

Invisible Stitch: Generating Smooth 3D Scenes with Depth Inpainting (Supplementary Materials)

Anonymous 3DV submission

Paper ID 387

1. Overview

The supplementary material is organized as follows: First, we provide an extended discussion on the limitations of the CLIP score evaluation protocol for 3D scene generation. We explicitly show that it is insensitive to geometric inconsistencies, motivating the need for our SSG-3D benchmark (Sec. 2). Second, we provide further details for our SSG-3D benchmark and specific choices of hyperparameters for the fine-tuning process of DIne. We also detail how we replaced the depth prediction and fusion components of existing methods (for the experiments in Section 5.3.1), the specifics of our minimal pipeline, and an exhaustive ablation study of DIne (Sec. 3). Third, we provide further qualitative results produced by our scene generation pipeline, including a new set of scenes for different real-world images (Sec. 4). Finally, we briefly discuss the limitations of DIne that will hopefully inspire future research to further this field (Sec. 5).

We invite the reader to consider the accompanying scenes and videos that are part of the supplementary materials. The scenes¹ were created with our minimal pipeline (see Section 5.3.2), `dog-DIne.ply` using DIne and `dog-zoedepth-aligned.ply` with ZoeDepth, where the predicted depth is aligned with the existing scene with a global scale-and-shift optimization (similar to existing methods). Please be aware that we reduced the output resolution of the generated scenes due to file size constraints. The videos show renderings of further 3D scenes generated with DIne.

We publish the code to train our depth completion model, the trained checkpoint, as well as our minimal pipeline to generate 3D scenes.

¹The PLY files may be opened with any viewer that supports Gaussian splat scenes. Please note that we directly export the point cloud and only use this file format for convenience. No Gaussian splat optimization is performed. We suggest using the SuperSplat viewer: <https://playcanvas.com/supersplat/editor/>.

2. CLIP Score & Geometric Inconsistencies

In Section 5.3, we demonstrate that DIne can extend scenes more faithfully than existing depth prediction-and-fusion components. However, as discussed in Section 5.3.1, *the CLIP score does not reflect these improvements*. This aligns with our observation in Figure 3, where replacing the depth prediction-and-fusion component in LucidDreamer improves geometry but not the CLIP score.

In Table S1, we further demonstrate the inability of the CLIP score to evaluate the *structural* quality of a generated 3D scene through a controlled experiment. Specifically, we show that geometric inconsistencies in a 3D scene cannot be captured by the CLIP score.

To this end, we utilize an intentionally misaligned scene. If the CLIP score is sensitive to the geometric qualities of a scene, we expect it to decrease with increasing misalignment.

For this experiment, we build a scene by starting from the *kyoto* input image seen in Figure 5 and project it with a depth prediction from DIne. Then, we render a novel view point V_i , extend the scene with Stable Diffusion, and use DIne again, conditioned on the existing depth, to obtain a depth map $D \in \mathbb{R}^{H \times W}$. Here, we introduce our intervention, offsetting D with increasingly larger values to create a misalignment between the starting image and the newly generated frame. Now, from a different novel view point V_j , we obtain a rendering of the scene where the misalignment is clearly visible and query Stable Diffusion to outpaint the scene. This image is then evaluated with the CLIP score, mirroring the evaluation protocol for scene generation methods.

The results of this experiment are shown in Table S1. First, we render the view V_j , clearly showing this purposefully introduced geometric inconsistency. Second, we present the outpainting result of Stable Diffusion, and evaluate it with the CLIP score. Third, we show the point cloud of the generated scene.

We find that the CLIP score remains quite stable despite the presence of major misalignment. While both the 3D

scene and the rendered views show visible breaks, the views inpainted by Stable Diffusion appear to sufficiently *counter-act* these breaks, thus the CLIP score remains unaffected. However, we note that inpainting only improves the 2D visual quality while concealing—not fixing—any flaws in the 3D scene. Consequently, we deduce that this protocol is not suitable to assess the geometric qualities of scene generation methods.

Our proposed SSG-3D benchmark fills this void, providing a rigorous tool to assess the geometric qualities of the depth prediction and fusion components in scene generation methods.

3. Further Implementation Details

In the following, we provide a more extensive of description of the implementation details for our proposed SSG-3D benchmark (Section 3.1), the procedure to fine-tune DINE from ZoeDepth [3] (Section 3.2), and how we replaced the depth prediction and fusion components of existing methods (Section 3.3). We also provide an exhaustive ablation of our training procedure for DINE (Section 3.4).

3.1. SSG-3D

Scenes in ScanNet are described by highly-overlapping sequential frames. Thus, we chunk them into blocks of 50 frames and consider the first and tenth frames in each block for our evaluation. This allows us to yield ample views from each scene and maintains diversity while limiting the number of evaluations to run. As the sequential frames are naturally highly overlapping, we refrain from setting a specific threshold τ . To maintain reasonable evaluation times, we only consider the first 50 scenes, which yields a total of 7,832 view pairs.

With Hypersim, we compute ϕ across *all* views of a single camera trajectory within *each* scene and set $\tau := 0.8$. We exclude scenes rendered with non-standard projection matrices². The resulting number of view pairs that we evaluate on is 19,243.

3.2. DINE Fine-Tuning

We base our model on ZoeDepth, which uses a dense prediction transformer (DPT) [10] with a BeiT (Bidirectional Encoder representation from Image Transformers) [2] backbone at a resolution of 512×384 . We fine-tune the model for 5 epochs with batch size 8, using a low learning rate of 0.00025 with a weight decay of 0.01. We train on four NVIDIA Tesla P40 GPUs.

3.3. Drop-In Replacements

In the following, we briefly describe how we replaced the depth prediction and fusion components of existing meth-

²See <https://github.com/apple/ml-hypersim/issues/24>

ods with DINE, as presented in Section 5.3.1.

WonderJourney. This method uses the depth estimation model MiDaS v3.1 [11], using a global scale-and-shift operation to align the depth, and fine-tuning the model to further improve the alignment. Further, segments discovered by Segment Anything [8] are grouped if they have similar disparity. Further, the sky is separated.

We replaced all of these individual steps with a single prediction of DINE, directly attaching the predicted frame to the existing scene.

LucidDreamer. Here, ZoeDepth [3] is utilized to predict depth for newly generated frames. A global-scale-and shift operation is run for coarse alignment. Then, to eliminate seams, the connecting edges of a new frame receive depth values of the existing scene, which are then extrapolated to eventually match the predicted depth (post-alignment).

This entire procedure is dropped in favor of a single prediction by DINE to immediately attach a frame to the scene.

Text2Room. To attach newly generated frames to the existing mesh, IronDepth [1], which is a depth inpainting model, is utilized, prior to a mesh fusion process, which ensures holes are eliminated by connecting existing vertices. We replace IronDepth with DINE, keeping the mesh fusion intact, which cannot be replicated by our model.

3.4. Ablations

To validate the effectiveness of the design choices in our training pipeline, we ablate them and provide their results on our scene geometry evaluation benchmark SSG-3D in Table S3.

3.4.1 Distillation of High-Resolution Models

In our training procedure, we utilize a teacher model to obtain dense pseudo-ground truth depth maps. Marigold [7] is our model of choice, which was (also) trained on the Hypersim dataset. Due to the simulated nature of this dataset, Hypersim has ground-truth depth maps with notably finer structures than ScanNet, which was captured with less precise real-world tools, that are reproduced by Marigold. As we observe improved performance for this dataset once we use Marigold predictions as ground-truth, we deem our knowledge distillation setup to be effective.

3.4.2 Inpainting Task Probability

We observe that there is merit to not allocating barely any or most training steps to learning the inpainting task (see Section 3.2 for the specifics of our training procedure). Not dedicating any steps to the original depth estimation task
















Rendered View				
				
Outpainted View				
				
Point Cloud				
				
CLIP Score				
27.97 \pm 0.55	27.87 \pm 0.56	27.78 \pm 0.74	27.91 \pm 0.73	28.11 \pm 0.75

Table S1. **CLIP score for increasingly misaligned 3D scenes.** We show a scene that has been extended with a newly generated frame from Stable Diffusion, aligning it with DIne. The predicted depth is perturbed, creating a misalignment between both frames. For the CLIP score, we report the mean and standard deviation across ten samples with different seeds. Despite the presence of these geometric inconsistencies in the three-dimensional scene, the reported CLIP score remains essentially unchanged.

without any sparse input appears to negatively impact the performance. We find that spending between 50-75% (i.e., $p \in [0.25, 0.5]$) of the time in the fine-tuning process training the inpainting task yields performant models.

3.4.3 Masking Strategy

We observe that using warped masks that mimic the characteristic inpainting patterns that occur when changing viewpoints is critical to yield a high-performing depth inpainting model for depth inpainting in a scene generation setting. A naive patch-based masking approach produces inferior results, where regions of varying size are randomly masked. Masking patterns generated with *much* smaller viewpoint changes than encountered during scene generation ($\leq 5^\circ$, indicated by \circ in Table S3) also lead to inferior performance. We show that our design yields a method that generalizes to viewpoint changes not encountered during training by presenting qualitative results for SceneScape [5]-like camera trajectories.

3.4.4 Original Task Preservation

In our fine-tuning setup, we can set a probability p to zero out the sparse depth input to the model, effectively reverting to the monocular depth estimation task. In this setting, our model is highly competitive with the original ZoeDepth model (see Table 1), suggesting the inpainting ability has been bolted onto the network in our fine-tuning setup with only minor degradation of the original task. However, this setting cannot reach the performance of a network that is given sparse depth input. This supports our hypothesis that adding sparse depth information of the existing scene leads to geometrically more coherent predictions and more faithful to observed depth.

4. Additional Qualitative Results

We present an additional set of generated scenes from real-world images in Figure S2. As in Figure 4, we provide individual images of the hallucinated views, a rendering of the entire generated 3D scene, as well as a cut-away that provides more detail.

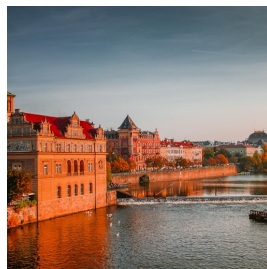
To show that our method generalizes to other camera tra-



kyoto
„a street with traditional buildings in Kyoto, Japan”



nc
„A suburban street in North Carolina on a bright, sunny day”



prague
„Prague during the golden hour”



indoor0
„a living room filled with furniture and a large window”



indoor1
„a living room with couches and a coffee table”



indoor2
„a living room filled with furniture and a fire place”



indoor3
„a living room filled with furniture and a fire place”

Table S2. **Input images and prompts used in Table 2.** The names are solely used to identify these image in the original table. All images have the dimensions 512x512. indoor0-3 are taken from the GitHub repository associated with Text2Room [6].

Input	Depth Annot.	p	Warped Masks	Align.	ScanNet [4]	Hypersim [12]
RGB+sd	Original	0.5	-	-	0.7734	2.2913
RGB+sd	Original	0.5	✓	-	0.1015	0.7615
RGB+sd	ZoeDepth	0.5	✓	-	0.0793	0.7555
RGB+sd	Marigold	0.0	✓	-	0.0864	0.7547
RGB+sd	Marigold	0.25	✓	-	0.0791	0.7301
RGB+sd	Marigold	0.75	✓	-	0.0869	0.7578
RGB+sd	Marigold	0.5	○	-	0.1373	0.8732
RGB	Marigold	0.5	✓	-	0.2553	1.1536
RGB	Marigold	0.5	✓	✓	0.1335	0.8152
RGB+sd	Marigold	0.5	✓	-	0.0816	0.7295

Table S3. **Scene geometry evaluation results for ablations of our method.** We consider the input for our model (image-only or supplemented with sparse depth), the source of depth annotations in our fine-tuning process to learn the inpainting task, the probability p that we mask out the sparse depth input, whether we use warped masks during the fine-tuning process that mimic characteristic inpainting patterns in scene generation, and if the final depth prediction is aligned with the existing point cloud through a global scale-and-shift operation.

jectories, we adopt those used in SceneScape [5], which generate tunnel-like scenes. Starting from a single image, we translate and rotate the camera with each step, simu-

lating the camera focused on the scene center while being pulled back, thus leading to different inpainting patterns than encountered in training. In Figure S1, we show that

our method creates believable 3D scenes despite not being trained for these particular trajectories. For this task, we use a visual question answering model [9] to reject poor image inpainting results (e.g., the image is framed or borders appear), following previous works [13]. We utilize the query “does the image have a frame?”, which yields “yes” or “no” responses.

5. Discussion & Limitations

The performance of our depth inpainting model may be limited by a shift in the data distribution between training and inference. The data the model is trained on (such indoor scenes from NYU Depth or the datasets used to pre-train DPT [10], which forms the core of ZoeDepth [3]), differ from the data it is typically applied to during 3D scene generation (i.e., synthetic images generated by Stable Diffusion). The differences may be due to the type of imagery (e.g., landscapes such as the *Mountains in Peru* in Figure S2), but also due to imperfections or artifacts caused by image generators that the depth network has never encountered during training.

Another challenge lies in the inherently limited resolution of all depth estimation networks, which constrains the model’s ability to accurately predict the depth of fine structures, especially around object boundaries. Consequently, these fine details might become detached from their corresponding objects during projection, affecting the image inpainting step. An example is shown in Figure S3.

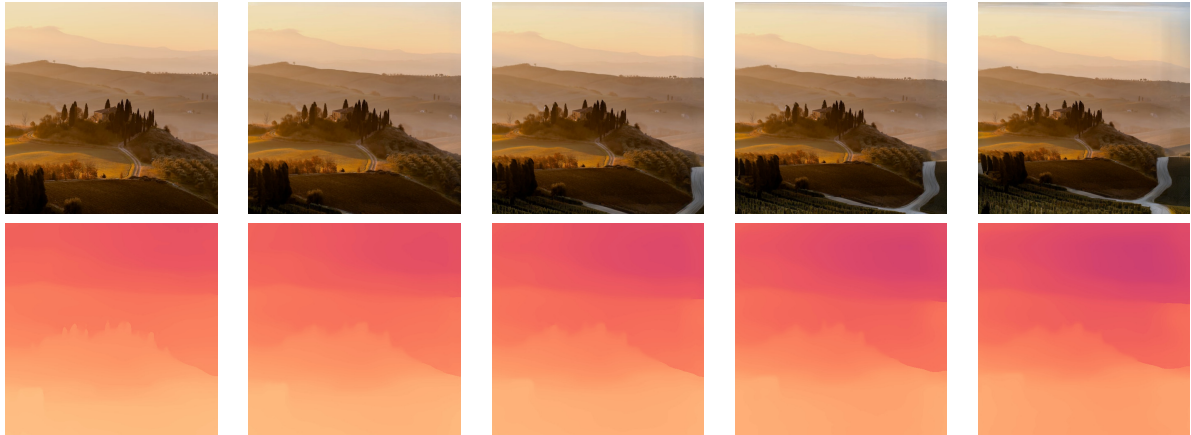
References

- [1] Bae, G., Budvytis, I., Cipolla, R.: Irondepth: Iterative refinement of single-view depth using surface normal and its uncertainty. In: British Machine Vision Conference (BMVC) (2022) 2
- [2] Bao, H., Dong, L., Piao, S., Wei, F.: Beit: Bert pre-training of image transformers. arXiv preprint arXiv:2106.08254 (2021) 2
- [3] Bhat, S.F., Birkel, R., Wofk, D., Wonka, P., Müller, M.: Zoedepth: Zero-shot transfer by combining relative and metric depth. arXiv preprint arXiv:2302.12288 (2023) 2, 5
- [4] Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5828–5839 (2017) 4
- [5] Fridman, R., Abecasis, A., Kasten, Y., Dekel, T.: Scenescape: Text-driven consistent scene generation. Advances in Neural Information Processing Systems 36 (2024) 3, 4, 7
- [6] Höllein, L., Cao, A., Owens, A., Johnson, J., Nießner, M.: Text2Room: Extracting textured 3D meshes from 2D text-to-image models. In: ICCV (2023) 4
- [7] Ke, B., Obukhov, A., Huang, S., Metzger, N., Daudt, R.C., Schindler, K.: Repurposing diffusion-based image generators for monocular depth estimation. arXiv preprint arXiv:2312.02145 (2023) 2
- [8] Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., Dollár, P., Girshick, R.: Segment anything. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023) 2
- [9] Li, J., Li, D., Xiong, C., Hoi, S.: Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In: International conference on machine learning. pp. 12888–12900. PMLR (2022) 5
- [10] Ranftl, R., Bochkovskiy, A., Koltun, V.: Vision transformers for dense prediction. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 12179–12188 (2021) 2, 5
- [11] Ranftl, R., Lasinger, K., Hafner, D., Schindler, K., Koltun, V.: Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. IEEE transactions on pattern analysis and machine intelligence 44(3), 1623–1637 (2020) 2
- [12] Roberts, M., Ramapuram, J., Ranjan, A., Kumar, A., Bautista, M.A., Paczan, N., Webb, R., Susskind, J.M.: Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10912–10922 (2021) 4

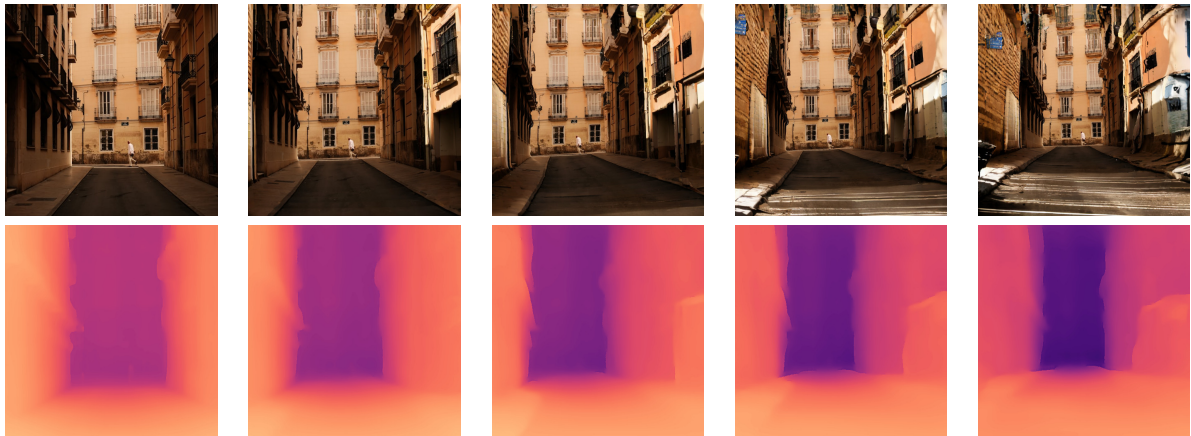
- 288 [13] Yu, H.X., Duan, H., Hur, J., Sargent, K., Rubinstein,
289 M., Freeman, W.T., Cole, F., Sun, D., Snavely, N., Wu,
290 J., et al.: Wonderjourney: Going from anywhere to
291 everywhere. In: CVPR. pp. 6658–6667 (2024) 5



“an alley in rural Spain on a bright, sunny day”



“a foggy morning in Tuscany”



“a long, narrow street fully engulfed in shadows in Valencia, Spain”

Figure S1. **Qualitative results of our method on SceneScape [5]-like trajectories.** Despite not being trained with the distortion patterns resulting from these trajectories, our method is able to generate convincing scenes.

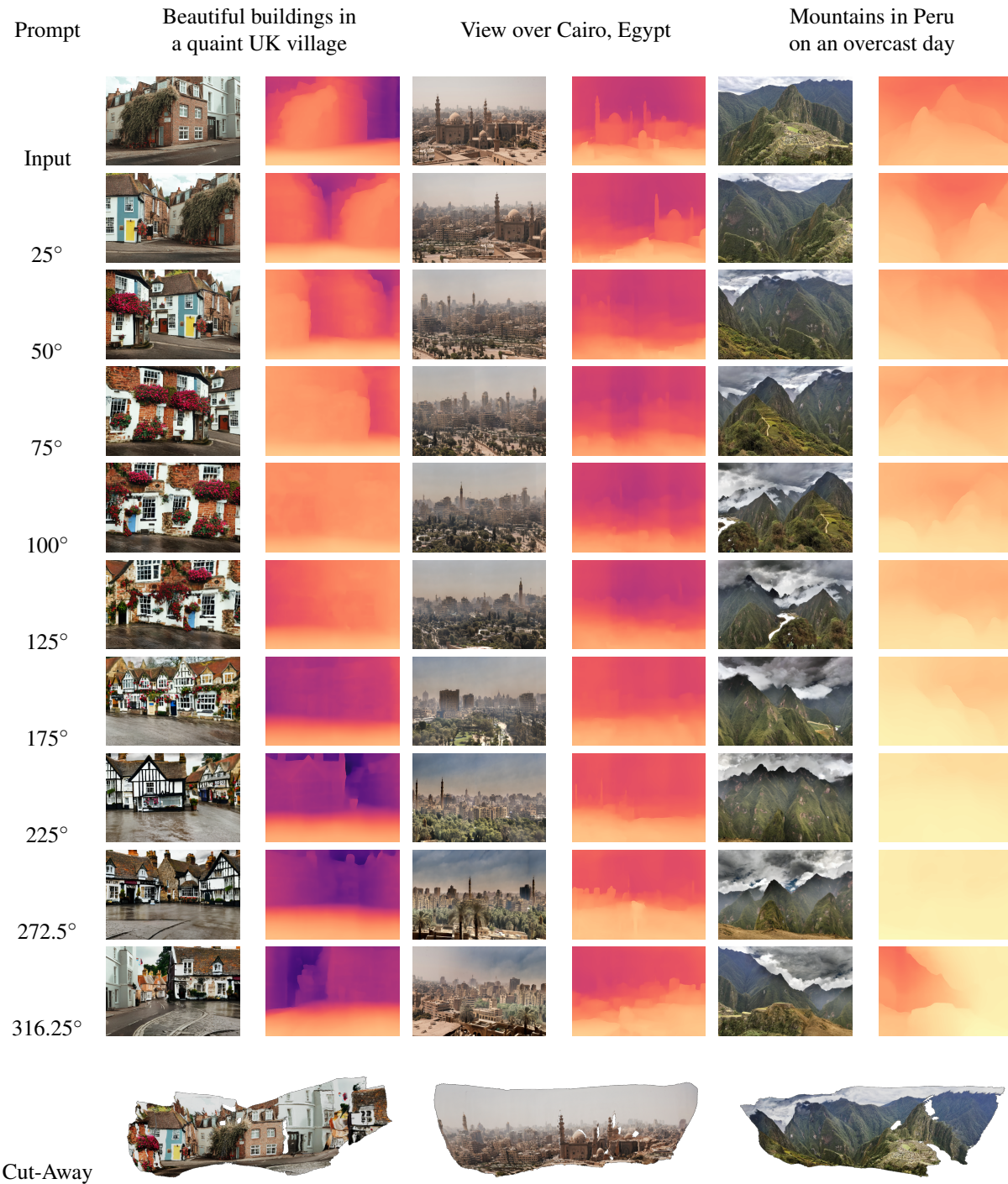


Figure S2. **Qualitative results of our method on additional real-world images.** It is able to generate convincing, immersive scenes for a wide range of input images and associated prompts.



Input image

Projection based on depth prediction,
after depth snapping

Figure S3. **Loss of fine object details after projection.** Due to the limited resolution of depth predictions, fine details might be detached from their corresponding objects and become part of the background. We present an example of this for the given image, showing the resulting projection after applying our depth snapping to remove floating points (as outlined in Section 5.3.2). The fine hairs of the dog at its boundary have become part of the background.