OVL-MAP: An Online Visual Language Map Approach for Vision-and-Language Navigation in Continuous Environments

Shuhuan Wen, *Senior Member, IEEE*, Ziyuan Zhang, Yuxiang Sun, and Zhiwen Wang

***Abstract*—Vision-and-Language Navigation in Continuous Environments (VLN-CE) requires agents to navigate 3D environments based on visual observations and natural language instructions. Existing approaches, focused on topological and semantic maps, often face limitations in accurately understanding and adapting to complex or previously unseen environments, particularly due to static and offline map constructions. To address these challenges, this letter proposes OVL-MAP, an innovative algorithm comprising three key modules: an online vision-and-language map construction module, a waypoint prediction module, and an action decision module. The online map construction module leverages robust open-vocabulary semantic segmentation to dynamically enhance the agent's scene understanding. The waypoint prediction module processes natural language instructions to identify task-relevant regions, predict sub-goal locations, and guide trajectory planning. The action decision module utilizes the DD-PPO strategy for effective navigation. Evaluations on the Robo-VLN and R2R-CE datasets demonstrate that OVL-MAP significantly improves navigation performance and exhibits stronger generalization in unknown environments.**

***Index Terms*—Navigation maps, vision-based navigation, multimodal perception, embodied intelligence.**

I. Introduction

THE integration of visual perception and language comprehension is pivotal for intelligent robotic systems, enabling robots to autonomously navigate complex environments based on natural language instructions. the objective of Vision-and-Language Navigation (VLN) is to guide an agent to autonomously reach a specified target from a starting point by understanding and executing natural language instructions, making informed action decisions in complex environments. Traditional VLN research has primarily focused on discrete environments, where agents navigate between predefined graph nodes [1]. Although this simplified setup reduces research complexity, it presents numerous challenges in real-world applications [2]. To enhance the adaptability of navigation agents in realistic scenarios, recent studies have gradually shifted their focus to continuous environments. Krantz et al. [3] introduced Vision-and-Language Navigation in Continuous Environments (VLN-CE), which abandons the idealized assumptions of discrete graphs and adopts a more realistic continuous environment setting, instantiated within the 3D simulator Habitat. Irshad et al. [4] further improved this model by transforming the action space into linear and angular velocities, making the navigation process more akin to the real world.

While the introduction of continuous environments has improved the realism of navigation, the success rate in more complex environments has significantly decreased. Inspired by research in discrete environments, some researchers have attempted to construct topological maps in continuous environments [5], [6]. Although topological maps effectively represent the structure and connectivity of environments, their primary drawback lies in their inability to capture detailed environmental information, such as the precise location of obstacles and the true complexity of the environment. Recent studies [7], [8], [9] have employed top-down semantic maps to model the navigation environment, which can more accurately represent spatial relationships. However, since semantic maps rely on predefined labels, their representational capacity is limited. For instance, they may not cover objects or scenes not included in the predefined labels, and objects with different attributes may not be fully represented due to the lack of detailed attributes in the semantic map.

To address these limitations, we propose an Online Vision-Language Map (OVL-MAP), a method that combines pre-trained visual language features with 3D reconstruction techniques. This approach preserves spatial relationships while integrating more visual detail, thereby assisting the agent in better understanding the scene and making optimized navigation decisions. This letter makes three main contributions: First, we propose an online vision-language map construction method for Robo-VLN and VLN-CE tasks, integrating global spatiotemporal relationships with fine-grained details to enhance scene perception. Second, we introduce an LSTM-based waypoint prediction module that generates sub-goals from spatiotemporal data, thereby improving navigation precision. Third, we develop an action decision module to simulate agent navigation in virtual environments. Our method, evaluated on the Habitat platform,

Received 14 September 2024; accepted 25 January 2025. Date of publication 11 February 2025; date of current version 25 February 2025. This article was recommended for publication by Associate Editor Julie Stephany Berrio Perez and Editor Tetsuya Ogata upon evaluation of the reviewers' comments. This work was supported in part by the National Natural Science Foundation of China under Grant 62273296 and in part by Hebei Innovation Capability Improvement Plan Project under Grant 22567619H. *(Corresponding author: Shuhuan Wen.)*

Shuhuan Wen, Ziyuan Zhang, and Zhiwen Wang are with the Key Laboratory of Intelligent Control and Neural Information Processing, Ministry of Education, Yanshan University, Qinhuangdao 066004, China, and also with the Engineering Research Center of the Ministry of Education for Intelligent Control System and Intelligent Equipment, Yanshan University, Qinhuangdao 066004, China (e-mail: swen@ysu.edu.cn; zhang2023@stumail.ysu.edu.cn; 511409256@qq.com).

Yuxiang Sun is with the Department of Mechanical Engineering, City University of Hong Kong, Kowloon, Hong Kong (e-mail: yx.sun@cityu.edu.hk).

Digital Object Identifier 10.1109/LRA.2025.3540577

2377-3766 © 2025 IEEE. All rights reserved, including rights for text and data mining, and training of artificial intelligence and similar technologies.
Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: CITY UNIV OF HONG KONG. Downloaded on February 25,2025 at 15:04:45 UTC from IEEE Xplore. Restrictions apply.

achieves state-of-the-art navigation success rates on both the Robo-VLN and R2R-CE datasets.

II. Related Work

In this section, we review the progression of VLN and related studies on vision-based map representations for navigation. Compared to previous research, we emphasize the distinctive contributions and features of our study.

Vision and Language Navigation: The task of Vision-and-Language Navigation (VLN) was first introduced by Anderson et al. [1]. and it has since gained significant academic attention with the rapid advancements in embodied intelligence. Early research primarily focused on methods such as data augmentation, search strategies, and pre-training techniques to address challenges in VLN tasks within discrete environments [10], [11], [12], [13]. However, these methods relied on perfect topological localization at each navigation point, which did not effectively tackle the complexities of real-world navigation scenarios, revealing a significant gap between current methods and true embodied intelligence applications.

In recent years, considerable efforts have been made to bridge the gap between discrete and continuous environments. For instance, Krantz et al.'s Sim-2-Sim framework [14] reduced domain gaps during transitions, thus improving VLN performance. Furthermore, Hong [15] proposed a transfer learning framework for continuous environments, addressing the challenges of training agents with natural language instructions through reinforcement learning and multimodal alignment. Meanwhile, innovations such as Wang's structured memory mechanism [16] and Liu's integration of bird's-eye view (BEV) representations with scene graphs [17] have contributed to enhanced semantic understanding and spatial navigation in complex environments. Despite these advancements, several challenges remain. One significant issue is that, while the environment may be continuous, the action space often remains discretized, limiting the complexity and realism of these tasks. To address this limitation, Irshad et al. [4] expanded the VLN-CE framework by introducing Robo-VLN, which incorporates a continuous action space, thereby making the task setup more representative of real-world scenarios.

Recently, the Energy-based Navigation Policy (ENP) framework [18] has explicitly modeled joint state-action distributions and shown promising performance, offering new insights for VLN development. In continuous environments, BEVBert [19] leveraged BEV representations and vision-language pretraining to improve multimodal fusion. However, challenges persist in handling complex environments and aligning features effectively. Similarly, goal-directed semantic exploration [20] has made notable contributions to object recognition and path planning, particularly in dynamic and unknown environments, showing robust adaptability.

While significant progress has been made in VLN within continuous environments, several key challenges remain, particularly in the integration of continuous action spaces and the refinement of multimodal fusion techniques. This letter proposes a framework that integrates continuous action spaces, refines multimodal fusion methods, and enhances semantic understanding, with the goal of improving agent performance in complex environments.

Map for Navigation: Navigation maps play a central role in an agent's environmental understanding and path decision-making, particularly in vision-based navigation tasks. In such tasks, agents typically rely on initial global or local environmental information, such as maps and object features [21]. Methods such as CM2 [8] and WS-MGMAP [9] improved spatial relationship modeling by constructing top-down semantic maps, thereby enhancing the granularity of environmental representation and, to some extent, increasing the success rate of Vision-and-Language Navigation (VLN) tasks. However, these methods were constrained by their reliance on predefined semantic labels, which leads to ineffective representation of unannotated objects and scenes, limiting their applicability. To mitigate this, GridMM [22] introduced Grid Memory Maps, which enhanced task performance by constructing global spatiotemporal relationships and aggregating instruction relevance, although challenges in managing multi-layered environments remain. ETPNav [5] proposed an online mapping approach based on waypoint self-organization, enabling robust long-range planning without prior environmental knowledge, thus improving navigation success. Despite these advancements, existing methods still face challenges, particularly their reliance on predefined semantic labels and limited generalization to unknown environments. To tackle these issues, we propose the OVLMap system, a novel approach for online vision-language map construction. OVLMap integrates spatial occupancy information and semantic prior visual features via the LSeg model, enabling online map updates and dynamic adaptation to environmental changes, thereby significantly enhancing the system's adaptability and generalization.

In recent years, with the continuous advancement of embodied visual-language intelligence, researchers have enhanced the adaptability of agents in complex environments by integrating visual features with linguistic instructions. Several vision-language mapping methods, including ConceptFusion [23], OpenScenes [24], NLMap-SayCan [25], CLIPFields [26] and VLMaps [27], primarily focus on constructing offline maps for scene perception or short-term goal navigation. In contrast, this study combines vision-language mapping with VLN tasks, proposing the OVLMap system, which transforms traditional offline map construction into an online updating framework, enabling dynamic map updates during navigation and avoiding reliance on static pre-built maps. Compared to existing methods, OVLMap offers the following advantages: First, it introduces task-driven viewpoint and semantic fusion, incorporating semantic instruction information into map updates to better meet navigation task requirements. Second, OVLMap processes and updates online streaming data, dynamically adjusting the map according to environmental changes and task needs. Finally, OVLMap integrates waypoint prediction and action decision-making modules, significantly improving the stability and accuracy of long-trajectory navigation, while enhancing cross-modal consistency and overall navigation performance.

III. Method

This method aims to enhance the performance of visual language navigation tasks by dividing the process into three main modules: online visual language map construction, waypoint prdiction, and action decision-making. This modular approach facilitates more effective management of the complexities associated with environmental perception, task instruction parsing, and decision generation. The overall framework of the method is illustrated in the Fig. 1.

Authorized licensed use limited to: CITY UNIV OF HONG KONG. Downloaded on February 25,2025 at 15:04:45 UTC from IEEE Xplore. Restrictions apply.

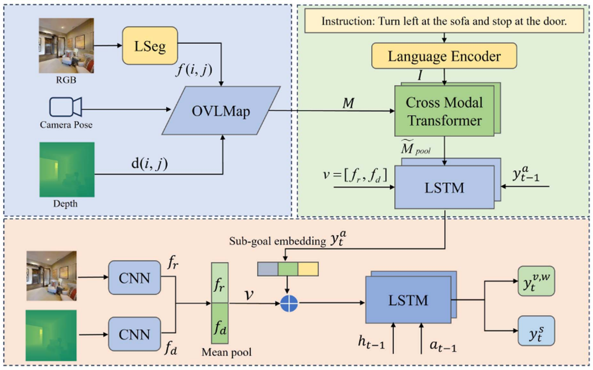


Fig. 1. Method Overview. We utilize the LSeg model to process the RGB images observed by the robot at each time step, extracting dense pixel-level visual language features. These features are then combined with depth information to construct a visual language map in online. Using a Cross-modal Transformer, attention is computed between the map features and natural language instructions to determine the next waypoint. Finally, the agent makes appropriate action decisions based on the waypoint and the RGBD information.

A. Problem Formulation

Following the Robo-VLN approach [4], our goal is to achieve visual language navigation in real-world environments. The agent, equipped with RGB and depth cameras, learns a policy $a_t = \pi(V_t, I, \theta)$ to navigate to a target using visual inputs V_t and instructions I, where θ represents the policy's learnable parameters.

In terms of action space, the Robo-VLN approach includes continuous linear and angular velocities, as well as a stop action. In contrast, the VLN-CE approach features discrete actions such as turning 15°, moving forward 0.25 meters, and stopping. The task is considered successful if the distance to the target is within 3 meters, the stop action is triggered, or the angular velocity drops below a threshold.

B. Construction of Online Visual Language Maps

The Online Visual Language Maps framework integrates features from a pre-trained visual language model (LSeg) with 3D reconstruction data of the environment, thereby enhancing scene understanding. The detailed implementation process is as follows.

Visual Language Feature Extraction: We leverage the pre-trained LSeg encoder to perform semantic segmentation on RGB images captured by the robot at each time step. This process generates semantic information $(q = f(i,j) \in R^{C_s})$ for each pixel (i,j). LSeg is a large-scale visual language model designed for high-precision semantic segmentation driven by language labels. It introduces language-driven semantic priors without requiring manual annotations, thereby demonstrating superior generalization capability. The extracted semantic information from the RGB images is subsequently fused with the environment's geometric information by associating LSeg pixel embeddings with their corresponding 3D map locations, thereby achieving an integration of semantic and geometric data.

Map Construction: Online Visual Language Maps are structured as grid maps, represented by $M \in \mathbb{R}^{H\times W\times D}$, where H and W refer to the map's height and width, and D denotes the dimensionality of the visual language features stored in each grid cell. At each time step, the robot receives new depth images and updates its relative pose. The depth information from each pixel is back-projected into 3D points in the camera's coordinate frame using the intrinsic matrix K and extrinsic matrix T, and then converted into the world coordinate frame. Specifically, all depth pixels $d(i,j)$ are back-projected to generate a local depth point cloud, which is then transformed into the world coordinate system.

$$P = K^{-1} d(i,j)[i,j,1]^T \tag{1}$$

$$P_{world} = TP \tag{2}$$

Here, K represents the depth camera's intrinsic matrix, $d(i,j)$ is the depth value of pixel (i,j), P is the 3D point in the K-th frame, and P_{world} refers to the 3D point in the world coordinate system. Then, P_{world} is projected onto the ground plane to determine the position of pixel (i,j) on the grid map. Next, the visual language features of each pixel (i,j) are stored in the corresponding grid cell, achieving the fusion of semantic and geometric information.

Map Update: The map features are updated according to the following rule:

$$M_t[x,y] = \begin{cases} \hat{M}[x,y], & \text{if } M_{t-1}[x,y] = \text{None}, \\ \frac{\alpha \cdot \hat{M}[x,y] + M_{t-1}[x,y]}{n+1}, & \text{otherwise}. \end{cases} \tag{3}$$

where n denotes the number of features accumulated in the current grid cell, and $[x,y]$ represents the coordinates of the grid cell. The variable $M_t[x,y]$ refers to the global feature at position $[x,y]$,while $\hat{M}[x,y]$ represents the feature vector observed at the same position during time step t, Both $M_t[x,y]$ and $\hat{M}[x,y]$ encode semantic and spatial information. The weight factor α quantifies the relevance of $\hat{M}[x,y]$ to the navigation instruction feature I and is computed as the cosine similarity between them:

$$\alpha = \max\left(0, \frac{\hat{M}[x,y] \cdot I}{\|\hat{M}[x,y]\| \|I\|}\right) \tag{4}$$

This update mechanism prioritizes regions relevant to the navigation task. Semantic instruction information is integrated with observations from new viewpoints to enhance task alignment. By averaging features within each grid cell, the system effectively incorporates multi-perspective observations of the same object while maintaining focus on task-relevant regions.

C. Waypoint Prediction Module

The map constructed in the previous section is updated step by step over time, while the instruction contains the complete navigation trajectory. To capture the navigation progress, we designed an instruction localization module that combines the visual information of the current time step to predict waypoints.

Text Encoder: For a natural language instruction consisting of K words, we utilize the BERT model to extract text features, represented as:

$$I = \{l_t^1, l_t^2, l_t^3, \ldots\ldots l_t^k\} \tag{5}$$

where l_t^i denotes the encoded feature of the i-th word.

Visual Encoder: The observed RGB-D information $(r_t \in R^{h_0 \times W_0 \times 3}, d_t \in R^{h_0 \times w_0})$ is encoded using a pre-trained ConvNet, resulting in RGB features $f_r \in R^{h_s \times w_s \times c_s}$ and depth features $f_d \in R^{h_s \times w_s \times c_s}$.

Authorized licensed use limited to: CITY UNIV OF HONG KONG. Downloaded on February 25,2025 at 15:04:45 UTC from IEEE Xplore. Restrictions apply.

Cross-Modal Inference: The Waypoint Prediction Module employs a cross-modal encoder to align and fuse features from multiple modalities, enabling the agent to perceive and locate itself within the environment. Given the complexity of the navigation environment, many features in the constructed grid map may not be relevant for navigation. The agent requires information that is highly pertinent to the current instruction to understand the environment. Hence, we propose an instruction-based attention mechanism using a Transformer to aggregate key features from the grid, outputting a subgoal y_t^a.

To achieve this, we use the textual features I from the text encoder as the key matrix K and value matrix V, while the grid map features M serve as the query matrix Q. The Transformer attention mechanism then computes the most relevant map features $\tilde{M}$ with respect to the current instruction. The map features $\tilde{M}$ are still represented as a matrix of shape $H \times W \times D$, capturing the spatial and semantic context relevant to the task.

After obtaining the map features $\tilde{M}$, we apply a region-based pooling mechanism to reduce the dimensionality of $\tilde{M}$ by focusing on the areas most relevant to the current task. The pooling operation converts the map feature matrix $\tilde{M}$ from a $H \times W \times D$ matrix into a pooled feature vector $\tilde{M}_{pool}$ of dimensionality $\acute{D}$. The pooled map features are then concatenated with the visual features f_r,depth features f_d and the previous LSTM hidden state h_{t-1}^h to form the input feature vector for the LSTM. This concatenated feature vector is passed into the LSTM network, which processes the sequential information over time. The LSTM output h_t^h is then passed through a fully connected layer to predict the next waypoint y_t^a:

$$h_t^h = LSTM([\tilde{M}_{pool}, f_r, f_d, h_{t-1}^h]) \tag{6}$$

$$y_t^a = W_a h_t^h + b_a \tag{7}$$

Here, W_a is the weight matrix and b_a is the bias term for the fully connected layer that maps the LSTM output to the predicted waypoint y_t^a.

D. Action Decision Module

This study introduces a dual-action module to address the distinct requirements of two Visual-Language Navigation (VLN) tasks: the VLN-CE task and the Robo-VLN task. The method combines reinforcement learning (RL) strategies and deep learning techniques to handle high-level decision-making and low-level control for robot navigation. The module is divided into two sub-modules: for the VLN-CE task, it outputs four discrete actions; for the Robo-VLN task, it generates continuous control signals, such as linear velocity v and angular velocity ω.

Action Module for VLN-CE Task: We adopt DD-PPO as the local strategy, which takes the goal point y_t^a as input and generates a probability distribution over four discrete actions. Action selection is achieved via a policy network, optimized through reinforcement learning to maximize expected rewards. The action probabilities p_a^h are computed as follows:

$$p_a^h = softmax(W_a h_t^h + b_a) \tag{8}$$

where W_a and b_a are learnable parameters. The Softmax function normalizes the LSTM output to produce a valid probability distribution. Actions are sampled or selected from the distribution p_a^h as:

$$a_t^h \sim \pi(a \mid h_t^h) \tag{9}$$

Action Module for Robo-VLN Task: In contrast to the VLN-CE task, the Robo-VLN task requires continuous control signals. We use an LSTM network to predict low-level actions, combining visual features f_r, depth information f_d, and high-level action commands a_t^h from previous decisions to generate the current control signals. The LSTM is defined as:

$$h_t^l = LSTM([f_r, f_d, a_t^h, h_{t-1}^l]) \tag{10}$$

The LSTM output is passed through a fully connected layer g_a and an activation function σ to produce the predicted low-level action probabilities p_a^l (linear and angular velocity) and the stop probability p_a^s:

$$p_a^l, p_a^s = \sigma(g_a([h_t^l, a_{t-1}^l])) \tag{11}$$

Loss Function Design and Training: To optimize the localization and decision-making modules, we employ three distinct loss functions:

Multiclass Cross-Entropy Loss: Measures the difference between the true high-level navigable action y_t^a and the predicted action probability p_a^h:

$$\mathcal{L}_a = -\sum_i y_t^{a_i} \log(p_a^h) \tag{12}$$

Mean Squared Error Loss: Compares the true velocity commands $y_t^{v,\omega}$ with the predicted low-level action probabilities p_a^l:

$$\mathcal{L}_v = \|y_t^{v,\omega} - p_a^l\|^2 \tag{13}$$

Binary Cross-Entropy Loss: Evaluates the difference between the true stop action $y_t^s$and the predicted stop probability p_a^s:

$$\mathcal{L}_s = -\left(y_t^s \log(p_a^s) + (1 - y_t^s) \log(1 - p_a^s)\right) \tag{14}$$

The total loss function is a weighted sum of these individual losses, allowing for simultaneous optimization of both high-level navigation decisions and low-level control actions.

IV. Experiment

This section presents an overview of the experimental setup, including datasets, training configurations, and evaluation metrics. We then compare our method with state-of-the-art approaches in both continuous and discrete action spaces, highlighting its effectiveness. Finally, ablation studies demonstrate the contribution of our method.

A. Experiment Setup

Datasets: Experiments are conducted in the Habitat simulator using the Robo-VLN and R2R-CE datasets. Robo-VLN uses a continuous action space with linear and angular velocities (average trajectory length: 326 steps), while R2R-CE uses a discrete action space with actions like moving forward 0.25 meters, turning 15° left/right, and stop (average trajectory length: 55.8 steps).

Evaluation Metrics: We evaluate the agent's performance using standard visual navigation metrics, including Success Rate (SR), Success weighted by Path Length (SPL), Normalized Dynamic Time Warping (NDTW), Trajectory Length (TL), Oracle

Authorized licensed use limited to: CITY UNIV OF HONG KONG. Downloaded on February 25,2025 at 15:04:45 UTC from IEEE Xplore. Restrictions apply.

TABLE I
COMPARISON OF OUR MODEL WITH EXISTING MODELS (ROBO-VLN DATASET)

Method	Val-Seen					Val-Unseen				
	SR↑	SPL↑	NDTW↑	TL↓	NE↓	SR↑	SPL↑	NDTW↑	TL↓	NE↓
Seq2Seq [28]	36	34	32	11.84	8.63	33	30	28	11.92	8.97
PM [29]	32	27	23	14.12	9.33	28	24	22	13.85	9.82
CMA [30]	28	25	22	11.52	9.95	28	25	23	11.57	9.63
HCM [4]	49	43	35	13.53	7.48	46	40	35	14.06	7.94
Ours(OVL-MAP)	**57**	**53**	**39**	12.36	**7.26**	**55**	**52**	35	14.54	7.98

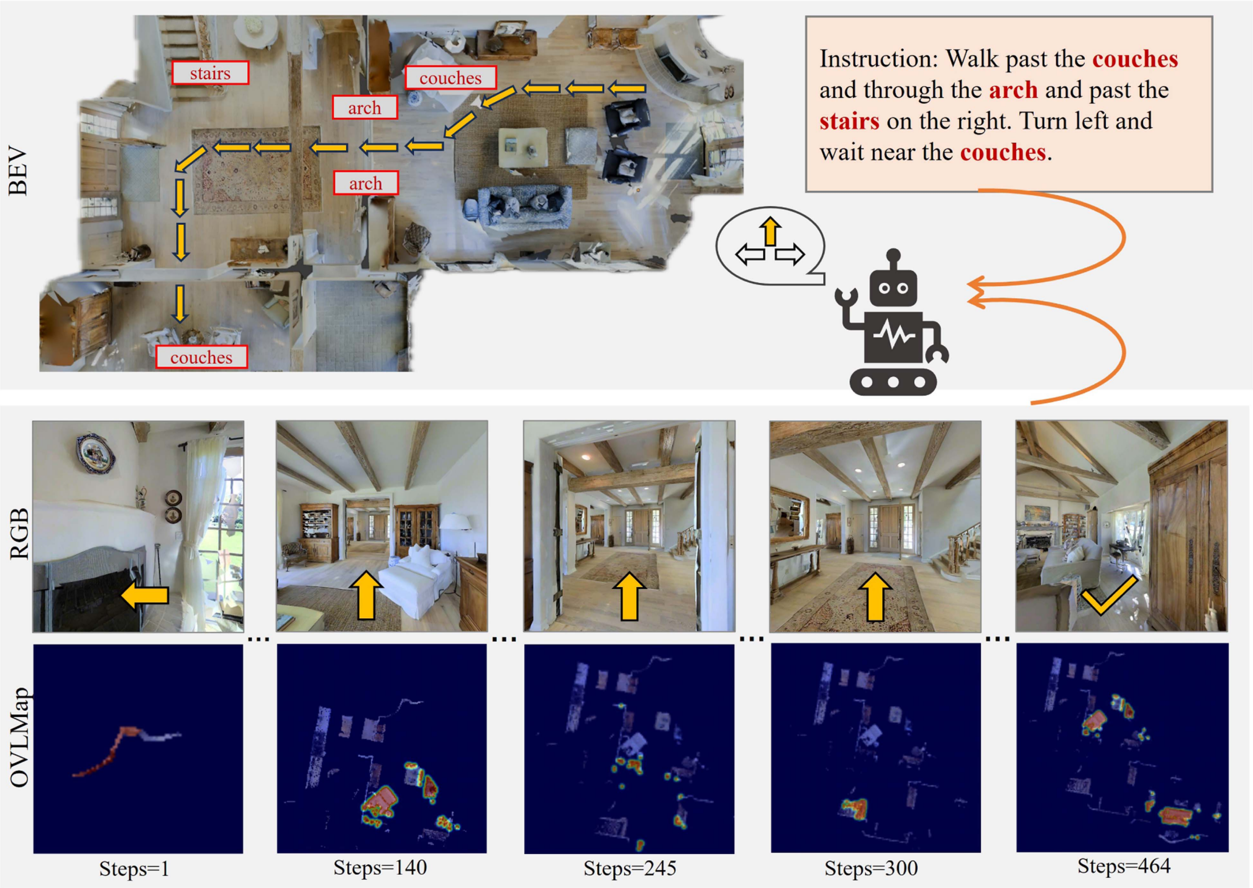

BEV
stairs
couches
arch
arch
couches
Instruction: Walk past the couches and through the arch and past the stairs on the right. Turn left and wait near the couches.
RGB
OVLMap
Steps=1
Steps=140
Steps=245
Steps=300
Steps=464

Fig. 2. Visualization of a successful navigation episode in a complex task. The first row presents a bird's-eye view with red boxes marking the predicted sub-goals. The second row displays the RGB images observed by the agent, with yellow arrows indicating the movement direction and path. The third row shows the 2D plane view of the visually evolving language map constructed online during navigation.The highlighted areas represent the regions of the map most relevant to the current view and task instructions. Specifically, they correspond sequentially to the objects in the bird's-eye view, including "couches," "arch," "stairs," and "couches." This illustrates how the map updates incrementally throughout the navigation process, reflecting online changes and progress toward the task objectives.

Success Rate (OSR), and Navigation Error (NE). SPL and SR serve as the primary indicators of navigation performance. For detailed descriptions of these metrics, please refer to [31], [32], [33].

B. Performance Comparison

Comparison of Robo-VLN Baselines: Table I compares our method to the Robo-VLN baseline on the Robo-VLN dataset, showing that our approach significantly outperforms the baseline across several metrics. Notably, we achieve a 10% improvement in SPL for seen environments and a 12% increase for unseen environments. This enhancement is largely due to our method's visual language map, which boosts the agent's generalization in novel settings. Fig. 2 illustrates the agent's performance in a navigation episode, where it dynamically constructs a visual language map from online RGB-D data. The localization module uses this map to identify areas relevant to the given natural language instructions, determining sub-goal points such as "couches," "arch," and "stairs." The agent navigates accurately to these points and reaches the final destination, demonstrating the method's effectiveness in spatial understanding and executing instructions in complex environments.

Authorized licensed use limited to: CITY UNIV OF HONG KONG. Downloaded on February 25,2025 at 15:04:45 UTC from IEEE Xplore. Restrictions apply.

TABLE II
COMPARISON OF OUR MODEL WITH EXISTING MODELS (R2R-CE DATASET)

Method	Val Seen					Val Unseen					Test Unseen				
	TL↓	NE↓	OSR↑	SR↑	SPL↑	TL↓	NE↓	OSR↑	SR↑	SPL↑	TL↓	NE↓	OSR↑	SR↑	SPL↑
SASRA [7]	8.89	7.17	-	36.0	34.0	7.89	8.32	-	24.0	22.0	-	-	-	-	-
CM2 [8]	12.05	6.10	50.7	42.9	34.8	11.54	7.02	41.5	34.3	27.6	13.9	7.7	39	31	24
WS-MGMAP [9]	10.12	5.65	51.7	46.9	43.4	10.00	6.28	47.6	38.9	34.3	12.30	7.11	45	35	28
GridMM [22]	12.69	4.21	69	59	51	13.36	5.11	61	49	41	13.31	5.64	56	46	39
ETPNav [5]	11.78	3.95	72	66	59	11.99	4.71	65	57	49	12.87	5.12	63	55	48
Safe-VLN [6]	13.71	3.35	79	71	60	15.00	4.48	68	60	47	15.44	5.01	64	56	45
ENP-ETPNav [18]	11.82	3.90	73	68	59	11.45	4.69	65	58	50	12.71	5.08	64	56	48
Ours(OVL-MAP)	11.89	3.88	71	70	59	12.12	4.62	64	**60**	**50**	**11.64**	**4.98**	62	**57**	**48**

TABLE III
ABLATION EXPERIMENTS OF OVL-MAP ON THE R2R-CE VAL-UNSEEN DATASET

Method	Module		Val-Unseen				
	OVLMap	Waypoint prediction	TL↓	NE↓	OSR↑	SR↑	SPL↑
OVL-MAP	×	×	13.84	9.11	36	22	20
	✓	×	12.37	6.90	43	36	22
	✓	✓	12.12	4.62	64	60	50

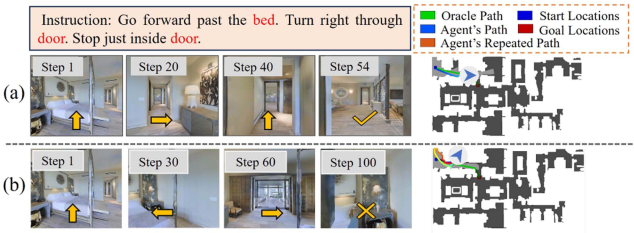


Fig. 3. Qualitative Analysis of the Impact of OVLMap and Waypoint prediction Modules on Navigation Performance.This figure compares the navigation performance of our proposed method with and without the map and waypoint prediction modules. Panel (a) illustrates the complete framework with both modules, where the agent successfully identifies and navigates towards key targets like "bed" and "door", reaching the goal via the shortest path. Panel (b) shows a framework lacking these modules, where the agent, relying solely on RGB-D and instruction input, fails to correctly associate visual cues with instructions, leading to errors and eventually getting trapped in a loop, resulting in navigation failure.

Comparison of VLN-CE baselines: To further validate the effectiveness of our proposed OVL-MAP method, we compared it with state-of-the-art approaches on the R2R-CE dataset,as shown in Table II. Our model outperforms existing baselines in most evaluation metrics, particularly in success rate(SR) and success weighted path length(SPL). Compared to methods relying on fixed semantic labels, such as CM2 and WS-MGMap, our model surpasses them and shows approximately 10% improvement in SR and SPL over GridMM. Its performance is on par with ETPNav and Safe-VLN. In the Validation Unseen set, our method achieved 60% SR, matching Safe-VLN, but improved SPL by 3%, from 47% to 50%, indicating better path optimization. Moreover, the path length (TL) was reduced by 2.8 compared to Safe-VLN, demonstrating the ability to complete the task with shorter paths. In the Test Unseen set, our method achieved 48% SPL, comparable to ETPNav, with a 2% improvement in SR, from 55% to 57%. This improvement is attributed to the semantic priors and the online construction of vision-language maps. By combining BERT and LSTM networks for information processing, our method enhanced performance in unseen environments. It is worth noting that the performance of the ENP-ETPNav model was similar across metrics when compared to our model, highlighting the importance of energy-based navigation strategies (ENP) in vision-language navigation tasks. This suggests that incorporating ENP into our model could further enhance performance. Overall, our method demonstrates superior generalization, especially in unseen environments, emphasizing the importance of semantic priors and vision-language fusion for improved navigation capabilities.

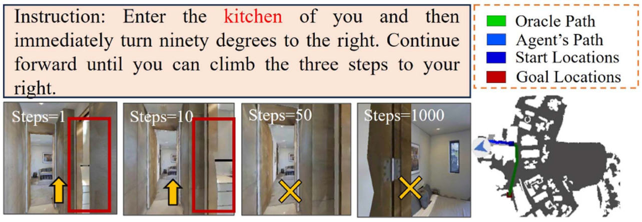
C. Ablation Study.

The Impact of OVLMap and Waypoint Prediction Modules on Navigation Performance: An ablation study was conducted on the unseen validation split of the R2R-CE dataset to evaluate the contributions of the OVLMap and waypoint prediction modules to navigation performance. The results, shown in Table III, reveal that, when only RGB-D data (f_r, f_d) is input into the LSTM network (first row), navigation performance is inferior to when map features (M) are included without attention-based instruction processing (second row). This comparison highlights the superiority of using an online visual language map constructed from RGB-D data. However, not all map features are essential for successful navigation, as demonstrated by the improved results when attention mechanisms are applied in the waypoint prediction module (third row), which enhances navigation success rates by focusing on the most relevant map features.

A qualitative analysis further supports these findings, as shown in Fig. 3. Fig. 3(a) illustrates the full framework, where both the OVLMap and waypoint prediction modules enable the agent to successfully identify targets like "bed" and "door" and reach the goal efficiently. In contrast, Fig. 3(b) depicts the framework without these modules, where the agent, relying solely on RGB-D data and instructions, begins to make errors at step 30. The agent fails to link the "door" in its view to the target, leading to a looping behavior and eventual navigation failure, as highlighted by the overhead view in Fig. 3(b).

Despite its advantages, the OVL-MAP method has some limitations. As shown in Fig. 4, navigation failure occurs when the agent, due to a restricted field of view, cannot properly identify kitchen-related objects. The red box in the figure highlights that the agent perceives only part of the sink related to the "kitchen", limiting its ability to fully perceive the environment. As a result, the agent moves along an incorrect trajectory, resulting in navigation failure. Future work could address this limitation through panoramic views or forward-looking exploration strategies to broaden the agent's field of view. Additionally,

Authorized licensed use limited to: CITY UNIV OF HONG KONG. Downloaded on February 25,2025 at 15:04:45 UTC from IEEE Xplore. Restrictions apply.



Instruction: Enter the kitchen of you and then immediately turn ninety degrees to the right. Continue forward until you can climb the three steps to your right.

Fig. 4. Navigation failure caused by restricted field of view, Leading to incomplete object recognition and incorrect trajectory.

TABLE IV
Comparison Among Different Maps on the R2R-CE Val-Unseen Dataset

Mapping methods	TL↓	NE↓	OSR↑	SR↑	SPL↑
Semantic map [8]	12.80	6.24	42.8	33	29
Multi-Granularity map [9]	12.34	6.18	49.9	40	35
OVLMap	12.12	4.62	64.0	60	50

the current method struggles with multi-level environments, warranting further investigation.

Comparison with Semantic Map Approaches: We conducted a systematic comparison between the proposed OVLMap method and existing explicit semantic map approaches, as summarized in Table IV. The results demonstrate that OVLMap outperforms baseline methods across most key metrics. The baseline methods include the CM2-based approach [8], which generates self-centered top-down semantic maps using convolutional layers, and the WS-MGMap method [9], which integrates color and other detailed information into CM2 maps. However, both methods rely heavily on predefined semantic labels, significantly limiting their adaptability to unseen categories. OVLMap exhibits several notable advantages. By integrating the LSeg model with depth information and camera pose data, it provides stronger semantic priors compared to the UNet-based feature extraction employed in CM2 and WS-MGMap, enabling superior spatial representation in complex environments. In contrast, CM2 and WS-MGMap are constrained by fixed-category semantic features, reducing their capability to handle novel categories. Additionally, OVLMap supports dynamic online map updates based on task instructions, enhancing task-specific adaptability and operational flexibility. In comparison, CM2 and WS-MGMap perform waypoint predictions on static maps of fixed size, lacking the ability to adjust dynamically. Furthermore, OVLMap integrates waypoint prediction and action decision-making modules and leverages LSTM networks to effectively handle long-term dependencies, outperforming the GRU networks used in WS-MGMap in modeling capacity. The results in Table IV further validate OVLMap's superior performance in vision-language navigation tasks, highlighting its accuracy and adaptability to complex environments.

Impact of Map Scale on Navigation Performance: The results (Table V) show that increasing the map scale improves navigation performance, though it also increases computational cost, with diminishing returns in performance gain. Therefore, a map scale of 20× 20 is selected as the optimal configuration.

TABLE V
The Effect of Different OVL-Map Scale on the R2R-CE Val-Unseen Dataset

Map scale	TL↓	NE↓	OSR↑	SR↑	SPL↑
15×15	13.64	6.61	59.2	49.96	38.69
20×20	12.12	4.62	64.0	60.02	50.09
25×25	11.54	4.14	58.7	60.84	51.32

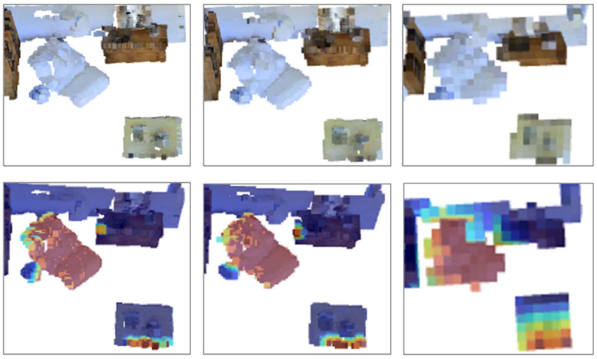


Fig. 5. Visualization of OVLMap's local maps at different resolutions. From left to right, the grids correspond to resolutions of 3 cm, 5 cm, and 10 cm. The second row illustrates the impact on waypoint prediction, showing that lower resolutions lead to less accurate predictions.

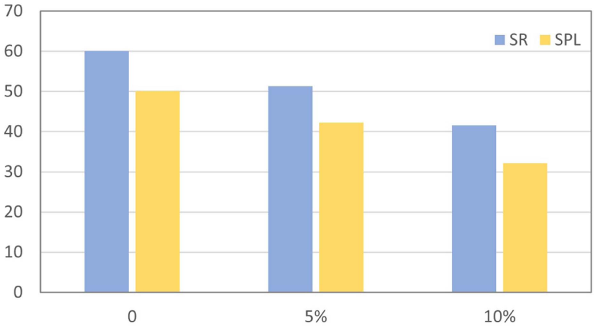


Fig. 6. The impact of sub-goal errors on navigation performance. The x-axis represents the error magnitude, with α taking values of 0%, 5%, and 10%.

Additionally, higher grid resolutions (eg.3 cm) significantly enhance sub-goal prediction accuracy, while lower resolutions (eg.10 cm) result in blurred predictions, impacting navigation precision(Fig. 5).

Impact of Waypoint Prediction Errors: We further analyzed the impact of waypoint prediction errors and action executors on navigation performance. Specifically, we introduced different levels of noise error to the first LSTM output y_t^a by modifying it as $y_t^a = y_t^a + \alpha \cdot N(0, \sigma^2)$. The results show that increasing prediction errors significantly degrades performance (see Fig 6). In terms of action executors, both our method and the method in [15] use a point navigation strategy, which outperforms the FMM method in [14], but slightly trails behind the RF method in [5]. This demonstrates the effectiveness of point navigation strategies, as previously studied in [5].

V. Conclusion

In this letter, we focused on the navigation capabilities of agents in continuous environments and proposed an online

Authorized licensed use limited to: CITY UNIV OF HONG KONG. Downloaded on February 25,2025 at 15:04:45 UTC from IEEE Xplore. Restrictions apply.

method for constructing vision-and-language navigation maps, OVL-MAP. First, we introduced robust semantic priors to construct visual language maps, thereby enhancing the agent's environmental perception. Secondly, we designed an attention-based waypoint prediction module that outputs sub-goals, and finally, we used an action decision-making module to guide the agent's navigation. In the future, we will consider how to mitigate the impact of limited visibility and explore how to apply the proposed OVL-MAP method to multi-story buildings, and subsequently deploy it in real robotic systems to evaluate its navigation performance.

References

[1] P. Anderson et al., "Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments," in *Proc. 2018 IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3674–3683.

[2] J. Gu, E. Stefani, Q. Wu, J. Thomason, and X. Wang, "Vision-and-language navigation: A survey of tasks, methods, and future directions," in *Proc. 60th Annu. Meeting Assoc. Comput. Linguistics (Volume 1: Long Papers)*, S. Muresan, P. Nakov, and A. Villavicencio, Eds. Dublin, Ireland, May 2022, pp. 7606–7623.

[3] J. Krantz, D. Chen, D. Batra, A. Schwing, D. Parikh, and M. Savva, "Beyond the Nav-graph: Vision and language navigation in continuous environments," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 104–120.

[4] M. Z. Irshad, C.-Y. Ma, and Z. Kira, "Hierarchical cross-modal agent for robotics vision-and-language navigation," in *Proc. 2021 IEEE Int. Conf. Robot. Automat.*, 2021, pp. 13238–13246.

[5] D. An et al., "ETPNav: Evolving topological planning for vision-language navigation in continuous environments," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–16, Apr. 2024, doi: 10.1109/TPAMI.2024.3386695.

[6] L. Yue, D. Zhou, L. Xie, F. Zhang, Y. Yan, and E. Yin, "Safe-VLN: Collision avoidance for vision-and-language navigation of autonomous robots operating in continuous environments," *IEEE Robot. Automat. Lett.*, vol. 9, no. 6, pp. 4918–4925, Jun. 2024.

[7] M. Z. Irshad, N. C. Mithun, Z. Seymour, H.-P. Chiu, S. Samarasekera, and R. Kumar, "Semantically-aware spatio-temporal reasoning agent for vision-and-language navigation in continuous environments," in *Proc. 26th Int. Conf. Pattern Recognit.*, 2022, pp. 4065–4071.

[8] G. Georgakis et al., "Cross-modal map learning for vision and language navigation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 15460–15470.

[9] P. Chen et al., "Weakly-supervised multi-granularity map learning for vision-and-language navigation," *Adv. Neural Inf. Process. Syst.*, vol. 35, pp. 38149–38161, 2022.

[10] S. Wen, X. Lv, F. R. Yu, and S. Gong, "Vision-and-language navigation based on cross-modal feature fusion in indoor environment," *IEEE Trans. Cogn. Develop. Syst.*, vol. 15, no. 1, pp. 3–15, Mar. 2023.

[11] J. Lu, D. Batra, D. Parikh, and S. Lee, "Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 13–23.

[12] D. Fried et al., "Speaker-follower models for vision-and-language navigation," in *Proc. Adv. Neural Inf. Syst. 31: Annu. Conf. Neural Inf. Process. Syst. 2018*, vol. 31, Montreal, Canada, Dec. 2018, pp. 3318–3329.

[13] H. Tan, L. Yu, and M. Bansal, "Learning to navigate unseen environments: Back translation with environmental dropout," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, 2019, pp. 2610–2621.

[14] J. Krantz and S. Lee, "Sim-2-sim transfer for vision-and-language navigation in continuous environments," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 588–603.

[15] Y. Hong, Z. Wang, Q. Wu, and S. Gould, "Bridging the gap between learning in discrete and continuous environments for vision-and-language navigation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 15439–15449.

[16] H. Wang, W. Wang, W. Liang, C. Xiong, and J. Shen, "Structured scene memory for vision-language navigation," in *Proc. 2021 IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 8451–8460.

[17] R. Liu, X. Wang, W. Wang, and Y. Yang, "Bird's-eye-view scene graph for vision-language navigation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2023, pp. 10968–10980.

[18] R. Liu, W. Wang, and Y. Yang, "vision-language navigation with energy-based policy," in *Proc. Adv. Neural Inf. Process. Syst. 38: Annu. Conf Neural Inf. Process. Syst. 2024*, Vancouver, BC, Canada, Dec. 2024.

[19] D. An et al., "BevBert: Multimodal map pre-training for language-guided navigation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 2737–2748.

[20] D. S. Chaplot, D. Gandhi, A. Gupta, and R. Salakhutdinov, "Object goal navigation using goal-oriented semantic exploration," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, vol. 33, pp. 4247–4258.

[21] X. Liu, S. Wen, J. Zhao, T. Z. Qiu, and H. Zhang, "Edge-assisted multi-robot visual-inertial SLAM with efficient communication," *IEEE Trans. Automat. Sci. Eng.*, vol. 22, pp. 2186–2198, 2025.

[22] Z. Wang, X. Li, J. Yang, Y. Liu, and S. Jiang, "GridMM: Grid memory map for vision-and-language navigation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 15625–15636.

[23] K. Jatavallabhula et al., "Conceptfusion: Open-set multimodal 3D mapping," in *Proc. Rob. Sci. Syst. XIX*, Daegu, Republic of Korea, Jul. 2023, doi: 10.15607/RSS.2023.XIX.066.

[24] S. Peng, K. Genova, C. M. Jiang, A. Tagliasacchi, M. Pollefeys, and T. Funkhouser, "Openscene: 3D scene understanding with open vocabularies," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 815–824.

[25] B. Chen et al., "Open-vocabulary queryable scene representations for real world planning," in *Proc. 2023 IEEE Int. Conf. Robot. Automat.*, 2023, pp. 11509–11522.

[26] N. M. Shafiullah, C. Paxton, L. Pinto, S. Chintala, and A. Szlam, "Clip-fields: Weakly supervised semantic fields for robotic memory," in *Proc. Robot. Sci. Syst. XIX*, Daegu, Republic of Korea, Jul. 2023, doi: 10.15607/RSS.2023.XIX.074.

[27] C. Huang, O. Mees, A. Zeng, and W. Burgard, "Visual language maps for robot navigation," in *Proc. 2023 IEEE Int. Conf. Robot. Automat.*, 2023, pp. 10608–10615.

[28] A. Chang et al., "Matterport3D: Learning from RGB-D data in indoor environments," in *Proc. Int. Conf. 3D Vis.*, 2017, pp. 667–676.

[29] C. Ma et al., "Self-monitoring navigation agent via auxiliary progress estimation," in *Proc. 7th Int. Conf. Learn. Representations*, New Orleans, LA, USA, May 2019. [Online]. Available: https://openreview.net/forum?id=r1GAsjC5Fm

[30] X. Wang et al., "Reinforced cross-modal matching and self-supervised imitation learning for vision-language navigation," in *Proc. IEEE/CVF Conf. Comput. Vis. pattern Recognit.*, 2019, pp. 6629–6638.

[31] G. I. Magalhaes, V. Jain, A. Ku, E. Ie, and J. Baldridge, "General evaluation for instruction conditioned navigation using dynamic time warping," in *Proc. Visually Grounded Interact. Lang. NeurIPS 2019 Workshop*, Vancouver, Canada, Dec. 2019. [Online]. Available: https://vigilworkshop.github.io/static/papers/33.pdf

[32] A. B. Vasudevan, D. Dai, and L. Van Gool, "Talk2Nav: Long-range vision-and-language navigation with dual attention and spatial memory," *Int. J. Comput. Vis.*, vol. 129, no. 1, pp. 246–266, 2021, doi: 10.1007/s11263-020-01374-3.

[33] W. Hao, C. Li, X. Li, L. Carin, and J. Gao, "Towards learning a generic agent for vision-and-language navigation via pre-training," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 13134–13143.

Authorized licensed use limited to: CITY UNIV OF HONG KONG. Downloaded on February 25,2025 at 15:04:45 UTC from IEEE Xplore. Restrictions apply.