

Supplementary Materials: iControl3D: An Interactive System for Controllable 3D Scene Generation

Anonymous Authors

For a comprehensive evaluation and demonstration of our system, please refer to our supplementary video. This document includes the following contents:

- (1) Contribution revisited.
- (2) Implementation details.
- (3) Baselines.
- (4) Experiment details.
- (5) Additional results.
- (6) Details of the user study.
- (7) Reasons of 3D meshes as an intermediate proxy.
- (8) Limitation discussion.

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 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 [10], 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 [12] 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 [1] 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 [8]. To segment remote content such as the sky, we first employ Grounding DINO [7] to detect remote regions based on text inputs. Then, we apply SAM [6] for precise segmentation.

Our 3D creator interface is based on an open-source project¹ 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 [11], 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

In our experiments, we primarily compare ours against four representative works including LOR [9], SceneDreamer [3], Persistent Nature [2], and Text2Room [5]. Specifically, LOR [9] 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 [3] and Persistent Nature [2] 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 [5] 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.

D EXPERIMENT DETAILS

In the main paper, we provide a comprehensive comparison with LOR [9], SceneDreamer [3], Persistent Nature [2], and Text2Room [5]. 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 [9] to generate indoor scenes. For SceneDreamer [3] and Persistent Nature [2], we only utilize them to generate the "mountain" scene. For ours and Text2Room [5], 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.

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

E ADDITIONAL RESULTS

In this section, we present additional qualitative comparisons in Fig. E.1, Fig. E.2, Fig. E.3, and Fig. E.4. As shown in Fig. E.5 and Fig. E.6, 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.

F 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. F.7. 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 [9], Persistent Nature [2], SceneDreamer [3], Text2Room [5], 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 F.1.

F.1 User Interview

To study how users react to using iControl3D, we randomly select five participants from the user study and invite them to test our system for 3D scene generation. After that, we conduct interviews with each participant to gather feedback on how well our system fulfills their preferences and requirements. We summarize their comments as follows:

- User-friendly interface.
- Customizable camera trajectories.
- The generation process is easy to control.
- The system supports a wide variety of scene types.
- The processing speed is deemed acceptable.
- May need to make multiple attempts or retries to achieve satisfactory results in creating their scenes.

G 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

Comparison	PQ \uparrow	SD \uparrow
LOR [9] / Ours	15.8% / 84.2%	10.34% / 89.7%
SceneDreamer [3] / Ours	36.8% / 63.2%	20.0% / 80.0%
Persistent Nature [2] / Ours	29.8% / 70.2%	21.3% / 78.7%
Text2Room [5] / Ours	27.2% / 72.8%	36.1% / 63.9%

Table F.1: 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.

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.

H 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 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

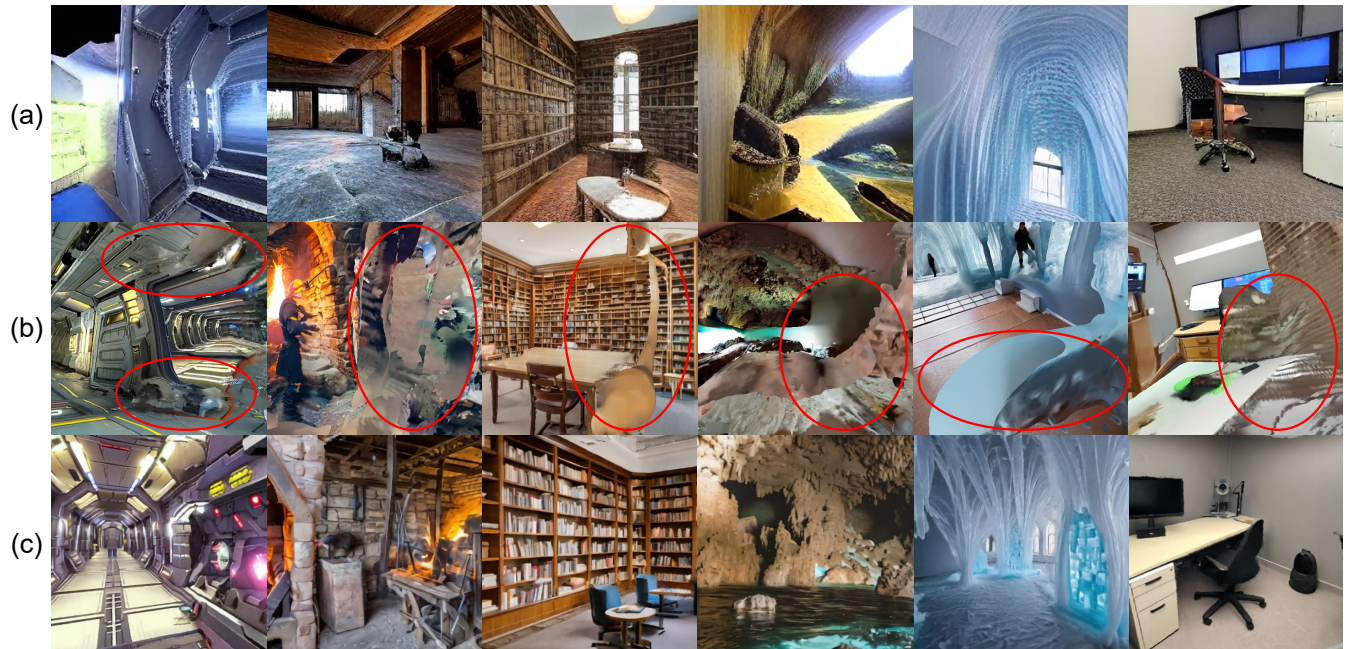


Figure E.1: Additional qualitative comparison on indoor scenes. Here we present the qualitative results of six indoor scenes, displayed alternately from left to right. The scenes, in sequence, are “spaceship”, “forge”, “library”, “cave”, “ice castle”, and “small office”. As can be seen, (a) LOR [9] is prone to domain drifting and a decline in output quality. Although (b) Text2Room [5] 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.

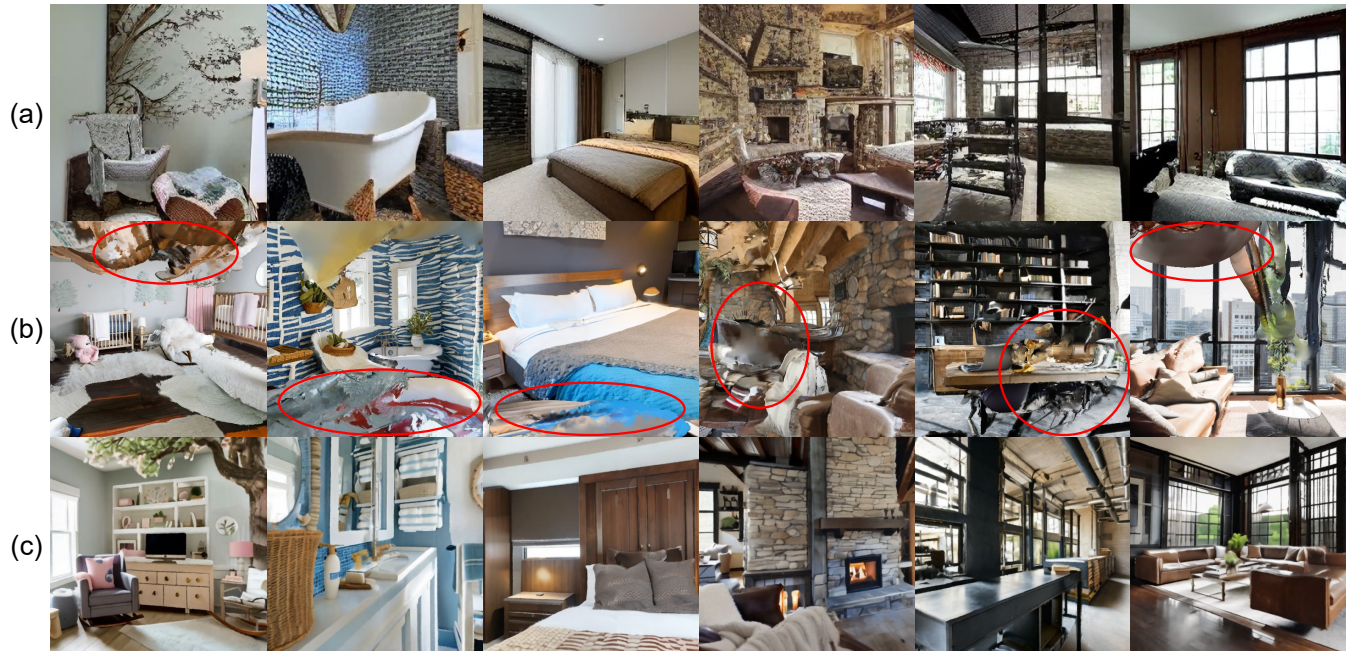


Figure E.2: Additional qualitative comparison on indoor scenes. We present the qualitative results of another six indoor scenes, displayed alternately from left to right. The scenes, in sequence, are “baby room”, “bathroom”, “bedroom”, “farmhouse living room”, “large office”, and “modern living room”. (a) LOR [9], (b) Text2Room [5], and (c) ours.

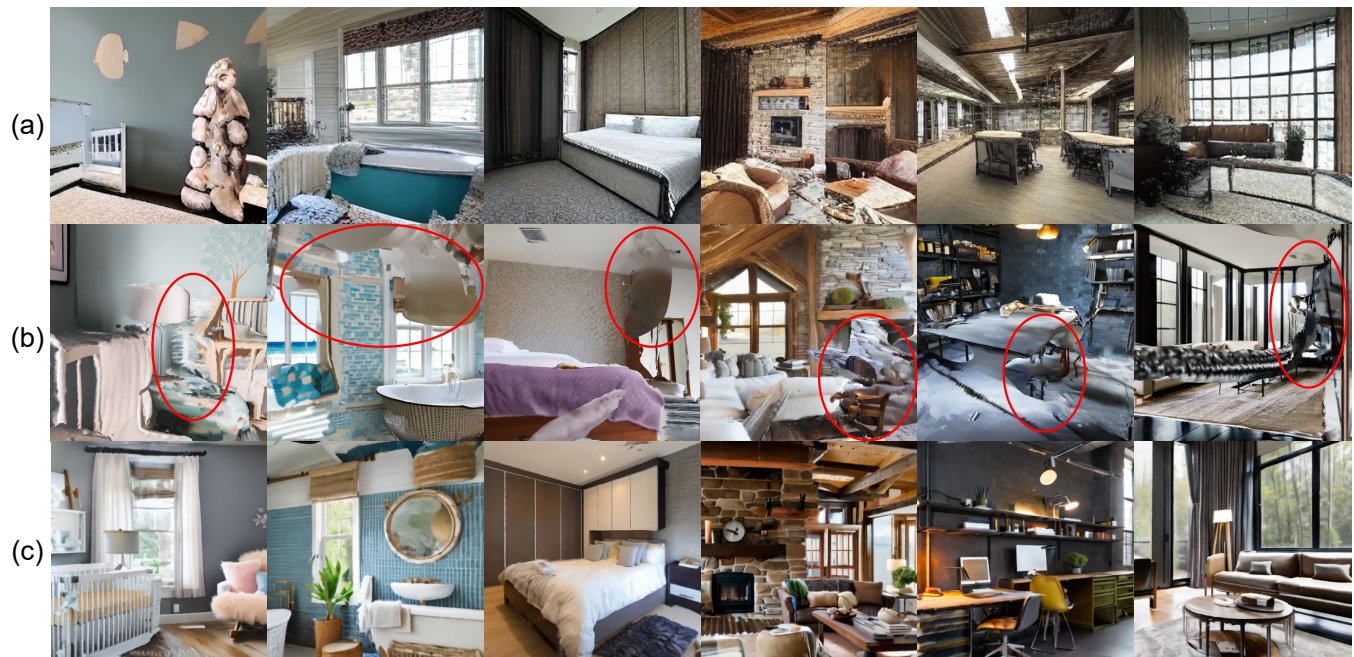


Figure E.3: Continuation of the qualitative comparison on indoor scenes. We present the qualitative results of another six indoor scenes, displayed alternately from left to right. The scenes, in sequence, are “baby room”, “bathroom”, “bedroom”, “farmhouse living room”, “large office”, and “modern living room”. (a) LOR [9], (b) Text2Room [5], and (c) ours.

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.

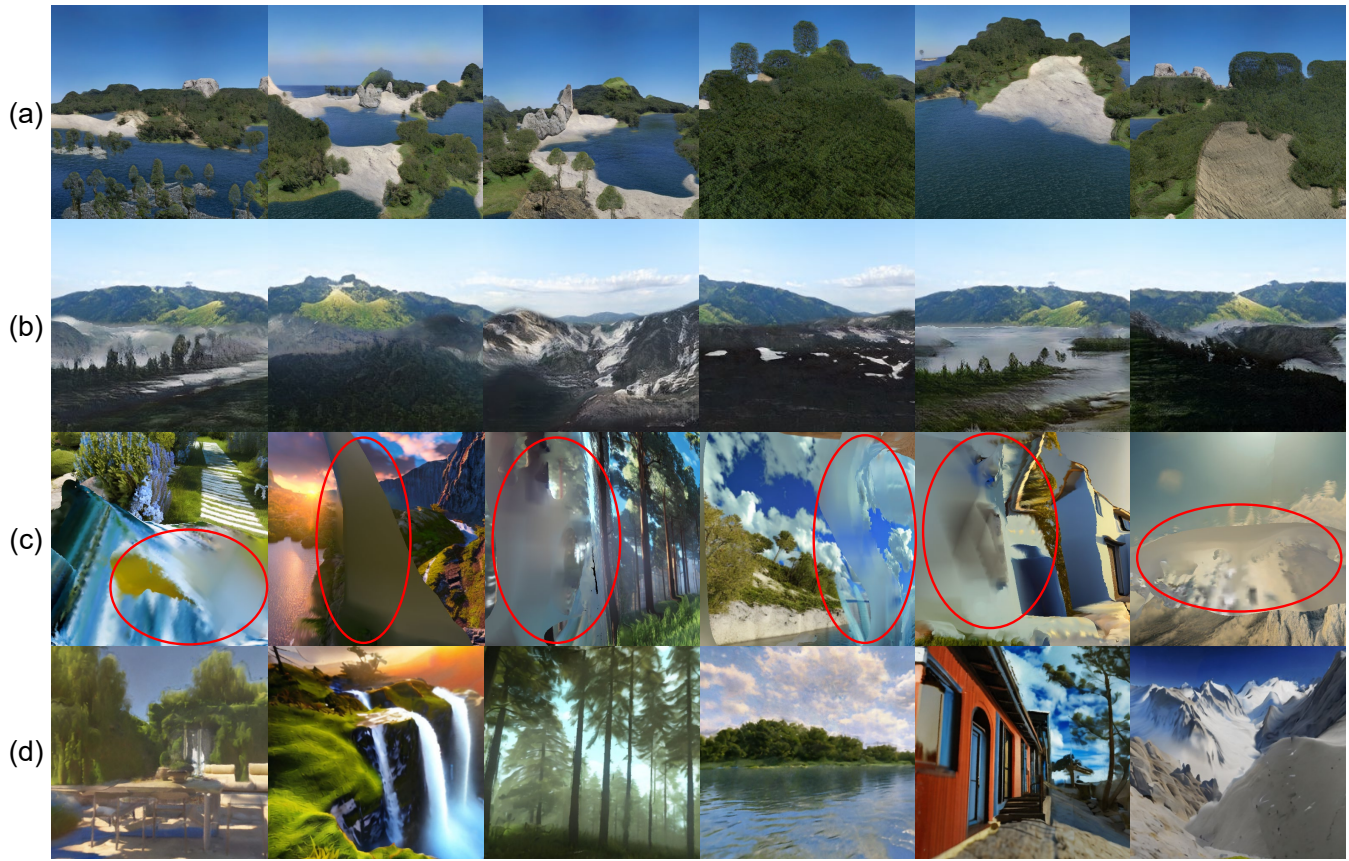


Figure E.4: Additional qualitative comparison on outdoor scenes. We present the qualitative results of six outdoor scenes, displayed alternately from left to right. The scenes, in sequence, are “garden”, “waterfall”, “forest”, “river”, “house”, and “mountain”. Note that (a) SceneDreamer [3] and (b) Persistent Nature [2] require extensive training and are limited to a specific domain, i.e., landscapes. While (c) Text2Room [5] 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 E.5: Fine-grained control. From top to bottom, we sequentially show “small office”, “house”, and “library”. Compared to (a) current text-driven methods [4, 5], (b) our system can achieve fine-grained control over the output by adding extra conditions such as scribbles, depth, and semantic segmentation maps.



Figure E.6: Fine-grained control. From top to bottom, we sequentially show “baby room”, “bedroom”, and “garden”. Compared to (a) current text-driven methods [4, 5], (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.

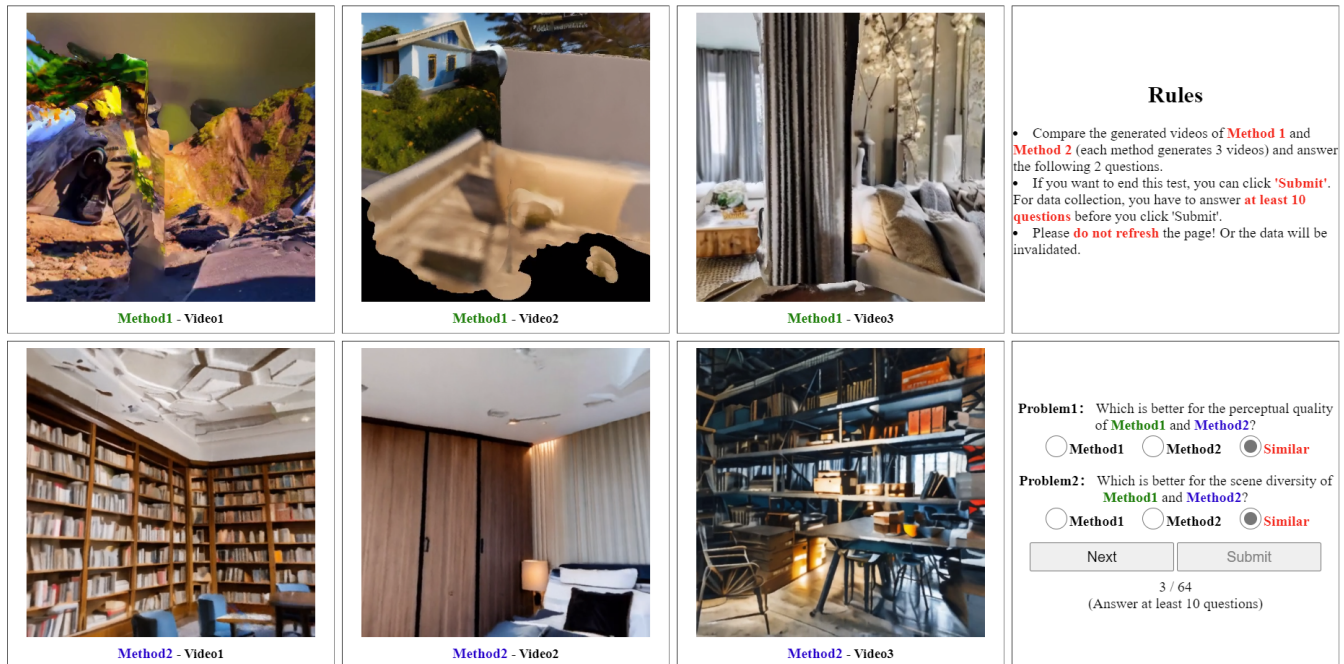


Figure F.7: Interface of the user study website.

REFERENCES

[1] Shariq Farooq Bhat, Reiner Birkel, Diana Wofk, Peter Wonka, and Matthias Müller. 2023. Zoedepth: Zero-shot transfer by combining relative and metric depth. *arXiv preprint arXiv:2302.12288* (2023).

[2] 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).

[3] Zhaoxi Chen, Guangcong Wang, and Ziwei Liu. 2023. Scenedreamer: Unbounded 3d scene generation from 2d image collections. *arXiv preprint arXiv:2302.01330* (2023).

[4] Rafail Fridman, Amit Abecasis, Yoni Kasten, and Tali Dekel. 2023. Scenescape: Text-driven consistent scene generation. *arXiv preprint arXiv:2302.01133* (2023).

[5] Lukas Höllein, 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).

[6] Alexander Kirillov, Eric 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).

[7] 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).

[8] 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).

[9] 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.

[10] 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.

[11] 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).

[12] Lvmin Zhang and Maneesh Agrawala. 2023. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543* (2023).

871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928