2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Open-Vocabulary Object-Goal Navigation by Generalizing Semantic Mapping with Dense CLIP

Meng Wei[1,2], Chenyang Wan[1,3], Tai Wang[1], Wenzhe Cai[1], Yilun Chen[1], Hanqing Wang[1], Jiangmiao Pang[1,‡], Xihui Liu[2,‡]

***Abstract*— Object-oriented embodied navigation tasks require agents to locate specific objects, either defined by category or images, in unseen environments. While recent methods have made progress in extending closed-set models to open-vocabulary scenarios with foundation models, they typically rely on training-free large language models (LLMs) or finetuning with end-to-end reinforcement learning (RL). However, they face challenges in efficiency (e.g., the overhead and cost of LLM inference) and limited generalization from intensive RL training. In this paper, we propose OVExp, a training-efficient framework for open-vocabulary exploration. We make the first effort to demonstrate the generalization capabilities of semantic map-based goal prediction networks using Dense CLIP models. A major challenge is that preserving both precise point-wise object locations and generalizable visual representations in the semantic map leads to unaffordable training costs. To address this, we design a Cross-Modal Transfer on Semantic Mapping strategy which adapts an intriguing text-only training and transfer to multi-model semantic mapping and goals in test-time. Despite relying on text-based spatial layouts with limited objects, OVExp demonstrates robust generalization to unseen targets on established ObjectNav benchmarks.**

I. Introduction

Object-oriented visual navigation tasks require an embodied agent to locate and reach object goals in unseen environments. The goal can be specified either by language prompts that describe the object category [1] or by an image of the target object [2]. Since the environment is unknown, agents must explore efficiently before the target becomes visible, which involves reasoning about room layouts, object placements, and inter-object relations. Although previous methods [3], [4], [5], [6], [7], [8] have made strides in efficient exploration, they are inherently limited to handling only a fixed set of object goals seen during training. Recent works [9], [10], [11], [12], [13], [14] attempt to overcome this by leveraging commonsense knowledge from LLMs in a training-free manner, but the lack of grounding in embodied navigation leads to suboptimal performance.

Extending models to open-vocabulary scenarios with limited task-specific data has made great progress in computer vision [15], [16], [17], [18], [19], [20], [21], [22], leveraging pretrained image-text representation from foundation models like CLIP [23]. However, applying it to open-vocabulary exploration policies proves to be non-trivial. Existing attempts, like EmbodiedCLIP [24] and ZSON [25], integrate CLIP into end-to-end Reinforcement Learning (RL)-based policies but suffer from poor generalization due to the low sample efficiency and sparse rewards. In contrast to the intensive RL training, recent modular methods [7], [8] rely on object semantic maps and directly train a goal prediction network using supervised learning. Despite achieving better training efficiency and performance, it remains unexplored how to enable open vocabulary goal prediction in this paradigm.

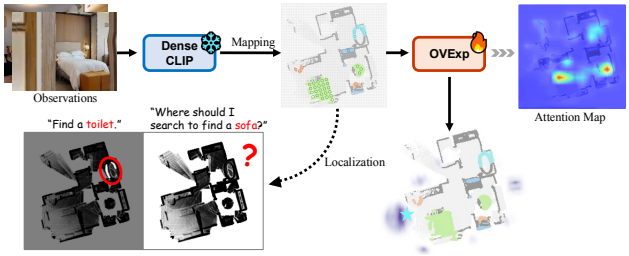

Fig. 1. Although VLMaps can localize objects precisely in explored areas, object relationships are not explicitly encoded when querying for unmapped objects. Through fine-tuning, OVExp effectively learns the spatial layout and infers the target's position in unexplored areas.

Recently, some methods [26], [27] propose to fuse pixel embeddings from dense CLIP models into VLMaps to support open vocabulary landmark localization ("where it is?"). In this paper, we take a step further to address the critical aspect of exploration: "where to search?". We propose the *OVExp* framework, which is based on goal prediction networks, to integrate the VLMaps into the training loop for open vocabulary exploration. As shown in Figure 1, the spatial object relationships are initially not encoded in the map with the exploration-based query. But with precise object locations and generalizable representations, OVExp is fine-tuned to understand the spatial layout in explored areas and infer the target position in unexplored areas.

The key strategy in OVExp is the *Cross-Modal Transfer on Semantic Mapping* (CMT-SM). Scaling the original VLMaps [26], [27] will create significant storage demands and I/O burden to the training process. Hence, we propose to train the OVExp policy in *a text-only way* and directly transfer it to handle vision-based map representations and goals in test-time. Specifically, we build the low-cost semantic maps with ground-truth layout and encode object labels with CLIP *text* encoder for training and use the well-aligned *visual* features to build the map for inference. The frozen map feature is then modulated by a Feature-wise Linear Modulation (FiLM) module conditioned on the goal features. Finally, the modulated features pass through a few self-attention layers and a transposed convolution model to generate the prediction map. Notably, the training procedure

[1]Shanghai AI Lab, [2]The University of Hong Kong, [3]Zhejiang University
‡Corresponding Authors

only needs the geometry layout in the text-form, which indicates an intriguing possibility that we may decouple visual perception from training exploration policies, *e.g.*, we can directly generate text-only layouts instead of scaling up the scene data with realistic textures.

To verify the open-vocabulary exploration capabilities of our framework, we conduct extensive experiments on object-oriented navigation benchmarks, including **HM3D-ObjectNav** [28], **HM3D-InstanceImageNav** [29], and the open-vocabulary **HM3D-OVON** [30]. The results demonstrate that: (1) supervised text-only training provides significantly better performance and efficiency than other training-based models; (2) with reasonable data collection and training costs, OVExp substantially outperforms training-free methods that rely on unstable and cumbersome LLM interactions (e.g., GPT-2, GPT-3.5); (3) although MLLMs like GPT-4 achieve competitive results without training, their inference cost is prohibitive; (4) models trained with text-only input can be effectively adapted to vision-only inference.

II. Related Work

Map-based Navigation. Map-based representations have proven effective for navigation, enabling efficient path planning with spatial awareness and history information. Common map types include occupancy maps [31], topological graphs [11], semantic maps [5], and implicit maps [32]. Semantic maps, which incorporate high-level information, allow follow-up works [7], [8] to predict long-term goal probabilities, achieving state-of-the-art performance without reinforcement learning. Leveraging advances in semantic representation from Large Vision-Language Models (VLMs), VLMaps [26] project pixel features from LSeg [33] onto high-dimensional semantic maps, but they require constructing a complete map before open-vocabulary goal indexing. In contrast, we explore how VLMs can directly empower open-vocabulary exploration. **Open Vocabulary Object Navigation.** Navigating to open-vocabulary object goals is a realistic yet challenging task, prompting the creation of benchmarks [34], [35], [36] to advance solutions. Some methods [25] use images as goals, encoded with CLIP to generalize to diverse objects. A recent trend leverages Large Language Models (LLMs) for training-free exploration [9], [10], [11], [12], [13], exploiting commonsense knowledge to guide goal-directed exploration. However, relying solely on observation images can limit understanding of the full 3D environment, often resulting in suboptimal decisions. Consequently, effective open-vocabulary exploration policies remain an open challenge.

III. Method

A. Object-Oriented Navigation Task Definitions

Object Goal Navigation (ObjectNav): In the ObjectNav task, the embodied agent is required to navigate to an instance of a given object category in unseen environments, such as "chair" or "table". The agent is equipped with RGB and Depth cameras capturing observation images and is provided with the 3-DoF current pose relative to the start position at each timestep t. The agent continuously explores its surroundings until it finds an object instance from the target category. The discrete action space $a_t \in \mathcal{A}$ includes `move_forward`, `turn_left`, `turn_right`, `look_up`, `look_down`, and `stop`. The navigation episode ends upon execution of the stop action or upon reaching the timestep limit. Success is achieved when the agent predicts a stop action at a location where the distance to the target object is less than 1 meter and the target object is within view.

Instance-Specific Image Goal Navigation (InstanceImageNav): While in the InstanceImageNav task, the embodied agent must navigate to a *specific* object instance depicted in a provided RGB image. Compared to the ObjectNav task, InstanceImageNav is more challenging as it demands the agent to precisely identify and locate the sole target instance in the environment, which adds extra complexity to the navigation process. Despite this complexity, both tasks are object-centric and the evaluation protocol of ObjectNav can be directly applicable to InstanceImageNav.

B. Cross-Modal Transfer on Semantic Mapping

We adopt different types of high-dimensional semantic maps during training and testing phrases, because collecting semantic maps with dense pixel-level visual embeddings from LSeg [33] makes the training impractical (*requiring 1000× storage demand and 20× training time than text-only training*). Therefore, we employ the semantic mapping procedure detailed in Section III-B.1 to construct categorical semantic maps. We curate a list of 92 object categories from the HM3DSem [37] dataset, which offers abundant pixel-level annotated semantics. These categorical maps are subsequently transformed into language-based maps using CLIP's text embeddings, detailed in section III-B.2. While in inference, we directly construct maps through vision-based mapping III-B.3, as used in VLMaps [26].

1) Semantic Mapping: In learning-based goal-oriented exploration, semantic maps [5] have proven to be an effective representation for encoding episodic navigation history. Semantic maps can capture both geometric priors like spatial layout, obstacles, navigable areas and semantic information such as scene object categories, their locations, and spatial relationships. To build the semantic map, egocentric visual observations is first segmented into semantic categories using a pre-trained segmentation model. Next, a point cloud is extracted from the depth image by back-projection into the 3D world. Each point is associated with the corresponding semantic label in the 2D observation image. The point cloud is then binned into a voxel grid. Summing along the height dimension, the voxel grid is projected into an egocentric top-down semantic map. To be merged with the global map, the egocentric map is finally transformed into the allocentric coordinate system using agent's pose information. The built semantic map $\mathcal{M}^s \in \mathbb{R}^{(N+2)\times M\times M}$ has 2 non-semantic channels (1: obstacle map, 2: explored map) and N semantic channels related to N object categories of interests. Each cell within the $M\times M$ grid corresponds to a region of $5cm\times 5cm$.

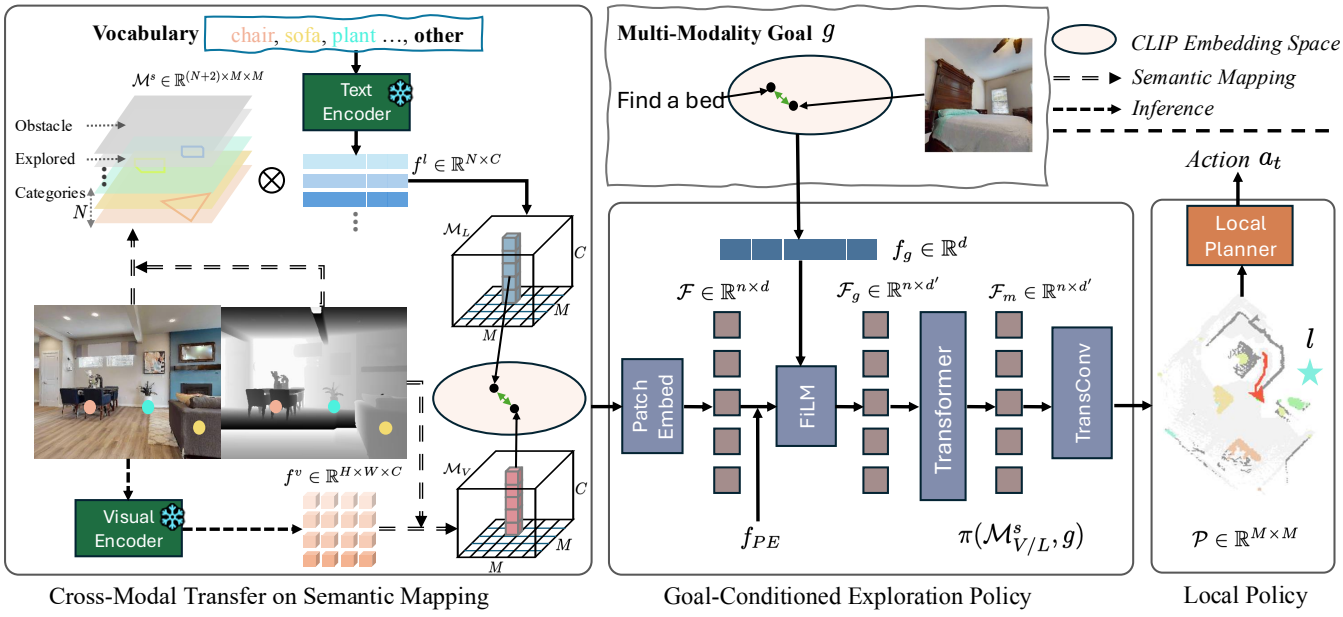

Fig. 2. The overall framework of OVExp for open vocabulary object-oriented exploration. OVExp can accept either language-based or vision-based maps as input and accommodates textual and visual object goals. For simplicity, the goal identification model is omitted.

2) Language-based Mapping: To enrich the representation of collected semantic maps, we encode the N semantic channels of $\mathcal{M}^s$ with language features derived from CLIP's embedding space. Formally, we input the text list of N objects $\{O_1, O_2, \ldots, O_N\}$ (O_N represents the background class and use "other" as the text) into CLIP's text encoder to produce high-dimensional language-based object features $f^l \in \mathbb{R}^{N\times C}$. Each semantic channel $\mathcal{M}^s_{O_i} \in \mathbb{R}^{1\times M\times M}$ of the semantic map encodes the confidence score of the object's existence in the grid cells. To transform $\mathcal{M}$ into a language-enhanced map representation $\mathcal{M}_L$, we compute the weighted sum of the semantic channels and the corresponding language embeddings:

$$\mathcal{M}_L = \frac{\sum_{i=1}^{N} \mathcal{M}^s_{O_i} * f^l_{O_i}}{\sum_{i=1}^{N} \mathcal{M}^s_{O_i}} \tag{1}$$

where $\mathcal{M}_L \in \mathbb{R}^{C\times M\times M}$. In $\mathcal{M}_L$, the objects are no longer encoded in separate channels; instead, each grid cell contains the averaged language features of all objects present.

3) Vision-based Mapping: During inference, we construct an equivalent vision-based map input with the pretrained image encoder of LSeg, which produces pixel embeddings aligned with their corresponding label embeddings, resembling real-world scene representations. The vision-based map construction is a one-stage process. At each timestep t, we extract the dense pixel-wise visual features $f^v \in \mathbb{R}^{H\times W\times C}$ of the observation RGB image $\mathcal{I} \in \mathbb{R}^{H\times W}$. The transformation from 2D egocentric pixel representations to a top-down grid map is similar to the semantic mapping process described in Section III-B.1. The only difference lies in how the grid map is updated during the online mapping process. In updating the semantic categorical maps, the maximum object confidence from all timesteps is consistently taken for each grid cell. However, the vision-based map $\mathcal{M}^t_V$ is updated as follows:

$$\mathcal{M}^t_V[i,j] = \frac{\mathcal{M}^{t-1}_V[i,j] \times \mathcal{N}^{t-1}[i,j] + m^t_v[i,j]}{\mathcal{N}^{t-1}[i,j] + 1}; \quad \mathcal{N}^t[i,j] = \mathcal{N}^{t-1}[i,j] + 1 \tag{2}$$

where $\mathcal{M}_V \in \mathbb{R}^{C\times M\times M}$. $[i,j]$ represents the locations that will be updated by the incoming map feature $m^t_v \in \mathbb{R}^{C\times M\times M}$ which is projected from the current observation feature f^v. $\mathcal{N} \in \mathbb{R}^{M\times M}$ records the number of updates in each grid cell over time.

C. Goal-Conditioned Exploration Policy

Given robust high-dimensional semantic maps $\mathcal{M}^s_{V/L} \in \mathbb{R}^{(C+2)\times M\times M}$ (the first two channels represent the obstacle map and explored area), which encode the spatial and semantic information of the environment, and an object goal g, our objective is to learn a global policy π that outputs the long-term goal location l within the local map:

$$\pi(\mathcal{M}_{V/L}, g) \rightarrow l \tag{3}$$

We use $\mathcal{M}^s_L$ for training and switch to $\mathcal{M}^f{}_V$ during inference. Following previous map-based methods [38], [8], the policy π will output an object probability map and the long-term goal location l is selected with the largest probability. This policy is flexible to handle goals specified in different modalities, such as a textual goal g_t in ObjectNav and an image goal g_i in InstanceImageNav.

Goal-Conditioned Map Encoder. The map $\mathcal{M}^s_{V/L}$ is first partitioned into non-overlapping patches which are projected into map token embeddings through the patch embed operation. Then we add learnable positional embedding f_{PE} to the token embeddings and obtain the map features $\mathcal{F}$:

$$\mathcal{F} = \textsc{PatchEmbed}(\mathcal{M}_{V/L}) + f_{PE} \tag{4}$$

where $\mathcal{F} \in \mathbb{R}^{n \times d}$, n denotes the number of map tokens and d denotes the hidden size.

To condition the long-term goal prediction on the object goal embedding $f_g \in \mathbb{R}^d$, which is text or image embedding from CLIP, we employ an efficient Feature-wise Linear Modulation (FiLM) [39] layer. This layer applies a feature-wise affine transformation to fuse the two sources of features $\mathcal{F}$ and f_g , producing a goal-conditioned map feature $\mathcal{F}_g$:

$$\mathcal{F}_g = \gamma(f_g) \odot h(\mathcal{F}) + \beta(f_g); \mathcal{F}_m = \text{TRM}(\mathcal{F}_g) \quad (5)$$

where $\mathcal{F}_g \in \mathbb{R}^{n \times d'}$.$\gamma(.)$, $\beta(.)$ and $h(.)$ are three linear transformations. $\gamma(.)$ and $\beta(.)$ generate the scaling and shifting vectors from f_g respectively. $h(.)$ reduces the dimension of $\mathcal{F}$ from $d = 512$ to $d' = 64$. This fusion process effectively integrates the semantic goal information into the map features. Finally, the encoded map features $\mathcal{F}_m \in \mathbb{R}^{n \times d'}$ is produced by feeding $\mathcal{F}_g$ to a two-layer transformer to facilitate feature representation learning.

Goal Location Prediction. To generate the object probability map for selecting the goal location, we design a convolution network as the decoder. The decoder $\mathcal{D}$ consists of one convolution layer and two transposed convolution layers which upsamples the map feature $\mathcal{F}_m$ to the original map resolution and generate the goal probability map $\mathcal{P} \in \mathbb{R}^{M \times M}$. The location with the highest value in this map is selected as the predicted long-term goal location. Following [8], to improve the efficiency of exploration, the final long-term goal location is determined by weighting $\mathcal{P}$ with the geodesic distance to the agent's current location:

$$\mathcal{P} = \mathcal{D}(f_g); l = arg\max_{i,j}(\mathcal{P}_{ij} \times g_{ij}) \quad (6)$$

(i, j) denotes the map index and g is the exponential weight derived from the geodesic distance. We use the binary cross-entropy loss to train the exploration policy.

D. Analytical Local Planner

To reach the predicted goal locations from exploration policy or the identified goal points, we adopt an analytical local planner to translate the long-term goal location into an executable action $a_t \in \mathcal{A}$ as in previous modular map-based methods. The Fast Marching Method (FMM) is employed incrementally to calculate the shorted path from the agent's current location to the goal location. Then a waypoint along this path is selected considering the agent's step distance.

IV. Experiment

A. Experimental Setup

Training Dataset. We generate semantic maps from the HM3DSem [37] dataset using the Habitat [40] simulator. HM3DSem annotates object instances across 216 3D scene reconstructions with 1660 raw object names. We set a goal-agnostic agent to explore the training scenes from random start locations, allowing it to wander for 500 steps. During this process, we save the semantic map every 25 steps. To avoid using maps with minimal unexplored areas, we select only the first half of the saved maps for training.

Evaluation Datasets. The evaluation is conducted on the validation sets of three navigation datasets, including one InstanceImageNav dataset and two ObjectNav datasets:

HM3D-ObjectNav [28]: Released in the Habitat 2022 challenge, this dataset has 2000 episodes from 20 validation scenes in HM3D, targeting 6 specific goal objects. We collect the semantic maps based on 86 objects, **excluding the 6 goal objects.**

HM3D-OVON [30]: An **open-vocabulary** ObjectNav dataset comprising 379 goal objects across 181 scenes. We collect semantic maps using annotations for 100 objects from the **Train** split, and evaluate on the **Val Unseen** split with 49 novel goal objects that **are absent during training**.

HM3D-InstanceImageNav [29]: Released in the Habitat 2023 challenge, this dataset comprises 1000 episodes from 36 validation scenes in HM3D, targeting the instance image goals of 6 objects.

Evaluation Metrics. We evaluate performance with two standard metrics. ***Success***: measures the proportion of episodes in which the agent successfully stops near the goal object. ***SPL*** (Success weighted by Path Length): assesses the efficiency of the navigation by weighting the success rate by agent path length relative to shortest path length.

B. Implementation Details

We use a global map of size 960×960 for training, applying random crop operations to 720×720, along with random flips and random rotations for data augmentation. We embed the map features with a patch size of 16×16. Then a 2-layer transformer with the hidden size of 512 and 8 attention heads is used to update the map features. For fusion with FiLM, both the encoded map features and goal embeddings are reduced to a hidden size of 64. For the transposed convolutional decoder, the first convolutional layer uses a kernel size of 3 and a padding size of 1. The two transposed convolutional layers upsample the feature maps using transposed kernels with a size of 4. The prediction model is trained for 20 epochs with a batch size of 8, requiring 20 hours with 8 NVIDIA V100 GPUs. We use AdamW optimizer with initial learning rate of 1^{e-4} and weight decay of 1^{e-4}. We use cosine decay.

C. Open-Vocabulary ObjectNav Performance.

Settings. We evaluate OVExp's generalization ability in navigating to unseen goal objects on HM3D-ObjectNav, which has 6 goal objects. To ensure OVExp acquires zero experience with these objects, the model is trained by excluding these 6 objects, i.e., their locations are not learned.

Baselines. We compare OVExp with three types of existing open vocabulary navigation methods: leftmargin=*

- `Heuristic-based Exploration`: CoW [34] combines a goal-agnostic Frontier-Based Exploration (FBE) algorithm with an open vocabulary goal detector.
- `Learning-based Exploration`: ZSON [25] trains the policy on extensive image-goal data which

TABLE I

COMPARISON WITH STATE-OF-THE-ART OBJECTNAV METHODS ON THE VAL SET OF **HM3D-OBJECTNAV**.

Method	Mapping	Trainable	VLM	Success↑	SPL↑
Closed-Set Setting					
SemExp [5]	✓	RL	-	37.9	18.8
PIRLNav [6]	×	RL&IL	-	**61.9**	27.9
PEANUT* [8]	✓	SL	-	60.5	**30.7**
OVExp*	×	SL	-	60.6	29.7
Open-Vocabulary Setting					
CoW [34]	✓	×	CLIP	32.0	18.1
ZSON [25]	×	✓	CLIP	25.5	12.6
L3MVN† [10]	✓	×	GPT-2	50.4	23.1
PixelNav [41]	×	✓	GPT-4	37.9	20.5
ZSC [9]	✓	×	GPT-3.5	39.2	22.3
VoroNav [11]	✓	×	GPT-3.5	42.0	26.0
VLFM+FMM [42]	✓	×	BLIP2	50.9	23.6
GAMap [43]	✓	×	CLIP	53.1	26.0
InstructNav‡ [44]	✓	×	GPT-4	58.5	20.9
OVExp*	✓	✓	LSeg	**59.7**	**28.8**
OVExp†	✓	✓	LSeg	58.9	26.2
OVExp‡	✓	✓	LSeg	56.3	27.9

"-" indicates results run from their officially released checkpoint. Methods with the same detection module are marked with superscripts: * for PEANUT, † for L3MVN, and ‡ for InstructNav.

TABLE II

COMPARISON ON THE **OPEN VOCABULARY HM3D-OVON [30]**

Method	Policy	VLM	Success↑	SPL↑
DAgger	DA	SigLIP	10.2	4.7
PPO	RL	SigLIP	18.6	7.5
BCRL	BC+RL	SigLIP	8.0	2.8
DAgRL	DA+RL	SigLIP	18.3	7.9
VLFM	-	BLIP2	35.2	19.6
OVExp	SL	LSeg	**37.8**	**19.6**

We use the val_unseen split which contains 49 goal objects that are distinct from the 79 objects used during training.

is transferred to object-goal navigation by embedding both types of goals in the same CLIP embedding space.

– `VLM-based Exploration`: Most existing open vocabulary exploration methods are training-free and based on Large Vision Language Models to extract the prior knowledge about object arrangements. L3MVN [10], ESC [9], PixelNav [41] and VoroNav [11] convert observation images to captions mainly about object presence, which are then used to score potential waypoints based on LLM's knowledge of relationships between the observed objects and the goal object. VLFM [42] and InstructNav [44] compute explicit value map of target object presence by extracting the similarity scores from large vision language models like BLIP2 [45] and GPT-4.

TABLE III

COMPARISON OF USING VISION-BASED OR LANGUAGE-BASED MAPS DURING INFERENCE ON HM3D VAL SPLIT.

Semantic Map Modality	Success↑	SPL↑
Text (GroundTruth)	59.6	28.7
Vision	**60.6**	**29.7**

TABLE IV

EVALUATION OF THE VISION-ONLY INFERENCE ON THE VAL SET OF HM3D-INSTANCEIMAGENAV.

Method	Config	Success↑	SPL↑
Mod-IIN [47]	Stretch	56.1	**23.3**
OVExp	Stretch	59.7	21.5
OVExp	LoCoBot	**63.0**	20.5

Performance Analysis. Given the variety of object detectors employed in different methods, a comprehensive comparison poses challenges. Hence, as shown in Table I, we report results of OVExp using three types of goal detectors from open-source projects: PEANUT, L3MVN and InstructNav for a fair comparison. Notably, when using the same goal detector, the open vocabulary performance of OVExp (purely unseen) is even competitive with the closed-set model (all seen). Moreover, with reasonable finetuning cost, OVExp outperforms L3MVN by $+8.5\%$ in Success and $+3.1\%$ in SPL. Although InstructNav achieves better performance in Success rate with powerful but expensive GPT-4 model, we achieves much better performance in SPL $+7.0\%$, demonstrating better efficiency.

To validate OVExp's generalization ability on more unseen objects, we evaluate OVExp on the HM3D-OVON [30] as shown in Table II. We compare OVExp with various learning-based methods which are trained with frozen SigLIP [46] vision, text encoders and then tested on unseen objects. By training on a fixed set of objects and generalizing to novel objects, OVExp achieves significant improvement over the Behavioral cloning (BC), DAgger (DA) and Reinforcement learning (RL) policies. Comparing to the training-free method VLFM, OVExp achieves better Success Rate.

D. Effectiveness of Text-only Training

Vision-Based Inference Even Surpasses Text-Based Inference. While text and image embeddings from contrastive models like CLIP share some structural alignment, a modality gap still exists. Using pixel embeddings instead of text embeddings during inference naturally introduces richer semantic information when constructing the maps. To investigate the disparity, we also evaluate the performance using text embedding based semantic maps with the ground truth object annotations. The results are reported in Table III. Surprisingly, ground-truth text-based mapping underperforms vision-based mapping. We conjecture that using pixel embeddings during testing introduces variability, which appears as noise in the semantic maps. This noise aids generalization to novel scenes by reducing reliance on overfitted patterns

Fig. 3. (a) When detection fails, the text-based map lacks context, while the vision-based map remains robust with richer information. (b) In the self-attention layers, LSeg focuses more precisely on relevant objects, while MaskCLIP's attention is more dispersed.

Fig. 4. Visualization of attention maps under different goal conditions. When searching for the sofa, the model attends to the bathroom and bed to predict its location. For the toilet, it attends to the chairs, sofa (misdetected) in the living room, bed in the bedroom, and the hallways.

from supervised learning. Moreover, we find that vision-based mapping is more robust to detection errors. As shown in Figure 3 (a), when the sofa is mis-detected, text-based semantic mapping provides no information, whereas vision-based maps can capture this.

Vision-only Inference. OVExp is trained in a text-only manner, with both the semantic map and goal features encoded through text. Since text-to-vision transfer is a relatively unexplored paradigm, we conduct an experiment to assess how well OVExp handles this domain shift. Specifically, we perform vision-only inference on the HM3D-InstanceImageNav benchmark to test its adaptability. In this task, the textual goal embedding is replaced with the image embedding of the goal object, which is more challenging than ObjectNav because the agent must locate a specific object rather than just the first encountered instance. We compared with the state-of-the-art modular method Mod-IIN [47] which proposes a instance re-identification module for goal matching and use FBE as the exploration policy. The results are reported in Table IV. We observe that OVExp exhibits higher success rates but lower performance on SPL compared to Mod-IIN in both embodiments. These findings suggest that OVExp, learned for category-based prediction, may not efficiently locate specific instances. However, it still achieves a better search strategy than FBE to find the target object eventually.

E. Ablation Study.

Impact of Semantic Map Resolution The use of global high-dimensional semantic maps for training incurs the most significant computational cost in our framework. We analyze how different map resolutions can impact effectiveness considering our computational resources. We examine patch sizes of $36, 24, 16$, resulting map sizes of $20 \times 20, 30 \times 30, 45 \times 45$ respectively. As shown in table V, using maps with higher resolution contributes to better performance. However, further increasing the map size will result in prohibitively computational cost. Therefore, we adopt the map size of 45×45.

TABLE V

COMPARISON OF DIFFERENT MAP SIZE ON HM3D VAL SPLIT.

Map Size	20×20	30×30	45×45
Success↑	56.8	58.9	**60.6**
SPL↑	26.6	27.8	**29.7**

TABLE VI

COMPARISON ON DIFFERENT TYPES OF DENSE CLIP FEATURES.

Dense CLIP Model	LSeg	MaskCLIP
Success↑	**59.4**	58.9
SPL↑	**28.7**	28.5

Impact of Different Dense CLIP Feature. We evaluate two types of dense CLIP features—LSeg [33] and MaskCLIP [48]—to understand the impact on OVExp's performance. As shown in Table VI, using LSeg features achieve better performance than MaskCLIP features. LSeg, fine-tuned on object segmentation datasets, produce pixel-level embeddings that are closely aligned with object text labels, enabling precise localization in semantic mapping. This fine-grained alignment allow OVExp to identify and map detailed scene elements accurately. While MaskCLIP adapts CLIP's global self-attention into a convolutional layer to produce region-based, dense patch features. These features lack the precision needed for scene objects, resulting in less detailed semantic maps. As shown in Figure 3 (b), when

Fig. 5. Qualitative results on `HM3D-ObjectNav`. First row: *FBE*. Second row: *OVExp*.

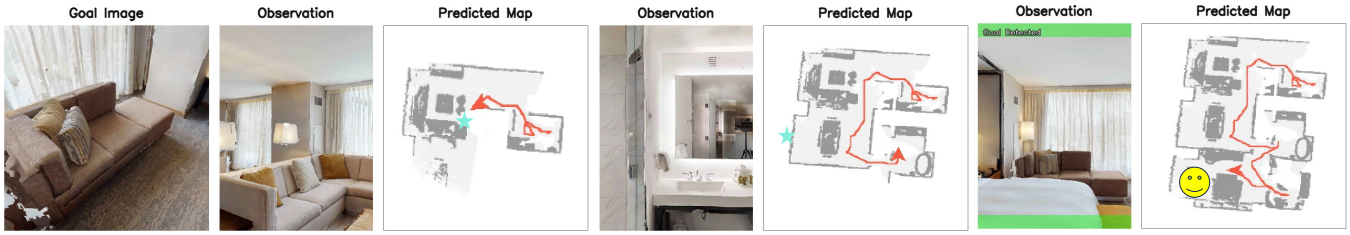

Fig. 6. Qualitative results of vision-only inference on `HM3D-InstanceImageNav`.

predicting the same target, "sofa", LSeg attends to more surrounding context objects compared to MaskCLIP.

V. Qualitative Results.

As *OVExp* adopts a trade-off between predicted goal locations and closest locations, we conducted a comparison against the FBE baseline to show the efficiency of *OVExp*. As shown in Figure 5, while FBE initially explores the coatroom, leading to inefficient back-and-forth movements, *OVExp* directly navigates out of the bedroom and onwards.

Furthermore, we offer a qualitative analysis of transferring *OVExp* to the InstanceImageNav task. As *OVExp* is primarily object-oriented, it may lack a nuanced understanding of specific goal details within an image. However, it still manages to generate a reasonable exploration path. As shown in Figure 6, *OVExp* first generates a goal location in the living room, where another "sofa" is present, before proceeding to the bedroom to locate the goal instance. And we visualize the attention maps of the transformer to check if the model learn the spatial layout as context as show in Figure 4. We provide more qualitative results in the *demo video*.

VI. Real-world deployment.

We deploy OVExp on a Turtlebot4 equipped with an RGB-D camera and a 2D LiDAR for odometry and obstacle avoidance. The robot continuously sends real-time images, depth data, and poses to a remote server running OVExp. Using these inputs, the server extracts dense visual features, detects target objects, and projects them into a top-down semantic map. OVExp generates open-vocabulary exploration goals within this map and sends them back to the robot, after which Turtlebot4's oracle planner computes an optimal global path to reach the goal. We evaluate OVExp in an indoor office setting that contains objects unseen during training. As shown in the *demo video*, OVExp can identify and navigate towards novel goals such as "sofa" or "printer" efficiently. This real-world deployment highlights the practicality and scalability of our framework beyond simulation.

VII. Conclusion

In this paper, we introduced *OVExp*, a novel modular framework that leverages Dense CLIP models and semantic mapping for open-vocabulary exploration. Our approach encodes RGB-D observations with VLMs and project them onto high-dimensional semantic maps. Specifically, a novel cross-modal transfer on semantic mapping strategy is designed to ensure efficient training with aligned visual-language features. We train a goal-conditioned exploration policy with only text information and change to the vision-based maps during inference. The overall design contributes to a flexible framework which can not only process multi-modality maps and goals but also generalize to diverse goals.

Acknowledgements. This work is supported by Shanghai Artificial Intelligence Laboratory. The research work described in this paper was conducted in the JC STEM Lab of Autonomous Intelligent Systems funded by The Hong Kong Jockey Club Charities Trust.

References

[1] D. Batra, A. Gokaslan, A. Kembhavi, O. Maksymets, R. Mottaghi, M. Savva, A. Toshev, and E. Wijmans, "Objectnav revisited: On evaluation of embodied agents navigating to objects," *arXiv preprint arXiv:2006.13171*, 2020.

[2] J. Krantz, S. Lee, J. Malik, D. Batra, and D. S. Chaplot, "Instance-specific image goal navigation: Training embodied agents to find object instances," *arXiv preprint arXiv:2211.15876*, 2022.

[3] W. Yang, X. Wang, A. Farhadi, A. Gupta, and R. Mottaghi, "Visual semantic navigation using scene priors," *arXiv preprint arXiv:1810.06543*, 2018.

[4] J. Ye, D. Batra, A. Das, and E. Wijmans, "Auxiliary tasks and exploration enable objectnav," *ICCV*, 2021.

[5] D. S. Chaplot, D. P. Gandhi, A. Gupta, and R. R. Salakhutdinov, "Object goal navigation using goal-oriented semantic exploration," *Advances in Neural Information Processing Systems*, vol. 33, pp. 4247–4258, 2020.

[6] R. Ramrakhya, D. Batra, E. Wijmans, and A. Das, "Pirlnav: Pretraining with imitation and rl finetuning for objectnav," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 17 896–17 906.

[7] S. K. Ramakrishnan, D. S. Chaplot, Z. Al-Halah, J. Malik, and K. Grauman, "Poni: Potential functions for objectgoal navigation with interaction-free learning," in *Computer Vision and Pattern Recognition (CVPR), 2022 IEEE Conference on*. IEEE, 2022.

[8] A. J. Zhai and S. Wang, "PEANUT: Predicting and navigating to unseen targets," in *ICCV*, 2023.

[9] K. Zhou, K. Zheng, C. Pryor, Y. Shen, H. Jin, L. Getoor, and X. E. Wang, "Esc: Exploration with soft commonsense constraints for zero-shot object navigation," *arXiv preprint arXiv:2301.13166*, 2023.

[10] B. Yu, H. Kasaei, and M. Cao, "L3mvn: Leveraging large language models for visual target navigation," *arXiv preprint arXiv:2304.05501*, 2023.

[11] P. Wu, Y. Mu, B. Wu, Y. Hou, J. Ma, S. Zhang, and C. Liu, "Voronav: Voronoi-based zero-shot object navigation with large language model," 2024.

[12] J. Chen, G. Li, S. Kumar, B. Ghanem, and F. Yu, "How to not train your dragon: Training-free embodied object goal navigation with semantic frontiers," 2023.

[13] D. Shah, M. Equi, B. Osinski, F. Xia, B. Ichter, and S. Levine, "Navigation with large language models: Semantic guesswork as a heuristic for planning," in *7th Annual Conference on Robot Learning*, 2023.

[14] V. S. Dorbala, J. F. Mullen Jr, and D. Manocha, "Can an embodied agent find your" cat-shaped mug"? llm-guided exploration for zero-shot object navigation," *ICML*, 2023.

[15] M. Minderer, A. Gritsenko, and N. Houlsby, "Scaling open-vocabulary object detection," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[16] S. Zhao, Z. Zhang, S. Schulter, L. Zhao, B. Vijay Kumar, A. Stathopoulos, M. Chandraker, and D. N. Metaxas, "Exploiting unlabeled data with vision and language models for object detection," in *European conference on computer vision*. Springer, 2022, pp. 159–175.

[17] X. Zhou, R. Girdhar, A. Joulin, P. Krähenbühl, and I. Misra, "Detecting twenty-thousand classes using image-level supervision," in *European Conference on Computer Vision*, 2022.

[18] S. Wu, W. Zhang, S. Jin, W. Liu, and C. C. Loy, "Aligning bag of regions for open-vocabulary object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[19] G. Ghiasi, X. Gu, Y. Cui, and T.-Y. Lin, "Scaling open-vocabulary image segmentation with image-level labels," in *European Conference on Computer Vision*, 2022.

[20] M. Xu, Z. Zhang, F. Wei, Y. Lin, Y. Cao, H. Hu, and X. Bai, "A simple baseline for open-vocabulary semantic segmentation with pre-trained vision-language model," in *European Conference on Computer Vision*, 2022.

[21] T. Lüddecke and A. Ecker, "Image segmentation using text and image prompts," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[22] S. Cho, H. Shin, S. Hong, A. Arnab, P. H. Seo, and S. Kim, "Cat-seg: Cost aggregation for open-vocabulary semantic segmentation," 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 *ICML*, 2021.

[24] A. Khandelwal, L. Weihs, R. Mottaghi, and A. Kembhavi, "Simple but effective: Clip embeddings for embodied ai," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[25] A. Majumdar, G. Aggarwal, B. S. Devnani, J. Hoffman, and D. Batra, "ZSON: Zero-shot object-goal navigation using multimodal goal embeddings," in *Advances in Neural Information Processing Systems*, A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, Eds., 2022. [Online]. Available: https://openreview.net/forum?id=VY1dqOF2RjC

[26] 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)*, 2023.

[27] K. Jatavallabhula, A. Kuwajerwala, Q. Gu, M. Omama, T. Chen, S. Li, G. Iyer, S. Saryazdi, N. Keetha, A. Tewari, J. Tenenbaum, C. de Melo, M. Krishna, L. Paull, F. Shkurti, and A. Torralba, "Conceptfusion: Open-set multimodal 3d mapping," *Robotics: Science and Systems (RSS)*, 2023.

[28] K. Yadav, S. K. Ramakrishnan, J. Turner, A. Gokaslan, O. Maksymets, R. Jain, R. Ramrakhya, A. X. Chang, A. Clegg, M. Savva, E. Undersander, D. S. Chaplot, and D. Batra, "Habitat challenge 2022," https://aihabitat.org/challenge/2022/, 2022.

[29] K. Yadav, J. Krantz, R. Ramrakhya, S. K. Ramakrishnan, J. Yang, A. Wang, J. Turner, A. Gokaslan, V.-P. Berges, R. Mootaghi, O. Maksymets, A. X. Chang, M. Savva, A. Clegg, D. S. Chaplot, and D. Batra, "Habitat challenge 2023," 2023.

[30] N. Yokoyama, R. Ramrakhya, A. Das, D. Batra, and S. Ha, "Hm3d-ovon: A dataset and benchmark for open-vocabulary object goal navigation," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024, pp. 5543–5550.

[31] D. S. Chaplot *et al.*, "Learning to explore using active neural SLAM," in *ICLR*, 2020.

[32] S. Chen, T. Chabal, I. Laptev, and C. Schmid, "Object goal navigation with recursive implicit maps," in *Proc. of The International Conference on Intelligent Robots and Systems (IROS)*, 2023.

[33] B. Li, K. Q. Weinberger, S. J. Belongie, V. Koltun, and R. Ranftl, "Language-driven semantic segmentation," *ICLR*, 2022.

[34] S. Y. Gadre, M. Wortsman, G. Ilharco, L. Schmidt, and S. Song, "Cows on pasture: Baselines and benchmarks for language-driven zero-shot object navigation," *CVPR*, 2023.

[35] J. Ma, H. Dai, Y. Mu, P. Wu, H. Wang, X. Chi, Y. Fei, S. Zhang, and C. Liu, "Doze: A dataset for open-vocabulary zero-shot object navigation in dynamic environments," *arXiv preprint arXiv:2402.19007*, 2024.

[36] M. Khanna, R. Ramrakhya, G. Chhablani, S. Yenamandra, T. Gervet, M. Chang, Z. Kira, D. S. Chaplot, D. Batra, and R. Mottaghi, "Goat-bench: A benchmark for multi-modal lifelong navigation," *arXiv preprint arXiv:2404.06609*, 2024.

[37] K. Yadav, R. Ramrakhya, S. K. Ramakrishnan, T. Gervet, J. Turner, A. Gokaslan, N. Maestre, A. X. Chang, D. Batra, M. Savva *et al.*, "Habitat-matterport 3d semantics dataset," in *ICCV*, 2023.

[38] Z. Zeng, A. Röfer, and O. C. Jenkins, "Semantic linking maps for active visual object search," *ICRA*, 2020.

[39] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville, "Film: Visual reasoning with a general conditioning layer," in *Proceedings of the AAAI conference on artificial intelligence*, 2018.

[40] A. Szot, A. Clegg, E. Undersander, E. Wijmans, Y. Zhao, J. Turner, N. Maestre, M. Mukadam, D. Chaplot, O. Maksymets, A. Gokaslan, V. Vondrus, S. Dharur, F. Meier, W. Galuba, A. Chang, Z. Kira, V. Koltun, J. Malik, M. Savva, and D. Batra, "Habitat 2.0: Training home assistants to rearrange their habitat," in *NeurIPS*, 2021.

[41] W. Cai, S. Huang, G. Cheng, Y. Long, P. Gao, C. Sun, and H. Dong, "Bridging zero-shot object navigation and foundation models through pixel-guided navigation skill," *arXiv preprint arXiv:2309.10309*, 2023.

[42] N. Yokoyama, S. Ha, D. Batra, J. Wang, and B. Bucher, "Vlfm: Vision-language frontier maps for zero-shot semantic navigation," in *International Conference on Robotics and Automation (ICRA)*, 2024.

[43] H. Huang, Y. Hao, C. Wen, A. Tzes, Y. Fang *et al.*, "Gamap: Zero-shot object goal navigation with multi-scale geometric-affordance guidance," *Advances in Neural Information Processing Systems*, vol. 37, pp. 39 386–39 408, 2024.

[44] Y. Long, W. Cai, H. Wang, G. Zhan, and H. Dong, "Instructnav: Zero-shot system for generic instruction navigation in unexplored environment," 2024.

[45] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," in *International conference on machine learning*, 2023.

[46] X. Zhai, B. Mustafa, A. Kolesnikov, and L. Beyer, "Sigmoid loss for language image pre-training," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 11 975–11 986.

[47] J. Krantz, T. Gervet, K. Yadav, A. Wang, C. Paxton, R. Mottaghi, D. Batra, J. Malik, S. Lee, and D. S. Chaplot, "Navigating to objects specified by images," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

[48] C. Zhou, C. C. Loy, and B. Dai, "Extract free dense labels from clip," in *European Conference on Computer Vision*. Springer, 2022, pp. 696–712.