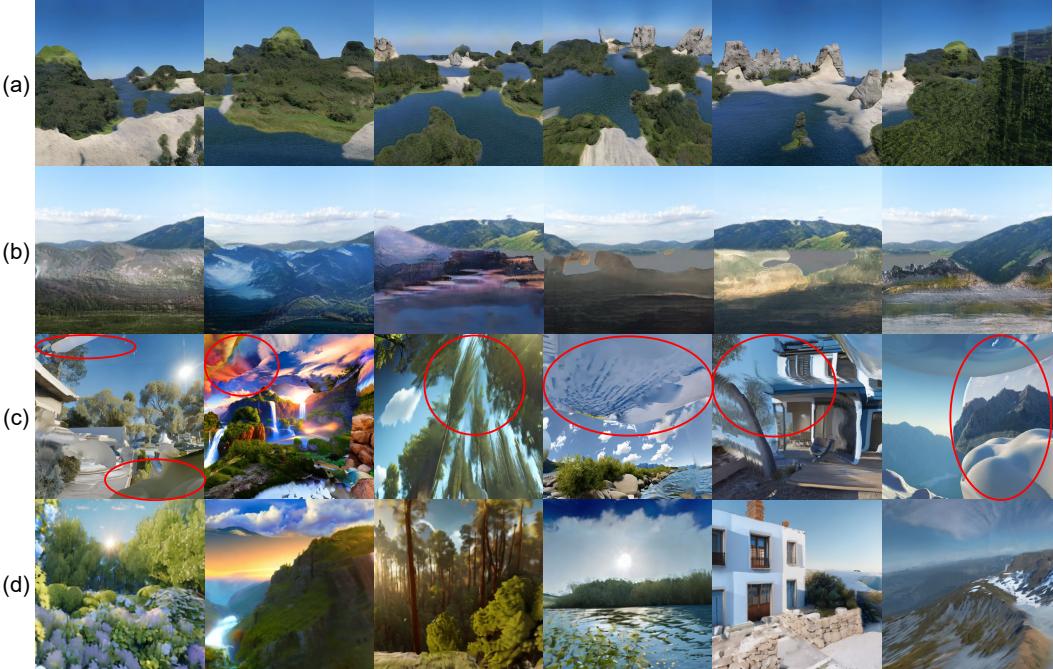
## 5 Conclusion

In this paper, we introduce iControl3D, an interactive system for controllable 3D scene generation and rendering. To achieve this, we develop a 3D creator interface to provide users with fine-grained control over the creation process and a neural rendering interface to allow them to navigate the entire scene. We show that our system can generate diverse 3D scenes with user control. We conduct extensive experiments to verify the effectiveness of our system. We hope that our system will inspire and empower users to unleash their creativity and bring their imaginations to life in the world of 3D content creation.

**Limitation.** While our method provides a user-friendly platform for interactive 3D content creation, certain challenges can impact its performance. One such challenge arises when the depth prediction module produces inaccurate geometry based on the input image, or when the segmentation model fails to predict with precision. These issues can compromise the quality of the generated 3D scenes. Moreover, distortions in the 3D meshes can further contribute to inaccuracies and inconsistencies, ultimately affecting the overall realism and quality.



**Figure 7: Qualitative comparison on indoor scenes.** Here we present the qualitative results of indoor scenes, displayed alternately from left to right. The scenes, in sequence, are “spaceship”, “forge”, “library”, “cave”, “ice castle”, and “small office”. (a) LOR [37], (b) Text2Room [17], and (c) ours. As can be seen, (a) LOR [37] is prone to domain drifting and a decline in output quality. Although (b) Text2Room [17] performs well on indoor scenes, it often produces over-smoothed artifacts in the reconstructions. In contrast, (c) our system presents diverse and photo-realistic results.



**Figure 8: Qualitative comparison on outdoor scenes.** We present the qualitative results of outdoor scenes, displayed alternately from left to right. The scenes, in sequence, are “garden”, “waterfall”, “forest”, “river”, “house”, and “mountain”. (a) Scene-Dreamer [12], (b) Persistent Nature [6], (c) Text2Room [17], and (d) ours. Note that (a) SceneDreamer [12] and (b) Persistent Nature [6] require extensive training and are limited to a specific domain, i.e., landscapes. While (c) Text2Room [17] can also generate outdoor scenes, it suffers from notable mesh distortions and artifacts. By contrast, (d) our system can generate high-quality and consistent novel views across diverse domains.

## Acknowledgments

This study is supported under the RIE2020 Industry Alignment Fund – Industry Collaboration Projects (IAF-ICP) Funding Initiative, as well as cash and in-kind contribution from the industry partner(s). This work is also supported by the MOE AcRF Tier 2 grant (MOE-T2EP20220-0007).

## References

- [1] Miguel Angel Bautista, Pengsheng Guo, Samira Abnar, Walter Talbott, Alexander Toshev, Zhuoyuan Chen, Laurent Dinh, Shuangfei Zhai, Hanlin Goh, Daniel Ulbricht, et al. 2022. Gaudi: A neural architect for immersive 3d scene generation. *Advances in Neural Information Processing Systems (NeurIPS) 35* (2022), 25102–25116.
- [2] Marcelo Bertalmio, Andrea L Bertozzi, and Guillermo Sapiro. 2001. Navier-stokes, fluid dynamics, and image and video inpainting. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, Vol. 1. IEEE, I–I.
- [3] Sharif Farooq Bhat, Reiner Birk, Diana Wofk, Peter Wonka, and Matthias Müller. 2023. Zoedepth: Zero-shot transfer by combining relative and metric depth. *arXiv preprint arXiv:2302.12288* (2023).
- [4] Ruojin Cai, Guandao Yang, Hadar Averbuch-Elor, Zekun Hao, Serge Belongie, Noah Snavely, and Bharath Hariharan. 2020. Learning gradient fields for shape generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 364–381.
- [5] Shenggu Cai, Eric Ryan Chan, Songyou Peng, Mohamad Shahbazi, Anton Obukhov, Luc Van Gool, and Gordon Wetzstein. 2022. DiffDreamer: Consistent Single-view Perpetual View Generation with Conditional Diffusion Models. *arXiv preprint arXiv:2211.12131* (2022).
- [6] Lucy Chai, Richard Tucker, Zhengqi Li, Phillip Isola, and Noah Snavely. 2023. Persistent Nature: A Generative Model of Unbounded 3D Worlds. *arXiv preprint arXiv:2303.13515* (2023).
- [7] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. 2022. Efficient geometry-aware 3D generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 16123–16133.
- [8] Eric R Chan, Marco Monteiro, Petr Kellnhofer, Jiajun Wu, and Gordon Wetzstein. 2021. 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 (CVPR)*. 5799–5809.
- [9] Eric R Chan, Koki Nagano, Matthew A Chan, Alexander W Bergman, Jeong Joon Park, Axel Levy, Miika Aittala, Shalini De Mello, Tero Karras, and Gordon Wetzstein. 2023. GeNVS: Generative Novel View Synthesis with 3D-Aware Diffusion Models. *arXiv preprint arXiv:2304.02602* (2023).
- [10] Kevin Chen, Christopher B Choy, Manolis Savva, Angel X Chang, Thomas Funkhouser, and Silvio Savarese. 2019. Text2shape: Generating shapes from natural language by learning joint embeddings. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*. Springer, 100–116.
- [11] Zhaoxi Chen, Guangcong Wang, and Ziwei Liu. 2022. Text2Light: Zero-Shot Text-Driven HDR Panorama Generation. *ACM Transactions on Graphics (TOG)* 41, 6, Article 195 (2022), 16 pages.
- [12] Zhaoxi Chen, Guangcong Wang, and Ziwei Liu. 2023. Scenedreamer: Unbounded 3d scene generation from 2d image collections. *arXiv preprint arXiv:2302.01330* (2023).
- [13] Terrance DeVries, Miguel Angel Bautista, Nitish Srivastava, Graham W Taylor, and Joshua M Susskind. 2021. Unconstrained scene generation with locally conditioned radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 14304–14313.
- [14] Prafulla Dhariwal and Alexander Nichol. 2021. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems (NeurIPS) 34* (2021), 8780–8794.
- [15] Rafail Friedman, Amit Abecasis, Yoni Kasten, and Tali Dekel. 2023. Scenescape: Text-driven consistent scene generation. *arXiv preprint arXiv:2302.01133* (2023).
- [16] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems (NeurIPS) 33* (2020), 6840–6851.
- [17] Lukas Höller, Ang Cao, Andrew Owens, Justin Johnson, and Matthias Nießner. 2023. Text2Room: Extracting Textured 3D Meshes from 2D Text-to-Image Models. *arXiv preprint arXiv:2303.11989* (2023).
- [18] Ajay Jain, Ben Mildenhall, Jonathan T. Barron, Pieter Abbeel, and Ben Poole. 2022. Zero-Shot Text-Guided Object Generation With Dream Fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 867–876.
- [19] Varun Jampani, Huiwen Chang, Kyle Sargent, Abhishek Kar, Richard Tucker, Michael Krainin, Dominik Kaeser, William T Freeman, David Salesin, Brian Curless, and Ce Liu. 2021. SLIDE: Single Image 3D Photography with Soft Layering and Depth-aware Inpainting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- [20] Biliana Kaneva, Josef Sivic, Antonio Torralba, Shai Avidan, and William T Freeman. 2010. Infinite images: Creating and exploring a large photorealistic virtual space. *Proc. IEEE* 98, 8 (2010), 1391–1407.
- [21] Seung Wook Kim, Bradley Brown, Kangxue Yin, Karsten Kreis, Katja Schwarz, Daqing Li, Robin Rombach, Antonio Torralba, and Sanja Fidler. 2023. NeuralField-LDM: Scene Generation with Hierarchical Latent Diffusion Models. *arXiv preprint arXiv:2304.09787* (2023).
- [22] Alexander Kirillov, Eri Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. 2023. Segment anything. *arXiv preprint arXiv:2304.02643* (2023).
- [23] Jing Yu Koh, Honglak Lee, Yinfei Yang, Jason Baldridge, and Peter Anderson. 2021. Pathdreamer: A world model for indoor navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 14738–14748.
- [24] Han-Hung Lee and Angel X Chang. 2022. Understanding pure clip guidance for voxel grid nerf models. *arXiv preprint arXiv:2209.15172* (2022).
- [25] Zhengqi Li, Qianqian Wang, Noah Snavely, and Angjoo Kanazawa. 2022. Infinitenature-zero: Learning perpetual view generation of natural scenes from single images. In *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 515–534.
- [26] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. 2022. Magic3D: High-Resolution Text-to-3D Content Creation. *arXiv preprint arXiv:2211.10440* (2022).
- [27] Andrew Liu, Richard Tucker, Varun Jampani, Ameesh Makadia, Noah Snavely, and Angjoo Kanazawa. 2021. Infinitenature: Perpetual view generation of natural scenes from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 14458–14467.
- [28] Minghua Liu, Chao Xu, Haian Jin, Linghao Chen, Zexiang Xu, Hao Su, et al. 2023. One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization. *arXiv preprint arXiv:2306.16928* (2023).
- [29] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. 2023. Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 9298–9309.
- [30] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. 2023. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499* (2023).
- [31] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. 2020. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *Proceedings of the European Conference on Computer Vision (ECCV)*.
- [32] Michael Niemeyer and Andreas Geiger. 2021. Giraffe: Representing scenes as compositional generative neural feature fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 11453–11464.
- [33] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. 2022. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988* (2022).
- [34] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning (ICML)*. PMLR, 8748–8763.
- [35] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. 2022. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125* (2022).
- [36] Nikhila Ravi, Jeremy Reizenstein, David Novotny, Taylor Gordon, Wan-Yen Lo, Justin Johnson, and Georgia Gkioxari. 2020. Accelerating 3d deep learning with pytorch3d. *arXiv preprint arXiv:2007.08501* (2020).
- [37] Xuanchi Ren and Xiaolong Wang. 2022. Look outside the room: Synthesizing a consistent long-term 3d scene video from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 3563–3573.
- [38] Chris Rockwell, David F Fouhey, and Justin Johnson. 2021. Pixelsynth: Generating a 3d-consistent experience from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 14104–14113.
- [39] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-Resolution Image Synthesis With Latent Diffusion Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 10684–10695.
- [40] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems (NeurIPS) 35* (2022), 36479–36494.
- [41] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. 2016. Improved techniques for training gans. *Advances in Neural*

- Information Processing Systems (NeurIPS) 29* (2016).
- [42] Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. 2020. Graf: Generative radiance fields for 3d-aware image synthesis. *Advances in Neural Information Processing Systems (NeurIPS) 33* (2020), 20154–20166.
- [43] Yuan Shen, Wei-Chiu Ma, and Shenlong Wang. 2022. SGAM: Building a Virtual 3D World through Simultaneous Generation and Mapping. In *Advances in Neural Information Processing Systems (NeurIPS)*.
- [44] Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. 2023. Mvdream: Multi-view diffusion for 3d generation. *arXiv preprint arXiv:2308.16512* (2023).
- [45] Meng-Li Shih, Shih-Yang Su, Johannes Kopf, and Jia-Bin Huang. 2020. 3D Photography using Context-aware Layered Depth Inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [46] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning (ICML)*. PMLR, 2256–2265.
- [47] Chunyi Sun, Junlin Han, Weijian Deng, Xinlong Wang, Zishan Qin, and Stephen Gould. 2023. 3D-GPT: Procedural 3D Modeling with Large Language Models. *arXiv preprint arXiv:2310.12945* (2023).
- [48] Matthew Tancik, Ethan Weber, Evonne Ng, Ruilong Li, Brent Yi, Justin Kerr, Terrance Wang, Alexander Kristoffersen, Jake Austin, Kamyar Salahi, et al. 2023. Nerfstudio: A modular framework for neural radiance field development. *arXiv preprint arXiv:2302.04264* (2023).
- [49] Hung-Yu Tseng, Qimbo Li, Changil Kim, Suhib Alsisan, Jia-Bin Huang, and Johannes Kopf. 2023. Consistent View Synthesis with Pose-Guided Diffusion Models. *arXiv preprint arXiv:2303.17598* (2023).
- [50] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A Yeh, and Greg Shakhnarovich. 2022. Score Jacobian Chaining: Lifting Pretrained 2D Diffusion Models for 3D Generation. *arXiv preprint arXiv:2212.00774* (2022).
- [51] Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. 2023. ProlificDreamer: High-Fidelity and Diverse Text-to-3D Generation with Variational Score Distillation. *arXiv preprint arXiv:2305.16213* (2023).
- [52] Olivia Wiles, Georgia Gkioxari, Richard Szeliski, and Justin Johnson. 2020. Synsin: End-to-end view synthesis from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 7467–7477.
- [53] Lap-Pai Yu, Sai-Kit Yeung, and Demetri Terzopoulos. 2015. The Clutterpalette: An Interactive Tool for Detailing Indoor Scenes. *IEEE Transactions on Visualization and Computer Graphics* 22, 2 (2015), 1138–1148.
- [54] Lvmin Zhang and Maneesh Agrawala. 2023. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543* (2023).
- [55] Shao-Kui Zhang, Yi-Xiao Li, Yu He, Yong-Liang Yang, and Song-Hai Zhang. 2021. MageAdd: Real-Time Interaction Simulation for Scene Synthesis. In *Proceedings of the 29th ACM International Conference on Multimedia (ACM MM)*, 965–973.
- [56] Shao-Kui Zhang, Hou Tam, Yike Li, Ke-Xin Ren, Hongbo Fu, and Song-Hai Zhang. 2023. SceneDirector: Interactive Scene Synthesis by Simultaneously Editing Multiple Objects in Real-Time. *IEEE Transactions on Visualization and Computer Graphics* 30, 8 (2023), 4558–4569.
- [57] Linqi Zhou, Yilun Du, and Jiajun Wu. 2021. 3D Shape Generation and Completion Through Point-Voxel Diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 5826–5835.
- [58] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. 2018. Stereo magnification: learning view synthesis using multiplane images. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–12.

## A Contribution Revisited

Existing 3D scene generation methods often lack adequate user control, leading to the generation of scenes that may not align with users' preferences. Our primary objective is to overcome this limitation and provide users with enhanced controllability in the creation of 3D scenes. To achieve this goal, we propose an interactive system that seamlessly combines sequential scene generation with user controllability. Besides the interactive UI, one of the challenges is that the depth estimation network may produce depth maps that exhibit inconsistencies in scale between two frames. To solve this, we introduce a novel technique called boundary-aware depth alignment. This approach ensures a smooth integration of 3D meshes. Moreover, to better handle outdoor scenes, we integrate an environment map into our system, further enhancing the overall scene generation process.

Our proposed system differs from previous works in four ways: i) Boundary-aware depth alignment (faster, comparable performance); ii) Outdoor scene generation (not fully addressed in previous works); iii) Adding finer-grained control to the scene generation process (not fully addressed in previous works); iv) Consolidating the final results in neural radiance fields to ameliorate artifacts common to meshes.

## B Implementation Details

Our system leverages Stable Diffusion [39], which has been pre-trained on a large number of 2D images, to generate a diverse range of images. To enable controllable 3D scene generation, we integrate ControlNet [54] with the pre-trained large diffusion models to support additional input conditions and enhance control over the generated scenes. For estimating depth maps, we utilize an off-the-shelf monocular depth estimator [3] to estimate the underlying geometry of the input image. This allows us to predict dense depth maps for in-the-wild photos. The implementation of mesh reconstruction, projection, and fusion is carried out using PyTorch3D [36]. To segment remote content such as the sky, we first employ Grounding DINO [30] to detect remote regions based on text inputs. Then, we apply SAM [22] for precise segmentation. Our 3D creator interface is based on an open-source project<sup>1</sup> and implemented using PyScript and Gradio, providing a user-friendly interface for creating 3D scenes. The neural rendering interface is built on top of Nerfstudio [48], which enables users to navigate the entire scene freely and render customizable videos according to their preferences. We conduct all experiments on a single NVIDIA GeForce RTX 3090 GPU. Our code will be publicly available upon acceptance for academic purposes.

## C Baselines

Table 4 presents a comparison of our method with other relevant works. In our experiments, we primarily compare ours against four representative works including LOR [37], SceneDreamer [12], Persistent Nature [6], and Text2Room [17]. Specifically, LOR [37] is an autoregressive method that can generate long-term 3D indoor scene video from a single image but presents challenges when it comes to generating consistent 3D structures and textures on a scene-scale

<sup>1</sup><https://github.com/lkwq007/stablediffusion-infinity>

level. SceneDreamer [12] and Persistent Nature [6] learn a generative model for unconditional synthesis of unbounded 3D nature scenes with a persistent 3D scene representation, but necessitate significant training on large-scale datasets and are restricted to a specific domain. Text2Room [17] uses pre-trained 2D text-to-image diffusion models to create textured 3D meshes of indoor scenes but lacks fine-grained control over the synthesis process. We implement these works using the official codes released on GitHub.

We emphasize that our method distinguishes Text2Room in four significant ways. The most notable difference from Text2Room is the introduction of an interactive system designed to facilitate the comprehensive creation of a 3D scene by the user. The key strength of this system lies in its capacity to empower users with finer control over generated content. It allows for the integration of text with other modalities such as scribbles and semantic segmentation maps, offering users the capability to select specific parts of the scene for focus. Secondly, while Text2Room employs scale-and-shift depth alignment, our method goes a step further by incorporating a depth blending technique around the boundary. This enhancement ensures a smoother depth transition in the generated scenes. Thirdly, Text2Room is limited to handling indoor scenes, whereas our method extends its capabilities to generate outdoor scenes through the incorporation of environment maps. This broadens the scope of scene generation possibilities beyond indoor environments. Furthermore, we integrate Neural Radiance Fields into our system to further smooth the artifacts shown in 3D meshes.

## D Differences from Interactive Scene Synthesis Frameworks

Broadly speaking, those interactive scene synthesis frameworks [53, 55, 56] are tailored to assist modelers in manipulating groups of objects—usually CAD models—by enabling insertion, removal, translation, and rotation to enhance scene complexity. However, these methods primarily focus on indoor scenes and operating known objects. In contrast, our system exhibits versatility by generating diverse results and is not confined to indoor environments alone.

## E Experiment Details

In the paper, we provide a comprehensive comparison with LOR [37], SceneDreamer [12], Persistent Nature [6], and Text2Room [17]. We utilize their official codes and pre-trained models for comparison. To evaluate the performance of our system, we adopt 21 scene settings. This includes 6 challenging outdoor settings “mountain”, “garden”, “house”, “river”, “waterfall”, and “forest” as well as 15 indoor settings “baby room”, “bathroom”, “bedroom”, “cave”, “forge”, “ice castle”, “library”, “living room”, “farmhouse living room”, “modern living room”, “bedroom-bathroom combo”, “kitchen-living room combo”, “large office”, “small office”, and “spaceship”. Note that we only use LOR [37] to generate indoor scenes. For SceneDreamer [12] and Persistent Nature [6], we only utilize them to generate the “mountain” scene. For ours and Text2Room [17], we randomly generate outdoor scenes twice and indoor scenes once, resulting in a total of 12 outdoor scenes and 15 indoor scenes. For each scene, we render 200 images to compute both Inception Score and CLIP Score. For a fair evaluation, when compared with baselines such as Text2Room, we use identical camera trajectories and text prompts as Text2Room to generate 3D scenes and compute quantitative metrics.

**Table 4: Comparison of ours and relevant works.** *Indoor scene*: Designed for handling indoor scenes. *Outdoor scene*: Designed for handling outdoor scenes. *No large-scale training*: Not requiring large-scale training. *Radiance field*: If radiance fields are used. *Interactive generation*: If interactive generation is supported using an interface. *Local generation*: If local generation is supported. *Conditional synthesis*: If the synthesis can be conditioned on additional input. *Text control*: If the generation can be controlled by text prompts. *Fine-grained control*: If having precise control over the generation process, e.g., scribbles.

Method	Indoor scene	Outdoor scene	No large-scale training	Radiance field	Interactive generation	Local generation	Conditional synthesis	Text control	Fine-grained control
PixelSynth [38]	✓	✗	✗	✗	✗	✗	✓	✗	✗
InfinNat-Zero [25]	✗	✓	✗	✗	✗	✗	✓	✗	✗
LOR [37]	✓	✗	✗	✗	✗	✗	✓	✗	✗
SceneScape [15]	✓	✗	✓	✗	✗	✗	✓	✓	✗
GSN [13]	✓	✗	✗	✓	✗	✗	✓	✗	✗
SGAM [43]	✓	✓	✗	✗	✗	✗	✓	✗	✗
SceneDreamer [12]	✗	✓	✗	✓	✗	✗	✗	✗	✗
Persistent Nature [6]	✗	✓	✗	✓	✗	✗	✗	✗	✗
Text2Room [17]	✓	✗	✓	✗	✗	✗	✓	✓	✗
NF-LDM [21]	✓	✓	✗	✓	✗	✗	✓	✗	✗
Ours	✓	✓	✓	✓	✓	✓	✓	✓	✓

**Table 5: User study with “Similar” options accounted. All methods are evaluated on the perceptual quality (PQ) of the imagery and scene diversity (SD). Here we only present pairwise comparison results between our system and baselines.**

Comparison	PQ ↑	SD ↑
LOR [37] / Ours	15.8% / 84.2%	10.34% / 89.7%
SceneDreamer [12] / Ours	36.8% / 63.2%	20.0% / 80.0%
Persistent Nature [6] / Ours	29.8% / 70.2%	21.3% / 78.7%
Text2Room [17] / Ours	27.2% / 72.8%	36.1% / 63.9%

## F Additional Results

In this section, we present additional qualitative comparisons in Fig. 9, Fig. 10, Fig. 11, and Fig. 12. As shown in Fig. 13 and Fig. 14, besides text prompts, our system allows users to achieve fine-grained control over the output by adding extra conditions such as scribbles, depth maps, semantic segmentation maps, Canny edge maps, Hough line maps, and HED maps.

## G User Study

To evaluate the performance of our system, we organize a user study involving 65 participants with diverse backgrounds and expertise in the field. The study is conducted using an online website designed specifically for this purpose. A screenshot of the website interface is shown in Fig. 15. Note that our user study is completely anonymous, and no personally identifiable data is collected from the participants. During the study, we present participants with synthesized videos generated by two different methods, labeled as “Method 1” and “Method 2”. To ensure fairness and eliminate bias, each time we randomly select two sets of three videos, where both sets consist of videos generated by randomly chosen methods, including LOR [37], Persistent Nature [6], SceneDreamer [12], Text2Room [17], and ours, rather than having one set generated exclusively by our system and the other set by another method. This prevents participants from guessing which results are generated by ours during the user study. Participants are asked to compare two key aspects of the videos: the perceptual quality of the imagery and scene diversity. Specifically, they are invited to choose the method that exhibits better perceptual quality and scene diversity or select

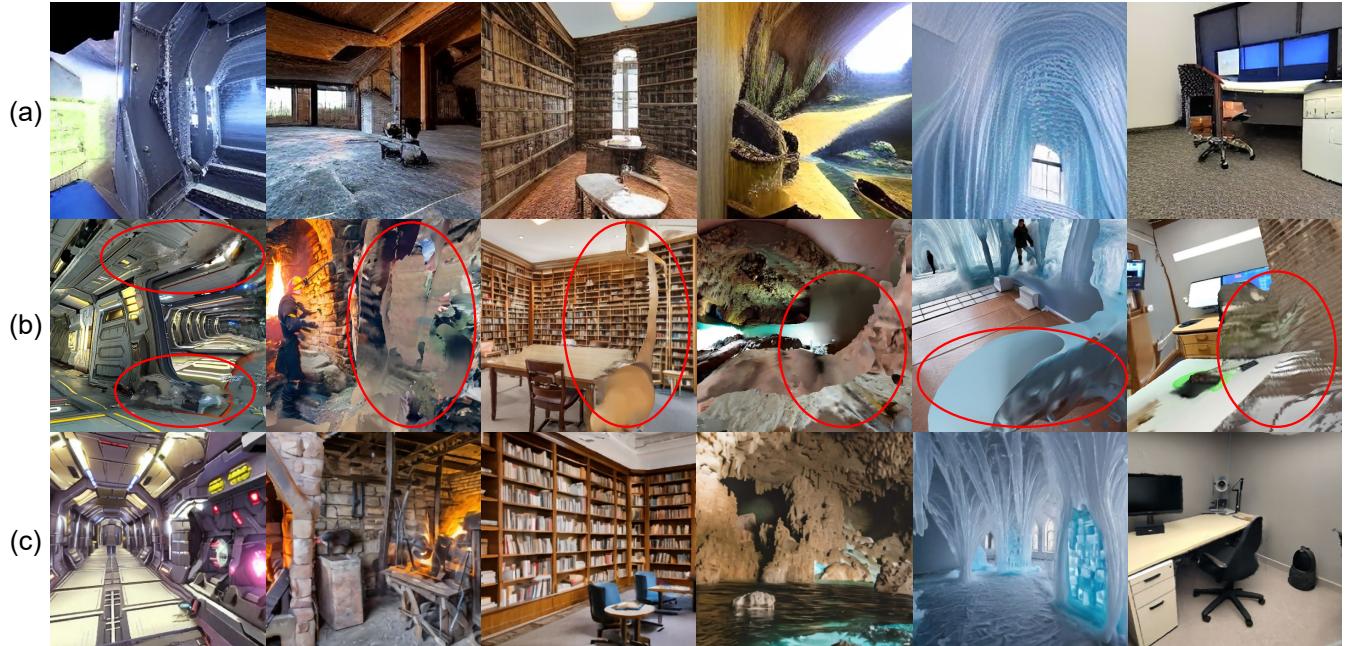
“Similar” if it is difficult to judge. We have a total of 65 participants, and we collect a substantial amount of data, 2142 data points in total. On average, each participant answered approximately 33 questions. In the user study of our main paper, we exclude data points with “Similar” options. As a reference, here we provide the version with “Similar” options accounted in Table 5.

## H Why 3D Meshes as an Intermediate Proxy?

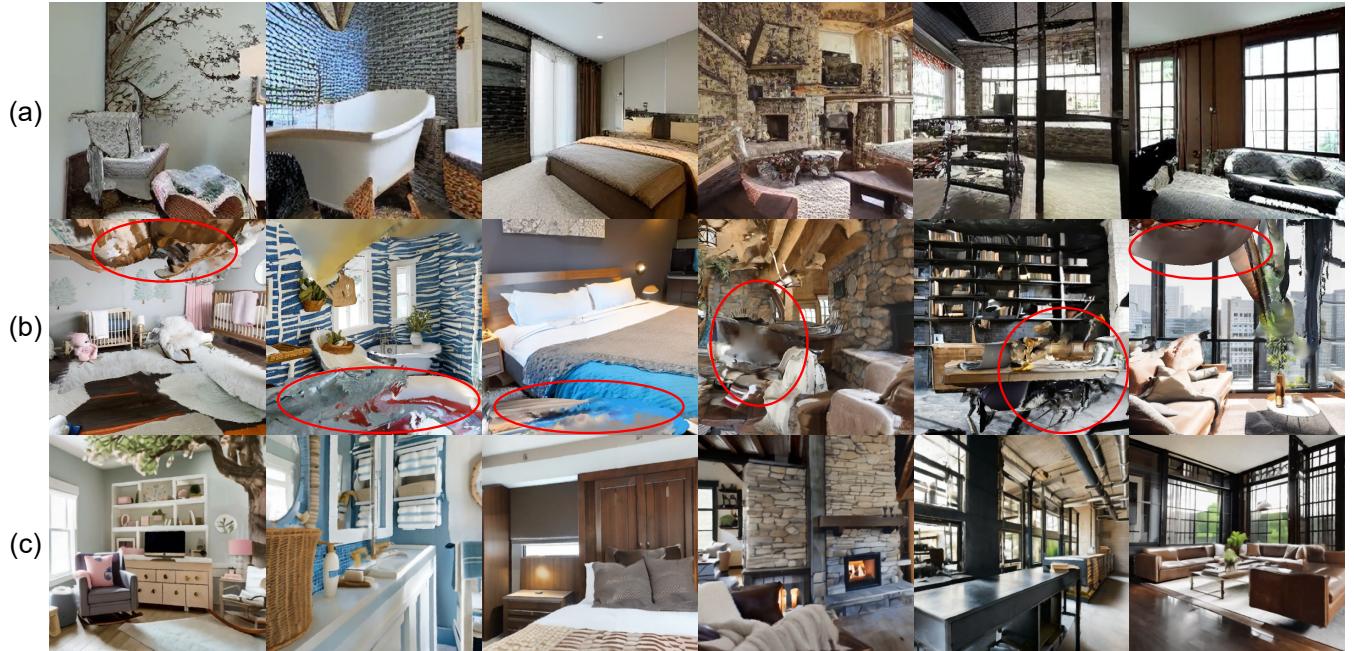
The primary reason for using meshes is that they provide an explicit representation that allows for the iterative build-up of the scene. On the other hand, point clouds are collections of individual points in 3D space, lacking structural information like connectivity and orientation between points. While they are useful for certain tasks like point cloud-based object recognition, they lack the explicit structure necessary for scene creation. Voxel grids divide the 3D space into small cubes or voxels, each representing a discrete volume element. While they offer a more straightforward representation for volumetric data, they often require high memory usage and can be less flexible in handling detailed geometry and shape variations. Implicit representations like NeRFs, are generally not well-suited because the underlying surface geometry is not explicitly represented, which makes it difficult to manipulate and edit the resulting 3D scene representation.

## I Limitation Discussion

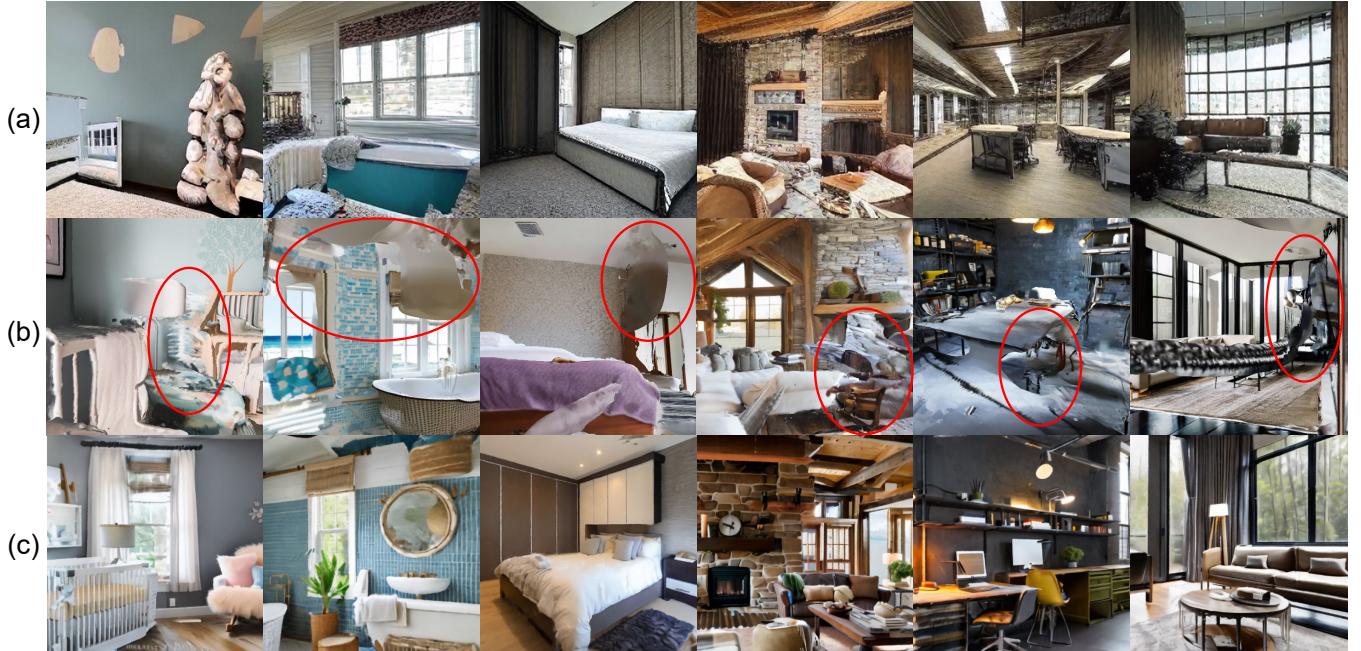
While our system provides a user-friendly platform for interactive 3D content creation, certain challenges can impact its performance. (a) The quality of the scene depends on how users create it. Our system offers users the freedom to create 3D scenes according to their will. However, this may be a double-edged sword. For example, if a user only chooses viewpoints in a “circular rotation” without changing the camera position, the generated scene might degenerate into a panorama. If a user selects suboptimal viewpoints, the generated scene might contain artifacts or fail to close the loop, i.e., create a complete scene, due to failure cases of depth alignment. In addition, a scene might remain incomplete, e.g., with holes, because parts of the scene are never viewed by the user; (b) The extent to which users can move the camera during the rendering process



**Figure 9: Additional qualitative comparison on indoor scenes.** Here we present the qualitative results of six indoor scenes, displayed alternately from left to right. The scenes, in sequence, are “spaceship”, “forge”, “library”, “cave”, “ice castle”, and “small office”. As can be seen, (a) LOR [37] is prone to domain drifting and a decline in output quality. Although (b) Text2Room [17] performs well on indoor scenes, it often produces over-smoothed regions and artifacts in the reconstructions. In contrast, (c) our system presents diverse and photo-realistic results.



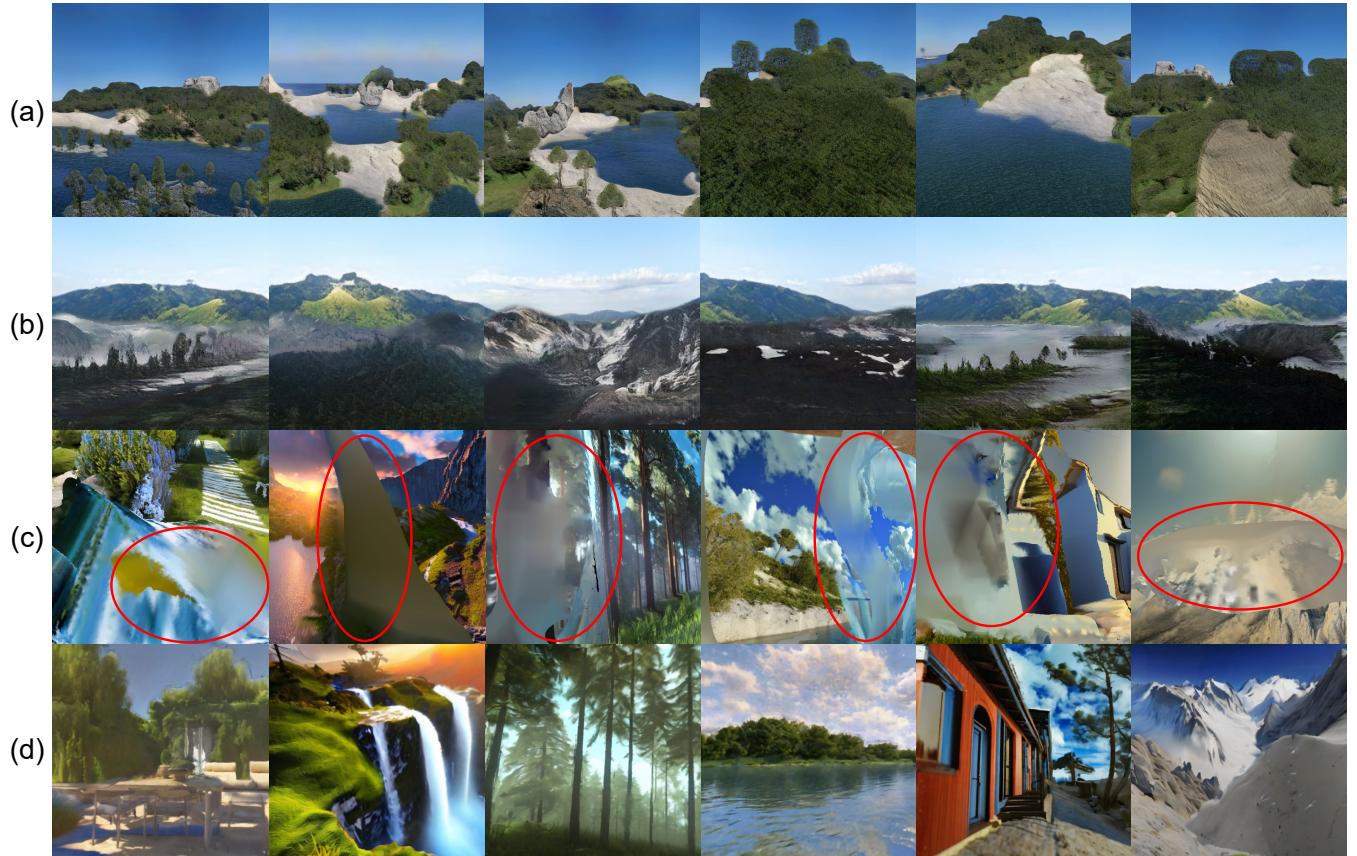
**Figure 10: Additional qualitative comparison on indoor scenes.** We present the qualitative results of another six indoor scenes, displayed alternately from left to right. The scenes, in sequence, are “baby room”, “bathroom”, “bedroom”, “farmhouse living room”, “large office”, and “modern living room”. (a) LOR [37], (b) Text2Room [17], and (c) ours.



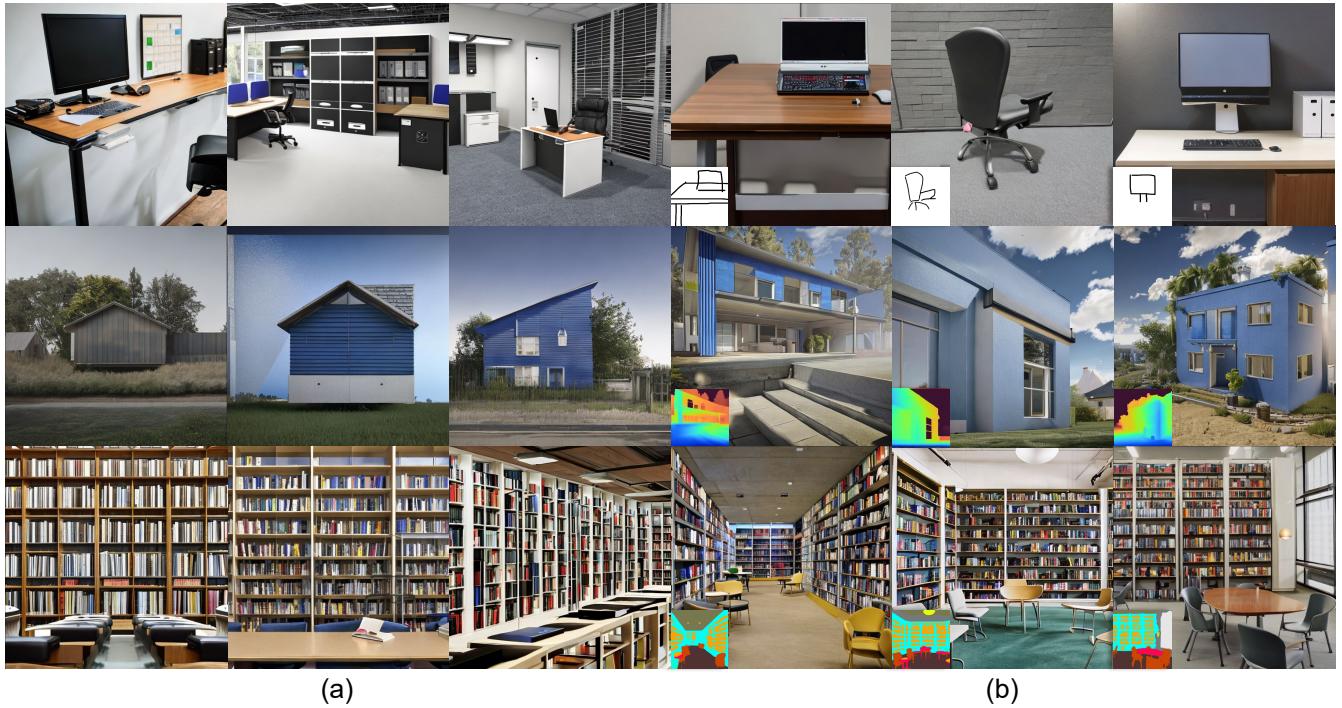
**Figure 11: Additional qualitative comparison on indoor scenes.** We present the qualitative results of another six indoor scenes, displayed alternately from left to right. The scenes, in sequence, are “baby room”, “bathroom”, “bedroom”, “farmhouse living room”, “large office”, and “modern living room”. (a) LOR [37], (b) Text2Room [17], and (c) ours.

depends on how users build their scenes. When users create their scenes, the camera’s motion can be varied significantly. If users continuously move the camera away from the world origin and progressively build the world, our system can generate videos with substantial camera motion beyond mere circular rotations. In such cases, the rendered videos showcase diverse perspectives and views. However, if users opt to only rotate the camera without changing its position, our system can still generate novel views, but the camera movement will be limited to rotational and slight positional changes. Nevertheless, it is essential to emphasize that our method is not limited to “circular rotation”. Users have the flexibility to customize their own generation trajectories, enabling a broader range

of camera motions. (c) A challenge arises when the depth prediction module produces inaccurate geometry based on the input image, or when the segmentation model fails to predict with precision. These issues can compromise the quality of the generated 3D scenes; (d) Distortions in the 3D meshes may contribute to inaccuracies and inconsistencies, ultimately affecting the overall realism and quality of the output. We leave them for our future work. However, we believe that the proposed system will empower users to unleash their creativity and may open up exciting possibilities for the field of 3D scene generation.



**Figure 12: Additional qualitative comparison on outdoor scenes.** We present the qualitative results of six outdoor scenes, displayed alternately from left to right. The scenes, in sequence, are “garden”, “waterfall”, “forest”, “river”, “house”, and “mountain”. Note that (a) SceneDreamer [12] and (b) Persistent Nature [6] require extensive training and are limited to a specific domain, i.e., landscapes. While (c) Text2Room [17] can also generate outdoor scenes, it suffers from notable mesh distortions and artifacts. By contrast, (d) our system can generate high-quality and consistent novel views across diverse domains.



**Figure 13: Fine-grained control.** From top to bottom, we sequentially show “small office”, “house”, and “library”. Compared to (a) current text-driven methods [15, 17], (b) our system can achieve fine-grained control over the output by adding extra conditions such as scribbles, depth, and semantic segmentation maps.



**Figure 14: Fine-grained control.** From top to bottom, we sequentially show “baby room”, “bedroom”, and “garden”. Compared to (a) current text-driven methods [15, 17], (b) our system can achieve fine-grained control over the output by adding extra conditions such as Canny edge maps, Hough line maps, and HED maps.

Method1 - Video1	Method1 - Video2	Method1 - Video3	<b>Rules</b>
Method2 - Video1	Method2 - Video2	Method2 - Video3	<ul style="list-style-type: none"> <li>Compare the generated videos of <b>Method 1</b> and <b>Method 2</b> (each method generates 3 videos) and answer the following 2 questions.</li> <li>If you want to end this test, you can click ‘Submit’. For data collection, you have to answer <b>at least 10 questions</b> before you click ‘Submit’.</li> <li>Please <b>do not refresh</b> the page! Or the data will be invalidated.</li> </ul>
			<b>Problem1:</b> Which is better for the perceptual quality of <b>Method1</b> and <b>Method2</b> ? <input type="radio"/> Method1 <input type="radio"/> Method2 <input checked="" type="radio"/> Similar <b>Problem2:</b> Which is better for the scene diversity of <b>Method1</b> and <b>Method2</b> ? <input type="radio"/> Method1 <input type="radio"/> Method2 <input checked="" type="radio"/> Similar
<input type="button" value="Next"/> <input type="button" value="Submit"/>			3 / 64 (Answer at least 10 questions)

**Figure 15: Interface of the user study website.**