Open Peer Review on Qeios

RESEARCH ARTICLE

# g3D-LF: Generalizable 3D-Language Feature Fields for Embodied Tasks

Zihan Wang[1], Gim Hee Lee[1]

1 School of Computing, National University of Singapore, Singapore

Funding: No specific funding was received for this work.

Potential competing interests: No potential competing interests to declare.

## Abstract

We introduce Generalizable 3D-Language Feature Fields (g3D-LF), a 3D representation model pre-trained on large-scale 3D-language dataset for embodied tasks. Our g3D-LF processes posed RGB-D images from agents to encode feature fields for: 1) Novel view representation predictions from any position in the 3D scene; 2) Generations of BEV maps centered on the agent; 3) Querying targets using multi-granularity language within the above-mentioned representations.Our representation can be generalized to unseen environments, enabling real-time construction and dynamic updates. By volume rendering latent features along sampled rays and integrating semantic and spatial relationships through multiscale encoders, our g3D-LF produces representations at different scales and perspectives, aligned with multi-granularity language, via multi-level contrastive learning. Furthermore, we prepare a large-scale 3D-language dataset to align the representations of the feature fields with language. Extensive experiments on Vision-and-Language Navigation under both Panorama and Monocular settings, Zero-shot Object Navigation, and Situated Question Answering tasks highlight the significant advantages and effectiveness of our g3D-LF for embodied tasks. The code is available at https://github.com/MrZihan/g3D-LF.

## 1. Introduction

Embodied agents seek to understand 3D environments, enabling interaction with environments and human by performing tasks such as Question Answering[1][2][3], Navigation[4][5][6][7][8][9], etc. To this end, various 3D scene representation models tailored for embodied tasks have been proposed, including point cloud-based models[10][11][12], 3D occupancy[13], hybrid voxel[14], and feature fields[15][16][17][18].

For multimodal embodied tasks in large-scale scenes, 3D representation models typically need: 1) generalization to unseen scenes, 2) construct and update representations in real time, and 3) open-vocabulary semantic space. The generalizable 3D feature fields provides the above advantages and has been widely explored across various embodied tasks. Unlike point cloud-based models that depend on complete and low-noise point clouds which are less robust, the implicit representations of the feature fields are derived from the 2D foundation model, preserving semantic expressiveness even with few-shot observations from 3D scenes. As shown in Figure 1, the feature fields model uses RGB-D images as input to encode and update implicit scene representations, which are then used to predict novel view, panorama and BEV map representations associated with language through volume rendering. These predicted representations can assist embodied tasks such as navigation planning[17][19][18], etc.

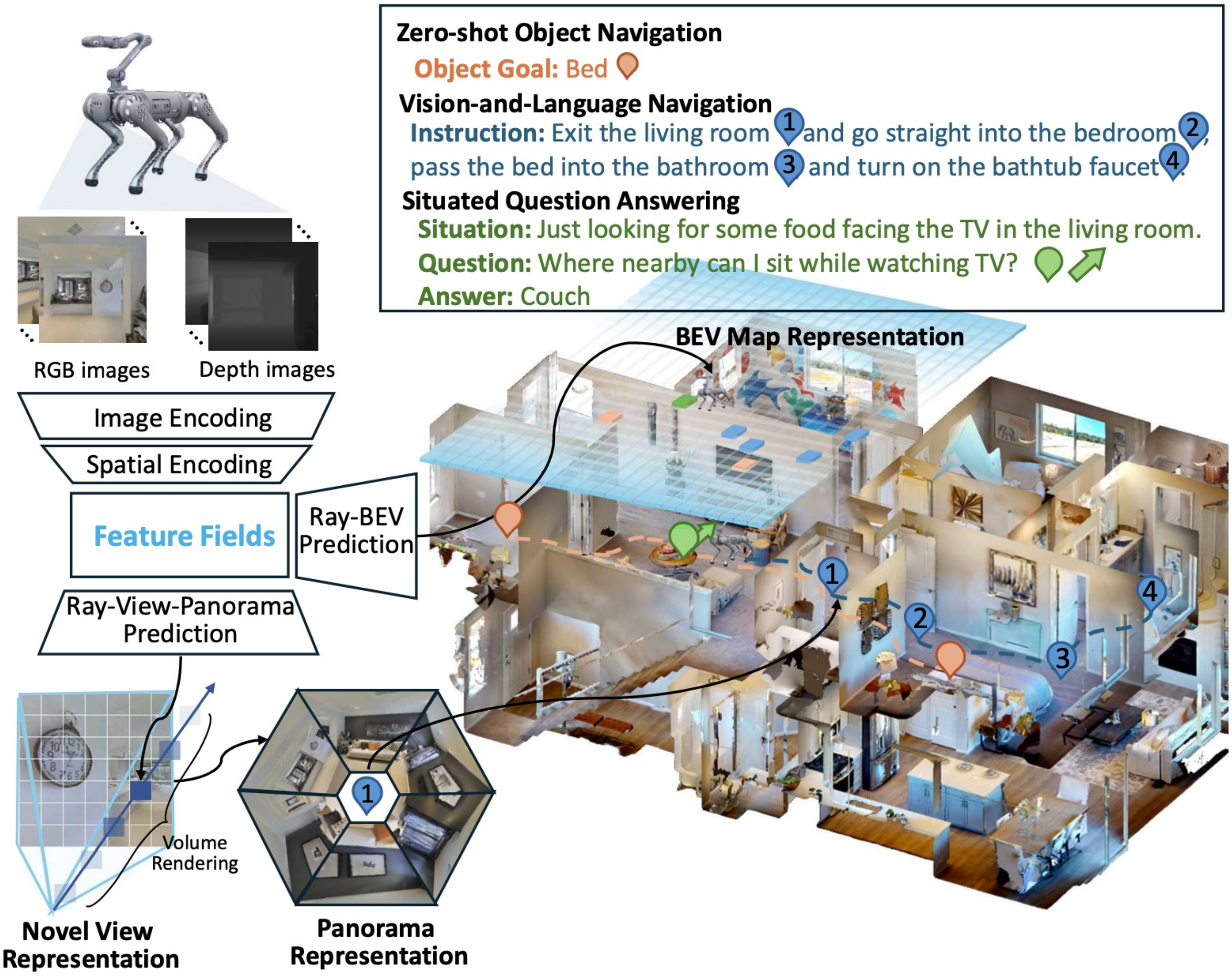

Zero-shot Object Navigation
Object Goal: Bed
Vision-and-Language Navigation
Instruction: Exit the living room 1 and go straight into the bedroom 2, pass the bed into the bathroom 3 and turn on the bathtub faucet 4.
Situated Question Answering
Situation: Just looking for some food facing the TV in the living room.
Question: Where nearby can I sit while watching TV?
Answer: Couch
RGB images
Depth images
Image Encoding
Spatial Encoding
Feature Fields
Ray-BEV Prediction
Ray-View-Panorama Prediction
BEV Map Representation
Volume Rendering
Novel View Representation
Panorama Representation
1
2
3
4

**Figure 1.** Our g3D-LF uses posed RGB-D images from the agent to predict novel view and BEV map representations at various scales within the 3D scene, aligned with multi-granularity language through 3D-language pre-training. The representation is applicable to embodied tasks like visual navigation and embodied question answering, facilitating scene representation, language-guided querying, and navigation planning.

However, several significant drawbacks remain in these feature fields models: 1) The supervision for the predicted representations comes from 2D foundation models, e.g., CLIP[20] and DINOv2[21] greatly limits the understanding for 3D spatial relationships; 2) These models are trained without language supervision, resulting in a substantial gap with language semantics; 3) The large-scale representations, e.g., panorama and BEV map from feature fields is particularly challenging for long text understanding. These issues severely limit the potential of the feature fields model on language-guided embodied tasks.

To circumvent the above-mentioned issues, we introduce Generalizable 3D-Language Feature Fields (g3D-LF), a 3D representation model pre-trained on large-scale 3D-language dataset for embodied tasks. We first curate and consolidate a large amount of 3D-language data from previous works[22][23][24] to train our g3D-LF model. These data include 5K indoor scenes and almost 1M language descriptions of multiple granularities. The text annotations include object categories, object characteristics, object relationships, and the spatial layout of the entire scene, which are employed to supervise multiscale encoders of the g3D-LF model. We then design our g3D-LF model to learn generalizable 3D-language feature fields. To this end, we employ multi-level contrastive learning for multi-scale encoders to align predicted representations and language across different scales. For the regional representation within the novel view, a contrastive loss is calculated across 1,883 indoor object categories. For the predicted novel view representation, both the CLIP visual representations and language are employed for contrastive training to balance generalization ability and language alignment. For large-scale panorama and BEV representations, we propose the fine-grained contrastive learning based on the affinity matrix to achieve long text understanding.

The pre-trained g3D-LF model is subsequently evaluated on various embodied tasks, including vision-and-language navigation (monocular setting[19] and panorama setting[17]), zero-shot object navigation[6], and situated question answering[1], gains significant performance improvements. In this work, our**main contributions** include:

- We organize a large-scale 3D-language dataset to train the feature fields model.
- This work proposes the Generalizable 3D-Language Feature Fields (g3D-LF) with a multi-level contrastive learning framework to align the multi-scale representations of feature fields with multi-granularity language.
- Our proposed g3D-LF model improves multiple baseline methods to state-of-the-art performance across various embodied tasks, thus validating the potential of our generalizable feature fields for Embodied AI.

## 2. Related Work

Generalizable 3D Feature Fields.

The neural radiance field (NeRF)[25] has gained significant popularity in various AI tasks, which predicts the RGB image from an arbitrary viewpoint in a 3D scene. Furthermore, some works leverage NeRF-based methods to predict novel view representations instead of RGB values, enabling 3D semantic segmentation[26] and 3D language grounding[27]. However, these methods with implicit MLP networks can only synthesize novel view representations in seen scenes, which makes it difficult to generalize to unseen large-scale scenes and adapt to many embodied AI tasks (e.g., navigation). To this end, some works[17][18][28] attempt to encode 2D visual observations into 3D representations (called Generalizable 3D Feature Fields) via the depth map. Through volume rendering[25], these models decode novel view representations from the feature fields and align them with open-world features (e.g., CLIP embeddings[20]). The 3D feature fields can generalize to unseen scenes, enabling real-time construction and dynamic updates. However, the drawback of these models lies in the fact that the supervision of their predicted representations comes from 2D visual models, which limits their performance in language-guided embodied tasks. Our work offers a feasible approach to training the 3D feature fields model with large-scale 3D-language data.

Vision-and-Language Navigation.

Vision-and-Language Navigation (VLN) [7][8][29][30][31][32][33] requires the agent understand complex natural language instructions and navigate to the described destination using low-level actions, e.g., turn left 15 degrees, turn right 15 degrees, or move forward 0.25 meters. To address inefficiencies and poor performance in atomic action prediction, some works[34][35][19] develop waypoint predictors to generate several candidate waypoints around the agent. The navigation policy model can then select the optimal waypoint as the next sub-goal and execute atomic actions to move, greatly enhancing planning efficiency. In this context, how to represent waypoints and carry out planning have become critical. Some works use a topological map[36][37] or BEV map[38][39][40] to represent semantic relationships between waypoints, while some[17][19] explore feature fields to predict waypoint representations of novel views and improve navigation planning. Our g3D-LF model further improves the performance of methods using feature fields.

Zero-shot Object Navigation.

In object-goal navigation[4][41][42], an agent is tasked with locating a specified object within indoor environments. Typically, reinforcement learning[43] is used to train a policy network that predicts actions, while object detection[44][45] or segmentation models[46][47][48] help identify the object. However, these navigation models are often limited to specific objects, making open-vocabulary navigation challenging and hindering generalization in real-world applications[49]. To address this issue, zero-shot navigation methods have emerged[5][50][51][6], leveraging Vision-and-Language Models (VLMs)[20][52][53] to identify potential directions or areas containing the target, followed by using the pre-trained pointgoal navigation models[54] to search the potential areas. Considering that general 2D VLMs are not fully suited for indoor 3D environments and to the best of our knowledge, we are the first to attempt using the indoor 3D feature fields model for zero-shot object navigation.

Situated Question Answering.

The Embodied Question Answering tasks[2][55][3] require the agent to observe the 3D environment and answer questions from humans. Furthermore, Situated Question Answering[1] requires advanced 3D spatial understanding of the agent to answer the question and to interpret and locate the position and orientation of the textual description. Compared to previous works[11][22][14] using point clouds, we only use RGB-D images to encode feature fields and leverage their multi-scale representations for localization and question answering.

## 3. Our Method

### 3.1. 3D-Language Data

We prepare a large-scale 3D-language dataset to align the representations of the feature fields with language. Our dataset includes about 5K 3D indoor scenes, mainly sourced from the single-room scans ScanNet[56], multi-room house scans of the Habitat-Matterport 3D dataset (HM3D)[57][58], and the photo-realistic multi-room scenes of Structured3D[59]. The total number of language annotations is close to one million, which are mainly sourced from the SceneVerse dataset[22]. SceneVerse uses 3D scene graphs and large language models (LLMs) to automate high-quality object-level and scene-level descriptions. The annotations also includes the large set of human-annotated object referrals[23].

We organize the dataset as follows to streamline feature fields training: 1) For each 3D scene, the agent can observe numerous RGB-D images and its corresponding poses as inputs. 2) An instance-level point clouds mark each instance in the scene with an instance ID which can be used to retrieve associated language descriptions from the database. It is thuseasy to get instances that are near any given point in the 3D scene and obtain their language descriptions. This enables thetraining code to efficiently obtain language annotations for specific regions within a novel view or a BEV map.

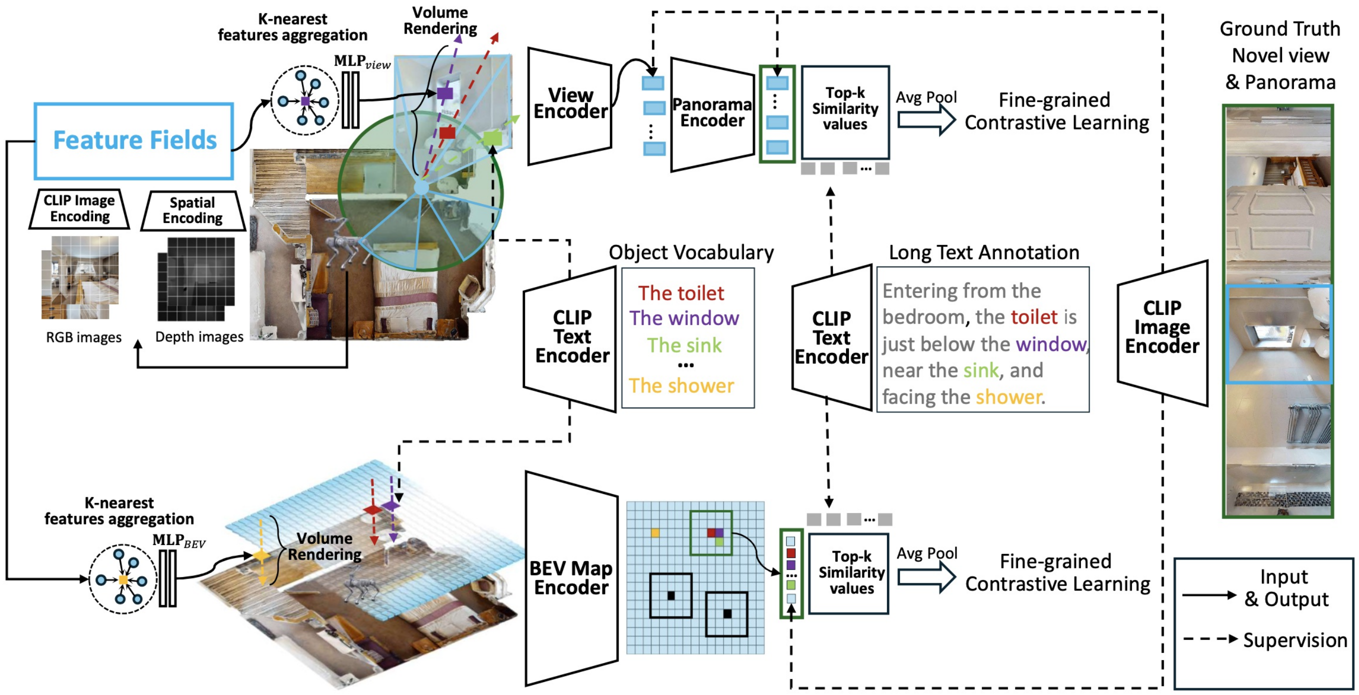


**Figure 2. Overview of our g3D-LF model.** Our model encodes the observed RGB-D images into the feature fields (consists of many feature points). Through aggregating k-nearest features, the MLP networks predict the latent feature and volume density of sampled points along the rendered ray. The hierarchical encoders further generate representations of novel view, panorama, and BEV map, then conduct multi-level contrastive learning with multi-granularity language.

## 3.2. 3D-Language Feature Fields

### Feature Fields Encoding.

As shown in Figure 2, our g3D-LF model follows HNR[17] to take a posed RGB image as input and uses the CLIP image encoder to extract fine-grained visual features $\{\mathbf{g}_{t,i} \in R^{768}\}_{i=1}^{I}$. $\mathbf{g}_{t,i}$ denotes the $i$-th feature patch of the CLIP feature map extracted from $t$-th frame observed by the agent. We then map$\mathbf{g}_{t,i}$ to the corresponding 3D world coordinates$\{P_{t,i}\}_{i=1}^{I}$

using the depth map and camera parameters.

For each feature $\mathbf{g}_{t,i}$, the observed horizontal orientation $\theta_{t,i}$ and the regional size $s_{t,j}$ are also calculated and stored to enhance the spatial representation. The set of feature points M can therefore be dynamically updated as:

$$\mathrm{M}_t = \mathrm{M}_{t-1} \cup \{\mathbf{g}_{t,i}, P_{t,i}, \theta_{t,i}, s_{t,i}\}_{i=1}^{I}.$$

Ray-View-Panorama Encoding.

The MLP$_{view}$ network aggregates nearby features within feature fields M and encode their spatial information[17] (ie, relative positions and relative directions) to predict semantic representations $\mathbf{r} \in \mathrm{R}^{768}$ and volume density $\sigma \in \mathrm{R}^1$ at any point from any direction in the continuous fields.

For each novel view, our g3D-LF model generates a feature map $\mathbf{R} \in \mathrm{R}^{12 \times 12 \times 768}$ by predicting subregion features through volume rendering within feature fields. The model samples $N$ points along the ray from the camera position to each subregion center to search for the k-nearest features and predicting volume density $\sigma_n$ and latent representation $\mathbf{r}_n$, which then are composited into a subregion feature:

$$\mathbf{R}_{(u,v)} = \sum_{n=1}^{N} \tau_n (1 - \exp(-\sigma_n \Delta_n)) \mathbf{r}_n,$$
$$\text{where} \quad \tau_n = \exp(-\sum_{i=1}^{n-1} \sigma_i \Delta_i).$$

Here, $\tau_n$ represents volume transmittance and $\Delta_n$ is the distance between sampled points. $\mathbf{R}_{(u,v)}$ denotes the regional feature at the $u$-th row and $v$-th column of the novel view feature map R. We integrate context of the surrounding by feeding the feature map R together with a learnable view token $\mathbf{V} \in \mathrm{R}^{768}$ into the transformer-based view encoder to obtain the encoded $\mathbf{R}'$ and novel view representation $\mathbf{V}'$ that represent the entire novel view. Furthermore, to reason relationships across multiple views within a panorama, our g3D-LF model predicts 12 novel views $\{\mathbf{V}_i'\}_{i=1}^{12}$ around the viewpoint at 30-degree intervals and combines them into a transformer-based panorama encoder to obtain $\{\mathbf{V}_i''\}_{i=1}^{12}$.

Ray-BEV Encoding.

The novel view and panorama representations are insufficient for larger-scale scene understanding. To circumvent this problem, we propose to construct BEV map representation via our g3D-LF as shown in Figure 2. Unlike novel view prediction where rays are emitted from the viewpoint along the viewing cone, the rendering rays for the BEV map are rendered vertically from top to bottom. The starting point of the rendered ray is set slightly below the ceiling to avoid being blocked.

Specifically, the MLP$_{BEV}$ network is used to aggregate the nearest feature points to the sampled point and predict its semantic representation $\hat{\mathbf{r}}_n$ and volume density $\hat{\sigma}_n$ in the continuous field. Subsequently, the ray representation $\hat{\mathbf{R}}_{(h,w)} \in \mathrm{R}^{768}$ can be obtained using the similar volume rendering method of Equation 2, where$(h, w)$ denotes the $h$-th row and $w$-th column of the BEV map $\hat{\mathbf{R}} \in \mathrm{R}^{168 \times 168 \times 768}$. To cover the large scene, the BEV map $\hat{\mathbf{R}}$ encompasses a 16.8m × 16.8m area centered on the agent. After downsampling the BEV map to $\hat{\mathbf{R}}_{conv} \in \mathrm{R}^{24 \times 24 \times 768}$ through a non-overlapping 7 × 7 convolution layer, the transformer-based BEV map encoder captures semantic relationships between different regions to get the encoded BEV map representations $\hat{\mathbf{R}}' \in \mathrm{R}^{24 \times 24 \times 768}$.

## 3.3. Multi-level Contrastive Learning

### Balanced Object-level Alignment.

We apply contrastive supervision using an object vocabulary $\mathrm{O} \in \mathrm{R}^{1883 \times 768}$ that spans 1,883 indoor object categories for supervision of the $\mathrm{MLP}_{view}$ and $\mathrm{MLP}_{BEV}$ networks to predict latent features in feature fields. For ray representations R obtained via volume rendering, the cosine similarities $\{\mathrm{CosSim}(\mathbf{R}, \mathrm{O}_i)\}_{i=1}^{1883}$ are computed with each vocabulary embedding. The training objective is to maximize and minimize similarity for the correct and other object category, respectively, *i.e.*:

$$\mathrm{L}_{object} = \mathrm{CrossEntropy}(\{\mathrm{CosSim}(\mathbf{R}, \mathrm{O}_i)/\tau\}_{i=1}^{1883}, \mathrm{O}^{gt}),$$

where $\mathrm{O}^{gt}$ denotes the ground-truth category and $\tau$ is the temperature coefficient for contrastive learning. Similarly, the object alignment loss for the ray representations $\hat{\mathrm{R}}$ of the BEV map denoted as $\hat{\mathrm{L}}_{object}$ can also be calculated.

We notice the network struggles to recognize smaller objects such as the*lamp* due to the dominance of some objects (e.g., *floor* and *walls*) leading to long-tailed distribution in the indoor scenes. To address this issue, we implement a balanced loss that emphasizes harder-to-recognize objects. Specifically, the weight of loss for the rays of top 10% cross entropy aresignificantly increased using a scaling factor $\alpha$ for ray representations within the novel view or BEV map. In short, rays with higher cross entropy indicate harder-to-recognize objects and therefore have a higher loss weight.

### Fine-grained Contrastive for Long Text

To enable our g3D-LF model to understand object relationships and spatial layouts, we propose a fine-grained contrastive learning method for long text alignment. As shown in Figure 2, our g3D-LF aligns the BEV features in a window (e.g., 5 × 5) with the long text features to enhance the representation of the BEV map for spatial semantics. Specifically, centered on an instance, the BEV features $\{\hat{\mathrm{R}}'_i\}_{i=1}^{25}$ within the window are associated with $L$ word features $\{\mathrm{W}_l\}_{l=1}^{L}$ from the CLIP text encoder through an affinity matrix $\mathbf{A}$:

$$\mathbf{A}_{(i,l)} = \mathrm{CosSim}(\hat{\mathrm{R}}'_i, \mathrm{W}_l)/\tau.$$

The highest $L$ similarity scores (equal to the number of words) are extracted from the affinity matrix$\mathbf{A}$, and their average is used as the fine-grained similarity score between the BEV window and the long text features:

$$FineSim(\{\hat{\mathrm{R}}'_i\}_{i=1}^{25}, \{\mathrm{W}_l\}_{l=1}^{L}) = Avg(Topk(\mathbf{A}, L)).$$

Denoting the BEV features within the *i*-th window as $\mathbf{B}_i$ and the *j*-th text features as $\mathbf{T}_j$, the fine-grained contrastive learning loss can be calculated as:

$$\hat{L}_{long\_text} = \frac{1}{J}\sum_{j=1}^{J} CrossEntropy\left(\left\{FineSim\left(\mathbf{B}_i, \mathbf{T}_j\right)\right\}_{i=1}^{I}, j\right) + \frac{1}{I}\sum_{i=1}^{I} CrossEntropy\left(\left\{FineSim\left(\mathbf{T}_j, \mathbf{B}_i\right)\right\}_{j=1}^{J}, i\right).$$

Similarly, our g3D-LF model performs fine-grained contrastive learning between encoded panoramic representations $\{V_i''\}_{i=1}^{12}$ and long-text features $\{W_l\}_{l=1}^{L}$ to compute the fine-grained contrastive loss $L_{long\_text}$.

CLIP Knowledge Distillation

Since the 3D-language data is orders of magnitude smaller than image-language data (millions vs. billions[20]), our g3D-LF model still distills visual features from CLIP model[20] to ensure robust generalization. Specifically, our g3D-LF uses CLIP features extracted from the ground-truth novel view or corresponding region image for contrastive supervision on the predicted new view representation $V'$, the panorama representation $V_i''$, and the BEV map representation $\hat{R}_i'$, ie:

$$L_{view\_clip} = \frac{1}{I}\sum_{i=1}^{I} CrossEntropy\left(\left\{CosSim\left(V_i', V_j^{gt}\right)/\tau\right\}_{j=1}^{J}, i\right),$$

where $V_j^{gt}$ denotes the ground truth CLIP feature for *j*-th novel view representation $V_j'$. Similarly, the contrastive loss $L_{pano\_clip}$ for the panoramic representation and $L_{bev\_clip}$ for the BEV map can also be computed.

## 3.4. Embodied Tasks

To verify the effectiveness of our g3D-LF model for embodied tasks, we integrate the predicted representations from our model into existing baseline methods and evaluates performance on Vision-and-Language Navigation, Zero-shot Object Navigation, and Situated Question Answering tasks.

Vision-and-Language Navigation.

We evaluate the g3D-LF model on VLN tasks with two settings. The first setting is with the monocular camera, which only allows the agent to observe the forward-facing view. As shown in Figure 3, the VLN-3DFF[19] is a monocular VLN model that predicts candidate waypoints around the agent using a semantic map, and predicts each candidate's representation with generalizable feature fields[17] and then selects the optimal waypoint to move through a cross-modal graph encoder[36][37]. Based on this baseline method, we incorporate novel view representations from our g3D-LF model and input the BEV map into the cross-modal graph encoder following GridMM[39] to enhance spatial layout understanding. The second setting is with the panorama camera, in which the agent can observe 12 RGB-D view images within the panorama. Following HNR[17], a waypoint predictor[35] is used to predict candidate waypoints, and our g3D-LF model generates panorama representations of these waypoints for navigation planning.

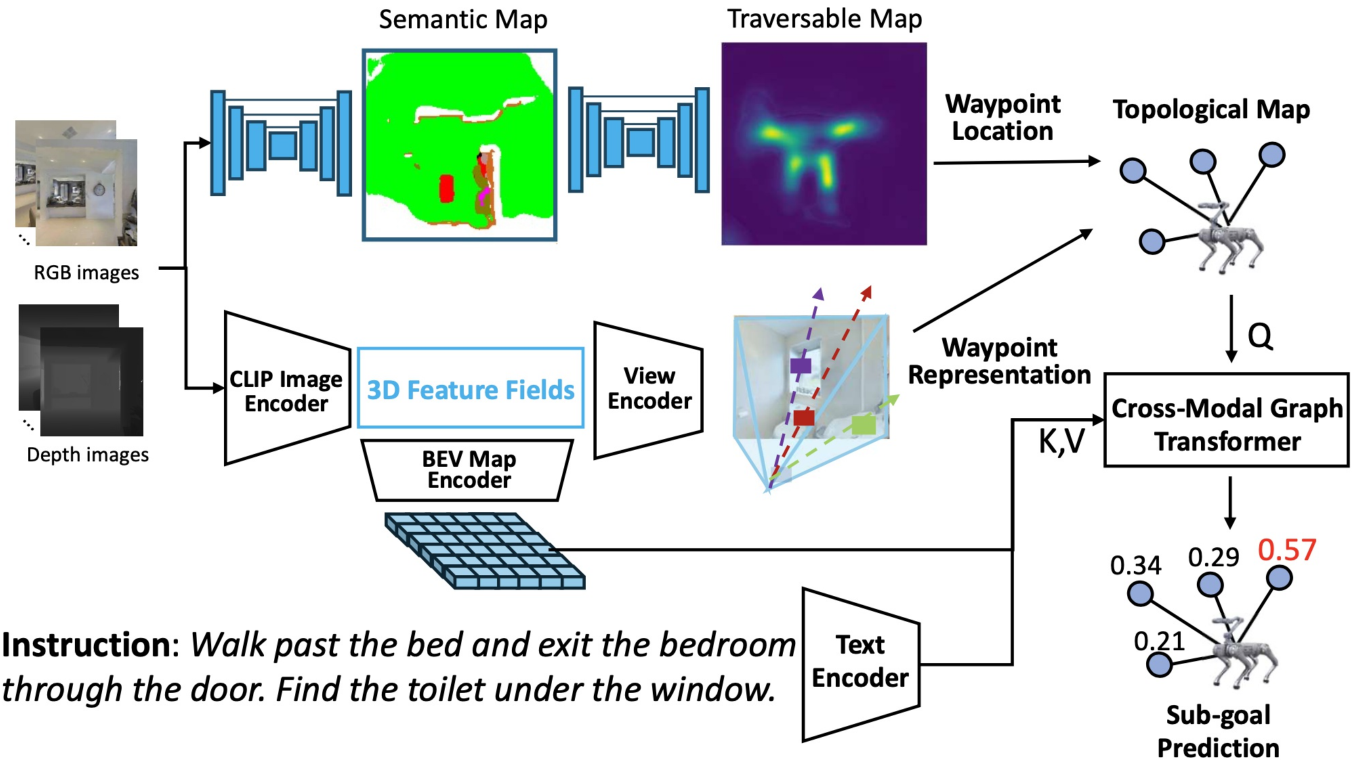


**Figure 3.** Monocular VLN framework based on VLN-3DFF [19].

Zero-shot Object Navigation.

As shown in Figure 4, unlike the baseline method VLFM[6] that uses the 2D foundation model BLIP-2[53] to calculate the similarity between the target object and visual observations to construct the value map, we use our g3D-LF to predict the value of potential regions. Although the monocular agent can only observe the forward view, our g3D-LF predicts 12 novel view feature maps surrounding the agent within panorama based on historical observations, and calculates max similarity in feature map with the target object. The text features of the target object are also used to calculate the similarity with each region representation on the BEV map to obtain a larger-scale value map. Combining these two value maps, the navigation agent prioritizes traveling to the candidate waypoint with the highest similarity score.

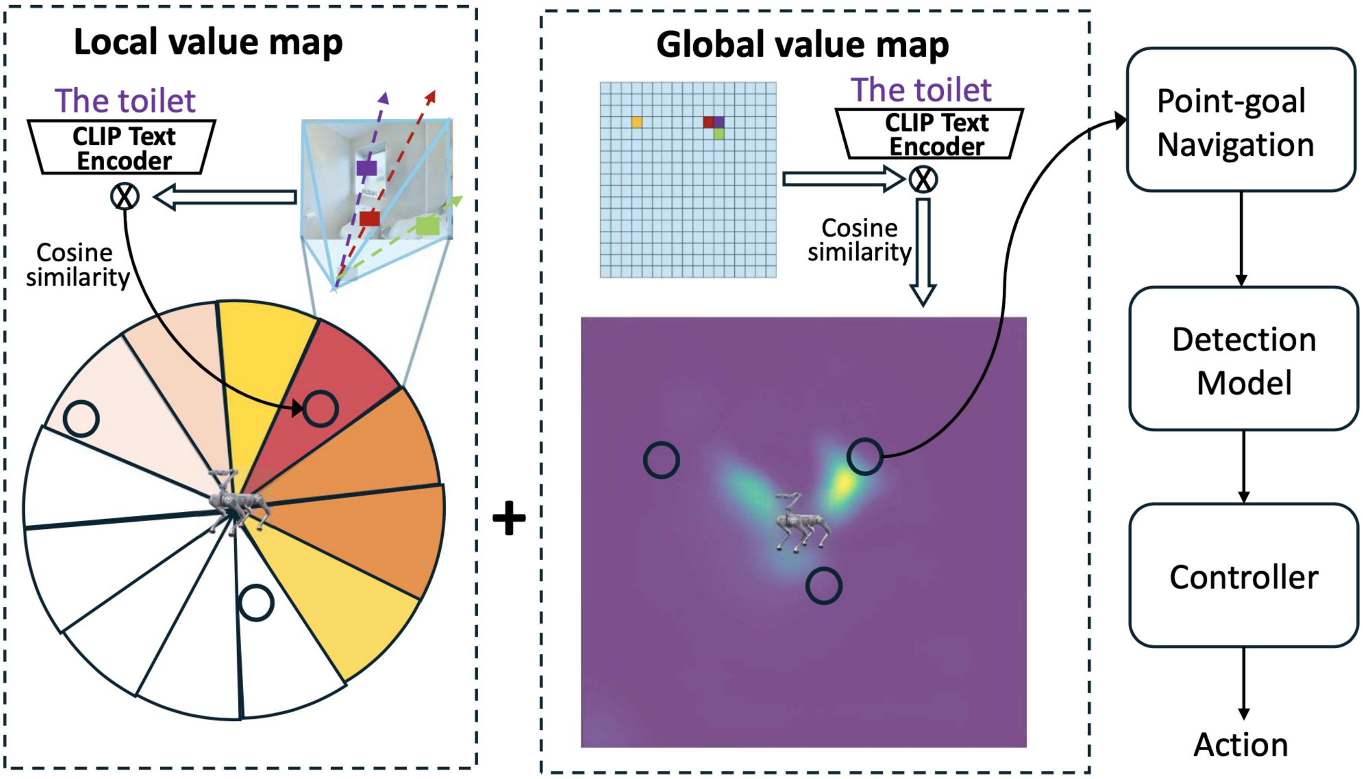


**Figure 4.** Zero-shot object navigation framework based on VLFM [6].

Situated Question Answering.

A three-stage framework is shown in Figure 5, where we use our g3D-LF to train three transformer-based decoders for position, orientation and answer predictions. First, the Localization Decoder predicts the heatmap for location of the textual description based on the BEV map. Our g3D-LF model generates the panorama representations around the predicted location, which are then processed by the Orientation Decoder to predict the orientation. Finally, the textual description, question, BEV map, and panorama representations are fed into the Answer Decoder to generate the final answer.

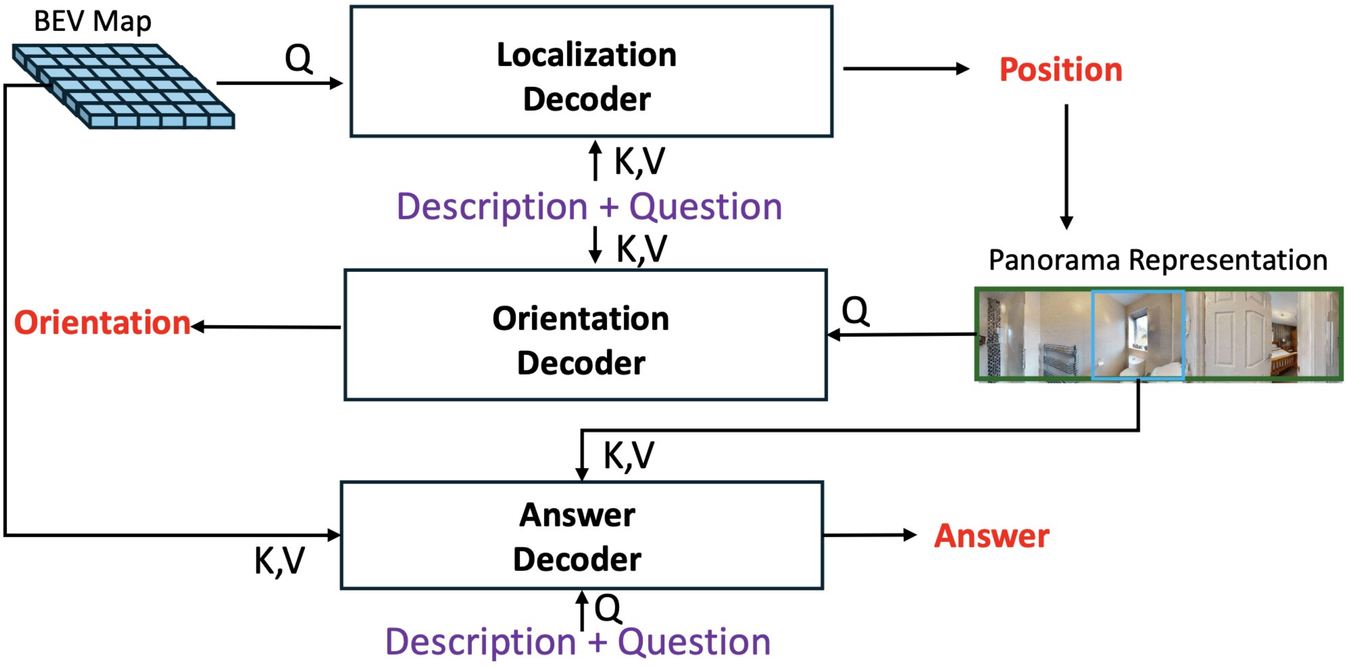


**Figure 5.** The framework of situated question answering[1].

# 4. Experiments

## 4.1. Experiment Setup and Metrics

g3D-LF Pre-training.

We pre-train our g3D-LF model shown in Figure 2 on 5K 3D scenes. During training, 30 frames are uniformly sampled from the RGB-D video of each scene in the ScanNet[56] dataset to construct the feature fields, with an additional frame randomly selected as the novel view for prediction. The g3D-LF then predicts the panorama representation and BEV map centered on the camera of this novel view. For each ray in the novel view or BEV map, the corresponding instance ID can be searched by calculating the nearest instance point to the rendered surface within the annotated instance point cloud. The language annotations of the novel view, panorama, and BEV map can thus be obtained by retrieving language annotations with their instance IDs from the database for training. Due to the limited number of images per scene (fewer than 20), we use all available images from the Structured3D[59] dataset for training. We follow HNR[17] for the HM3D[57][58] dataset using the Habitat simulator[60] to randomly sample navigation trajectories and the observed RGB-D images to predict the novel views and panoramas around candidate waypoints, and construct the BEV map centered on the agent. The multi-level contrastive losses described in Section 3.3 are utilized to optimize the g3D-LF model.

Finally, we combine scenes from all datasets and pretrain our g3D-LF model for 50K episodes (about 10 days) on two

RTX 6000 Ada GPUs. **To ensure fair comparisons on downstream tasks, all training data only includes the train split, the val and test splits are removed.**

Vision-and-Language Navigation.

We evaluate the VLN model on the VLN-CE dataset[8] in both monocular[19] and panorama[17] settings. **R2R-CE** is collected based on the Matterport3D[61] scenes with the Habitat simulator[60]. The R2R-CE dataset includes 5,611 trajectories divided into train, validation seen, validation unseen, and test unseen splits. Each trajectory has three English instructions with an average path length of 9.89 meters and an average instruction length of 32 words. Several standard metrics[7] are used to evaluate VLN performance: Navigation Error (**NE**), Success Rate (**SR**), SR given the Oracle stop policy (**OSR**), Success Rate weighted by normalized inverse Path Length (**SPL**).

Zero-shot Object Navigation.

For object navigation, we evaluate our approach using the Habitat simulator[60] on the validation splits of two different datasets HM3D[57] and MP3D[61]. The **HM3D** validation split contains 2,000 episodes across 20 scenes and 6 object categories. The MP3D validation split contains 2,195 episodes across 11 scenes and 21 object categories. The main metrics[7] include Success Rate (**SR**) and Success Rate weighted by normalized inverse Path Length (**SPL**).

Situated Question Answering.

Following ScanNet[56], the SQA3D dataset comprises 20.4k descriptions and 33.4k diverse questions, which is splited into train, val, and test sets. The main metric is the **Exact Match (EM@1)** of the answer. Additionally, for localization evaluation, **Acc@0.5m** and **Acc@1.0m** metric means the prediction is counted as correct when the predicted position is within 0.5 meter and 1.0 meter range to the ground truth position. The **Acc@15°** and **Acc@30°** metric means the prediction is counted as correct when the prediction orientation is within 15° and 30° range to the ground truth orientation.

## 4.2. Comparison with SOTA Methods

As shown in Table 1 and Table 2, we evaluate the VLN performance of our g3D-LF model on the R2R-CE dataset in both monocular and panorama settings, respectively. Table 1 shows that our g3D-LF significantly outperforms previous monocular VLN methods on the Success Rate (**SR**) metric, even compared to LLM-based methods such as NaVid[62] and InstructNav[63]. Compared to the panorama setting, monocular VLN has the advantage of beingcompatible with a broader range of real-world monocular robots. Our g3D-LF model overcomes the limitations of monocular cameras, enhancing the multi-view and BEV perception capabilities of the agent for monocular VLN.

**Table 1.** Evaluation of VLN on R2R-CE with **monocular** setting. * denotes zero-shot method.

| aMethods | LLM | Val Unseen | | | | Test Unseen | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | NE↓ | OSR↑ | SR↑ | SPL↑ | NE↓ | OSR↑ | SR↑ | SPL↑ |
| **CM$^2$ [64]** | × | 7.02 | 41.5 | 34.3 | 27.6 | 7.7 | 39 | 31 | 24 |
| **WS-MGMap[65]** | × | 6.28 | 47.6 | 38.9 | 34.3 | 7.11 | 45 | 35 | 28 |
| **NaVid[62]** | ✓ | **5.47** | 49.1 | 37.4 | **35.9** | - | - | - | - |
| **InstructNav * [63]** | ✓ | 6.89 | - | 31 | 24 | - | - | - | - |
| **VLN-3DFF[19]** | × | 5.95 | 55.8 | 44.9 | 30.4 | 6.24 | 54.4 | 43.7 | 28.9 |
| **g3D-LF (Ours)** | × | 5.70 | **59.5** | **47.2** | 34.6 | **6.00** | **57.5** | **46.3** | **32.2** |

We follow HNR[17] to perform lookahead exploration through predicted candidate waypoint representations for the panorama setting in Table 2. Although the results show minor performance gains and the advanatges are not as pronounced as its monocular counterpart in Table 1, our g3D-LF model still achieves SOTA performance on the SPL metric and demonstrated competitive results on the SR metric.

**Table 2.** Evaluation of VLN on R2R-CE with **panorama** setting.

| Methods | LLM | Val Unseen | | | | Test Unseen | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | NE↓ | OSR↑ | SR↑ | SPL↑ | NE↓ | OSR↑ | SR↑ | SPL↑ |
| **Sim2Sim[34]** | × | 6.07 | 52 | 43 | 36 | 6.17 | 52 | 44 | 37 |
| **VLN-BERT[35]** | × | 5.74 | 53 | 44 | 39 | 5.89 | 51 | 42 | 36 |
| **GridMM[39]** | × | 5.11 | 61 | 49 | 41 | 5.64 | 56 | 46 | 39 |
| **Ego$^2$-Map[66]** | × | 4.94 | - | 52 | 46 | 5.54 | 56 | 47 | 41 |
| **DREAM[67]** | × | 5.53 | 59 | 49 | 44 | 5.48 | 57 | 49 | 44 |
| **ScaleVLN[68]** | × | 4.80 | - | 55 | 51 | 5.11 | - | 55 | 50 |
| **ETPNav[37]** | × | 4.71 | 65 | 57 | 49 | 5.12 | 63 | 55 | 48 |
| **BEVBert[38]** | × | 4.57 | 67 | 59 | 50 | **4.70** | 67 | **59** | 50 |
| **HNR[17]** | × | **4.42** | 67 | **61** | 51 | 4.81 | 67 | 58 | 50 |
| **Energy[69]** | × | 4.69 | 65 | 58 | 50 | 5.08 | 64 | 56 | 48 |
| **g3D-LF (Ours)** | × | 4.53 | **68** | **61** | **52** | 4.78 | **68** | 58 | **51** |

In Table 3 for the Zero-shot Object Navigation, our g3D-LF achieves SOTA performance in the SPL metric and achieves competitive results in the SR metric. Notably, our g3D-LF is the only method that queries targets using feature fields instead of VLM. Replacement of BLIP-2[53] in VLFM[6] with g3D-LF improves the navigation success rate (SR) by nearly 3%. Although the MP3D benchmark includes some targets outside the g3D-LF object vocabulary, our model still performs well, demonstrating strong generalization. Compared to methods using LLM: InstructNav[63] and SG-Nav[70], our g3D-LF also offers significant advantages in response time and computational cost.

**Table 3.** Evaluation of Zero-shot Object Navigation on the HM3D and MP3D benchmarks.

| Methods | LLM | VLM | Feature Fields | HM3D | | MP3D | |
|---|---|---|---|---|---|---|---|
| | | | | SR↑ | SPL↑ | SR↑ | SPL↑ |
| ZSON[5] | × | ✓ | × | 25.5 | 12.6 | 15.3 | 4.8 |
| ESC[50] | ✓ | ✓ | × | 39.2 | 22.3 | 28.7 | 14.2 |
| VLFM[6] | × | ✓ | × | 52.5 | 30.4 | 36.4 | 17.5 |
| InstructNav[63] | ✓ | ✓ | × | **58.0** | 20.9 | - | - |
| GAMap[71] | ✓ | ✓ | × | 53.1 | 26.0 | - | - |
| SG-Nav[70] | ✓ | ✓ | × | 54.0 | 24.9 | **40.2** | 16.0 |
| g3D-LF (Ours) | × | × | ✓ | 55.6 | **31.8** | 39.0 | **18.8** |

In Table 4 for the Situated Question Answering task, our g3D-LF achieves good localization performance in metrics of Acc@0.5m, Acc@1m, Acc@15° and Acc@30°. Although our performance on the answering accuracy (EM@1) is significantly lower than that of LLM-based methods: LEO[11] and Scene-LLM[14], it is worth noting that our g3D-LF ***only uses images*** as input without low-noise 3D point clouds. This actually offers a significant advantage in agent-centered embodied tasks since it is more adaptable to unseen dynamic real-world environments, where the low-noise point clouds are difficult to collect.

**Table 4.** Evaluation of Situated Question Answering (SQA3D) task. **PCD** denotes methods that use point clouds as input, while **Image** represents methods that use images as input.

| Methods | LLM | PCD | Image | Position | | Orientation | | Answer |
|---|---|---|---|---|---|---|---|---|
| | | | | 0.5m | 1.0m | 15° | 30° | EM@1 |
| ClipBERT[72] | × | × | ✓ | - | - | - | - | 43.3 |
| ScanQA[2] | × | ✓ | × | - | - | - | - | 46.6 |
| SQA3D[1] | × | ✓ | × | 14.6 | 34.2 | 22.4 | 42.3 | 47.2 |
| 3D-VisTA[10] | × | ✓ | × | - | - | - | - | 48.5 |
| SceneVerse[22] | × | ✓ | × | - | - | - | - | 49.9 |
| LEO[11] | ✓ | ✓ | × | - | - | - | - | 52.4 |
| Scene-LLM[14] | ✓ | ✓ | ✓ | - | - | - | - | **54.2** |
| g3D-LF (Ours) | × | × | ✓ | **23.4** | **45.7** | **29.8** | **54.7** | 47.7 |

### 4.3. Ablation Study

Perfromance impact of g3D-LF on embodied tasks.

In row 1 of Table 5, the performance of monocular VLN and object navigation drops significantly without representations from g3D-LF. In this setting, the VLN model only uses the CLIP features from the forward-facing view with features of all other directions set to zero. The object navigation model uses BLIP-2[53] instead of g3D-LF to construct the value map. Examining rows 2 and 3 shows that removing either the novel view or the BEV map reduces the performance of both two tasks, highlighting the role of each g3D-LF module.

Novel views are crucial for monocular VLN.

As shown in row 1 and row 2 of Table 5, the novel view representations significantly boost VLN performance by overcoming the narrow perception of the monocular camera[19], enabling the monocular agent to have panoramic perception capabilities. To some extent, this confirms that novel view prediction is a very important and valuable capability for monocular agents. Based on this capability, the g3D-LF model predicts the novel view representations of candidate waypoints around the agent to construct the topological map for better navigation planning.

Object navigation requires balancing local and global targets.

As shown in row 3 of Table 5, we observe that relying solely on BEV representation significantly reduces object navigation performance. This decline occurs because the global value map from the BEV map fails to select optimal nearby waypoints if the target is far from these waypoints. In this case, a local value map constructed from novel views is also essential to identify the optimal short-term goal, *i.e.*, nearby waypoints around the agent.

**Table 5.** Ablation study for the modules of g3D-LF.

| View & Pano | BEV | Monocular VLN | | | | Object Nav. | |
|---|---|---|---|---|---|---|---|
| | | NE↓ | OSR↑ | SR↑ | SPL↑ | SR↑ | SPL↑ |
| × | × | 6.54 | 44.6 | 33.1 | 23.4 | 52.5 | 30.4 |
| ✓ | × | 5.78 | 58.3 | 46.9 | 32.7 | 53.9 | 30.8 |
| × | ✓ | 6.02 | 53.1 | 42.8 | 26.5 | 50.2 | 27.1 |
| ✓ | ✓ | **5.70** | **59.5** | **47.2** | **34.6** | **55.6** | **31.8** |

**Table 6.** Ablation study for the multi-level contrastive pre-training. **OBJ-CL**: object-level contrastive learning. **CLIP-CL**: knowledge distillation using CLIP visual features from ground-truth view. **FG-CL**: fine-grained contrastive learning for long text understanding.

| OBJ-CL | CLIP-CL | FG-CL | Monocular VLN | | | | Object Nav. | |
|---|---|---|---|---|---|---|---|---|
| | | | NE↓ | OSR↑ | SR↑ | SPL↑ | SR↑ | SPL↑ |
| × | × | × | 6.21 | 50.2 | 40.7 | 24.9 | 34.2 | 13.9 |
| × | ✓ | × | 5.84 | 56.1 | 44.6 | 31.1 | 47.6 | 27.8 |
| ✓ | × | ✓ | 6.01 | 53.5 | 42.4 | 26.7 | **55.8** | 31.6 |
| unbalanced | ✓ | ✓ | 5.73 | 58.3 | 46.6 | 33.0 | 51.7 | 28.8 |
| ✓ | ✓ | coarse | 5.81 | 57.1 | 45.7 | 33.2 | 55.5 | 31.2 |
| ✓ | ✓ | ✓ | **5.70** | **59.5** | **47.2** | **34.6** | **55.6** | **31.8** |

Pre-training is essential for generalizable feature fields model.

Table 6 analyzes the impact of multi-level contrastive pre-training on downstream embodied tasks. As shown in row 1 of Table 6, the performance on VLN and object navigation drops significantly when the model is optimized solely by the navigation loss[37] without pre-training.

Both CLIP distillation and language supervision are indispensable.

For row 3 of Table 6 without supervision from the CLIP visual features, the VLN performance lags behind the model distilled by CLIP. This suggests that millions of language annotations are still far from sufficient for g3D-LF pre-training, and distilling representations from 2D foundation models to enhance semantic generalization remains necessary. However, in Table 6, we can also see that language supervision significantly improves g3D-LF performance on embodied tasks , the model performs poorly in row 2 when using only CLIP distillation.

Long-tail distribution limits object-level semantic learning.

As shown in row 4 of Table 6, the performance of object navigation decreases drastically without the balanced loss mentioned in Section 3.3. The long-tail distribution of object categories in indoor environments leads models to overlook of rare or small objects such as *towels* and *cups*, significantly limiting the ability of our g3D-LF model to query target objects. Fortunately, row 6 of Table 6 shows that the balanced object alignment works well by balancing the weight for loss of hard-to-recognize objects.

Fine-grained contrastive benefits long text understanding.

In the row 5 of Table 6, we use the [SEP] feature (single vector) from the CLIP text encoder to supervise panorama and BEV representations. However, compared to the fine-grained contrastive learning in row 6, compressing long text into a coarse vector significantly limits g3D-LF's performance on long-text understanding tasks such as VLN. As shown in Figure 2, fine-grained contrastive learning between long texts and windows within the BEV map helps g3D-LF understand spatial layouts, overcoming the limitations of semantic representation in large-scale scenes.

**Table 7.** Runtime analysis measured on one RTX 4090 GPU. **FPS** denotes Frames Per Second.

| Rays for View | View | Panorama | Rays for BEV | BEV |
|---|---|---|---|---|
| 73.6 FPS | 71.1 FPS | 5.9 FPS | 6.3 FPS | 6.1 FPS |

**g3D-LF enables real-time inference.** As shown in Table 7, we calculate the inference time of our g3D-LF model on the val unseen split of the R2R-CE dataset in the VLN task. Our g3D-LF achieves novel view volume rendering at 73.6 FPS, which slightly drops to 71.1 FPS when rays are further encoded by the View Encoder. For a panorama containing 12 views, the inference speed is 5.9 FPS. Due to the large rendered range, our g3D-LF renders BEV maps at 6.3 FPS, which drops slightly to 6.1 FPS with the BEV Map Encoder. Our g3D-LF model adopts the same *sparse sampling* strategy as in HNR[17], where the MLP network is only used to render sampled regions containing feature points nearby, while skipping empty regions. This reduces rendering time by over 10 times, enabling real-time embodied tasks.

## 5. Conclusion

In this work, we propose Generalizable 3D-Language Feature Fields (g3D-LF), a 3D representation model pre-trained on large-scale 3D-language data for embodied tasks. We organize the first large-scale 3D-language dataset for feature fields training, demonstrating the feasibility of using generalizable feature fields for large-scale scene understanding, *i.e.*, panorama and BEV. Our proposed g3D-LF leverages multi-level contrastive learning strategies such as balanced object semantic alignment, fine-grained text alignment, and CLIP knowledge distillation to optimize generalized feature fields. More importantly, the value of g3D-LF has been widely evaluated in multiple embodied tasks. We believe that our g3D-LF can provide sufficient inspiration for subsequent research on feature fields and embodied AI.

Limitations and future works.

Our g3D-LF still has some limitations with significant potential for future research: 1) g3D-LF cannot be adapted to dynamic environments, where objects or people are moving in real time. This requires better update strategies for implicit representations. 2) g3D-LF has not been evaluated on dynamic tasks such as object manipulation. 3) The scale and quality of 3D-language data used for training g3D-LF remain limited, which essentially restricts the ability of generalizable feature field models. 4) The 3D feature fields combined with LLM can enable better text generation. These may become the guiding directions for the next phase of generalizable feature fields.

## Supplementary Material

## Appendix A. More Details of the g3D-LF Model

Model structure.

Figure 6 illustrates the structure of main modules in the g3D-LF model. Compared to HNR[17], g3D-LF improve the MLP network for volume rendering by adding residual connections and replacing ReLU with LeakyReLU, which helps alleviate gradient explosion and neuron death issues during HNR training. Since the number of k-nearest features is set to 4 and the dimension of each aggregated feature is 768, the input dimension of both MLP$_{view}$ and MLP$_{BEV}$ networks is 3072. As shown in Figure 6, all transformer-based encoders consist of four-layer transformers.

Settings of novel view prediction.

For each sampled point in the rendered ray, we set the search radius for k-nearest features as 0.5 meter. Using*sparse sampling*[17], if no nearby feature points are found within a sampled point's search radius, the latent feature and volume density are set to zero. The rendered ray is uniformly sampled from 0 to 10 meters, and the number of sampled points is set as 501. After volume rendering, the number of rays within a novel view is set as 12 × 12.

Settings of BEV map prediction.

The search radius for k-nearest features is set as 0.4 meter. The rendered ray is uniformly sampled from 0 to 1.6 meters (i.e., vertically from the camera's position to bottom), and the number of sampled points is set as 17. After volume rendering, the number of rays within a BEV map is set as 168 × 168.

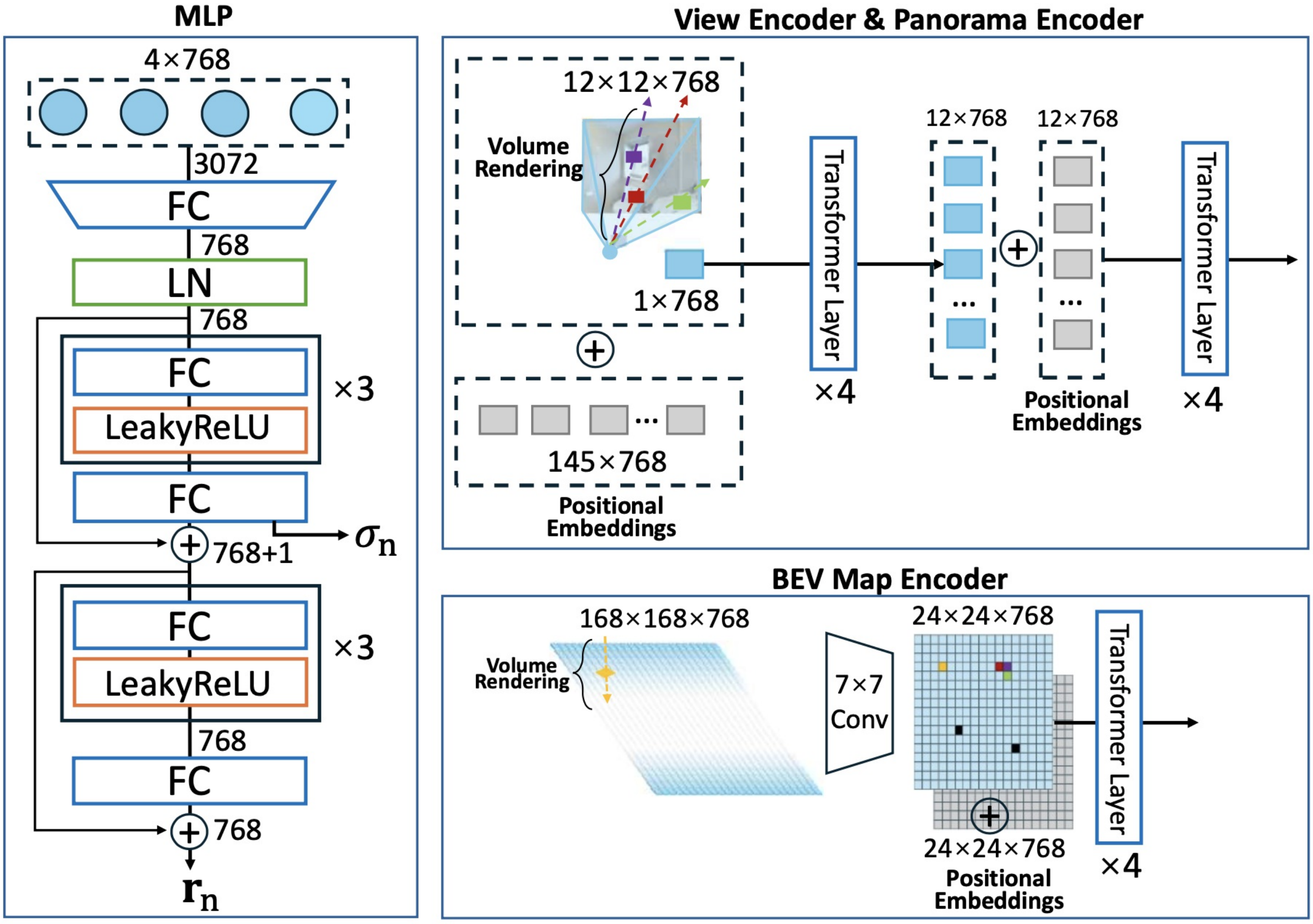
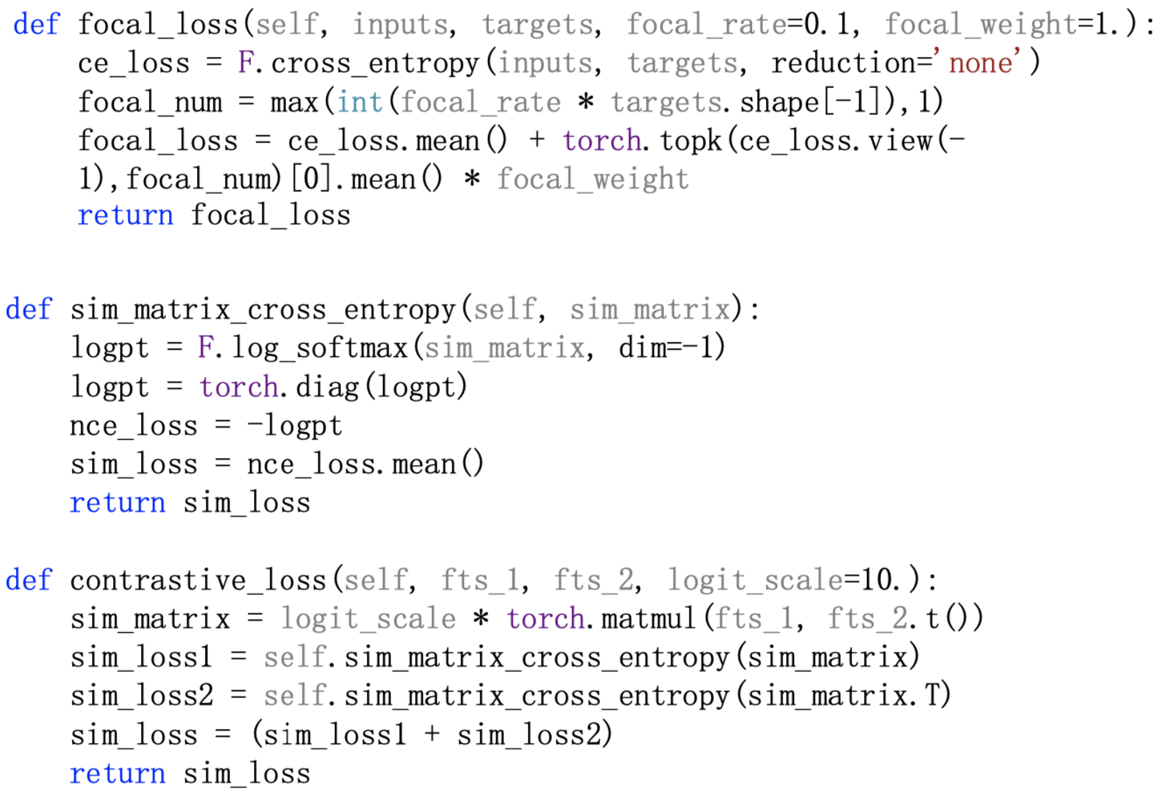


**Figure 6.** Architecture of modules in the g3D-LF model. FC denotes a fully connected layer, LN denotes layer normalization and LeakyReLU [73] is the activation function.

Loss functions.

As illustrated in Figure 7 and 8, we present the code for the primary loss functions used in g3D-LF pre-training to provide further details. During training, we apply constant coefficients to balance the contributions of each loss, ensuring they remain within the same order of magnitude.

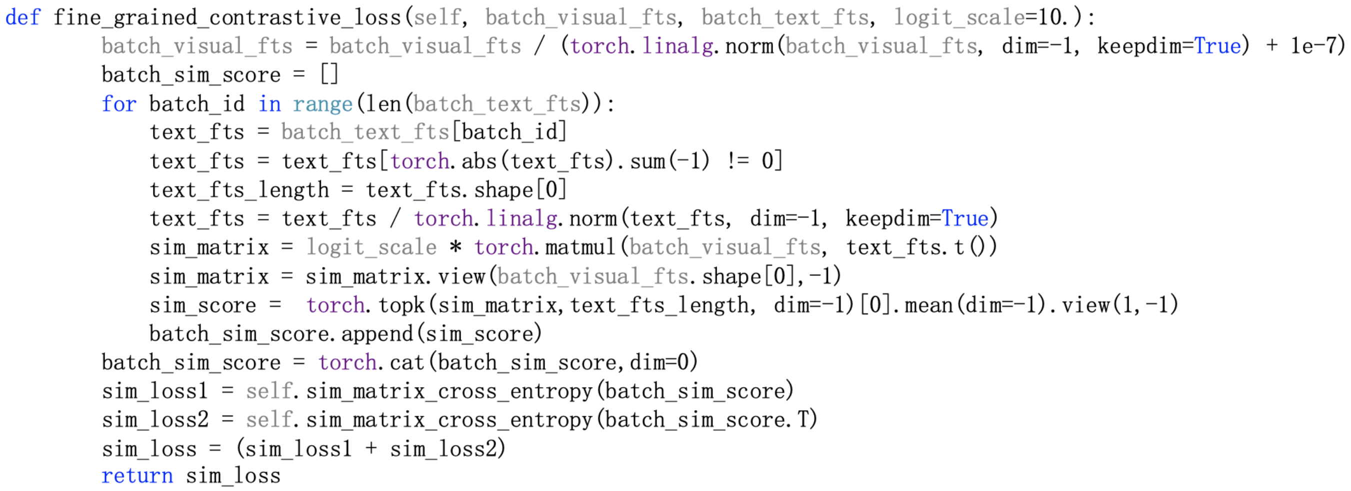


```
def focal_loss(self, inputs, targets, focal_rate=0.1, focal_weight=1.):
    ce_loss = F.cross_entropy(inputs, targets, reduction=’none’)
    focal_num = max(int(focal_rate * targets.shape[-1]),1)
    focal_loss = ce_loss.mean() + torch.topk(ce_loss.view(-
    1),focal_num)[0].mean() * focal_weight
    return focal_loss

def sim_matrix_cross_entropy(self, sim_matrix):
    logpt = F.log_softmax(sim_matrix, dim=-1)
    logpt = torch.diag(logpt)
    nce_loss = -logpt
    sim_loss = nce_loss.mean()
    return sim_loss

def contrastive_loss(self, fts_1, fts_2, logit_scale=10.):
    sim_matrix = logit_scale * torch.matmul(fts_1, fts_2.t())
    sim_loss1 = self.sim_matrix_cross_entropy(sim_matrix)
    sim_loss2 = self.sim_matrix_cross_entropy(sim_matrix.T)
    sim_loss = (sim_loss1 + sim_loss2)
    return sim_loss
```

**Figure 7.** PyTorch implementation of loss functions for the balanced object semantic alignment and the CLIP knowledge distillation.



```
def fine_grained_contrastive_loss(self, batch_visual_fts, batch_text_fts, logit_scale=10.):
        batch_visual_fts = batch_visual_fts / (torch.linalg.norm(batch_visual_fts, dim=-1, keepdim=True) + 1e-7)
        batch_sim_score = []
        for batch_id in range(len(batch_text_fts)):
            text_fts = batch_text_fts[batch_id]
            text_fts = text_fts[torch.abs(text_fts).sum(-1) != 0]
            text_fts_length = text_fts.shape[0]
            text_fts = text_fts / torch.linalg.norm(text_fts, dim=-1, keepdim=True)
            sim_matrix = logit_scale * torch.matmul(batch_visual_fts, text_fts.t())
            sim_matrix = sim_matrix.view(batch_visual_fts.shape[0],-1)
            sim_score =  torch.topk(sim_matrix,text_fts_length, dim=-1)[0].mean(dim=-1).view(1,-1)
            batch_sim_score.append(sim_score)
        batch_sim_score = torch.cat(batch_sim_score,dim=0)
        sim_loss1 = self.sim_matrix_cross_entropy(batch_sim_score)
        sim_loss2 = self.sim_matrix_cross_entropy(batch_sim_score.T)
        sim_loss = (sim_loss1 + sim_loss2)
        return sim_loss
```

**Figure 8.** PyTorch implementation of loss function for the fine-grained contrastive learning.

## B. Visualization of the Training Data

As shown in Figure 9, we present a 3D scene from our dataset along with some associated language annotations (scene 00800-TEEsavR23oF from HM3D[57]). The instance-level point cloud precisely annotates instances within the 3D scene, allowing retrieval of language annotations for any position by calculating its neighboring instance points and using the instance IDs.

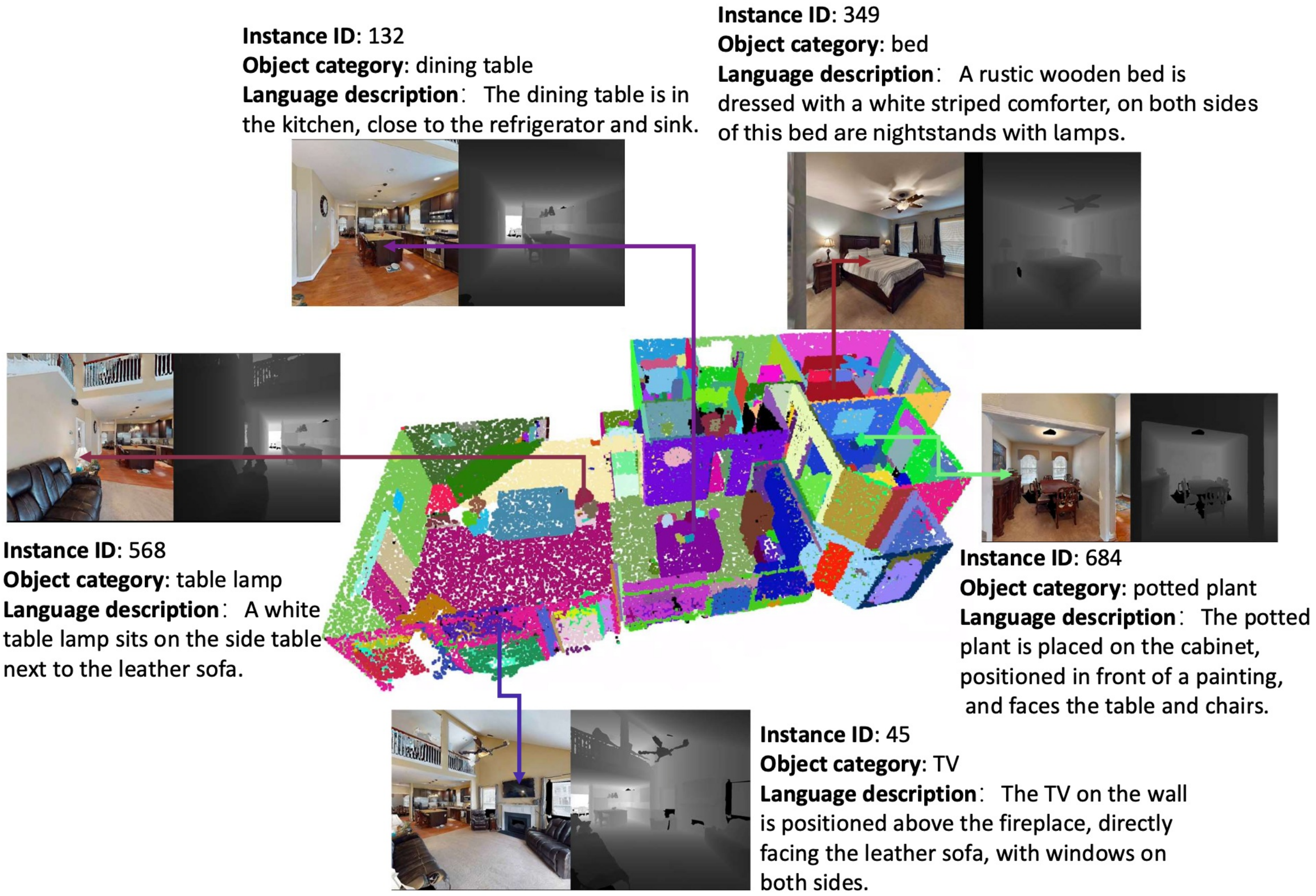

**Instance ID**: 132
**Object category**: dining table
**Language description**： The dining table is in the kitchen, close to the refrigerator and sink.
**Instance ID**: 349
**Object category**: bed
**Language description**： A rustic wooden bed is dressed with a white striped comforter, on both sides of this bed are nightstands with lamps.
**Instance ID**: 568
**Object category**: table lamp
**Language description**： A white table lamp sits on the side table next to the leather sofa.
**Instance ID**: 684
**Object category**: potted plant
**Language description**： The potted plant is placed on the cabinet, positioned in front of a painting, and faces the table and chairs.
**Instance ID**: 45
**Object category**: TV
**Language description**： The TV on the wall is positioned above the fireplace, directly facing the leather sofa, with windows on both sides.

**Figure 9.** Demonstration of a 3D scene in the training data. Instance-level point clouds mark all instances with object categories, and some instances enriched with language descriptions.

## C. Visualization of the g3D-LF model

As shown in Figure 10 and 11, the g3D-LF model query targets with language on the BEV map. In Figure 10, the left side of each example shows the position of the ground-truth target, while the right side displays the result of querying objects on rays of the BEV map during navigation. The BEV map accurately recognizes both large objects, like *window* and *sofa*, and smaller objects, like *table lamp* and *tap*, by calculating the cosine similarity between ray representations and target text features.

In Figure 11, the left side of each example shows the position of the objects, the middle is the ground-truth position of the long text that contains the target object, while the right side displays the result of querying the long text on the BEV map during navigation. In the 3D scene, multiple objects of the same category often appear. With the excellent ability to understand long texts, our g3D-LF model can achieve more fine-grained long-text queries, distinguishing different instances of the same object category.

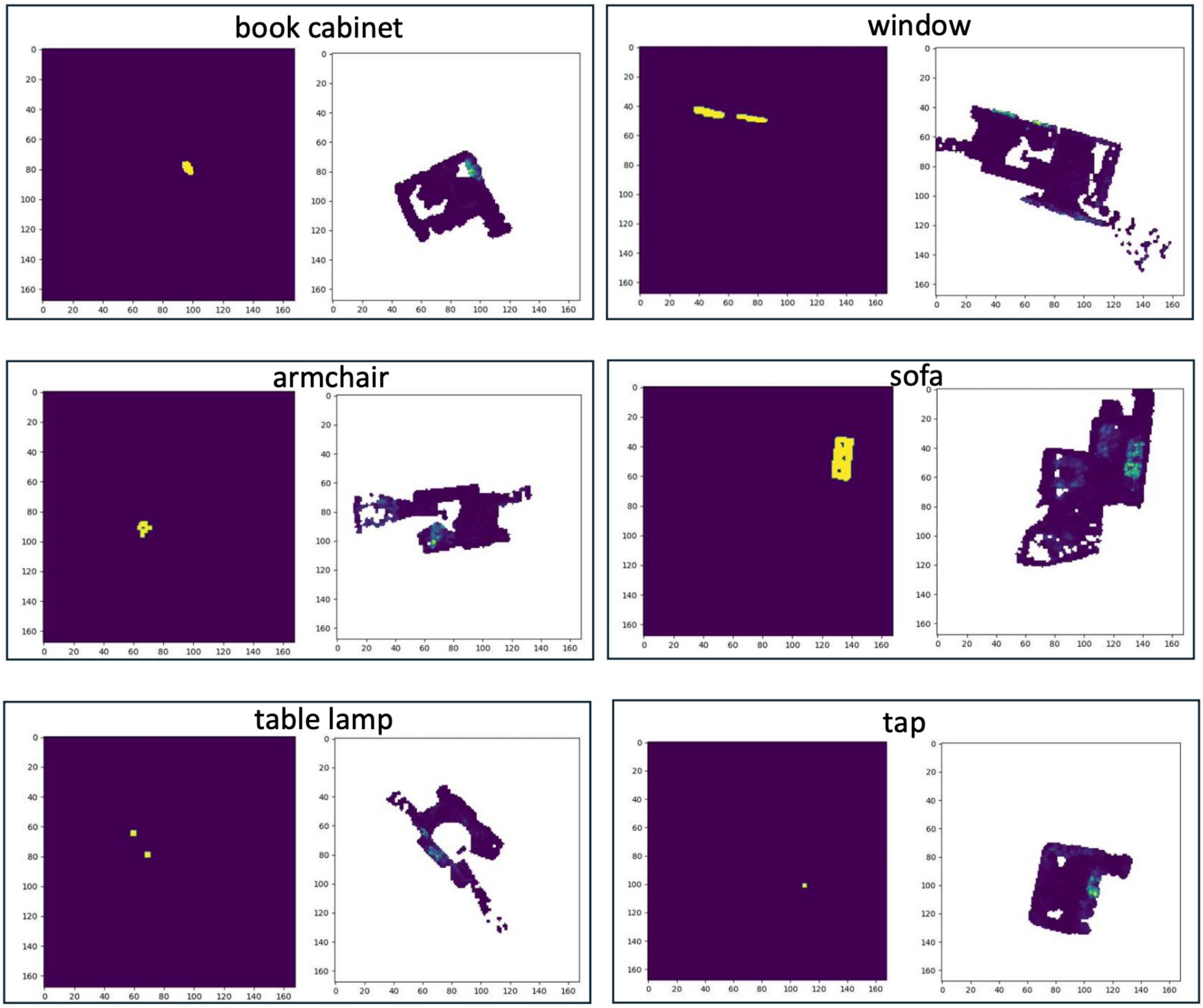


**Figure 10.** Visualization of querying objects on rays of the g3D-LF's BEV map. The left side of each example is GT, and the right side is the query result. Please zoom in for a better view.

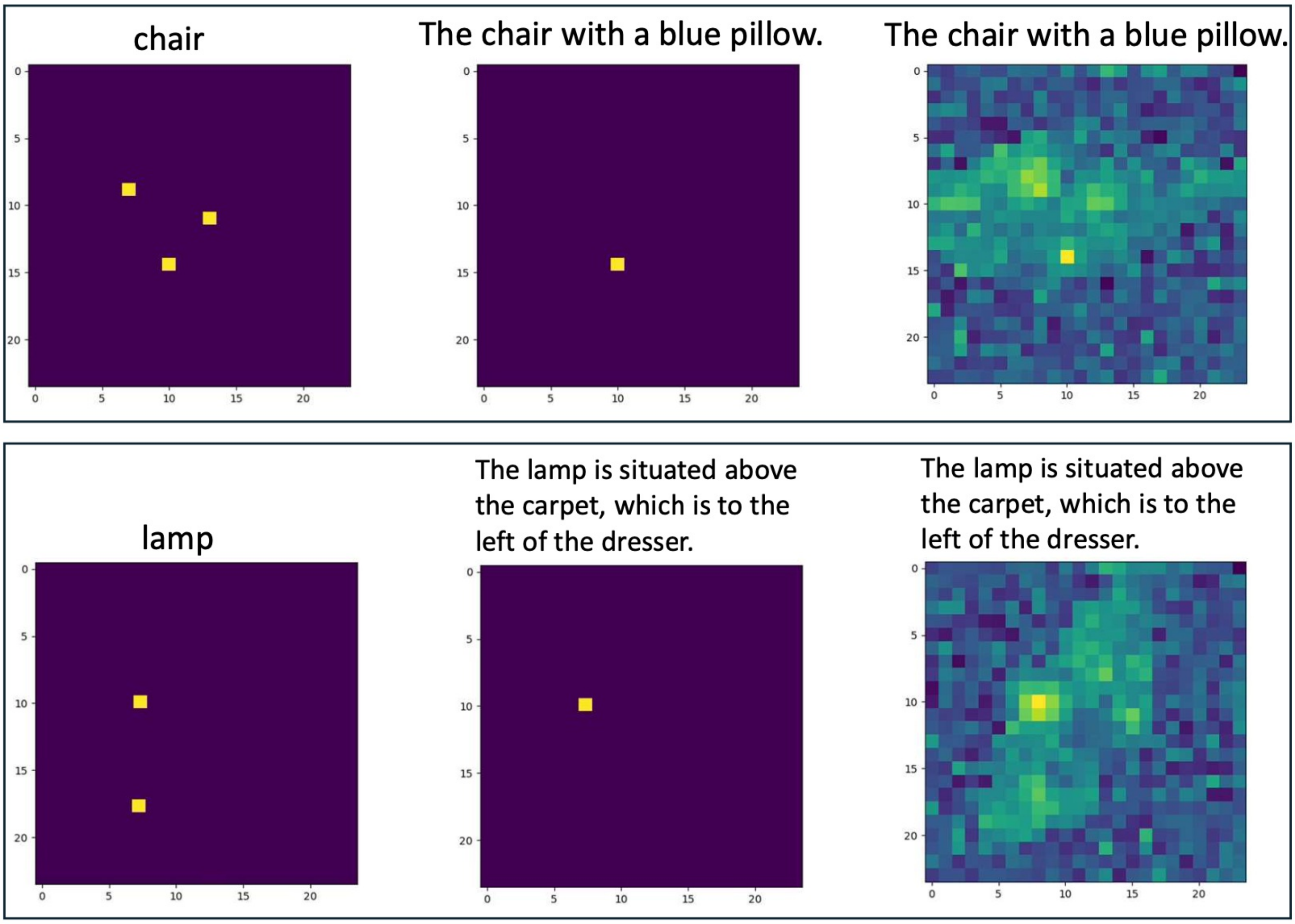


**Figure 11.** Visualization of querying long texts on the BEV map of our g3D-LF. Each example has the object's GT on the left, the long text GT in the middle, and the query result of the long text on the right. Please zoom in for a better view.

## References

1. a, b, c, d, e *Ma X, Yong S, Zheng Z, Li Q, Liang Y, Zhu SC, Huang S (2023). "SQA3D: Situated Question Answering in 3D Scenes". In: The Eleventh International Conference on Learning Representations.*
2. a, b, c *Azuma D, Miyanishi T, Kurita S, Kawanabe M (2022). "Scanqa: 3d question answering for spatial scene understanding". In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition pp. 19129–19139.*
3. a, b *Majumdar A, Ajay A, Zhang X, Putta P, Yenamandra S, Henaff M, Silwal S, Mcvay P, Maksymets O, Arnaud S, et al. Openeqa: Embodied question answering in the era of foundation models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2024. p. 16488–16498.*
4. a, b *Chaplot DS, Gandhi DP, Gupta A, Salakhutdinov RR (2020). "Object goal navigation using goal-oriented semantic exploration". Advances in Neural Information Processing Systems **33**: 4247–4258.*
5. a, b, c *Majumdar A, Aggarwal G, Devnani B, Hoffman J, Batra D (2022). "Zson: Zero-shot object-goal navigation using multimodal goal embeddings". Advances in Neural Information Processing Systems **35**: 32340–32352.*

6. a, b, c, d, e, f, g *Yokoyama N, Ha S, Batra D, Wang J, Bucher B. Vlfm: Vision-language frontier maps for zero-shot semantic navigation. In: 2024 IEEE International Conference on Robotics and Automation (ICRA). IEEE; 2024. p. 42–48.*
7. a, b, c, d *Anderson P, Wu Q, Teney D, Bruce J, Johnson M, Sünderhauf N, Reid I, Gould S, Van Den Hengel A. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. Proceedings of the IEEE conference on computer vision and pattern recognition. 2018:3674--3683.*
8. a, b, c *Krantz J, Wijmans E, Majumdar A, Batra D, Lee S. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In: Computer Vision--ECCV 2020: 16th European Conference, Glasgow, UK, August 23--28, 2020, Proceedings, Part XXVIII 16. Springer; 2020. p. 104--120.*
9. ^ *Kwon O, Park J, Oh S (2023). "Renderable neural radiance map for visual navigation".Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 9099--9108.*
10. a, b *Zhu Z, Ma X, Chen Y, Deng Z, Huang S, Li Q (2023). "3d-vista: Pre-trained transformer for 3d vision and text alignment". Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023: 2911--2921.*
11. a, b, c, d *Huang J, Yong S, Ma X, Linghu X, Li P, Wang Y, Li Q, Zhu S-C, Jia B, Huang S. An embodied generalist agent in 3D world. In: Proceedings of the International Conference on Machine Learning (ICML); 2024.*
12. ^ *Chen Y, Yang S, Huang H, Wang T, Lyu R, Xu R, Lin D, Pang J (2024). "Grounded 3D-LLM with Referent Tokens". arXiv preprint arXiv:2405.10370.*
13. ^ *Liu R, Wang W, Yang Y (2024). "Volumetric Environment Representation for Vision-Language Navigation". Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition pp. 16317–16328.*
14. a, b, c, d *Fu R, Liu J, Chen X, Nie Y, Xiong W (2024). "Scene-llm: Extending language model for 3d visual understanding and reasoning". arXiv preprint arXiv:2403.11401.*
15. ^ *Shen W, Yang G, Yu A, Wong J, Kaelbling LP, Isola P. "Distilled feature fields enable few-shot language-guided manipulation." In: Tan J, Toussaint M, Darvish K, editors. Proceedings of The 7th Conference on Robot Learning. PMLR; 2023. p. 405-424.*
16. ^ *Ze Y, Yan G, Wu YH, Macaluso A, Ge Y, Ye J, Hansen N, Li LE, Wang X. "Gnfactor: Multi-task real robot learning with generalizable neural feature fields." In: Conference on Robot Learning. PMLR; 2023. p. 284-301.*
17. a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p *Wang Z, Li X, Yang J, Liu Y, Hu J, Jiang M, Jiang S. Lookahead exploration with neural radiance representation for continuous vision-language navigation. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2024:13753-13762.*
18. a, b, c *Qiu RZ, Hu Y, Yang G, Song Y, Fu Y, Ye J, Mu J, Yang R, Atanasov N, Scherer S, et al. Learning generalizable feature fields for mobile manipulation. arXiv preprint arXiv:2403.07563. 2024.*
19. a, b, c, d, e, f, g, h, i *Wang Z, Li X, Yang J, Liu Y, Jiang S. "Sim-to-Real Transfer via 3D Feature Fields for Vision-and-Language Navigation." In: 8th Annual Conference on Robot Learning; 2024.*
20. a, b, c, d, e *Radford A, Kim JW, Hallacy C, Ramesh A, Goh G, Agarwal S, Sastry G, Askell A, Mishkin P, Clark J, et al. Learning transferable visual models from natural language supervision. In: International conference on machine learning. PMLR; 2021. p. 8748–8763.*

21. ^Oquab M, Darcet T, Moutakanni T, Vo H, Szafraniec M, Khalidov V, Fernandez P, Haziza D, Massa F, El-Nouby A, et al. DINOv2: Learning Robust Visual Features without Supervision. Transactions on Machine Learning Research Journal. 2024:1–31.

22. a, b, c, dJia B, Chen Y, Yu H, Wang Y, Niu X, Liu T, Li Q, Huang S. Sceneverse: Scaling 3d vision-language learning for grounded scene understanding. In: European Conference on Computer Vision (ECCV); 2024.

23. a, bChen DZ, Chang AX, Nießner M. Scanrefer: 3d object localization in rgb-d scans using natural language. In: European conference on computer vision. Springer; 2020. p. 202–221.

24. ^Zhang H, Zantout N, Kachana P, Wu Z, Zhang J, Wang W (2024). "VLA-3D: A Dataset for 3D Semantic Scene Understanding and Navigation". arXiv preprint arXiv:2411.03540.

25. a, bMildenhall B, Srinivasan PP, Tancik M, Barron JT, Ramamoorthi R, Ng R (2021). "Nerf: Representing scenes as neural radiance fields for view synthesis". Communications of the ACM. **65** (1): 99–106.

26. ^Vora S, Radwan N, Greff K, Meyer H, Genova K, Sajjadi MSM, Pot E, Tagliasacchi A, Duckworth D. "NeSF: Neural Semantic Fields for Generalizable Semantic Segmentation of 3D Scenes". Transactions on Machine Learning Research.

27. ^Kerr J, Kim CM, Goldberg K, Kanazawa A, Tancik M (2023). "Lerf: Language embedded radiance fields". Proceedings of the IEEE/CVF International Conference on Computer Vision. 19729--19739.

28. ^Taioli F, Cunico F, Girella F, Bologna R, Farinelli A, Cristani M. "Language-enhanced rnr-map: Querying renderable neural radiance field maps with natural language." In: Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023. p. 4669-4674.

29. ^Zhang Y, Ma Z, Li J, Qiao Y, Wang Z, Chai J, Wu Q, Bansal M, Kordjamshidi P (2024). "Vision-and-language navigation today and tomorrow: A survey in the era of foundation models". arXiv preprint arXiv:2407.07035.

30. ^Hong Y, Wu Q, Qi Y, Rodriguez-Opazo C, Gould S. "Vln bert: A recurrent vision-and-language bert for navigation". In: Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition 2021. p. 1643–1653.

31. ^Chen S, Guhur P-L, Schmid C, Laptev I (2021). "History aware multimodal transformer for vision-and-language navigation". Advances in neural information processing systems **34**: 5834–5847.

32. ^Qiao Y, Qi Y, Hong Y, Yu Z, Wang P, Wu Q (2023). "Hop+: History-enhanced and order-aware pre-training for vision-and-language navigation". IEEE Transactions on Pattern Analysis and Machine Intelligence. **45** (7): 8524–8537.

33. ^Wang L, He Z, Dang R, Shen M, Liu C, Chen Q (2024). "Vision-and-Language Navigation via Causal Learning". In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition pp. 13139–13150.

34. a, bKrantz J, Lee S (2022). "Sim-2-Sim Transfer for Vision-and-Language Navigation in Continuous Environments". In: European Conference on Computer Vision (ECCV), 2022.

35. a, b, cHong Y, Wang Z, Wu Q, Gould S. "Bridging the gap between learning in discrete and continuous environments for vision-and-language navigation." In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2022.

36. a, bChen S, Guhur P-L, Tapaswi M, Schmid C, Laptev I. Think global, act local: Dual-scale graph transformer for vision-

*and-language navigation. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2022:16537-16547.*

37. a, b, c, d*An D, Wang H, Wang W, Wang Z, Huang Y, He K, Wang L (2024). "Etpnav: Evolving topological planning for vision-language navigation in continuous environments". IEEE Transactions on Pattern Analysis and Machine Intelligence. IEEE.*
38. a, b*An D, Qi Y, Li Y, Huang Y, Wang L, Tan T, Shao J (2023). "Bevbert: Multimodal map pre-training for language-guided navigation". Proceedings of the IEEE/CVF International Conference on Computer Vision 2023: 2737–2748.*
39. a, b, c*Wang Z, Li X, Yang J, Liu Y, Jiang S. "Gridmm: Grid memory map for vision-and-language navigation." In: Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023. p. 15625-15636.*
40. ^*Liu R, Wang X, Wang W, Yang Y. "Bird's-Eye-View Scene Graph for Vision-Language Navigation." In:Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023:10968-10980.*
41. ^*Ramakrishnan SK, Chaplot DS, Al-Halah Z, Malik J, Grauman K. "Poni: Potential functions for objectgoal navigation with interaction-free learning." In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022. p. 18890-18900.*
42. ^*Zhang S, Song X, Bai Y, Li W, Chu Y, Jiang S (2021). "Hierarchical object-to-zone graph for object navigation". Proceedings of the IEEE/CVF international conference on computer vision 2021: 15130–15140.*
43. ^*Zhu Y, Mottaghi R, Kolve E, Lim JJ, Gupta A, Fei-Fei L, Farhadi A. Target-driven visual navigation in indoor scenes using deep reinforcement learning. In: 2017 IEEE international conference on robotics and automation (ICRA). IEEE; 2017. p. 3357–3364.*
44. ^*Wang CY, Bochkovskiy A, Liao HY. "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors". Proceedings of the IEEE/CVF conference on computer vision and pattern recognition 2023: 7464-7475.*
45. ^*Liu S, Zeng Z, Ren T, Li F, Zhang H, Yang J, Jiang Q, Li C, Yang J, Su H, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. arXiv preprint arXiv:2303.05499. 2023.*
46. ^*Kirillov A, Mintun E, Ravi N, Mao H, Rolland C, Gustafson L, Xiao T, Whitehead S, Berg AC, Lo WY, et al. Segment anything. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023. p. 4015–4026.*
47. ^*Zhang C, Han D, Qiao Y, Kim JU, Bae SH, Lee S, Hong CS (2023). "Faster segment anything: Towards lightweight sam for mobile applications". arXiv preprint arXiv:2306.14289.*
48. ^*He K, Gkioxari G, Dollár P, Girshick R. "Mask r-cnn". In:Proceedings of the IEEE international conference on computer vision. 2017. p. 2961--2969.*
49. ^*Gervet T, Chintala S, Batra D, Malik J, Chaplot DS (2023). "Navigating to objects in the real world"Science Robotics. **8** (79): eadf6991.*
50. a, b*Zhou K, Zheng K, Pryor C, Shen Y, Jin H, Getoor L, Wang XE. "Esc: Exploration with soft commonsense constraints for zero-shot object navigation." In: International Conference on Machine Learning. PMLR; 2023. p. 42829-42842.*
51. ^*Gadre SY, Wortsman M, Ilharco G, Schmidt L, Song S (2023). "Cows on pasture: Baselines and benchmarks for language-driven zero-shot object navigation". Proceedings of the IEEE/CVF Conference on Computer Vision and*

*Pattern Recognition. pages 23171–23181.*

52. ^*Li LH, Zhang P, Zhang H, Yang J, Li C, Zhong Y, Wang L, Yuan L, Zhang L, Hwang JN, et al. Grounded language-image pre-training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2022. p. 10965–10975.*
53. a, b, c, d*Li J, Li D, Savarese S, Hoi S. "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models." In: International conference on machine learning. PMLR; 2023. p. 19730-19742.*
54. ^*Wijmans E, Kadian A, Morcos A, Lee S, Essa I, Parikh D, Savva M, Batra D (2019). "Dd-ppo: Learning near-perfect pointgoal navigators from 2.5 billion frames". arXiv preprint arXiv:1911.00357. 2019.*
55. ^*Das A, Datta S, Gkioxari G, Lee S, Parikh D, Batra D (2018). "Embodied question answering".Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1--10.*
56. a, b, c*Dai A, Chang AX, Savva M, Halber M, Funkhouser T, Nießner M. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: Proceedings of the IEEE conference on computer vision and pattern recognition 2017. p. 5828–5839.*
57. a, b, c, d*Ramakrishnan SK, Gokaslan A, Wijmans E, Maksymets O, Clegg A, Turner JM, Undersander E, Galuba W, Westbury A, Chang AX, et al. Habitat-Matterport 3D Dataset (HM3D): 1000 Large-scale 3D Environments for Embodied AI. In: Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2).*
58. a, b*Yadav K, Ramrakhya R, Ramakrishnan SK, Gervet T, Turner J, Gokaslan A, Maestre N, Chang AX, Batra D, Savva M, et al. Habitat-matterport 3d semantics dataset. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023. p. 4927–4936.*
59. a, b*Zheng J, Zhang J, Li J, Tang R, Gao S, Zhou Z (2020). "Structured3D: A Large Photo-realistic Dataset for Structured 3D Modeling". In: Proceedings of The European Conference on Computer Vision (ECCV).*
60. a, b, c*Savva M, Kadian A, Maksymets O, Zhao Y, Wijmans E, Jain B, Straub J, Liu J, Koltun V, Malik J, et al. Habitat: A platform for embodied ai research. In: Proceedings of the IEEE/CVF international conference on computer vision 2019. p. 9339–9347.*
61. a, b*Chang A, Dai A, Funkhouser T, Halber M, Niebner M, Savva M, Song S, Zeng A, Zhang Y. Matterport3D: Learning from RGB-D Data in Indoor Environments. In: International Conference on 3D Vision (3DV); 2017.*
62. a, b*Zhang J, Wang K, Xu R, Zhou G, Hong Y, Fang X, Wu Q, Zhang Z, Wang H. "NaVid: Video-based VLM Plans the Next Step for Vision-and-Language Navigation." In: Proceedings of Robotics: Science and Systems (RSS); 2024.*
63. a, b, c, d*Long Y, Cai W, Wang H, Zhan G, Dong H. "InstructNav: Zero-shot System for Generic Instruction Navigation in Unexplored Environment." In: 8th Annual Conference on Robot Learning, 2024. Available from: https://openreview.net/forum?id=fCDOfpTCzZ.*
64. ^*Georgakis G, Schmeckpeper K, Wanchoo K, Dan S, Miltsakaki E, Roth D, Daniilidis K. "Cross-modal map learning for vision and language navigation." In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022:15460-15470.*
65. ^*Chen P, Ji D, Lin K, Zeng R, Li T, Tan M, Gan C (2022). "Weakly-supervised multi-granularity map learning for vision-*

*and-language navigation". Advances in Neural Information Processing Systems* ***35***: 38149–38161.

66. ^Hong Y, Zhou Y, Zhang R, Dernoncourt F, Bui T, Gould S, Tan H. "Learning navigational visual representations with semantic map supervision." In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023. p. 3055-3067.
67. ^Wang H, Liang W, Van Gool L, Wang W. "Dreamwalker: Mental planning for continuous vision-language navigation." *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023:10873-10883.
68. ^Wang Z, Li J, Hong Y, Wang Y, Wu Q, Bansal M, Gould S, Tan H, Qiao Y (2023). "Scaling data generation in vision-and-language navigation". *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 12009–12020.
69. ^Liu R, Wang W, Yang Y. "Vision-Language Navigation with Energy-Based Policy." In:*Advances in Neural Information Processing Systems*. 2024.
70. a, bYin H, Xu X, Wu Z, Zhou J, Lu J. "SG-Nav: Online 3D Scene Graph Prompting for LLM-based Zero-shot Object Navigation". In: *Advances in Neural Information Processing Systems*, 2024.
71. ^Yuan S, Huang H, Hao Y, Wen C, Tzes A, Fang Y. GAMap: Zero-Shot Object Goal Navigation with Multi-Scale Geometric-Affordance Guidance. In: *Advances in Neural Information Processing Systems*; 2024.
72. ^Lei J, Li L, Zhou L, Gan Z, Berg TL, Bansal M, Liu J (2021). "Less is more: Clipbert for video-and-language learning via sparse sampling". *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* 2021: 7331–7341.
73. ^Maas AL, Hannun AY, Ng AY, et al. Rectifier nonlinearities improve neural network acoustic models. In:*Proc. icml*. Atlanta, GA; 2013. p. 3.