LOP-Field: Brain-inspired Layout-Object-Position Fields for Robotic Scene Understanding

Jiawei Hou
School of Computer Science
Fudan University
Shanghai, China.
jwhou23@fudan.edu.cn

Wenhao Guan
School of Computer Science
Fudan University
Shanghai, China.
whguan21@fudan.edu.cn

Xiangyang Xue
School of Computer Science
Fudan University
Shanghai, China.
xyxue@fudan.edu.cn

Taiping Zeng
Institute of Science and Technology for
Brain-Inspired Intelligence
Fudan University
Shanghai, China.
zengtaiping@fudan.edu.cn

Abstract

Spatial cognition empowers animals with remarkably efficient navigation abilities, largely depending on the scene-level understanding of spatial environments. Recently, it has been found that a neural population in the postrhinal cortex of rat brains is more strongly tuned to the spatial layout rather than objects in a scene. Inspired by the representations of spatial layout in local scenes to encode different regions separately, we proposed LOP-Field that realizes the Layout-Object-Position(LOP) association to model the hierarchical representations for robotic scene understanding. Powered by foundation models and implicit scene representation, a neural field is implemented as a scene memory for robots, storing a queryable representation of scenes with position-wise, object-wise, and layout-wise information. To validate the built LOP association, the model is tested to infer region information from 3D positions with quantitative metrics, achieving an average accuracy of more than 88%. It is also shown that the proposed method using region information can achieve improved object and view localization results with text and RGB input compared to state-of-the-art localization methods.

1 Introduction

Spatial cognition is a fundamental function that enables humans and animals to achieve long-term autonomy in their environment. A cognitive map is considered a mental representation of spatial information about the relative locations and attributes of phenomena in our everyday spatial environment [1]. Place cells encode the specific locations of rodents in the environment depending on both scene content and spatial layout [2]. Spatial view cells in the hippocampus become active when scene contents of the environment are in the animal's field of view [3]. Various boundary cells [4, 5, 6, 7] encode the allocentric scene border whatever the scene contents are. A particular population of neurons in the postrhinal cortex (POR) is more sensitive to the spatial layout of a local scene than the spatial contents [8]. A theory of geometry representations is proposed to describe various boundary-related cells and representations of POR in a unified framework. The predicted

Preprint. Under review.

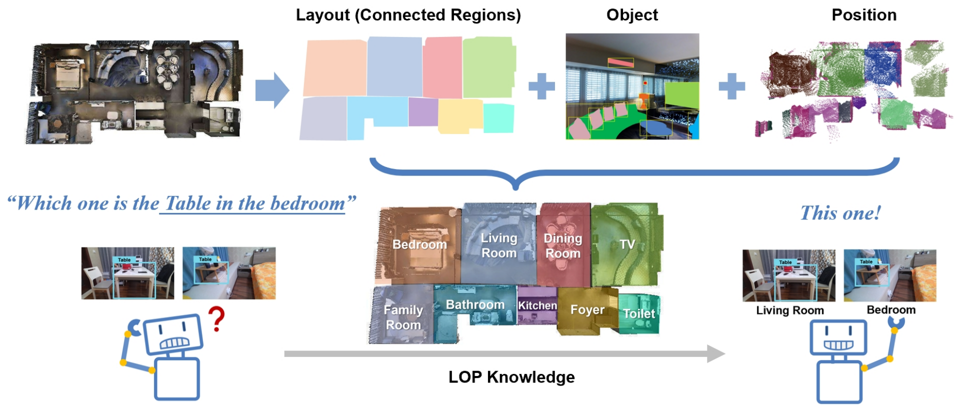


Figure 1: Dividing the scene information into layout, object, and position, and modeling them explicitly, layout-object-position association enables robots to address relative problems and realize a more comprehensive spatial cognition.

geometry cells by the theory are able to encode spatial layouts with different geometric structures, which helps to quickly form a high-level cognitive map representation [9]. The spatial layout, connected by regions, may play a vital role in spatial cognition, reasoning, and navigation, integrating with the purpose of the scene and object content semantics.

Inspired by neural representations of spatial layout, scene contents, and locations, spatial information can be categorized into (1) layout-level information, which includes the layout, region, and connectivity of spaces, (2) object-level information, which includes the attributes, appearance, and positions of various objects, and (3) position-level information, which includes the relative positions, associations, and modes of interaction among object parts. For example, people can distinguish different regions within their homes, recognizing the differences between the living room, bedroom, and kitchen. They can build knowledge and memories about the relationship between target objects (e.g., bed) and their corresponding regions (e.g., bedroom), and they can distinguish similar objects within different regions (e.g., a cup in the living room versus a cup in the kitchen). Similarly, if a robot could understand the relationships between spatial regions as humans do, it would be able to perform tasks such as spatial reasoning and layout-object associations. Fig. 1 shows that with layout-object-position association, robots could have enhanced spatial cognition and understanding capabilities.

In robotics research on spatial scene understanding, current efforts have yielded impressive results in tasks such as 3D environment reconstruction[10, 11, 12, 13, 14, 15], object detection[16, 17, 18, 19, 20], and object segmentation[21, 22, 23, 24, 25, 26]. However, most of these works have focused on producing lifelike scene reconstructions and precise geometric and semantic information about objects, with relatively few studies addressing the modeling and recognition of spatial layouts, such as scene regions, and the association with spatial contents. The lack of layout information and scalable association in scenes hinders a robot's comprehensive understanding and makes it difficult to interpret related commands.

How to enable robots to learn about spatial regions and association with contents remains a challenging problem, however, recent advances in large foundation models offer potential solutions. Large foundation models trained on massive datasets across various scenes, such as vision-language models(VLMs) like CLIP[27] and large language models(LLMs) like Sentence-BERT[28], are believed to have the ability to reason with general knowledge and perform zero-shot inference on multiple tasks. Numerous studies leverage these models to process visual-textual features of scenes, establishing links between spatial coordinates and these features. For instance, works like CLIP-Fields[29] and VLMaps[30] establish mappings between spatial positions and object visual-language features, while GARField[31] proposes hierarchical segmentation and grouping, dividing scenes into different physical scales. These efforts facilitate linking object features to 3D positions, but most existing research does not establish region recognition and lacks the integration of layout-object-position information.

To effectively integrate spatial layouts, scene objects, and position information, we introduce the LOP-Field, which realizes the Layout-Object-Position (LOP) association to model the hierarchical representations for robotic scene understanding. It integrates the spatial layout connected by regions and object-level semantics with context on 3D positions. It is equipped with the ability to reason about the relationship between the regions of the scene and its content, thus enhancing the object-level 3D reasoning capabilities of the previous work. Such a neural field can serve as a scene memory for robots, storing a queryable representation of scenes with hierarchical LOP information. By inputting RGB-D sequences, the LOP-Field is optimized using a contrast loss between its predicted features and features from the VLM and LLM, resulting in little need for annotation. To validate the established LOP relationship, we conducted experiments on several multi-room apartment scenes. We evaluated the model's ability to infer region information from 3D positions, providing quantitative metrics. We also demonstrated improved object and view localization results using object-region relations with text and RGB inputs. These experiments conclusively prove that LOP-Field effectively associates information of layout and scene contents from different scales.

Our contributions can be listed as follows:

- Inspired by the recent significant findings in neuroscience, we propose a neural scene representation named LOP-Field that integrates spatial layouts, scene objects, and 3D positions for robotic scene understanding.
- By fusing the object information from detected objects and layout region information from background contexts, LOP-Field builds layout-object-position association in a neural scene representation to match the vision-language and semantic feature space of large foundation models with little need for annotation.
- Various experiments are conducted to validate the layout-object-position association. LOP-Field achieves an accuracy of more than 95% on region inference using 3D positions and we demonstrate the help of scalable information association in downstream object and image localization tasks.

2 Related Works

2.1 Spatial Understanding with Layout Information

Understanding the mechanisms of spatial cognition in humans has been a challenging and active areas of cognitive science, which also serves as an important reference for enabling robots with scene understanding. During decades of research, scientists have made great efforts to understand the mechanism of spatial cognition. A mental representation of spatial information is proposed to describe the relative locations and attributes of phenomena in our everyday spatial environment, called a cognitive map. Place cells, as the embodiment of the cognitive map, encode the specific locations of rodents in the environment depending on both scene content and spatial layout [2]. Scene content of the environment is represented by spatial view cells in the hippocampus, while Various boundary cells [4, 5, 6, 7] encode the allocentric scene boundary regardless of the scene content. Recently, Patrick et al.[8] showed that a population code in the POR is more strongly tuned to the spatial layout than to the content in a scene. The firing activities remain consistent even when the environmental content and lighting conditions change. 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 [9]. We propose that the spatial layout connected by regions, as a high-level abstract semantic feature, is closely related to the object contents and purposes of the scene, and therefore it can establish connections with object semantics more easily than other layout information such as area, volume, and boundary. However, the layout regions of the scene and their association with scene content have received little attention in current robotics research on scene understanding.

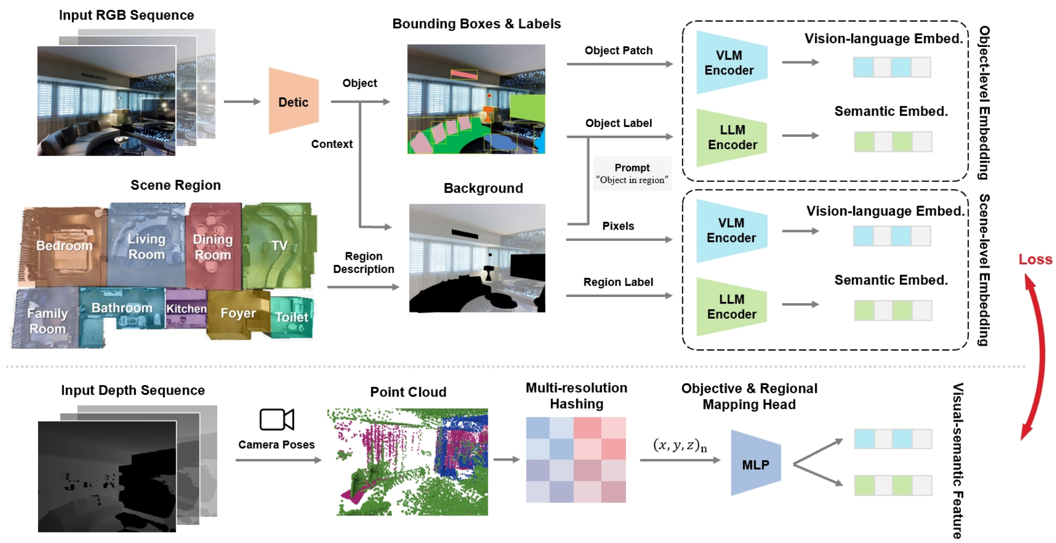
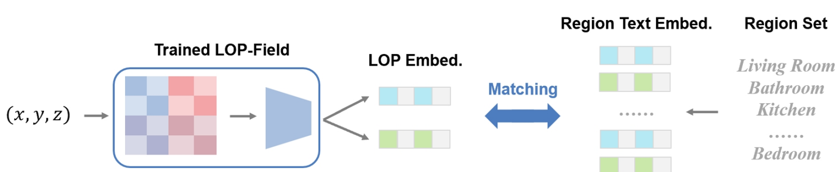
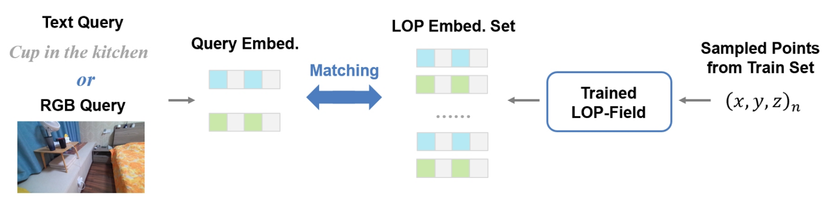


Figure 2: Pipeline of the target embedding processing and neural implicit rendering during training. Above is the ground truth generation of layout-object-position vision-language and semantic embeddings for weakly-supervising. Below is the neural implicit network mapping 3D positions to target feature space. A contrastive loss is optimized against each other.



(a) Region inference with position inputs



(b) Localization with text and image queries

Figure 3: The application examples pipeline of the LOP-Field. The region inference using position input is shown in (a). The LOP association helped localization of text and image query is shown in (b).

2.2 Neural Scene Representation

Traditional robotic scene recognition methods, such as multi-view synthesis[32, 33, 34, 35] and grid-based scene representation[36, 37], aim to reconstruct realistic new views and predict complete geometry and appearance information. However, approaches based on reprojection losses struggle to obtain sufficient constraints for optimization, and voxel-based representations face challenges when scaling to large-scale or high-resolution scenes. To address these issues, NeRF (Neural Radiance Fields)[38] introduced a novel approach that uses implicit neural fields to represent scene information. Subsequently, numerous efforts have been made to improve the training and inference speed of neural rendering fields[39, 40, 41, 42, 43], to adapt them to larger scenes[44, 45, 46], and to explore extended application scenarios[47, 48, 49]. A popular research direction is to integrate high-dimensional information, such as semantics, with NeRF to achieve a more comprehensive understanding of scenes and to address a wider range of downstream tasks[50, 51, 52]. However, training accurate

NeRF models that incorporate semantic information requires costly manual annotation and presents challenges in adapting and applying them to different scenes.

2.3 Large Foundation Model Powered Scene Understanding

Recently, several robotics works have utilized large foundation models trained on web-image data to assist in the understanding of semantic information in scenes. These works have demonstrated that models trained on web-image data can be used for self-supervised learning. Seal[53] employed a detection model trained on web-image data to establish the connection between semantics and 3D voxels. Cliport[54] and [55] achieved scene understanding using weakly supervised models trained on web-image data, leveraging techniques such as CLIP[27]. Huy Ha et al.[56] utilized CLIP features to annotate 3D points in space. CLIP-Fields[29] and VLMaps[30] directly train an implicit representation of a scene using visual-linguistic features, establishing correspondence between 3D spatial points and semantics. However, the semantic feature field learned in the above methods represents object semantics and does not include scene-level features. In contrast, in our work, CLIP[27] and Sentence-BERT[28] are used to generate vision-linguistic and semantic features for objects, spatial regions, and contexts, respectively. In addition to using object semantics generated by back-projection for 3D points in the scene, we annotate the belonging regions of 3D points based on spatial layout and regional division of scenes. Such annotations incur minimal cost but establish connections between the position of 3D points, object semantics, and scene regions.

3 Method

In this section, we first elaborate on the problem formulation of how to associate layout regions, object semantics, and 3D positions. We provide examples of the usage of LOP association, like region inference from positions and downstream localization tasks. Next, we present the process of generating the target features for training. Furthermore, we explain the structure of the employed implicit scene representation. Lastly, we describe the training procedures.

3.1 Problem Formulation

3.1.1 Foundation-Model-Based Neural Implicit Representation

Our goal is to learn an implicit representation of a scene by establishing associations between 3D positions and their corresponding layout regions and object features. Therefore, we need to design a scene-dependent implicit function, denoted as

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

where for any point P in space, $F(P)$ represents the layout-object-position associated features of that point. CLIP[27] is introduced as the VLM in this work to encode the object and region information, integrating the vision and language feature space. Besides, the Sentece-BERT[28] feature is also introduced 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. Consequently, $\mathbb{R}^n$ stands for embeddings:

$$\mathcal{E} = \{(e_v, e_s)\}$$

including vision-language embedding e_v and semantic embedding e_s in our approach. These predicted implicit representation outputs are targeted to match the features from the pre-trained CLIP[27] C and Sentence-BERT[28] S separately.

3.1.2 Target Feature Processing

To get the target layout-object-position features, RGB-D image sequences with poses are accepted as input, what's more, for pure RGB image sequences, depth point clouds and camera poses estimated through methods like COLMAP[57] or simultaneous localization and mapping(SLAM) can also be used. For each image I, we employ Detic[58] D as the detection model to generate bounding-box-constrained object patches $B = \{b_1, b_2, \ldots, b_i\}$ and labels $L = \{l_1, l_2, \ldots, l_i\}$, followed by CLIP[27] and Sentence-BERT[28] to process the vision-language and semantic features. Given the related region r_P and object instance o_P of point P, P can be labeled with $\{C(b_P), S(o_P, r_P)\}$, where the text prompt is formed as o_P *in* r_P. What's more, the background appearance is also considered which we proposed to include context information for region layout. For background pixel Q out of the object masks, its related region $r_Q \in R = \{r_1, r_2, \ldots, r_m\}$ is regarded as the text label and its label can be calculated as $\{C(Q), S(r_Q)\}$. To obtain region labels of image pixels, we back-project them to 3D space based on depth and simply consider the top-down view of the 3D point cloud. The space can be partitioned into different regions using walls as dividers. Consequently, the target feature space processed by foundation models can be denoted as

$$\mathcal{F} = \{(f_v, f_s)\},$$

where f_v is the visual-language feature from CLIP[27] and f_s is the semantic feature from Sentence-BERT[28]. The processing pipeline is shown in Fig. 2.

3.1.3 Layout-Object-Position Association

With the function and feature representation mentioned above, we can infer the region information and utilize it for various downstream tasks.

Region Inference. Using spatial 3D point P_i as input, assuming a collection of space regions R, we compute the vision-language features $\mathcal{C}_R = \{C(r_1), C(r_2), \ldots, C(r_m)\}$ and semantic features $\mathcal{S}_R = \{S(r_1), S(r_2), \ldots, S(r_m)\}$. Then the similarity between $\mathcal{E}_{P_i} = \{(e_v, e_s)\}$ and $\{\mathcal{C}_R, \mathcal{S}_R\}$ is calculated to find the most likely region to which P_i belongs. The inference process is shown in Fig. 3.

LOP Guided Object Localization. For text input t, such as "cup in the bedroom," most existing robotic scene representations struggle to locate specific objects of interest (differentiating between cups in the living room and the bedroom, for example). However, with our proposed LOP-Field that includes scene region information, we can calculate the similarity between $\{\mathcal{C}_t, \mathcal{S}_t\}$ and the embeddings $\mathcal{E}_{P^*}$ of the sampled point P^* set from the scene. Compared with previous object localization methods, $\mathcal{E}_{P^*} = \{(e_v, e_s)\}$ includes contexts between region layout and objects by considering the object information of detected objects and region information of the background appearance.

LOP Guided View Localization. Another common robotic application is to localize a captured image of the scene. Unlike previous methods that only encode the object semantics to find matches, LOP-Field introduces region features to constrain the prediction. For image input I, the similarity of $\{\mathcal{C}_I, \mathcal{S}_I\}$ with $\mathcal{E}_{P^*} = \{(e_v, e_s)\}$ is calculated. Compared to previous methods that only encode objects, the text label of object point P is formed as o_P *in* r_P(e.g., cup in the kitchen), and the background appearance with region label is also encoded. These all contribute to a more accurate localization of a specific image view. The localization of both text query and image query is shown in Fig. 3.

3.2 Model Architecture

Our proposed LOP-Field involves an implicit mapping function to encode the 3D positions and separate head processing encodings to match the target feature space. 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 Hierarchical Encoding (MHE) introduced in Instant-NGP[59]. 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[38] network structures. For mapping the position encodings to the target feature space, we employ a unified and simple Multi-Layer Perceptron(MLP)

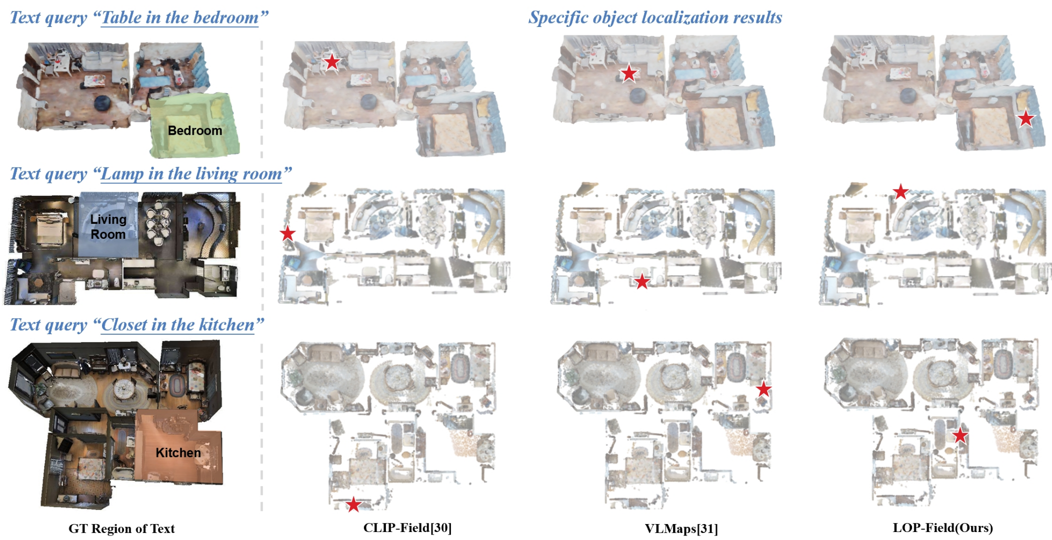


Figure 4: The object localization results among state-of-the-art methods and our method with text input in the form of *object in the region*. Red stars show the position of the found results of input texts.

network structure. It includes heads $head_v$ for obtaining vision-language features and $head_s$ for semantic features. The model for training is shown in Fig. 2.

3.3 Training

The pipeline of ground truth data generation is described in Section 3.1.2. To fit the multiple embeddings generated by the implicit representation introduced in Section 3.1.1 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

$$\text{Sim}_v = \tau e_v C(P),$$

where τ is the temperature term. Using cross-entropy loss, the vision-language loss can be calculated as

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

where dist_P is the distance from P to camera, and H is the cross-entropy function. For the semantic loss, similarity on object points P_o and background points P_b can be calculated as

$$\text{Sim}_s^{P_o} = \tau e_s S(o_{P_o}, r_{P_o}),\ \text{Sim}_s^{P_b} = \tau e_s S(r_{P_b}).$$

Similarly, semantic loss can be denoted as

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

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

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

In our experiments, an NVIDIA RTX3090 GPU is utilized and the batch size is set to 12544 to maximize the capability of our VRAM. The MHE has 18 levels of grids and the dimension of each grid is 8. We train the neural implicit network for 100 epochs with a decayed learning rate of $1e-4$. Each epoch contains $3e6$ samples.

4 Experimental Results

To validate the established layout-object-position relationship of LOP-Field, we designed the following experiments related to region information in scenes. Our experimental data consists of multi-room

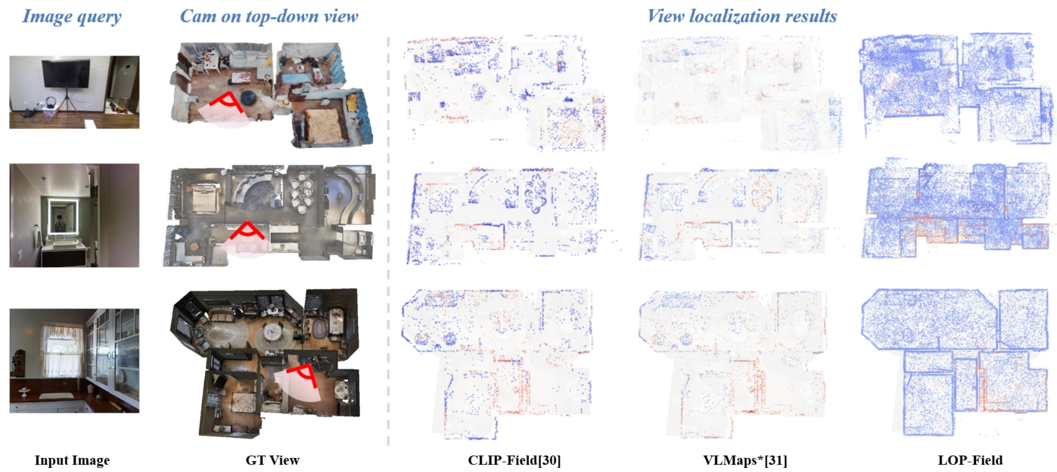


Figure 5: The image localization heatmaps among state-of-the-art methods and our method with text input in the form of *object in the region*. Red lines and the sector represent the field of view region.

Regions	Scene1			Scene2			Scene3			Scene4		
	Acc.	Pre.	F1	Acc.	Pre.	F1	Acc.	Pre.	F1	Acc.	Pre.	F1
Living Room	0.948	0.970	0.959	0.870	0.881	0.875	0.778	0.810	0.793	0.902	0.949	0.925
Bedroom	0.943	0.825	0.880	0.925	0.923	0.924	0.687	0.767	0.725	0.920	0.870	0.894
Bathroom	0.466	0.680	0.554	0.903	0.898	0.901	0.875	0.463	0.605	0.797	0.831	0.814
Dining Room	-	-	-	0.961	0.794	0.870	0.774	0.732	0.752	0.933	0.887	0.910
Lobby	0.681	0.941	0.790	0.853	0.951	0.899	0.978	0.510	0.671	0.855	0.698	0.769
Family Room	-	-	-	-	-	-	0.903	0.571	0.700	0.926	0.936	0.931
Kitchen	0.994	0.654	0.789	0.788	0.836	0.811	0.833	0.833	0.833	0.758	0.854	0.803
Office	-	-	-	0.969	0.848	0.905	-	-	-	0.953	0.883	0.917
Toilet	-	-	-	-	-	-	0.900	0.711	0.795	-	-	-
Avg. Acc./Samples	0.886 / 169k			0.900 / 185k			0.884 / 111k			0.894 / 112k		

Table 1: Region prediction results on the test set of different scenes from the Matterport3D[60] dataset. Accuracy, precision, and F1 score are used as metrics.

environment from Matterport3D[60] as well as apartment environment[48], which allows us to demonstrate that our approach can be generalized in diverse scenarios. The data environment is of single-floor residential buildings which is the common working scenario of household robots widely studied in this field.

4.1 Region Inference

To demonstrate the built LOP association integrates positions with layout, we designed experiments that accept 3D positions as input to infer the region information. For quantitative evaluation, we divided the RGB-D sequences of data into training and testing sets. The LOP-Field is trained according to Section 3.3 on the training set and tested with data from 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. 4.1 shows the region inference results. It can be seen that in multi-region environments with different scales and layouts, the average accuracy exceeds 88%. This experiment demonstrates that the implicit representation of the scene can successfully establish the connection between 3D points and their corresponding region features.

4.2 LOP Guided Localization

Text Input Object Localization: For objects of the same category existing in multiple 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 of current state-of-the-art visual-language algorithms. Fig. 4 demonstrates the advancements of LOP-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. We tested over 160 text queries on 4 scenes of Apartment[48] and Matterport3D[60] dataset. The accuracy of LOP-Field to localize the specific objects in the target regions exceeds 90%, while other methods have a significant fluctuation in accuracy. More results can be seen in the appendix.

Image Input View Localization: To validate the help of region information in the image view localization task. We localize the images from the test set in the trained LOP-Field. The localization results are shown in Fig. 5 in the form of heatmaps. VLMaps* is a self-implemented version, because origin VLMaps[30] does not implement the image localization task. To align with CLIP-Field[29] and our work, the LSeg[61] used in VLMap[30] is replaced by CLIP[27]. The results show that LOP-Field constrains the localization results to a smaller range in the exact region. We sampled more than 40 images on each of the 4 scenes from Apartment[48] and Matterport3D[60] dataset. By drawing the predicted camera view on the top-down view, we estimated the localization precision and found that almost all views can be ranged into a specific view on the target field of view, while other methods struggle to get precise results.

5 Conclusion and Limitations

Inspired by neural representations of spatial layout, scene contents, and locations, this paper proposed the LOP field, an implicit scene representation field that associates layout-object-position information, powered by foundation models for robotic scene understanding. Our experiments show that with the help of LOP association, region inference ability and better results in several downstream tasks are achieved. However, due to time and page constraints, this study explored only a very limited application of scalable associative information. In addition, the accuracy of the model decreases when distinguishing between regions with similar functions (such as living room, family room, and TV room). Furthermore, we currently lack a good method to model the confidence of the predicted results when presented with images that do not contain representative objects (with sufficient information to infer the region) or with degraded visual features.

We will investigate how the layout-object-position association information can be effectively used to perform challenging tasks that previously relied solely on the semantic information of the objects. Examples of such tasks include complex environment relocalization problems, reasoning about logical relationships between regions and objects, and efficient navigation between different regions. We believe that the integration of layout-object-position can significantly enhance a robot's spatial perception capabilities and encourage researchers to pay attention to the encoding of scene-specific information.

References

[1] Edward C Tolman. Cognitive maps in rats and men. *Psychological review*, 55(4):189, 1948.

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

[3] Edmund T Rolls, Alessandro Treves, Robert G Robertson, Pierre Georges-François, and Stefano Panzeri. Information about spatial view in an ensemble of primate hippocampal cells. *Journal of Neurophysiology*, 79(4):1797–1813, 1998.

[4] Trygve Solstad, Charlotte N. Boccara, Emilio Kropff, May-Britt Moser, and Edvard I. Moser. Representation of geometric borders in the entorhinal cortex. *Science*, 322(5909):1865–1868, 2008.

[5] Francesco Savelli, D. Yoganarasimha, and James J. Knierim. Influence of boundary removal on the spatial representations of the medial entorhinal cortex. *Hippocampus*, 18(12):1270–1282, 2008.

[6] Bruno Rivard, Yu Li, Pierre-Pascal Lenck-Santini, Bruno Poucet, and Robert U. Muller. Representation of Objects in Space by Two Classes of Hippocampal Pyramidal Cells . *Journal of General Physiology*, 124(1):9–25, 2004.

[7] Colin Lever, Stephen Burton, Ali Jeewajee, John O'Keefe, and Neil Burgess. Boundary vector cells in the subiculum of the hippocampal formation. *Journal of Neuroscience*, 29(31):9771–9777, 2009.

[8] Patrick A. LaChance, Travis P. Todd, and Jeffrey S. Taube. A sense of space in postrhinal cortex. *Science*, 365(6449):eaax4192, 2019.

[9] Taiping Zeng, Bailu Si, and Jianfeng Feng. A theory of geometry representations for spatial navigation. *Progress in Neurobiology*, 211:102228, 2022.

[10] Shimon Ullman. The interpretation of structure from motion. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 203(1153):405–426, 1979.

[11] Christian Forster, Matia Pizzoli, and Davide Scaramuzza. Svo: Fast semi-direct monocular visual odometry. In *2014 IEEE international conference on robotics and automation (ICRA)*, pages 15–22. IEEE, 2014.

[12] Richard A Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *2011 10th IEEE international symposium on mixed and augmented reality*, pages 127–136. Ieee, 2011.

[13] Angela Dai, Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Christian Theobalt. Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface reintegration. *ACM Transactions on Graphics (ToG)*, 36(4):1, 2017.

[14] Chengzhou Tang and Ping Tan. Ba-net: Dense bundle adjustment network. *arXiv preprint arXiv:1806.04807*, 2018.

[15] Michael Bloesch, Jan Czarnowski, Ronald Clark, Stefan Leutenegger, and Andrew J Davison. Codeslam—learning a compact, optimisable representation for dense visual slam. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2560–2568, 2018.

[16] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 2018.

[17] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12697–12705, 2019.

[18] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl. Center-based 3d object detection and tracking. *CVPR*, 2021.

[19] Yue Wang, Vitor Guizilini, Tianyuan Zhang, Yilun Wang, Hang Zhao, , and Justin M. Solomon. Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. In *The Conference on Robot Learning (CoRL)*, 2021.

[20] Zhijian Liu, Haotian Tang, Alexander Amini, Xingyu Yang, Huizi Mao, Daniela Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2023.

[21] Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(4):640–651, 2017.

[22] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[23] Yuhui Yuan, Xilin Chen, and Jingdong Wang. Object-contextual representations for semantic segmentation. In *ECCV*, 2020.

[24] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. *arXiv preprint arXiv:2012.15840*, 2020.

[25] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *arXiv preprint arXiv:2105.15203*, 2021.

[26] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *Advances in Neural Information Processing Systems*, 34:17864–17875, 2021.

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

[28] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.

[29] Nur Muhammad Mahi Shafiullah, Chris Paxton, Lerrel Pinto, Soumith Chintala, and Arthur Szlam. Clip-fields: Weakly supervised semantic fields for robotic memory. *arXiv preprint arXiv:2210.05663*, 2022.

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

[31] Chung Min* Kim, Mingxuan* Wu, Justin* Kerr, Matthew Tancik, Ken Goldberg, and Angjoo Kanazawa. Garfield: Group anything with radiance fields. In *arXiv*, 2024.

[32] Tinghui Zhou, Shubham Tulsiani, Weilun Sun, Jitendra Malik, and Alexei A Efros. View synthesis by appearance flow. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 286–301. Springer, 2016.

[33] John Flynn, Michael Broxton, Paul Debevec, Matthew DuVall, Graham Fyffe, Ryan Overbeck, Noah Snavely, and Richard Tucker. Deepview: View synthesis with learned gradient descent. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2367–2376, 2019.

[34] Nima Khademi Kalantari, Ting-Chun Wang, and Ravi Ramamoorthi. Learning-based view synthesis for light field cameras. *ACM Transactions on Graphics (TOG)*, 35(6):1–10, 2016.

[35] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *arXiv preprint arXiv:1805.09817*, 2018.

[36] Hai Li, Xingrui Yang, Hongjia Zhai, Yuqian Liu, Hujun Bao, and Guofeng Zhang. Vox-surf: Voxel-based implicit surface representation. *IEEE Transactions on Visualization and Computer Graphics*, 2022.

[37] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. *Advances in Neural Information Processing Systems*, 33:15651–15663, 2020.

[38] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. In *The European Conference on Computer Vision (ECCV)*, 2020.

[39] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. *NeurIPS*, 2020.

[40] Christian Reiser, Songyou Peng, Yiyi Liao, and Andreas Geiger. Kilonerf: Speeding up neural radiance fields with thousands of tiny mlps, 2021.

[41] Alex Yu, Ruilong Li, Matthew Tancik, Hao Li, Ren Ng, and Angjoo Kanazawa. PlenOctrees for real-time rendering of neural radiance fields. In *ICCV*, 2021.

[42] Alex Yu, Sara Fridovich-Keil, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. *arXiv preprint arXiv:2112.05131*, 2(3):6, 2021.

[43] Junyi Cao, Zhichao Li, Naiyan Wang, and Chao Ma. Lightning NeRF: Efficient hybrid scene representation for autonomous driving. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[44] Ricardo Martin-Brualla, Noha Radwan, Mehdi S. M. Sajjadi, Jonathan T. Barron, Alexey Dosovitskiy, and Daniel Duckworth. NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections. In *CVPR*, 2021.

[45] Xingyu Chen, Qi Zhang, Xiaoyu Li, Yue Chen, Ying Feng, Xuan Wang, and Jue Wang. Hallucinated neural radiance fields in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12943–12952, 2022.

[46] Kim Jun-Seong, Kim Yu-Ji, Moon Ye-Bin, and Tae-Hyun Oh. Hdr-plenoxels: Self-calibrating high dynamic range radiance fields. In *ECCV*, 2022.

[47] Bowen Wen, Jonathan Tremblay, Valts Blukis, Stephen Tyree, Thomas Muller, Alex Evans, Dieter Fox, Jan Kautz, and Stan Birchfield. Bundlesdf: Neural 6-dof tracking and 3d reconstruction of unknown objects. *CVPR*, 2023.

[48] Zihan Zhu, Songyou Peng, Viktor Larsson, Weiwei Xu, Hujun Bao, Zhaopeng Cui, Martin R. Oswald, and Marc 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.

[49] Edgar Sucar, Shikun Liu, Joseph Ortiz, and Andrew Davison. iMAP: Implicit mapping and positioning in real-time. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021.

[50] Shuaifeng Zhi, Tristan Laidlow, Stefan Leutenegger, and Andrew Davison. In-place scene labelling and understanding with implicit scene representation. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021.

[51] Zhiwen Fan, Peihao Wang, Yifan Jiang, Xinyu Gong, Dejia Xu, and Zhangyang Wang. Nerf-sos: Any-view self-supervised object segmentation on complex scenes, 2022.

[52] Christopher Xie, Keunhong Park, Ricardo Martin-Brualla, and Matthew Brown. Fig-nerf: Figure-ground neural radiance fields for 3d object category modelling. In *International Conference on 3D Vision (3DV)*, 2021.

[53] Devendra Singh Chaplot, Murtaza Dalal, Saurabh Gupta, Jitendra Malik, and Russ R Salakhutdinov. Seal: Self-supervised embodied active learning using exploration and 3d consistency. *Advances in neural information processing systems*, 34:13086–13098, 2021.

[54] Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Cliport: What and where pathways for robotic manipulation. In *Conference on robot learning*, pages 894–906. PMLR, 2022.

[55] Jesse Thomason, Mohit Shridhar, Yonatan Bisk, Chris Paxton, and Luke Zettlemoyer. Language grounding with 3d objects. In *Conference on Robot Learning*, pages 1691–1701. PMLR, 2022.

[56] Huy Ha and Shuran Song. Semantic abstraction: Open-world 3d scene understanding from 2d vision-language models. *arXiv preprint arXiv:2207.11514*, 2022.

[57] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[58] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. Detecting twenty-thousand classes using image-level supervision. In *ECCV*, 2022.

[59] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM transactions on graphics (TOG)*, 41(4):1–15, 2022.

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

[61] Boyi Li, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and Rene Ranftl. Language-driven semantic segmentation. In *International Conference on Learning Representations*, 2022.

A Appendix / supplemental material

A.1 Scene Partation Example

The scene can be partitioned into different regions using walls as dividers and lines can be aligned to these walls. This is similar in most scenarios, making the annotation of scene regions a straightforward task as shown in Fig. A1.

A.2 Vision-language Embeddings Similarity of Region and Objects

To demonstrate that the relationship of the vision-language and semantic embeddings for different regions is related to our intuition, we compare the similarity in region-region and object-region form and show the results in Fig. A2. It can be seen that based on general knowledge, cognitively related regions(e.g., the dining room and kitchen) and object-region pairs(e.g., sink and kitchen) are also more correlated in the vision-language and semantic feature spaces.

A.3 Ablation Study

To explicitly encode the region information, we apply the LVM to process the background pixels out of the object bounding box and LLM to encode the region label text. What's more, for object pixels, object label text is combined with the region text in the form of 'object in the region' before being encoded by LLM.

Source of Region Information. In our very initial version, we assume that objects with region text include enough information to encode region layouts rather than encoding the background appearance. The region embeddings completely come from the region text label, and object embeddings are learned separately. Fig. A3 shows the difference in embedding processing between the initial version and the current method. Ablation results in Fig. A4 show that context and layout information in background pixels is necessary for layout-object-position association.

Vision-language and Semantic Embeddings Weight. To ablate the contribution of vision-language embeddings from CLIP and semantic embeddings from Sentence-BERT in encoding region features, we compare different weight settings between the v-s embeddings when inferring the regions with 3D position inputs. Results are shown in Fig. A4. It can be seen that both vision-language embeddings and semantic embeddings are indispensable, and weight settings with the greatest results are used for LOP-Field.

A.4 Additional Experiment Results

Additional experiments results of object localization using text query inputs and view localization using image query inputs.

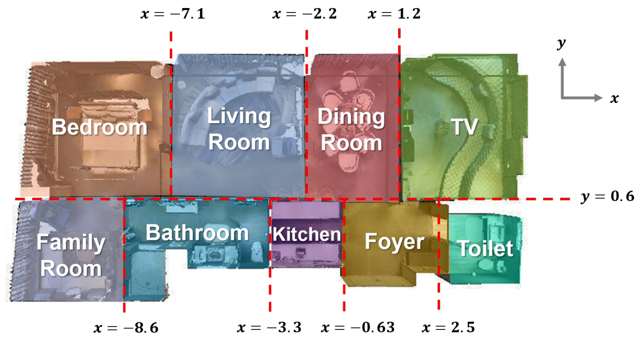
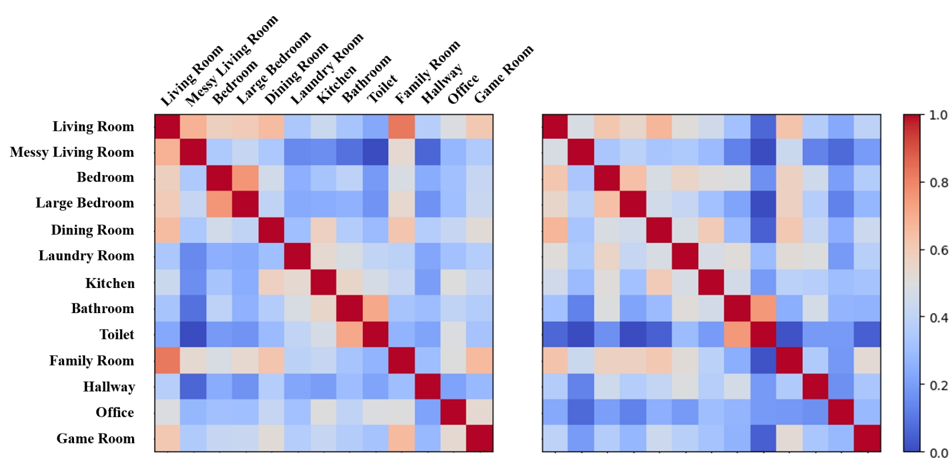
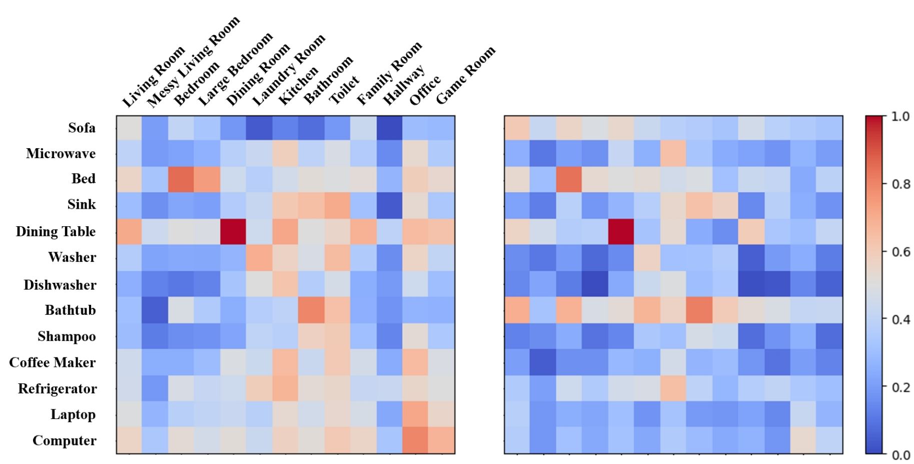


Figure A1: Using walls as dividers to associate lines with them, the scene can be divided into various regions and 3D points can be labeled with related regions easily.



(a)



(b)

Figure A2: The similarity of a set of region embeddings(as shown in a) and object-region embeddings(as shown in b). The left graph shows the vision-language embedding similarity and the right one shows the semantic embedding similarity.

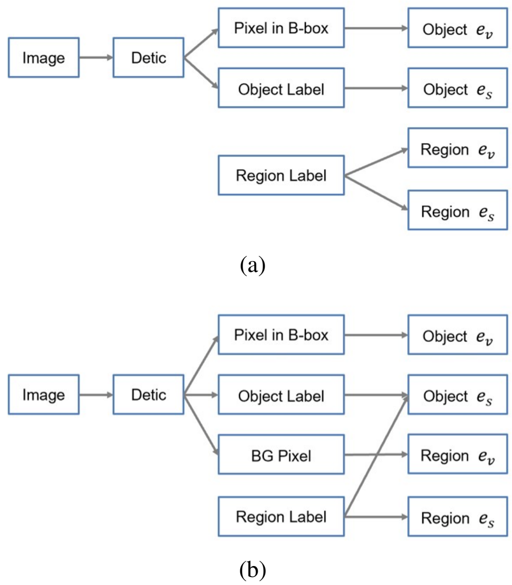


Figure A3: The different source of region information. The initial version which encodes regions from text description is shown in (a), and the current method which encodes background context is shown in (b).

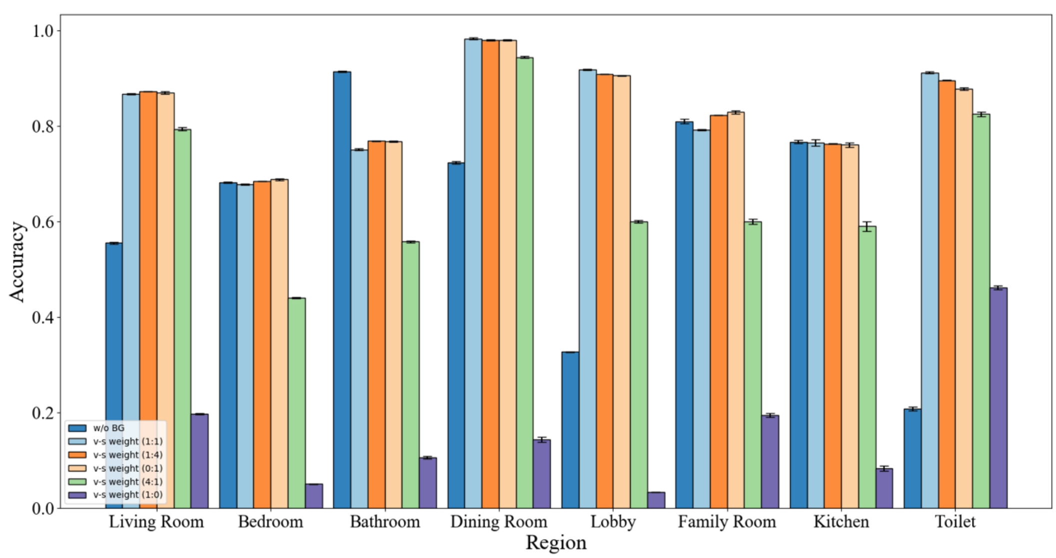


Figure A4: Ablation results on the accuracy of region prediction on Matterport3D[60] with 3D positions input. The w/o BG stands for not encoding background pixels to get region embeddings, and v-s weight ablates the weight of vision-language and semantic embeddings in the embeddings similarity contribution. Error bars show the results among samples from different scenes in Matterport3D[60].

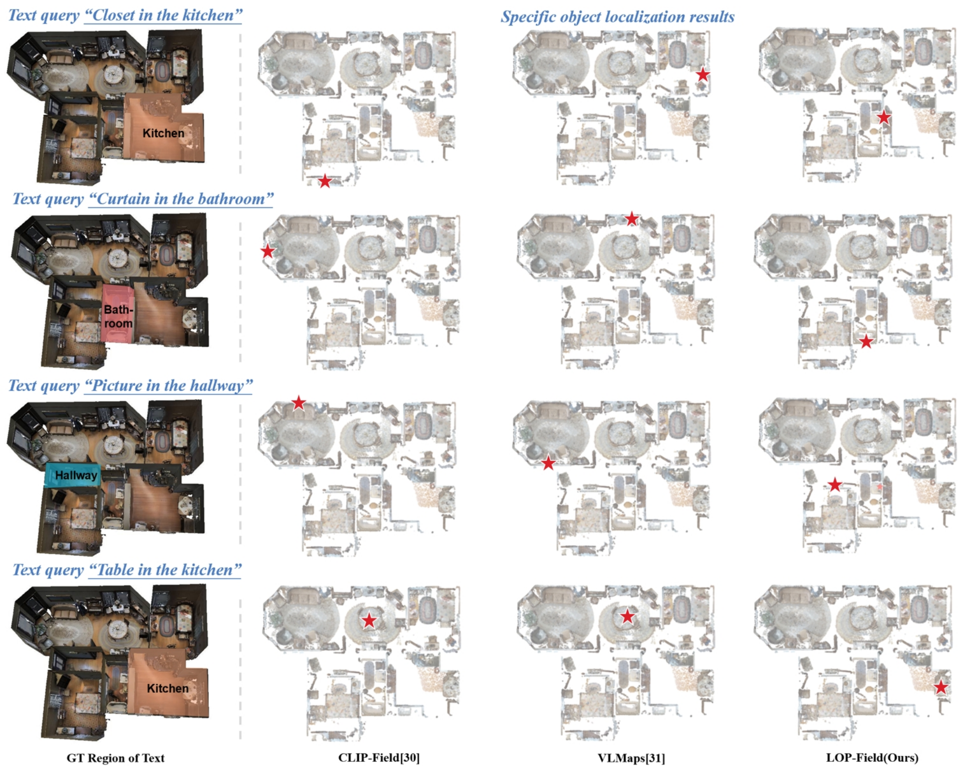


Figure A5: Text query localization on scene 2t7WUuJeko7[60].

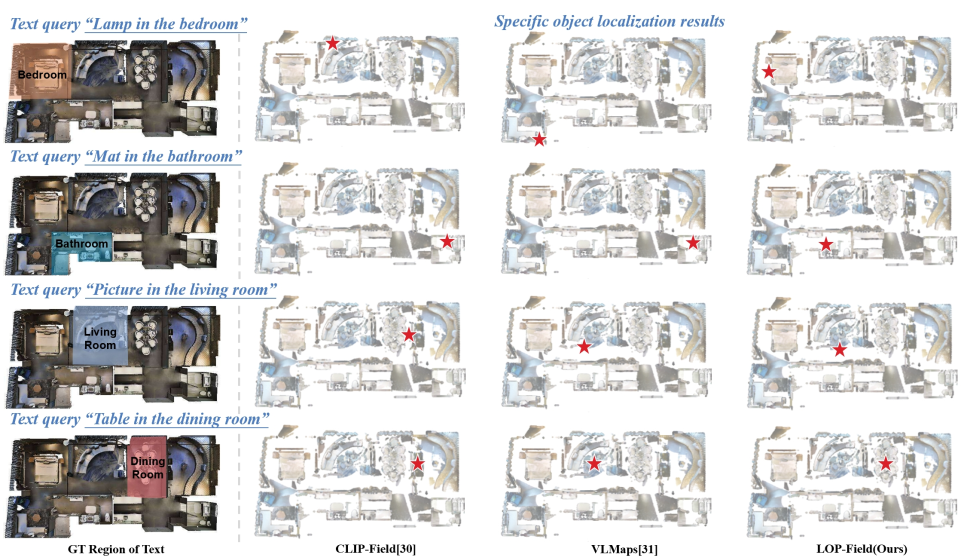


Figure A6: Text query localization on scene 17DRP5sb8fy[60].

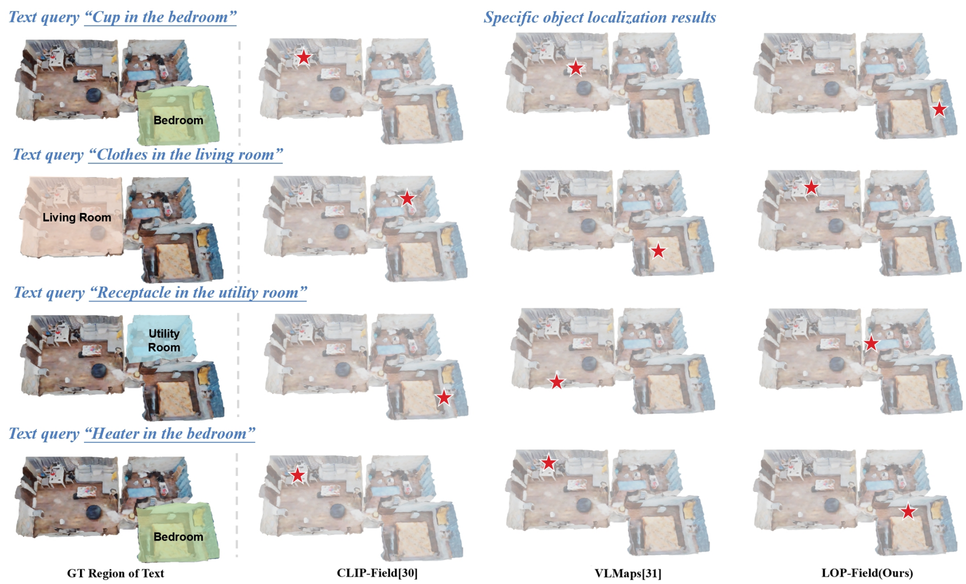


Figure A7: Text query localization on scene Apartment[48].

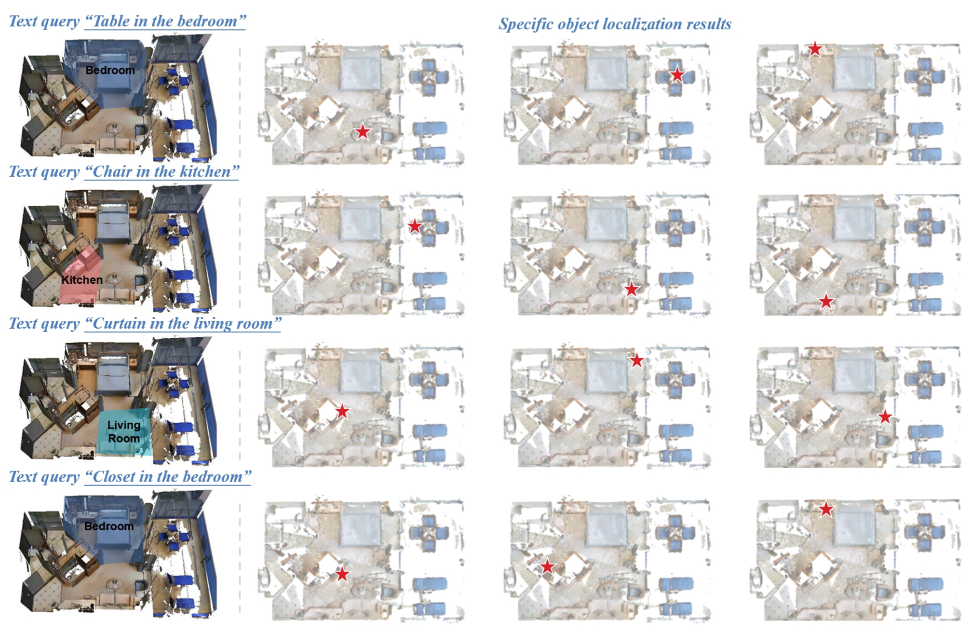


Figure A8: Text query localization on scene HxpKQynjfin[60].

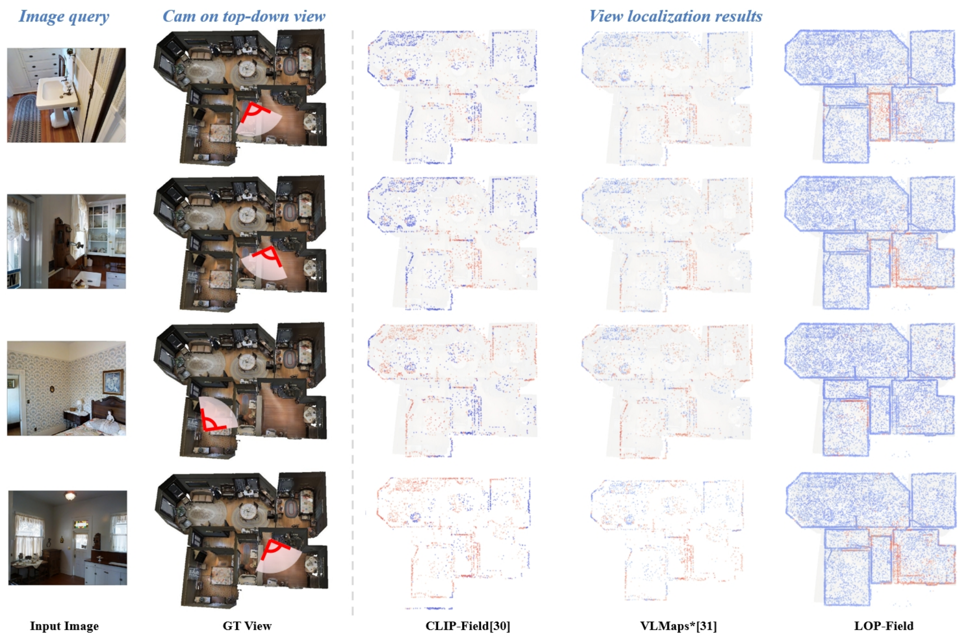


Figure A9: Image query localization on scene 2t7WUuJeko7[60].

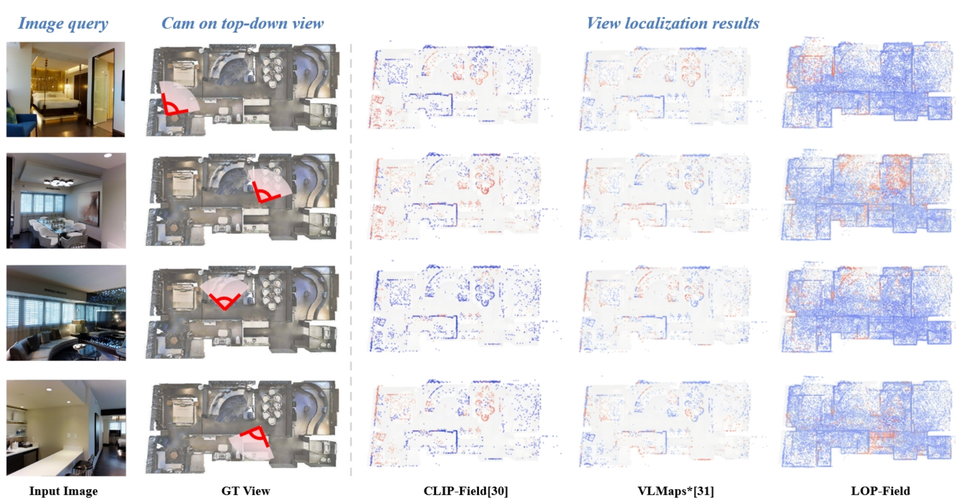


Figure A10: Image query localization on scene 17DRP5sb8fy[60].

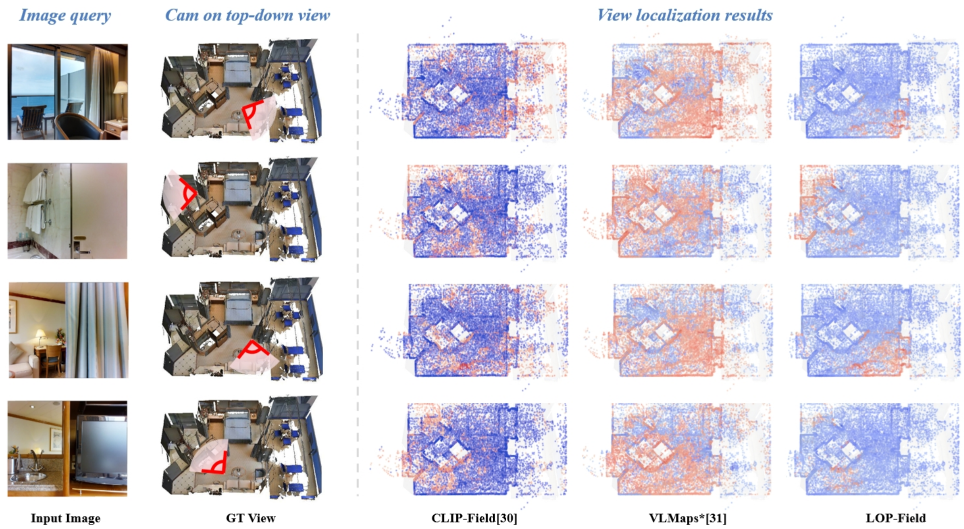


Figure A11: Image query localization on scene HxpKQynjfin[60].

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: This paper claims that with the help of region layout information, robotics can reach better spatial cognition inspired by neuroscience. The claim is explained in detail in the abstract and introduction, and the main contributions are listed in the introduction in Sec. 1.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: The limitations of this work lie in the incomplete usage of region layout and the lack of confidence evaluation for challenging inputs, which is declared in Sec. 5.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.

- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: This paper does not include theoretical results. We use experiments to prove that considering region layout for robotics leads to better results, and this idea comes from neuroscience.

   Guidelines:

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

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The data processing pipeline and method architecture of the proposed LOP-Field which builds layout-object-position association is shown in Fig. 3.2. The details of problem formulation, target feature processing, model architecture, and training strategy are introduced in Sec 3.

Guidelines:

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

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

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We use open-sourced dataset in experiments which is declared in Sec. 4. Codes with model architecture, training, and testing scripts are provided.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.

- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The training details and model setups are declared in Sec. 3 and also in codes.

   Guidelines:

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

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The error bars are available for experiments on the encoding of background appearance which we claim to include region information and on the usage of vision-language and semantic embeddings from large foundation models as shown in Fig. A4.

   Guidelines:

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

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The computation setup is declared in Sec. 3.3 and it is the same for training and testing.

   Guidelines:

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

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: This research conform with the NeurIPS Code of Ethics.

   Guidelines:

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

10. **Broader Impacts**

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

    Answer: [NA]

    Justification: This research focuses on improving the spatial cognition ability of robotics, which is currently in the stage of pre-research of algorithms and has no social impact.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: This research utilizes open-sourced datasets and the model does not have a risk for misuse.

    Guidelines:

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

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The using of existing datasets and foundation models is correctly cited. We obey the license and usage instructions.

    Guidelines:

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

13. **New Assets**

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

    Answer: [Yes]

    Justification: We provide instructions to utilize our codes and models for researching usage.

    Guidelines:

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

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This research does not include crowdsourcing experiments or human subjects.

    Guidelines:

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

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects. It is a robotic research on the existing open-sourced dataset.

    Guidelines:

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