# Topo-Field: Topometric mapping with Brain-inspired Hierarchical Layout-Object-Position Fields

Jiawei Hou[1], Wenhao Guan[1], Longfei Liang[2], Jianfeng Feng[3], Xiangyang Xue[1], and Taiping Zeng[3,*]

***Abstract*— Mobile robots require comprehensive scene understanding to operate effectively in diverse environments, enriched with contextual information such as layouts, objects, and their relationships. Although advances like neural radiation fields (NeRFs) offer high-fidelity 3D reconstructions, they are computationally intensive and often lack efficient representations of traversable spaces essential for planning and navigation. In contrast, topological maps are computationally efficient but lack the semantic richness necessary for a more complete understanding of the environment. Inspired by a population code in the postrhinal cortex (POR) strongly tuned to spatial layouts over scene content rapidly forming a high-level cognitive map, this work introduces Topo-Field, a framework that integrates Layout-Object-Position (LOP) associations into a neural field and constructs a topometric map from this learned representation. LOP associations are modeled by explicitly encoding object and layout information, while a Large Foundation Model (LFM) technique allows for efficient training without extensive annotations. The topometric map is then constructed by querying the learned neural representation, offering both semantic richness and computational efficiency. Empirical evaluations in multi-room environments demonstrate the effectiveness of Topo-Field in tasks such as position attribute inference, query localization, and topometric planning, successfully bridging the gap between high-fidelity scene understanding and efficient robotic navigation.**

## I. Introduction

Mobile robots are rapidly moving from research labs to widespread use. For these robots to operate autonomously in complex environments, a deep understanding of their surroundings is crucial [1]. Hierarchical graph-like scene representation along with detailed environmental reconstruction enabling efficient path planning, will be key for robotic deployment in real-world scenarios [2].

Recently, detailed environmental reconstruction has made great progress in producing lifelike 3D reconstructions [3]–[6], in which NeRF [7] is a prime instance. As improvements, works like [8]–[10] introduce semantic information for better scene understanding. Further, features powered by Large-Foundation-Models (LFMs), trained on massive datasets across various scenes, are employed with general knowledge for open scene understanding [11]–[13]. However, it is computationally demanding and lacks global layout information using detailed neural fields for planning and navigation.

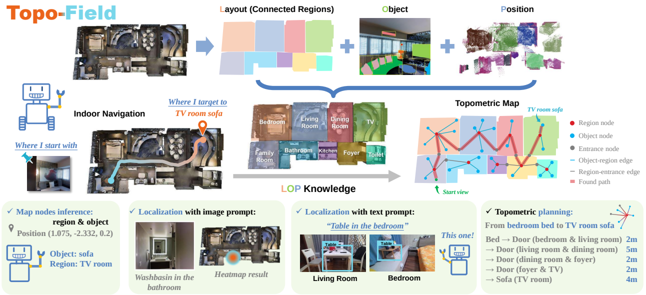

Fig. 1: **Illustration of the Topo-Field strategy and capabilities.** Hierarchically dividing scene information into layout, object, and position to model them explicitly, layout-object-position associated knowledge enables robots with a topometric map representing the scene and planning navigable path to realize a more comprehensive spatial cognition.

In contrast, existing topological maps for path planning and navigation in complex environments are often derived from LiDAR Simultaneous-Localization-and-Mapping (SLAM) using 3D dense submaps [14] or visual SLAM by clustering free-space regions and extracting occupancy information from point clouds [2]. While this approach increases path planning accuracy, computing topology with traditional methods comes with high computational costs and tends to strip away essential semantic information, reducing the robot's ability to fully understand and interpret the environment, which is critical for advanced autonomous functions such as language/image-prompted localization and navigation.

To this end, we propose to build a neural representation as spatial knowledge and construct a topometric map based on this, originating from a brain-inspired approach. Theoretically, neuroscientists have long discovered that animals process their surroundings using topological coding, forming what is known as a ”cognitive map” [15], a concept embodied by place cells [16]. These place cells, along with spatial view cells [17], respond to specific scene contents. More recently, research has shown that a population code in the postrhinal cortex (POR) is strongly tuned to spatial layout rather than scene content [18], capturing spatial representations relative to environmental centers to form a high-level cognitive map from egocentric perception to allocentric understanding [19].

Most related works either do not explicitly represent the layout features [20] or build the topo-map in a clustering and incremental mapping way [21], [22]. On the contrary,

*Corresponding author

[1] School of Computer Science, Fudan University, Shanghai, China {jwhou23, whguan21}@m.fudan.edu.cn xyxue@fudan.edu.cn

[2] Shanghai NeuHelium Neuromorphic Intelligence Tech. Co., Ltd., Shanghai, China longfei.liang@neuhelium.com

[3] Institute of Science and Technology for Brain-Inspired Intelligence, Fudan University, Shanghai, China {jffeng, zengtaiping}@fudan.edu.cn

we intuitively abstract the neural representations of space to build topo-field in three key aspects: 1) The cognitive map corresponds to a topometric map, which uses graph-like representations to encode relationships among its components, e.g. layouts and objects. 2) The population of place cells is analogous to a neural implicit representation with position encoding, enabling location-specific responses. 3) POR, which prioritizes spatial layouts over content, aligns with our spatial layout encoding of connected regions.

This work proposes a Topo-Field, integrating the Layout-Object-Position (LOP) association into neural field training and constructing a topometric map based on the learned neural implicit representation for hierarchical robotic scene understanding. By inputting RGB-D sequences, objects and background contexts are encoded separately as contents and layout information to train a neural field, forming a detailed scene representation. A contrast loss against features from LFMs is employed, resulting in little need for annotation. Further, a topometric map is built by querying the learned field, which is efficient for navigable path planning. To validate the effectiveness of Topo-Field, we conduct quantitative and qualitative experiments on several multi-room apartment scenes evaluating the abilities including position attributes inference, text/image query localization, and planning.

Our contributions can be listed as follows:

- **Brain-inspired Topo-Field:** We introduce a Topo-Field that combines neural scene representation with efficient topometric mapping, enabling hierarchical robotic scene understanding and navigable path planning.
- **Cognitive Map Representation:** Inspired by the population code in postrhinal cortex (POR) strongly tuned to spatial layouts over scene content rapidly forming a high-level cognitive map, we incorporate the concepts of neural representations of spatial layouts, objects, and place cells to construct hierarchical robotic topometric maps.
- **Layout-Object-Position (LOP) Representation:** We develop an implicit neural representation associating layout, object, and position information, which is explicitly supervised using an LFM-powered strategy, requiring minimal human annotation.
- **Topometric Map Construction:** We propose a two-stage pipeline for building a topometric map by querying the learned neural field and validating edges among vertices using LLMs, enabling efficient path planning.

## II. Related Works

### A. Dense Representation with Neural Radiance Field

Detailed 3D scene reconstruction has made great efforts in producing lifelike results, among which NeRF (Neural Radiance Fields) [7] has widely attracted researchers' attention. A popular research direction is to integrate semantics with NeRF to achieve a more comprehensive understanding of scenes [8]–[10]. Recently, several robotic works have demonstrated that features from LFMs can be used for self-supervised learning, which reduces the costly manual annotation [11]–[13]. However, they focus on object semantics but do not include layout-level features. RegionPLC [20] considered region information but with no explicit representation of layout features. In contrast, in our work, CLIP [23] and Sentence-BERT [24] are employed to generate vision-language and semantic features for objects and layout learned respectively.

### B. Topometric Map for Scene Structure Understanding

Using detailed neural fields for planning and navigation is computationally demanding, on the other hand, hybrid topometric mapping has been known for its efficiency in terms of managing the information and being queried for downstream tasks [25]–[27]. It takes advantage of both metric maps and topological maps. However, most topological maps have not introduced information such as semantics. This makes it unsuitable for language/image-guided planning tasks, which is a growing trend in scene representation applications. Concept-graph [28] makes a step forward utilizing LFM to model the object structure with a topo map. CLIO [21]built a task-driven scene graph forming task-relevant clusters of primitives. HOV-SG [22] proposed using feature point cloud clustering and mapping in an incremental approach. On the contrary, we propose to build the topometric map by querying the trained neural field which serves as knowledge-like memory base, whose nodes and edges include attributes representing object and layout information.

### C. Spatial Understanding with Layout Information

Generally, topology is built based on clustering from occupancy information or Voronoi diagrams [29], regardless of the contents and layout relationship. However, neuroscience findings suggest a mechanism to form a high-level cognitive map from egocentric perception to allocentric representation [15], [19]. Place cells [16], as the embodiment of cognitive map, together with spatial view cells show activity to contents [17]. Recently, Patrick et al. [18] showed that a population code in the POR is more strongly tuned to the spatial layout than to the content in a scene. This suggests that there are specialized cells and signaling mechanisms to process layout in the process of scene understanding, which captures the spatial layout of complex environments to rapidly form a high-level cognitive map representation [19]. Inspired by the above research, we mimic the neural scene understanding mechanism by employing egocentric neural field with content and layout knowledge to construct allocentric topometric map.

## III. Overview

We propose to learn an implicit representation of a scene with the neural encoding approach by establishing associations between 3D positions and their corresponding layout and object features as the scene knowledge. Then, a topometric map is built with the learned neural field to form an efficient and queriable representation with a comprehensive understanding of the scene. Therefore, we need to train a scene-dependent implicit function, denoted as

$$F : \mathbb{R}^3 \to \mathbb{R}^n, \tag{1}$$

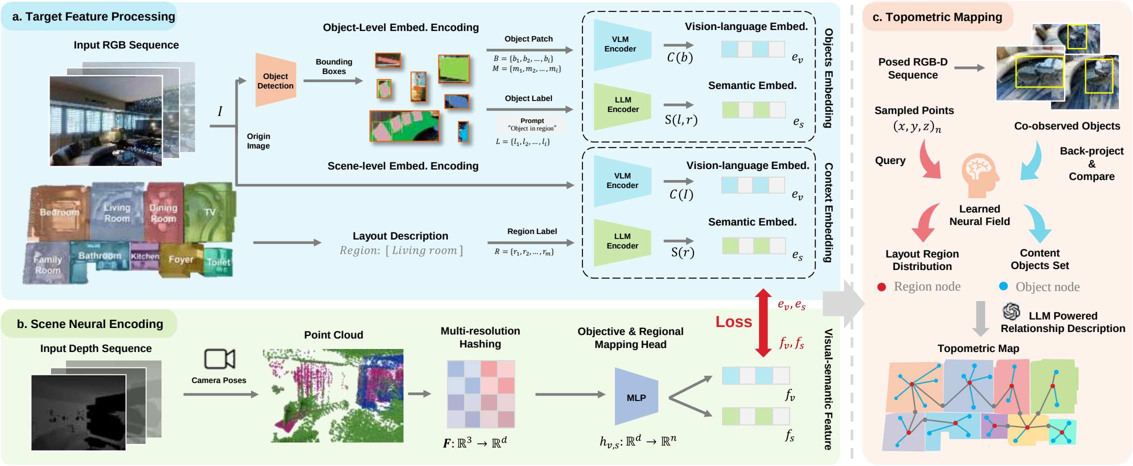

Fig. 2: **Pipeline of the Topo-Field. (a)** The ground truth generation of layout-object-position vision-language and semantic embeddings for weakly-supervising. **(b)** The neural implicit network mapping 3D positions to target feature space. A contrastive loss is optimized against each other. **(c)** Topometric mapping process with trained neural field.

where for any 3D point $P$ in space, $F(P)$ is supposed to match with $\mathcal{E}\{(e_v, e_s)\} \in \mathbb{R}^n$, representing the layout-object-position associated embedding of that point $e_v$ and $e_s$ are vision-language embedding and semantic embedding of image point where $P$ is back-projected from. CLIP [23] image encoder is introduced to encode $e_v$ integrating the vision and language feature space. Besides, the Sentence-BERT [24] feature is also introduced to encode $e_s$ in this work. Because intuitively, unlike objects that can have similar appearances within a certain category, region information often lacks specific visual appearances and is closely related to semantic representations like the integration purpose of the scene and object semantics. Models trained on large-scale question-answering datasets can aid in understanding the semantic relationships between regions and objects. Target feature processing and training strategy to match the embeddings to targets are described in Section IV-A and IV-D. Applications utilizing the learned field are discussed in Section IV-C.1.

Based on the trained $F$, we aim to build a topometric map denoted as

$$G = (V, E), \tag{2}$$

where vertices $V$ include object vertices $\mathbf{v}_o$ and region vertices $\mathbf{v}_r$ and edges $E$ include edges between objects $\mathbf{e}_{o-o}$, edges between regions $\mathbf{e}_{r-r}$, and edges between object and region $\mathbf{e}_{o-r}$. The topological map architecture and construction pipeline are described in Section IV-C.2.

## IV. Method

### A. Target Feature Processing

RGB-D image sequences with poses are accepted as input to get the target layout-object-position features for training $F$. For pure RGB image sequences, depth point clouds and camera poses can also be estimated through methods like COLMAP [30] or simultaneous localization and mapping (SLAM). The only employed GT annotation is the layout distribution of environment where the region of each 3D point $P$ is denoted as $r_P \in R = \{r_1, r_2, \dots, r_q\}$, where $q$ is the number of regions. Such information is available in datasets like Matterport3D [31]. However, in fact, partitioning the buildings needs little human labor, where in most human-made buildings spatial layouts are easily available divided by straight walls. As in our practice, region annotation of a house with 8 rooms only takes 3 min by drawing lines from top-down view according to walls to form a rule to separate $(x, y)$ coordinates, bounding 3D points to different regions.

For each image $I$, we employ Detic [32] $D$ to generate object instance patches with number $i$, including bounding-boxes $B = \{b_1, b_2, \dots, b_i\}$, masks $M = \{m_1, m_2, \dots, m_i\}$, and labels $L = \{l_1, l_2, \dots, l_i\}$.

For object pixels $p_o$ in instance mask $j$, CLIP [23] $C$ is employed to compute per-pixel features in mask $b_j$ and Sentence-BERT [24] $S$ is employed to process the semantic feature of $l_j$, prompted in the form of "$l_j$ *in* $r_{p_o}$". Given the related region $r_{p_o}$ of $p_o$, embedding of $p_o$ can be denoted as $e_{p_o} = \{C(b_j), S(l_j, r_{p_o})\}$.

What's more, the background appearance is also considered which we proposed to include context information for region layout. For background pixels $p_b$ out of masks, per-pixel feature of the whole image $I$ is encoded. Its related region $r_{p_b} \in R = \{r_1, r_2, \dots, r_m\}$ is regarded as the text label and embedding of $p_b$ can be calculated as $e_{p_b} = \{C(I), S(r_{p_b})\}$.

Then, pixel-wise embeddings are back-projected to 3D space based on depth and pose and averagely counted to form a distilled 3D feature point cloud. Consequently, the target feature space $\mathcal{E}\{(e_v, e_s)\}$ consists of object and layout features, where $(e_v, e_s)$ directs from $\{e_{p_o}, e_{p_b}\}_{p_o, p_b \in P}$. The pipeline is shown in Fig. 2

Compared with previous implicit neural field methods, $(e_v, e_s)$ includes (1) separately encoded vision-language and semantic information by supervising embeddings from object and background pixels. (2) region information consisted of vision-language embeddings from per-pixel image encoding

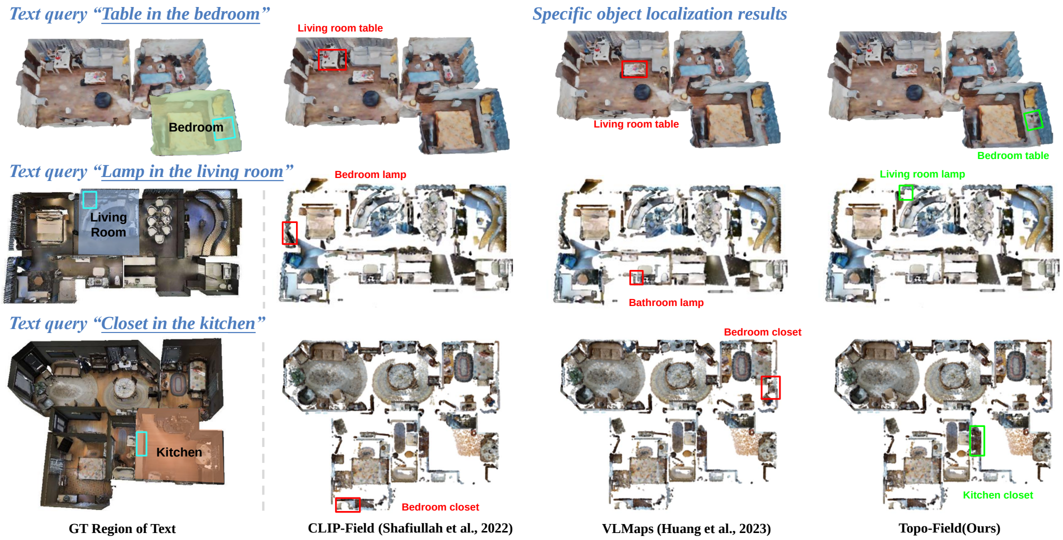
Text query "Table in the bedroom"
Bedroom
Living room table
Specific object localization results
Living room table
Bedroom table
Text query "Lamp in the living room"
Living Room
Bedroom lamp
Bathroom lamp
Living room lamp
Text query "Closet in the kitchen"
Kitchen
Bedroom closet
Bedroom closet
Kitchen closet
GT Region of Text
CLIP-Field (Shafiullah et al., 2022)
VLMaps (Huang et al., 2023)
Topo-Field(Ours)

Fig. 3: **Qualitative comparison of text query localization** results among state-of-the-art methods and our method with text input in the form of "*object in the region*". Blue box shows the ground truth bounding box of object. Red box means miss-predicted box, while green box means the correctly predicted results.

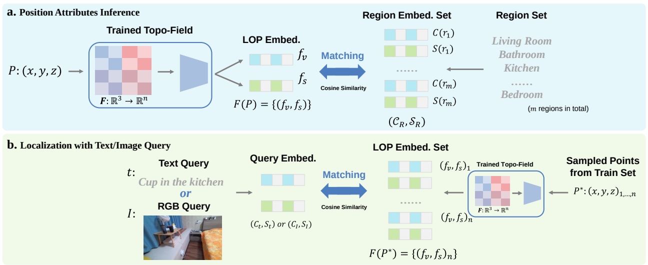

Fig. 4: **Capabilities of the learned neural field**. **(a)** The attributes inference using position input. **(b)** The LOP association helped localization of text and image queries.

| Methods | Scene1 | | Scene2 | | Scene3 | | Scene4 | |
|---|---|---|---|---|---|---|---|---|
| | Dist. | Acc. | Dist. | Acc. | Dist. | Acc. | Dist. | Acc. |
| CLIP-Field(2022) | 2.97 | 0.24 | 3.35 | 0.21 | 2.98 | 0.20 | 3.06 | 0.17 |
| VLMaps(2023) | 2.78 | 0.28 | 3.63 | 0.16 | 3.05 | 0.24 | 3.12 | 0.12 |
| LERF(2023) | 2.86 | 0.32 | 2.82 | 0.11 | 3.49 | 0.17 | 3.04 | 0.20 |
| Topo-Field | **0.92** | **0.85** | **0.86** | **0.84** | **0.36** | **0.95** | **0.27** | **0.97** |
| Text queries | 100 | | 100 | | 60 | | 60 | |

TABLE I: **Quantitative comparison of text query localization** results on different scenes from the Matterport3D dataset. The average distance (m) from the target to the localized point cloud and the accuracy evaluating whether predicted positions are in the correct region are used as metrics.

and semantic embeddings from region text labels. (3) context included object label in the form of "$l_p$ *in* $r_p$", where $l_p$ and $r_p$ is object label and region label at point $p$ (e.g., cup in the kitchen). Ablation studies of these improvements are conducted in Section IV with more details.

### B. Scene Neural Encoding

Our proposed Topo-Field involves an implicit mapping function to encode the 3D position into a spatial vector representation $g : \mathbb{R}^3 \rightarrow \mathbb{R}^d$ and separate heads $h : \mathbb{R}^d \rightarrow \mathbb{R}^n$ processing encodings to match the target feature space $\mathcal{E}\{(e_v, e_s)\}$. To select an appropriate implicit function, considering that the target feature space includes object-level local features and layout-level region feature representation, we employ the Multi-scale Hash Encoding (MHE) introduced in Instant-NGP [33] as $g$ with $d = 144$. The feature pyramid structure used in MHE allows for considering structural features ranging from coarse to fine in the spatial domain. Additionally, MHE has a faster training speed compared to traditional NeRF [7] network structures. For mapping the position encodings to the target feature space, we employ a unified and simple Multi-Layer Perceptron (MLP) network structure. It includes heads $h_v : \mathbb{R}^d \rightarrow f_v$ for obtaining vision-language features and $h_s : \mathbb{R}^d \rightarrow f_s$ for semantic features, which together form the high dimension embeddings $\{f_v, f_s\} \in \mathbb{R}^n$. The model is shown in Fig. 2.

In this way, given a posed RGB-D image, the target feature of each pixel is processed as mentioned in Section IV-A denoted as $\mathcal{E}\{(e_v, e_s)\}$. At the same time the related pixel in depth image is back-projected into 3D space according to depth and pose value and processed by the above mentioned $g, h$ to form $\{f_v, f_s\}$. A contrastive loss is conducted between $\{(e_v, e_s)\}$ and $\{f_v, f_s\}$ to train the neural representation. Training details are declared in Section IV-D.

### C. Topometric Mapping

With the function and feature representation mentioned above, we can integrate 3D positions with the object and region information and construct a topometric map. The topo map construction process is formed in a mapping and updating strategy, while the implicit neural representation is introduced and queried as scene knowledge in this process.

Detailed pipeline is introduced as follows.

*1) Knowledge from Learned Neural Field:* **Position Attributes Inference.** Using spatial 3D point $P$ as input, assuming a collection of space regions $R$ (e.g., "living room""bathroom""bedroom"...), we compute the vision-language features $\mathcal{C}_R = \{C(r_1), C(r_2), \dots, C(r_m)\}$ and semantic features $\mathcal{S}_R = \{S(r_1), S(r_2), \dots, S(r_m)\}$ using CLIP [23] encoder $C$ and Sentence-BERT [24] encoder $S$, where $m$ is the number of rooms. Then the cosine similarity between $F(P) = \{(f_v, f_s)\}_P$ and $\{\mathcal{C}_R, \mathcal{S}_R\}$ is calculated to find the most likely region to which $P$ belongs. The inference process is shown in Fig. 4 (a). Similarly, the object information of $P$ can be inferred with the same approach replacing the region set $R$ with object set $O$.

**Localization with Text/Image Query.** For natural language text input $t$ (e.g., "cup in the bedroom"), most existing robotic scene representations struggle to locate specific objects of interest (e.g., differentiating between cups in the living room and the bedroom). However, with our proposed Topo-Field that includes region information, we can calculate the cosine similarity between $\{\mathcal{C}_t, \mathcal{S}_t\}$ and the embeddings $F(P^*) = \{(f_v, f_s)\}_{P^*}$ to find the most likely position of queries, where $P^*$ are sampled from 3D points set to train $F$. As for image input $I$, we can calculate the cosine similarity of $\{\mathcal{C}_I, \mathcal{S}_I\}$ with $F(P^*) = \{(f_v, f_s)\}_{P^*}$ in the same way to find the 3D points set with highest similarity. Localization process of text query and image query is shown in Fig. 4.

*2) Topometric Map Construction:* As defined in Section III, topometric map $G = (V, E)$ consists of vertices and edges. We define a vertice $\mathbf{v}$ { id, node_type, class, bounding_box, caption} and edge $\mathbf{e}$ { id, edge_type, start_node, end_node, relationship, caption }. Mimicking the mental representation of cognitive maps, we construct the topometric map in a **mapping and updating** strategy based on the learned Topo-Field $F$.

**Mapping**. we first averagely sample $k$ points $P_{1,\dots,k}$ in the environment (each grid of $0.5m \times 0.5m$ with a point in our practice) and infer their related regions according to Section IV-C.1. Supposing there are $m$ regions in total $r_{1,\dots,m}$, we calculate the extent of each region in the bounding-box format according to positions of points within the same region. The topo map region vertice set is then initialized as $\mathbf{v}_r = \{\mathbf{v}_{r_1}, \mathbf{v}_{r_2}, \dots, \mathbf{v}_{r_m}\}$. For each $\mathbf{v}$, {id} is set, {node_type} is {region}, {class} and {caption} is set according to the inferred region label, and {bounding_box} is set to the bound of coordinates. On the other hand, while employing Detic [32] to detect object instances as mentioned in Section IV-A, instances with high confidence (more than 60% in our practice) are recorded as object vertices candidates. For each $\mathbf{v}$, {node_type} is {object}, {class} and {caption} is set according to the prediction result, and {bounding_box} is set according to the back-projected masked pixels similarly. With the mapped nodes, we leverage LLM to describe the layouts with connectivity, distances, and relationships of regions and objects in JSON format based on the vertices' attributes and poses. During this process, edges are built among vertices. For object-object edge $\mathbf{e}_{o-o}$, we follow [28] which mainly consider bounding-box overlap. For object-region edge $\mathbf{e}_{o-r}$, we consider an object belongs to the region if the object b-box is in the region b-box and filter the unreasonable relation noise powered by LLM (e.g., it's almost impossible that a bike is in bedroom). For region relationships, the adjacency and position relationship of region b-box is considered. Examples of LLM prompts to build relationships and JSONs are listed in appendix for reference. Fig. 2 shows the pipeline of metric-topological map construction.

**Updating**. RGB-D image sequence for training $F$ or a newly captured sequence can be used for constructed topometric map fine-tuning. For object vertices, if an object is detected by more than 3 frames in sequence, the object b-box will be compared with the constructed vertices. A new vertice will be added if no vertice corresponds to it with the above-mentioned process. For region vertices, we calculate embeddings $F(p_I)$ of sampled back-projected pixels $p_I$ in each image $I$. $F(p_I)$ will be matched with the constructed region set $r_{1,\dots,m}$, and extent of a region $r$ will be updated if $F(p_I)$ matches $\{\mathcal{C}_r, \mathcal{S}_r\}$ and $p_I$ exceeds the {bounding_box} extent of vertice $\mathbf{v}_r$. LLM to update edges will be called each 50 frames.

### *D. Training*

The pipeline of ground truth data generation is described in Section IV-A to train $F$. To fit the implicit representation introduced in Section IV-B to the target feature space, we design the loss function through a contrastive approach. For the vision-language feature optimization, the tempered similarity matrix on point $P$ is denoted as

$$\text{Sim}_v = \tau \{f_v\}_P \{e_v\}_P, \tag{3}$$

where $\tau$ is the temperature term, $\{f_v\}_P$ and $\{e_v\}_P$ is the calculated implicit representation feature and target embedding according to $P$. Using cross-entropy loss, the vision-language loss can be calculated as

$$\mathcal{L}_v = -exp(-\text{dist}_P)(H(\text{Sim}_v) + H(\text{Sim}_v{}^T)), \tag{4}$$

where $\text{dist}_P$ is the distance from $P$ to camera, and $H$ is the cross-entropy function. For the semantic loss, similarity on points $P$ can be calculated as

$$\text{Sim}_s = \tau \{f_s\}_P \{e_s\}_P. \tag{5}$$

Similarly, semantic loss can be denoted as

$$\mathcal{L}_s = -\text{conf}(H(\text{Sim}_s) + H(\text{Sim}_s{}^T)), \tag{6}$$

where $conf$ is the prediction confidence from the detection model. The total loss is computed by:

$$\mathcal{L} = \mathcal{L}_v + \mathcal{L}_s. \tag{7}$$

In our experiments, an NVIDIA RTX3090 GPU is utilized and the batch size is set to 12544 to maximize the capability of our VRAM. As model instances, CLIP with SwinB is employed in Detic [32], CLIP [23] encoder is ViT-B/32 and Sentence-BERT [24] encoder is all-mpnet-base-v2. The

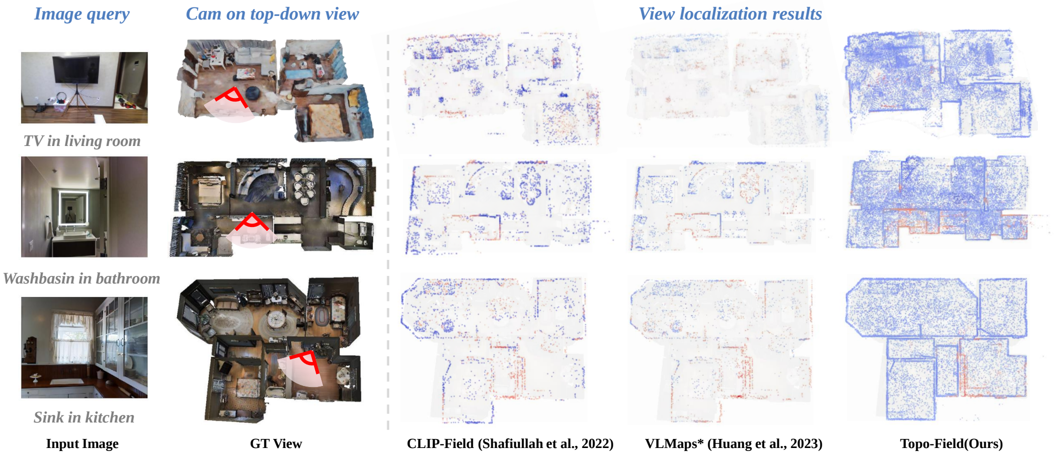

Fig. 5: **Qualitative comparison of image query localization** results in heatmaps form among state-of-the-art methods and our method with image input. Our approach localizes the position of queried image in an exact smaller range.

| Methods | Scene1 | Scene2 | Scene3 | Scene4 | Scene5 | Scene6 | Scene7 | Scene8 | Scene9 | Scene10 |
|---|---|---|---|---|---|---|---|---|---|---|
| CLIP-Field(2022) | 0.242 | 0.165 | 0.130 | 0.142 | 0.127 | 0.138 | 0.227 | 0.200 | 0.102 | 0.060 |
| VLMaps(2023) | 0.177 | 0.194 | 0.127 | 0.098 | 0.148 | 0.187 | 0.199 | 0.221 | 0.092 | 0.087 |
| LERF(2023) | 0.268 | 0.189 | 0.165 | 0.153 | 0.136 | 0.169 | 0.216 | 0.252 | 0.110 | 0.091 |
| RegionPLC(2023) | 0.290 | 0.202 | 0.173 | 0.168 | 0.152 | 0.154 | 0.243 | 0.248 | 0.086 | 0.088 |
| Topo-Field | **0.886** | **0.900** | **0.884** | **0.894** | **0.872** | **0.858** | **0.901** | **0.897** | **0.821** | **0.839** |
| Position Samples | 169k | 185k | 111k | 112k | 106k | 176k | 130k | 121k | 205k | 211k |

TABLE II: **Comparison of position attributes inference results** on the test set of different scenes from the Matterport3D dataset. The average region prediction accuracy of sampled 3D points is used as metric.

| Methods | Scene1 | Scene2 | Scene3 | Scene4 |
|---|---|---|---|---|
| CLIP-Field(2022) | 2.541 | 2.748 | 2.922 | 2.651 |
| VLMaps*(2023) | 2.112 | 1.894 | 1.181 | 1.595 |
| LERF(2023) | 1.276 | 1.175 | 1.148 | 1.129 |
| Topo-Field | **0.742** | **0.830** | **0.374** | **0.327** |

TABLE III: **Quantitative comparison of image query localization** results with other methods. The similarity weighted average distance (m) between the target view point cloud and the predicted point cloud is evaluated. VLMaps* is a self-implemented version with image localization ability.

MHE has 18 levels of grids and the dimension of each grid is 8, with $log_2$ hash map size of 20 and only 1 hidden MLP layer of size 600. We train the neural implicit network for 100 epochs with optimizer $Adam$, employing a decayed learning rate of $1e-4$ and $3e-3$ decay rate. Each epoch contains $3e6$ samples. Codes and scripts are released in supplementary for reproducibility.

## V. Experimental Results

Our experiments are conducted on real-world datasets to validate the established layout-object-position association. The data environment is of single-floor residential buildings with multiple rooms which is the common working scenario of household robots widely studied in this field. We employed Matterport3D [31] as well as apartment environment [34] dataset to demonstrate that our approach can be generalized in diverse scenarios.

### A. Position Attributes Inference

To demonstrate the built LOP association integrates positions with layout features, we designed experiments that accept 3D positions as input to infer the region information. For quantitative evaluation, we divided the RGB-D sequences into training and testing sets. The Topo-Field is trained according to Section IV-D on the training set and tested in the test set. As the region inference task can be treated as a multi-class classification task for each input, the accuracy, precision, and F1-score are used as metrics. Tab. II shows the region inference results on 10 real-world scenes in Matterport 3D [31] with different scales and layouts indicating the average accuracy exceeds 85%.

### B. Localization with Prompt Queries

**Localization with Text Queries:** For objects of the same category in different regions, we input the textual description of the target object in the form of "object in the region" and infer the specific location of the target, comparing the results with the predictions from current state-of-the-art visual-language algorithms. Fig. 3 demonstrates the advancements of Topo-Field in object localization tasks involving region information, which allows for the localization of specific target objects based on the description and features of the region, while other methods confuse objects from different regions. Tab. I shows the quantitative results on 4 scenes of different layouts compared to other methods with an average accuracy of more than 88% and less distance from targets.

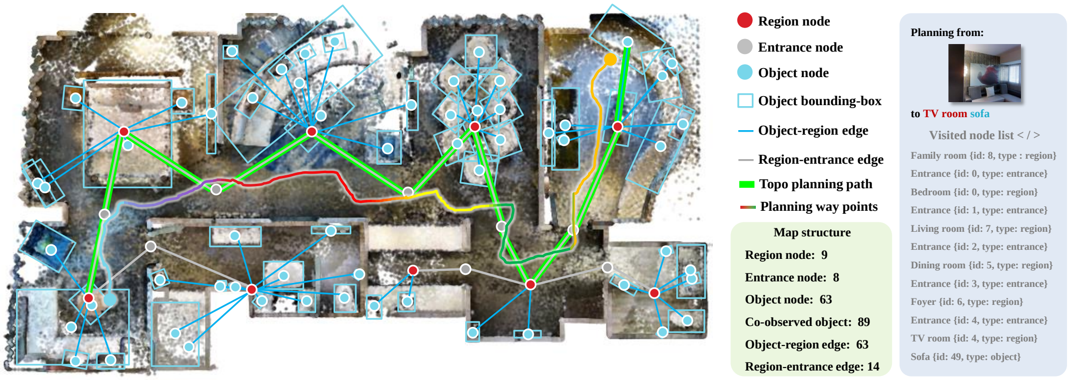
Region node
Entrance node
Object node
Object bounding-box
Object-region edge
Region-entrance edge
Topo planning path
Planning way points
Map structure
Region node: 9
Entrance node: 8
Object node: 63
Co-observed object: 89
Object-region edge: 63
Region-entrance edge: 14
Planning from:
to TV room sofa
Visited node list < / >
Family room {id: 8, type : region}
Entrance {id: 0, type: entrance}
Bedroom {id: 0, type: region}
Entrance {id: 1, type: entrance}
Living room {id: 7, type: region}
Entrance {id: 2, type: entrance}
Dining room {id: 5, type: region}
Entrance {id: 3, type: entrance}
Foyer {id: 6, type: region}
Entrance {id: 4, type: entrance}
TV room {id: 4, type: region}
Sofa {id: 49, type: object}

Fig. 6: **Topometric map construction example.** The topometric map is represented as a graph from a top-down view according to the position of nodes. Map structure shows number of nodes and edges. A planning path from a seen view to target is shown as an example employing topometric map, the path is highlighted in green showing the related nodes and edges. Visited nodes are listed on the right. The line with gradient colors represents the waypoints based on the planning results while different colors represent different predicted regions of waypoints.

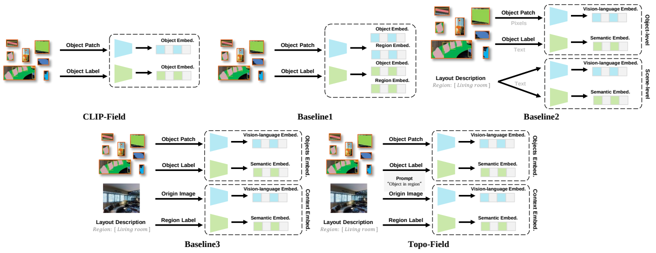

Fig. 7: Ablation of our LOP information encoding and feature fusion strategy for target features.

For the metrics, the average distance ($m$) of predicted point cloud and ground truth point cloud is evaluated, together with counting whether the center of predicted points is in the correct room. Ground truth comes from the Matterport3D provided object instance labels. More results can be seen in the appendix.

**Localization with Image Queries:** To validate the help of region information in the image view localization task. We localize the images from the test set in the trained Topo-Field. Selected views include representative objects of the scene (e.g., TV in the living room) and views with similar-looking objects or context (e.g., bathroom washbasin and kitchen sink) which is challenging. The localization results are shown in Fig. 5 in the form of heatmaps and Tab. III shows the quantitative results which evaluates the weighted average distance of the target view and localized point cloud among all samples in a scene, using similarity as weight. VLMaps* is a self-implemented version, because origin VLMaps [12] does not implement the image localization task. To align with CLIP-Field [11] and our work, the LSeg [35] used in VLMap [12] is replaced by CLIP [23]. The results show that Topo-Field constrains the localization results to a smaller range in the exact region. We sampled more than 40 images on each scene from Apartment [34] and Matterport3D [31] dataset. By drawing the predicted camera view on the top-down view, we estimated the localization precision and found that most views can be ranged into a specific view on the target field of view, while other methods struggle to get precise results.

| Methods | Scene1 | Scene2 | Scene3 | Scene4 |
|---|---|---|---|---|
| CLIP-Field | 0.242 | 0.165 | 0.130 | 0.142 |
| Baseline1 | 0.852 | 0.891 | 0.863 | 0.874 |
| Baseline2 | 0.865 | 0.887 | 0.872 | 0.879 |
| Baseline3 | 0.872 | 0.891 | 0.875 | 0.886 |
| Topo-Field | **0.886** | **0.900** | **0.884** | **0.894** |

TABLE IV: **Ablation of target feature processing pipeline** of the neural field construction. The average region prediction accuracy of sampled points from different scenes on the Matterport3D dataset is used as the metric.

### *C. Topometric Map Construction*

Fig. 6 shows an example of the built topometric map. Layout region nodes, object nodes with bounding boxes, and entrance nodes connecting regions are shown with edges representing relationships. A planned navigable path is shown in the graph from an observed view in family room to the TV room sofa in green. The path planning A* algorithm is employed to explore the topological structure to generate waypoints between nodes, and the waypoints are generated with the planning API in Habitat Simulator [36] and shown in a line with gradient colors, while different colors indicate different predicted regions of the waypoints.

### *D. Ablation Study*

Fig. 7 and Tab IV. show the ablation of our neural field LOP encoding strategy and feature fusion where: 1) CLIP-Field [11] means the origin feature encoding strategy that doesn't explicitly consider the layout features. 2) Baseline1 is our first crude approach that directly supervises the learned embedding from the encoded objects with region semantics. 3) Baseline2 encodes the region description to the target vision-language and semantic feature space for supervision. 4) Baseline3 takes the background pixels into account with the region labels. 5) Topo-Field further considers the context of the layout when supervising the object label semantics.

These four main versions of our numerous iterations of trying are listed as examples to show our work on the neural field encoding of LOP association.

## VI. Conclusion and Limitations

We propose a brain-inspired Topo-Field, which integrates Layout-Object-Position (LOP) associations into a neural field and constructs a topometric map from the learned field for hierarchical robotic scene understanding. However, there are some limitations: 1) Querying and path planning are currently implemented using traditional methods (e.g. A*). Future work will explore using LLMs for more advanced path planning. 2) Real-world deployment on mobile robots for long-term navigation is needed. 3) Future research will focus on updating and editing the topometric map to accommodate environmental changes.

## References

[1] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Transactions on robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.

[2] F. Blochliger, M. Fehr, M. Dymczyk, T. Schneider, and R. Siegwart, "Topomap: Topological mapping and navigation based on visual slam maps," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 3818–3825.

[3] S. Ullman, "The interpretation of structure from motion," *Proceedings of the Royal Society of London. Series B. Biological Sciences*, vol. 203, no. 1153, pp. 405–426, 1979.

[4] C. Forster, M. Pizzoli, and D. Scaramuzza, "Svo: Fast semi-direct monocular visual odometry," in *2014 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2014, pp. 15–22.

[5] A. Dai, M. Nießner, M. Zollhöfer, S. Izadi, and C. Theobalt, "Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface reintegration," *ACM Transactions on Graphics (ToG)*, vol. 36, no. 4, p. 1, 2017.

[6] C. Tang and P. Tan, "Ba-net: Dense bundle adjustment network," *arXiv preprint arXiv:1806.04807*, 2018.

[7] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing scenes as neural radiance fields for view synthesis," in *The European Conference on Computer Vision (ECCV)*, 2020.

[8] S. Zhi, T. Laidlow, S. Leutenegger, and A. Davison, "In-place scene labelling and understanding with implicit scene representation," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021.

[9] Z. Fan, P. Wang, Y. Jiang, X. Gong, D. Xu, and Z. Wang, "Nerf-sos: Any-view self-supervised object segmentation on complex scenes," 2022.

[10] C. Xie, K. Park, R. Martin-Brualla, and M. Brown, "Fig-nerf: Figure-ground neural radiance fields for 3d object category modelling," in *International Conference on 3D Vision (3DV)*, 2021.

[11] N. M. M. Shafiullah, C. Paxton, L. Pinto, S. Chintala, and A. Szlam, "Clip-fields: Weakly supervised semantic fields for robotic memory," *arXiv preprint arXiv:2210.05663*, 2022.

[12] C. Huang, O. Mees, A. Zeng, and W. Burgard, "Visual language maps for robot navigation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 10 608–10 615.

[13] J. Kerr, C. M. Kim, K. Goldberg, A. Kanazawa, and M. Tancik, "Lerf: Language embedded radiance fields," in *International Conference on Computer Vision (ICCV)*, 2023.

[14] C. Gomez, M. Fehr, A. Millane, A. C. Hernandez, J. Nieto, R. Barber, and R. Siegwart, "Hybrid topological and 3d dense mapping through autonomous exploration for large indoor environments," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 9673–9679.

[15] E. C. Tolman, "Cognitive maps in rats and men." *Psychological review*, vol. 55, no. 4, p. 189, 1948.

[16] J. O'Keefe and J. Dostrovsky, "The hippocampus as a spatial map: preliminary evidence from unit activity in the freely-moving rat." *Brain research*, 1971.

[17] E. T. Rolls, A. Treves, R. G. Robertson, P. Georges-François, and S. Panzeri, "Information about spatial view in an ensemble of primate hippocampal cells," *Journal of Neurophysiology*, vol. 79, no. 4, pp. 1797–1813, 1998.

[18] P. A. LaChance, T. P. Todd, and J. S. Taube, "A sense of space in postrhinal cortex," *Science*, vol. 365, no. 6449, p. eaax4192, 2019.

[19] T. Zeng, B. Si, and J. Feng, "A theory of geometry representations for spatial navigation," *Progress in Neurobiology*, vol. 211, p. 102228, 2022.

[20] J. Yang, R. Ding, Z. Wang, and X. Qi, "Regionplc: Regional point-language contrastive learning for open-world 3d scene understanding," *arXiv preprint arXiv:2304.00962*, 2023.

[21] D. Maggio, Y. Chang, N. Hughes, M. Trang, D. Griffith, C. Dougherty, E. Cristofalo, L. Schmid, and L. Carlone, "Clio: Real-time task-driven open-set 3d scene graphs," *IEEE Robotics and Automation Letters*, vol. 9, no. 10, pp. 8921–8928, 2024.

[22] A. Werby, C. Huang, M. Büchner, A. Valada, and W. Burgard, "Hierarchical open-vocabulary 3d scene graphs for language-grounded robot navigation," *Robotics: Science and Systems*, 2024.

[23] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PMLR, 2021, pp. 8748–8763.

[24] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 3982–3992.

[25] Q. Zhang, *Autonomous indoor exploration and mapping using hybrid metric/topological maps*. McGill University (Canada), 2015.

[26] Q. Zhang, I. Rekleitis, and G. Dudek, "Uncertainty reduction via heuristic search planning on hybrid metric/topological map," in *2015 12th Conference on Computer and Robot Vision*. IEEE, 2015, pp. 222–229.

[27] L. Garrote, C. Premebida, D. Silva, and U. J. Nunes, "Hmaps-hybrid height-voxel maps for environment representation," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1197–1203.

[28] Q. Gu, A. Kuwajerwala, S. Morin, K. M. Jatavallabhula, B. Sen, A. Agarwal, C. Rivera, W. Paul, K. Ellis, R. Chellappa, *et al.*, "Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 5021–5028.

[29] Z. He, H. Sun, J. Hou, Y. Ha, and S. Schwertfeger, "Hierarchical topometric representation of 3d robotic maps," *Autonomous Robots*, vol. 45, no. 5, pp. 755–771, 2021.

[30] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[31] A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niessner, M. Savva, S. Song, A. Zeng, and Y. Zhang, "Matterport3d: Learning from rgb-d data in indoor environments," *International Conference on 3D Vision (3DV)*, 2017.

[32] X. Zhou, R. Girdhar, A. Joulin, P. Krähenbühl, and I. Misra, "Detecting twenty-thousand classes using image-level supervision," in *ECCV*, 2022.

[33] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *ACM transactions on graphics (TOG)*, vol. 41, no. 4, pp. 1–15, 2022.

[34] Z. Zhu, S. Peng, V. Larsson, W. Xu, H. Bao, Z. Cui, M. R. Oswald, and M. Pollefeys, "Nice-slam: Neural implicit scalable encoding for slam," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022.

[35] B. Li, K. Q. Weinberger, S. Belongie, V. Koltun, and R. Ranftl, "Language-driven semantic segmentation," in *International Conference on Learning Representations*, 2022.

[36] M. Savva, A. Kadian, O. Maksymets, Y. Zhao, E. Wijmans, B. Jain, J. Straub, J. Liu, V. Koltun, J. Malik, D. Parikh, and D. Batra, "Habitat: A Platform for Embodied AI Research," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.