

DreamAnywhere: Object-Centric Panoramic 3D Scene Generation

Anonymous WACV Applications Track submission

Paper ID

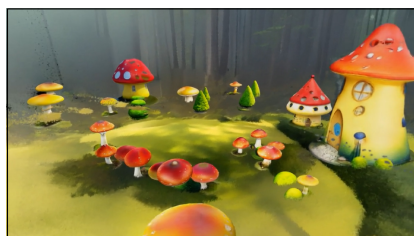
001	Contents				
002	1. Additional results	1			
003	2. 2D Inpainting ablation	1			
004	3. Object Generation Details	1			
005	3.1. Implementation Details	1			
006	3.2. NeRF to 3DGS conversion	3			
007	4. Pose Estimation and Scene Alignment Details	3			
008	5. Background Generation Details	4			
009	6. 3D Inpainting Details	4			
010	7. Object-Light interactions	5			
011	8. Details of Language Models	5			
012	9. User Study Details	5			
013	1. Additional results				
014	In figure 1 we show results of our method for additional				
015	scenes used for evaluation, the respective text prompt and				
016	post-processed frames with Cosmos [1].				
017	2. 2D Inpainting ablation				
018	In Figure 2 we show qualitative results of our 2D inpainting				
019	pipeline. Our complete approach that includes LaMA pre-				
020	inpainting and a fine-tuned LoRA leads to better results than				
021	what can be achieved with off-the-shelf Stable Diffusion				
022	inpainting. LaMa inpainting alone is not enough since the				
023	inpainted regions are still blurry. Our approach reduces the				
024	introduction of new objects in the inpainting regions when				
025	compared to Stable Diffusion inpainting.				
026	3. Object Generation Details				
027	To improve the similarity between objects as they occur in				
028	the original panorama image and the high-fidelity regener-				
029	ated object, we employ depth and style conditioning in the				
			text-to-image process. Figure 3 gives an overview of the		
			effect that different conditioning signals have on the object		
			generation process. Although unconstrained diffusion mod-		
			els might offer greater creative freedom, conditioning the		
			new image on depth serves a dual purpose: not only does it		
			preserve the object’s structural fidelity, it also enhances the		
			robustness of the subsequent pose estimation between the		
			original and newly obtained image. This approach proved		
			to be more effective than conditioning on text-only. We also		
			experimented with Canny edge control, but found it to be		
			problematic, especially in case of partial objects for which it		
			is incapable of completing the object.		
			3.1. Implementation Details		
			For the object generation step we use the <i>large</i> version of		
			the TENCENTARC/INSTANTMESH model with a render		
			resolution of 512, 75 diffusion steps and a point cloud reso-		
			lution of 128 ³ . We generate 144 views uniformly distributed		
			across a unit sphere (12 azimuth × 12 elevation angles) and		
			generate color images from the triplane, while depth maps		
			are extracted using the DEPTH-ANYTHING-V2 [10] model.		
			We train 3DGS with depth regularization enabled, following		
			the methodology of Kerbl et al. [4, 5], for 5000 iterations.		
			Although 1000 iterations typically suffice for good-quality		
			object reconstruction, the extended training ensures opti-		
			mal results. On average, a high-quality object is typically		
			comprised of 30-50k Gaussians.		
			As depth-based ControlNet for the high-quality		
			objects we use STABILITYAI/STABLE-DIFFUSION-3.5-		
			LARGE-CONTROLNET-DEPTH in combination with		
			STABILITYAI/STABLE-DIFFUSION-3.5-LARGE and a		
			separate cross-attention layer for style conditioning [12].		
			We empirically set the control strength to 0.8 (depth)		
			and 0.3 (style), a guidance scale of 7.5 and 50 inference		
			steps. As a negative prompt we include “low quality, low		
			resolution, blurry”. The depth control image is generated		
			using DepthFM [3] after padding the crop image to a square		
			size.		

Novel View Renderings

Cosmos-Transfer1



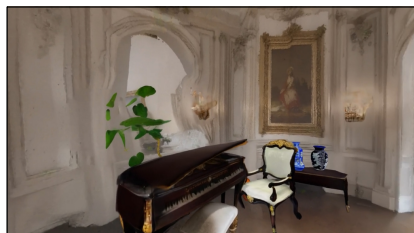
A lighthouse in the Arctic, van Gogh painting



A whimsical forest clearing with mushroom houses



A galactic saloon with an octopus serving drinks



A grand piano sits in a spacious room... painting and vases... chairs...



Rolling hills with oversized pumpkins and colorful flower fields, vibrant sky



A majestic peacock surfing a tall wave

Figure 1. Additional results generated with our method across different indoor and outdoor environments. We show renderings from novel viewpoints unseen at training times as well as the result using Cosmos-Transfer1 [1], which can be used to increase sharpness as a post processing step, or if preferred viewing positions are known.

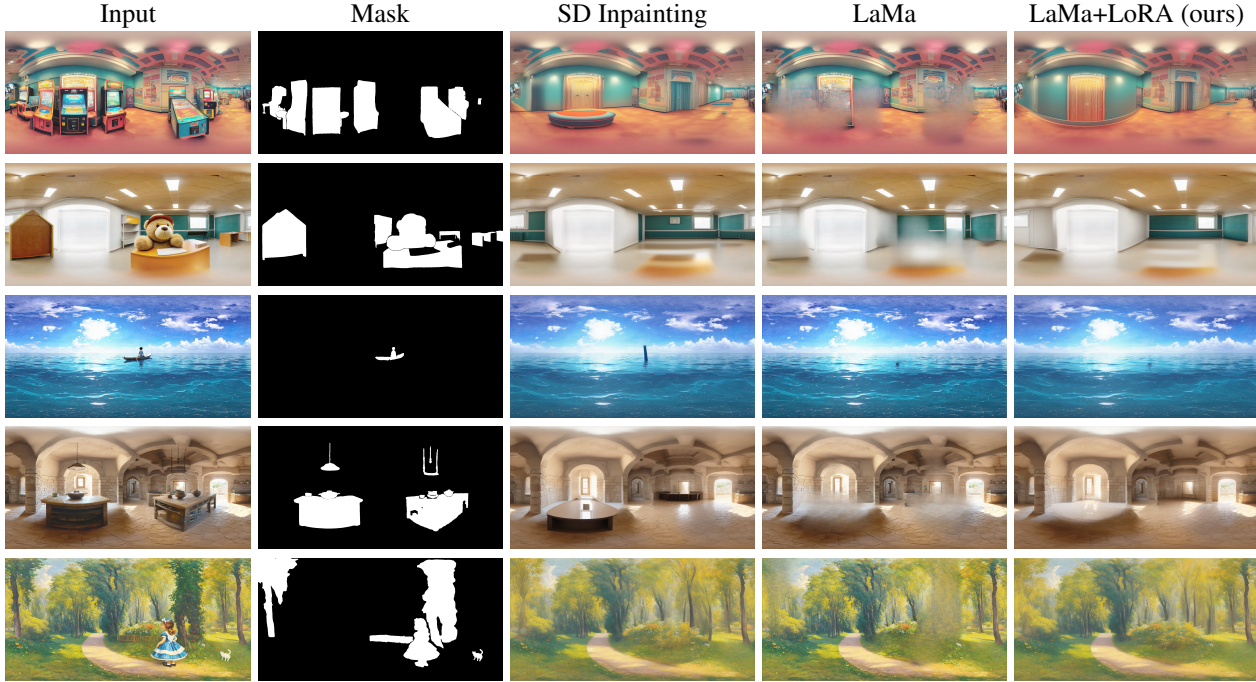


Figure 2. Our combination of LaMa pre-inpainting and a fine-tuned LoRA outperforms both, LaMa and Stable Diffusion (SD) inpainting, delivering superior results in terms of detail and coherence. This combination enables effective inpainting for both indoor and outdoor scenes.

3.2. NeRF to 3DGS conversion

To translate the triplane representation to 3DGS, we begin by generating a point cloud that identifies likely surface points in the volumetric representation using a regular grid. For each sample point, we compute density values through triplane sampling. Surface points are then identified using neighbor-based thresholding, guided by an empirically determined threshold of 0.2. These surface points are added as Gaussian locations in the point cloud, with their initial color values derived directly from the triplane representation. Next, we render 144 RGB-D images from positions uniformly distributed across a sphere. This spherical distribution ensures comprehensive coverage of the object’s geometry while providing robust supervision signals for reconstruction for standard 3DGS optimization with depth supervision [4, 5]. Notably, this extraction process is significantly more efficient and robust than using COLMAP for point initialization.

4. Pose Estimation and Scene Alignment Details

Reference Pose Estimation To estimate the reference pose of the original object O_{crop} , we utilize the crop mask to determine the mean depth d_m and the midpoint of the crop c_p . Considering the spherical model used for the panorama, with the camera placed at the origin, we move the object’s

origin along a ray through c_p to depth d_m . By default, our object generation stage orients the object’s front-face along the Z-axis to match the 2D image input, *i.e.*, the image-plane of the 2D crop is fixed at the x/y plane in object-space. To ensure that the front face of the object is aligned towards the camera, we simply rotate the object around the vertical axis. To estimate the object’s scale, we unproject the crop’s corner points at depth d_m into world-space, treating the crop as the x/y front-face of the object’s bounding box. Finally, we apply a grounding offset based on the minimal z -coordinate of the object’s crop in the panorama.

Relative Pose Estimation. The point cloud of the high-quality object O_{hq} is not guaranteed to be semantically aligned with that of the original object O_{crop} as the generated objects may vary in size and orientation. Thus, for each object, we perform an image-based relative pose estimation step that computes the relative pose between O_{crop} and O_{hq} . This process operates in two stages. First, we extract DINO [2] embeddings and compute cosine similarities between the perspective image of the cropped object (I_{crop}) and a set of perspective projections of O_{hq} with known camera poses (the intermediate results of the object generation process described in Section 3.1). Based on this we select the top N most similar reference images. We pair each of these with the image of O_{crop} and compute the relative pose

“The arcade machine is teal with vibrant yellow and red artwork. It’s rectangular, slanted back, and made of metal and plastic. Visible text includes: “Emperor” and “Ananadapurs”. There are buttons and a coin slot. The arcade machine appears whole.”

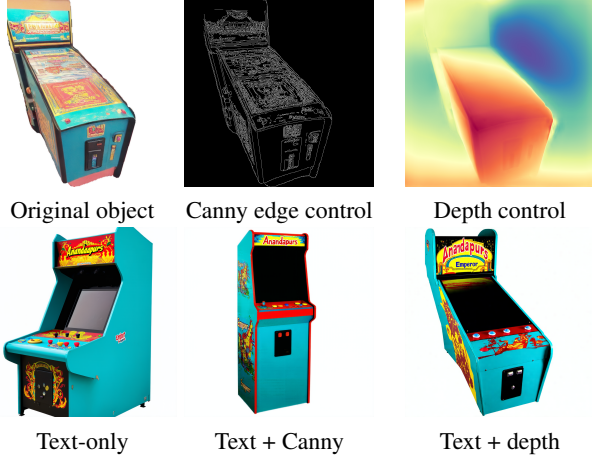


Figure 3. Using Stable Diffusion in combination with a ControlNet gives object images that are much closer to the original object in comparison to images generated based on only the text description. From the original image we extract the Canny edge and depth control images. All re-generated images are generated using the text prompt shown at the top. From left to right we have the image generated without additional conditioning, generated with additional Canny edge conditioning and generated with additional depth conditioning.

using MAST3R [7]. The final relative pose P_r is determined from the image pair with the lowest optimization loss.

5. Background Generation Details

To generate an initial Gaussian point cloud from the generated panorama I_B and its corresponding aligned depth map \tilde{D}_B , we first unproject the pixels into 3D positions \mathbf{P} using spherical coordinates ϕ, θ (obtained from the pixel coordinates) and depth values d : $x = d \cos \phi \cos \theta$, $y = d \cos \phi \sin \theta$, $z = d \sin \phi$. To convert 2D pixel coordinates (x, y) to spherical angles, we use the following equations:

$$\phi = \frac{\pi}{2}, \quad \theta = x\pi. \quad (1)$$

Next, the normal map \mathbf{N}_B is computed as the cross product of the partial derivatives of 3D coordinates \mathbf{P} in x and y -direction.

To keep things tractable, we sub-sample \mathbf{P} , \mathbf{N} , and \mathbf{G}_B in spherical space: we use a uniform lat/lon lattice for horizon regions ($|\phi| \leq 30^\circ$) and a Fibonacci lattice for polar regions to achieve a more uniform sampling distribution in 3D. Each

sample spawns a Gaussian with:

$$\begin{aligned} \mu &= \mathbf{P}(\theta, \phi), \\ \mathbf{c} &= \text{sh}(I_B(\theta, \phi)), \\ \alpha &= 1, \end{aligned}$$

where μ is the mean, \mathbf{c} is the RGB color encoded as a spherical harmonic and α is the opacity.

Following Kerbl et al. [4], we represent the covariance matrix as $\Sigma = R S S^T R^T$, where R is the rotation matrix aligning the z -axis with the normal direction, and $S = \text{diag}(\psi, \psi, 0.01)$ encodes scale based on sampling density. We set the scale factor ψ relative to the sampling density factor ((F_h, F_p) for horizon and polar regions respectively):

$$\psi = \frac{2\pi}{2H_s} \cdot F(\phi),$$

with $F(\phi) \in \{F_h, F_p\}$, depending on the region and H_s the vertical sampling resolution. Experimentally, we set $F_p = 8$ and $F_h = 1$.

We set each Gaussians’ scale S based on the vertical sampling resolution H_s and a sampling density factor ((F_h, F_p) for horizon and polar regions respectively). The original sample count for either the uniform lattice or Fibonacci lattice strategy is then computed as $2 \cdot (H_s/F(\phi))^2$.

Finally, we design the Gaussian scale S to be directly related to the sampling density, with a constant scale in Z :

$$\begin{aligned} \psi &= \frac{2\pi}{2H_s} \cdot F(\phi), \\ S &= \text{diag}(\psi, \psi, 0.01), \end{aligned}$$

with $F(\phi) \in \{F_h, F_p\}$, depending on the region. Experimentally, we set $F_p = 8$ and $F_h = 1$.

6. 3D Inpainting Details

Implementation Details. After initializing the 3DGS point cloud from the panoramic image, we execute a three-stage pipeline for 3D inpainting (*Pretuning*, *Inpainting* and *Multi-view Fine-tuning*), where we use 3k/2k/2k optimization steps for the individual stages. We use MCMC-densification [6] with $\lambda_{\text{noise}} = 5e^4$, and also include their regularization terms for opacity and scale, with $\lambda_o = \lambda_\Sigma = 0.5$. For optimization on panorama reference views and inpainted images, we use a combination of ℓ_1 -loss, a SSIM term [8] following Kerbl et al. [4], as well as a ℓ_2 -loss against the predicted reference depth D_B to avoid overfitting to specific views and maintaining the structure of the initialization. For all views, we leverage an opacity loss, forcing the per-pixel transmittance T towards 1:

$$\mathcal{L}_{\text{img-opacity}} = \lambda_{\text{img-opacity}} \|\mathbf{1} - T\|^2, \quad (2)$$



Figure 4. We reintroduce contact shadows which were removed during 2D object removal and inpainting. Simple light interaction between generated objects and the background is essential to ground the generated objects.

with $\lambda_{\text{img-opacity}} = 1e^3$. To reduce noise on the final renderings, we also use a TV-loss throughout, with $\lambda_{\text{TV}} = 1e^3$. Finally, during *Pretuning* and *Multi-view Fine-tuning*, we reset the opacity and scale for Gaussians that are either too large or outside the scene bounds, similar to Kerbl et al. [4]. Specifically, Gaussians are removed when $\max(S) > 0.05 \cdot \lambda_{\text{max}}$ or $\|\mu\| > 1.5 \cdot \lambda_{\text{max}}$. Here, λ_{max} denotes our estimated scene extent, which we define as

$$\lambda_{\text{max}} = \max_{\mu_i \in \mathcal{P}_{\text{initial}}} \|\mu_i\|_{\infty}, \quad (3)$$

where $\mathcal{P}_{\text{initial}}$ is the set of initial Gaussians (before inpainting). This pruning step is crucial for the elimination of elongated, disk-like Gaussians. We use LYKON/DREAMSHAPER-8-INPAINTING to inpaint the reference views, with a guidance scale of 7.5 and the same prompts as used to generate the initial panorama. The reference panorama image I_B is up-scaled in two steps, first using a diffusion-based upscaler [11], followed by a non-generative one [9].

We typically use point clouds of $\sim 4M$ points as input to the inpainting stage. We experimentally found that this offers an optimal balance between rendering efficiency, optimization performance, and sufficient density for high-quality inpainting results. The *Pretuning* stage permits a 10% increase of Gaussians for small gap filling, while the *Inpainting* step typically introduces an additional 100k Gaussians.

7. Object-Light interactions

To enhance scene perception, we reintroduce light interactions removed during 2D object removal and inpainting, focusing on contact shadows, a critical cue for depth perception. Due to the lack of reliable surface normals and the absence of Gaussian raytracing in our renderer, we rely on shadow mapping to recover contact shadows, as shown in Figure 4.

As a simple heuristic for contact shadows, we generate an orthographic shadow map using a vertically positioned camera. We ignore the first layer of Gaussians which typically represents the sky or roof. We create a soft shadow mask by computing the distance to the next discontinuity in the depth map which we use during shadow evaluation

to approximate a penumbra. To apply shadows to every Gaussian, we compare their positions to the shadow map, while considering a small offset due to the fuzzy nature of depth maps generated for Gaussians. We darken occluded Gaussians by mixing black into their spherical harmonics, effectively baking shadows into the representation. We also support spotlights by estimating light source locations in the panorama.

8. Details of Language Models

For the high-quality object generation described in Section 3.2, we prompt GPT-4o as follows: “*Describe the j object-category $_i$ in the image with a focus on its detailed attributes. Include its color, pose or orientation relative to the scene, shape, size, and material. If there is any text visible on the object, transcribe it accurately. Do not describe the background or any elements not part of the object. Be concise but complete, ensuring the total description aids accurate object reconstruction. Max 40 words.*”.

9. User Study Details

Participant Details. We recruited 28 participants, all with normal or corrected vision, 23-40 years old, with moderate to expert familiarity with computer graphics and vision. The study was deployed through a web interface (Google Forms) showing side-by-side video comparisons of 6s walkthroughs of the scenes. The user study first explained the criteria reported in the main text and outlined the scoring system from -2 to +2, then proceeded to show them video pairs (left and right). We did not restrict answer times, and allowed the user to watch the video repeatedly. In general, each user study lasted about ~ 20 minutes and the results showed a consistent overall preference for our method. The preferences for *Coherence* and *Immersiveness* were particularly strong, while text alignment was weaker, reflecting the CLIP score in the quantitative evaluation.

References

- [1] Hassan Abu Alhaija, Jose Alvarez, Maciej Bala, Tiffany Cai, Tianshi Cao, Liz Cha, Joshua Chen, Mike Chen, Francesco Ferroni, Sanja Fidler, et al. Cosmos-transfer1: Conditional world generation with adaptive multimodal control, 2025.
- [2] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.
- [3] Ming Gui, Johannes Schusterbauer, Ulrich Prestel, Pingchuan Ma, Dmytro Kotovenko, Olga Grebenkova, Stefan Andreas Baumann, Vincent Tao Hu, and Björn Ommer. Depthfm: Fast monocular depth estimation with flow matching, 2024.
- [4] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D Gaussian Splatting for Real-Time Radi-

266 ance Field Rendering. *ACM Transactions on Graphics*, 42(4),
267 2023.

268 [5] Bernhard Kerbl, Andreas Meuleman, Georgios Kopanas,
269 Michael Wimmer, Alexandre Lanvin, and George Drettakis.
270 A hierarchical 3d gaussian representation for real-time ren-
271 dering of very large datasets. *ACM Transactions on Graphics*,
272 43(4), 2024.

273 [6] Shakiba Kheradmand, Daniel Rebain, Gopal Sharma, Weiwei
274 Sun, Yang-Che Tseng, Hossam Isack, Abhishek Kar, Andrea
275 Tagliasacchi, and Kwang Moo Yi. 3D Gaussian Splatting as
276 Markov Chain Monte Carlo. In *Proceedings of the Conference*
277 *on Neural Information Processing Systems (NeurIPS)*, 2024.

278 [7] Vincent Leroy, Yohann Cabon, and Jerome Revaud. Ground-
279 ing image matching in 3d with mast3r. In *Proceedings of the*
280 *European Conference on Computer Vision (ECCV)*, 2024.

281 [8] Saswat Subhajyoti Mallick, Rahul Goel, Bernhard Kerbl,
282 Markus Steinberger, Francisco Vicente Carrasco, and Fer-
283 nando De La Torre. Taming 3dgs: High-quality radiance
284 fields with limited resources. In *SIGGRAPH Asia 2024 Con-*
285 *ference Papers*, New York, NY, USA, 2024. Association for
286 Computing Machinery.

287 [9] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan.
288 Real-esrgan: Training real-world blind super-resolution with
289 pure synthetic data. In *Proceedings of the IEEE/CVF Inter-*
290 *national Conference on Computer Vision (ICCV) Workshops*,
291 2021.

292 [10] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiao-
293 gang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything
294 v2. In *Proceedings of the Conference on Neural Information*
295 *Processing Systems (NeurIPS)*, 2024.

296 [11] Tao Yang, Rongyuan Wu, Peiran Ren, Xuansong Xie, and
297 Lei Zhang. Pixel-aware stable diffusion for realistic image
298 super-resolution and personalized stylization. In *Proceedings*
299 *of the European Conference on Computer Vision (ECCV)*,
300 2023.

301 [12] Hu Ye, Jun Zhang, Sibio Liu, Xiao Han, and Wei Yang. Ip-
302 adapter: Text compatible image prompt adapter for text-to-
303 image diffusion models, 2023.