LiveScene: Language Embedding Interactive Radiance Fields for Physical Scene Rendering and Control

Delin Qu[1,2*] **Qizhi Chen**[3,2*] **Pingrui Zhang**[1,2] **Xianqiang Gao**[2]
Bin Zhao[2] **Zhigang Wang**[2] **Dong Wang**[2†] **Xuelong Li**[2]
[1]Fudan University [2]Shanghai AI Laboratory [3]Zhejiang University

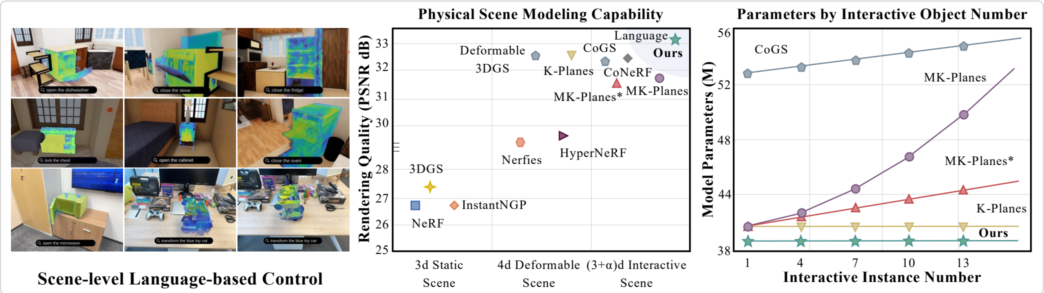


Figure 1: LiveScene enables scene-level reconstruction and control with language grounding. Left: Language-interactive articulated object control in Nerfstudio. Right: LiveScene achieves SOTA rendering quality on OmniSim dataset and exhibits a significant advantage in parameter efficiency.

Abstract

This paper scales object-level reconstruction to complex scenes, advancing interactive scene reconstruction. We introduce two datasets, OmniSim and InterReal, featuring 28 scenes with multiple interactive objects. To tackle the challenge of inaccurate interactive motion recovery in complex scenes, we propose **LiveScene**, a scene-level language-embedded interactive radiance field that efficiently reconstructs and controls multiple objects. By decomposing the interactive scene into local deformable fields, LiveScene enables separate reconstruction of individual object motions, reducing memory consumption. Additionally, our interaction-aware language embedding localizes individual interactive objects, allowing for arbitrary control using natural language. Our approach demonstrates significant superiority in novel view synthesis, interactive scene control, and language grounding performance through extensive experiments. Project page: https://livescenes.github.io.

1 Introduction

Interactive objects are prevalent in our daily lives, and modeling interactable scenes from the real physical world plays an essential role in various research fields, including content generation [33, 35, 52], animation [58, 37, 30], virtual reality [50, 9, 32], robotics [1, 61, 44], and world understanding [16, 14, 53]. This paper tackles the challenging and rarely explored task of reconstructing and controlling multiple interactive objects in complex scenes from a single, casually captured monocular video without previous independent modeling of geometry and kinematics. Prior research on interactable scene modeling, such as CoNeRF [20] and K-Planes [10], typically adopts a joint

[*] Authors contributed equally: dlqu22@m.fudan.edu.cn.
[†] Corresponding author: dongwang.dw93@gmail.com.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

https://doi.org/10.52202/079017-0393

modeling approach, combining spatial coordinates and all interaction variables as input and representing interactive scene by either implicit MLPs or feature planes. Meanwhile, CoGS [63] learns parameter offsets for different scene parts using multiple independent MLPs after establishing a 3D deformable Gaussian scene. However, these methods primarily focus on capturing interactions for a single object within a clear background, such as a single drawer, toy car, or face [66, 20, 63, 68]. As modeling extends from single objects to multiple objects in complex scenes, as shown in Figure. 1, the interaction spaces become increasingly high-dimensional, complicating these methods for accurate modeling and significantly increasing computational time and memory cost, e.g., 4× A100 GPU for 2 weeks to converge training in CoNeRF [20] and 500M Gaussian storage for a regular indoor living room in CoGS [63]. Moreover, natural language is an intuitive and necessary interface for interacting with 3D scenes, but language embedding of interactive scenes faces an even more daunting challenge: interaction variation inconsistency. For instance, methods like LERF [23], and OpenNeRF [67], which distill CLIP features into static 3D fields, suffer from significant failures when confronted with scene topology structure changes induced by interactions, such as the distinct structures variation of a cabinet before and after opening.

To address these challenges, we propose LiveScene, the first scene-level language-embedded radiance fields, which compresses high-dimensional interaction spaces into compact 4D feature planes, reducing model parameters while improving optimization effectiveness. LiveScene models multiple object interactions via novel high-dimensional factorization, decomposing the scene into local deformable fields that model individual objects with multi-scale 4D deformable feature planes. To achieve independent control, we introduce a multi-scale interaction probability sampling strategy for factorized local deformable fields. Our interaction-aware language embedding method generates varying language embeddings to localize and control objects under arbitrary states, enabling natural language control. Finally, we construct the first scene-level physical interaction datasets, OmniSim and InterReal, featuring 28 scenes with 70 interactive objects for evaluation.

Experiment results show that our approach achieves SOTA novel view synthesis quality, outperforming existing best methods by +9.89, +1.30, and +1.99 in PSNR on the CoNeRF Synthetic, OmniSim #chanllenging, and InterReal #chanllenging subsets, respectively. Surpassing LeRF [23], LiveScene significantly improves language grounding accuracy by +65.12 of mIOU on the OmniSim dataset. Notably, our method maintains a lightweight, constant model parameter of 39M, scaling well with increasing scene complexity, as shown in Figure. 1. Contributions can be summarized as:

- We propose LiveScene, the first scene-level language-embedded interactive radiance field, which efficiently reconstructs and controls complex physical scenes, enabling manipulation of multiple articulated objects and language-based interaction.
- We propose a factorization technique that decomposes interactive scenes into local deformable fields and samples relevant 3D points, enabling control of individual objects. Additionally, we introduce an interaction-aware language embedding method that generates varying embeddings, allowing for language-based control and localization.
- We construct the first scene-level physical interaction dataset **OmniSim** and **InterReal**, containing 28 subsets and 70 interactive objects for evaluation. Extensive experiments demonstrate our SOTA performance and robust interaction capabilities.

2 Related Work

Dynamic Scene Representation. Extending NeRF [38] to dynamic scene reconstruction has made significant progress. Related methods can generally be categorized into *time-varying methods*, *deformable-canonical methods*, and *hybrid representation methods*. The time-varying methods [7, 41, 54, 65] typically model the radiance field directly over time, but struggle to separate dynamic and static objects. Deformable-canonical methods [11, 31, 42, 60] decouple dynamic deformable field and static canonical space, modeling 4D by warping points with deformable field to query the canonical features. However, these methods face challenges in scene topology changes [10]. Hybrid representation methods, on the other hand, have achieved high-quality reconstruction and fast rendering by utilizing time-space feature planes [47, 10, 3], 4D hash encoding [56], dynamic voxels [57], or triple fields [49]. Recently, several works [36, 62, 59, 27] have introduced 3D gaussians [21] into dynamic scene reconstruction, achieving high-quality real-time rendering speeds.

https://doi.org/10.52202/079017-0393

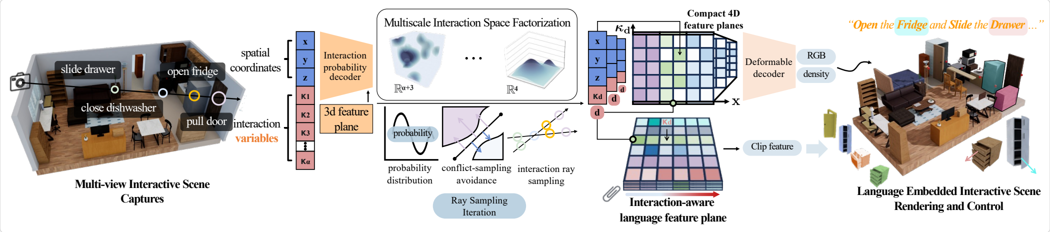

slide drawer
open fridge
close dishwasher
pull door
Multi-view Interactive Scene Captures
spatial coordinates
interaction variables
x
y
z
K1
K2
K3
Kα
Interaction probability decoder
3d feature plane
Multiscale Interaction Space Factorization
$\mathbb{R}^{\alpha+3}$
$\mathbb{R}^4$
probability
probability distribution
conflict-sampling avoidance
interaction ray sampling
Ray Sampling Iteration
κ_d
Kd
d
Compact 4D feature planes
Deformable decoder
RGB
density
Clip feature
Interaction-aware language feature plane
"Open the Fridge and Slide the Drawer ..."
Language Embedded Interactive Scene Rendering and Control

Figure 2: The overview of LiveScene. Given a camera view and control variable $\boldsymbol{\kappa}$ of one specific interactive object, a series of 3D points are sampled in a local deformable field that models the interactive motions of this specific interactive object, and then the interactive object with novel interactive motion state is generated via volume-rendering. Moreover, an interaction-aware language embedding is utilized to localize and control individual interactive objects using natural language.

However, these methods are limited to reconstructing dynamic scenes and lack the ability to control and understand interactive scenes.

3D Vision-language Fields. Vision-language foundational models [45, 40, 25] with strong generalizability and adaptability inspires numerous language embedded scene representation for 3D scene understanding [2, 70], such as open-vocabulary segmentation [34, 13, 55, 24], 3D visual question answering [15, 6, 19, 18], and 3D language grounding [17, 46, 5, 18]. LeRF [23] is the first to achieve open-vocabulary 3D queries by combining CLIP [45] and DINO [4] with NeRF through feature distillation. Open-NeRF [67] introduces an integrate-and-distill paradigm and leverages hierarchical embeddings to address 2D knowledge-distilling issues from SAM. LEGaussians [48] and LangSplat [43] integrate semantic features into 3D gaussians [21] and achieve precision language query and efficient rendering. However, these methods are limited to static scene understanding and fail to generalize when the interactive scene topology changes.

Controllable Scene Representation. Manipulating reconstructed assets or neural fields is of significant importance for avatar and robotic tasks [8, 12, 20, 28]. CoNeRF [20] pioneered this effort by extending HyperNeRF [42] and introduce a fine-grained controlable neural field with 2D attribute mask and value annotations. CoNFies [64] proposes an automatic controllable avatar system and accelerates rendering by distilling. More recently, CoGS [63] leveraged 3D Gaussians [21] to achieve real-time control of dynamic scenes without requiring explicit control signals. However, these methods typically lack natural language interaction capabilities, relying solely on manual control. Furthermore, most works focus on single or few object interactions, disregarding the interaction between different scene parts, limiting their real-world applications.

3 Methodology

We aim to establish a representation that models α interactive articulated objects in a complex scene from a monocular video via a rendering-based self-supervised manner. Control variables $\boldsymbol{\kappa} = [\kappa_1, \kappa_2, ..., \kappa_\alpha]$ indicating object motion states and camera poses of each video frame are given. The overview of LiveScene is shown in Figure. 2. Section. 3.1 introduces the high-dimensional interactive space modeling and challenges. Section. 3.2 presents a multi-scale interactive space factorization and sampling strategy to compress the high-dimensional interactive space into local 4D deformable fields, and model complicated interactive motions of individual objects. Section. 3.3 introduces an interaction-aware language embedding method to localize and control interactive objects using natural language.

3.1 Interactive Space

Assuming a non-rigidly interactive scene with α control variables $\boldsymbol{\kappa} = [\kappa_1, \kappa_2, ..., \kappa_\alpha]$ corresponding to α objects, we delineate its representation by a high-dimensional function:

$$\mathbf{y} = \rho(\mathbf{x}, \boldsymbol{\kappa}, \mathcal{H}; \boldsymbol{\theta}), \tag{1}$$

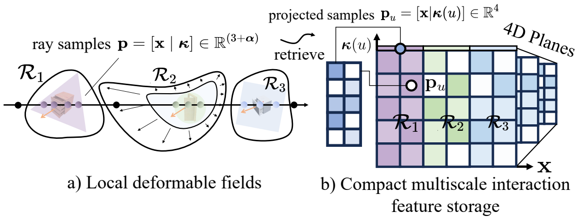
where ρ is the model of the representation, $\mathbf{x} \in \mathbb{R}^3$ are spatial coordinates, $\boldsymbol{\kappa} \in \mathbb{R}^\alpha$ are control variables, $\mathcal{H}$ is a set of optional additional parameters (e.g., the view direction), and $\boldsymbol{\theta}$ stores the scene information. The function outputs scene properties $\mathbf{y}$ for the given position $\mathbf{x}$ and $\boldsymbol{\kappa}$ sampling

https://doi.org/10.52202/079017-0393

from a ray $\mathbf{r}$, where $\mathbf{y}$ can be represented with color, occupancy, signed distance, density, and BRDF parameters. This paper focuses on color, probability, and language embedding. Distinguishing from 3d static scene or 4d dynamic scene modeling, the sampling point $\mathbf{p} = [\mathbf{x}|\boldsymbol{\kappa}] \in \mathbb{R}^{3+\alpha}$ in the interactive scene is high-dimensional and variable in topological structure, complicating the scene feature storage and the optimization of representation model, leading to significant time-consuming or memory-intensive training in [20, 63].

3.2 Multi-scale Interaction Space Factorization

For the $(3+\alpha)$-dimensional interactive space containing α control variables, we aim to explicitly represent the high-dimensional space in a concise and compact storage, thereby reducing memory usage and improving optimization. As illustrated in Figure. 2 and Figure. 10, objects exhibit mutual independence, and interaction features are distributed in the $(3+\alpha)$-dimensional interactive space and aggregate into cluster centers. Thus, there exists a set of hyperplanes that partition the space into disjoint regions, with each region containing a local 4D deformable field, as shown in Figure. 3. Hence, the interaction features at sampling point $\mathbf{p} \in \mathbb{R}^{3+\alpha}$ can be projected into a compact 4-dimensional space $\mathbb{R}^4$ in Figure. 3 by a transformation.


projected samples $\mathbf{p}_u = [\mathbf{x}|\boldsymbol{\kappa}(u)] \in \mathbb{R}^4$
ray samples $\mathbf{p} = [\mathbf{x} \mid \boldsymbol{\kappa}] \in \mathbb{R}^{(3+\alpha)}$
retrieve
$\boldsymbol{\kappa}(u)$
4D Planes
$\mathcal{R}_1$
$\mathcal{R}_2$
$\mathcal{R}_3$
$\mathbf{p}_u$
$\mathcal{R}_1$
$\mathcal{R}_2$
$\mathcal{R}_3$
$\mathbf{x}$

a) Local deformable fields

b) Compact multiscale interaction feature storage

Figure 3: Illustration of hyperplanar factorization for compact storage. We maintain multiple local deformable fields for each interactive object region $\mathcal{R}_i$, and project high-dimensional interaction features into a compact 4D space, which can be further compressed into multiscale feature planes.

Multi-scale Interactive Ray Sampling. We consider using ray sampling to perform the projection transformation. As shown in Figure. 3, assuming a ray $\mathbf{r}(t) = [\mathbf{o}_{\mathbf{x}}|\mathbf{o}_{\boldsymbol{\kappa}}] + t\,[\mathbf{d}_{\mathbf{x}}|\mathbf{d}_{\boldsymbol{\kappa}}]$ with origin $\mathbf{o}_{\mathbf{x}} \in \mathbb{R}^3, \mathbf{o}_{\boldsymbol{\kappa}} \in \mathbb{R}^\alpha$, and direction $\mathbf{d}_{\mathbf{x}} \in \mathbb{R}^3, \mathbf{d}_{\boldsymbol{\kappa}} \in \mathbb{R}^\alpha$, the ray intersects with the interaction region at a point $\mathbf{p} = [\mathbf{x}|\boldsymbol{\kappa}] \in \mathbb{R}^{3+\alpha}$, where $\mathbf{x} \in \mathbb{R}^3$ is the 3D position and $\boldsymbol{\kappa} \in \mathbb{R}^\alpha$ is the interaction variables. For a given intersection point $\mathbf{p}$, the deformable features can be retrieved from the corresponding local 4D deformable field by maximizing sampling probability $\mathbf{P}$:

$$\mathbf{p}_u = [\mathbf{x}|\boldsymbol{\kappa}(u)]\,, \quad u = \arg\max_i\{\mathbf{P}_i\}, \quad \mathbf{P} = \Theta(\boldsymbol{\kappa}, \boldsymbol{\theta}), \tag{2}$$

where $\mathbf{p}_u$ and $\boldsymbol{\theta}$ are the 4D sampling point and probability features at position $\mathbf{x}$ from 3D feature planes. The maximum argument operation of probability decoder $\Theta(\boldsymbol{\kappa}, \boldsymbol{\theta})$ maps the interaction variables $\boldsymbol{\kappa}$ to the most probable cluster region in the 4D space, and can be optimized by minimizing the focal loss $\mathcal{L}_{\text{focus}}$ of mask across all the training camera views:

$$\mathcal{L}_{\text{focus}} = \beta \cdot \left(1 - e^{\sum_{i=1}^{\alpha} \mathbf{M}_i \log(\hat{\mathbf{P}}_i)}\right)^{\gamma} \cdot \left(-\sum_{i=1}^{\alpha} \mathbf{M}_i \log(\hat{\mathbf{P}}_i)\right), \tag{3}$$

where $\mathbf{M}$ is the ground truth mask label, $\hat{\mathbf{P}}$ is the probability map rendering from the interactive probability field, β is the balancing factor, and γ is the focusing parameter.

Next, the deformable features are used to render local deformable scene color at the sampling point $\mathbf{p}$. In this way, the high-dimensional interaction features are factorized into a 4D space by a lightweight transformation modeling with interaction probability decoder and 3D feature planes in Figure. 2, supervised by deformable masks $\mathbf{M}$. Moreover, leveraging K-Planes [10], the multiple 4D local deformable space can be further compressed in only $\mathbf{C}_4^2 = 6$ feature planes. We iteratively sample from coarse to fine within the multi-scale feature plane, retrieving the maximum probability interaction variables $\boldsymbol{\kappa}_u$ and indices u at each scale.

Feature Repulsion and Probability Rejection. A latent challenge is that optimizing the interaction probability decoder with varying masks can lead to blurred boundaries in the local deformable field, further causing ray sampling and feature storage conflicts. As illustrated in Figure. 4(a), consider two adjacent local deformable regions $\mathcal{R}_i$ and $\mathcal{R}_j$, and a point $\mathbf{p}$ in high-dimensional space, suppose $\mathbf{p}$ moves from the cluster center of $\mathcal{R}_i$ towards the cluster center of $\mathcal{R}_j$, then the probability of $\mathbf{p}$ belonging to $\mathcal{R}_i$ gradually decreases, while the probability of $\mathbf{p}$ belonging to $\mathcal{R}_j$ increases. To avoid sampling conflicts and feature oscillations at the boundaries, we introduce a repulsion loss for ray

https://doi.org/10.52202/079017-0393

pairs $(\mathbf{r}_i, \mathbf{r}_j)$, and amplify the feature differences between distinct deformable regions, promoting the separation of deformable field:

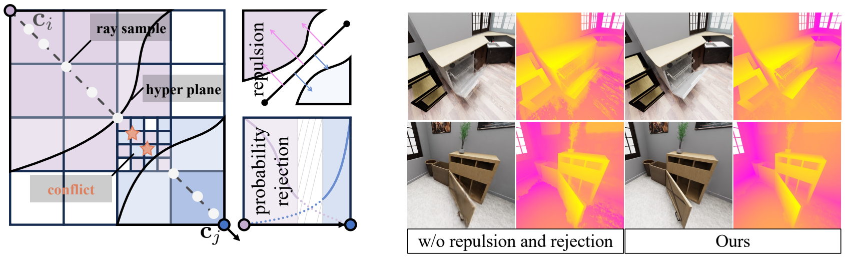
$$\mathcal{L}_{\text{repuls}} = \mathbf{ELU}(K - \|(\mathbf{M}_i \odot \mathbf{M}_j)(\mathcal{F}_i - \mathcal{F}_j)\|), \tag{4}$$

where K is a constant hyperparameter, $\mathbf{M}_i$ and $\mathbf{M}_j$ are the ground truth mask of rays. $\mathcal{F}_i$ and $\mathcal{F}_j$ are the last-layer features of interaction probability decoder in Figure. 2. During training, we randomly select ray pairs and apply $\mathcal{L}_{\text{repuls}}$ to enforce the separation of interactive probability features across local deformable spaces. The initiative is inspired by [24], which demonstrated the efficacy of repulsive forces in disambiguating 3D segmentation results.

Additionally, the probability rejection is proposed to truncate the low-probability samples if the maximum deformable probability $\mathbf{P}$ at sample $\mathbf{p}$ is smaller than threshold s, and selects the background feature directly. The operation is defined as:

$$u = \begin{cases} \arg\max_i\{\mathbf{P}_i\}, & \text{if} \quad \mathbf{P}_i \geq s \\ -1, & \text{otherwise} \end{cases}. \tag{5}$$

As shown in Figure. 4(b), the proposed operations help the model achieve higher rendering quality, demonstrating their effectiveness in alleviating boundary sampling conflicts.



a) Local Feature Conflicts b) Rendering Quality Comparison

Figure 4: Illustration of a) boundary sampling conflicts, b) rendering quality comparison.

3.3 Interaction-Aware Language Embedding

Language embedding in interactive scenes is complex and storage-intensive, as 3D distillation faces the dual challenges of high-dimensional optimization and interaction scene variation inconsistency, such as the distinct topological structures of a transformer toy before and after transformation, leading to the failure of SAM [25] segmentation or LERF [23] grounding. As shown in Figure. 2, leveraging the proposed multi-scale interaction space factorization of 3.2, we efficiently store language features in lightweight planes by indexing them according to maximum probability sampling instead of 3D fields in LERF. For any sampling point $\mathbf{p}$, we project it onto $\mathbf{p}_u = [\mathbf{x}|\boldsymbol{\kappa}(u)]$, retrieving a local language feature group by index d, and perform bilinear interpolation using $\boldsymbol{\kappa}_u$ to obtain a language embedding that adapts to interactive variable changes from surrounding clip features. By interpolating language embeddings, our method not only perceives topological structure changes but also achieves a storage complexity of $\mathbf{O}(C \times \alpha \times \mathbf{dim})$, much smaller than language distillation methods like LERF that operate in 3D scenes, where $\mathbf{dim}$ is the dimension of CLIP feature.

4 Dataset

To our knowledge, existing view synthetic datasets for interactive scene rendering are primarily limited to a few interactive objects [66, 20, 63, 68] due to necessitating a substantial amount of manual annotation of object masks and states, making it impractical to scale up to real scenarios involving multi-object interactions. To bridge this gap, we construct two scene-level, high-quality annotated datasets to advance research progress in reconstructing and understanding interactive scenes: **OmniSim** and **InterReal**, as shown in Figure. 5. Besides, we use the CoNeRF Synthetic and Controllable [20] dataset for evaluation as well. **1) OmniSim Dataset** is rendered through OmniGibson [29] simulator, leveraging 7 indoor scene models: `#rs`, `#ihlen`, `#beechwod`, `#merom`, `#pomaria`, `#wainscott` and `#benevolence`. By varying the rotation vectors of the articulated objects' joints and the camera's trajectory within the scene, we generated 20 high-definition subsets, each

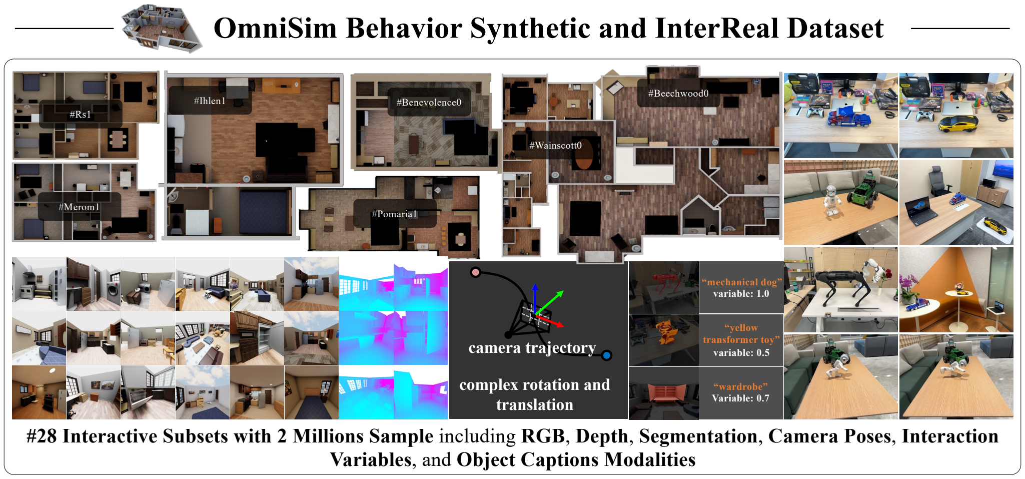

OmniSim Behavior Synthetic and InterReal Dataset
#Rs1
#Ihlen1
#Benevolence0
#Beechwood0
#Wainscott0
#Merom1
#Pomaria1
camera trajectory
complex rotation and translation
"mechanical dog" variable: 1.0
"yellow transformer toy" variable: 0.5
"wardrobe" Variable: 0.7
#28 Interactive Subsets with 2 Millions Sample including RGB, Depth, Segmentation, Camera Poses, Interaction Variables, and Object Captions Modalities

Figure 5: **Overview of the OmniSim and InterReal datasets**.

consisting of RGBD images, camera trajectory, interactive object masks, and corresponding object state quantities at each time step. **2) InterReal Dataset** is captured from 8 real Interactable scenes and finely annotated with interaction variables and masks, camera poses encompassing multiple objects, and articulated motion variables. More details can be found in the supplementary.

5 Experiment

Baselines. We compare LiveScene with the existing 3D static rendering methods [38, 39, 22, 21], 4D deformable methods [10, 42, 41], and controllable scene reconstruction methods [20, 63]. Note that we reimplemented CoGS [63] based on Deformable Gaussian [62] since the official code is unavailable. Additionally, we extended K-Planes [10] from $\mathbf{C}_4^2$ planes to $\mathbf{C}_{3+\alpha}^2$ planes, denoted as MK-Planes, where α represents the number of interactable objects in dynamic scenes. By leveraging the fact that each instance occupies a distinct region, we further compressed the model, denoted as MK-Planes*, requiring only $3 + 3\alpha$ planes.

Implementation. LiveScene is implemented in Nerfstudio [51] from scratch. We represent the field as a multi-scale feature plane with resolutions of $512 \times 256 \times 128$, and feature dimension of 32. The proposal network adopts a coarse-to-fine sampling process, where each sampling step concatenates the position feature and the state quantity as the query for the 4D deformation probability field, which is a 1-layer MLP with 64 neurons and ReLU activation. We use the Adam optimizer with initial learning rates of 0.01 and a cosine decay scheduler with 512 warmup steps for all networks. The model is trained with an NVIDIA A100 GPU for 80k steps on the OmniSim dataset and 100k steps on the InterReal dataset, using a batch size of 4096 rays with 64 samples. More implementation can be found in the supplemental materials.

5.1 View Synthesis Quality Comparison

Evaluation on CoNeRF Synthetic and Controllable Datasets. We report the quantitative results on CoNeRF Synthetic and Controllable scenes in Table. 1. LiveScene outperforms all the existing PSNR, SSIM, and LPIPS metrics methods on CoNeRF Synthetic scenes with a large margin. In particular, LiveScene achieves 43.349, 0.986, and 0.011 in PSNR, SSIM, and LPIPS, respectively, outperforming the second-best method by 9.894, 0.009, and 0.053. On CoNeRF Controllable, LiveScene achieves the best PSNR of 32.782 and comparable SSIM and LPIPS to the SOTA methods. According to the novel view synthesis results in Figure. 6, LiveScene achieves more detailed and higher rendering quality, demonstrating the effectiveness of LiveScene in modeling object-level interactive scenarios.

Evaluation on OmniSim Datasets. OmniSim dataset is categorized into 3 interaction level subsets: #easy, #medium, and #challenging, based on the number of interactive objects in each scene. As shown in Table. 2, LiveScene achieves the best PSNR, SSIM, and LPIPS on all interaction level subsets of OmniSim, with average PSNR, SSIM, and LPIPS of 33.158, 0.962, and 0.074, respectively. Notably, substantial performance degradation is observed across all methods as the quantity and complexity of interactive objects increase, e.g., CoGS [63] experiences a 3.641 dB PSNR drop from

https://doi.org/10.52202/079017-0393

Table 1: **Quantitative results on CoNeRF synthetic and controllable datasets.** LiveScene achieves the best results in all metrics on synthetic scenes and the best PSNR on the controllable datasets.

Method	CoNeRF Syntetic			CoNeRF Controllable		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
NeRF [38]	25.299	0.843	0.197	28.795	0.951	0.210
InstantNGP [39]	27.057	0.903	0.230	26.391	0.884	0.278
3DGS [21]	32.576	0.977	0.077	25.945	0.834	0.414
NeRF + Latent [38]	28.447	0.939	0.115	32.653	0.981	0.182
Nerfies[41]	—	—	—	32.274	0.981	0.180
HyperNeRF[42]	25.963	0.854	0.158	32.520	0.981	0.169
K-Planes [10]	33.301	0.933	0.150	31.811	0.912	0.262
CoNeRF-$\mathcal{M}$[20]	27.868	0.898	0.155	32.061	0.979	0.167
CoNeRF[20]	32.394	0.972	0.139	32.342	0.981	0.168
CoGS [63]	33.455	0.960	0.064	32.601	**0.983**	**0.164**
LiveScene (Ours)	**43.349**	**0.986**	**0.011**	**32.782**	0.932	0.186

Table 2: **Quantitative results on OmniSim Dataset**. LiveScene outperforms prior works on most metrics and achieves the best PSNR on the #challenging subset with a significant margin.

Method	#Easy Sets			#Medium Sets			#Challenging Sets			#Avg (all 20 Sets)		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
NeRF [38]	25.817	0.906	0.167	25.645	0.928	0.138	26.364	0.927	0.128	25.776	0.916	0.153
InstantNGP [39]	25.704	0.902	0.183	25.627	0.930	0.140	26.367	0.920	0.143	25.706	0.914	0.164
HyperNeRF [42]	30.708	0.908	0.316	31.621	0.936	0.265	27.533	0.897	0.318	30.748	0.917	0.299
K-Planes [10]	32.841	0.952	0.093	32.548	0.954	0.100	29.833	0.937	0.118	32.573	0.952	0.097
CoNeRF [20]	32.104	0.932	0.254	33.256	0.951	0.207	30.349	0.923	0.238	32.477	0.939	0.234
MK-Planes*	31.630	0.948	0.098	31.880	0.951	0.104	26.565	0.887	0.218	31.477	0.946	0.106
MK-Planes	31.677	0.948	0.098	32.165	0.952	0.099	29.254	0.933	0.119	31.751	0.949	0.099
CoGS [63]	32.315	0.961	0.108	32.447	**0.965**	0.086	28.701	**0.970**	**0.073**	32.187	**0.963**	0.097
LiveScene (Ours)	**33.221**	**0.962**	**0.072**	**33.262**	**0.965**	**0.072**	**31.645**	0.948	0.093	**33.158**	0.962	**0.074**

Table 3: **Quantitative results on InterReal Dataset**. Our method outperforms others in most settings, with a significant advantage of PSNR, SSIM, and LPIPS on the #challenging subset.

Method	#Medium Sets			#Challenging Sets			#Avg (all 8 Sets)		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
NeRF [38]	20.816	0.682	0.190	21.169	0.728	0.337	20.905	0.694	0.227
InstantNGP [39]	21.700	0.776	0.215	21.643	0.745	0.338	21.686	0.769	0.245
HyperNeRF [42]	25.283	0.671	0.467	25.261	0.713	0.517	25.277	0.682	0.480
K-Planes [10]	27.999	0.813	0.177	26.427	0.756	0.331	27.606	0.799	0.215
CoNeRF [20]	27.501	0.745	0.367	26.447	0.734	0.472	27.237	0.742	0.393
CoGS [63]	30.774	**0.913**	0.100	✗	✗	✗	**30.774**	**0.913**	0.100
LiveScene (Ours)	**30.815**	0.911	**0.066**	**28.436**	**0.846**	**0.185**	30.220	0.895	**0.096**

#easy to #challenging. While LiveScene maintains a relatively stable high performance across all subsets, demonstrating its robustness in modeling complex interactive scenarios.

Evaluation on InterReal Datasets. We divided InterReal dataset into #medium and #challenging subsets. In Table. 3, CoGS [63] underperforms compared to LiveScene on the #medium subset and fails to converge when faced with long camera trajectories and a large number of interactive objects in the scene (#challenging), highlighting the limitation of existing methods in modeling real-world interactive scenarios. In contrast, LiveScene achieves the highest PSNR of 28.436 and the lowest LPIPS of 0.185 on the #challenging subset, indicating its superiority in modeling real-world large-scale interactive scenarios.

View Synthesis Visulization. Figure. 7 presents the novel view synthesis results of LiveScene and the SOTA methods on OmniSim dataset. The results reveal that LiveScene generates more detailed results than SOTA methods, particularly in complex interactive scenarios. For instance, on the #pomaria scene featuring an openable dishwasher, CoNeRF [20] fails to capture details, while CoGS [63] and MK-Planes* exhibit residual artifacts. In contrast, our method accurately reconstructs the internal details. Another challenge arises in the #rs scene, where other methods struggle to reconstruct distant and static objects. In comparison, our method not only overcomes the challenging problem of dramatic topology changes in interactive scenes but also maintains the ability to reconstruct high-quality static scenes.

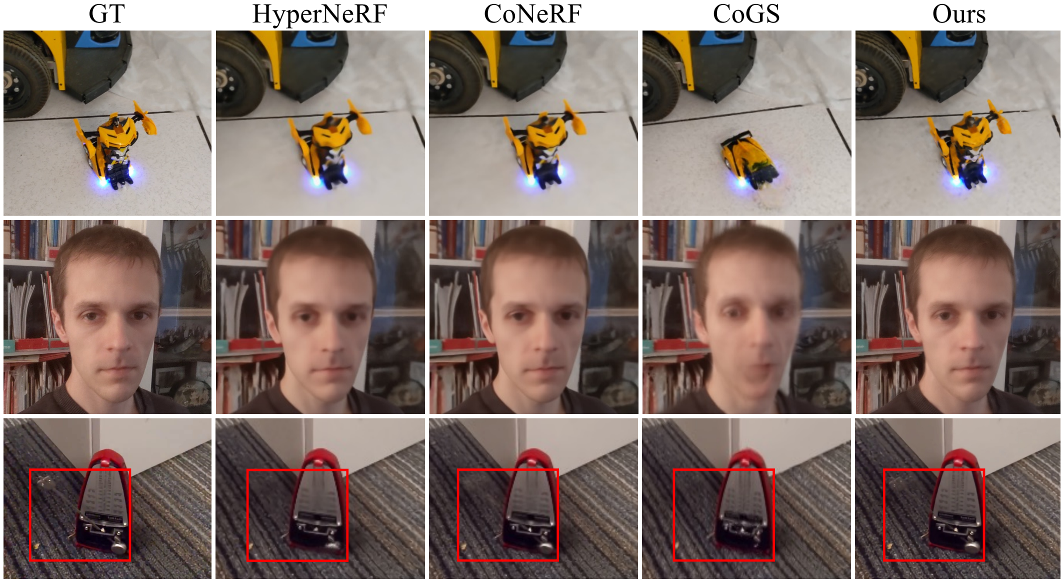


Figure 6: **View Synthesis Visualization on CoNeRF Controllable Dataset**. The proposed method achieves higher-quality rendering results compared with the existing methods.

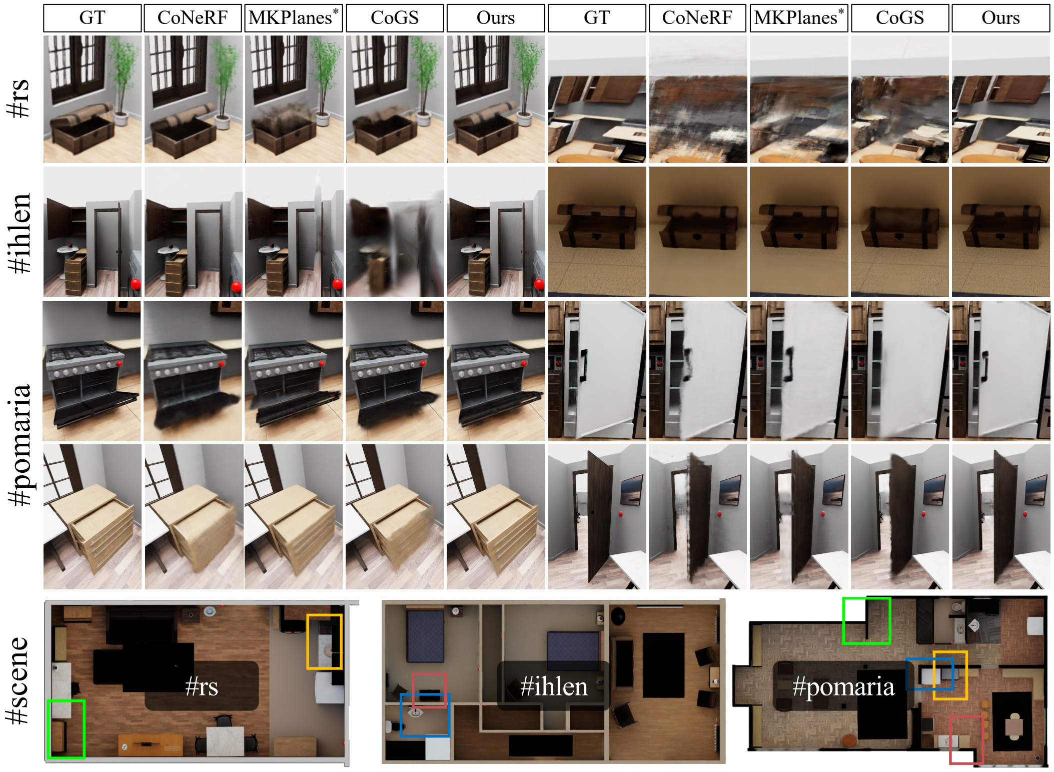


Figure 7: **View Synthesis Visualization on OmniSim Dataset**. We compare our method with SOTA methods on RGB rendering across three scenes: #rs, #ihlen, and #pomaria. Boxes of different colors represent distinct interactive objects within the scene.

5.2 Language Grounding Comparison

We assess the language grounding performance on the OmniSim dataset using mIOU metric. Figure. 8 suggests that our method obtains the highest mIOU score, with an average of 86.86. In contrast, traditional methods like LERF [23] encounter difficulties in locating objects precisely, with an average mIOU of 21.74. Meanwhile, 2D methods like SAM [25] fail to accurately segment the whole target

https://doi.org/10.52202/079017-0393

	mIOU ↑			
	setting	SAM [25]	LERF [23]	Ours
OmniSim	#easy	61.58	23.60	**86.94**
	#medium	55.13	19.40	**86.32**
	#challenging	63.86	19.87	**90.41**
	#avg	59.11	21.74	**86.86**
InterReal	#medium	93.27	27.63	84.37
	#challenging	**91.50**	34.39	**91.90**
	#avg	**92.82**	29.32	86.26

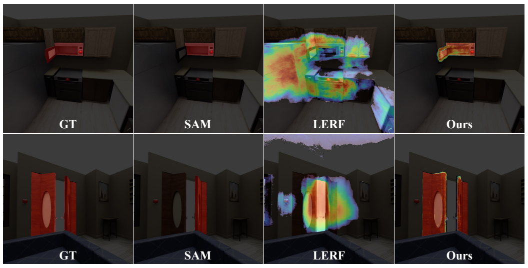


Figure 8: **Language Grounding Performance on OmniSim Dataset** left): Our method gains the highest mIOU score. right): LiveScene's grounding exhibits clearer boundaries than other methods.

under specific viewing angles, as objects appear discontinuous in the image. Conversely, our method perceives the completeness of the object and has clear knowledge of its boundaries, demonstrating its advantage in language grounding tasks.

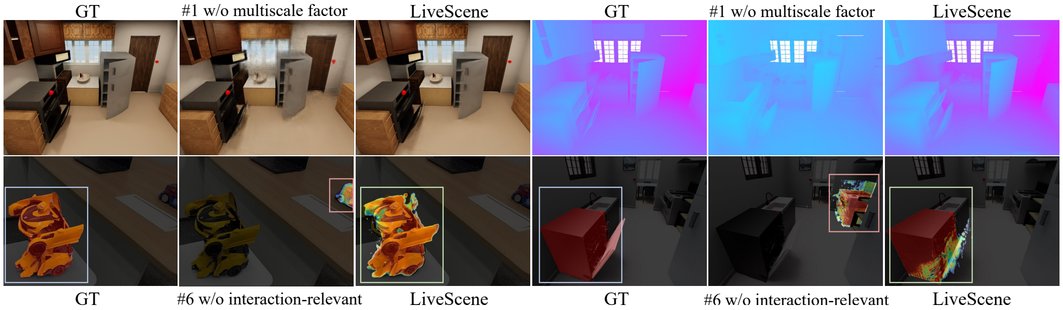


Figure 9: **Rendering and Grounding Performance for #1 and #6.** above): Multi-scale factorization greatly boosts the performance of RGB rendering and geometry reconstruction. below): Without view consistency, the model struggles when objects have similar appearances.

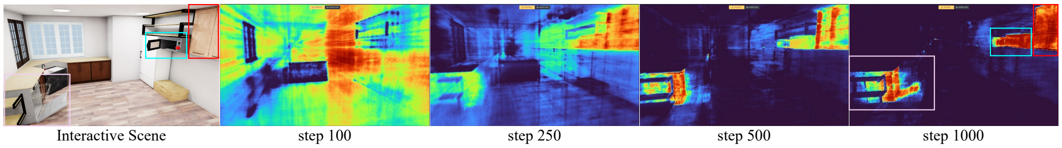


Figure 10: Learning process of the probability fields from 0 to 1000 training steps. The model progressively converges to the vicinity of the interactive objects, establishing interactive regions.

5.3 Ablation Study

In this section, we present ablative studies to investigate the effectiveness of each component in LiveScene. We selected 1 scene from the #medium subset, 2 scenes from the #easy subset of OmniSim dataset, and 1 scene each from the #medium and #challenging settings for InterReal. Notably, ground-truth state quantities are only available in OmniSim, not in InterReal. Therefore, we use GT quantities on OmniSim and introduce a learnable variable on InterReal to infer state changes. Figure. 9 reports the rendering quality and grounding performance for #1 and #6.

Effectiveness of components. As illustrated in Table. 4, the multi-scale factorization significantly improves the rendering performance on both datasets, with PSNR on OmniSim increasing from 31.74 to 35.094, shown in #1. Introducing learnable variables for each frame (#2) yields corresponding improvements on InterReal dataset since this latent code can perceive the change in object states. The feature repulsion loss and probability rejection (#3 and #4) together make rendering quality better in InterReal as well as in OmniSim dataset. As for grounding, #5 shows that rendering embeddings along a ray [26, 69] struggles to locate objects precisely. Ensuring view consistency further boosts grounding performance on OmniSim, as demonstrated in #6.

Probability field training. We provide additional experiments in Figure. 10 of the disjoint regions to illustrate the learning process of the probability field from 0 to 1000 training steps. The results demonstrate a clear trend that, as training advances, the proposed method can progressively converge to the vicinity of the interactive objects, thereby establishing interactive regions. With the establishment of the probability field, the model can focus on different interactive objects and guide the sampling process by maximizing probability, thereby achieving disentanglement of interactive scenes.

Table 4: **Ablation Study** on the subset of InterReal and OmniSim Datasets.

#exp	#settings (rendering)						InterReal			OmniSim			
	I	II	III	IV	V	VI	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	Depth L1↓
#0							25.329	0.731	0.329	31.740	0.938	0.118	0.238
#1	✓						28.289	0.819	0.226	35.094	0.969	0.059	0.086
#2	✓	✓					29.577	0.865	0.162	—	—	—	—
#3	✓	✓*		✓			29.959	0.883	0.131	34.989	0.969	0.059	0.085
#4	✓	✓*	✓				30.123	0.884	0.132	34.977	0.967	0.061	0.086
LiveScene	✓	✓*	✓	✓			30.591	0.896	0.115	35.254	0.971	0.057	0.042
	#settings (grounding)						mIOU ↑			mIOU ↑			
#5	✓	✓*	✓	✓		✓	30.40			32.87			
#6	✓	✓*	✓	✓	✓		93.10			71.64			
LiveScene	✓	✓*	✓	✓	✓	✓	93.02			78.52			

I: multi-scale factorization, II: learnable variable, III: feature repulsion $\mathcal{L}_{\text{repuls}}$, IV: probability rejection, V: maximum probability embeds retrieval, VI: interaction-aware language embedding. ✓* denotes enable II for InterReal but disable for OmniSim.

6 Conclusion and Limitation

We present LiveScene, the first language-embedded interactive neural radiance field for complex scenes with multiple interactive objects. A parameter-efficient factorization technique is proposed to decompose interactive spaces into local deformable fields to model individual interactive objects. Moreover, we introduce a novel interaction-aware language embedding mechanism that effectively localizes and controls interactive objects using natural language. Finally, We construct two challenging datasets that contain multiple interactive objects in complex scenes and evaluate the effectiveness and robustness of LiveScene.

Limitations: The control ability of LiveScene is limited by label density. Additionally, our natural language control is currently restricted to closed vocabulary, and it is inherently tied to the capabilities of the underlying foundation model, such as OpenCLIP. In future work, we plan to extend our method to enable open-vocabulary grounding and control, increasing the model's flexibility and range of applications.

Acknowledgements. This work is partially supported by the Shanghai AI Laboratory, National Key R&D Program of China (2022ZD0160101), the National Natural Science Foundation of China (62376222), and Young Elite Scientists Sponsorship Program by CAST (2023QNRC001).

References

[1] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton Van Den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3674–3683, 2018.

[2] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

[3] Ang Cao and Justin Johnson. Hexplane: A fast representation for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 130–141, 2023.

[4] 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*, pages 9650–9660, 2021.

[5] Boyuan Chen, Fei Xia, Brian Ichter, Kanishka Rao, Keerthana Gopalakrishnan, Michael S Ryoo, Austin Stone, and Daniel Kappler. Open-vocabulary queryable scene representations for real world planning. In *IEEE International Conference on Robotics and Automation*, pages 11509–11522. IEEE, 2023.

[6] Sijin Chen, Xin Chen, Chi Zhang, Mingsheng Li, Gang Yu, Hao Fei, Hongyuan Zhu, Jiayuan Fan, and Tao Chen. Ll3da: Visual interactive instruction tuning for omni-3d understanding reasoning and planning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26428–26438, 2024.

[7] Jiemin Fang, Taoran Yi, Xinggang Wang, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Matthias Nießner, and Qi Tian. Fast dynamic radiance fields with time-aware neural voxels. In *SIGGRAPH Asia*, pages 1–9, 2022.

[8] Roya Firoozi, Johnathan Tucker, Stephen Tian, Anirudha Majumdar, Jiankai Sun, Weiyu Liu, Yuke Zhu, Shuran Song, Ashish Kapoor, Karol Hausman, et al. Foundation models in robotics: Applications, challenges, and the future. *arXiv preprint arXiv:2312.07843*, 2023.

[9] Carlos Flavián, Sergio Ibáñez-Sánchez, and Carlos Orús. The impact of virtual, augmented and mixed reality technologies on the customer experience. *Journal of Business Research*, 2019.

[10] Sara Fridovich-Keil, Giacomo Meanti, Frederik Rahbæk Warburg, Benjamin Recht, and Angjoo Kanazawa. K-planes: Explicit radiance fields in space, time, and appearance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12479–12488, 2023.

[11] Chen Gao, Ayush Saraf, Johannes Kopf, and Jia-Bin Huang. Dynamic view synthesis from dynamic monocular video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5712–5721, 2021.

[12] Stephan J Garbin, Marek Kowalski, Virginia Estellers, Stanislaw Szymanowicz, Shideh Rezaeifar, Jingjing Shen, Matthew A Johnson, and Julien Valentin. Voltemorph: Real-time, controllable and generalizable animation of volumetric representations. In *Computer Graphics Forum*, volume 43, page e15117. Wiley Online Library, 2024.

[13] Rahul Goel, Dhawal Sirikonda, Saurabh Saini, and PJ Narayanan. Interactive segmentation of radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4201–4211, 2023.

[14] Zoey Guo, Yiwen Tang, Ray Zhang, Dong Wang, Zhigang Wang, Bin Zhao, and Xuelong Li. Viewrefer: Grasp the multi-view knowledge for 3d visual grounding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15372–15383, 2023.

[15] Yining Hong, Chunru Lin, Yilun Du, Zhenfang Chen, Joshua B Tenenbaum, and Chuang Gan. 3d concept learning and reasoning from multi-view images. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[16] Yining Hong, Zishuo Zheng, Peihao Chen, Yian Wang, Junyan Li, and Chuang Gan. Multiply: A multisensory object-centric embodied large language model in 3d world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26406–26416, 2024.

https://doi.org/10.52202/079017-0393

[17] Chenguang Huang, Oier Mees, Andy Zeng, and Wolfram Burgard. Visual language maps for robot navigation. In *IEEE International Conference on Robotics and Automation*, pages 10608–10615. IEEE, 2023.

[18] Krishna Murthy Jatavallabhula, Ali Kuwajerwala, Qiao Gu, Mohd. Omama, Tao Chen, Shuang Li, Ganesh Iyer, Soroush Saryazdi, Nikhil Varma Keetha, A. Tewari, J. Tenenbaum, Celso M. de Melo, M. Krishna, L. Paull, F. Shkurti, and A. Torralba. Conceptfusion: Open-set multimodal 3d mapping. *arXiv.org*, 2023.

[19] Baoxiong Jia, Yixin Chen, Huangyue Yu, Yan Wang, Xuesong Niu, Tengyu Liu, Qing Li, and Siyuan Huang. Sceneverse: Scaling 3d vision-language learning for grounded scene understanding. *arXiv preprint arXiv:2401.09340*, 2024.

[20] Kacper Kania, Kwang Moo Yi, Marek Kowalski, Tomasz Trzciński, and Andrea Tagliasacchi. Conerf: Controllable neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18623–18632, 2022.

[21] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), July 2023.

[22] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023.

[23] Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. Lerf: Language embedded radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19729–19739, 2023.

[24] Chung Min Kim, Mingxuan Wu, Justin Kerr, Ken Goldberg, Matthew Tancik, and Angjoo Kanazawa. Garfield: Group anything with radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21530–21539, 2024.

[25] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023.

[26] Sosuke Kobayashi, Eiichi Matsumoto, and Vincent Sitzmann. Decomposing nerf for editing via feature field distillation. *Advances in Neural Information Processing Systems*, 35:23311–23330, 2022.

[27] Agelos Kratimenos, Jiahui Lei, and Kostas Daniilidis. Dynmf: Neural motion factorization for real-time dynamic view synthesis with 3d gaussian splatting. *arXiV*, 2023.

[28] Verica Lazova, Vladimir Guzov, Kyle Olszewski, Sergey Tulyakov, and Gerard Pons-Moll. Control-nerf: Editable feature volumes for scene rendering and manipulation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4340–4350, 2023.

[29] Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martín-Martín, Chen Wang, Gabrael Levine, Wensi Ai, Benjamin Martinez, et al. Behavior-1k: A human-centered, embodied ai benchmark with 1,000 everyday activities and realistic simulation. *arXiv preprint arXiv:2403.09227*, 2024.

[30] Ruilong Li, Julian Tanke, Minh Vo, Michael Zollhöfer, Jürgen Gall, Angjoo Kanazawa, and Christoph Lassner. Tava: Template-free animatable volumetric actors. In *European Conference on Computer Vision*, pages 419–436. Springer, 2022.

[31] Tianye Li, Mira Slavcheva, Michael Zollhoefer, Simon Green, Christoph Lassner, Changil Kim, Tanner Schmidt, Steven Lovegrove, Michael Goesele, Richard Newcombe, et al. Neural 3d video synthesis from multi-view video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5521–5531, 2022.

[32] Bangyan Liao, Delin Qu, Yifei Xue, Huiqing Zhang, and Yizhen Lao. Revisiting rolling shutter bundle adjustment: Toward accurate and fast solution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4863–4871, June 2023.

[33] Jingbo Zhang3 Zhihao Liang4 Jing Liao, Yan-Pei Cao, and Ying Shan. Advances in 3d generation: A survey. *arXiv preprint arXiv:2401.17807*, 2024.

[34] Kunhao Liu, Fangneng Zhan, Jiahui Zhang, Muyu Xu, Yingchen Yu, Abdulmotaleb El Saddik, Christian Theobalt, Eric Xing, and Shijian Lu. 3d open-vocabulary segmentation with foundation models. *arXiv preprint arXiv:2305.14093*, 2(3):6, 2023.

https://doi.org/10.52202/079017-0393

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

[36] Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. *arXiv preprint arXiv:2308.09713*, 2023.

[37] Haimin Luo, Min Ouyang, Zijun Zhao, Suyi Jiang, Longwen Zhang, Qixuan Zhang, Wei Yang, Lan Xu, and Jingyi Yu. Gaussianhair: Hair modeling and rendering with light-aware gaussians. *arXiv preprint arXiv:2402.10483*, 2024.

[38] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

[39] T. Müller, Alex Evans, Christoph Schied, and A. Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics*, 2022.

[40] M. Oquab, Timoth'ee Darcet, T. Moutakanni, Huy Q. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russ Howes, Po-Yao (Bernie) Huang, Shang-Wen Li, Ishan Misra, Michael G. Rabbat, Vasu Sharma, Gabriel Synnaeve, Huijiao Xu, H. Jégou, J. Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision. *arXiv.org*, 2023.

[41] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5865–5874, 2021.

[42] Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T. Barron, Sofien Bouaziz, Dan B Goldman, Ricardo Martin-Brualla, and Steven M. Seitz. Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. *ACM Trans. Graph.*, 40(6), dec 2021.

[43] Minghan Qin, Wanhua Li, Jiawei Zhou, Haoqian Wang, and Hanspeter Pfister. Langsplat: 3d language gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20051–20060, 2024.

[44] Delin Qu, Chi Yan, Dong Wang, Jie Yin, Qizhi Chen, Dan Xu, Yiting Zhang, Bin Zhao, and Xuelong Li. Implicit event-rgbd neural slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19584–19594, June 2024.

[45] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[46] Nur Muhammad (Mahi) Shafiullah, Chris Paxton, Lerrel Pinto, Soumith Chintala, and Arthur D. Szlam. Clip-fields: Weakly supervised semantic fields for robotic memory. *ArXiv*, 2022.

[47] Ruizhi Shao, Zerong Zheng, Hanzhang Tu, Boning Liu, Hongwen Zhang, and Yebin Liu. Tensor4d: Efficient neural 4d decomposition for high-fidelity dynamic reconstruction and rendering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2023.

[48] Jin-Chuan Shi, Miao Wang, Hao-Bin Duan, and Shao-Hua Guan. Language embedded 3d gaussians for open-vocabulary scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5333–5343, 2024.

[49] Liangchen Song, Anpei Chen, Zhong Li, Zhang Chen, Lele Chen, Junsong Yuan, Yi Xu, and Andreas Geiger. Nerfplayer: A streamable dynamic scene representation with decomposed neural radiance fields. *IEEE Transactions on Visualization and Computer Graphics*, 29(5):2732–2742, 2023.

[50] Jonathan Steuer. Defining virtual reality: dimensions determining telepresence. 1992.

[51] Matthew Tancik, Ethan Weber, Evonne Ng, Ruilong Li, Brent Yi, J. Kerr, Terrance Wang, Alexander Kristoffersen, Jake Austin, Kamyar Salahi, Abhik Ahuja, David McAllister, and Angjoo Kanazawa. Nerfstudio: A modular framework for neural radiance field development. *ArXiv*, 2023.

[52] Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. In *The Twelfth International Conference on Learning Representations*, 2024.

https://doi.org/10.52202/079017-0393

[53] Yiwen Tang, Ray Zhang, Jiaming Liu, Zoey Guo, Bin Zhao, Zhigang Wang, Peng Gao, Hongsheng Li, Dong Wang, and Xuelong Li. Any2point: Empowering any-modality large models for efficient 3d understanding. In *European Conference on Computer Vision*, pages 456–473. Springer, 2025.

[54] Edgar Tretschk, Ayush Tewari, Vladislav Golyanik, Michael Zollhöfer, Christoph Lassner, and Christian Theobalt. Non-rigid neural radiance fields: Reconstruction and novel view synthesis of a dynamic scene from monocular video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12959–12970, 2021.

[55] Vadim Tschernezki, Iro Laina, Diane Larlus, and Andrea Vedaldi. Neural feature fusion fields: 3d distillation of self-supervised 2d image representations. In *International Conference on 3D Vision*, pages 443–453. IEEE, 2022.

[56] Feng Wang, Zilong Chen, Guokang Wang, Yafei Song, and Huaping Liu. Masked space-time hash encoding for efficient dynamic scene reconstruction. *Advances in Neural Information Processing Systems*, 36, 2024.

[57] Feng Wang, Sinan Tan, Xinghang Li, Zeyue Tian, Yafei Song, and Huaping Liu. Mixed neural voxels for fast multi-view video synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19706–19716, 2023.

[58] Zian Wang, Tianchang Shen, Merlin Nimier-David, Nicholas Sharp, Jun Gao, Alexander Keller, Sanja Fidler, Thomas Muller, and Zan Gojcic. Adaptive shells for efficient neural radiance field rendering. *ACM Transactions on Graphics*, 42:1–15, 2023.

[59] Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20310–20320, 2024.

[60] Wenqi Xian, Jia-Bin Huang, Johannes Kopf, and Changil Kim. Space-time neural irradiance fields for free-viewpoint video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9421–9431, 2021.

[61] Chi Yan, Delin Qu, Dan Xu, Bin Zhao, Zhigang Wang, Dong Wang, and Xuelong Li. Gs-slam: Dense visual slam with 3d gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19595–19604, June 2024.

[62] Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20331–20341, 2024.

[63] Heng Yu, Joel Julin, Zoltán Á Milacski, Koichiro Niinuma, and László A Jeni. Cogs: Controllable gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21624–21633, 2024.

[64] Heng Yu, Koichiro Niinuma, and László A Jeni. Confies: Controllable neural face avatars. In *2023 IEEE 17th International Conference on Automatic Face and Gesture Recognition (FG)*, pages 1–8. IEEE, 2023.

[65] Wentao Yuan, Zhaoyang Lv, Tanner Schmidt, and Steven Lovegrove. Star: Self-supervised tracking and reconstruction of rigid objects in motion with neural rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13144–13152, 2021.

[66] Raza Yunus, Jan Eric Lenssen, Michael Niemeyer, Yiyi Liao, Christian Rupprecht, Christian Theobalt, Gerard Pons-Moll, Jia-Bin Huang, Vladislav Golyanik, and Eddy Ilg. Recent trends in 3d reconstruction of general non-rigid scenes. In *Computer Graphics Forum*, page e15062. Wiley Online Library, 2024.

[67] Hao Zhang, Fang Li, and Narendra Ahuja. Open-nerf: Towards open vocabulary nerf decomposition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3456–3465, 2024.

[68] Chengwei Zheng, Wenbin Lin, and Feng Xu. Editablenerf: Editing topologically varying neural radiance fields by key points. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8317–8327, 2023.

[69] Shuaifeng Zhi, Tristan Laidlow, Stefan Leutenegger, and Andrew J Davison. In-place scene labelling and understanding with implicit scene representation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15838–15847, 2021.

[70] Ce Zhou, Qian Li, Chen Li, Jun Yu, Yixin Liu, Guan Wang, Kaichao Zhang, Cheng Ji, Qi Yan, Lifang He, Hao Peng, Jianxin Li, Jia Wu, Ziwei Liu, P. Xie, Caiming Xiong, Jian Pei, Philip S. Yu, Lichao Sun Michigan State University, B. University, Lehigh University, M. University, Nanyang Technological University, University of California at San Diego, D. University, U. Chicago, and Salesforce AI Research. A comprehensive survey on pretrained foundation models: A history from bert to chatgpt. *ArXiv*, 2023.

https://doi.org/10.52202/079017-0393

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: Both the abstract and introduction accurately reflect the contributions and scope of LiveScene.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

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

   Answer: [Yes]

   Justification: We discuss the limitations of our work in the conclusion section.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

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: [Yes]

https://doi.org/10.52202/079017-0393

Justification: The paper includes theoretical results, assumptions, and proofs, additional details can be found in the supplemental material.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

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 provide detailed information on the experimental setup and results in the main paper and supplemental material, and provide a link to our project.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

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

https://doi.org/10.52202/079017-0393

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 will provide open access to the data and code, and detailed instructions to reproduce the main experimental results. But in the review process, we only provide a anonymous link to our project.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

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 provide detailed information on the experimental setup and results in the main paper and supplemental material.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

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: [No]

   Justification: We do not report error bars or statistical significance in the paper, but the results are reproducible and the code will be released. Besides, we provide average results.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

https://doi.org/10.52202/079017-0393

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

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 detailed information on the experimental setup and results in the main paper and supplemental material.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

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 reviewed the NeurIPS Code of Ethics and believe that our research conforms to it.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

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

    Answer: [NA]

    Justification: We focus on the interactive scene reconstruction and manipulation task, and societal impacts can be ignored.

    Guidelines:

https://doi.org/10.52202/079017-0393

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

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: We focus on the interactive scene reconstruction and manipulation task, it is safe for release.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

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: We use existing datasets and models, and properly credit the original owners and respect the licenses. Besides, we reproduce the results with the released code, data and paper.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.

https://doi.org/10.52202/079017-0393

- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

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

    Answer: [Yes]

    Justification: We contribute a new dataset and code, and provide detailed documentation in the supplemental material. In the future, we will release the dataset and code with clear documentation.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

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.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

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.

    Guidelines:

https://doi.org/10.52202/079017-0393

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

https://doi.org/10.52202/079017-0393