HoloScene: Simulation-Ready Interactive 3D Worlds from a Single Video

Anonymous Author(s)
Affiliation
Address
email

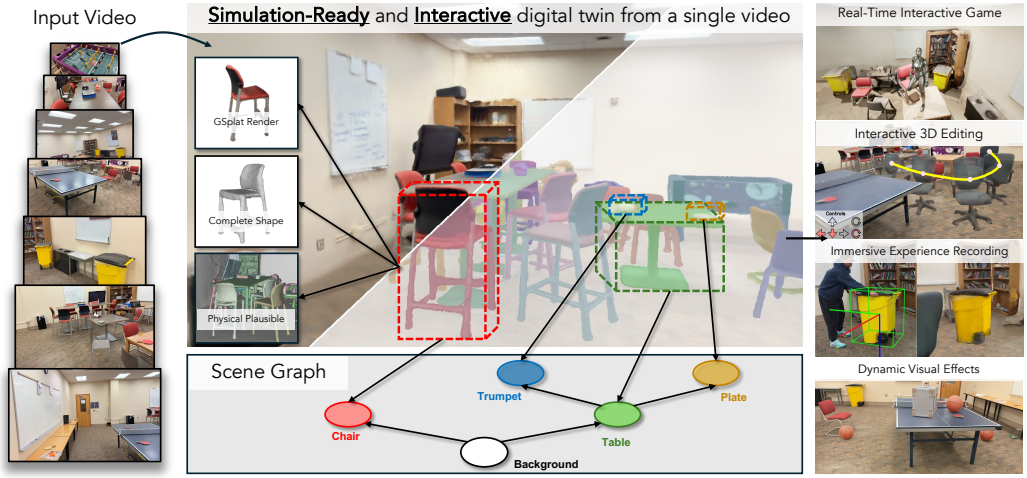

Figure 1: **Overview of HoloScene:** From a single input video—along with visual cues such as segmentation and monocular depth—HoloScene reconstructs a simulation-ready, interactive 3D digital twin represented as a scene graph with complete geometry, physically plausible dynamics, and realistic rendering. The resulting model enables a variety of downstream applications, including real-time interactive gaming, 3D editing, immersive experience capture, and dynamic visual effects.

Abstract

Digitizing the physical world into accurate simulation-ready virtual environments offers significant opportunities in a variety of fields such as augmented and virtual reality, gaming, and robotics. However, current 3D reconstruction and scene-understanding methods commonly fall short in one or more critical aspects, such as geometry completeness, object interactivity, physical plausibility, photorealistic rendering, or realistic physical properties for reliable dynamic simulation. To address these limitations, we introduce HoloScene, a novel interactive 3D reconstruction framework that simultaneously achieves these requirements. HoloScene leverages a comprehensive interactive scene-graph representation, encoding object geometry, appearance, and physical properties alongside hierarchical and inter-object relationships. Reconstruction is formulated as an energy-based optimization problem, integrating observational data, physical constraints, and generative priors into a unified, coherent objective. Optimization is efficiently performed via a hybrid approach combining sampling-based exploration with gradient-based refinement. The resulting digital twins exhibit complete and precise geometry, physical stability, and realistic rendering from novel viewpoints. Evaluations conducted on multiple benchmark datasets demonstrate superior performance, while practical use-cases in interactive gaming and real-time digital-twin manipulation illustrate HoloScene's broad applicability and effectiveness.

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

1 Introduction

Imagine wanting, decades later, to revisit the home you live in and love today—how would you capture its memory? Photographs and videos record authentic details but lack immersion; 3D Gaussian splats or photogrammetry can be immersive, yet static chairs and tables feel lifeless. Ideally, we would digitize our environment into a fully interactive digital twin: complete, composable, photorealistic, and manipulable just like the real world. Our work takes a step toward this goal by enabling users to create in silico twins of their surroundings from a single video.

Digitizing the physical world into a simulation-ready virtual environment offers immense opportunities in augmented and virtual reality, gaming, and robotics. However, despite advances in 3D modeling and scene understanding, key challenges remain: capturing complete geometry and appearance in occluded regions, inferring inter-object relationships, and ensuring physical plausibility and interactivity. Existing Real2Sim methods produce incomplete geometry [83, 43, 68] or unstable physics [73, 71]; existing amodal reconstruction focuses on single-image setting [81, 11], individual objects [72, 14, 37], neglects physical plausibility [47] or relies on asset retrieval [10] —sacrificing fidelity and practicality; and prior physically plausible reconstruction [46, 14] is limited to simple object–scene interactions or requires full observations.

To address these gaps, we introduce **HoloScene**, an interactive 3D reconstruction framework that unifies geometry completeness, object completeness, physical plausibility, realistic rendering, and physical interaction. HoloScene represents a scene as an interactive scene graph encoding object geometry, appearance, and physical properties in a hierarchical structure. We cast scene-graph recovery from video as a structured energy-based optimization, integrating observational data, physical constraints, and generative priors into a single objective. To solve this challenging problem, we propose a novel divide-and-conquer strategy combining sampling-based tree-structured search with gradient-based refinement. The resulting scene models exhibit complete, accurate geometry; stable physical interactions; and realistic rendering from novel viewpoints.

Experiments on three challenging benchmarks demonstrate superior geometry accuracy and physical plausibility, with rendering performance comparable to state-of-the-art amodal and physics-aware reconstruction methods. We further showcase HoloScene's versatility through practical applications in interactive gaming, realistic video effects, and real-time digital-twin manipulation.

2 Related Works

Interactive 3D Scene Model Recent advances in 3D scene modeling [50, 88, 16, 5, 65] reconstruct 3D scene from input images or videos, representing the scene as neural fields [44, 62, 18, 59, 58, 24, 3, 4], signed distance functions (SDF) [68, 82, 49, 51, 83, 86, 28], and 3D Gaussians [22, 19, 87, 85]. While producing realistic renderings from novel views, these works cannot provide 3D assets that allow user interactions (e.g, move the chairs to different poses). Reconstructing realistic and interactive environments from real images and videos remains challenging due to limited observation, occlusion, and physical reasoning. Some previous works reconstruct 3D objects from sparse viewpoints [37, 33, 75, 7, 39], and some estimate physical properties from visual observation [89, 14]. Nevertheless, these works focus on object-level tasks and cannot handle large and complex indoor scenes. PhyRecon [46] optimizes stable 3D scenes with differentiable physical engines, but does not model inter-object interactions and realistic appearance. DP-Recon [47], Video2Game [73], Drawer [74] leverage generative prior [1, 53] and foundation models [56, 57] to reconstruct decomposed 3D scenes. However, these works can only produce limited components or lack physical stability. In this work, we propose to reason object interaction with the scene graph, and utilize generative priors and a novel sampling strategy to reconstruct the geometry and appearance of every component, constructing realistic, physically plausible, and interactable 3D environments.

Method	Visual Input	Real-time Rendering	Amodal 3D Recon	Twin Fidelity	Physics Capacity	Physics Optimization
ACDC [10]	image	✓	✓	✗	✗	✗
Gen3DSR [11]	image	✓	✓	✓✗	✗	✗
PhysComp [14]	image	✗	✓	✓✗	Single Object	Differentiable
CAST [81]	image	✓	✓	✓✗	Scene	Differentiable
NeRF [43]	video	✗	✗	✗	✗	✗
BakedSDF [83]	video	✗	✗	✓	✗	✗
ObjectSDF++ [71]	video	✗	✗	✓	✗	✗
Video2Game [73]	video	✓	✗	✓	Single Object	✗
PhyRecon [46]	video	✗	✓	✓	Objects-Ground	Differentiable
DP-Recon [47]	video	✓	✓	✓	✗	✗
HoloScene (Ours)	video	✓	✓	✓	Scene	Diff & Sampling

Table 1: **Comparison of Interactive 3D Scene Models.**

Data-driven Simulation Simulation plays a pivotal role across robotics, self-driving, and content creation, but building high-fidelity virtual scenes remains costly, and the sim-to-real gap poses great challenges. To address this, data-driven simulation [2, 8, 38, 42, 60, 79] has emerged, enabling the modeling of physical dynamics [36, 27, 76, 20, 13, 21, 90], lighting conditions [31, 55, 30], and action-conditioned outcomes [20, 34, 6, 73, 21, 35, 74, 40], directly from real-world data. These methods have also been applied in robot learning [8, 42, 54, 78, 80, 79], LiDAR simulation [32, 42, 77, 79, 94, 93], and interactive media [17, 73]. In robotics, related *real-to-sim* approaches [9, 10, 64, 67, 26, 41] reconstruct interactable environments from the real world for reproducible embodiment. However, they still lack physical realism. Recent works [46, 6, 92] leverage differentiable physics or priors in reconstruction, but they neglect complex inter-object relationships. The closest work to ours is CAST [81], which also targets physically plausible scene reconstruction. The key differences are: (1) CAST takes single image and relies heavily on generative models, which may cause noticeable inconsistencies with the observation. (2) CAST uses differentiable optimization without feedback from physical simulators, so physical stability might not be guaranteed. In contrast, HoloScene reconstructs scenes from videos to replicate observations and adopts sampling-based optimization with Isaac Sim [45] to ensure physical stability. We compare HoloScene with prior works in Tab. 1.

3 Method

Given the observations $\mathcal{O} = \{\mathcal{O}_t\}_{t=0}^{T}$, which include the input video sequence $\{\mathbf{I}_t\}$, camera poses $\{\boldsymbol{\xi}_t\}$ (inferred or ground truth), and instance masks $\{\mathbf{M}_t\}$ (inferred or ground truth), our goal is to reconstruct a realistic, complete, and physically plausible digital twin of the input scene, yielding interactive, sim-ready assets compatible with simulators and game engines, and which can be used to generate novel visual content. To this end, we represent the scene as an interactive 3D scene graph representation that encodes object geometry, appearance, physical properties, and hierarchical inter-object relationships (Sec. 3.1). We combine observational evidence, generative priors for shape completion, and physical simulation for stability to formulate scene-graph recovery as an energy minimization problem (Sec. 3.2). Finally, we propose an inference method that integrates sampling-based tree search with differentiable optimization (Sec. 3.3). Fig.2 summarizes our approach.

3.1 Scene Representation

We represent the scene as an interactive 3D scene graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Each node $\mathbf{v}_i \in \mathcal{V} = \{\mathbf{v}_i\}_{i=0}^{N}$ represents either the background scene or one of the N objects present. A node $\mathbf{v}_i = (\mathbf{g}_i, \mathbf{f}_i, \mathbf{p}_i, \mathbf{T}_i)$ is comprised of geometry $\mathbf{g}_i$, appearance $\mathbf{f}_i$, physical properties $\mathbf{p}_i$, and dynamic states $\mathbf{T}_i$. Each edge $\mathbf{e}_{i,j} = (\mathbf{v}_i, \mathbf{v}_j) \in \mathcal{E}$ encodes an object–object relationship in $\mathcal{G}$.

Geometry: We represent the geometry of each node $\mathbf{v}_i$ in the scene with an instance-level neural SDF $g_i(\mathbf{x}; \theta) \colon \mathbb{R}^3 \to \mathbb{R}$, where $\mathbf{x} \in \mathbb{R}^3$ is any point in space and θ are learnable parameters. Additionally, to facilitate physical simulation and efficient rendering, we maintain a mesh representation $\mathcal{M}_i = \texttt{MarchingCube}(\texttt{g}_\texttt{i})$ for each object, extracted from its SDF using the marching cubes algorithm.

Appearance: For each object $\mathbf{v}_i$, we encode appearance $\mathbf{f}_i = (\mathbf{c}_i, \alpha_i, \boldsymbol{\mu}_i, \Sigma_i)$ as Gaussian splats, enabling real-time, high-quality rendering. $\mathbf{c}_i, \alpha_i, \boldsymbol{\mu}_i, \Sigma_i$ are color, opacity, mean, and covariance of Gaussians, respectively. Gaussians capture finer detail than colored meshes but hinder consistency between appearance, geometry, and simulation; following recent work [74], we adopt a Gaussians-on-Mesh (GoM) approach and attach each splat to its mesh to ensure alignment and enable physical interactions. Given camera intrinsics $\mathbf{K}$ and extrinsics $\boldsymbol{\xi}$, we denote the splat-rendered RGB images, masks, depth and normal maps as $\mathbf{I}, \mathbf{M}, \mathbf{D}, \mathbf{N} = \texttt{SplatRender}(\mathcal{G};\, \mathbf{K},\, \boldsymbol{\xi})$.

Physics: Each object in our scene graph is modeled as a rigid body. Its physical parameters $\mathbf{p}_i = (m_i, \kappa_i, \zeta_i, r_i)$ comprise mass m_i, friction κ_i (resistance to sliding against other surfaces), damping ζ_i (energy dissipation during motion), and restitution r_i (elasticity upon impact). These parameters are used in downstream physical simulations to model the object's response to external forces and its interactions with other objects and the background scene.

Object States: All object intrinsic attributes above are defined in object-centric coordinates and remain invariant under motion. To handle dynamic changes, we encode each object's rigid body state by a rigid transform $\mathbf{T}_i$ from its object-centric frame to the world frame. During static reconstruction, $\mathbf{T}_i$ is fixed; in dynamic simulation, it may vary over time. Let $\mathcal{T} = \{\mathbf{T}_i\}_{i=0}^{N}$ denote the set of all

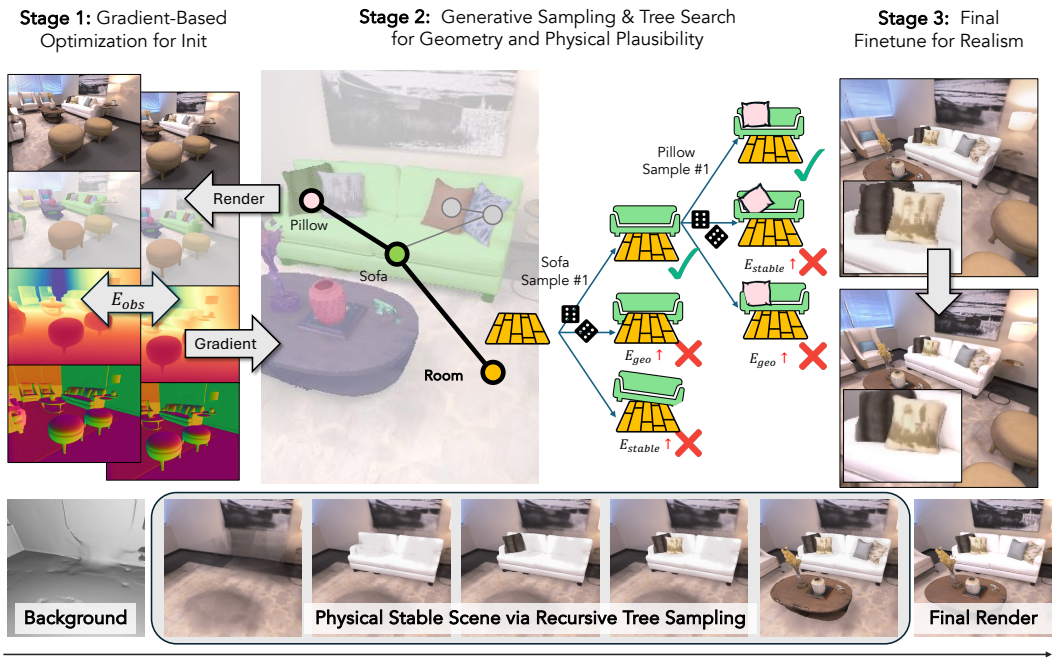

Figure 2: **Overview of HoloScene Optimization Stages:** Given multiple posed images as well as some visual cues (instance masks, monocular geometry priors), we first employ a gradient-based optimization as the initialization. Then we adopt a generative sampling and tree search strategy along the topology of the scene graph to obtain the complete geometry with physical plausibility. Finally, the final fine-tuning over the scene further enhances the realism of the reconstructed scene.

object states, $\mathcal{G}_{\mathcal{T}}$ the scene graph under those states, and $\mathcal{G}$ the scene graph under the static state during reconstruction.

Object Relationships: Each edge $\mathbf{e}_{i,j}$ links two nodes with one of three relationships: 1) `support`, where $\mathbf{v}_i$ rests in stable equilibrium on its unique parent $\mathbf{v}_{\mathrm{pa}(i)}$ under gravity (each object has exactly one such parent, so support edges form a tree in the static scene graph); 2) `beside`, where siblings ($\mathrm{pa}(j) = \mathrm{pa}(i)$) have touching surfaces, causing occlusions without hierarchy or instability; and 3) `collide`, where contacts with nonzero momentum yield dynamic effects—ignored during static reconstruction but employed in simulation. Note that the object relationship might change depending on its dynamic status during simulation.

Interaction & Simulation: Our 3D scene graph's distinguishing feature is its support for physical interactions. Formally, at time t, given the dynamic scene graph $\mathcal{G}_{\mathcal{T}^t}$ with current object states $\mathcal{T}^t$ as well as an input action $\mathbf{a}^t$, the next states are computed as

$$\mathcal{T}^{t+1} = \texttt{Sim}(\mathcal{T}^t, \mathbf{a}^t; \mathcal{G}_{\mathcal{T}^t}), \tag{1}$$

where `Sim` is a rigid-body physical simulator using the mesh $\{\mathcal{M}_i\}$ as collision geometry. Here, $\mathbf{a}^t$ can represent external inputs— forces, torques, or control actions—applied to the objects at time t.

3.2 Problem Formulation

Our framework takes input observations $\mathcal{O}$ of a static scene and recovers the scene graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. The resulting scene graph must (i) explain the observations well; (ii) be geometrically complete and plausible; and (iii) reflect the scene's static, physically stable nature. To this end, we cast the problem as a structured energy-minimization problem:

$$\min_{\mathcal{G}} \underbrace{E_{\mathrm{rgb}}(\mathcal{I}, \mathcal{G}) + E_{\mathrm{mask}}(\mathcal{M}, \mathcal{G}) + E_{\mathrm{mono}}(\mathcal{D}, \mathcal{G})}_{\text{observation terms}} + \underbrace{E_{\mathrm{comp}}(\mathcal{G}) + E_{\mathrm{geo}}(\mathcal{G}) + E_{\mathrm{physics}}(\mathcal{G})}_{\text{regularization terms}}. \tag{2}$$

For simplicity, we omit the hyperparameter linear weights for each term. Next, we discuss each energy term.

Observation Terms: The observation terms quantify the discrepancy between the reconstructed 3D scene and the input observations. Let $\hat{\mathbf{I}}_t, \hat{\mathbf{M}}_t, \hat{\mathbf{D}}_t, \hat{\mathbf{N}}_t = \texttt{SplatRender}(\mathcal{G}; \mathbf{K}, \boldsymbol{\xi}_t)$ denote the rendered RGB image, instance mask, depth map, and normal map at camera pose $\boldsymbol{\xi}_t$. We then define three energy terms: the **RGB energy** $E_{\text{rgb}}(\mathcal{I}, \mathcal{G}) = \sum_t \mathcal{L}_{\text{MSE}}(\hat{\mathbf{I}}_t, \mathbf{I}_t) + \mathcal{L}_{\text{LPIPS}}(\hat{\mathbf{I}}_t, \mathbf{I}_t)$, where $\mathbf{I}_t$ is the ground-truth color image and the loss combines MSE and LPIPS losses [91]; the **mask energy** $E_{\text{mask}}(\mathcal{M}, \mathcal{G}) = \sum_t \texttt{CE}(\hat{\mathbf{M}}_t, \mathbf{M}_t)$, where $\texttt{CE}$ is cross-entropy and $\mathbf{M}_t$ is either a given labeled mask [61, 84] or one inferred via segmentation tracking [23]; and the **monocular geometry energy** $E_{\text{mono}}(\mathcal{D}, \mathcal{G}) = \sum_t \|\hat{\mathbf{N}}_t - \mathbf{N}_t\|_2^2 + \mathcal{L}_{\text{norm}}(\hat{\mathbf{D}}_t, \mathbf{D}_t)$, where $\mathbf{N}_t$ and $\mathbf{D}_t$ are monocular normal and depth priors and $\mathcal{L}_{\text{norm}}$ is the scale- and shift-invariant L2 loss [86].

Regularization Terms: Because videos only partially observe a 3D scene, optimizing observations alone cannot yield a complete, plausible, and physically valid reconstruction; we therefore impose generative, geometric, and physical priors as regularizers to enable fully interactive 3D scenes.

The **completeness energy** E_{comp} encourages complete reconstruction of each object's shape despite the partial observations. Inspired by generative image-to-3D methods [37], for each object i we synthesize virtual observations $\tilde{\mathcal{O}}_i = \{\tilde{\mathcal{I}}_i, \tilde{\mathcal{D}}_i, \tilde{\mathcal{M}}_i, \tilde{\mathcal{N}}_i\}$ by "shooting" it from multiple virtual viewpoints with a pretrained multi-view diffusion model Wonder3D [37]. *Unlike* the single object setting for most image-to-3D works, because our complex scenes often feature inter-object occlusions (e.g., a sofa covered by a blanket), we first inpaint occluded regions using LaMa [63] before generating these views. Given the synthesized observations, we define the completeness energy as

$$E_{\text{comp}} = \sum_i \big(E_{\text{mask}}(\tilde{\mathcal{I}}_i, \{\mathbf{v}_i\}) + E_{\text{rgb}}(\tilde{\mathcal{M}}_i, \{\mathbf{v}_i\}) + E_{\text{mono}}(\tilde{\mathcal{D}}_i, \{\mathbf{v}_i\})\big), \tag{3}$$

where $E_{\text{rgb}}, E_{\text{mono}}, E_{\text{mask}}$ are the observation losses defined similarly in our observation terms, although they are measured at virtual viewpoints here.

The **geometry energy** E_{geo} ensures geometry compatibility between each object, such that their geometry does not intersect with each other:

$$E_{\text{geo}}(\mathcal{G}) = \sum_i \left(E_{\text{pene\_sdf}}(\mathbf{g}_i; \mathcal{G}) + E_{\text{pene\_mesh}}(\mathbf{g}_i; \mathcal{G})\right). \tag{4}$$

The SDF-penetration term $E_{\text{pene\_sdf}} = \sum_{\mathbf{x} \in R_i} \sum_{k \neq i} \max\big(0, -g_k(\mathbf{x}) - g_i(\mathbf{x})\big)$ ensures no two object SDFs overlap, where $R(i) = \{\mathbf{x} \in \mathbb{R}^3 | \arg\min_k g_k(\mathbf{x}) = i\}$ is the set of points belong to instance i. Intuitively, if $\mathbf{x}$ lies in instance i, then for any other instance k, $g_k(\mathbf{x}) \geq -g_i(\mathbf{x})$ must hold to prevent intersections. Similarly, each object's mesh should not intersect with any other object mesh. This can be measured by measuring whether intersecting two meshes resulting empty set or not: $E_{\text{pene\_mesh}} = \mathbf{1}(\texttt{inter}(\mathcal{M}_i, \mathcal{M}_j) \neq \emptyset)$.

Finally, it is important to ensure that our recovered digital twin of the scene is simulatable; hence, physical plausibility is crucial. To this end, we introduce **physics energy**, which measures physical plausibility via two terms:

$$E_{\text{physics}} = E_{\text{stable}} + E_{\text{touch}} = \texttt{Diff}\big(\mathcal{T}, \texttt{Sim}(\mathcal{T}, \mathbf{a}_{\text{gravity}}; \mathcal{G})\big) + \sum_{(i,j) \in \mathcal{E}} \texttt{dist}(\mathcal{M}_i, \mathcal{M}_j). \tag{5}$$

The stable term $E_{\text{stable}}(\mathcal{G}) = \texttt{Diff}\big(\mathcal{T}, \texttt{Sim}(\mathcal{T}, \mathbf{a}_{\text{gravity}}; \mathcal{G})\big)$ quantifies translational and rotational deviations of each object, with $\texttt{Diff}(\mathcal{T}, \mathcal{T}') = \sum_i (|\texttt{trans}(\mathbf{T}_i^{-1}\mathbf{T}_i')| + |\texttt{rad}(\mathbf{T}_i^{-1}\mathbf{T}_i')|)$ and $\texttt{Sim}$ is the forward physical simulator step as defined in Eq. 1; a low E_{stable} indicates static equilibrium under gravity, i.e. scene remains static in the simulator. The touch term $E_{\text{touch}} = \sum_{(i,j) \in \mathcal{E}} \texttt{dist}(\mathcal{M}_i, \mathcal{M}_j)$ encourages each supporting pair (i, j) to make contact, $\texttt{dist}$ is the Chamfer distance between meshes.

3.3 Inference

Optimizing the scene graph from Sec. 3.2 is challenging because it mixes discrete variables (graph topology, object–object relations) with continuous ones (neural SDFs, Gaussians, and physical parameters) and includes non-differentiable terms like physical stability. We therefore use a four-stage divide-and-conquer approach: first, infer topology via large foundation models; then, recover initial geometry and appearance by minimizing the observation terms; next, refine shapes and physical

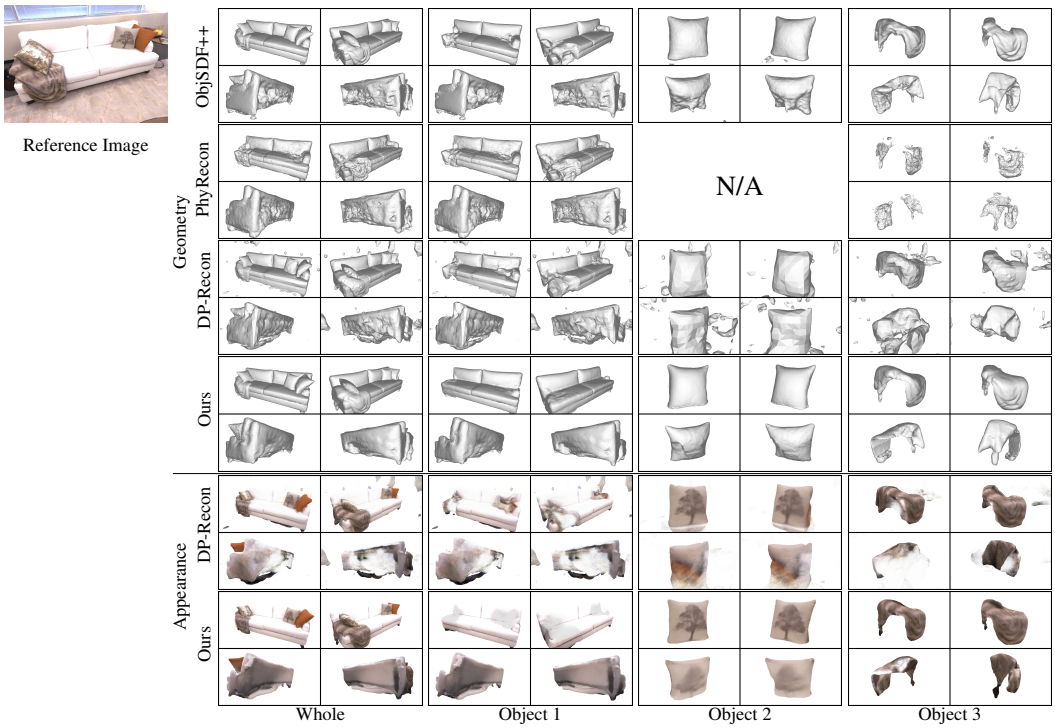

Figure 3: **Qualitative Comparisons on Object Geometry and Appearance Reconstruction:** Our method delivers superior reconstructions by smoothly inpainting occluded regions with LaMa and completing invisible back-facing geometry with Wonder3D. Unlike baselines, our approach eliminates object interpenetration, ensuring physical stability during simulation.

parameters by minimizing the geometry and physics terms through generative sampling combined with structured tree search; and finally, fine-tune appearance by re-minimizing observation terms. This yields a fully plausible, interactive 3D scene (Fig. 2).

Stage 0: Scene Graph Edges: Our framework infers the topology of the scene-graph $\mathcal{G}$ from observations $\mathcal{O}$, where edges $\mathcal{E}$ encode support relations in a tree rooted at $\mathbf{v}_0$ (the background, e.g., the room). We build this tree recursively using a VLM: at each step, we overlay the masks of already registered instances (annotated with their IDs) as a visual prompt, then ask the VLM to identify and register one new unseen instance and infer its physical relationship to the objects already in the tree. Starting from $\mathbf{v}_0$ in the first frame, we repeat until all observed instances have been added to the tree.

Stage 1: Gradient-based Optimization: After obtaining the scene-graph topology, we optimize each node's appearance $\mathbf{a}_i$ and geometry g_i to match the observations $\mathcal{O}$ via gradient-based optimization. Specifically, we minimize the observation terms plus SDF-penetration regularization through differentiable volume rendering—similar to neural SDF methods [71, 70, 29]—to obtain per-instance SDFs g_i. Additionally, we recover small objects by balancing training samples across all instances. We then extract initial meshes $\mathcal{M}_i$ via marching cubes and refine each object's Gaussians $\mathbf{f}_i$ via splat rendering and RGB rendering, mask, and monocular geometry losses, yielding our dual scene representation per each instance [15].

Stage 2: Sampling-based Optimization: The Stage 1 scene model supports freeview rendering and accurate visible-region geometry but remains incomplete, non–physical, and non–interactive. Directly minimizing E_{complete}, E_{physics}, and E_{geo}, however, is challenging due to complex high-order interactions (e.g., multi-object physical interaction), intrinsic multi-modality (invisible regions admit multiple solutions), and non-differentiable components (e.g., mesh intersections, physics simulations). To address this, we adopt an approach that combines the diverse proposal capability of **generative sampling** with the combinatorial optimization strength of **tree search** to minimize our structured objective.

Generative sampling: We begin by sampling diverse, complete shapes for each instance: we prompt Wonder3D's multi-diffusion model with real-world observations and generate virtual views $\tilde{\mathcal{I}}_i$ from

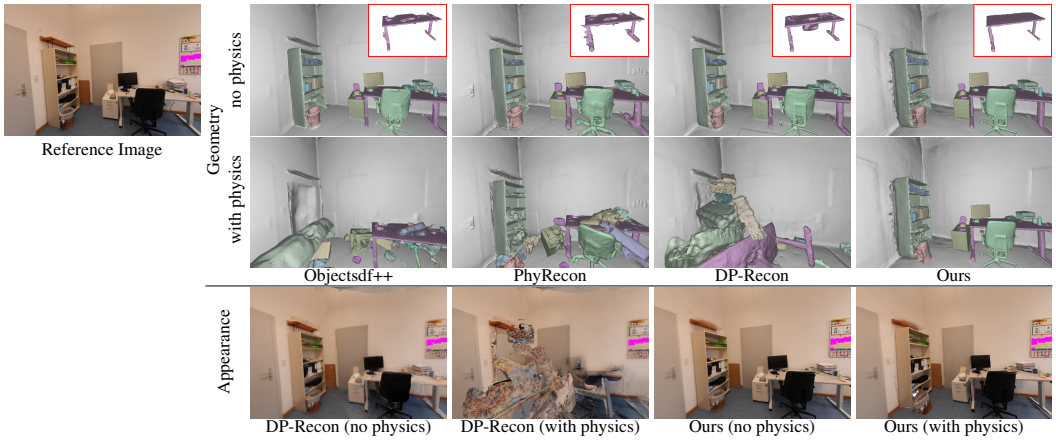

Figure 4: **Qualitative Comparisons on Physical Simulation:** We compare geometry layouts and appearance before and after physical simulation, with the table geometry reconstructions highlighted in inset figures. HoloScene's complete, non-interpenetrating geometry remains stable in physics simulators, unlike baseline methods. Our Gaussian on mesh delivers high-quality, real-time rendering throughout the simulation process.

various viewpoints. Thanks to its generative nature, resampling multiple times with different seeds yields diverse virtual observations even from the same viewpoints. Then, for each virtual observation, we minimize E_{comp} independently, producing a diverse set of 3D shape candidates per object.

Structured tree search: We have generated multiple complete shape samples per instance that all fit the observations, but it remains unclear which combination is most physically plausible. Exhaustively evaluating every combination is impractical, since the physical-plausibility energy E_{physics} entails a high-order combinatorial optimization. To address this, we perform a tree search over our generative samples. Starting at the root node, we traverse each node in breadth-first order; at each active node (object), we evaluate E_{physics} for all samples and retain the sample with the lowest energy among those evaluated. We then adjust its state and physical parameters to enforce stability and prevent interpenetration (see details in the supplementary material).

Remark: The key novelty and advantage of the proposed inference algorithm is the combination of generative sampling with a structured tree search for amodal, physically plausible reconstruction. Unlike scene-level amodal methods [10] that rely on asset retrieval, our sampling is asset-free and generates input consistent, diverse shape hypotheses. Unlike prior simulation-verification methods [47, 46], which only enforces object–ground consistency, our tree search ensures global stability along every support chain. By driving a non-differentiable simulator (e.g., IsaacSim) end-to-end, we eliminate any reconstruction-to-deployment gap.

Stage 3: Gradient-based Refinement Since Stage 2 adjusts object states, physical parameters, and shape, it is necessary to further refine the Gaussians attached to the surface to ensure a complete and realistic appearance. To this end, we fine-tune Gaussians for all objects using splat rendering by minimizing the observation terms via gradient descent. This yields our final scene graph.

4 Experiments

Dataset: We conduct the experiments across multiple datasets: 7 scenes from Replica [61], 6 scenes from Scannet++ [84], 2 scenes from iGibson [25], and one self-captured scene. The Replica and Scannet++ datasets cover diverse indoor structures and lighting conditions, and the iGibson dataset offers complete geometry of every object, allowing per-object reconstruction evaluation. Instance masks are provided by dataset annotations or estimated with SAM [23].

Metrics: We evaluate geometry quality with Chamfer Distance (CD), F-Score (F1), and Normal Consistency (NC) [86], and assess rendering quality using PSNR, SSIM, and LPIPS. For physical evaluation, we adopt consistent physical parameters, put all scene components in the Isaac Sim [45], and measure stability with translation/rotation changes when gravity is applied. The stability ratio

		Method	Geometry			Rendering				Physics		
			CD↓	F1↑	NC↑	PSNR↑	SSIM↑	LPIPS↓	Real Time	OR% ↑	Stable (Ground) % ↑	Stable (All) % ↑
Scene	Replica	ObjSDF++	6.72	64.36	88.53	29.12	0.851	0.355	✗	98.6	78.3	39.4
		PhyRecon	4.52	71.07	92.06	23.19	0.764	0.434	✗	77.5	56.5	5.6
		DP-Recon	3.45	87.66	94.23	22.10	0.728	0.420	✓	56.3	21.7	8.5
		Ours	4.05	83.21	92.21	27.82	0.849	0.304	✓	100.0	95.7	81.7
	Scannet++	ObjSDF++	25.20	70.71	87.15	27.46	0.887	0.292	✗	96.5	81.6	28.2
		PhyRecon	31.16	39.57	82.28	22.32	0.791	0.432	✗	92.9	67.3	9.4
		DP-Recon	22.96	65.48	87.13	21.44	0.715	0.466	✓	90.6	20.0	9.4
		Ours	21.93	63.11	88.09	25.88	0.873	0.268	✓	100.0	93.9	70.6
	iGibson	ObjSDF++	12.33	38.64	83.74	29.60	0.891	0.299	✗	65.0	44.2	36.1
		PhyRecon	11.27	45.49	83.85	27.40	0.860	0.333	✗	62.9	45.3	5.2
		DP-Recon	30.31	21.89	70.81	21.94	0.728	0.432	✓	74.2	16.3	4.1
		Ours	12.00	34.15	82.91	25.88	0.854	0.301	✓	100.0	74.4	71.1
Object	iGibson	ObjSDF++	3.52	79.03	75.30	11.03	0.571	0.134	✗	65.0	44.2	36.1
		PhyRecon	5.47	70.71	71.89	8.92	0.609	0.250	✗	62.9	45.3	5.2
		DP-Recon	5.81	61.31	70.61	13.90	0.770	0.301	✓	74.2	16.3	4.1
		Ours	3.17	81.31	78.13	16.55	0.863	0.185	✓	100.0	74.4	71.1

Table 2: **Quantitative Results on Scene Reconstruction:** HoloScene's generative sampling and scene graph-based tree search produce the most physically plausible reconstructions while preserving high-quality geometry and supporting real-time and realistic rendering. We highlight the best and second-best methods with distinct colors. For rendering quality, we only compare with DP-Recon's texture mesh-based rendering since ObjSDF++ and PhyRecon lack real-time rendering capabilities.

is calculated as: Stable $\% = \frac{\#\text{Stable Instances}}{\#\text{All Instances}}$, where instances are identified as stable if changes are under a certain threshold. We also report the object reconstruction ratio: $\text{OR}\% = \frac{\#\text{Reconstructed Instances}}{\#\text{All Instances}}$, indicating the completeness of whole scenes.

Baselines: We evaluate our framework against SOTA approaches in instance-aware amodal 3D scene reconstruction. **ObjectSDF++** [71] uses per-instance SDF for scene representation. **PhyRecon** [46] extends instance-aware scene reconstruction by incorporating a differentiable physical loss to optimize unstable objects. However, it only handles object–ground interactions and does not support hierarchical or inter–object relationships. **DP-Recon** [47] incorporates diffusion Score Distillation Sampling (SDS) [53] for amodal sparse-view reconstruction. However, it does not handle inter-object occlusions. All three methods use instance-level SDFs, so the SDF values of different instances might interfere with one another. We adopt their open-source codes and adapt them for the testing benchmarks. Please refer to the supplementary material for more implementation details.

4.1 Experimental Results

Scene-level evaluations: We first evaluate the reconstruction at the scene level. In Tab. 2, we compare our framework with the three baselines across Replica, Scannet++. and iGibson dataset, with qualitative results shown in Fig. 3 and Fig. 4. Our method achieves the best physical reconstruction results in terms of both object reconstruction ratio and stable object ratio, while maintaining comparable scene-level reconstruction quality in geometry and appearance. Unlike baselines [71, 46, 48] where small objects often disappear due to SDF interference from larger adjacent objects, our framework benefits from balancing training samples across all instances during the optimization stage 1, recovering all instances in the scenes. PhyRecon assumes that all objects rest directly on the ground, which damages geometry when objects are supported by other objects. DP-Recon prioritizes completeness via SDS over physical stability, failing to adequately address object interpenetration. In contrast, our sampling-based optimization and completion approach yields the most physically stable results.

Object-level evaluations: We evaluate object-level reconstruction using the iGibson dataset, which provides complete ground truth geometry and appearance for each object. This evaluation is especially challenging as it tests the reconstruction of occluded regions that models never observe. Our evaluation compares reconstructed geometry directly with ground truth and renders 6 viewpoints around each object to assess appearance quality. Results in Tab. 2 demonstrate that our method outperforms all baselines in reconstructing invisible and occluded regions, validating our framework's effectiveness in completing objects beyond directly observed surfaces.

Level	Dataset	TexMesh / GoM	Physical Energy	Scene Graph	CD↓	F1↑	NC↑	PSNR↑	SSIM↑	LPIPS↓	OR% ↑	Stable (All) % ↑
Scene	Replica	TexMesh	✗	✗	3.50	88.00	92.30	26.45	0.810	0.329	100.0	43.7
		GoM	✗	✗	3.50	88.00	92.30	28.30	0.845	0.349	100.0	43.7
		GoM	✓	✗	4.32	80.19	92.10	27.14	0.839	0.334	100.0	69.0
		GoM	✓	✓	4.05	83.21	92.21	27.82	0.849	0.304	100.0	81.7
Object	iGibson	GoM	✗	✗	4.20	79.22	76.96	13.74	0.735	0.204	100.0	41.2
		GoM	✓	✓	3.17	81.31	78.13	16.55	0.863	0.185	100.0	71.1

Table 3: **Ablation Study on Model Design:** We observe improved physical stability and object-level reconstruction with the help of generative priors, physical energy, and scene graph representation, and there is a trade-off between scene-level reconstruction and instance-level physical stability.

Ablation study: Our ablation study (Tab. 3) reveals the contribution of each component. Switching from textured mesh to Gaussian rendering improves visual quality, while adding physics energy with generative priors enhances physical stability. The scene graph inter-object relationships further improve physics performance by better reconstructing occluded regions, especially those from `support` relationships. However, we identify a trade-off between scene-level reconstruction accuracy and physical stability, where optimizing for physical plausibility may occasionally compromise pixel-alignment with original observations.

4.2 Interactive Environment Applications

Real-Time Interactive Game With our reconstructed environment, we can create a real-time interactive game with Unreal Engine [12]. As Fig. 1 shows, we build a third-person game with the reconstructed texture meshes. The objects could be physically rearranged in the game world, and the game agent could also interact with the scene through realistic physics.

Interactive 3D Editing In our simulation environment, we could also achieve high-quality interactive 3D editing by moving the object Gaussians with its underlying physical mesh geometry. In Fig. 1, we demonstrate this by changing the location and orientation of the interactable chair.

Immersive Experience Recording We show our interactable reconstructed 3D objects with immersive experience recording. Given a static RGB video of a person manipulating an object, we aim to recover the object's 6D pose and resimulate its motion in a virtual 3D scene. We recover the camera pose with VGGT [66], adjust the predicted depth [52] to align with the virtual scene, and adopt FoundationPose [69] for object tracking with our reconstructed 3D object for model-based 6D pose estimation. As shown in Fig. 1, we enable consistent replay of real-world interactions in virtual scenes while accurately recovering the object's pose from visual input.

Dynamic Visual Effects To enhance immersion, we augment the scene with dynamic visual effects, including rigid body simulations, character animations, and particle effects. We adopt visual effects from AutoVFX [17] to overlay virtual content and shadows onto the image. As Fig. 5 shows, we produce effects that blend naturally with the scene.

Figure 5: **Dynamic VFX Results.** We augment the inferred interactive 3D scene with various visual effects such as dropping objects, adding animations, and fires.

5 Conclusion & Limitation

We presented HoloScene, a novel interactive 3D modeling framework that uses scene graphs and energy-based optimization to reconstruct environments with realistic appearance, complete geometry, and interactive physical plausibility, achieving superior real-time rendering, geometric accuracy, and stable simulation. **Limitations**: HoloScene currently only handles videos of static indoor scenes; dynamic scenes and large outdoor environments remain challenging. Future work will focus on relightable reconstruction and extending support to articulated and deformable objects.

References

[1] Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)

[2] Amini, A., Wang, T.H., Gilitschenski, I., Schwarting, W., Liu, Z., Han, S., Karaman, S., Rus, D.: Vista 2.0: An open, data-driven simulator for multimodal sensing and policy learning for autonomous vehicles. In: ICRA (2022)

[3] Barron, J.T., Mildenhall, B., Tancik, M., Hedman, P., Martin-Brualla, R., Srinivasan, P.P.: Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In: ICCV (2021)

[4] Barron, J.T., Mildenhall, B., Verbin, D., Srinivasan, P.P., Hedman, P.: Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In: CVPR (2022)

[5] Berger, M., Tagliasacchi, A., Seversky, L.M., Alliez, P., Levine, J.A., Sharf, A., Silva, C.T.: State of the art in surface reconstruction from point clouds. In: 35th Annual Conference of the European Association for Computer Graphics, Eurographics 2014-State of the Art Reports. The Eurographics Association (2014)

[6] Chen, B., Jiang, H., Liu, S., Gupta, S., Li, Y., Zhao, H., Wang, S.: Physgen3d: Crafting a miniature interactive world from a single image. arXiv preprint arXiv:2503.20746 (2025)

[7] Chen, Y., Ni, J., Jiang, N., Zhang, Y., Zhu, Y., Huang, S.: Single-view 3D scene reconstruction with high-fidelity shape and texture. In: 2024 International Conference on 3D Vision (3DV). IEEE (2024)

[8] Chen, Y., Rong, F., Duggal, S., Wang, S., Yan, X., Manivasagam, S., Xue, S., Yumer, E., Urtasun, R.: Geosim: Realistic video simulation via geometry-aware composition for self-driving. In: CVPR (2021)

[9] Chen, Z., Walsman, A., Memmel, M., Mo, K., Fang, A., Vemuri, K., Wu, A., Fox, D., Gupta, A.: Urdformer: A pipeline for constructing articulated simulation environments from real-world images. arXiv (2024)

[10] Dai, T., Wong, J., Jiang, Y., Wang, C., Gokmen, C., Zhang, R., Wu, J., Fei-Fei, L.: Acdc: Automated creation of digital cousins for robust policy learning. arXiv (2024)

[11] Dogaru, A., Özer, M., Egger, B.: Generalizable 3d scene reconstruction via divide and conquer from a single view. arXiv preprint arXiv:2404.03421 (2024)

[12] Epic Games: Unreal engine `https://www.unrealengine.com`

[13] Feng, Y., Feng, X., Shang, Y., Jiang, Y., Yu, C., Zong, Z., Shao, T., Wu, H., Zhou, K., Jiang, C., et al.: Gaussian splashing: Dynamic fluid synthesis with gaussian splatting. arXiv e-prints pp. arXiv–2401 (2024)

[14] Guo, M., Wang, B., Ma, P., Zhang, T., Owens, C., Gan, C., Tenenbaum, J., He, K., Matusik, W.: Physically compatible 3d object modeling from a single image. Advances in Neural Information Processing Systems **37**, 119260–119282 (2024)

[15] Guédon, A., Lepetit, V.: Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering (2023)

[16] Han, X.F., Laga, H., Bennamoun, M.: Image-based 3d object reconstruction: State-of-the-art and trends in the deep learning era. IEEE transactions on pattern analysis and machine intelligence **43**(5), 1578–1604 (2019)

[17] Hsu, H.Y., Lin, Z.H., Zhai, A., Xia, H., Wang, S.: Autovfx: Physically realistic video editing from natural language instructions. arXiv preprint arXiv:2411.02394 (2024)

[18] Hu, S., Zhou, K., Li, K., Yu, L., Hong, L., Hu, T., Li, Z., Lee, G.H., Liu, Z.: Consistentnerf: Enhancing neural radiance fields with 3d consistency for sparse view synthesis. arXiv preprint arXiv:2305.11031 (2023)

[19] Huang, B., Yu, Z., Chen, A., Geiger, A., Gao, S.: 2d gaussian splatting for geometrically accurate radiance fields. In: SIGGRAPH. Association for Computing Machinery (2024)

[20] Jiang, H., Hsu, H.Y., Zhang, K., Yu, H.N., Wang, S., Li, Y.: Phystwin: Physics-informed reconstruction and simulation of deformable objects from videos. arXiv preprint arXiv:2503.17973 (2025)

[21] Jiang, Y., Yu, C., Xie, T., Li, X., Feng, Y., Wang, H., Li, M., Lau, H., Gao, F., Yang, Y., et al.: Vr-gs: A physical dynamics-aware interactive gaussian splatting system in virtual reality. In: ACM SIGGRAPH 2024 Conference Papers. pp. 1–1 (2024)

[22] Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. TOG (2023)

[23] Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., et al.: Segment anything. In: ICCV (2023)

[24] Kwak, M.S., Song, J., Kim, S.: Geconerf: Few-shot neural radiance fields via geometric consistency. arXiv preprint arXiv:2301.10941 (2023)

[25] Li, C., Xia, F., Martín-Martín, R., Lingelbach, M., Srivastava, S., Shen, B., Vainio, K., Gokmen, C., Dharan, G., Jain, T., et al.: igibson 2.0: Object-centric simulation for robot learning of everyday household tasks. arXiv preprint arXiv:2108.03272 (2021)

[26] Li, X., Li, J., Zhang, Z., Zhang, R., Jia, F., Wang, T., Fan, H., Tseng, K.K., Wang, R.: Robogsim: A real2sim2real robotic gaussian splatting simulator. arXiv preprint arXiv:2411.11839 (2024)

[27] Li, Y., Lin, Z.H., Forsyth, D., Huang, J.B., Wang, S.: Climatenerf: Physically-based neural rendering for extreme climate synthesis. arXiv (2022)

[28] Li, Z., Müller, T., Evans, A., Taylor, R.H., Unberath, M., Liu, M.Y., Lin, C.H.: Neuralangelo: High-fidelity neural surface reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8456–8465 (2023)

[29] Li, Z., Lyu, X., Ding, Y., Wang, M., Liao, Y., Liu, Y.: Rico: Regularizing the unobservable for indoor compositional reconstruction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 17761–17771 (2023)

[30] Lin, Z.H., Huang, J.B., Li, Z., Dong, Z., Richardt, C., Li, T., Zollhöfer, M., Kopf, J., Wang, S., Kim, C.: Iris: Inverse rendering of indoor scenes from low dynamic range images. arXiv preprint arXiv:2401.12977 (2024)

[31] Lin, Z.H., Liu, B., Chen, Y.T., Forsyth, D., Huang, J.B., Bhattad, A., Wang, S.: Urbanir: Large-scale urban scene inverse rendering from a single video (2023)

[32] Liu, J.Y., Chen, Y., Yang, Z., Wang, J., Manivasagam, S., Urtasun, R.: Real-time neural rasterization for large scenes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8416–8427 (2023)

[33] Liu, R., Wu, R., Van Hoorick, B., Tokmakov, P., Zakharov, S., Vondrick, C.: Zero-1-to-3: Zero-shot one image to 3d object (2023)

[34] Liu, S., Ren, Z., Gupta, S., Wang, S.: Physgen: Rigid-body physics-grounded image-to-video generation. In: European Conference on Computer Vision. pp. 360–378. Springer (2024)

[35] Liu, Y., Jia, B., Lu, R., Ni, J., Zhu, S.C., Huang, S.: Building interactable replicas of complex articulated objects via gaussian splatting. arXiv preprint arXiv:2502.19459 (2025)

[36] Liu, Z., Zhou, G., He, J., Marcucci, T., Li, F.F., Wu, J., Li, Y.: Model-based control with sparse neural dynamics. Advances in Neural Information Processing Systems **36**, 6280–6296 (2023)

[37] Long, X., Guo, Y.C., Lin, C., Liu, Y., Dou, Z., Liu, L., Ma, Y., Zhang, S.H., Habermann, M., Theobalt, C., et al.: Wonder3d: Single image to 3d using cross-domain diffusion. arXiv preprint arXiv:2310.15008 (2023)

[38] Lu, F., Xu, Y., Chen, G., Li, H., Lin, K.Y., Jiang, C.: Urban radiance field representation with deformable neural mesh primitives (2023)

[39] Lu, R., Chen, Y., Ni, J., Jia, B., Liu, Y., Wan, D., Zeng, G., Huang, S.: Movis: Enhancing multi-object novel view synthesis for indoor scenes. arXiv preprint arXiv:2412.11457 (2024)

[40] Luo, R., Geng, H., Deng, C., Li, P., Wang, Z., Jia, B., Guibas, L., Huang, S.: Physpart: Physically plausible part completion for interactable objects. arXiv preprint arXiv:2408.13724 (2024)

[41] Ma, S., Du, W., Yu, C., Jiang, Y., Zong, Z., Xie, T., Chen, Y., Yang, Y., Han, X., Jiang, C.: Grip: A general robotic incremental potential contact simulation dataset for unified deformable-rigid coupled grasping. arXiv preprint arXiv:2503.05020 (2025)

[42] Manivasagam, S., Wang, S., Wong, K., Zeng, W., Sazanovich, M., Tan, S., Yang, B., Ma, W.C., Urtasun, R.: Lidarsim: Realistic lidar simulation by leveraging the real world. In: CVPR (2020)

[43] Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. In: ECCV (2020)

[44] Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. ACM Communications (2021)

[45] Mittal, M., Yu, C., Yu, Q., Liu, J., Rudin, N., Hoeller, D., Yuan, J.L., Singh, R., Guo, Y., Mazhar, H., Mandlekar, A., Babich, B., State, G., Hutter, M., Garg, A.: Orbit: A unified simulation framework for interactive robot learning environments. IEEE Robotics and Automation Letters **8**(6), 3740–3747 (2023). https://doi.org/10.1109/LRA.2023.3270034

[46] Ni, J., Chen, Y., Jing, B., Jiang, N., Wang, B., Dai, B., Li, P., Zhu, Y., Zhu, S.C., Huang, S.: Phyrecon: Physically plausible neural scene reconstruction. arXiv preprint arXiv:2404.16666 (2024)

[47] Ni, J., Liu, Y., Lu, R., Zhou, Z., Zhu, S.C., Chen, Y., Huang, S.: Decompositional neural scene reconstruction with generative diffusion prior (2025)

[48] Ni, J., Liu, Y., Lu, R., Zhou, Z., Zhu, S.C., Chen, Y., Huang, S.: Decompositional neural scene reconstruction with generative diffusion prior. arXiv preprint arXiv:2503.14830 (2025)

[49] Oechsle, M., Peng, S., Geiger, A.: Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5589–5599 (2021)

[50] Özsoy, E., Czempiel, T., Örnek, E.P., Eck, U., Tombari, F., Navab, N.: Holistic or domain modeling: a semantic scene graph approach. International Journal of Computer Assisted Radiology and Surgery **19**(5), 791–799 (2024)

[51] Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S.: Deepsdf: Learning continuous signed distance functions for shape representation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 165–174 (2019)

[52] Piccinelli, L., Sakaridis, C., Yang, Y.H., Segu, M., Li, S., Abbeloos, W., Van Gool, L.: Unidepthv2: Universal monocular metric depth estimation made simpler. arXiv preprint arXiv:2502.20110 (2025)

[53] Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: Dreamfusion: Text-to-3D using 2D diffusion. arXiv preprint arXiv:2209.14988 (2022)

[54] Pun, A., Sun, G., Wang, J., Chen, Y., Yang, Z., Manivasagam, S., Ma, W.C., Urtasun, R.: Lightsim: Neural lighting simulation for urban scenes (2023), `https://arxiv.org/abs/2312.06654`

[55] Pun, A., Sun, G., Wang, J., Chen, Y., Yang, Z., Manivasagam, S., Ma, W.C., Urtasun, R.: Neural lighting simulation for urban scenes. NeurIPS (2024)

[56] Qian, S., Fouhey, D.F.: Understanding 3D object interaction from a single image. In: ICCV (2023)

[57] Ren, T., Liu, S., Zeng, A., Lin, J., Li, K., Cao, H., Chen, J., Huang, X., Chen, Y., Yan, F., Zeng, Z., Zhang, H., Li, F., Yang, J., Li, H., Jiang, Q., Zhang, L.: Grounded sam: Assembling open-world models for diverse visual tasks (2024), `https://arxiv.org/abs/2401.14159`

[58] Seo, S., Han, D., Chang, Y., Kwak, N.: Mixnerf: Modeling a ray with mixture density for novel view synthesis from sparse inputs. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20659–20668 (2023)

[59] Somraj, N., Soundararajan, R.: Vip-nerf: Visibility prior for sparse input neural radiance fields. In: ACM SIGGRAPH 2023 Conference Proceedings. pp. 1–11 (2023)

[60] Son, S., Qiao, Y.L., Sewall, J., Lin, M.C.: Differentiable hybrid traffic simulation. TOG (2022)

[61] Straub, J., Whelan, T., Ma, L., Chen, Y., Wijmans, E., Green, S., Engel, J.J., Mur-Artal, R., Ren, C., Verma, S., Clarkson, A., Yan, M., Budge, B., Yan, Y., Pan, X., Yon, J., Zou, Y., Leon, K., Carter, N., Briales, J., Gillingham, T., Mueggler, E., Pesqueira, L., Savva, M., Batra, D., Strasdat, H.M., Nardi, R.D., Goesele, M., Lovegrove, S., Newcombe, R.: The Replica dataset: A digital replica of indoor spaces. arXiv preprint arXiv:1906.05797 (2019)

[62] Sucar, E., Liu, S., Ortiz, J., Davison, A.J.: imap: Implicit mapping and positioning in real-time. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 6229–6238 (2021)

[63] Suvorov, R., Logacheva, E., Mashikhin, A., Remizova, A., Ashukha, A., Silvestrov, A., Kong, N., Goka, H., Park, K., Lempitsky, V.: Resolution-robust large mask inpainting with fourier convolutions. arXiv preprint arXiv:2109.07161 (2021)

[64] Torne, M., Simeonov, A., Li, Z., Chan, A., Chen, T., Gupta, A., Agrawal, P.: Reconciling reality through simulation: A real-to-sim-to-real approach for robust manipulation. arXiv (2024)

[65] Wang, H., Li, M.: A new era of indoor scene reconstruction: A survey. IEEE Access (2024)

[66] Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., Novotny, D.: Vggt: Visual geometry grounded transformer. arXiv preprint arXiv:2503.11651 (2025)

[67] Wang, L., Guo, R., Vuong, Q., Qin, Y., Su, H., Christensen, H.: A real2sim2real method for robust object grasping with neural surface reconstruction. In: 2023 IEEE 19th International Conference on Automation Science and Engineering (CASE). pp. 1–8. IEEE (2023)

[68] Wang, P., Liu, L., Liu, Y., Theobalt, C., Komura, T., Wang, W.: Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. arXiv (2021)

[69] Wen, B., Yang, W., Kautz, J., Birchfield, S.: Foundationpose: Unified 6d pose estimation and tracking of novel objects. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 17868–17879 (2024)

[70] Wu, Q., Liu, X., Chen, Y., Li, K., Zheng, C., Cai, J., Zheng, J.: Object-compositional neural implicit surfaces. In: European Conference on Computer Vision. pp. 197–213. Springer (2022)

[71] Wu, Q., Wang, K., Li, K., Zheng, J., Cai, J.: Objectsdf++: Improved object-compositional neural implicit surfaces. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 21764–21774 (2023)

[72] Wu, T., Zheng, C., Guan, F., Vedaldi, A., Cham, T.J.: Amodal3r: Amodal 3d reconstruction from occluded 2d images. arXiv preprint arXiv:2503.13439 (2025)

[73] Xia, H., Lin, Z.H., Ma, W.C., Wang, S.: Video2game: Real-time interactive realistic and browser-compatible environment from a single video. In: CVPR (2024)

[74] Xia, H., Su, E., Memmel, M., Jain, A., Yu, R., Mbiziwo-Tiapo, N., Farhadi, A., Gupta, A., Wang, S., Ma, W.C.: Drawer: Digital reconstruction and articulation with environment realism (2025), https://arxiv.org/abs/2504.15278

[75] Xiang, J., Lv, Z., Xu, S., Deng, Y., Wang, R., Zhang, B., Chen, D., Tong, X., Yang, J.: Structured 3d latents for scalable and versatile 3d generation. arXiv preprint arXiv:2412.01506 (2024)

[76] Xie, T., Zong, Z., Qiu, Y., Li, X., Feng, Y., Yang, Y., Jiang, C.: Physgaussian: Physics-integrated 3d gaussians for generative dynamics. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4389–4398 (2024)

[77] Xiong, Y., Ma, Wei-Chiu Wang, J., Urtasun, R.: Ultralidar: Learning compact representations for lidar completion and generation. CVPR (2023)

[78] Yang, J., Ivanovic, B., Litany, O., Weng, X., Kim, S.W., Li, B., Che, T., Xu, D., Fidler, S., Pavone, M., Wang, Y.: Emernerf: Emergent spatial-temporal scene decomposition via self-supervision. In: ICLR (2024)

[79] Yang, Z., Chen, Y., Wang, J., Manivasagam, S., Ma, W.C., Yang, A.J., Urtasun, R.: Unisim: A neural closed-loop sensor simulator. CVPR (2023)

[80] Yang, Z., Chai, Y., Anguelov, D., Zhou, Y., Sun, P., Erhan, D., Rafferty, S., Kretzschmar, H.: Surfelgan: Synthesizing realistic sensor data for autonomous driving. In: CVPR (2020)

[81] Yao, K., Zhang, L., Yan, X., Zeng, Y., Zhang, Q., Xu, L., Yang, W., Gu, J., Yu, J.: Cast: Component-aligned 3D scene reconstruction from an rgb image. arXiv preprint arXiv:2502.12894 (2025)

[82] Yariv, L., Gu, J., Kasten, Y., Lipman, Y.: Volume rendering of neural implicit surfaces. Advances in Neural Information Processing Systems **34**, 4805–4815 (2021)

[83] Yariv, L., Hedman, P., Reiser, C., Verbin, D., Srinivasan, P.P., Szeliski, R., Barron, J.T., Mildenhall, B.: Bakedsdf: Meshing neural sdfs for real-time view synthesis. In: SIGGRAPH Conference (2023)

[84] Yeshwanth, C., Liu, Y.C., Nießner, M., Dai, A.: Scannet++: A high-fidelity dataset of 3D indoor scenes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12–22 (2023)

[85] Yu, Z., Chen, A., Huang, B., Sattler, T., Geiger, A.: Mip-splatting: Alias-free 3d gaussian splatting (2024)

[86] Yu, Z., Peng, S., Niemeyer, M., Sattler, T., Geiger, A.: Monosdf: Exploring monocular geometric cues for neural implicit surface reconstruction. arXiv (2022)

[87] Yu, Z., Sattler, T., Geiger, A.: Gaussian opacity fields: Efficient and compact surface reconstruction in unbounded scenes. arXiv (2024)

[88] Yuan, Y., Bleier, M., Nuchter, A.: Scenefactory: A workflow-centric and unified framework for incremental scene modeling. IEEE Transactions on Robotics (2025)

[89] Zhai, A.J., Shen, Y., Chen, E.Y., Wang, G.X., Wang, X., Wang, S., Guan, K., Wang, S.: Physical property understanding from language-embedded feature fields (2024)

[90] Zhang, K., Li, B., Hauser, K., Li, Y.: Particle-grid neural dynamics for learning deformable object models from rgb-d videos. In: Proceedings of Robotics: Science and Systems (RSS) (2025)

[91] Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: CVPR (2018)

[92] Zhao, H., Wang, H., Zhao, X., Wang, H., Wu, Z., Long, C., Zou, H.: Automated 3d physical simulation of open-world scene with gaussian splatting. arXiv preprint arXiv:2411.12789 (2024)

[93] Zyrianov, V., Che, H., Liu, Z., Wang, S.: Lidardm: Generative lidar simulation in a generated world (2024), https://arxiv.org/abs/2404.02903

[94] Zyrianov, V., Zhu, X., Wang, S.: Learning to generate realistic lidar point clouds. In: ECCV (2022)

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Our claims match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations of the work in our paper.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: Our paper doesn't focus on the theoretical derivation.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We fully disclose our method and the experiment settings in the paper.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We would open-source the data and code of the paper.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We elaborate the experiment settings in detail in our paper.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We conduct multiple sets of experiments to alleviate the statistical significance of the experiments.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide the training details in the supplementary material.

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: We have read and conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics.

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: We discuss the potential impacts in the paper.

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: The paper has no such risks.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The creators or original owners of assets used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: The paper does not release new assets.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.