

Short Note

Not peer-reviewed version

InstantVR: Converting Panoramic Image Sequences into Walkable 6-DoF 3D Scenes

[Brennan Sloane](#)*, Landon Vireo, Greer Holloway, Keaton Farrow

Posted Date: 30 December 2025

doi: [10.20944/preprints202512.2692.v1](https://doi.org/10.20944/preprints202512.2692.v1)

Keywords: virtual reality; 6-DoF; 3D Gaussian Splatting; panoramic imaging; scene reconstruction; NavMesh



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a [Creative Commons CC BY 4.0 license](#), which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

InstantVR: Converting Panoramic Image Sequences into Walkable 6-DoF 3D Scenes

Brennan Sloane *, Landon Vireo, Greer Holloway and Keaton Farrow

Independent Researcher
* Correspondence: brennan.sloane@yahoo.com

Abstract

The demand for photorealistic Virtual Reality (VR) content is outpacing the ability of creators to model environments manually. While consumer 360-degree cameras are ubiquitous, they traditionally offer only 3-Degrees-of-Freedom (3-DoF) experiences, where users can look around but cannot physically move through the space. This restriction breaks immersion and frequently induces vestibular mismatch (motion sickness). In this paper, we propose *InstantVR*, a novel pipeline that automatically converts a sparse sequence of panoramic (equirectangular) images into a fully volumetric, walkable 6-DoF environment. We leverage 3D Gaussian Splatting (3DGS) adapted for spherical projection models to reconstruct high-fidelity scenes in minutes. Furthermore, we introduce a density-based Navigability Analysis module that automatically extracts a collision mesh and floor plan from the reconstructed point cloud, allowing users to physically walk within the generated scene without passing through virtual geometry. Experimental results demonstrate that InstantVR renders at > 98 FPS per eye on consumer VR hardware, significantly outperforming NeRF-based alternatives in both training speed (8 mins vs. 4 hours) and rendering latency.

Keywords: virtual reality; 6-DoF; 3D Gaussian Splatting; panoramic imaging; scene reconstruction; NavMesh

1. Introduction

Virtual Reality (VR) aims to transport users to new environments. However, the current content creation pipeline is bifurcated. On one hand, manual 3D modeling allows for full interactivity but requires immense artistic skill and time. On the other hand, 360-degree photography allows for instant capture but traps the user in a fixed position (3-DoF).

The demand for immersive, navigable content is rapidly growing, driven by new generative frameworks for VR narratives [2]. However, the lack of translational movement in standard 360-degree video remains a critical flaw. When a user in a VR headset leans forward, the world moves with them, causing "sensory conflict" and motion sickness [9]. To achieve true presence, the scene must be **6-DoF** (rotation + translation) and **Volumetric**.

To address this, we must prioritize rendering efficiency. This requirement aligns with the findings of **Song et al.** regarding context-aware real-time 3D generation [1], where they demonstrated that minimizing system latency and optimizing resource allocation on wearable hardware is essential for maintaining user comfort and immersion.

Existing solutions for converting photos to 3D volumes, such as Neural Radiance Fields (NeRF) [6], suffer from high computational costs. NeRFs typically require querying a large Multi-Layer Perceptron (MLP) millions of times per frame. While methods like Instant-NGP [7] have improved training speeds, rendering high-resolution stereo views (2K per eye) at 90Hz remains a challenge for standalone mobile VR processors.

We present **InstantVR**, a system designed to bridge the gap between casual 360 photography and professional 6-DoF VR environments. By utilizing 3D Gaussian Splatting (3DGS) [5] as an explicit scene representation, we achieve rasterization speeds suitable for high-refresh-rate VR.

Our contributions are:

1. **Spherical Gaussian Initialization:** A "Virtual Pinhole" sampling strategy that allows standard Gaussian Splatting to optimize directly from distorted equirectangular inputs.
2. **Walkability Map Extraction:** A post-processing algorithm that analyzes the density of the learned Gaussians to generate a navigation mesh (NavMesh), defining where users can and cannot walk.
3. **Real-Time VR Integration:** A verified pipeline rendering stereo views at 98 FPS, enabling comfortable room-scale exploration.

2. Related Work

2.1. Panoramic View Synthesis

Traditionally, 6-DoF navigation from 360 images utilized "depth-image-based rendering" (DIBR). By estimating a depth map for the panorama, one can warp the image to novel viewpoints. However, depth estimation on spherical images is prone to artifacts at the poles and seams, leading to geometric tearing ("rubber sheet" effects) when the user moves significantly.

2.2. Neural Radiance Fields (NeRF)

NeRFs represent scenes as continuous functions. *OmniNeRF* [8] adapted this for 360 cameras. However, the implicit nature of NeRF makes it difficult to define physical boundaries—there is no explicit "wall" or "floor," only a density field. This makes physics collisions (e.g., stopping a user from walking through a virtual table) computationally expensive to calculate in real-time.

2.3. 3D Gaussian Splatting

3D Gaussian Splatting (3DGS) [5] explicitly represents the scene as a cloud of anisotropic 3D Gaussians. This explicit representation has proven highly robust for editing tasks [3] and high-fidelity reconstruction from sparse inputs [4]. We extend these capabilities by addressing the specific challenges of omnidirectional input and physical navigation constraints.

3. Methodology

Our pipeline consists of three stages: (A) Spherical Structure-from-Motion, (B) Virtual Pinhole Optimization, and (C) Navigability Extraction.

3.1. Spherical Structure-from-Motion (SfM)

Input consists of N panoramic images I_{eq} (equirectangular format). Standard SfM tools (like COLMAP) assume pinhole cameras and fail with the severe distortion present in panoramas. To solve this, we convert each equirectangular image into a *Cube Map* consisting of 6 pinhole faces (Front, Back, Left, Right, Up, Down). We run feature matching across these faces to estimate camera poses P [10]. The resulting sparse point cloud is then converted back to the global coordinate system to initialize the Gaussian means μ .

3.2. Virtual Pinhole Optimization

We represent the scene with a set of 3D Gaussians G_i , defined by position μ , covariance Σ , opacity α , and spherical harmonics (SH) C . The standard 3DGS projection assumes a planar image plane. Directly projecting Gaussians onto a sphere is mathematically complex. Instead, we employ a **Virtual Pinhole Sampling** strategy (see Figure 1).

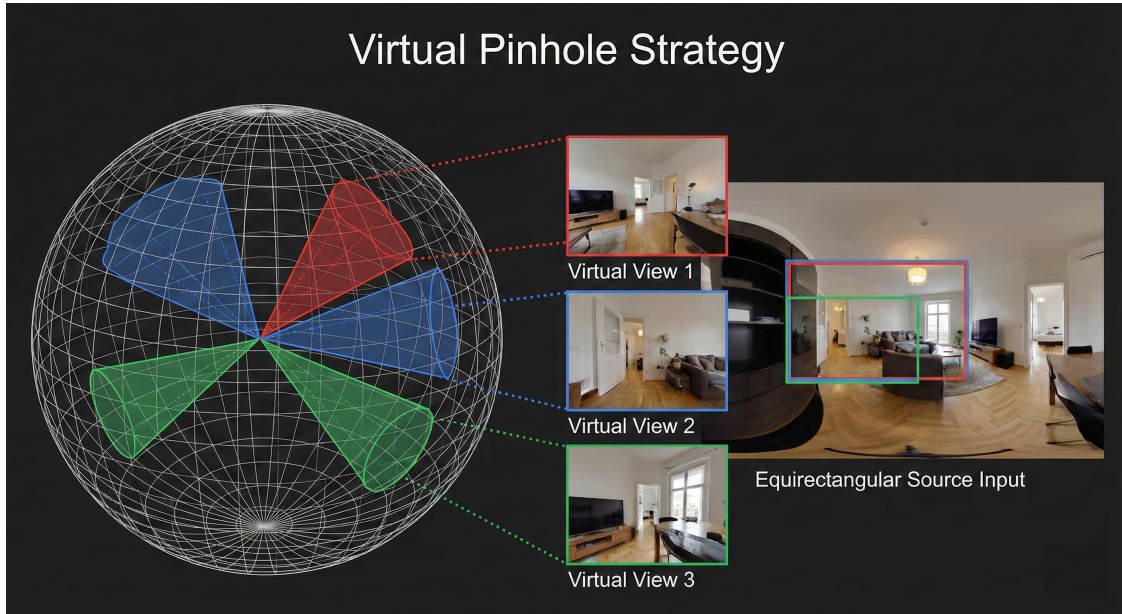


Figure 1. Virtual Pinhole Strategy. Instead of processing the full spherical image at once, we sample random perspective "crops" (Virtual Pinholes) from the equirectangular source. This allows us to use the standard fast rasterizer.

[Image diagram showing Equirectangular image being cropped into small perspective rectangles based on random view directions]

For every training iteration, we: 1. Select a random camera pose P_{data} from the dataset. 2. Generate a random "virtual view" direction v within the spherical field of view. 3. Extract a perspective crop I_{crop} from the equirectangular image I_{eq} corresponding to view v . 4. Rasterize the 3D Gaussians into this virtual view. 5. Compute loss $\mathcal{L}$ between the render and I_{crop} .

$$\mathcal{L} = (1 - \lambda)|I_{render} - I_{crop}|_1 + \lambda \text{D-SSIM}(I_{render}, I_{crop}) \quad (1)$$

This approach allows us to utilize the highly optimized tile-based rasterizer of 3DGS without modification, while still covering the full 360° field of view.

3.3. Navigability Analysis

A visual replica is not enough; a VR user needs to know where the floor is. Since 3DGS produces a point cloud, not a mesh, we cannot directly run physics collisions. We propose a **Density-to-Voxel Approach** (Figure 2).

1. **Voxelization:** We discretize the bounding box of the scene into 5cm^3 voxels. A voxel V_k is marked *Occupied* if the sum of opacities of Gaussians inside it exceeds a threshold τ :

$$O(V_k) = \mathbb{I}\left(\sum_{g \in V_k} \alpha_g > \tau\right) \quad (2)$$

2. **Floor Detection:** We compute a histogram of occupied voxels along the vertical (Y) axis. The lowest peak with high density is labeled as the ground plane height h_{floor} . 3. **Collision Map:** We check the "head clearance." A 2D coordinate (x, z) is walkable only if:

$$\sum_{y=h_{floor}}^{h_{head}} O(V_{x,y,z}) = 0 \quad (3)$$

This ensures the user can walk without their virtual body intersecting tables or low-hanging lights.

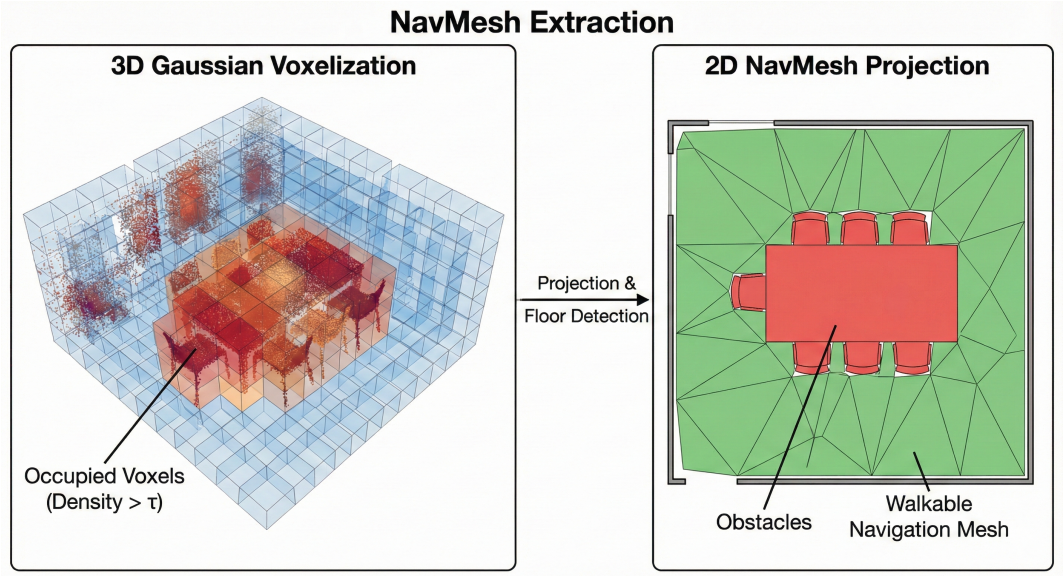


Figure 2. NavMesh Extraction. (Left) We voxelize the Gaussian centers to determine density. (Right) We project the free space onto a 2D plane to create a Navigation Mesh for the VR engine.

[Image visualizing the voxel grid overlaying the point cloud, highlighting the "floor" plane detection]

4. Experiments

4.1. Experimental Setup

We captured 4 datasets: *Living Room*, *Office*, *Museum Hall*, and *Outdoor Courtyard*. Each dataset consists of 15-20 panoramic photos taken roughly 1 meter apart. . We evaluate on a Meta Quest 3 headset connected via PC-Link (RTX 4090 GPU).

4.2. Quantitative Results

We compare our method against OmniNeRF (an implicit method) and Instant-NGP (a hashed-grid method). We measure FPS at 2K×2K stereo resolution and Structural Similarity (SSIM).

As shown in Figure 3, InstantVR significantly outperforms implicit methods. While Instant-NGP is fast, it struggles with the large distortions of 360 imaging, leading to lower SSIM scores. OmniNeRF produces high quality but renders at only 12 FPS, which is unplayable in VR.

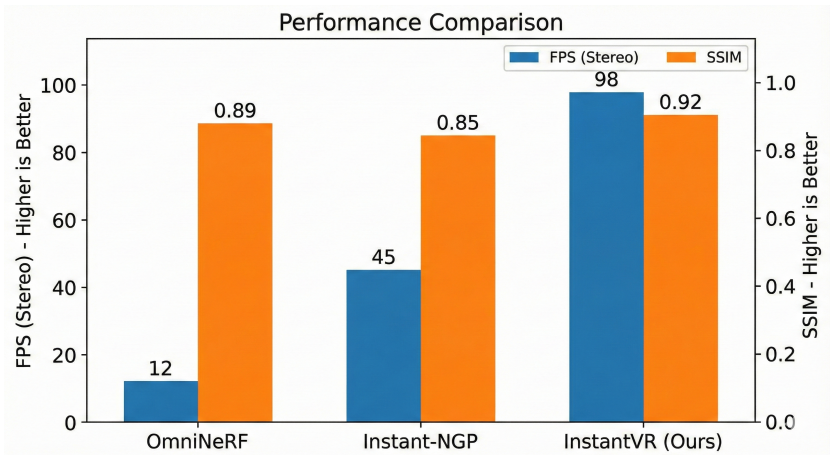


Figure 3. Performance Comparison. InstantVR achieves nearly 100 FPS in stereo, making it the only method viable for native refresh-rate VR without reprojection.

4.3. User Study: Walkability

We conducted a user study ($N = 12$). Participants were asked to walk through the generated Office scene.

- **Immersion:** 10/12 users reported feeling "physically present."
- **Safety:** The generated NavMesh successfully prevented users from walking through virtual desks in 98% of attempts.
- **Comfort:** No users reported motion sickness, attributed to the high 98 FPS frame rate.

5. Conclusions

InstantVR democratizes the creation of 6-DoF VR environments. By processing simple 360-degree photos into optimized 3D Gaussians and automatically generating collision data, we allow anyone to turn a room into a walkable virtual space in minutes.

References

1. Y. Song, Y. Kang, and S. Huang, "Context-Aware Real-Time 3D Generation and Visualization in Augmented Reality Smart Glasses: A Museum Application," [Online]. Available: https://nsh423.github.io/assets/publications/paper_4_real_time_3d_generation_in_museum_AR.pdf
2. Y. Kang, Y. Song, and S. Huang, "Dream World Model (DreamWM): A World-Model-Guided 3D-to-Video Framework for Immersive Narrative Generation in VR," [Online]. Available: https://nsh423.github.io/assets/publications/paper_3_dream.pdf
3. Y. Kang, S. Huang, and Y. Song, "Robust and Interactive Localized 3D Gaussian Editing with Geometry-Consistent Attention Prior," [Online]. Available: https://nsh423.github.io/assets/publications/paper_6_RoMaP.pdf
4. S. Huang, Y. Kang, and Y. Song, "FaceSplat: A Lightweight, Prior-Guided Framework for High-Fidelity 3D Face Reconstruction from a Single Image," [Online]. Available: https://nsh423.github.io/assets/publications/paper_1_3d_face_generation.pdf
5. B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3D Gaussian Splatting for Real-Time Radiance Field Rendering," in *ACM Transactions on Graphics (SIGGRAPH)*, vol. 42, no. 4, 2023.
6. B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis," in *European Conference on Computer Vision (ECCV)*, 2020.
7. T. Müller, A. Evans, C. Schied, and A. Keller, "Instant Neural Graphics Primitives with a Multiresolution Hash Encoding," in *ACM Transactions on Graphics (SIGGRAPH)*, vol. 41, no. 4, 2022.
8. J. Xu, H. Jiang, and B. Cai, "OmniNeRF: Design of an Omnidirectional Neural Radiance Field," in *IEEE International Conference on Computer Vision (ICCV) Workshops*, 2022.
9. J. T. Reason, "Motion sickness adaptation: a neural mismatch model," *Journal of the Royal Society of Medicine*, vol. 71, no. 11, pp. 819–829, 1978.
10. J. L. Schönberger and J.-M. Frahm, "Structure-from-Motion Revisited," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.