

# RelationField: Relate Anything in Radiance Fields

Sebastian Koch<sup>1,2</sup> Johanna Wald<sup>3</sup> Mirco Colosi<sup>2</sup> Narunas Vaskevicius<sup>2</sup>  
Pedro Hermosilla<sup>4</sup> Federico Tombari<sup>3,5</sup> Timo Ropinski<sup>1</sup>

<sup>1</sup>University Ulm <sup>2</sup>Bosch Center for AI <sup>3</sup>Google <sup>4</sup>TU Vienna <sup>5</sup>TU Munich

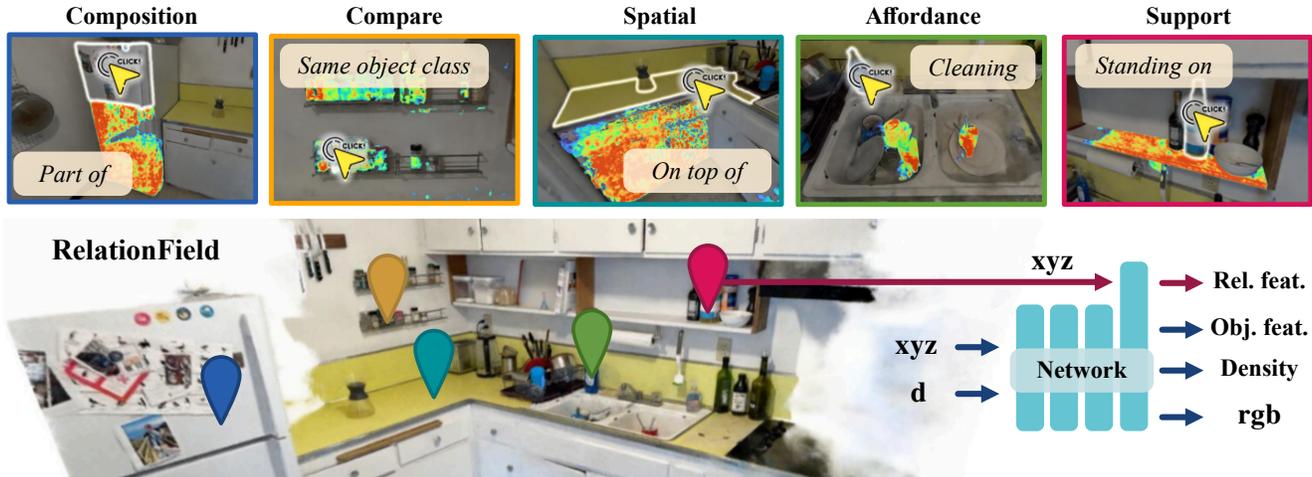


Figure 1. **Open-Vocabulary Relationship Understanding.** We propose RelationField, the first framework to extract open-vocabulary inter-object relationships directly from neural radiance fields. RelationField can answer a wide variety of relationship queries, such as “composition”, “compare”, “spatial”, “affordance” and “support” relationships.

## Abstract

Neural radiance fields are an emerging 3D scene representation and recently even been extended to learn features for scene understanding by distilling open-vocabulary features from vision-language models. However, current method primarily focus on object-centric representations, supporting object segmentation or detection, while understanding semantic relationships between objects remains largely unexplored. To address this gap, we propose RelationField, the first method to extract inter-object relationships directly from neural radiance fields. RelationField represents relationships between objects as pairs of rays within a neural radiance field, effectively extending its formulation to include implicit relationship queries. To teach RelationField complex, open-vocabulary relationships, relationship knowledge is distilled from multi-modal LLMs. To evaluate RelationField, we solve open-vocabulary 3D scene graph generation tasks and relationship-guided instance segmentation, achieving state-of-the-art performance in both tasks. See the project website at [relationfield.github.io](https://relationfield.github.io).

## 1. Introduction

3D scene understanding bridges the gap between the physi-

cal and the digital world, by enabling machines to perceive environments in a way similar to humans. In robotics, 3D scene understanding is required to navigate complex environments, interact with objects, and perform tasks autonomously. In AR/VR it enables realistic and immersive experiences, e.g., by allowing accurate placing of and interacting with virtual content in the real world. Notably, many applications require a level of understanding that goes beyond just localizing and segmenting a known list of objects categories [8, 36, 39, 47] but are also able to segment novel entities beyond the closed-set class assumption [16, 38, 51].

True holistic and adaptable scene understanding needs to go a step further and not only reconstruct and identify individual objects within a scene but also understand complex inter-object relationships, functionalities, and the overall context of the environment. This aspect of scene understanding, particularly the ability to recognize and reason about relationships between objects, is often overlooked. Yet, it is essential to interact with the surroundings in a sophisticated, adaptive and natural manner. Significant progress has been made in understanding relationships in 2D images, mainly driven by the exploration of foundation models [4, 24, 42] and in particular by multi-modal LLMs [1, 12]. These mod-

els are extremely powerful, although they primarily operate on 2D representations and do not fully leverage the richness of 3D data.

3D scenes provide more complete captures of the environment and are able to represent a high level of complexity, with overlapping objects and occlusions that make it difficult to consistently infer relationships with 2D models alone. 3D approaches have been shown to reduce per-frame noise and resolve occlusions. Despite this advantage, 3D foundation models have yet to emerge, as the data available in 3D remains limited compared to 2D.

3D scene graphs on the other hand, are a promising and compact representation for scene understanding and capture not only scene objects but also inter-object relationships. However, several scene graph approaches either rely on a closed set of relationships [25, 53, 54, 56], depend on class-agnostic instance segmentation [27], and/or require an explicit 3D representation such as point clouds.

A recent work, Open3DSG [27], distills relationship knowledge from foundation models [9, 42] into a 3D graph neural network, which can then predict open-vocabulary graphs. Capturing both objects and relationships with open-vocabulary features allows capturing a wide range of objects, functions, and relationships without prior training on specific object or relationship classes. This flexibility is crucial for handling the diversity and complexity of real-world scenes. However, Open3DSG still relies on given class-agnostic instance segmentation [27] and is bound by the quality of the explicit 3D mesh representation of the underlying dataset. These approaches furthermore require the availability of depth sensors. In contrast to 3D scene graphs, radiance fields are 3D representations that do not require 3D sensor data, but instead represent 3D scenes solely based on a set of posed 2D images [21, 35]. While they were first introduced for novel view synthesis and 3D reconstruction, they have since then been extended in several works to also capture semantic information [13, 22, 40, 48].

LERF [22], as well as a few follow-up works [13, 23, 40] present alternative approaches to distill features from 2D foundation models, such as CLIP [42], DINO [4] or SAM [24], into 3D by means of radiance fields. Yet, these approaches predominantly focus on object-centric semantic features, limiting their application in high-level scene reasoning tasks.

To enable holistic and high-level scene reasoning tasks based on neural radiance fields, we propose RelationField, a rich radiance field representation that learns open-vocabulary features for objects and their relationships. This allows us to reason about complex scenes and object interactions such as compositional, spatial, support, or affordances, see Fig. 1. In summary, this work has the following contributions:

- We present the first method for open-vocabulary scene

segmentation enabling interactive and textual relationship queries by extending the semantic neural radiance formulation with inter-object relationships distilled from a foundation model into a dense and multi-view consistent 3D representation.

- This novel representation not only facilitates relationship-based queries but also allows us to obtain state-of-the-art 3D scene graphs – making it the first time scene graphs have been inferred from neural radiance fields.
- Furthermore, we introduce a new task – relationship-guided instance segmentation – on ScanNet++ [62]. This task involves segmenting an instance based on an object-relationship search query, e.g., “picture *standing on* the shelf”, providing a benchmarking for future research in this direction.

## 2. Related Work

**Open-Vocabulary 3D Scene Understanding.** Recent 3D scene understanding approaches for detection, semantic segmentation, or instance segmentation have moved from closed-set categories [8, 36, 39, 47] to open-vocabulary, removing the limitation to a pre-defined vocabulary. To do so, 2D features from vision-language models (VLMs) are lifted into 3D by either using feature distillation and feature lifting. The latter extract vision-language features directly on 2D images and then project these to 3D by utilizing depth or by separately training 2D and 3D feature encoders that are combined at inference time [10, 16, 18, 37, 51]. Feature distillation on the other hand, trains a 3D model using semantic features extracted from a VLM from posed 2D images [13, 27, 38] and does not assume the availability of 2D frames at test time. Both feature lifting and distillation methods require 2D and 3D data either for training or for inference.

While open-vocabulary 3D scene understanding approaches have shown impressive progress in semantic object segmentation, they do not holistically capture the scene lacking knowledge about high-level compositions and/or inter-object relationships.

**Relationships in 3D Scenes.** Understanding the full 3D scene involves extracting compositional knowledge and relationships between objects and has been shown to improve object-centric predictions [28, 56]. 3D scene graphs [3, 53] have emerged as the predominant representation for modeling these relationships with applications in several different tasks such as place recognition [53], registration [46], change detection [32, 53], task planning [2, 31, 43], and navigation [55]. By representing objects as nodes into graphs and explicitly encoding their connections (spatial, semantic, etc.) as edges, 3D scene graphs offer an efficient representation of the environment. [3] proposes to represent buildings, rooms, objects, and cameras as 3D scene graphs and later works extended this idea by learning hierarchical 3D scene graphs directly from sensor data [19, 44, 45]. On the other

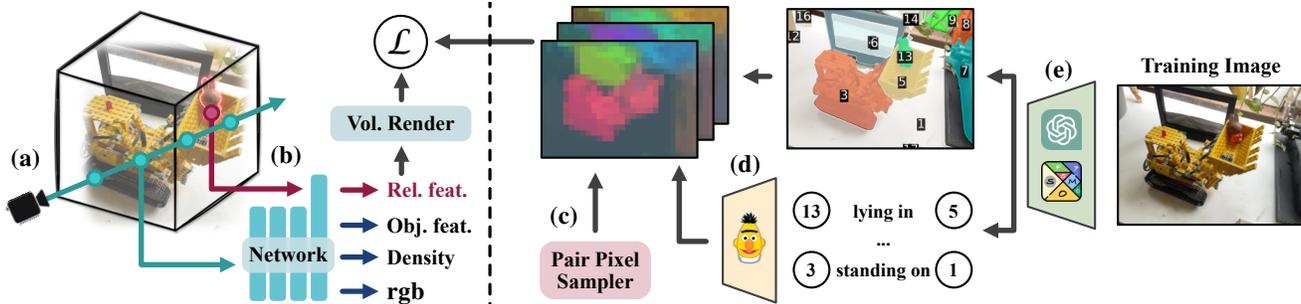


Figure 2. **RelationField Training.** *Left:* RelationField learns a 3D feature field (a) that can be queried with a relationship query location (b) which changes the relationship field of the 3D volume depending on what position is selected. The relationship feature is sampled and rendered along a ray according to NeRF’s rendering weights. The language loss maximizes the cosine similarity between the extracted sparse features from the 2D views and the rendered 3D relationship features. *Right:* We estimate 2D relationship proposals from a multi-model LLM prompted with SoM (e) for each training view and encode extracted textual relationship description into the image plane (d). A pair pixel sampler samples subject and object pixels (c) for which the relationship feature is distilled into the 3D volume.

hand, [53] introduced semantic 3D scene graphs, focusing more on the semantic components of a scene including inter-object relationships. Subsequent works have advanced this research area by refining semantic 3D scene graphs from point clouds using scene priors [64], pre-training [25, 26] and improved message passing in graphs [56, 57].

While all these works have a close-set assumption, only a few very recent works have investigated the use of VLMs and large language models (LLMs) to obtain open-vocabulary scene graphs which capture a more flexible representation of the environment [5, 6, 15, 27, 34]. However, these approaches often require depth data and a complete and explicit 3D representation of the scene e.g. in the form of a 3D mesh or point cloud [25, 27, 53] which often is not available or of poor quality.

**Radiance and Feature Fields.** Radiance Fields [21, 35] were first introduced for novel view synthesis and have the benefit that they do not require explicit 3D supervision. Recently, radiance fields have been adapted for several different 3D scene understanding tasks such as segmentation [48, 65] or detection [17, 58]. Notably, some methods propose to extend radiance fields to predict features obtained from 2D foundation models in 3D. For instance, LERF [22] and OpenNeRF [13] learn vision-language features using a separate MLP-head in the NeRF model to produce CLIP [42] embeddings for open-vocabulary 3D segmentation. Similarly, GARField [23] learns instance embeddings using a contrastive formulation provided by SAM [24] using a separate MLP-head in their NeRF. Among others, LangSplat [40] and ClickGaussians [7] extend these ideas to Gaussian Splatting for faster training and rendering. While these works show impressive results, they mainly investigate object-centric semantics and also do not explore the composition of a scene or object relationships.

Inspired by these works, our method learns open-vocabulary vision-language features directly from multiple posed

2D views. Therefore, we similarly do not require any explicit 3D scene representation in the form of depth or point cloud data. Instead, our approach aims to obtain open-vocabulary scene understanding beyond objects by also encoding object relationships, creating a consistent and rich representation. This way, our approach – as the first of its kind – supports interactive relationship queries and allows to extract 3D semantic scene graphs directly from the radiance field.

### 3. Method

Given a set of posed RGB images, our goal is to build a queryable 3D representation of the scene that supports understanding object instances using open-vocabulary object and relationship descriptions. To achieve this, we introduce a novel approach, RelationField, as illustrated in Fig. 2. Our proposed approach is independent of the underlying radiance field, and can be adapted to NeRFs [35] as well as Gaussian Splatting [21], in the following section we demonstrate how our method incorporates implicit open-set relationship feature prediction into NeRFs<sup>1</sup> [35], enabling the querying of arbitrary object and relationship concepts within a continuous volumetric 3D scene representation. To enhance NeRF with object-centric semantics, we distill CLIP-feature [42] prediction and SAM [24] supervision for instance grouping of each ray. Our method is the first to introduce an implicit open-set relationship feature prediction head as explained in Sec. 3.1. It is supervised by the embedded features of a multi-modal LLM using set-of-mark prompting (SoM) [60] (see Sec. 3.2). The learned RelationField then can be queried to retrieve relationships such as “the light switch *turns on* the lamp” by defining the predicate “turns on” as a pair of input rays within the feature field for all rays that hit the *light switch* and *lamp* (see Sec. 3.3).

<sup>1</sup>An adaption to Gaussian Splatting is detailed in the supplementary material.

### 3.1. RelationField

**Radiance Field.** A radiance field describes a function that models the color  $\mathbf{c} \in [0, 1]^3$  and density  $\sigma \in [0, \infty)$  for a given 3D point  $\mathbf{x} \in \mathbb{R}^3$  and ray direction  $\mathbf{d} \in \mathbb{S}^2$ . Mildenhall et al. [35] first proposed to model this implicit function as a neural radiance field (NeRF) that implements a multilayer perceptron  $f$  with the training objective of learning the parameters  $\theta$  with supervision from multi-images of the scene

$$f_\theta(\mathbf{x}, \mathbf{d}) \mapsto (\mathbf{c}, \sigma). \quad (1)$$

**Object-level Semantics in Radiance Fields.** To learn object-level open-vocabulary instances within the radiance field, we extend NeRF with two additional output embedding heads: one predicts open-vocabulary features  $\mathbf{s}$  in the CLIP embedding space, inspired by [13], and the other predicts a grouping embedding  $\mathbf{i}$  that co-locates rays of the same instance in the same region of the embedding space for easy instance clustering, similar to [23]. The open-vocabulary feature is therefore defined as a tuple  $\mathbf{o} = (\mathbf{s}, \mathbf{i})$  of semantic and instance features. These object-level open-vocabulary features allow us to query object entities but do not capture relationships. Therefore, it is necessary to model relationships explicitly.

**Relationship Semantics in Radiance Fields.** Unlike radiance fields, which only predict color and density for a point  $\mathbf{x}$ , relationship modeling requires an additional point  $\mathbf{z}$  to specify the relationship between  $\mathbf{x}$  and  $\mathbf{z}$ . Therefore, to capture relationships within the radiance field, we extend the input by an additional implicit query location  $\mathbf{z} \in \mathbb{R}^3$  (Fig. 2b). With this query location, our approach implicitly models the relationship feature  $\mathbf{r}$  between the ray  $(\mathbf{x}, \mathbf{d})$  and the location  $\mathbf{z}$ . The relationship feature  $\mathbf{r}$  is located within the language embedding space and can be queried for arbitrary relationships based on the cosine similarity.

The complete function  $g_\theta$  that models the color, density, open-vocabulary instance feature as well as open-vocabulary relationships of the objects in the 3D scene is given by

$$g_\theta(\mathbf{x}, \mathbf{d}, \mathbf{z}) \mapsto (\mathbf{c}, \sigma, \mathbf{o}, \mathbf{r}). \quad (2)$$

### 3.2. Relationship Supervision

While vision-language models such as CLIP [42] excel at modeling individual objects and concepts, their understanding of relationships remains limited [63]. To address this, we distill relationship knowledge from multi-modal LLMs, which better represent complex relationships. However, a challenge arises because multi-modal LLMs produce textual descriptions, while models like CLIP generate pixel- or patch-level features that can be queried using various text encodings. Our goal is to transfer relationship features into the radiance field representation, enabling open-vocabulary

querying similar to object-centric approaches with CLIP [13, 22]. The following paragraphs outline our approach for extracting such high-dimensional, pixel-aligned features from multi-modal LLMs, effectively bridging the gap between textual understanding and visual feature extraction.

**Set-of-Mark (SoM).** To extract dense pixel-aligned visual relationship features, we utilize SoM prompting [61]. SoM is a visual prompting approach that enhances the visual grounding abilities of multi-modal LLMs by overlaying marks, masks, or bounding boxes to help the model answer fine-grained visual questions. By using SoM over a direct approach, it has been shown, that it improves the spatial reasoning of LLMs, such as GPT-4 [61].

**Feature extraction.** To generate sparse high-dimensional pixel-aligned visual relationship features, we use SAM [24] to extract  $m$  segmentation masks each corresponding to a detected object in the image from a training view. Using these masks, we annotate the image with alphanumeric marks for each segmented object following the SoM prompting technique. Next, we prompt a multi-modal LLM to identify and extract inter-object relationships for closely positioned marked object pairs (Fig. 2e)<sup>2</sup>. The output text  $t$  includes a textual description of the relationships between object pairs  $(i, j)$  using the identifiers from the SoM annotations. Each textual relationship description  $t_{ij}$  is then encoded to a high-dimensional feature representation  $\phi_{t_{ij}}$  using an encoder-only language model such as [20], resulting in  $d$  dimensional features for each relationship (Fig. 2d). These features are then projected onto the image plane using the SAM segmentation masks and the SoM marks as a reference to generate a high-dimensional feature representation of the extracted relationships that are aligned with the pixel locations of the objects in the image.

**Training.** During training we randomly sample ray and query origins uniformly throughout the input views in a pairwise manner using a pair-pixel sampler (Fig. 2c). Using the density prediction of the radiance field, we estimate the query positions along the ray of the query origin. Ray and query samples are concatenated and fed together into an MLP-head that predicts the relationship feature along the sample ray. The feature is rendered onto the image plane using the radiance field’s rendering weights. We minimize a loss

$$\mathcal{L} = 1 - \frac{\mathbf{r}}{\|\mathbf{r}\|_2} \cdot \frac{\hat{\mathbf{r}}}{\|\hat{\mathbf{r}}\|_2}, \quad (3)$$

that maximizes the cosine similarity between the rendered relationship feature  $\mathbf{r}$  and the ground-truth relationship feature  $\hat{\mathbf{r}}$ . Similarly, the rendered object-centric features, such as color and open-vocabulary semantics, as well as instance

<sup>2</sup>A detailed analysis of our prompting technique is provided in the supplementary material.

features for each ray, are supervised by their respective ray origin features.

### 3.3. Querying RelationField

To effectively explore and understand the relationships between objects in a scene, it is natural to first identify and query the objects themselves before investigating their inter-object relationships. In this context, RelationField supports both object querying and subsequent relationship querying, providing a comprehensive framework for scene understanding. The querying process of RelationField consists of two steps. First, *selecting a query location*, which involves determining for which object in the scene to investigate relationships. This location can be specified directly by the user or chosen based on detected object instances. The second step requires to *query a textual relationship* for the selected object. Once a query location is chosen, users can specify a particular relationship they wish to investigate, such as “standing on” or “similar to” using a text query. Alternatively, a set of possible relationships for exploration can be provided, which is particularly useful for an open-ended investigation of the scene.

To evaluate the response of a queried relationship, we assign a score to each ray in the radiance field by calculating the cosine similarity between the language encoding of the query  $\phi_q$ , and the relationship embedding,  $\mathbf{r}$ . However, since it is difficult to interpret the cosine similarity directly without context, we follow the approach introduced by [22] and output the pairwise softmax with regard to canonical phrase embeddings  $\phi_{\text{canon}}$  such as “and”, “next to” and “none”. The relationship response is then

$$\rho = \min_i \frac{\exp(\phi_q \cdot \mathbf{r})}{\exp(\phi_{\text{canon}}^i \cdot \mathbf{r}) + \exp(\phi_q \cdot \mathbf{r})}. \quad (4)$$

Intuitively, this softmax probability represents how much the model favors a certain relationship query over no relationship.

### 3.4. Implementation Details

RelationField is built in Nerfstudio [52] on top of the Nerfacto model for color and density estimation of a given ray from posed training images with known intrinsic and optionally depth supervision. We define separate heads to estimate the open-vocabulary semantic object, instance, and relationship feature fields. The open-vocabulary segmentation head outputs 768-dimensional features in CLIP [42] / OpenSeg [14] embedding space for a given location vector without view-direction. Similarly, the instance head outputs a 256-dimensional grouping feature in the instance embedding space for a given location vector. Our relationship field encodes a pair of location vectors for the ray and query locations by concatenating them and outputs a language-aligned relationship feature of 512 dimensions in the jina-

embeddings-v3 [50] embedding space. For relationship feature supervision, we use GPT-4o [1] to extract relationship features from the training image together with SoM [60] using numeric marks and semi-transparent masks. The language outputs are encoded using jina-embeddings-v3 [50].

## 4. Experiments

In the following, we present both qualitative and quantitative results that highlight the capabilities of our method. To highlight the performance of our method in an *in-the-wild* setting, we provide a qualitative analysis of various relationship queries in different indoor environments in Sec. 4.1. To quantify RelationField performance, we leverage the task of 3D scene graph prediction in Sec. 4.2. Our approach outperforms several competitive baselines and establishes a new state-of-the-art on the 3DSSG benchmark. We then perform comprehensive ablation studies to demonstrate the importance of 3D consistency and knowledge distillation. Specifically, we compare our method against various 2D multi-modal LLMs. Further ablation studies justify our choice of relationship encoders by comparing different multi-modal LLMs for this purpose. Furthermore, we demonstrate the capabilities of our model in Sec. 4.3 by reporting its performance on a new task – *relationship-guided 3D instance segmentation* – which leverages natural language prompts e.g., “picture standing on the shelf” for 3D segmentation. Notably, our method outperforms all recent open-vocabulary feature fields, demonstrating its ability to understand object relationships accurately.

### 4.1. Relationship Segmentation

Fig. 3, shows our method’s ability to segment relationships. We visualize the model’s response for a given textual relationship prompt together with the selected target location. Results are reported on 4 different scenes taken from three datasets: LERF [22], Scannet++ [62], and Replica [49]. The scenes consist of several complex object interactions such as compositional relationships like “the freezer being *part of* the refrigerator”, support relationships such as “the pillow *lying on* the couch”, comparative or similarity relationships like “one ottoman being the *same as* another ottoman”, or even affordances such as “the light switch *turns on* the lamp”. The colormap which shows the top 50% confidence for each query respectively, shows that our model is able to segment these complex relationships.

### 4.2. 3D Scene Graph Prediction

Our method’s ability to estimate both open-vocabulary relationships as well as object instances enables the generation of 3D scene graphs. The following section details the extraction process of these graphs from our radiance field representation and presents quantitative comparisons against state-of-the-art open-vocabulary 3D scene graph prediction

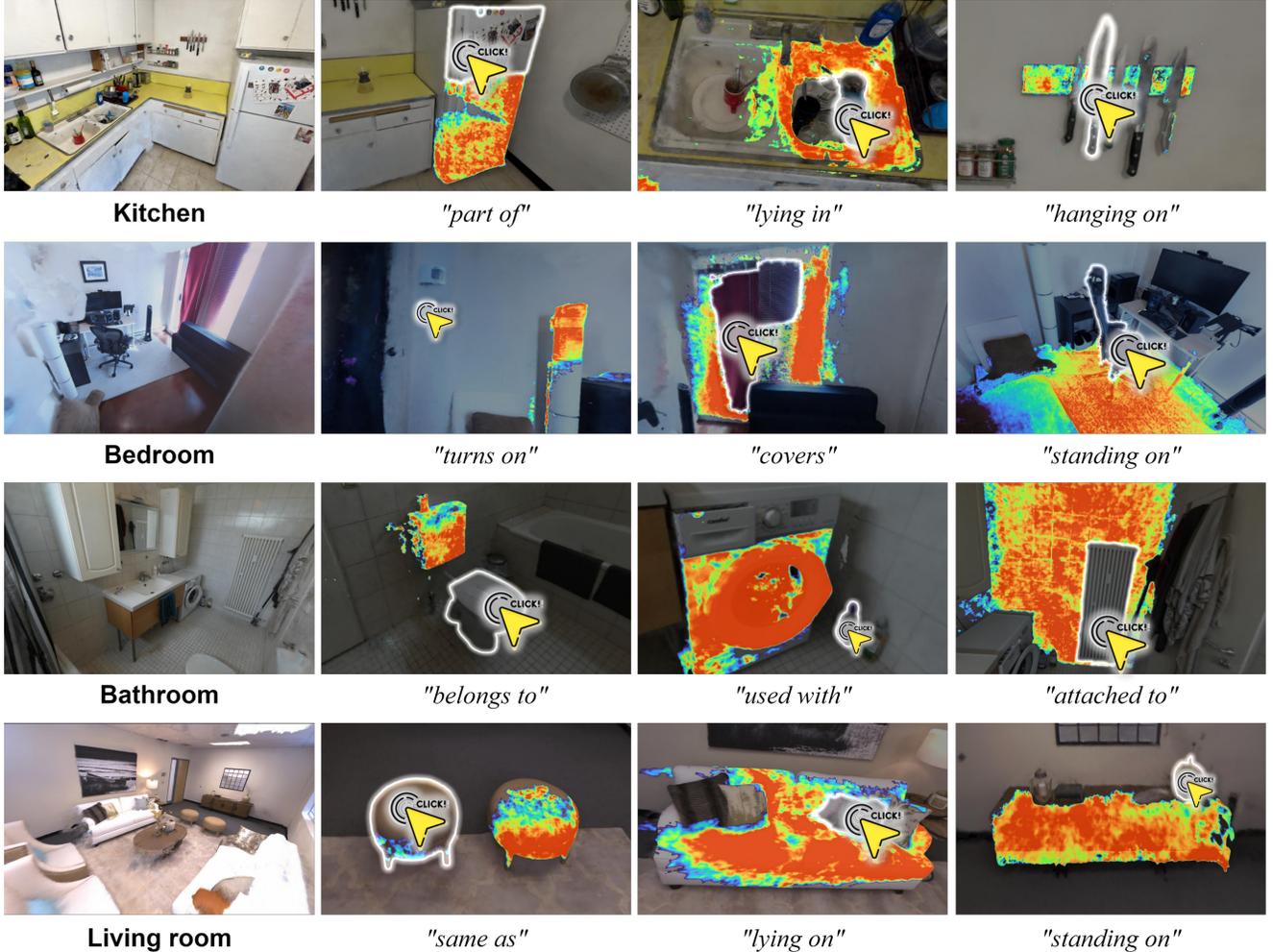


Figure 3. **Results with RelationField in 4 in-the-wild scenes.** Each image shows a rendering from RelationField, along with the relationship response for each query relationship. The relevancy score describes the answer of the model to the question: What is  *standing on/attached to/similar to etc.?* For demonstration purposes, we highlight the click as well as the outline of the clicked object, which is not needed when querying the model. Our model is able to understand complex relationships, such as the functionality of light switches or uncommon support structures, such as “knives hanging on a magnetic mount”.

models. Our proposed approach is not only able to predict open-vocabulary relationships but also open-vocabulary object instances. Combining both predictions enables the inference of open-vocabulary 3D scene graphs.

**3D Scene Graph Construction.** To extract explicit 3D scene graphs from our implicit representation requires an automated querying process. For a fair comparison with point cloud-based methods, we query the radiance field directly on the provided 3D point cloud. This ensures alignment between the extracted graph and the provided point cloud. Please note that while our method is trained solely on RGB data, the 3D point cloud is utilized exclusively for evaluation.

To do so, for each 3D point  $\mathbf{p}$  in the point cloud  $\mathcal{P}$ , we extract semantic and instance features by querying the radiance field at the given location. Since this process is viewpoint-

independent, it does not require a ray direction  $\mathbf{d}$ . We then identify instances by clustering the instance embeddings using DBSCAN [11]. For each instance  $i \in \mathcal{I}$ , the open-vocabulary object embedding  $\mathcal{S}_i$  is obtained by aggregating the respective semantic features.

To extract relationships, each instance  $i$ , comprising of points  $\mathcal{P}_i$ , serves as a query for the relationship field, which predicts relationship embeddings  $\mathcal{R}$  for the remaining point cloud. The relationship embedding  $R_{ij}$  is then obtained for each pair  $(i, j)$  by aggregating the relationship embeddings  $R_i$  for all other instances  $j \in \mathcal{I}, j \neq i$ .

Since the scene graph benchmark evaluates on a closed-set of object and relationship classes, we query with pre-defined benchmark labels. Object and relationship classes are encoded with CLIP [42] and Jina [50] respectively. We then compute the pair-wise cosine similarity between the

Method	Object		Predicate		Relationship	
	R@5	R@10	R@3	R@5	R@50	R@100
GPT-4 [1] (2D+depth)	0.34	0.42	0.55	0.58	0.52	0.54
Llama 3.2 [12] (2D+depth)	0.40	0.52	0.46	0.48	0.45	0.51
Open3DSG [27]	0.56	0.61	0.58	0.65	0.55	0.56
ConceptGraphs [15]	0.37	0.46	0.74	0.79	0.69	0.71
<b>RelationField</b>	<b>0.69</b>	<b>0.80</b>	<b>0.76</b>	<b>0.82</b>	<b>0.73</b>	<b>0.74</b>

Table 1. **3D Scene Graph Prediction on 3DSSG.** RelationField outperforms existing open-vocabulary 3D scene graph approaches as well as 2D-only frontier models. RelationField can lift different frontier models into 3D with similarly strong performance.

ground truth label encodings with the predicted embeddings. To evaluate the predictions, we use the top-k recall metric, selecting the top-k highest-scoring classes as introduced in [33]. For relationship prediction, we follow [59]; ranking our relationship predictions by multiplying the object and relationship scores.

Implementation details on the 3D scene graph extraction can be found in the supplementary.

**Data.** In the following, we report quantitative 3D scene graph evaluation results on the RIO10 subset of the 3DSSG dataset [53]. The 3DSSG dataset consists of semantic scene graphs for 3D point clouds and posed RGB-D frames obtained from a Google Tango device. It contains a closed vocabulary with 160 object classes and 27 relationship types.

**Baselines.** We compare our approach against ConceptGraphs [15], which also uses GPT-4, but in combination with a SLAM pipeline that predicts image captions. Once, the scene is reconstructed, GPT-4 is used to provide scene-consistent object and relationship caption. Additionally, we compare against Open3DSG [27], which uses a combination of CLIP [42], and InstructBLIP [9] distilled into a 3D graph neural network. Furthermore, we propose additional 2D-based baselines for GPT-4 [1] and Llama 3.2 [12], which utilize recorded depth data to lift their 2D predictions to 3D.

**Results.** A quantitative 3D scene graph comparisons is reported in Tab. 1. We query the 160 object and 27 relationship classes and obtain the embedding similarity of the language feature with the feature field and treat the extracted similarity as a label confidence. RelationField demonstrates state-of-the-art results compared to other recent open-vocabulary 3D scene graph approaches and compared to ConceptGraphs [15]. Our method demonstrates improved performance across all tasks: object, predicate as well as relationship prediction. Furthermore, 2D methods exhibit inferior performance compared to the 3D approaches, potentially due to occlusions and view-dependent challenges. Please note, our approach, compared to closed-set segmentation methods does not require any semantic labels for training and can be deployed on any dataset that provides posed RGB frames.

Fig. 4 show a subset of extracted relationships with subject, predicate, and object labels, respectively, on a scene

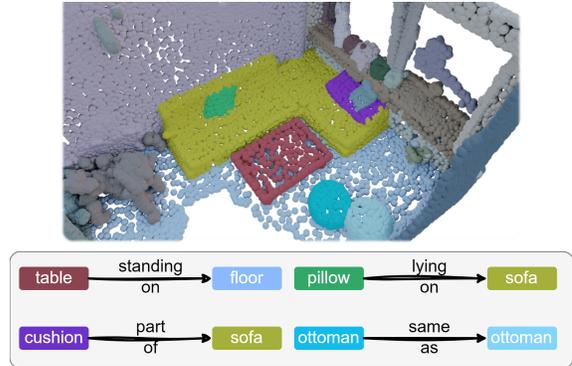


Figure 4. **3D Scene Graph Prediction.** Our open-vocabulary approach is able to predict complete 3D scene graph edges containing a subject-predicate-object relationship.

from the 3DSSG dataset. For clarity, we omit the complete graph but show the most interesting relationships. More 3D scene graph results can be found in the supplementary.

**Ablation – Advantages of 3D relationship modeling over 2D inference.** This paper demonstrates a process to distill knowledge from multi-modal LLMs such as GPT-4 into a 3D consistent representation. In Tab. 1 and Fig. 5, we analyze the benefit of a 3D representation over a 2D-only approach which directly utilizes our knowledge provider GPT-4. It can be seen that the 2D approach will always suffer from view-dependent effects. Fig. 5 shows how GPT-4 is missing the *lying on* relationship because some objects are only partially visible in the current frame. Meanwhile, when rendering the 3D prediction from RelationField, our model is able to predict the correct relationships since it relies on the underlying 3D representation. The quantitative results confirm this observation, see Tab. 1 where RelationField clearly outperforms the 2D-only GPT-4 model. This shows that our model generalizes beyond simple view-level supervision and, indeed, learns a consistent 3D representation, which improves over simple aggregated 2D inference.

**Ablation – Impact of multi-modal LLM choice on relationship understanding.** While we utilize GPT-4 as our backbone model for extracting relationships, our approach is agnostic to the backbone model and can accommodate any LLM capable of reasoning about object relationships. In Fig. 6, we compare our approach which is using the latest version of GPT-4o against the popular open-source alternative Llama 3.2 [12] (90B). Llama 3.2, which is considerably smaller than GPT-4o, has only a minor recall drop for relationship prediction. This shows that our model can be trained with any sufficiently powerful multi-modal LLM.

### 4.3. Relationship-guided 3D Instance Segmentation

To highlight the advantages of understanding relationships, we propose a new evaluation task for quantitative relationship-guided 3D instance segmentation. In this task,

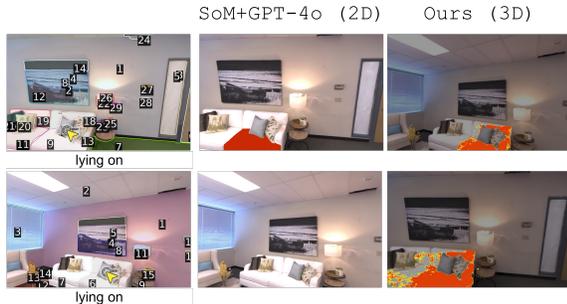


Figure 5. **3D Consistency Ablation.** *Left:* Extracted SoM marks per image with  query. *Center:* Existing relationship in GPT-4 caption. *Right:* Relationship response from RelationField rendered into image space. While GPT-4 struggles with partially visible objects, RelationField produces more robust results, independent of the view, because our volumetric rendering incorporates information from multiple views and models the underlying 3D relationship representation.

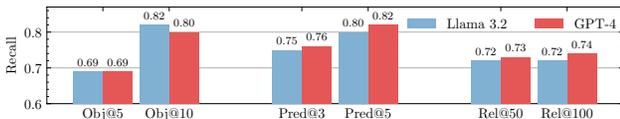


Figure 6. **Language Model Ablation.** We compare GPT-4 with Llama 3.2 as the relationship extractor of RelationField for 3D scene graph prediction.

we want to highlight the benefit of understanding relationships from open-vocabulary textual descriptions for localizing objects of interest.

**Data.** We label a small benchmark on Scannet++ [62] of language-based relationship queries across 8 scenes with instance annotations for  $\sim 30$  relationship queries spanning  $\sim 40$  unique object types and  $\sim 10$  semantic predicates. More details can be found in the supplementary.

**Baselines.** For a fair comparison, we compare RelationField against three state-of-the-art feature field methods for open-vocabulary object segmentation, LERF [22], OpenNeRF [13], and LangSplat [40] which all rely on posed RGB for training and inference. All approaches are able to process open-vocabulary queries in natural language and localize them in the 3D scene by associating the CLIP [42] embedding of the query with the learned features in the NeRF. The experiments show that our approach is the only capable method to reliably understand complex prompts such as “the picture standing on the shelf” explicitly.

**Localization.** To localize target queries with RelationField, we split the language queries into nouns and verbs. First, the nouns are localized using the object field by computing the cosine-similarity to the nouns in the language query. Then, we refine the localization by combining the object prediction with the relationship embedding by rejecting all candidate predictions that do not have a relationship feature

Method	IoU	Acc
LERF [22]	0.25	0.50
OpenNeRF [13]	0.45	0.83
LangSplat [40]	0.49	0.87
<b>RelationField</b>	<b>0.53</b>	<b>0.96</b>

Table 2. **Open-Vocabulary relationship-guided Instance Segmentation.** Comparison of open-vocabulary radiance field-based methods on instance segmentation performance for challenging relationship queries.

aligned with the verb from the query. For LERF, OpenNeRF and LangSplat, the full query is processed directly, as these models do not distinguish between verbs and nouns in their query parsing.

**Results.** In Tab. 2, we report the segmentation accuracy and IoU for the set of target queries. The performance of LERF, OpenNeRF and LangSplat degrades in this specialized setting where all queries contain complex relationships. We observe most failure cases for duplicate objects where the *bag-of-words* representation of CLIP cannot differentiate these objects by their relationship. Meanwhile, RelationField clearly outperforms LangSplat, OpenNeRF and LERF since it is able to model the relationship feature directly.

## 5. Limitations

The experiments conducted in this paper demonstrate the potential and advantages of learning 3D relationships in radiance fields. However certain limitations remain. For instance, the relationship knowledge embedded in RelationField is highly dependent on the multi-modal LLM prompting and its output. Furthermore, while posed RGB recordings are easier to acquire than point clouds, RelationField requires known calibrated camera intrinsics and high-quality multi-view captures, which are not always available or easy to capture. In general, the quality of RelationField is bounded by the quality of the radiance field reconstruction.

## 6. Conclusions

In this paper, we present RelationField, the first 3D scene representation based on radiance fields that allow for open-vocabulary object and relationship queries. By distilling knowledge from 2D multi-modal LLMs into radiance fields, we are able to not only extract relationship information but also to obtain state-of-the-art open-vocabulary 3D scene graphs. We demonstrate that RelationField effectively learns a consistent 3D representation that surpasses the performance of simple aggregated 2D inference. Furthermore, we introduce a new task of relationship-guided 3D instance segmentation, to highlight the importance of understanding relationships for localizing objects of interest. We hope this work will encourage future 3D scene understanding techniques to not only focus on object-centric features but explicitly incorporate the relations between them.

**Acknowledgement.** We sincerely thank Jonathan Francis for providing support in running Llama 3.2 experiments. Our appreciation extends to David Adrian for helpful discussions. We are also grateful to Timm Linder and Andrey Rudenko for their proofreading. This work was partly supported by the EU Horizon 2020 research and innovation program under grant agreement No. 101017274 (DARKO).

## References

- [1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 1, 5, 7
- [2] Christopher Agia, Krishna Murthy Jatavallabhula, Mohamed Khodeir, Ondrej Miksik, Vibhav Vineet, Mustafa Mukadam, Liam Paull, and Florian Shkurti. Taskography: Evaluating robot task planning over large 3d scene graphs. In *Proceedings of the 5th Conference on Robot Learning*, pages 46–58. PMLR, 2022. 2
- [3] Iro Armeni, Zhi-Yang He, JunYoung Gwak, Amir R. Zamir, Martin Fischer, Jitendra Malik, and Silvio Savarese. 3d scene graph: A structure for unified semantics, 3d space, and camera. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 2
- [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 International Conference on Computer Vision (ICCV)*, 2021. dino. 1, 2
- [5] Lianggangxu Chen, Xuejiao Wang, Jiale Lu, Shaohui Lin, Changbo Wang, and Gaoqi He. Clip-driven open-vocabulary 3d scene graph generation via cross-modality contrastive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 27863–27873, 2024. 3
- [6] An-Chieh Cheng, Hongxu Yin, Yang Fu, Qiushan Guo, Ruihan Yang, Jan Kautz, Xiaolong Wang, and Sifei Liu. Spatialrgpt: Grounded spatial reasoning in vision-language models. In *NeurIPS*, 2024. 3
- [7] Seokhun Choi, Hyeonseop Song, Jaechul Kim, Taehyeong Kim, and Hoseok Do. Click-gaussian: Interactive segmentation to any 3d gaussians. *arXiv preprint arXiv:2407.11793*, 2024. 3
- [8] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3075–3084, 2019. 1, 2
- [9] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *arXiv preprint arXiv:2305.06500*, 2023. 2, 7
- [10] Alexandros Delintzas, Ayca Takmaz, Federico Tombari, Robert Sumner, Marc Pollefeys, and Francis Engelmann. Scenefun3d: Fine-grained functionality and affordance understanding in 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14531–14542, 2024. 2
- [11] Dingsheng Deng. Dbscan clustering algorithm based on density. In *2020 7th International Forum on Electrical Engineering and Automation (IFEEA)*, pages 949–953, 2020. 6
- [12] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. 1, 7
- [13] Francis Engelmann, Fabian Manhardt, Michael Niemeyer, Keisuke Tateno, Marc Pollefeys, and Federico Tombari. OpenNerf: Open Set 3D Neural Scene Segmentation with Pixel-Wise Features and Rendered Novel Views. In *International Conference on Learning Representations*, 2024. 2, 3, 4, 8, 1
- [14] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *European Conference on Computer Vision (ECCV)*, pages 540–557, 2022. 5, 1
- [15] Qiao Gu, Ali Kuwajerwala, Sacha Morin, Krishna Murthy Jatavallabhula, Bipasha Sen, Aditya Agarwal, Corban Rivera, William Paul, Kirsty Ellis, Rama Chellappa, Chuang Gan, Celso Miguel de Melo, Joshua B. Tenenbaum, Antonio Torralba, Florian Shkurti, and Liam Paull. Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5021–5028, 2024. 3, 7
- [16] Huy Ha and Shuran Song. Semantic abstraction: Open-world 3D scene understanding from 2D vision-language models. In *Proceedings of the 2022 Conference on Robot Learning*, 2022. 1, 2
- [17] Benran Hu, Junkai Huang, Yichen Liu, Yu-Wing Tai, and Chi-Keung Tang. Nerf-rpn: A general framework for object detection in nerfs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 23528–23538, 2023. 3
- [18] Zhening Huang, Xiaoyang Wu, Xi Chen, Hengshuang Zhao, Lei Zhu, and Joan Lasenby. Openins3d: Snap and lookup for 3d open-vocabulary instance segmentation. In *European Conference on Computer Vision (ECCV)*, 2024. 2
- [19] N. Hughes, Y. Chang, and L. Carlone. Hydra: A real-time spatial perception system for 3D scene graph construction and optimization. In *Robotics: Science and Systems (RSS)*, 2022. 2
- [20] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT*, page 2. Minneapolis, Minnesota, 2019. 4, 1
- [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), 2023. 2, 3
- [22] Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. Lrf: Language embed-

- ded radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 19729–19739, 2023. 2, 3, 4, 5, 8, 1
- [23] 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 (CVPR)*, pages 21530–21539, 2024. 2, 3, 4, 1
- [24] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4015–4026, 2023. 1, 2, 3, 4
- [25] Sebastian Koch, Pedro Hermosilla, Narunas Vaskevicius, Mirco Colosi, and Timo Ropinski. Sgrec3d: Self-supervised 3d scene graph learning via object-level scene reconstruction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 3404–3414, 2024. 2, 3
- [26] Sebastian Koch, Pedro Hermosilla, Narunas Vaskevicius, Mirco Colosi, and Timo Ropinski. Lang3dsg: Language-based contrastive pre-training for 3d scene graph prediction. In *2024 International Conference on 3D Vision (3DV)*, pages 1037–1047. IEEE, 2024. 3
- [27] Sebastian Koch, Narunas Vaskevicius, Mirco Colosi, Pedro Hermosilla, and Timo Ropinski. Open3dsg: Open-vocabulary 3d scene graphs from point clouds with queryable objects and open-set relationships. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14183–14193, 2024. 2, 3, 7
- [28] Nilesh Kulkarni, Ishan Misra, Shubham Tulsiani, and Abhinav Gupta. 3d-relnet: Joint object and relational network for 3d prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 2
- [29] Hyunjee Lee, Youngsik Yun, Jeongmin Bae, Seoha Kim, and Youngjung Uh. Rethinking open-vocabulary segmentation of radiance fields in 3d space. *arXiv preprint arXiv:2408.07416*, 2024. 4
- [30] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019. 1
- [31] Yuchen Liu, Luigi Palmieri, Sebastian Koch, Ilche Georgievski, and Marco Aiello. Delta: Decomposed efficient long-term robot task planning using large language models. *arXiv preprint arXiv:2404.03275*, 2024. 2
- [32] Samuel Looper, Javier Rodriguez-Puigvert, Roland Siegwart, Cesar Cadena, and Lukas Schmid. 3d vsg: Long-term semantic scene change prediction through 3d variable scene graphs. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8179–8186. IEEE, 2023. 2
- [33] Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Visual relationship detection with language priors. In *Computer Vision—ECCV 2016: 14th European Conference, Proceedings, Part I 14*, pages 852–869. Springer, 2016. 7
- [34] Dominic Maggio, Yun Chang, Nathan Hughes, Matthew Trang, Dan Griffith, Carlyn Dougherty, Eric Cristofalo, Lukas Schmid, and Luca Carlone. Clio: Real-time task-driven open-set 3d scene graphs. *IEEE Robotics and Automation Letters*, 9(10):8921–8928, 2024. 3
- [35] 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. In *European Conference on Computer Vision (ECCV)*, pages 405–421. Springer International Publishing, 2020. 2, 3, 4
- [36] Ishan Misra, Rohit Girdhar, and Armand Joulin. An End-to-End Transformer Model for 3D Object Detection. In *ICCV*, 2021. 1, 2
- [37] Phuc Nguyen, Tuan Duc Ngo, Evangelos Kalogerakis, Chuang Gan, Anh Tran, Cuong Pham, and Khoi Nguyen. Open3dis: Open-vocabulary 3d instance segmentation with 2d mask guidance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4018–4028, 2024. 2
- [38] Songyou Peng, Kyle Genova, Chiyu “Max” Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. Openscene: 3d scene understanding with open vocabularies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–824, 2023. 1, 2
- [39] Charles R Qi, Or Litany, Kaiming He, and Leonidas J Guibas. Deep hough voting for 3d object detection in point clouds. In *Proceedings of the IEEE International Conference on Computer Vision*, 2019. 1, 2
- [40] 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 (CVPR)*, pages 20051–20060, 2024. 2, 3, 8
- [41] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019. 1, 3
- [42] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. CLIP. 1, 2, 3, 4, 5, 6, 7, 8
- [43] Krishan Rana, Jesse Haviland, Sourav Garg, Jad Abou-Chakra, Ian Reid, and Niko Suenderhauf. Sayplan: Grounding large language models using 3d scene graphs for scalable task planning. In *7th Annual Conference on Robot Learning*, 2023. 2
- [44] Antoni Rosinol, Arjun Gupta, Marcus Abate, Jingnan Shi, and Luca Carlone. 3d dynamic scene graphs: Actionable spatial perception with places, objects, and humans. In *Robotics: Science and Systems (RSS)*, 2020. 2
- [45] Antoni Rosinol, Andrew Violette, Marcus Abate, Nathan Hughes, Yun Chang, Jingnan Shi, Arjun Gupta, and Luca Carlone. Kimera: From slam to spatial perception with 3d

- dynamic scene graphs. *The International Journal of Robotics Research*, 40(12-14):1510–1546, 2021. 2
- [46] Sayan Deb Sarkar, Ondrej Miksik, Marc Pollefeys, Daniel Barath, and Iro Armeni. Sgaligner: 3d scene alignment with scene graphs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 21927–21937, 2023. 2
- [47] Jonas Schult, Francis Engelmann, Alexander Hermans, Or Litany, Siyu Tang, and Bastian Leibe. Mask3D: Mask Transformer for 3D Semantic Instance Segmentation. In *International Conference on Robotics and Automation (ICRA)*, 2023. 1, 2
- [48] Yawar Siddiqui, Lorenzo Porzi, Samuel Rota Bulò, Norman Müller, Matthias Nießner, Angela Dai, and Peter Kotschieder. Panoptic lifting for 3d scene understanding with neural fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9043–9052, 2023. 2, 3
- [49] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, et al. The replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019. 5
- [50] Saba Sturua, Isabelle Mohr, Mohammad Kalim Akram, Michael Günther, Bo Wang, Markus Krimmel, Feng Wang, Georgios Mitrakos, Andreas Koukounas, Andreas Koukounas, Nan Wang, and Han Xiao. jina-embeddings-v3: Multilingual embeddings with task lora, 2023. 5, 6, 1
- [51] Ayça Takmaz, Elisabetta Fedele, Robert W. Sumner, Marc Pollefeys, Federico Tombari, and Francis Engelmann. OpenMask3D: Open-Vocabulary 3D Instance Segmentation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 1, 2
- [52] Matthew Tancik, Ethan Weber, Evonne Ng, Ruilong Li, Brent Yi, Terrance Wang, Alexander Kristoffersen, Jake Austin, Kamyar Salahi, Abhik Ahuja, et al. Nerfstudio: A modular framework for neural radiance field development. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–12, 2023. 5
- [53] Johanna Wald, Helisa Dharmo, Nassir Navab, and Federico Tombari. Learning 3d semantic scene graphs from 3d indoor reconstructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2, 3, 7
- [54] Ziqin Wang, Bowen Cheng, Lichen Zhao, Dong Xu, Yang Tang, and Lu Sheng. VI-sat: Visual-linguistic semantics assisted training for 3d semantic scene graph prediction in point cloud. *arXiv preprint arXiv:2303.14408*, 2023. 2
- [55] Abdelrhman Werby, Chenguang Huang, Martin Büchner, Abhinav Valada, and Wolfram Burgard. Hierarchical Open-Vocabulary 3D Scene Graphs for Language-Grounded Robot Navigation. In *Proceedings of Robotics: Science and Systems*, Delft, Netherlands, 2024. 2
- [56] Shun-Cheng Wu, Johanna Wald, Keisuke Tateno, Nassir Navab, and Federico Tombari. Scenegrphfusion: Incremental 3d scene graph prediction from rgb-d sequences. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7515–7525, 2021. 2, 3
- [57] Shun-Cheng Wu, Keisuke Tateno, Nassir Navab, and Federico Tombari. Incremental 3d semantic scene graph prediction from rgb sequences. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5064–5074, 2023. 3
- [58] Chenfeng Xu, Bichen Wu, Ji Hou, Sam Tsai, Ruilong Li, Jialiang Wang, Wei Zhan, Zijian He, Peter Vajda, Kurt Keutzer, and Masayoshi Tomizuka. Nerf-det: Learning geometry-aware volumetric representation for multi-view 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 23320–23330, 2023. 3
- [59] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 670–685, 2018. 7
- [60] Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv preprint arXiv:2310.11441*, 2023. 3, 5
- [61] Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv preprint arXiv:2310.11441*, 2023. 4, 1
- [62] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12–22, 2023. 2, 5, 8
- [63] Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it? *arXiv preprint arXiv:2210.01936*, 2022. 4
- [64] Shoulong Zhang, Shuai Li, Aimin Hao, and Hong Qin. Knowledge-inspired 3d scene graph prediction in point cloud. In *Advances in Neural Information Processing Systems*, pages 18620–18632. Curran Associates, Inc., 2021. 3
- [65] 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 (ICCV)*, 2021. 3

# RelationField: Relate Anything in Radiance Fields

## Supplementary Material

In this **supplementary material**, we first provide additional training details in Sec. A. Next, we offer further insights into our design choices for RelationField in Sec. B. Sec. C contains additional details on the scene graph extraction. We then present qualitative results for relationship segmentation and the relationship-guided 3D instance segmentation task in Sec. D. In Sec. E, we include an adaption of RelationField to Gaussian Splatting together with a quality comparison. Finally, we provide examples from our curated relationship-guided 3D instance segmentation benchmark in Sec. F.

### A. Training details

To accelerate training speed, existing feature field approaches [13, 22, 23] store extracted 2D training features to disk and load them in RAM at training start for efficient retrieval at each training step instead of computing the features online. However, storing all relationship features in RAM or disk is infeasible for RelationField since for  $n$  input images of shape  $w, h$ , with  $m$  generated masks it would require storing  $n \times m \times (m - 1) \times w \times h \times d$  relationship features. This would result in  $\sim 5.66TB$  when we store the features in FP16 and assume 10 instances per image for a scene of 200 images each with VGA resolution of  $640 \times 480$ . Instead, we optimize the memory resources by storing a dictionary of all relationship features in combination with a singular segmentation mask and compute the relationship map for each sampled pixel-pair using a two-step lookup in the segmentation map and then in the relationship dictionary. Using this strategy we are able to reduce the memory requirements to  $\sim 500MB$  per scene when using FP16 precision. Inspired by [23], we begin training the relationship field after 2000 steps of NeRF optimization to let the geometry converge. We train for 30000 steps on a single Nvidia A100, which takes around 60 minutes and consumes around 40GB of GPU memory. The feature extraction of the object features from OpenSeg [14] and SAM [24], as well as the relationship features with GPT-4 increases the training time by about 30 minutes for the first run.

### B. Design choices

**Prompting.** To extract textual relationships using GPT-4 [1] or Llama [12], we employ a combination of visual and textual prompting. For visual prompting, we utilize SoM [61] to overlay semi-transparent masks and numeric marks. The textual prompt consists of a two-stage approach which queries the model first to extract objects by their mark-id and then to extract relationships referenced by the previ-

ously extracted object-ids together with a relationship label. The complete prompt looks as follows:

**1. Object Identification:** Identify all objects in the image by their tag. Create a dict that maps tag\_id to class\_name.

**2. Affordance/Relationship Detection:** For every pair of tagged objects that are clearly related, describe the semantic relationships and affordances as a list of dictionaries using the format [s\_id: #n1, subject\_class: x, o\_id: #n2, object\_class: y, predicates: [p1, p2, ...]]. For subjects and objects sharing multiple relationships/affordances, concatenate predicates with a comma in the [predicate] field.

- Avoid generic terms like "next to" for ambiguous relationships. Instead, specify relationships with precise relationships and affordances describing spatial relationships [over/under etc.], comparative relationships [larger/smaller than, similar/same type/color], functional relationships [part of/belonging to, turns on], support relationships [standing on, hanging on, lying on, attached to].

- Do not use left/right; always use 3D consistent relationships.

- Always combine a spatial relationship with a semantic, comparative, functional or support relationship using a comma (e.g., [A] [above, lying on] [B]).

- For symmetrical relationships, include both directions (e.g., [A] [above] [B] and [B] [below] [A]).

- Even for distant objects highlight if they are [same/similar/same color/same object type]

**Example Output:**

```
objects = {4: floor, 7: table, 12: chair, ...}
relationships_affordances = {
[s_id: 4, subject_class: table, o_id: 7, object_class: floor, predicates:
standing on],
[s_id: 12, subject_class: chair, o_id: 13, object_class: chair, predicates:
next to, same as],
[s_id: 6, subject_class: pillow, o_id: 8, object_class: couch, predicates:
belongs to],
...
}
```

After processing the image frames with the LLM we parse the output into a JSON format in an automatic manner.

**Text encoder.** To embed relationships in RelationField, we encode the output from a multi-modal LLM into the radiance field using an encoder-only language model. The choice of the encoder is important since it determines the structure and queryability of the embedding space in the radiance field. We want an embedding space, that is highly structured and embeds similar (relationship) concepts close together, while contradictory relationships are supposed to be far apart in embedding space. In Fig. 7, we provide an analysis for different popular open-source text encoders such as CLIP [42], BERT [20], Jina-v3 [50], RoBERTa [30] and GPT-2 [41]. We have a set of 41 distinct relationships with varying semantic similarity to each other and plot their pair-wise cosine similarity in a similarity matrix. We observe that Jina-v3-

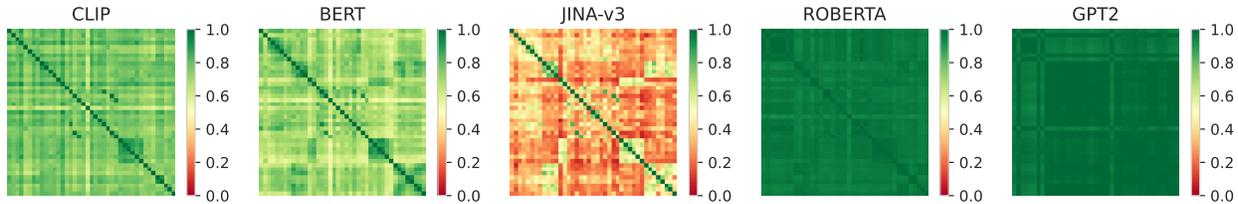


Figure 7. **Language Encoder Ablation.** We compare 5 language encoder-only model based on their separability in embedding space. For each language encoder, we plot a similarity matrix, for the pairwise cosine-similarity of 41 predicates taken from the 3DSSG dataset.

embeddings generate the most well-structured feature space, where related concepts exhibit a strong similarity, while the majority of relationships describing distinct concepts show a high degree of dissimilarity. As a counter-example, both RoBERTa and GPT2 embed all relationships in a very similar feature space, which would make fine-grained querying difficult.

**Relationship direction.** In Fig. 3 of the main manuscript, we present qualitative results from RelationField on 4 scenes. In these results, we present queries of the form “What is  $\langle \text{query} \rangle$  *standing on/attached to/similar to etc.*?”. In this scenario, we are interested in the object of a *subject-predicate-object* relationship. However similarly, it can be interesting to investigate to query the subject of a *subject-predicate-object* relationship by answering the question “What is *standing on/attached to/similar to etc.*  $\langle \text{query} \rangle$ ?”. To model this question in RelationField we simply have to invert the supervision signal during training by swapping the query ray origin with the ray origin. In Fig. 8, we demonstrate different directional relationship queries for the same objects and predicate.

### C. Scene Graph Construction

In Fig. 9, we visually supplement the reported process of extracting a 3D scene graph from RelationField. First, we extract groups of points from the instance field. These groups of points (Fig. 9a), serve as the queries for the RelationField and represent the subject in a *subject-predicate-object* relationship edge. In a second step, the RelationField gets evaluated on the remaining points of the point cloud given the query points and a textual relationship prompt such as lying on (Fig. 9b). The textual query represents the predicate in the *subject-predicate-object* relationship. This step returns a relationship activation map for the entire point cloud with each point having a unique relationship response. In the third step, the activations get aggregated based on the instance head (Fig. 9c). The instances that have a relationship response greater than a threshold of 0.5 represent objects in the *subject-predicate-object* relationship edge. Finally all edges for objects surpassing the threshold are added to the 3D scene graph.

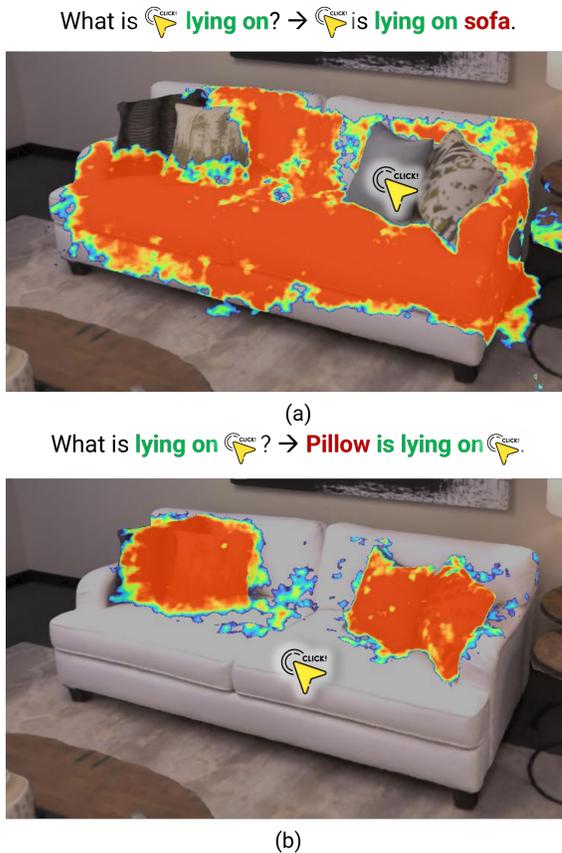


Figure 8. **Relationship Direction.** (a) visualizes the relationship response for the question “What is  $\langle \text{query} \rangle$  *standing on/lying on/similar to?*”, where we localize the object in a *subject-predicate-object* relationship. While (b) visualizes the relationship response for the question “What is *standing on/lying on/similar to*  $\langle \text{query} \rangle$ ?”, where we localize the subject in a *subject-predicate-object* relationship.

### D. Qualitative Results

**Relationship querying.** In Fig. 10, we present qualitative results for 4 additional scenes for the relationship querying with RelationField.

**Relationship-guided instance segmentation.** In Fig. 12, we qualitatively compare the 3D instance segmentation of RelationField against OpenNeRF [13] for relationship

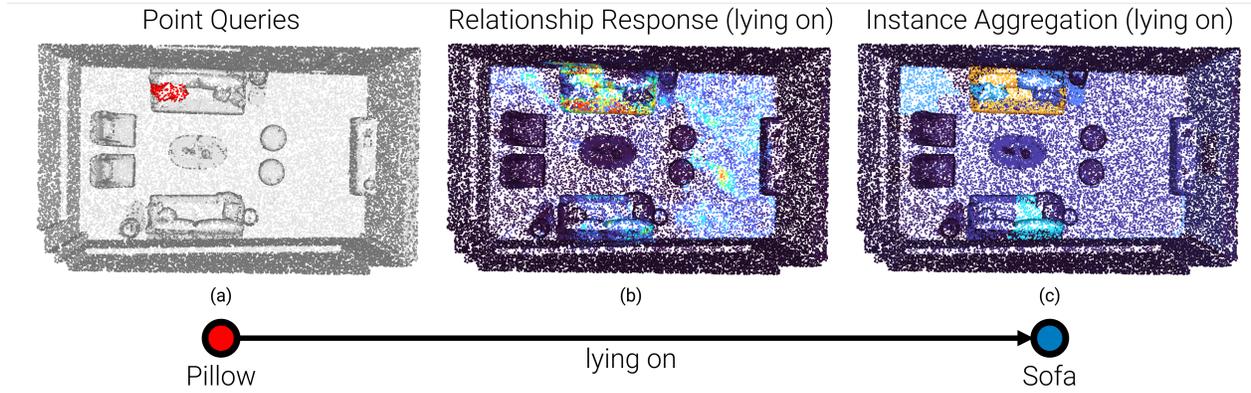


Figure 9. **Relationship Edge Construction.** To extract a 3D scene graph from RelationField, we automatically query instances (a), compute the relationship response for predicates such as “lying on” (b), and aggregate the relationship response for each instance (c). We add an edge to the scene graph for all objects whose relationship response for the subject and predicate is greater than a certain threshold.

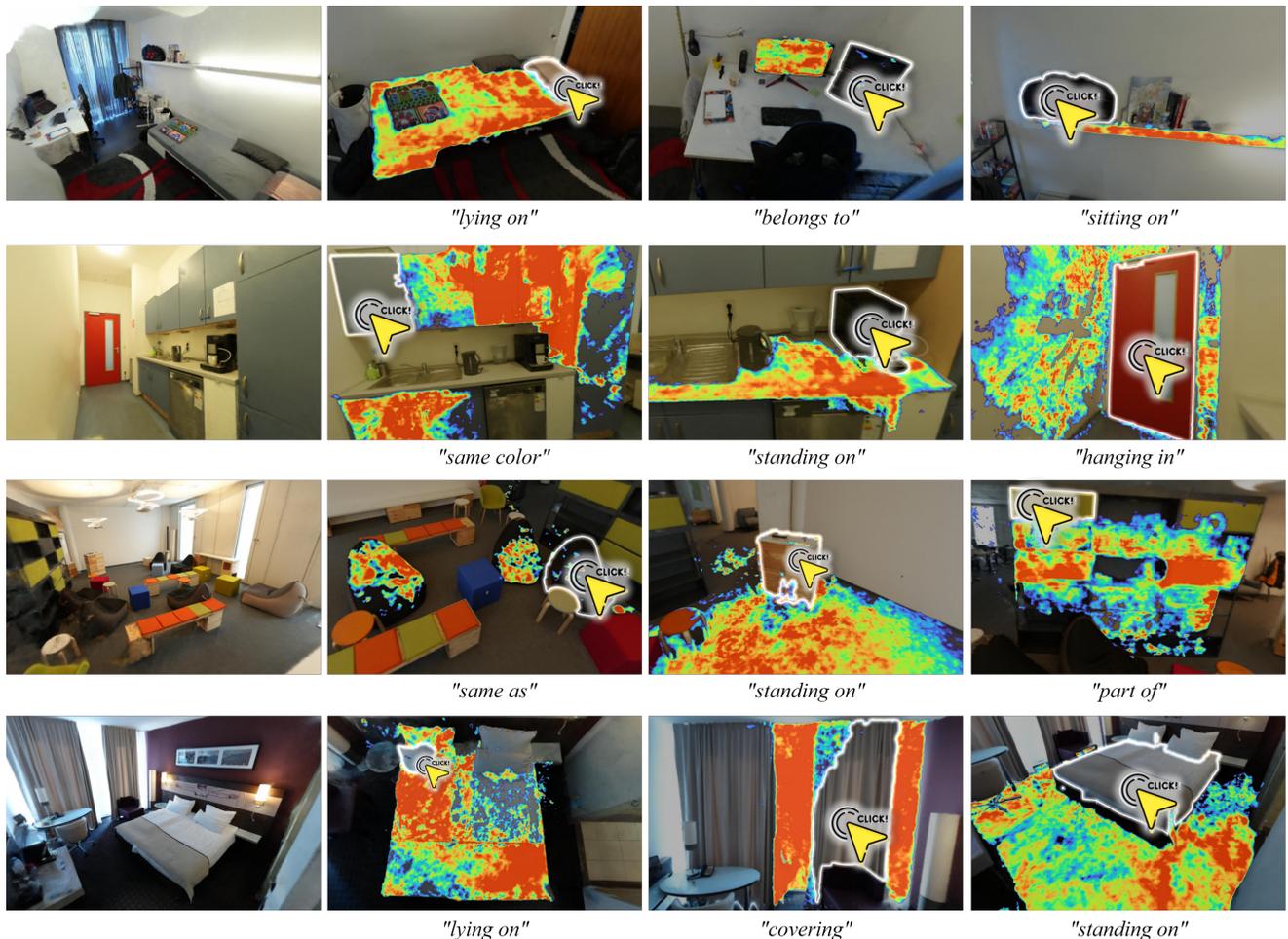


Figure 10. **Additional Qualitative Results.** We provide relationship responses for 4 additional scenes from Scannet++. The colormap visualizes the relationship response where blue is low and red is high. We visualize the relationships for the question: “What is  standing on/lying on/similar to?”

queries to supplement Tab. 2. OpenNeRF produces many false positives because it gets confused with the compo-

sitional queries arising from the bag-of-words behavior of CLIP [41]. Meanwhile, RelationField uses the object infor-

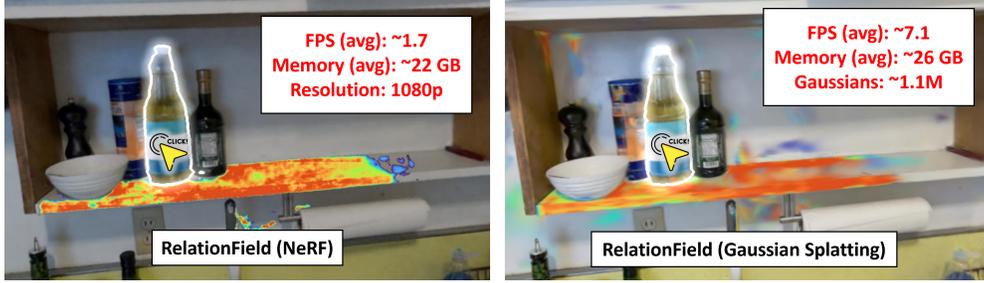


Figure 11. **RelationField w/ NeRF or w/ Gaussian Splatting geometry**. We compare the rendering speed (FPS), memory requirements and RelationField quality for the query “standing on”.

mation together with the relationship information from the prompt to accurately filter predictions that only correspond to the object in the prompt that has the described relationship.

## E. Gaussian Splatting Support

We build RelationField on NeRF [35], however since our approach is independent of the underlying 3D representation, it is possible to extend RelationField to Gaussian Splatting for faster training, inference and rendering. To train RelationField, we follow [29] and initialize the Gaussian Splatting training run with the exported point cloud of the NeRF training. This results in faster convergence and fewer Gaussians leading to improved memory utilization. For RelationField with Gaussian Splatting geometry, we reformulate our relationship definition from a pair of rays to a pair of 3D Gaussian centers. In Fig. 11, we compare the rendering speed, memory requirements and RelationField quality. RelationField based on Gaussian Splatting achieves 4x faster rendering compared to its NeRF variant with a lower memory footprint. Overall, Fig. 11 shows that RelationField is independent of the underlying geometry, and both NeRF and 3DGS produce high-quality RelationFields.

## F. Relationship-guided 3D Instance Segmentation Dataset

In Fig. 13, we present a subset of the annotated benchmark which we present in Sec 4.1 of the main paper. In the benchmark, we provide instance segmentations paired with textual relationship prompts. When curating the benchmark we focused on samples that appear multiple times in the scene, but which can be uniquely referenced by a relationship prompt.

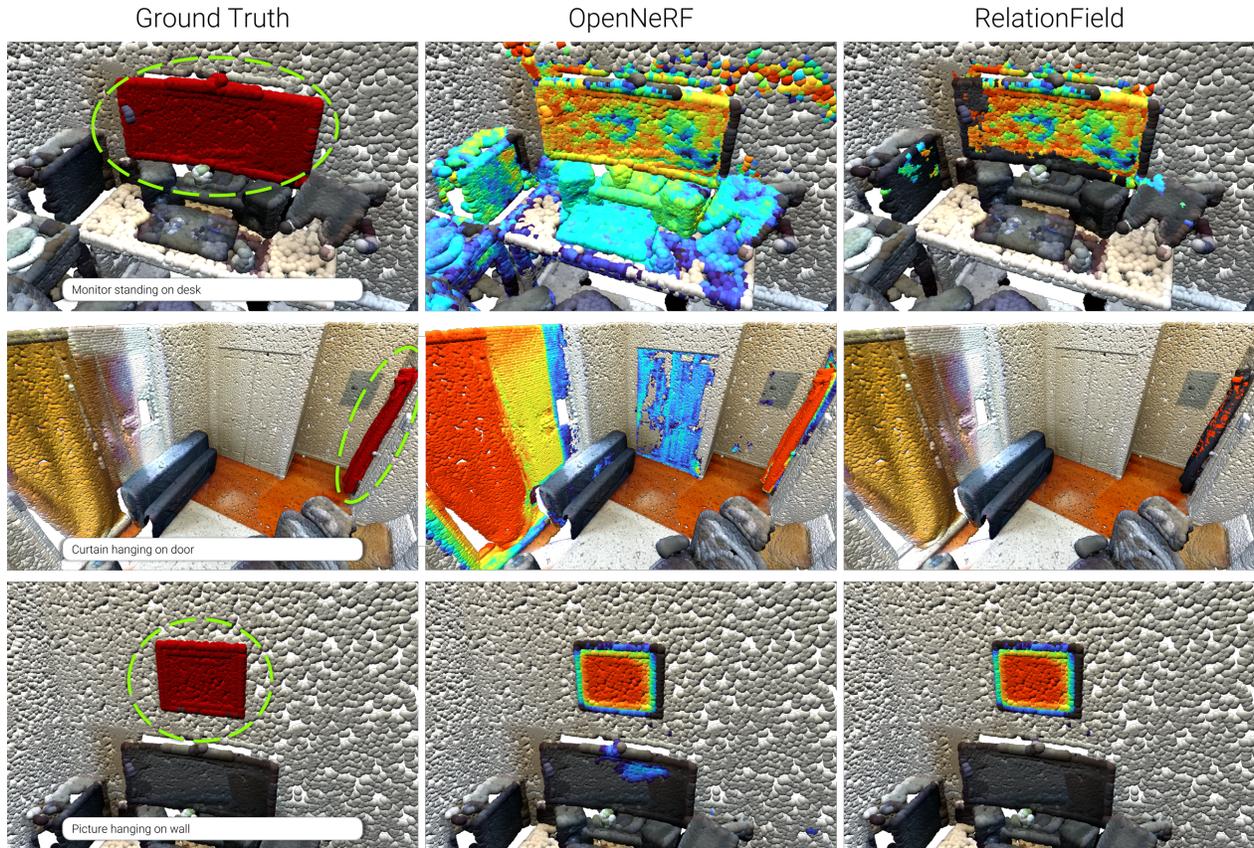


Figure 12. **Relationship-guided 3D Instance Segmentation.** We compare OpenNeRF with RelationField for relationship-guided 3D instance segmentation. While OpenNeRF produces many false positives because it gets confused with compositional queries arising from the *bag-of-words* behavior of CLIP. Meanwhile RelationField uses the object information together with the relationship information from the prompt to accurately filter predictions that only correspond to the subject in the prompt.

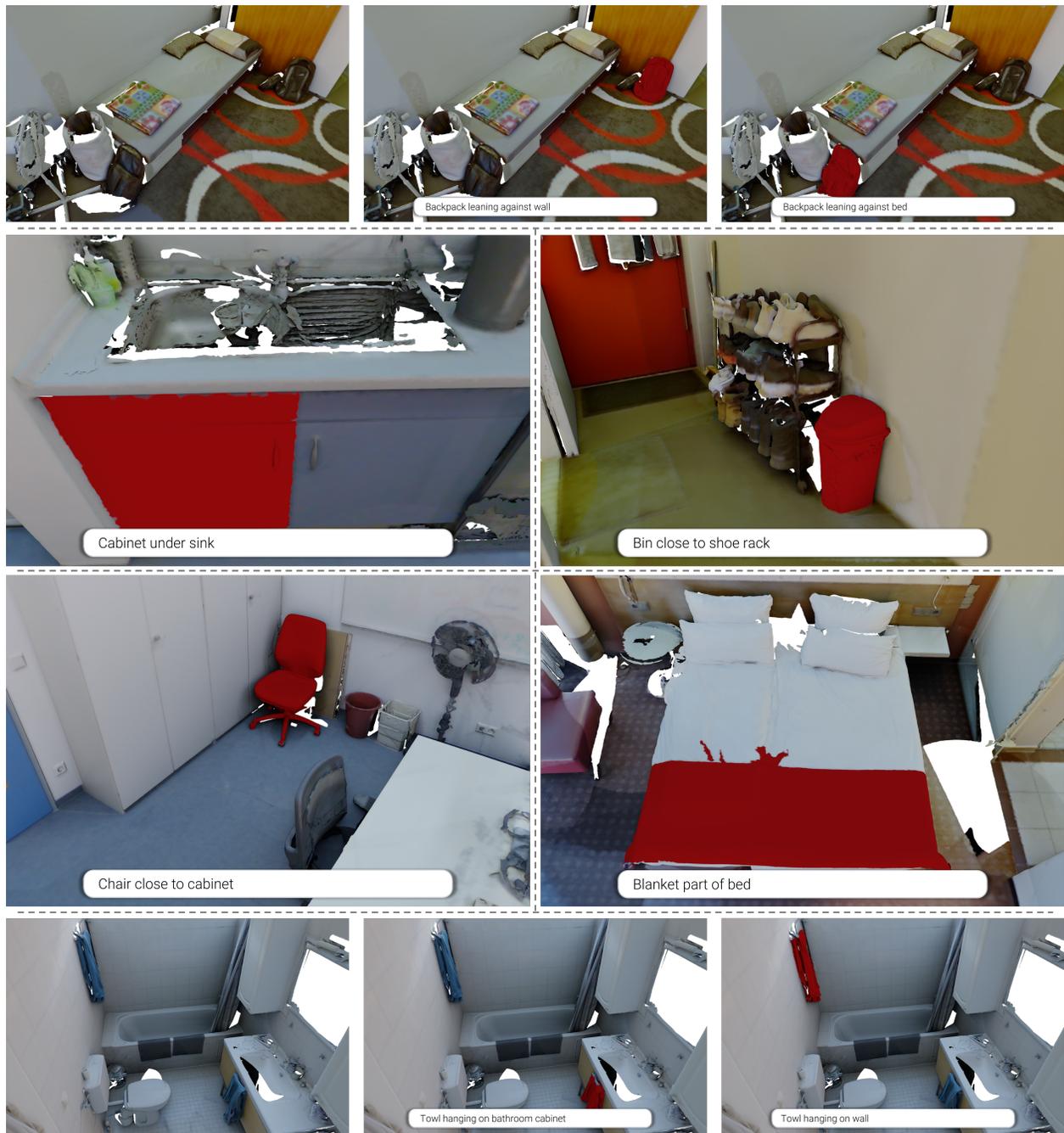


Figure 13. **Relation-guided 3D Instance Segmentation Task Overview.** We visualize a few annotated segments from our labeled benchmark on Scannet++ together with annotated relationship prompts. We focus on objects which appear multiple times in the scene, but that can be uniquely referenced by a relationship prompt.