WonderJourney: Going from Anywhere to Everywhere

Hong-Xing Yu[1] Haoyi Duan[1] Junhwa Hur[2] Kyle Sargent[1] Michael Rubinstein[2]
William T. Freeman[2] Forrester Cole[2] Deqing Sun[2] Noah Snavely[2] Jiajun Wu[1] Charles Herrmann[2]

[1]Stanford University [2]Google Research

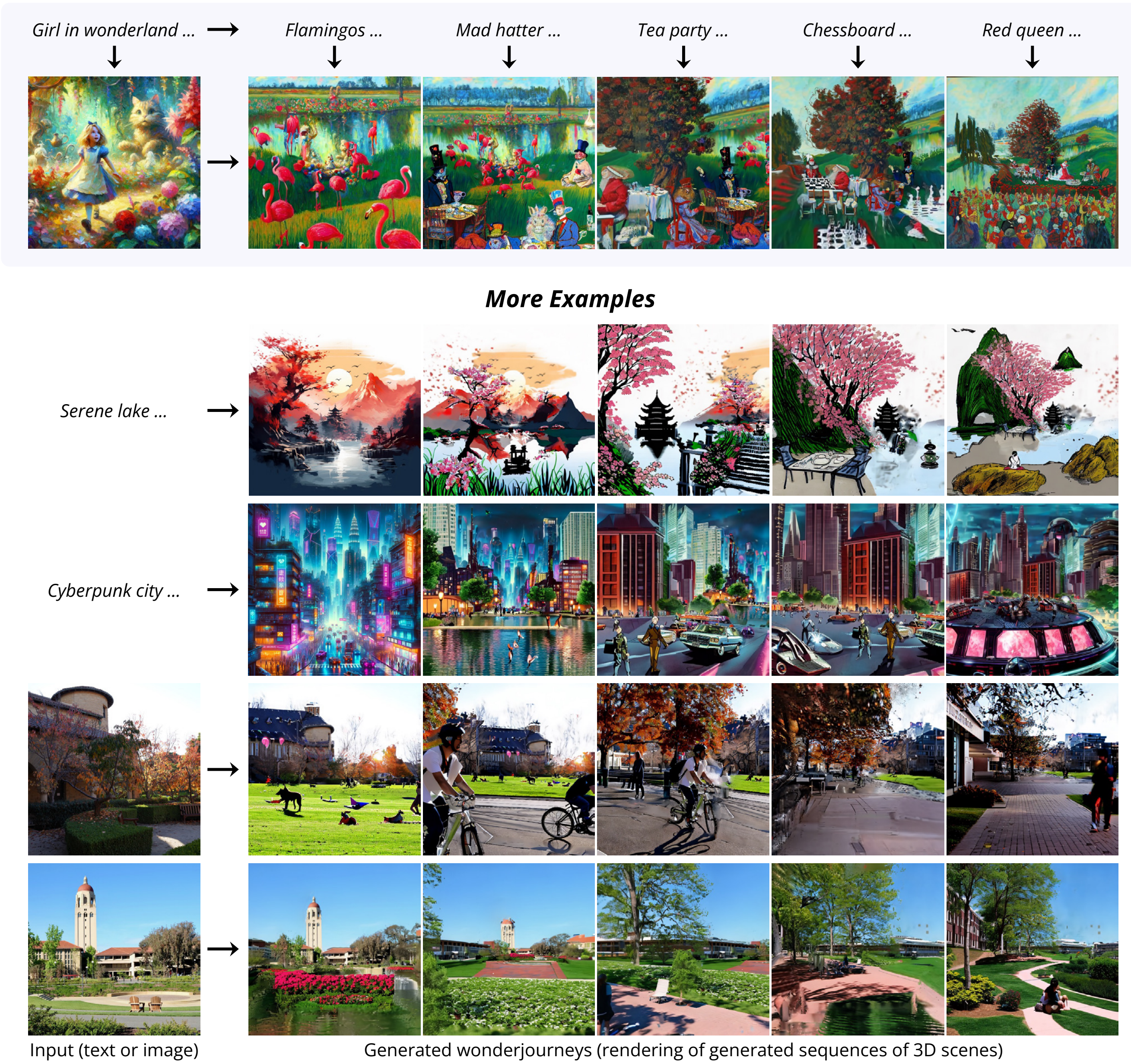

Figure 1. We propose **WonderJourney**—generating a sequence of diverse yet coherent 3D scenes from a text description or an arbitrary image such as photos or generated art ("from anywhere"). WonderJourney can generate various journeys (which we refer to as "wonderjourneys") for a fixed input, potentially ending "everywhere" (Fig. 4). We show rendered images along the generated sequence of 3D scenes. **We strongly encourage the reader to see video examples at `https://kovenyu.com/WonderJourney/`.**

Abstract

We introduce WonderJourney, a modular framework for perpetual 3D scene generation. Unlike prior work on view generation that focuses on a single type of scenes, we start at any user-provided location (by a text description or an image), and generate a journey through a long sequence of diverse yet coherently connected 3D scenes. We leverage an LLM to generate textual descriptions of the scenes in this journey, a text-driven point cloud generation pipeline to make a compelling and coherent sequence of 3D scenes, and a large VLM to verify the generated scenes. We show compelling, diverse visual results across various scene types and styles, forming imaginary "wonderjourneys". Project website: https://kovenyu.com/WonderJourney/.

"No, no! The adventures first, explanations take such a dreadful time." – *Alice's Adventures in Wonderland*

1. Introduction

In "Alice's Adventures in Wonderland", the story begins with Alice falling down the rabbit hole and emerging into a strange and captivating Wonderland. In her journey through this wonderland, Alice encounters many curious characters such as the Cheshire Cat and the Mad Hatter, and peculiar scenarios, such as the tea party and the rose garden – eventually ending at the royal palace. These characters and settings combine to form a compelling world that has fascinated countless readers over the years. In this paper, we follow in this creative tradition and explore how modern computer vision and AI models can similarly generate such interesting and varied visual worlds, which users can journey through, much like Alice did in her own adventures in Wonderland.

Toward this goal, we introduce the problem of *perpetual 3D scene generation*. Unlike previous work on perpetual view generation [21, 24] that only generates a single type of scene, such as nature photos, our objective is to synthesize a series of diverse 3D scenes starting at an arbitrary location specified via a single image or language description. The generated 3D scenes should be *coherently connected* along a long-range camera trajectory, traveling through various plausible places. The generated 3D scenes allow rendering a fly-through video through a long series of diverse scenes to simulate the visual experience of a journey in an imaginary "wonderland". We show examples in Fig. 1.

The main challenges of perpetual 3D scene generation center around generating *diverse yet plausible* scene elements. These scene elements need to support the formation of a path through coherently connected 3D scenes. They should include various objects, backgrounds, and layouts that fit in observed scenes and transit naturally to the next, yet unobserved, scenes. This generation process can be broken down into determining what objects to generate for a given scene; where to generate these objects; and how these scenes connect to each other geometrically. Determining what elements to generate calls for semantic understanding of a scene (e.g., a lion might not be a great fit for a kitchen); determining where to generate them calls for common sense regarding the visual world (e.g., a lion should not be floating in the sky); further, generating these elements in a new connected scene requires geometric understanding (e.g., occlusion and disocclusion, parallax, and appropriate spatial layouts). Notably, these challenges are absent in prior work on perpetual view generation, as they tend to focus on a single domain [24] or a single scene [11].

Our proposed solution, WonderJourney, addresses each of the above challenges in perpetual 3D scene generation with its own module. WonderJourney leverages an LLM to generate a long series of scene descriptions, followed by a text-driven visual scene generation module to produce a series of colored point clouds to represent the connected 3D scenes. Here, the LLM provides commonsense and semantic reasoning; the vision module provides visual and geometric understanding and the appropriate 3D effects. In addition, we leverage a vision-language model (VLM) to verify the generation and launch a re-generation when it detects undesired visual effects. Our framework is modular. Each of the modules can be implemented by the best pretrained models, allowing us to leverage the latest and future advancements in the rapidly growing vision and language research. Our main contributions are as follows:

- We study perpetual 3D scene generation and propose WonderJourney, a modular framework that decomposes this problem into core components: an LLM to generate scene descriptions, a text-driven visual module to generate the coherent 3D scenes, and a VLM to verify the generated scenes.
- We design a visual scene generation module that leverages off-the-shelf text-to-image and depth estimation models to generate coherent 3D point clouds. Our module handles boundary depth inaccuracy, sky depth inaccuracy, and disocclusion unawareness.
- We show compelling visual results and compare WonderJourney with SceneScape [11] and InfiniteNature-Zero [21] in a user study, which shows that WonderJourney produces interesting and varied journeys.

2. Related Work

Perpetual view generation. The seminal work on perpetual view generation is Infinite Images [18] which synthesizes the effect of navigating a 3D world by stitching and rendering images according to camera motions. Later works, such as Infinite Nature [24] and InfiniteNature-Zero [21], learn to auto-regressively generate next view based on the current view. Follow-up works improve global 3D consistency [5] and visual quality [4]. A recent work, SceneScape [11],

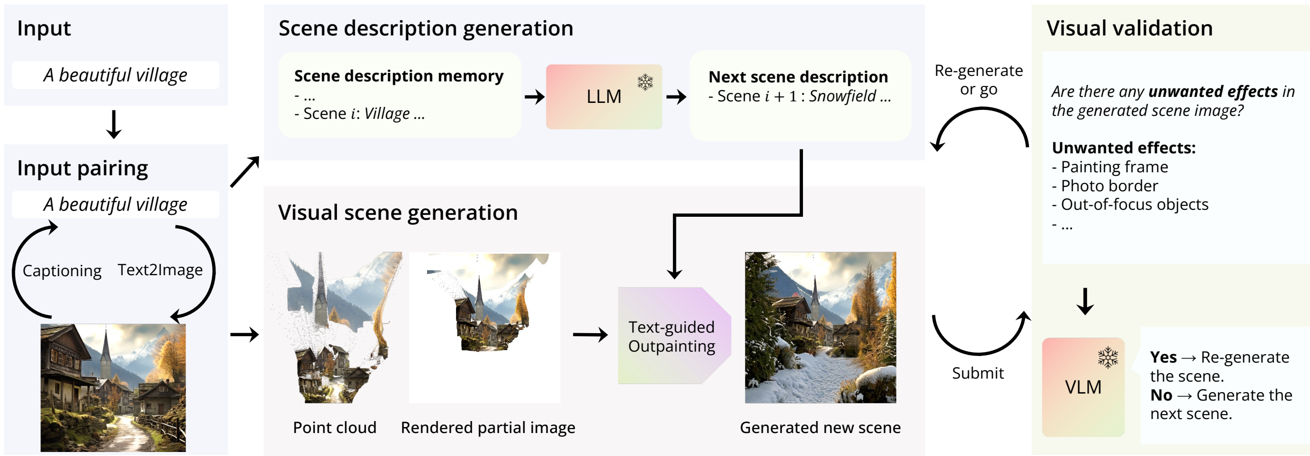

Figure 2. **The proposed WonderJourney framework and workflow across modules**. Our modular design does not require any training, allowing easy future improvements from the quick advances in vision and language models.

explores text-driven perpetual view generation by gradually constructing a single cave-like scene represented by a lengthy mesh. While these methods study perpetual generation, they are restricted to a single domain such as nature photos [21, 24] or a single scene [11].

3D scene generation. Considerable progress has recently been made in text-to-3D or image-to-3D generation, many of which focus on objects without background [9, 20, 22, 26, 27, 33, 34]. These works typically leverage a 2D image prior from an image generation model, e.g., an image diffusion model [36], and then build a 3D representation, such as a NeRF [42], by distilling the supervision of the 2D image generation model [33]. Other works on 3D object generation focus on learning a 3D generative model directly from 2D images [6, 7, 13, 14, 29, 30, 32, 37, 39].

Several works also focus on generating a single 3D scene with background [2, 10]. For example, Text2Room [15] generates a room-scale 3D scene from a single text prompt, using textured 3D meshes for their scene representation. Other relevant works have focused on generating (sometimes called "reconstructing") a scene from limited observations, such as a single image. Long-Term Photometric Consistent NVS [43] generates single scenes from a source image by auto-regressively generating with a conditional diffusion model. GeNVS [8] and Diffusion with Forward Models [40] use an intermediate 3D representation but are trained and evaluated on each scene separately. ZeroNVS [38] synthesizes a NeRF of a scene from a single image. Their work focuses on generating a single scene while ours targets generating a coherently connected sequence of diverse scenes.

Text-guided video generation. The idea of scene generation has also been explored in video generation. Several concurrent works like TaleCrafter [12] and others [16, 23, 25] also discuss the task of creating a series of videos which follow an LLM-generated story or script. However, all these works use different scenes as different clips in a video, resulting in hard cuts, while our perpetual 3D scene generation aims at generating sequences of coherently connected scenes.

3. Approach

Our goal is to generate a potentially endless sequence of diverse yet coherently connected 3D scenes, which requires both geometric understanding of 3D scenes and visual common sense and semantic understanding of scene structures. To tackle this challenging task, we propose WonderJourney. The main idea is to generate a text description of the visual elements the next scene would contain, and then employ a text-guided visual generation module to make that 3D scene.

WonderJourney is a modular framework that decomposes this task into scene description generation, visual scene generation, and visual validation, as in Fig. 2. Given an input image or text, we first pair it with the other modality by generating the image with a text-to-image model or by generating the description with a Vision Language Model (VLM). Then we generate the next-scene description by a Large Language Model (LLM). A visual scene generation module takes in the next-scene description and the current scene image to generate the next 3D scene represented by a colored point cloud. This generation process is then checked by a VLM to make sure there is no unwanted effects, or it gets regenerated. We note that our framework is modular such that each module can be implemented with the latest (pretrained) models and thus can easily leverage the quick advances of large language and vision models.

3.1. Scene description generation

We propose an auto-regressive scene description generation process, i.e., the scene description generation takes a set of past and current scene descriptions as input and predicts the subsequent scene description:

$$\mathcal{S}_{i+1} = g_{\text{description}}(\mathcal{J}, \mathcal{M}_i), \quad (1)$$

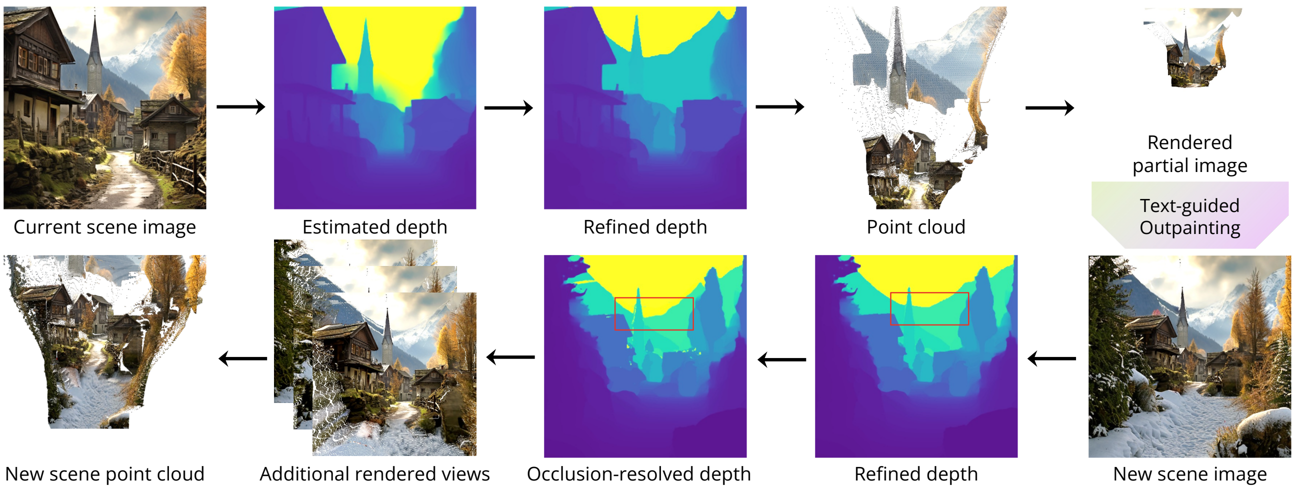

Figure 3. **The visual scene generation module**. Each arrow represents a parametric vision model (e.g., a depth estimator) or an operation (e.g., rendering). Our fully modular design easily benefits from advances in the corresponding research topics.

where $\mathcal{S}_i$ denotes the i^{th} scene description, and $g_{\text{description}}$ denotes the scene description generator which is implemented by an LLM that takes two inputs: 1) the task specification $\mathcal{J}$ = *"You are an intelligent scene generator. Please generate 3 most common entities in the next scene, along with a brief background description."* ; and 2) the scene description memory $\mathcal{M}_i = \{\mathcal{S}_0, \mathcal{S}_1, \cdots, \mathcal{S}_i\}$ which is a collection of past and current scenes. The latest description memory is:

$$\mathcal{M}_{i+1} = \mathcal{M}_i \cup \{\mathcal{S}_{i+1}\}. \tag{2}$$

We define the scene description $\mathcal{S}_i = \{S, O_i, B_i\}$, which consists of a style S that is kept consistent across scenes, objects in the scene O_i, and a concise caption B_i describing the background of the scene. We allow the LLM to output natural language descriptions, and then use a lexical category filter to process the raw text of O_i and B_i such that we only keep nouns for entities and adjectives for attributes. Empirically this generates more coherently connected scenes compared to requiring the LLM to directly output this structured description.

3.2. Visual scene generation

Since we want the generated next scene to be coherent with past scenes geometrically and semantically, we formulate our visual scene generation as a conditional generation problem, taking both the next-scene description and the 3D representation of the current scene as conditions:

$$\mathcal{P}_{i+1} = g_{\text{visual}}(I_i, \mathcal{S}_{i+1}), \tag{3}$$

where $\mathcal{P}_i$ denotes a colored point cloud that represents the next 3D scene, and I_i denotes the image of current scene. The visual scene generator g_{visual} consists of learning-free operations such as perspective unprojection and rendering, as well as components that use parametric (pretrained) vision models, including a depth estimator, a segmentation-based depth refiner, and a text-conditioned image outpainter. We show an illustration in Fig. 3.

Lifting image to point cloud. Given the current scene represented by an image I_i, we lift it to 3D by estimating depth and unproject it with a pinhole camera model. We use MIDAS v3.1 [35], one of the state-of-the-art depth estimators, in our experiments. However, we find that existing monocular depth estimators share two common issues. First, depth discontinuity is not well modeled, witnessed by previous work [1, 28, 41], resulting in overly smooth depth edges across object boundaries. Second, the depth of the sky is always underestimated, also observed by previous work [21, 24]. To address these two issues, we introduce a depth refinement process that leverages pixel grouping segments and sky segmentation.

Depth refinement. To enhance the depth discontinuity across object boundaries, we model scene elements with frontal planes when the elements have a limited disparity range. We use SAM [19] to generate pixel grouping segments $\{\texttt{seg}_j\}_{j=1}^{N_s}$ where $\texttt{seg}_j \in \{0,1\}^{H\times W}$ is a segment mask, sorted in descending order according to the size of the segment $\|\texttt{seg}_j\|$. We iteratively refine the estimated depth:

$$\texttt{depth}[\texttt{seg}_j] \leftarrow \begin{cases} \texttt{median}(\texttt{depth}[\texttt{seg}_j]), & \text{if } \Delta D_j < T, \\ \texttt{depth}[\texttt{seg}_j], & \text{otherwise}, \end{cases} \tag{4}$$

for $j = 1, \cdots, N_s$, where $\texttt{depth} \in \mathbb{R}_+^{H\times W}$ is initialized with the estimated monocular depth, $\texttt{median}(\cdot)$ is a function that returns the median value of the input set, $\Delta D_j = \texttt{max}(\texttt{disparity}[\texttt{seg}_j]) - \texttt{min}(\texttt{disparity}[\texttt{seg}_j])$ denotes the disparity (the reciprocal of $\texttt{depth}$) range within the segment $\texttt{seg}_j$. We keep the estimated depth of segments with high disparity range as they do not fit to a frontal-plane, such as roads. Note that the idea of frontal-plane modeling

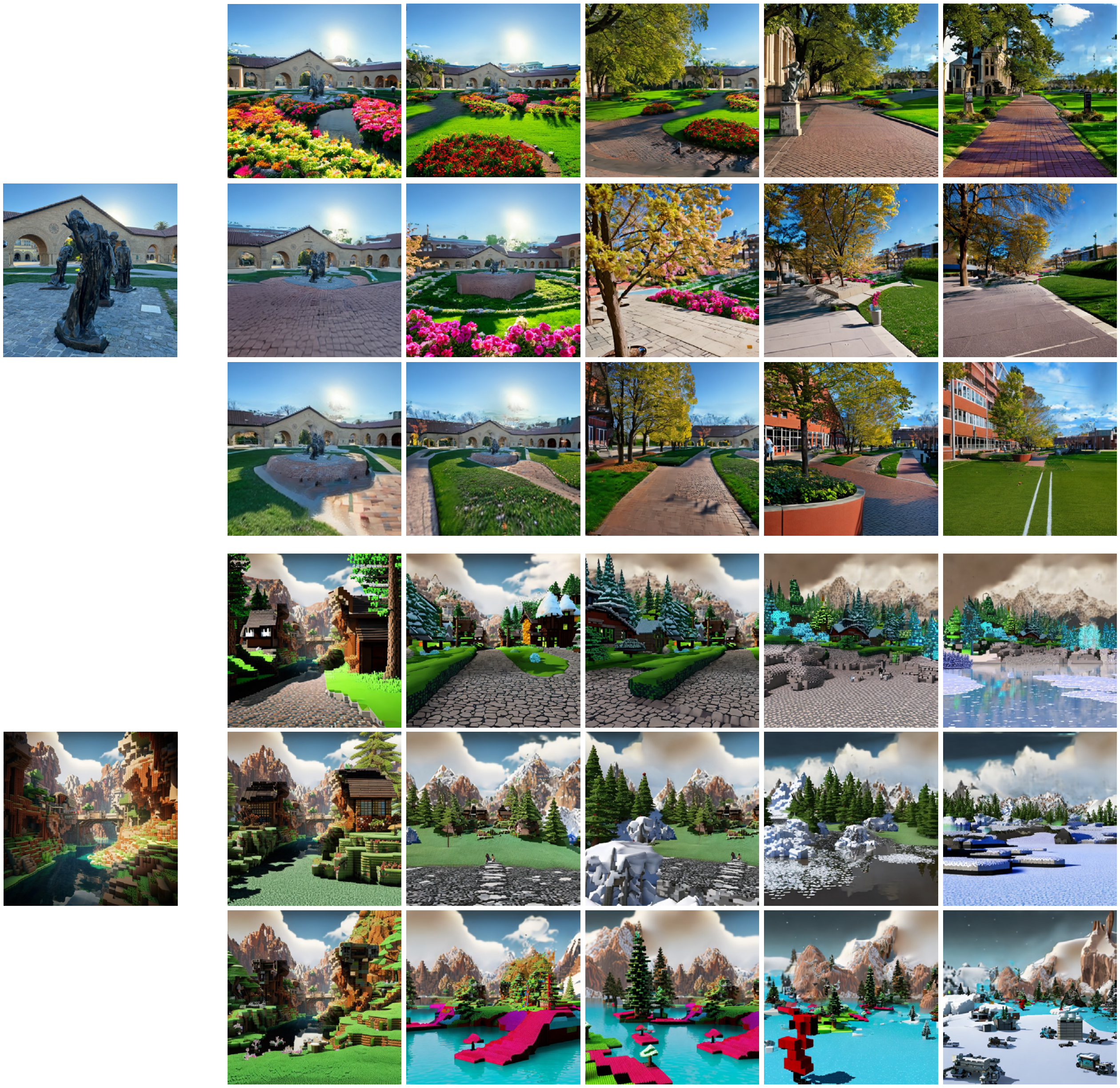

Figure 4. **Qualitative results** for diverse journeys generated from the same input image, showing that WonderJourney can go everywhere. The input in the top example is a real photo.

has also been explored in 3D Ken-Burns [31] with selected semantic classes such as car and people. In contrast, as we target at general scenes with diverse styles, we use the criterion of the disparity range for keeping estimated depth instead of selected semantic classes.

To handle the sky depth which is always underestimated, we use OneFormer [17] to segment sky region and assign a high depth value to it. However, this results in inaccurate depth estimates along the sky boundary; if we were to naively use the output segmentation, these errors result in accumulated severe artifacts in later scenes. To resolve it, we simply remove points along the sky boundary. Besides, we find that depth at distant pixels are generally not reliable. Thus, we also set a far background plane with depth F that cuts off all pixels' depth beyond it.

Description-guided scene generation. To generate a new scene that is connected to the current scene, we place a camera C_{i+1} with an appropriate distance to the current camera C_i. As shown in Fig. 3, we render the partial image $\hat{I}_{i+1}$ (more details on the camera and the renderer are in Appendix E) and outpaint it with a text-guided outpainter to generate a new scene image I_{i+1}:

$$I_{i+1} = g_{\text{outpaint}}(\hat{I}_{i+1}, \mathcal{S}_{i+1}), \tag{5}$$

where we use the Stable Diffusion model [36] for g_{outpaint} in our experiments. Note, we purposefully place the new camera at a location that creates enough empty space in the rendered image. We empirically find that text-guided outpainters tend to avoid generating partial objects, likely due to their curated training image datasets, which tend to not include truncated or partial objects. Leaving too little empty space therefore results in just a simple extrapolation of the partial image $\hat{I}_{i+1}$ and a lack of adherence to the text prompt $\mathcal{S}_{i+1}$, especially in regards to new objects. After generating the new scene image, we lift it to 3D by estimating and refining depth for it, and we obtain the new point cloud $\hat{\mathcal{P}}_{i+1} = \mathcal{P}_i \cup \mathcal{P}'_{i+1}$ where $\mathcal{P}'_{i+1}$ denotes the additional points from unprojecting the outpainted pixels.

New scene registration by depth consistency. However, as the depth estimator is unaware of geometry constraints, the depth for points in $\mathcal{P}'$ generally do not align with $\mathcal{P}_i$. Thus, we adapt the depth estimator by a depth alignment loss:

$$\mathcal{L}_{\text{depth}} = \max(0, \mathcal{D}^*_{\text{bg}} - \mathcal{D}'_{\text{bg}}) + \|\mathcal{D}^*_{\text{fg}} - \mathcal{D}'_{\text{fg}}\|, \tag{6}$$

where $\mathcal{D}^*_{\text{bg}}$ denotes the analytically computed depth of background pixels from I_i, $\mathcal{D}'_{\text{bg}}$ denotes the estimated depth for pixels corresponding to $\mathcal{D}^*_{\text{bg}}$, $\mathcal{D}^*_{\text{fg}}$ denotes the computed depth of all other visible pixels from I_i, and $\mathcal{D}'_{\text{fg}}$ denotes the estimated depth for pixels corresponding to $\mathcal{D}^*_{\text{fg}}$.

Occlusion handling by re-rendering consistency. Another geometric inconsistency is that disocclusion regions can have a lower depth values than their occluders, as the depth estimator is not aware of this 3D geometric constraint. We highlight the wrongly estimated disocclusion depth in the refined depth in Fig. 3. To resolve this issue, we re-render the new scene $\hat{\mathcal{P}}_{i+1}$ at the camera C_i and detect all inconsistent pixels. At each inconsistent pixel, we move back all the rasterized additional points from $\mathcal{P}'_{i+1}$ that have lower depth values than the one point from $\mathcal{P}_i$. This removes the disocclusion inconsistency and guarantees that the disocclusion comes after the occluder.

Scene completion. We obtain the final point cloud $\mathcal{P}_{i+1}$ by adding more points to $\hat{\mathcal{P}}_{i+1}$. We add points by repeating the following "complete-as-you-go" process: we place an additional camera along a camera trajectory connecting the new scene to the current scene, render a partial image at that camera, outpaint the image, and add the additional points to the point cloud. Note that in our visual scene generation formulation in Equation 3, one can replace the image input I_i with the point cloud $\mathcal{P}_i$ from the current scene, forming a persistent scene representation. This allows a trade-off between 3D persistence and empirical requirements. In practice, maintaining a large point cloud leads to prohibitively many points that require a large amount of GPU memory when generating a long trajectory of high-resolution scenes. Thus in our experiments we take the image formulation.

3.3. Visual validation

Empirically, in a large portion of generated photos and paintings, a painting frame or a photo border appears, disrupting the geometry consistency. Additionally, there are often unwanted blurry out-of-focus objects near the borders of the generated images. Thus, we propose a validation step to identify and reject these undesired generated scenes.

We formulate this as a text-based detection problem, where our objective is to detect a set of predefined undesirable effects in the generated scene image. We reject and regenerate the scene image if any unwanted effect is detected. Specifically, right after we generate a new scene image I_{i+1}, we immediately feed it to a VLM and prompt it with the query $\mathcal{J}^t_{\text{detect}}$ = *"Is there any X_t in this image?"* where $X_t \in \{X_1, \cdots, X_T\}$ is an unwanted effect specified by text, such as "photo border", "painting frame", or "out-of-focus objects". If any unwanted effect is detected, we regenerate I_{i+1} with a new description $\mathcal{S}_{i+1}$ or a new random seed.

4. Experiments

Dataset and baselines. Since the perpetual 3D scene generation is a new task without an existing dataset, we use a mixture of photos taken by ourselves, copyright-free photos from online, and generated examples, for evaluation in our experiments. We perform the pairing process by DALL-E 3 [3] for text-to-image pairing. We consider two state-of-the-art perpetual view generation methods as our baseline: image-based InfiniteNature-Zero and text-based SceneScape.

Qualitative demonstration. We show qualitative examples of the generated journey across different scenes and different styles in Fig. 1 and Fig. 5. These results show that WonderJourney is able to generate diverse yet coherently connected scenes from various types of input images, i.e., it can go from anywhere. We show more examples in the Appendix. We further show examples of diverse generation samples from the same input in Fig. 4. These diverse generated journeys suggest that WonderJourney supports going to different destinations at each run.

Additional evaluations. We show additional qualitative results in Appendix B, longer "wonderjourneys" (up to 30 scenes) in Appendix C, controlled "wonderjourneys" (i.e., using user-provided full text, such as poems and haiku, instead of LLM-generated text guidance) in Appendix D, and ablation studies in Appendix F.

Human preference evaluation. Since a main application of WonderJourney is for creative and entertainment purposes, we focus on human preference evaluation as our quanti-

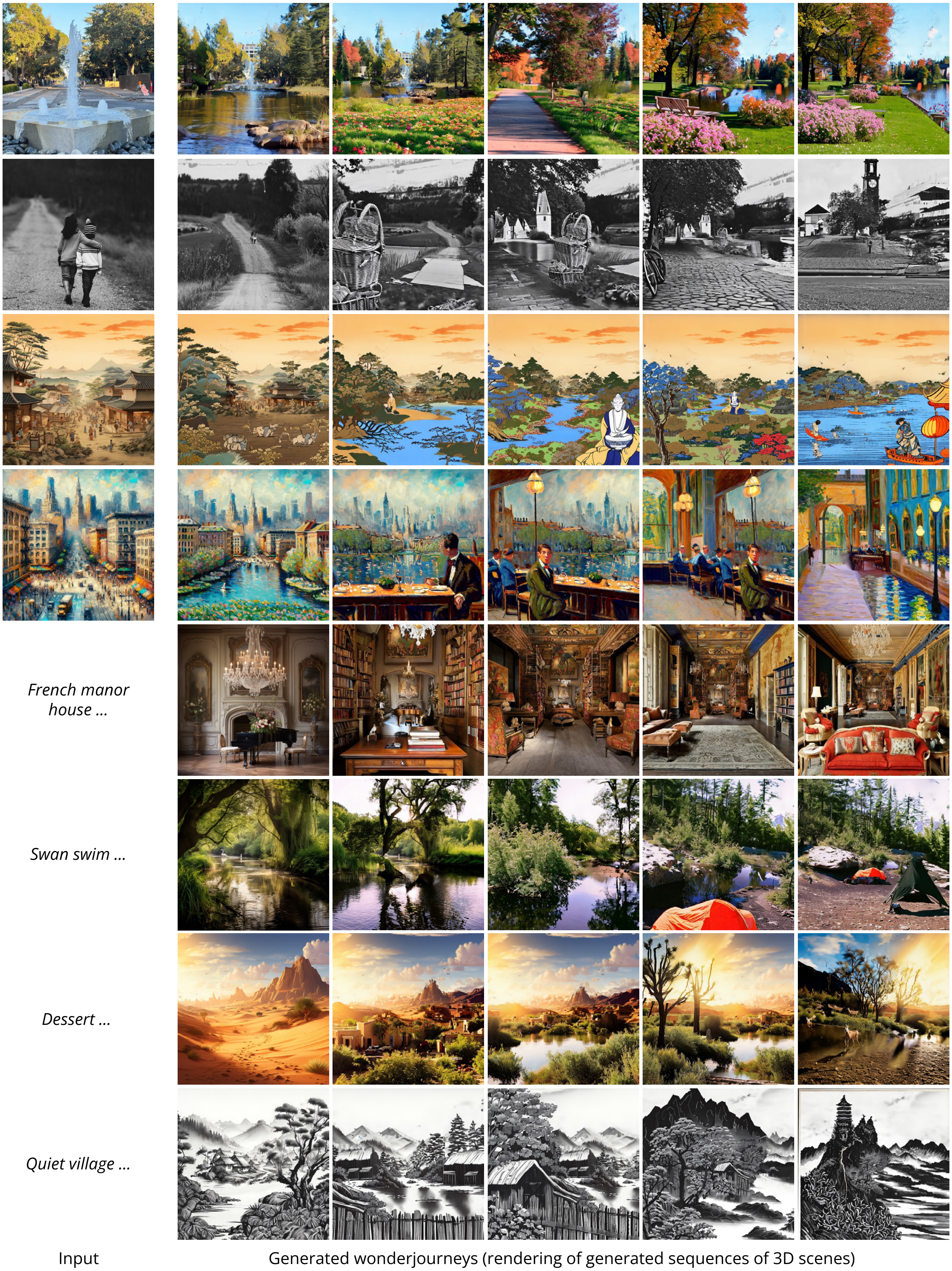

Figure 5. **From diverse starting scenes with different styles**, WonderJourney generates a sequence of diverse yet coherent 3D scenes, showing that it can go from anywhere to everywhere (e.g., nature, village, city, indoor, or fantasy). **The inputs in top two rows are real photos. We strongly encourage the reader to see the video results in the project website.**

Figure 6. **Comparison with InfiniteNature-Zero [21] and SceneScape [11].** Note that InfiniteNature-Zero is trained on nature photos, so we only compare with it using photorealistic nature images as input.

Table 1. Human preference of ours over baseline on diversity, visual quality, scene complexity, and overall interesting-ness.

	Div.	Qual.	Compl.	Overall
Ours over InfiniteNature-Zero	92.7%	94.9%	91.5%	88.6%
Ours over SceneScape	88.8%	83.4%	80.0%	90.3%

tative metrics, using the following four axes: diversity of generated scenes in a single journey, visual quality, scene complexity, and overall interesting-ness. We generate videos following each approach's own camera trajectories setup. Since InfiniteNature-Zero is trained on nature photos, we only compare to it using photorealistic nature images. For SceneScape, since it is text-based, we can use 3 examples of different styles for comparison. We show a side-by-side comparison of videos generated by WonderJourney and a baseline with randomized positions. We then ask one binary-choice question at a time. such as *"Compare the two videos below. Which video has a higher **diversity**? That is, which video shows more various different places?"*. We recruited 400 participants, 200 for the comparison with InfiniteNature-Zero and 200 for SceneScape. Each participant answers 12 questions. We provide more details in Appendix H.

As shown in Table 1, WonderJourney is strongly preferred over both baselines on all four axes. InfiniteNature-Zero synthesizes only nature scenes, as shown in Fig. 6, while WonderJourney generates more diverse scenes and objects such as mountaineers and houses that are naturally connected to the starting nature scene. SceneScape tends to generate cave-like scenes due to the usage of a textured mesh, and thus all examples converges to caves. Also, as discussed in Section 3.2, SceneScape tends to not generate new objects due to limited white space. All these factors might contribute to the much greater user preference for WonderJourney.

5. Conclusion

We introduce WonderJourney to generate a long sequence of diverse yet coherently connected 3D scenes starting at any user provided location. WonderJourney achieves compelling, diverse visual results across various scene types and different styles, enabling users to journey through their own adventures in the generated "wonderjourneys".

Acknowledgments. This work was supported by NSF RI #2211258, ONR N00014-23-1-2355, and ONR YIP N00014-24-1-2117. The work was done in part when Hong-Xing Yu was a student researcher at Google and has been supported by gift funding and GCP credits from Google.

References

[1] Filippo Aleotti, Fabio Tosi, Pierluigi Zama Ramirez, Matteo Poggi, Samuele Salti, Stefano Mattoccia, and Luigi Di Stefano. Neural disparity refinement for arbitrary resolution stereo. In *2021 International Conference on 3D Vision (3DV)*, pages 207–217. IEEE, 2021. 4

[2] Miguel Angel Bautista, Pengsheng Guo, Samira Abnar, Walter Talbott, Alexander Toshev, Zhuoyuan Chen, Laurent Dinh, Shuangfei Zhai, Hanlin Goh, Daniel Ulbricht, et al. Gaudi: A neural architect for immersive 3d scene generation. *Advances in Neural Information Processing Systems*, 35:25102–25116, 2022. 3

[3] James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, Wesam Manassra, Prafulla Dhariwal, Casey Chu, Yunxin Jiao, and Aditya Ramesh. Improving image generation with better captions. *Technical report*, 2023. 6

[4] Shengqu Cai, Eric Ryan Chan, Songyou Peng, Mohamad Shahbazi, Anton Obukhov, Luc Van Gool, and Gordon Wetzstein. DiffDreamer: Towards consistent unsupervised single-view scene extrapolation with conditional diffusion models. In *ICCV*, 2023. 2

[5] Lucy Chai, Richard Tucker, Zhengqi Li, Phillip Isola, and Noah Snavely. Persistent nature: A generative model of unbounded 3d worlds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20863–20874, 2023. 2

[6] Eric R Chan, Marco Monteiro, Petr Kellnhofer, Jiajun Wu, and Gordon Wetzstein. 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*, pages 5799–5809, 2021. 3

[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. Efficient geometry-aware 3d generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16123–16133, 2022. 3

[8] 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. Genvs: Generative novel view synthesis with 3d-aware diffusion models, 2023. 3

[9] Yen-Chi Cheng, Hsin-Ying Lee, Sergey Tulyakov, Alexander G Schwing, and Liang-Yan Gui. Sdfusion: Multimodal 3d shape completion, reconstruction, and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4456–4465, 2023. 3

[10] Terrance DeVries, Miguel Angel Bautista, Nitish Srivastava, Graham W. Taylor, and Joshua M. Susskind. Unconstrained scene generation with locally conditioned radiance fields. In *ICCV*, 2021. 3

[11] Rafail Fridman, Amit Abecasis, Yoni Kasten, and Tali Dekel. Scenescape: Text-driven consistent scene generation. *arXiv preprint arXiv:2302.01133*, 2023. 2, 3, 8

[12] Yuan Gong, Youxin Pang, Xiaodong Cun, Menghan Xia, Haoxin Chen, Longyue Wang, Yong Zhang, Xintao Wang, Ying Shan, and Yujiu Yang. Talecrafter: Interactive story visualization with multiple characters. *arXiv preprint arXiv:2305.18247*, 2023. 3

[13] Jiatao Gu, Lingjie Liu, Peng Wang, and Christian Theobalt. Stylenerf: A style-based 3d-aware generator for high-resolution image synthesis. *arXiv preprint arXiv:2110.08985*, 2021. 3

[14] Zekun Hao, Arun Mallya, Serge Belongie, and Ming-Yu Liu. Gancraft: Unsupervised 3d neural rendering of minecraft worlds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14072–14082, 2021. 3

[15] Lukas Höllein, Ang Cao, Andrew Owens, Justin Johnson, and Matthias Nießner. Text2room: Extracting textured 3d meshes from 2d text-to-image models. *arXiv preprint arXiv:2303.11989*, 2023. 3

[16] Hanzhuo Huang, Yufan Feng, Cheng Shi, Lan Xu, Jingyi Yu, and Sibei Yang. Free-bloom: Zero-shot text-to-video generator with llm director and ldm animator. *arXiv preprint arXiv:2309.14494*, 2023. 3

[17] Jitesh Jain, Jiachen Li, MangTik Chiu, Ali Hassani, Nikita Orlov, and Humphrey Shi. OneFormer: One Transformer to Rule Universal Image Segmentation. In *CVPR*, 2023. 5

[18] Biliana Kaneva, Josef Sivic, Antonio Torralba, Shai Avidan, and William T Freeman. Infinite images: Creating and exploring a large photorealistic virtual space. *Proceedings of the IEEE*, 98(8):1391–1407, 2010. 2

[19] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. Segment anything. In *ICCV*, pages 4015–4026, 2023. 4

[20] Gang Li, Heliang Zheng, Chaoyue Wang, Chang Li, Changwen Zheng, and Dacheng Tao. 3ddesigner: Towards photorealistic 3d object generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2211.14108*, 2022. 3

[21] Zhengqi Li, Qianqian Wang, Noah Snavely, and Angjoo Kanazawa. Infinitenature-zero: Learning perpetual view generation of natural scenes from single images. In *European Conference on Computer Vision*, pages 515–534. Springer, 2022. 2, 3, 4, 8

[22] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 300–309, 2023. 3

[23] Han Lin, Abhay Zala, Jaemin Cho, and Mohit Bansal. Videodirectorgpt: Consistent multi-scene video generation via llm-guided planning. *arXiv preprint arXiv:2309.15091*, 2023. 3

[24] Andrew Liu, Richard Tucker, Varun Jampani, Ameesh Makadia, Noah Snavely, and Angjoo Kanazawa. Infinite nature: Perpetual view generation of natural scenes from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14458–14467, 2021. 2, 3, 4

[25] Chang Liu, Haoning Wu, Yujie Zhong, Xiaoyun Zhang, and Weidi Xie. Intelligent grimm–open-ended visual storytelling

via latent diffusion models. *arXiv preprint arXiv:2306.00973*, 2023. 3

[26] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9298–9309, 2023. 3

[27] Luke Melas-Kyriazi, Iro Laina, Christian Rupprecht, and Andrea Vedaldi. Realfusion: 360deg reconstruction of any object from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8446–8455, 2023. 3

[28] S Mahdi H Miangoleh, Sebastian Dille, Long Mai, Sylvain Paris, and Yagiz Aksoy. Boosting monocular depth estimation models to high-resolution via content-adaptive multi-resolution merging. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9685–9694, 2021. 4

[29] Thu Nguyen-Phuoc, Chuan Li, Lucas Theis, Christian Richardt, and Yong-Liang Yang. Hologan: Unsupervised learning of 3d representations from natural images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7588–7597, 2019. 3

[30] Michael Niemeyer and Andreas Geiger. Giraffe: Representing scenes as compositional generative neural feature fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11453–11464, 2021. 3

[31] Simon Niklaus, Long Mai, Jimei Yang, and Feng Liu. 3d ken burns effect from a single image. *ACM Transactions on Graphics (ToG)*, 38(6):1–15, 2019. 5

[32] Roy Or-El, Xuan Luo, Mengyi Shan, Eli Shechtman, Jeong Joon Park, and Ira Kemelmacher-Shlizerman. Stylesdf: High-resolution 3d-consistent image and geometry generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13503–13513, 2022. 3

[33] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022. 3

[34] Amit Raj, Srinivas Kaza, Ben Poole, Michael Niemeyer, Nataniel Ruiz, Ben Mildenhall, Shiran Zada, Kfir Aberman, Michael Rubinstein, Jonathan Barron, et al. Dreambooth3d: Subject-driven text-to-3d generation. *arXiv preprint arXiv:2303.13508*, 2023. 3

[35] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(3), 2022. 4, 1

[36] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 3, 6

[37] Kyle Sargent, Jing Yu Koh, Han Zhang, Huiwen Chang, Charles Herrmann, Pratul Srinivasan, Jiajun Wu, and Deqing Sun. Vq3d: Learning a 3d-aware generative model on imagenet. *arXiv preprint arXiv:2302.06833*, 2023. 3

[38] Kyle Sargent, Zizhang Li, Tanmay Shah, Charles Herrmann, Hong-Xing Yu, Yunzhi Zhang, Eric Ryan Chan, Dmitry Lagun, Li Fei-Fei, Deqing Sun, et al. Zeronvs: Zero-shot 360-degree view synthesis from a single real image. *arXiv preprint arXiv:2310.17994*, 2023. 3

[39] Katja Schwarz, Axel Sauer, Michael Niemeyer, Yiyi Liao, and Andreas Geiger. Voxgraf: Fast 3d-aware image synthesis with sparse voxel grids. *Advances in Neural Information Processing Systems*, 35:33999–34011, 2022. 3

[40] Ayush Tewari, Tianwei Yin, George Cazenavette, Semon Rezchikov, Joshua B Tenenbaum, Frédo Durand, William T Freeman, and Vincent Sitzmann. Diffusion with forward models: Solving stochastic inverse problems without direct supervision. *arXiv preprint arXiv:2306.11719*, 2023. 3

[41] Fabio Tosi, Yiyi Liao, Carolin Schmitt, and Andreas Geiger. Smd-nets: Stereo mixture density networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8942–8952, 2021. 4

[42] Zirui Wang, Shangzhe Wu, Weidi Xie, Min Chen, and Victor Adrian Prisacariu. Nerf–: Neural radiance fields without known camera parameters. *arXiv preprint arXiv:2102.07064*, 2021. 3

[43] Jason J Yu, Fereshteh Forghani, Konstantinos G Derpanis, and Marcus A Brubaker. Long-term photometric consistent novel view synthesis with diffusion models. *arXiv preprint arXiv:2304.10700*, 2023. 3