This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Open-Vocabulary Functional 3D Scene Graphs for Real-World Indoor Spaces

Chenyangguang Zhang[1,2] Alexandros Delitzas[2,3] Fangjinhua Wang[2] Ruida Zhang[1]
Xiangyang Ji[1] Marc Pollefeys[2,4] Francis Engelmann[2,5]

[1]Tsinghua University [2]ETH Zürich [3]MPI for Informatics [4]Microsoft [5]Stanford University

Abstract

We introduce the task of predicting functional *3D scene graphs for real-world indoor environments from posed RGB-D images. Unlike traditional 3D scene graphs that focus on spatial relationships of objects, functional 3D scene graphs capture objects, interactive elements, and their functional relationships. Due to the lack of training data, we leverage foundation models, including visual language models (VLMs) and large language models (LLMs), to encode functional knowledge. We evaluate our approach on an extended SceneFun3D dataset and a newly collected dataset, FunGraph3D, both annotated with functional 3D scene graphs. Our method significantly outperforms adapted baselines, including Open3DSG and ConceptGraph, demonstrating its effectiveness in modeling complex scene functionalities. We also demonstrate downstream applications such as 3D question answering and robotic manipulation using functional 3D scene graphs. See our project page at https://openfungraph.github.io.*

1. Introduction

This paper introduces *functional* 3D scene graphs for real-world indoor spaces from posed RGB-D images. 3D scene graphs offer a lightweight, abstract representation for capturing the comprehensive semantic structure of an environment [4]. They support a variety of applications, including 3D scene alignment [66], image localization [51], graph-conditioned 3D scene generation [21, 97], as well as robotics navigation [83] and task planning [2, 61].

Recent advances in 3D scene graph prediction [4, 11, 27, 40, 41, 63, 64, 78, 84], have enabled exciting developments across multiple areas, including scene graph inference from 3D reconstructions [11, 78], applications for robotic interactions [27, 84], online scene graph generation [84], open-vocabulary 3D scene graphs [40, 41] and large-scale, hierarchical scene graphs [4, 63, 64]. The performance of recent scene graph methods also benefits from advancements in 3D scene understanding techniques [14, 57, 70], which they rely on to extract objects and their semantics for modeling inter-object relationships. However, existing 3D scene graph estimation methods [27, 40, 78, 84] face important limitations: graph nodes are typically restricted to *objects*, and edges represent only *spatial* relationships. For instance, edges primarily capture relative positions, such as *'the TV is mounted on the wall'* or *'the flower is placed on the table'*—information already implicitly encoded by object positions. Crucially, these methods lack representations of small interactive elements [17] and their *functional* relationships with other scene objects, which are essential for finer-grained interactions (*e.g.*, flipping a switch to turn on a light), making them less suitable for higher-level *functional reasoning*. The key idea of this paper is to enhance 3D scene graphs with the capability to represent *functional* relationships between objects and their interactive elements. A 3D scene graph that captures both functionalities and interactions opens up significant opportuni-

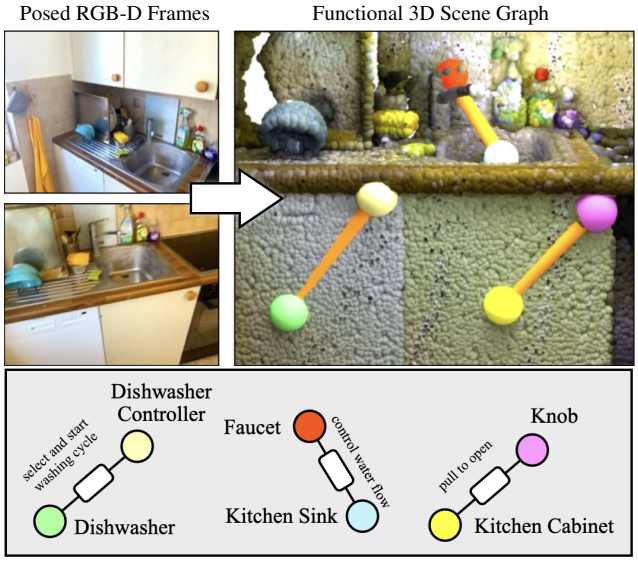


Fig. 1. **Functional 3D Scene Graphs.** Given an input sequence of posed RGB-D frames of an indoor environment, our method predicts a *functional* 3D scene graph by detecting objects, identifying interactive elements, and inferring functional relationships. This enables the representation of interactions, functions, and scene dynamics, going beyond existing 3D scene graph methods that are constrained to *spatial* relationships between static objects.

ties. For example, robotic agents can identify interactive elements and their functional relationships with objects to perform effective manipulation tasks, or graph-guided 3D scene generation methods [21, 97] can, with this enriched representation, generate more dynamic and realistic environments by incorporating interactive elements and their effects. However, creating functional 3D scene graphs is challenging. Most importantly, there is a lack of training data to learn the complex functional relationships between objects and their interactive elements. Unlike existing 3D scene graphs, functional 3D scene graphs require a more nuanced understanding of interactions and object affordances. To address this, our approach implements an open-vocabulary pipeline for functional 3D scene graph inference, termed *OpenFunGraph*, leveraging the extensive knowledge encoded within foundation models, including visual language models (VLM) and large language models (LLM). These models, pre-trained on vast amounts of multimodal data, include rich semantic information that can potentially be adapted for functional understanding. This leads us to the central question of this work: *"Can we harness foundation models to construct functional 3D scene graphs?"*

We evaluate our approach on two challenging datasets: an extended version of SceneFun3D [17] with newly added functional relationship annotations, and FunGraph3D, a freshly collected real-world dataset featuring high-precision 3D laser scans, accurately registered To address these limitations, we introduce functional 3D scene graphs, which model objects, interactive elements, and their functional relationships within a unified structure (formally defined in Section 3). This representation extends traditional 3D scene graphs by incorporating interactive sub-parts alongside objects and representing functional relationships beyond simple spatial ones. We argue that functional 3D scene graphs should possess the following characteristics. First, the representation should operate in an *open-vocabulary* manner to enhance generalization and applicability. Second, it should be *flexible*, allowing various attributes to be attached to nodes (*e.g.*, sensor data, natural language captions, semantic features) and edges (*e.g.*, relationship descriptions), thus ensuring adaptability for downstream applications.

In summary, our key contributions are:

- We introduce functional 3D scene graphs that extend traditional 3D scene graphs by capturing functional relationships between objects and interactive elements.
- We propose a novel approach that leverages the knowledge embedded in foundation models, specifically VLMs and LLMs, to construct functional 3D scene graphs without task-specific training.
- We present a new real-world dataset, FunGraph3D, with ground-truth functional annotations, and demonstrate that our method outperforms adapted baselines, including Open3DSG and ConceptGraph.

2. Related Work

3D indoor scene understanding. Many works concentrate on closed-set 3D semantic segmentation [5, 14, 31–33, 42, 45, 57, 58, 76, 80, 81] or instance segmentation [23, 28, 29, 38, 70, 74, 77, 96] on the existing 3D indoor scene understanding benchmarks [3, 7, 10, 15, 37, 65, 72, 93]. With the development of foundation models, subsequent researches explore open-vocabulary 3D semantic segmentation [24, 36, 39, 56, 59, 73, 75, 94, 105, 107], and complex 3D visual language grounding tasks [8, 16, 30, 34, 55, 62, 90, 103]. However, current studies mainly focus on object-level perception in indoor scene and seldom consider part-level interactive elements. Recently, SceneFun3D [17] proposes a benchmark for functionality and affordance understanding, with exhaustive annotations of indoor interactive elements. However, it does not provide the object annotations as well as the relationships between the elements and objects. This work extends SceneFun3D by exploiting such relationships with functional 3D scene graphs.

Affordance understanding. Understanding affordance, *i.e.*, properties of an environment to interact with, is a vital task in computer vision and robotics. Existing learning-based methods usually take inputs such as images [22, 98], videos [26, 54, 95] or 3D representations [18, 52, 53, 86], and then predict affordance maps. Some works learn affordance from human-scene interaction demonstrations [6, 12, 13, 25, 91, 92, 100, 101]. Nevertheless, existing works are often limited to object-level predictions and model affordances located on the corresponding objects. On the contrary, OpenFunGraph excavates all interactive elements at scene level, handling all kinds of functional relationships, especially those for remote operations.

3D scene graphs. 3D scene graph combines indoor entities into a unified structure and models inter-object relationships by building a graph of objects [4, 40, 63, 64, 75, 78, 79, 84, 85, 99, 102]. Functional 3D scene graph differs from the traditional 3D scene graph by adding interactive elements as nodes and modeling the functional relationships between objects and elements. Similarly, IFR-Explore [44] tries to excavate inter-object functional relationships based on reinforcement learning in synthetic scenarios. However, it is hard to be applied in complex real-world scenes due to its closed-set setting, requirement of ground-truth instances, and lack of consideration on part-level elements. In this paper, we propose an open-vocabulary framework for functional scene graph inference in complex real-world scenes. While there have been related efforts on open-vocabulary 3D scene graph generation, they are not well-suited for functional scene graph inference, particularly for interactive element recognition and functional relationship prediction. For example, Open3DSG [41] relies on object-level CLIP features [60]. It struggles with part-level interactive

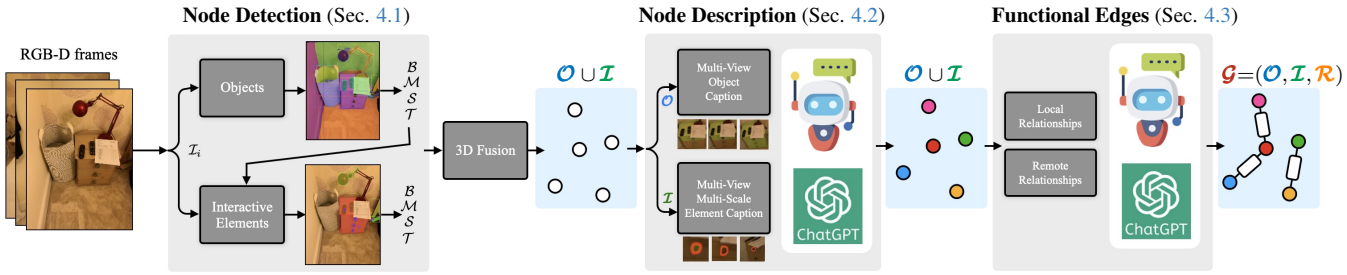


Fig. 2. **Illustration of the OpenFunGraph architecture.** Given a sequence of posed RGB-D frames $\{(\mathcal{I}_i, \mathcal{D}_i)\}_{i=1}^{n}$, we use RAM++ [104] and GroundingDINO [49] to detect and segment objects $\mathcal{O}$ and interactive elemens $\mathcal{I}$, forming the node candidates of the functional 3D scene graph. Next, a mechanism using the large language model (LLM) GPT [1] and the visual language model (VLM) LLAVA [48] generates natural language descriptions $\mathcal{L}$ for each node. Finally, we infer functional relationships $\mathcal{R}$ between objects $\mathcal{O}$ and interactive elements $\mathcal{I}$, represented as the edges in the functional 3D scene graph $\mathcal{G}$.

element recognition and is limited to inferring spatial relationships due to its design based on spatial-proximity edge feature distillation. ConceptGraph [27] uses a direct inference pipeline but focuses solely on object nodes and a narrow set of spatial relationships (*e.g.*, `on`, `in`). In contrast, our approach introduces adaptive detection and description stages for both objects and interactive elements, alongside a sequential reasoning strategy for accurately modeling a wide range of functional relationships.

3. Problem Formulation

Functional 3D Scene Graphs. We extend traditional 3D scene graphs [27, 41, 78] to facilitate their use in real-world scene interaction scenarios. Specifically, we introduce *Functional 3D Scene Graphs*, a representation designed to enable functional reasoning by jointly modeling *objects*, *interactive elements* and their *functional relationships*. We define a functional 3D scene graph as a directed graph $\mathcal{G} = (\mathcal{O}, \mathcal{I}, \mathcal{R})$ where $\mathcal{O}$ are the objects in the scene, $\mathcal{I}$ are the interactive elements and $\mathcal{R}$ are the functional relationships which point from the interactive element nodes $\mathcal{I}$ to object nodes $\mathcal{O}$. Following the definition in [17], we define interactive elements as components that agents interact with (*e.g.*, handles, knobs and buttons) to trigger specific functions within the environment such as opening a cabinet or turning off a light. Additionally, functional relationships fall into two categories: *local*, where the interactive element is part of the object (*e.g.*, door-handle), or *remote*, where the interactive element operates the object from a distance (*e.g.*, TV-remote control).

Task definition. We formulate the following novel 3D scene understanding task: Given an input sequence of posed RGB-D frames $\{(\mathcal{I}_i, \mathcal{D}_i)\}_{i=1}^{n}$ of an unseen indoor environment, the task is to construct the functional 3D scene graph $\mathcal{G}$ by inferring the functional relationships $\mathcal{R}$ among the objects $\mathcal{O}$ and interactive elements $\mathcal{I}$ in the scene.

4. Method

The goal of our method, *OpenFunGraph*, is to predict the functional 3D scene graph of a 3D environment, by accurately detecting objects and interactive elements, and inferring the functional relationships among them in an open-vocabulary manner (Figure 2). To overcome the challenge of limited training data, we harness the knowledge of foundation models [9] to detect objects and interactive elements within the scene, describe them in natural language, and reason about their functional relationships. In the detection stage (Section 4.1), we follow a progressive strategy where we prompt the foundation model to systematically first identify objects and then transition to finer-grained interactive elements given the input image sequence. The 2D detection results are then fused across multiple viewpoints in 3D space, constructing an initial set of node candidates. Next, we utilize a VLM and an LLM to collaboratively generate multi-view aware natural language descriptions of the candidate nodes (Section 4.2). To construct the graph, we proceed with inferring the functional relationships, *i.e.*, edges, among the object and interactive element nodes (Section 4.3). Specifically, we follow a sequential reasoning strategy, starting with local functional relationships (*e.g.*, door - handle) and extending to remote functional relationships (*e.g.*, TV – remote control), by leveraging the common sense knowledge of VLMs and LLMs. This allows us to progressively build the scene's functional graph by incrementally establishing connections between nodes.

4.1. Node Candidate Detection

In the first stage, we detect objects and interactive elements in the scene to construct a set of node candidates. We start by detecting 2D candidates on the input frames with a progressive foundation-model-based strategy that transitions from objects to finer-grained part-level interactive elements. Then, we associate and fuse the 2D detection results from multiple frames using geometric consistency, yielding the initial set of 3D node candidates.

Object candidates. To identify object candidates $\mathcal{C}_o^{\mathcal{I}_i}$, we utilize RAM++ [35, 104] to recognize objects in each input image $\mathcal{I}_i$, producing object tags $\mathcal{T}_{obj}^{\mathcal{I}i}$, such as 'cabinet' or 'door'. These object tags then serve as prompts for GroundingDINO [49], which detects 2D bounding boxes $\mathcal{B}^{\mathcal{I}_i}$, segmentation masks $\mathcal{M}^{\mathcal{I}_i}$, and confidence scores $\mathcal{S}^{\mathcal{I}_i}$.

Interactive element candidates. Despite the increasing success of foundation models in detecting object instances within scenes, the development of prompting strategies for identifying smaller elements, including interactive object parts (*e.g.*, knobs, handles), remains largely unexplored. Here, we propose a simple yet effective strategy to generate suitable text prompts for GroundingDINO to improve the detection of small interactive parts. We ask the LLM GPT-4 to provide a list of potential interactive element tags corresponding to each object candidate tag $\mathcal{T}_{obj}^{\mathcal{I}_i}$. We hold the valid object tags $\mathcal{T}_{val}^{\mathcal{I}_i}$ by filtering the cases where the LLM thinks the object is not interactable (*e.g.*, wall, bed). To create prompts for GroundingDINO, we concatenate $\mathcal{T}_{val}^{\mathcal{I}_i}$ (*e.g.*, door) as assistive tags with the functional element tags (*e.g.*, handle), forming prompts such as "door. handle". Finally, we yield the interactive element candidates $\mathcal{C}_{ie}^{\mathcal{I}_i}$ in each input image $\mathcal{I}_i$ by maintaining the detections corresponding to the functional element tags. Empirically, we observe that this approach leads to more accurate detection of small interactive parts. We support this observation with an ablation study in Section 6.3.

3D candidate fusion. After identifying the object and functional element candidates $\mathcal{C}_{obj}^{\mathcal{I}_i}$ and $\mathcal{C}_{ie}^{\mathcal{I}_i}$ in each image $\mathcal{I}_i$, we fuse their 2D segmentation masks using multi-view information to obtain the 3D node candidates of the graph. Following [27], we utilize the corresponding depth map $\mathcal{D}_i$ and camera projection matrix $\mathit{\Pi}_i$ to backproject the 2D mask to the 3D space, and merge them to receive the 3D object candidates $\mathcal{C}_o$ and interactive element candidates $\mathcal{C}_{ie}$. For each node candidate, we store the backprojected 3D point cloud $\mathcal{P}$ and 3D bounding box $\mathcal{B}$ along with the associated 2D image assets, *i.e.*, images, masks, 2D bounding boxes and confidence scores.

4.2. Node Candidate Description

We next outline the process of generating natural language descriptions $\mathcal{L}$ for each node by leveraging a combination of VLMs and LLMs. Precise language descriptions are critical for establishing functional relationships in the final phase.

Object candidates. To generate natural language descriptions for each object candidate node, we first select the top N_v views of each object, ranked by $\mathcal{S}^{\mathcal{I}_i} \times \frac{n_{\mathcal{P}^{\mathcal{I}_i}}}{n_{\mathcal{P}}}$, where $\mathcal{S}^{\mathcal{I}_i}$ is the 2D confidence score indicating the semantic confidence, while $n_{\mathcal{P}^{\mathcal{I}_i}}$ refers to the number of 3D points the view $\mathcal{I}_i$ contributes to the fused 3D pointcloud $\mathcal{P}$, presenting the geometric contribution of the view. Each object is then cropped based on its bounding box $\mathcal{B}$, and a caption describing the object crop is obtained using LLAVA v1.6 [46–48]. Finally, to derive a unified language description for each object candidate, we employ GPT-4 [1] to summarize the multi-view LLAVA captions.

Interactive element candidates. Captioning small interactive elements poses additional challenges: the bounding box crops are considerably smaller, often containing only a few pixels, which hinders LLAVA's ability to generate accurate captions. To address this, we enlarge the bounding boxes by multiple scales to incorporate richer contextual visual information. Similar multi-scale approaches have been shown to be effective in [39, 73]. To direct the VLM's attention to the interactive element within the expanded crop, we highlight the element with a red outline before passing it to LLAVA, as demonstrated in [71]. Finally, the multi-scale, multi-view captions are summarized into a single natural language description using GPT-4.

4.3. Functional Relationships

To model functional relationships between objects and interactive elements, we employ a sequential reasoning approach. Drawing on the concept of *Chain-of-Thought reasoning* [82], we decompose the task into a series of simpler steps rather than prompting the LLM to infer all possible element-object connections simultaneously. Initially, we concentrate on identifying direct, local relationships between objects and elements that are rigidly connected (*e.g.*, door – handle). Once these relationships are established, we extend the search to remote relationships, where object-element pairs are functionally related but physically separated (*e.g.*, TV – remote control).

Local relationship reasoning. First, we aim to construct the edges of the graph with local functional relationships, *e.g.*, the keypanel of a microwave or the knob of a cabinet. A common characteristic of these cases is that objects and interactive elements are rigidly connected. To identify such cases efficiently, we first perform a spatial filtering process: For each object node $\mathcal{C}_o^j$, we assess whether an element node $\mathcal{C}_{ie}^k$ has a significant spatial overlap. Subsequently, we leverage the LLM's common sense knowledge to reason whether a local functional relationship between these two nodes is feasible. To do this, we prompt the LLM with the language descriptions $\mathcal{L}^j$, $\mathcal{L}^k$ and 3D bounding boxes $\mathcal{B}^j$, $\mathcal{B}^k$ of $\mathcal{C}_o^j$ and $\mathcal{C}_{ie}^k$ respectively. It is tasked with reasoning whether a local rigid connection between the interactive element (*e.g.*, handle) and object (*e.g.*, fridge) is feasible, and then generate a language description $\mathcal{L}^{k\to j}$ of the functional relationship (*e.g.*, "opens"). This step produces the subgraph of local connections $\hat{\mathcal{G}}^L = \left(\mathcal{O}^L, \mathcal{I}^L, \mathcal{R}^L\right)$.

Confidence-aware remote relationship reasoning. In this step, we construct graph edges representing remote func-

tional relationships, such as those between a ceiling light and its switch. Determining these remote relationships is challenging, as visual cues alone often do not fully clarify which interactive element controls which specific object. To address this, we introduce a confidence-aware reasoning strategy that assigns a confidence score to each inferred remote relationship. This approach enhances decision-making in real-world scenarios by enabling the agent to prioritize interactions with higher confidence scores.

First, we form an initial set of potential candidates for remote connections, by considering the interactive element nodes that remained unassigned from the previous stage. To construct potential remote connections among the interactive elements and objects in the scene, we utilize the common sense knowledge of the LLM. Specifically, we provide the LLM with natural language descriptions $\mathcal{L}$ of the interactive element and object nodes, so that it can output a list of likely target objects that each interactive element could be functionally linked to. Next, for each element-object pair, we employ the VLM to assess the feasibility of a functional connection. The visual input for this step is prepared by the top-1 views of the interactive element and object. The VLM can exploit useful information in the images of the element and object to generate descriptions for the feasibility assessment. For example, it describes whether the appliance is physically plugged into the electric outlet, or whether the switch is mount on the wall under the ceiling light. The descriptions from all pairs are then provided to the LLM to form a global context, assisting it to assign a relative confidence score to each proposed connection and describe the nature of each relationship. This step outputs the subgraph of remote relations: $\hat{\mathcal{G}}^R = (\mathcal{O}^R, \mathcal{I}^R, \mathcal{R}^R)$.

4.4. Final Graph Formation

To construct the final graph, we combine the nodes and relationships identified in both the local and remote functional reasoning stages. The resulting predicted graph is formulated as $\hat{\mathcal{G}} = (\mathcal{O}^L \cup \mathcal{O}^R, \mathcal{I}^L \cup \mathcal{I}^R, \mathcal{R}^L \cup \mathcal{R}^R)$.

5. Data Collection

Existing datasets of high-fidelity 3D indoor spaces focus primarily on understanding either 3D objects [7, 93] or 3D interactive elements [17]. However, they lack ground-truth annotations of the functional relationships. In many cases, these relationships cannot be inferred from static visual observations alone but instead require video captures of physical interactions with the scene to determine which actions trigger specific responses. For example, a static 3D reconstruction cannot indicate which switch controls a particular light in a room with multiple switches and lights. To systematically evaluate our method, we construct a novel dataset of 3D real-world indoor environments along with multi-sensor data (*i.e.*, high-fidelity 3D reconstructions,

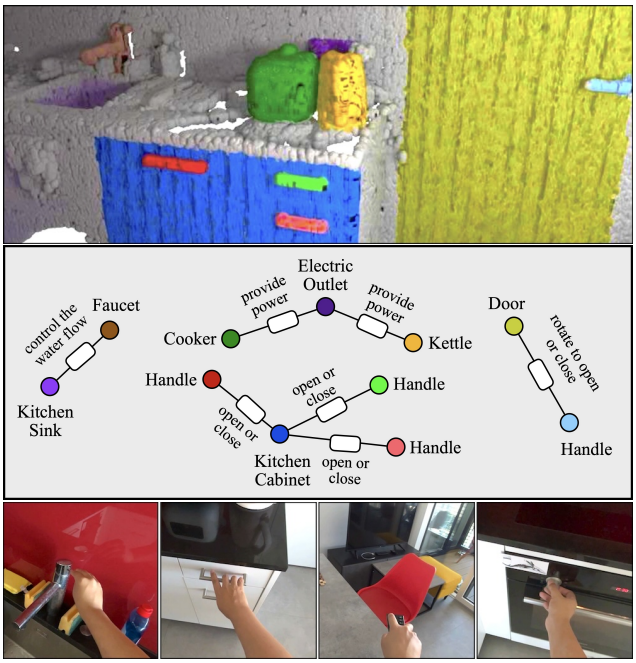


Fig. 3. **Modalities of our FunGraph3D dataset.** *Top:* 3D scans from a Faro laser scanner, annotated with 3D object and interactive element masks. *Middle:* Ground truth functional 3D scene graphs. *Bottom:* Egocentric video capturing human-scene interactions.

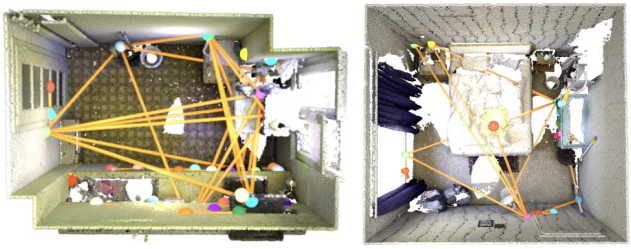


Fig. 4. **Example scenes from our FunGraph3D dataset.** The dataset includes typical indoor environments such as living rooms, bedrooms, bathrooms, and kitchens.

consumer-device video captures, egocentric human-scene interaction videos) and functional 3D scene graph annotations. We outline the steps towards building this dataset, which we refer to as *FunGraph3D* (Figure 4).

Laser scans. As illustrated in [17], we highlight that laser scans can capture a higher level of 3D geometry details, such as small interactive elements (*i.e.*, knobs, buttons), which is necessary for fine-grained scene understanding applications. To this end, we use a Leica RTC360 laser scanner to capture a high-resolution (5mm) 3D scan of the scene. To ensure high scene coverage during the capture, we place the scanner in multiple positions in the scene. We subsequently use the supporting software by Leica to fuse the multiple scans into a single one for the scene.

iPad video sequences. To enable scene understanding through multiple sensor data, we accompany the high-

fidelity 3D reconstruction with RGB-D image information from a commodity device. Specifically, we capture multiple videos of the static scene with the camera of an iPad 15 Pro.

Registration and alignment. To register the iPad video frames to the laser scan coordinate system, we build upon the COLMAP-based pipeline in [93]. Specifically, we run the COLMAP SfM pipeline [68, 69] by augmenting the collection of real iPad frames with rendered pseudo images of the laser scan. However, we notice that this pipeline leads to a large number of unregistered frames. To address this limitation, we incorporate the deep learning-based methods Superpoint [19] and Superglue [67] for feature extraction and matching, leading to a more accurate registration result. Afterwards, we utilize the optimized pose for each camera frame to render high-resolution depth maps for accurate back-projection from the iPad frames to the 3D space.

Egocentric videos. We include egocentric videos of property owners interacting with the environment using an Apple Vision Pro headset in our dataset. These videos facilitate accurate relationship labeling as they help clarify ambiguous connections among objects and interactive elements (*e.g.*, which light switch controls the ceiling light).

Annotation. For the annotation process, we extend the SceneFun3D annotation tool [17] to construct the ground-truth functional 3D scene graphs. Annotators can navigate the 3D scene and annotate the instances of objects and interactive elements along with a free-form label. Annotators are also asked to connect the interactive element to the corresponding object that it controls and provide a description of their relationship. An example of the collected annotations is displayed in Figure 3.

Statistics. FunGraph3D contains 14 in-the-wild scenes of various types (6 kitchens, 2 living rooms, 3 bedrooms and 3 bathrooms). In total, the dataset includes 201 interactive elements, 228 functional relationships and 146 objects of interest, along with open-vocabulary labels and relationships.

6. Experiments

6.1. Experimental Setup

Datasets. To evaluate our method, we utilize the developed FunGraph3D dataset, described in Section 5. Additionally, we use the SceneFun3D dataset [17], which provides high-resolution 5 mm laser scans of real-world environments along with iPad video sequences. Specifically, we randomly select 20 scenes (8 from the validation and 12 from the test split) and apply our annotation pipeline to annotate the functional 3D scene graph in each scene. Since we do not have physical access to the 3D environments, we restrict our evaluation to functional relationships that are visually unambiguous. In total, 212 interactive elements, 195 functional relationships, and 105 corresponding objects are annotated for these scenes.



Fig. 5. **Qualitative results.** *Top:* input images. *Bottom:* predicted functional 3D scene graph. Best seen zoomed in on a color screen.

Metrics. To evaluate open-vocabulary functional 3D scene graphs effectively, a new quantitative metric is essential. Existing approaches, such as ConceptGraph [27], rely on subjective human assessments, while Open3DSG [41] approaches evaluation as a label retrieval task, assuming all ground-truth nodes are known, an assumption that diverges from our real-world setting. To address this, we extend the Open3DSG Recall@K metric [41] with a node detection component, using spatial overlap between predicted and ground-truth nodes, inspired by evaluation techniques on 2D scene graph generation [50, 87–89, 106]. More specifically, our evaluation metric comprises two Recall@K scores: one for nodes, *i.e.*, $\mathcal{O}$ and $\mathcal{I}$, and one for triplets, *i.e.*, $(\mathcal{O},\mathcal{I},\mathcal{R})$. For node evaluation, we preprocess all ground-truth labels to enable top-K retrieval, following Open3DSG [41]. A retrieval is considered successful if a ground-truth node has a non-zero 3D IoU with a predicted node and the ground-truth label ranks within the top-K retrievals based on cosine similarity of CLIP embeddings [60] with the predicted label. We calculate overall node recall as $R_{no} = \frac{n^{re}_{no}}{n_{no}}$, where n^{re}_{no} is the number of successfully retrieved ground-truth nodes, and n_{no} is the total count of ground-truth nodes. Additionally, we assess recall for object and interactive element nodes separately, denoted as $R_o = \frac{n^{re}_{o}}{n_o}$ and $R_{ie} = \frac{n^{re}_{ie}}{n_{ie}}$, where n^{re}_{o} and n^{re}_{ie} are the counts of correctly retrieved objects and interactive elements and n_o and n_{ie} are their respective totals. For triplet $(\mathcal{O},\mathcal{I},\mathcal{R})$ evaluation, we apply stricter criteria: a ground-truth triplet is successfully retrieved in the top-K only when all its components $\mathcal{O}$, $\mathcal{I}$ and $\mathcal{R}$ are individually retrieved within the top-K. The retrieval process for $\mathcal{O}$ and $\mathcal{I}$ follows the same approach as above. To handle $\mathcal{R}$, we preprocess all relationship annotations by generating BERT embeddings [20], an approach effective for open-vocabulary predicates [41]. Successful retrieval is based on cosine similarity between ground-truth and predicted BERT embeddings. Triplet recall is defined as $R_{tr}=\frac{n_{re}}{n_{tr}}$, where n_{re} is

Methods	SceneFun3D [17]						FunGraph3D (Ours)					
	Objects		Inter. Elements		Overall Nodes		Objects		Inter. Elements		Overall Nodes	
	R@3	R@10	R@3	R@10	R@3	R@10	R@3	R@10	R@3	R@10	R@3	R@10
Open3DSG* [41]	61.2	70.7	54.4	61.8	56.7	64.7	50.9	58.1	21.8	33.9	33.4	43.6
Open3DSG*† [41]	42.9	50.0	33.8	38.3	37.4	43.0	30.9	44.1	13.0	19.6	20.2	29.4
ConceptGraph* [27]	71.3	77.1	6.6	8.6	28.3	31.4	58.0	66.3	2.5	4.1	20.1	25.2
ConceptGraph* [27] + IED	71.3	77.1	53.1	59.5	60.1	66.0	58.0	66.3	20.5	33.4	38.9	45.0
OpenFunGraph (Ours)	**81.8**	**87.8**	**71.0**	**79.5**	**73.0**	**82.8**	**70.7**	**79.1**	**44.4**	**57.6**	**55.5**	**65.8**

Tab. 1. **Node evaluation on the SceneFun3D [17] and FunGraph3D datasets.** * means to adapt the LLM prompts used for functional relationships inference. IED refers to the interactive element candidate detection in Section 4.1. † refers to the usage of the OpenFunGraph's fused 3D nodes rather than the ground-truth for fair comparison.

the count of retrieved triplets, and n_{tr} is the total count of ground-truth. We decompose triplet evaluation into node association ($R_{na}=\frac{n_{na}}{n_{tr}}$, with n_{na} being the number of triplets retrieved only considering $\mathcal{O}$,$\mathcal{I}$), indicating node recognition, and edge prediction ($R_{ep}=\frac{n_{re}}{n_{na}}$), showing relationship inference given correct node associations.

State-of-the-art comparisons. We compare our approach against ConceptGraph [27] and Open3DSG [41]-based baselines. Two ConceptGraph-based baselines are reimplemented: ConceptGraph* modifies the original LLM prompts to infer functional relationships, rather than focusing on spatial relationships such as `in` or `on`. ConceptGraph* + IED further incorporates the proposed interactive element candidate detection (IED) from Section 4.1, addressing ConceptGraph's initial limitation in detecting small parts. Both baselines use LLAVA v1.6 and GPT-4 for fair comparison with OpenFunGraph. We also reimplement two Open3DSG-based baselines. Open3DSG* modifies the LLM prompts to output functional relationships instead of spatial relationships. Since Open3DSG baselines rely on ground-truth node instance segmentation for graph neural network inference, we implement Open3DSG*†, which uses OpenFunGraph's fused 3D nodes for fair comparison. We report Recall@3 and Recall@10 for node metrics, and Recall@5 and Recall@10 for triplet metrics.

6.2. Results

Quantitative results are presented in Table 1 and 2. Overall, the FunGraph3D dataset poses a greater challenge than SceneFun3D [17] due to its more complex scenes, which contain a higher number of objects and interactive elements.

Node evaluation. As shown in Table 1, OpenFunGraph surpasses ConceptGraph* [27] by 160% on SceneFun3D and by 176% in R@3 on FunGraph3D. ConceptGraph* primarily focuses on object perception, resulting in poor recall scores for interactive elements. With the added interactive element candidate detection (IED), ConceptGraph* + IED improves node recognition, but still falls short of OpenFunGraph by 22% in R@3 on SceneFun3D, and 43% in R@3 on FunGraph3D, thanks to the specified node description stage proposed in OpenFunGraph. Our approach also outperforms Open3DSG-based baselines, achieving 95% and 29% higher scores than Open3DSG*† and Open3DSG* in R@3 on SceneFun3D, and 174% and 66% higher on FunGraph3D. The limited ability of Open3DSG-based methods to identify interactive elements arises from their focus on object-level features during training, whereas our approach employs a more practical open-vocabulary inference pipeline, free from these training constraints.

Triplet evaluation. Table 2 shows triplet prediction results. On SceneFun3D and FunGraph3D, benefiting from accurate node recognition and the sequential reasoning strategy for functional inference, OpenFunGraph outperforms ConceptGraph* + IED by 76% and 189% in R@5, and Open3DSG*† by 179% and 308%. Notably, Open3DSG-based baselines struggle with functional relationships, as they rely on spatial edge features from adjacent instances. ConceptGraph-based methods, which prompt the LLM to predict all possible connections, also perform worse when compared to our sequential reasoning strategy due to the increased interpretive complexity imposed on the LLM. Figure 5 visualizes qualitative results for OpenFunGraph. In the left scene, our confidence-aware remote relationship reasoning successfully infers that the light switch is more likely to control the ceiling light rather than the two table light bulbs. In the right scene, the local functional relationship between the handle and the door is accurately identified. Additionally, the fan is most confidently inferred to be powered by the nearby electric outlet.

6.3. Ablation studies

We ablate three key modules in our pipeline, *i.e.*, the GroundingDINO prompts for interactive element candidate detection, sequential reasoning, and confidence-aware remote relationship reasoning, presented in Table 3. The prompting strategy for GroundingDINO, which combines assistive object and element tags, proves effective. Using only element tags reduces node R@3 by 19% and 10%, as well as triplet R@5 by 20% and 22% on the two

Methods	SceneFun3D [17]						FunGraph3D (Ours)					
	Node Assoc.		Edge Pred.		Overall Triplets		Node Assoc.		Edge Pred.		Overall Triplets	
	R@5	R@10	R@5	R@10	R@5	R@10	R@5	R@10	R@5	R@10	R@5	R@10
Open3DSG* [41]	47.2	58.0	69.2	78.8	32.7	45.7	22.8	36.7	47.9	55.9	10.5	20.0
Open3DSG*† [41]	33.6	38.8	64.4	72.3	21.6	28.1	15.7	24.2	46.6	55.7	7.3	13.5
ConceptGraph* [27]	5.6	6.8	80.2	95.0	4.7	6.4	1.9	2.8	51.5	84.6	1.1	2.5
ConceptGraph* [27] + IED	45.4	49.3	75.6	90.9	34.3	44.5	18.8	22.8	46.1	79.7	10.3	18.9
OpenFunGraph (Ours)	**68.3**	**73.0**	**88.1**	**96.2**	**60.4**	**70.3**	**45.8**	**49.3**	**65.1**	**91.4**	**29.8**	**45.0**

Tab. 2. **Triplet evaluation on the SceneFun3D [17] and FunGraph3D datasets.** All marks keep the same meaning with Table 1. Node Assoc. refers to the node association metric while Edge Pred. means the edge prediction metric.



Fig. 6. **Functional 3D Scene Graphs for Robotic Manipulation.** *Left:* 3D scene and functional graph generated after querying 'turning on the light.' *Right:* Robot interacting with scene elements as guided by the functional scene graph.

datasets respectively, due to incomplete detections. Replacing sequential reasoning with a direct approach, where the LLM infers functional relationships across all nodes, significantly reduces triplet reasoning performance (42% and 32% in triplet R@5 on SceneFun3D and FunGraph3D respectively). Sequential reasoning decomposes complex relationships into distinct types, making LLM processing easier. Ablating confidence-aware remote relationship reasoning by randomly selecting connections, instead of using the highest-confident edge (*e.g.*, choosing a random light for the switch instead of the most confident ceiling light), leads to a decrease in triplet R@5 by 7% and 11% on the two datasets respectively. This illustrates more reasonable edges are selected correctly in our mechanism by incorporating the common sense understanding of the foundation models.

6.4. Downstream Applications

We showcase the versatility of the proposed functional 3D scene graph representation in downstream applications that require complex reasoning about indoor functionalities and task-oriented interactions.

3D inventory question answering. To enable functional reasoning, we convert the graph structure into a JSON list that the LLM can easily query. With this list, the LLM can answer questions such as "How can I turn on the ceiling light?". Using the functional 3D scene graph's nodes (objects, interactive elements) and edges (functional relationships), the LLM can provide responses such as "You can turn on the ceiling light using the light switch plate located at position [0.611, 0.113, 0.732]. From the provided JSON list, we can see the light switch plate with id 0 has the highest confidence level of 0.8 with the ceiling light fixture."

Experiments	Overall Nodes		Overall Triplets	
	R@3	R@10	R@5	R@10
w/o prompts for element detection	59.3	68.7	48.3	59.9
w/o sequential edge reasoning*	**73.0**	**82.8**	34.8	48.9
w/o confidence-aware edge reasoning*	**73.0**	**82.8**	56.0	65.1
Ours	**73.0**	**82.8**	**60.4 70.3**	
w/o prompts for element detection	49.9	59.1	23.1	37.6
w/o sequential edge reasoning*	**55.5**	**65.8**	20.2	33.8
w/o confidence-aware edge reasoning*	**55.5**	**65.8**	26.8	40.1
Ours	**55.5**	**65.8**	**29.8**	**45.0**

Tab. 3. **Ablation study** on SceneFun3D [17] (Top) and our FunGraph3D (Bottom). Note that edge reasoning (*) impacts only the triplet metric and does not affect node recognition performance.

Robotic manipulation. The functional 3D scene graph also supports robotic manipulation [43, 108] for user queries that involve functional reasoning, as illustrated in Figure 6. Similar to inventory question answering, the LLM queries the JSON list to locate the interactive element referenced in the query. The robot then navigates to and interacts with the element using the methods described in [43].

7. Conclusion

We introduce Functional 3D Scene Graphs, a novel representation that jointly models objects, interactive elements, and their functional relationships in 3D indoor environments. Our open-vocabulary pipeline leverages the common-sense knowledge of foundation models to infer functional 3D scene graphs and enable flexible querying. To support systematic benchmarking, we develop a high-fidelity dataset of real-world 3D indoor environments with multi-modal data and functional annotations. Experiments on this and existing datasets show that our method significantly outperforms baselines. We further demonstrate the versatility of our representation for downstream tasks such as 3D question answering and robotic manipulation.

Acknowledgments. We would like to thank colleagues and friends who helped us capture the data of FunGraph3D: Christine Engelmann, Dominik Faerber, Elisabetta Fedele, Xudong Jiang, Xin Kong, Aoxue Liu and Houssam Naous. This work was supported by the Swiss National Science Foundation Advanced Grant 216260: "Beyond Frozen Worlds: Capturing Functional 3D Digital Twins from the Real World". AD is supported by the Max Planck ETH Center for Learning Systems (CLS) and FE by an SNSF PostDoc.Mobility Fellowship.

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. 3, 4

[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 *Conference on Robot Learning (CoRL)*, 2022. 1

[3] Iro Armeni, Ozan Sener, Amir R Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3d semantic parsing of large-scale indoor spaces. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[4] 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 *International Conference on Computer Vision (ICCV)*, 2019. 1, 2

[5] Matan Atzmon, Haggai Maron, and Yaron Lipman. Point convolutional neural networks by extension operators. *ACM Transactions On Graphics (TOG)*, 2018. 2

[6] Prithviraj Banerjee, Sindi Shkodrani, Pierre Moulon, Shreyas Hampali, Fan Zhang, Jade Fountain, Edward Miller, Selen Basol, Richard Newcombe, Robert Wang, et al. Introducing hot3d: An egocentric dataset for 3d hand and object tracking. *arXiv preprint arXiv:2406.09598*, 2024. 2

[7] Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Tal Dimry, Yuri Feigin, Peter Fu, Thomas Gebauer, Brandon Joffe, Daniel Kurz, Arik Schwartz, et al. ARKitScenes: A diverse real-world dataset for 3d indoor scene understanding using mobile RGB-D data. In *International Conference on Neural Information Processing Systems (NeurIPS)*, 2021. 2, 5

[8] Valentin Bieri, Marco Zamboni, Nicolas S. Blumer, Qingxuan Chen, and Francis Engelmann. OpenCity3D: 3D Urban Scene Understanding with Vision-Language Models. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2025. 2

[9] 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

[10] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *International Conference on 3d Vision (3dV)*, 2017. 2

[11] 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 *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 1

[12] Zerui Chen, Yana Hasson, Cordelia Schmid, and Ivan Laptev. Alignsdf: Pose-aligned signed distance fields for hand-object reconstruction. In *European Conference on Computer Vision (ECCV)*, 2022. 2

[13] Woojin Cho, Jihyun Lee, Minjae Yi, Minje Kim, Taeyun Woo, Donghwan Kim, Taewook Ha, Hyokeun Lee, Je-Hwan Ryu, Woontack Woo, et al. Dense hand-object (ho) graspnet with full grasping taxonomy and dynamics. *European Conference on Computer Vision (ECCV)*, 2024. 2

[14] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 2

[15] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[16] Alexandros Delitzas, Maria Parelli, Nikolas Hars, Georgios Vlassis, Sotirios-Konstantinos Anagnostidis, Gregor Bachmann, and Thomas Hofmann. Multi-clip: Contrastive vision-language pre-training for question answering tasks in 3d scenes. In *British Machine Vision Conference (BMVC)*, 2023. 2

[17] Alexandros Delitzas, Ayca Takmaz, Federico Tombari, Robert Sumner, Marc Pollefeys, and Francis Engelmann. Scenefun3d: Fine-grained functionality and affordance understanding in 3d scenes. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 1, 2, 3, 5, 6, 7, 8

[18] Shengheng Deng, Xun Xu, Chaozheng Wu, Ke Chen, and Kui Jia. 3d affordancenet: A benchmark for visual object affordance understanding. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[19] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *International Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2018. 6

[20] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT*, 2019. 6

[21] Helisa Dhamo, Fabian Manhardt, Nassir Navab, and Federico Tombari. Graph-to-3d: End-to-end generation and

manipulation of 3d scenes using scene graphs. In *International Conference on Computer Vision (ICCV)*, 2021. 1, 2

[22] Thanh-Toan Do, Anh Nguyen, and Ian Reid. Affordancenet: An end-to-end deep learning approach for object affordance detection. In *International Conference on Robotics and Automation (ICRA)*, 2018. 2

[23] Francis Engelmann, Martin Bokeloh, Alireza Fathi, Bastian Leibe, and Matthias Nießner. 3d-mpa: Multi-proposal aggregation for 3d semantic instance segmentation. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[24] 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. *International Conference on Learning Representations (ICLR)*, 2024. 2

[25] Zicong Fan, Maria Parelli, Maria Eleni Kadoglou, Xu Chen, Muhammed Kocabas, Michael J Black, and Otmar Hilliges. Hold: Category-agnostic 3d reconstruction of interacting hands and objects from video. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2

[26] Kuan Fang, Te-Lin Wu, Daniel Yang, Silvio Savarese, and Joseph J Lim. Demo2vec: Reasoning object affordances from online videos. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[27] Qiao Gu, Ali Kuwajerwala, Sacha Morin, Krishna Murthy Jatavallabhula, Bipasha Sen, Aditya Agarwal, Corban Rivera, William Paul, Kirsty Ellis, Rama Chellappa, et al. ConceptGraphs: Open-vocabulary 3d scene graphs for perception and planning. In *International Conference on Robotics and Automation (ICRA)*, 2024. 1, 3, 4, 6, 7, 8

[28] Lei Han, Tian Zheng, Lan Xu, and Lu Fang. Occuseg: Occupancy-aware 3d instance segmentation. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[29] Ji Hou, Angela Dai, and Matthias Nießner. 3d-sis: 3d semantic instance segmentation of rgb-d scans. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[30] Joy Hsu, Jiayuan Mao, and Jiajun Wu. Ns3d: Neuro-symbolic grounding of 3d objects and relations. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[31] Zeyu Hu, Xuyang Bai, Jiaxiang Shang, Runze Zhang, Jiayu Dong, Xin Wang, Guangyuan Sun, Hongbo Fu, and Chiew-Lan Tai. Vmnet: Voxel-mesh network for geodesic-aware 3d semantic segmentation. In *International Conference on Computer Vision (ICCV)*, 2021. 2

[32] Binh-Son Hua, Minh-Khoi Tran, and Sai-Kit Yeung. Pointwise convolutional neural networks. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[33] Rui Huang, Songyou Peng, Ayca Takmaz, Federico Tombari, Marc Pollefeys, Shiji Song, Gao Huang, and Francis Engelmann. Segment3d: Learning fine-grained class-agnostic 3d segmentation without manual labels. *European Conference on Computer Vision (ECCV)*, 2024. 2

[34] Shijia Huang, Yilun Chen, Jiaya Jia, and Liwei Wang. Multi-view transformer for 3d visual grounding. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[35] Xinyu Huang, Yi-Jie Huang, Youcai Zhang, Weiwei Tian, Rui Feng, Yuejie Zhang, Yanchun Xie, Yaqian Li, and Lei Zhang. Open-set image tagging with multi-grained text supervision. *arXiv e-prints*, 2023. 4

[36] Krishna Murthy Jatavallabhula, Alihusein Kuwajerwala, Qiao Gu, Mohd Omama, Tao Chen, Alaa Maalouf, Shuang Li, Ganesh Iyer, Soroush Saryazdi, Nikhil Keetha, et al. Conceptfusion: Open-set multimodal 3d mapping. *ICRA2023 Workshop on Pretraining for Robotics (PT4R)*, 2023. 2

[37] Guangda Ji, Silvan Weder, Francis Engelmann, Marc Pollefeys, and Hermann Blum. Arkit labelmaker: A new scale for indoor 3d scene understanding. *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 2

[38] Li Jiang, Hengshuang Zhao, Shaoshuai Shi, Shu Liu, Chi-Wing Fu, and Jiaya Jia. Pointgroup: Dual-set point grouping for 3d instance segmentation. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[39] Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. Lerf: Language embedded radiance fields. In *International Conference on Computer Vision (ICCV)*, 2023. 2, 4

[40] Sebastian Koch, Pedro Hermosilla, Narunas Vaskevicius, Mirco Colosi, and Timo Ropinski. Lang3dsg: Language-based contrastive pre-training for 3d scene graph prediction. In *International Conference on 3d Vision (3dV)*, 2024. 1, 2

[41] 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 *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 1, 2, 3, 6, 7, 8

[42] Loic Landrieu and Martin Simonovsky. Large-scale point cloud semantic segmentation with superpoint graphs. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[43] Oliver Lemke, Zuria Bauer, René Zurbrügg, Marc Pollefeys, Francis Engelmann, and Hermann Blum. Spot-Compose: A framework for open-vocabulary object retrieval and drawer manipulation in point clouds. In *International Conference on Robotics and Automation (ICRA)*, 2024. 8

[44] Qi Li, Kaichun Mo, Yanchao Yang, Hang Zhao, and Leonidas Guibas. IFR-Explore: Learning inter-object functional relationships in 3d indoor scenes. *International Conference on Learning Representations (ICLR)*, 2022. 2

[45] Yangyan Li, Rui Bu, Mingchao Sun, Wei Wu, Xinhan Di, and Baoquan Chen. Pointcnn: Convolution on x-transformed points. *International Conference on Neural Information Processing Systems (NeurIPS)*, 2018. 2

[46] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *International Conference on Neural Information Processing Systems (NeurIPS)*, 2023. 4

[47] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[48] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, 2024. 3, 4

[49] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *European Conference on Computer Vision (ECCV)*, 2024. 3, 4

[50] Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Visual relationship detection with language priors. In *European Conference on Computer Vision (ECCV)*, 2016. 6

[51] Yang Miao, Francis Engelmann, Olga Vysotska, Federico Tombari, Marc Pollefeys, and Dániel Béla Baráth. SceneGraphLoc: Cross-modal coarse visual localization on 3d scene graphs. In *European Conference on Computer Vision (ECCV)*, 2024. 1

[52] Kaichun Mo, Yuzhe Qin, Fanbo Xiang, Hao Su, and Leonidas Guibas. O2o-afford: Annotation-free large-scale object-object affordance learning. In *Conference on Robot Learning (CoRL)*, 2022. 2

[53] Tushar Nagarajan and Kristen Grauman. Learning affordance landscapes for interaction exploration in 3d environments. *International Conference on Neural Information Processing Systems (NeurIPS)*, 2020. 2

[54] Tushar Nagarajan, Christoph Feichtenhofer, and Kristen Grauman. Grounded human-object interaction hotspots from video. In *International Conference on Computer Vision (ICCV)*, 2019. 2

[55] Maria Parelli, Alexandros Delitzas, Nikolas Hars, Georgios Vlassis, Sotirios Anagnostidis, Gregor Bachmann, and Thomas Hofmann. CLIP-Guided Vision-Language Pre-Training for Question Answering in 3D Scenes. In *International Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2023. 2

[56] Songyou Peng, Kyle Genova, Chiyu Jiang, Andrea Tagliasacchi, Marc Pollefeys, Thomas Funkhouser, et al. Openscene: 3d scene understanding with open vocabularies. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[57] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 2

[58] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *International Conference on Neural Information Processing Systems (NeurIPS)*, 2017. 2

[59] Minghan Qin, Wanhua Li, Jiawei Zhou, Haoqian Wang, and Hanspeter Pfister. Langsplat: 3d language gaussian splatting. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2

[60] 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 (ICML)*, 2021. 2, 6

[61] 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 robot task planning. In *Conference on Robot Learning (CoRL)*, 2023. 1

[62] Junha Roh, Karthik Desingh, Ali Farhadi, and Dieter Fox. Languagerefer: Spatial-language model for 3d visual grounding. In *Conference on Robot Learning (CoRL)*, 2022. 2

[63] Antoni Rosinol, Arjun Gupta, Marcus Abate, Jingnan Shi, and Luca Carlone. 3d dynamic scene graphs: Actionable spatial perception with places, objects, and humans. *Robotics, Science and Systems*, 2020. 1, 2

[64] 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. *International Journal on Robotics Research (IJRR)*, 2021. 1, 2

[65] David Rozenberszki, Or Litany, and Angela Dai. Language-grounded indoor 3d semantic segmentation in the wild. In *European Conference on Computer Vision (ECCV)*, 2022. 2

[66] Sayan Deb Sarkar, Ondrej Miksik, Marc Pollefeys, Daniel Barath, and Iro Armeni. SGAligner: 3d scene alignment with scene graphs. In *International Conference on Computer Vision (ICCV)*, 2023. 1

[67] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. SuperGlue: Learning feature matching with graph neural networks. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 6

[68] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-Motion Revisited. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 6

[69] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016. 6

[70] 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

[71] Aleksandar Shtedritski, Christian Rupprecht, and Andrea Vedaldi. What does Clip Know About a Red Circle? Visual Prompt Engineering for VLMs. In *International Conference on Computer Vision (ICCV)*, 2023. 4

[72] Tao Sun, Yan Hao, Shengyu Huang, Silvio Savarese, Konrad Schindler, Marc Pollefeys, and Iro Armeni. Nothing Stands Still: A Spatiotemporal Benchmark on 3D Point Cloud Registration Under Large Geometric and Temporal Change. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2025. 2

[73] Ayça Takmaz, Elisabetta Fedele, Robert W Sumner, Marc Pollefeys, Federico Tombari, and Francis Engelmann. Openmask3d: Open-vocabulary 3d instance segmentation. *International Conference on Neural Information Processing Systems (NeurIPS)*, 2023. 2, 4

[74] Ayça Takmaz, Jonas Schult, Irem Kaftan, Mertcan Akçay, Bastian Leibe, Robert Sumner, Francis Engelmann, and Siyu Tang. 3D Segmentation of Humans in Point Clouds with Synthetic Data. In *International Conference on Computer Vision (ICCV)*, 2023. 2

[75] Ayca Takmaz, Alexandros Delitzas, Robert W. Sumner, Francis Engelmann, Johanna Wald, and Federico Tombari. Search3D: Hierarchical Open-Vocabulary 3D Segmentation. *IEEE Robotics and Automation Letters (RA-L)*, 2025. 2

[76] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *International Conference on Computer Vision (ICCV)*, 2019. 2

[77] Thang Vu, Kookhoi Kim, Tung M Luu, Thanh Nguyen, and Chang D Yoo. Softgroup for 3d instance segmentation on point clouds. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[78] Johanna Wald, Helisa Dhamo, Nassir Navab, and Federico Tombari. Learning 3d semantic scene graphs from 3d indoor reconstructions. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1, 2, 3

[79] Ziqin Wang, Bowen Cheng, Lichen Zhao, Dong Xu, Yang Tang, and Lu Sheng. Vl-sat: Visual-linguistic semantics assisted training for 3d semantic scene graph prediction in point cloud. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[80] Silvan Weder, Francis Engelmann, Johannes L Schönberger, Akihito Seki, Marc Pollefeys, and Martin R Oswald. Alster: A Local Spatio-temporal Expert for Online 3D Semantic Reconstruction. *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2023. 2

[81] Silvan Weder, Hermann Blum, Francis Engelmann, and Marc Pollefeys. Labelmaker: Automatic semantic label generation from rgb-d trajectories. In *International Conference on 3d Vision (3dV)*, 2024. 2

[82] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *International Conference on Neural Information Processing Systems (NeurIPS)*, 2022. 4

[83] Abdelrhman Werby, Chenguang Huang, Martin Büchner, Abhinav Valada, and Wolfram Burgard. Hierarchical open-vocabulary 3d scene graphs for language-grounded robot navigation. In *First Workshop on Vision-Language Models for Navigation and Manipulation at ICRA 2024*, 2024. 1

[84] Shun-Cheng Wu, Johanna Wald, Keisuke Tateno, Nassir Navab, and Federico Tombari. SceneGraphFusion: Incremental 3d scene graph prediction from rgb-d sequences. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 1, 2

[85] Shun-Cheng Wu, Keisuke Tateno, Nassir Navab, and Federico Tombari. Incremental 3d semantic scene graph prediction from rgb sequences. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[86] Chao Xu, Yixin Chen, He Wang, Song-Chun Zhu, Yixin Zhu, and Siyuan Huang. Partafford: Part-level affordance discovery from 3d objects. *European Conference on Computer Vision (ECCV) Workshops*, 2022. 2

[87] Danfei Xu, Yuke Zhu, Christopher B Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 6

[88] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. In *European Conference on Computer Vision (ECCV)*, 2018.

[89] Jingkang Yang, Yi Zhe Ang, Zujin Guo, Kaiyang Zhou, Wayne Zhang, and Ziwei Liu. Panoptic scene graph generation. In *European Conference on Computer Vision (ECCV)*, 2022. 6

[90] Zhengyuan Yang, Songyang Zhang, Liwei Wang, and Jiebo Luo. Sat: 2d semantics assisted training for 3d visual grounding. In *International Conference on Computer Vision (ICCV)*, 2021. 2

[91] Yufei Ye, Abhinav Gupta, and Shubham Tulsiani. What's in your hands? 3d reconstruction of generic objects in hands. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[92] Yufei Ye, Abhinav Gupta, Kris Kitani, and Shubham Tulsiani. G-hop: Generative hand-object prior for interaction reconstruction and grasp synthesis. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2

[93] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *International Conference on Computer Vision (ICCV)*, 2023. 2, 5, 6

[94] Gonca Yilmaz, Songyou Peng, Marc Pollefeys, Francis Engelmann, and Hermann Blum. OpenDAS: Open-Vocabulary Domain Adaptation for 2D and 3D Segmentation. *arXiv preprint arXiv:2405.20141*, 2024. 2

[95] Tomoya Yoshida, Shuhei Kurita, Taichi Nishimura, and Shinsuke Mori. Text-driven affordance learning from egocentric vision. *arXiv preprint arXiv:2404.02523*, 2024. 2

[96] Yuanwen Yue, Sabarinath Mahadevan, Jonas Schult, Francis Engelmann, Bastian Leibe, Konrad Schindler, and Theodora Kontogianni. Agile3d: Attention guided interactive multi-object 3d segmentation. *International Conference on Learning Representations (ICLR)*, 2024. 2

[97] Guangyao Zhai, Evin Pınar Örnek, Shun-Cheng Wu, Yan Di, Federico Tombari, Nassir Navab, and Benjamin

Busam. Commonscenes: Generating commonsense 3d indoor scenes with scene graphs. *International Conference on Neural Information Processing Systems (NeurIPS)*, 2023. 1, 2

[98] Wei Zhai, Hongchen Luo, Jing Zhang, Yang Cao, and Dacheng Tao. One-shot object affordance detection in the wild. *International Journal on Computer Vision (IJCV)*, 2022. 2

[99] Chaoyi Zhang, Jianhui Yu, Yang Song, and Weidong Cai. Exploiting edge-oriented reasoning for 3d point-based scene graph analysis. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[100] Chenyangguang Zhang, Yan Di, Ruida Zhang, Guangyao Zhai, Fabian Manhardt, Federico Tombari, and Xiangyang Ji. Ddf-ho: Hand-held object reconstruction via conditional directed distance field. *International Conference on Neural Information Processing Systems (NeurIPS)*, 2023. 2

[101] Chenyangguang Zhang, Guanlong Jiao, Yan Di, Gu Wang, Ziqin Huang, Ruida Zhang, Fabian Manhardt, Bowen Fu, Federico Tombari, and Xiangyang Ji. Moho: Learning single-view hand-held object reconstruction with multi-view occlusion-aware supervision. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2

[102] Shoulong Zhang, Aimin Hao, Hong Qin, et al. Knowledge-inspired 3d scene graph prediction in point cloud. *International Conference on Neural Information Processing Systems (NeurIPS)*, 2021. 2

[103] Yiming Zhang, ZeMing Gong, and Angel X Chang. Multi3drefer: Grounding text description to multiple 3d objects. In *International Conference on Computer Vision (ICCV)*, 2023. 2

[104] Youcai Zhang, Xinyu Huang, Jinyu Ma, Zhaoyang Li, Zhaochuan Luo, Yanchun Xie, Yuzhuo Qin, Tong Luo, Yaqian Li, Shilong Liu, et al. Recognize anything: A strong image tagging model. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 3, 4

[105] Shijie Zhou, Haoran Chang, Sicheng Jiang, Zhiwen Fan, Zehao Zhu, Dejia Xu, Pradyumna Chari, Suya You, Zhangyang Wang, and Achuta Kadambi. Feature 3dgs: Supercharging 3d gaussian splatting to enable distilled feature fields. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2

[106] Zijian Zhou, Zheng Zhu, Holger Caesar, and Miaojing Shi. Openpsg: Open-set panoptic scene graph generation via large multimodal models. *European Conference on Computer Vision (ECCV)*, 2024. 6

[107] Xingxing Zuo, Pouya Samangouei, Yunwen Zhou, Yan Di, and Mingyang Li. Fmgs: Foundation model embedded 3d gaussian splatting for holistic 3d scene understanding. *International Journal on Computer Vision (IJCV)*, 2024. 2

[108] René Zurbrügg, Yifan Liu, Francis Engelmann, Suryansh Kumar, Marco Hutter, Vaishakh Patil, and Fisher Yu. ICGNet: A Unified Approach for Instance-centric Grasping. In *International Conference on Robotics and Automation (ICRA)*, 2024. 8