FUTURENAV: UNIFIED WORLD-ACTION MODELING FOR VISION-AND-LANGUAGE NAVIGATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision-and-language navigation (VLN) in continuous environments requires an agent to ground instructions in egocentric observations while maintaining spatial understanding across long action sequences. Recent navigation foundation models have shown strong progress by scaling vision-language models (VLMs), but they often learn navigation primarily as direct action generation, without explicitly modeling world states or predicting their future evolution. We introduce ***FutureNav***, a VLM-based unified world-action modeling framework for vision-and-language navigation. Specifically, ***FutureNav*** jointly encodes text, visual, and spatial features and feeds them into the LLM, and optimizes four objectives for simultaneous world and action modeling: an action policy objective for navigation action prediction, inverse and forward dynamics objectives for modeling state transitions, and a future generation objective for predicting future spatial states. This unified architecture strengthens action prediction while explicitly modeling the world, without sacrificing inference speed. Extensive experiments show that, with only a 4B-scale backbone, ***FutureNav*** achieves state-of-the-art performance on multiple VLN benchmarks and substantially outperforms prior VLN methods, paving the way toward future world-action models for VLN. We will release the code and models to support future research.

1 INTRODUCTION

Vision-and-language navigation (VLN) requires an embodied agent to follow natural language instructions and navigate to target locations in unseen 3D environments Anderson et al. (2018); Krantz et al. (2020); Ku et al. (2020). In continuous environments, this task is especially challenging because the agent must ground linguistic goals in egocentric observations, retain observation history, maintain spatial understanding over long action sequences, and execute low-level actions without accumulating irreversible errors. Early VLN-CE systems therefore relied on explicit navigation structures, such as waypoint prediction, topological planning, semantic maps, and learned spatial memories, to connect language grounding with physically executable actions Krantz et al. (2021); Hong et al. (2022; 2023); Wang et al. (2023b;a); Zhang et al. (2025b); Liu et al. (2026b); Zhang et al. (2024c; 2025c); Gong et al. (2026); Zhang et al. (2025e; 2026c); Kong et al. (2026); Zhang et al. (2025d); Xiao et al. (2025); Hao et al. (2025a; 2026c;b); Zhang et al. (2026e). These methods indicate that successful embodied navigation requires more than recognizing objects or instruction phrases: an agent must also understand the state of the surrounding world and how that state evolves under its actions.

Recent progress has increasingly shifted VLN toward large vision-language models (VLMs) and vision-language-action foundation models. Methods such as NaVid Zhang et al. (2024b), Uni-NaVid Zhang et al. (2024a), NaVILA Cheng et al. (2024), StreamVLN Wei et al. (2025b), and Embodied Navigation Foundation Model Zhang et al. (2026a) adapt pretrained multimodal backbones to predict navigation actions from language instructions and visual histories Zhang et al. (2025a); BAAI RoboBrain Team et al. (2025); Hao et al. (2025c); Zhang et al. (2026d); Lu et al. (2026); Zhang et al. (2025g; 2026f; 2025f); Tang et al. (2025); Hao et al. (2025b); Wu et al. (2025); Cheng et al. (2025); Li et al. (2025); Hao et al. (2026a); Zhang et al. (2026b;g). JanusVLN Zeng et al. (2026) further separates semantic and spatial information with dual implicit memories, demonstrating strong performance without external navigation data. These VLM-based methods provide a scalable action-decoding interface for VLN, but most of them still formulate navigation primarily as direct

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

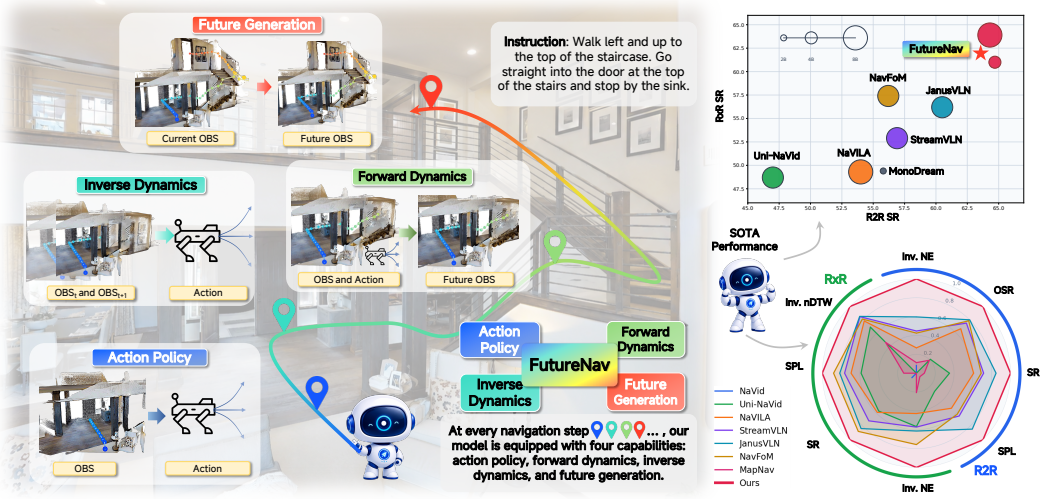

Figure 1: Overview of ***FutureNav***. Our model unifies four capabilities in a VLM-based world-action framework: action policy learning for direct navigation control, inverse dynamics for inferring actions from observation changes, forward dynamics for action-conditioned state transition modeling, and future generation for predicting upcoming spatial states. This unified design improves world understanding and action prediction while achieving state-of-the-art performance on VLN-CE benchmarks.

observation-to-action generation. Consequently, they do not explicitly supervise world-state representation, action-induced state transition modeling, or future-state prediction before action selection. In parallel, recent world-model navigation systems, including Navigation World Models (NWM) Bar et al. (2025), WorldVLN Zhao et al. (2026), WAM-Nav Yang et al. (2026), NavForesee Liu et al. (2025a), AstraNav-World Hu et al. (2025), and NavMorph Yao et al. (2025), have explored future observation prediction, latent visual foresight, or world-action modeling for navigation Liu et al. (2026c;a); Ma et al. (2026); Fu et al. (2026); Li et al. (2026a). However, their world-action modeling remains incomplete: policy learning, action-conditioned dynamics, and future-state generation are often treated as separate components or only partially modeled, rather than being unified within a single navigation policy. As a result, these methods do not fully couple what action to take with how the world state changes and what future state the action may induce. Moreover, their world-modeling modules often remain active during inference, incurring additional computation for future-state imagination, candidate rollouts, or auxiliary planning.

To bridge this gap, we propose ***FutureNav***, a VLM-based unified world-action modeling framework for continuous VLN. As illustrated in Figure 1, ***FutureNav*** augments a VLM backbone with explicit spatial features from a frozen spatial encoder. Given the instruction and egocentric observation history, the VLM backbone encodes text and visual tokens, while the spatial encoder extracts geometry-aware features from the same observations. A lightweight trainable merger projects the spatial features into the LLM hidden space and fuses them with visual tokens through residual addition. The LLM therefore receives a unified multimodal sequence containing text, visual, and spatial features, and predicts low-level navigation actions autoregressively. This design retains a single-pass action-decoding inference pathway while giving the policy explicit access to spatial state information.

FutureNav further trains the shared architecture with four complementary objectives for unified world-action modeling. The action policy autoregressively predicts the next navigation action from the instruction and geometry-enhanced visual context. Inverse dynamics infers the action connecting two adjacent observations, encouraging the representation to capture action-relevant visual changes. Forward dynamics predicts the next spatial state conditioned on the current observation and executed action, while future generation predicts upcoming spatial states without explicit action conditioning. Together, these objectives extend conventional action imitation with supervision over observed transitions, action-conditioned dynamics, and short-horizon world evolution. At inference, ***FutureNav*** retains only the action policy for direct action decoding, introducing no additional computation from the auxiliary world-modeling heads.

Extensive experiments on standard continuous VLN benchmarks demonstrate the effectiveness and efficiency of ***FutureNav***. With only a 4B-scale backbone, ***FutureNav*** achieves state-of-the-

art performance on multiple VLN benchmarks and substantially outperforms prior VLN methods, including larger 7B/8B-scale navigation models. We further demonstrate strong zero-shot real-world vision-and-language navigation capability without real-world fine-tuning. Our contributions are summarized as follows:

- We propose ***FutureNav***, a VLM-based unified world-action modeling framework for continuous VLN that equips a single LLM action-decoding policy with action prediction, inverse dynamics, forward dynamics, and future spatial-state generation capabilities.
- We introduce four complementary world-action training objectives that jointly supervise policy learning, action-induced state transitions, inverse action inference, and future-state prediction, enabling the model to learn both what action to execute and how the state evolves.
- We show through extensive experiments that our unified world-action modeling significantly improves navigation performance while preserving inference efficiency, enabling a 4B model to outperform prior larger VLN foundation models.

2 Related Work

2.1 Vision-and-Language Navigation

Vision-and-Language Navigation (VLN) requires an embodied agent to follow language instructions to a goal and is commonly evaluated on R2R Anderson et al. (2018), RxR Ku et al. (2020), and VLN-CE Krantz et al. (2020). Recent approaches increasingly adopt end-to-end VLM policies. NaVid Zhang et al. (2024b) predicts actions from monocular RGB and instructions without maps, odometry, or depth. Uni-NaVid Zhang et al. (2024a) and NaVILA Cheng et al. (2024) extend this paradigm to multi-task navigation, while NavA^3 Zhang et al. (2025a) handles open-ended instructions across diverse scenes. StreamVLN Wei et al. (2025b), MapNav Zhang et al. (2025b), and JanusVLN Zeng et al. (2026) improve streaming efficiency and long-horizon memory through slow-fast context, semantic maps, and decoupled implicit memory, respectively, while NavFoM Zhang et al. (2026a) scales VLN across embodiments. However, these methods primarily map observations directly to actions without explicitly modeling world states, action-induced transitions, or spatial futures. This limited world-level supervision motivates our unified world-action formulation.

2.2 World Modeling for Embodied Navigation

World models predict action-conditioned environment dynamics to support planning through imagined rollouts. PathDreamer Koh et al. (2021) synthesizes unseen views, Navigation World Models Bar et al. (2025) generates videos to rank candidate trajectories, and VISTAv2 Huang et al. (2025) converts egocentric futures into value maps. To avoid costly pixel-level generation, recent methods operate in latent space: NavMorph Yao et al. (2025) predicts future visual embeddings, while NavCoT Lin et al. (2025) imagines the next observation as an intermediate reasoning step. Other approaches integrate prediction and action: NavForesee Liu et al. (2025a) combines language planning with predictive imagination, and MonoDream Wang et al. (2025b) uses latent panoramic RGB-D prediction as auxiliary supervision Liu et al. (2026c;a); Ma et al. (2026); Yuan et al. (2026); Li et al. (2026b). However, existing methods often separate prediction from policy learning, require depth supervision, or capture only partial action–observation relations. In contrast, our ***FutureNav*** jointly learns action policy, forward and inverse dynamics, and future generation in a 3D-aware, instruction-conditioned latent framework.

3 Methodology

3.1 Overview

We formulate continuous vision-and-language navigation as conditional action generation augmented with auxiliary world modeling. At time step t, the agent receives a natural language instruction x and an egocentric observation history $O_t = \{I_1, \ldots, I_t\}$, where each I_i is an RGB frame, and predicts the next low-level action $a_t \in \mathcal{A}$, with $\mathcal{A} = \{\texttt{STOP}, \texttt{MOVE_FORWARD}, \texttt{TURN_LEFT}, \texttt{TURN_RIGHT}\}$. We tokenize the instruction, visual observations, and previous actions into a multimodal sequence and autoregressively decode the next action as text.

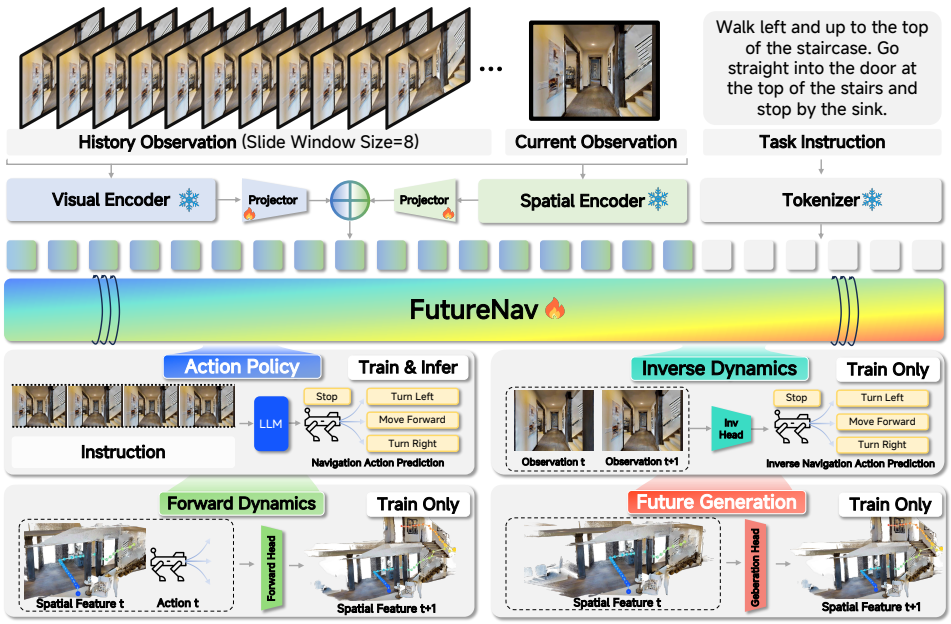

Figure 2: Architecture of ***FutureNav***. A VLM backbone encodes the instruction and egocentric observations, while a frozen spatial encoder provides spatial-aware features for world-state understanding. The model supports four world-action capabilities: action policy learning for navigation control, inverse dynamics for inferring actions from observation changes, forward dynamics for predicting action-conditioned state transitions, and future generation for predicting future spatial states.

As illustrated in Figure 2, ***FutureNav*** builds on a VLM backbone and introduces a frozen spatial encoder for world-state modeling. The VLM visual encoder and spatial encoder process the same observation sequence to extract visual tokens and spatial features, respectively. A lightweight trainable merger projects the spatial features into the language-model hidden space and fuses them with the visual tokens through residual addition. The resulting multimodal representations are jointly optimized for four objectives: autoregressive action prediction, action-conditioned forward dynamics, inverse dynamics, and action-free future generation.

3.2 Input Encoding

Text and Visual Tokenization. Each training sample is represented as a chat-style sequence containing the instruction, one or more `<image>` placeholders, and the target action response. The tokenizer converts the instruction and action text into language tokens, while the image placeholders reserve positions for visual embeddings in the final input sequence. During supervised training, labels on system and user tokens are masked with `-100`, so the language-model loss is applied only to assistant-side action tokens.

Visual Encoder. For every RGB observation frame, the VLM image processor produces pixel-level inputs and the corresponding spatial grid metadata. These inputs are fed into the VLM vision encoder, yielding visual embeddings E_t^q and deep visual features used by the language model. When an observation history contains multiple images, all frames are processed and their visual tokens are placed at the corresponding `<image>` positions in the multimodal sequence.

Spatial Encoder. In parallel, the same observation frames are loaded with the spatial-encoder preprocessing pipeline and passed through a frozen spatial encoder, implemented with VGGT Wang et al. (2025a). The spatial encoder is never updated during training. For each frame, the encoder consumes the current image along with its cached history and outputs spatial tokens from an intermediate transformer layer. We use these tokens as spatial-aware features $G_t \in \mathbb{R}^{N_g \times 2048}$, where N_g is the number of spatial patches. The implementation groups neighboring patches according to the VLM spatial merge size, applies an RMS normalization, and maps the grouped spatial features to the VLM hidden size with a two-layer MLP:

$$E_t^g = \mathrm{Merger}_\theta(G_t). \tag{1}$$

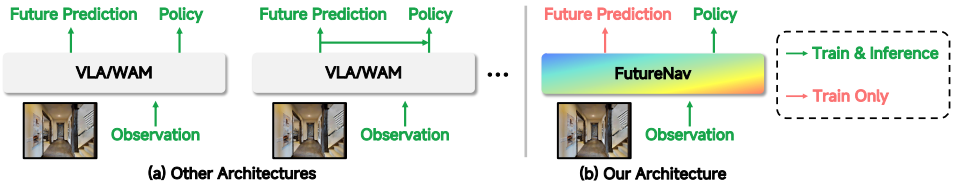

Figure 3: Inference comparison between ***FutureNav*** and existing world-model-based navigation methods. Prior methods involve future-state prediction during inference, introducing additional computation. In contrast, ***FutureNav*** jointly learns policy, inverse dynamics, forward dynamics, and future generation during training, while retaining only the policy branch for efficient inference.

If the number of projected spatial tokens differs from the VLM visual token count, we linearly interpolate E_t^g along the token dimension for alignment. The final visual embedding inserted into the language-model input is

$$E_t^v = E_t^q + \alpha E_t^g, \tag{2}$$

where α is a scalar residual weight. This fusion keeps the pretrained VLM visual representation intact while injecting explicit spatial-aware features into the same token stream consumed by the language model.

3.3 World-Action Modeling

Action Policy. The main navigation policy is the VLM language model conditioned on the spatial-enhanced multimodal sequence. Let $H_t = \{h_1, \ldots, h_L\}$ denote the hidden states output by the language model. A tied language-model head projects each hidden state to vocabulary logits, and the action is decoded autoregressively as an assistant response. In our implementation, each low-level navigation action is represented as text, including `STOP`, `MOVE_FORWARD`, `TURN_LEFT`, and `TURN_RIGHT`. The action loss is the standard next-token cross entropy over the unmasked assistant action tokens:

$$\mathcal{L}_{\text{policy}} = -\sum_{\ell \in \mathcal{Y}} \log p_\theta(y_\ell \mid y_{<\ell}, x, O_t), \tag{3}$$

where $\mathcal{Y}$ denotes unmasked assistant action tokens. This objective teaches the model to map instructions and observations to executable navigation actions.

Forward Dynamics. The forward dynamics head models action-conditioned state transition. During training, the action tokens are available under teacher forcing. We take the hidden state at the last image-token position as the current observation state h_t^{obs} and the hidden state at the last unmasked ground-truth action-token position as the action state h_t^{act}. The two states are concatenated and passed through a lightweight MLP:

$$\hat{z}_{t+1} = f_\theta\left([h_t^{\text{obs}}; h_t^{\text{act}}]\right), \quad \hat{z}_{t+1} \in \mathbb{R}^{2048}. \tag{4}$$

The target z_{t+1} is the pooled spatial-encoder feature of the next observation, represented as a latent vector. The forward dynamics loss is

$$\mathcal{L}_{\text{forward}} = \|\hat{z}_{t+1} - z_{t+1}\|_2^2. \tag{5}$$

This module explicitly asks the model to predict the next spatial state conditioned on the current observation and the executed action.

Inverse Dynamics. The inverse dynamics head encourages visual representations to encode action-relevant changes between adjacent observations. We first split the language-model hidden states at image-token positions into per-frame groups using the image grid metadata. The previous and current observation states are obtained by mean-pooling the last two image-token groups:

$$\bar{h}_{t-1}^v = \text{MeanPool}(H_{t-1}^v), \quad \bar{h}_t^v = \text{MeanPool}(H_t^v). \tag{6}$$

We also use a context hidden state h_t^{ctx}, taken from the token immediately before assistant action response, and an explicit visual-change feature $\Delta h_t^v = \bar{h}_t^v - \bar{h}_{t-1}^v$. The inverse head classifies the action that transforms the previous observation into current one:

$$\hat{a}_t = g_\theta\left([h_t^{\text{ctx}}; \bar{h}_{t-1}^v; \bar{h}_t^v; \Delta h_t^v]\right), \quad \hat{a}_t \in \mathbb{R}^4. \tag{7}$$

The classifier predicts one of four discrete actions in $\mathcal{A}$ and is supervised by previous-action label:

$$\mathcal{L}_{\text{inverse}} = \text{CE}(\hat{a}_t, a_t). \tag{8}$$

Unlike the forward head, which predicts the future spatial state from an action, inverse head recovers action from observed state change. The two objectives therefore provide complementary constraints on action-state consistency.

Future Generation. The generation head predicts future spatial state without using the action token. Unlike the forward head, it does not condition on the teacher-forced action representation. Instead, it combines the context hidden state h_t^{ctx} with the mean-pooled current image-token hidden state $\bar{h}_t^v$ and predicts the next spatial feature:

$$\hat{z}_{t+1}^{\text{gen}} = r_\theta\left([h_t^{\text{ctx}}; \bar{h}_t^v]\right), \quad \hat{z}_{t+1}^{\text{gen}} \in \mathbb{R}^{2048}. \tag{9}$$

It uses the same pooled next-observation spatial feature z_{t+1} as the prediction target and is trained with an MSE loss:

$$\mathcal{L}_{\text{gen}} = \left\|\hat{z}_{t+1}^{\text{gen}} - z_{t+1}\right\|_2^2. \tag{10}$$

Because this head does not receive the action token, it complements the forward dynamics head by learning an action-agnostic prior over short-horizon visual evolution and scene continuity.

3.4 Training Objective

The final training objective combines action generation with the three world-modeling losses:

$$\mathcal{L} = \mathcal{L}_{\text{policy}} + \lambda_f \mathcal{L}_{\text{forward}} + \lambda_i \mathcal{L}_{\text{inverse}} + \lambda_g \mathcal{L}_{\text{gen}}. \tag{11}$$

Only VLM language components, the multimodal merger, and the world-action modeling heads are optimized; the visual and spatial encoder remains frozen. As illustrated in Figure 3, the auxiliary heads shape the shared representation during training but are not required for evaluation. In our main setting, inference uses only the action policy branch for direct action decoding, so world modeling introduces no additional inference cost.

4 Experiments

4.1 Experimental Setup

Dataset and Evaluation Metrics. We evaluate ***FutureNav*** on two standard continuous VLN benchmarks, R2R-CE Krantz et al. (2020) and RxR-CE Ku et al. (2020). Both datasets instantiate instruction-following navigation in the Habitat simulator Savva et al. (2019) with high-fidelity Matterport3D indoor scans Chang et al. (2017). Following prior continuous VLN work Cheng et al. (2024); Wei et al. (2025b); Zeng et al. (2026), we report results on the validation-unseen splits, which contain 1,839 trajectories for R2R-CE and 3,669 trajectories for RxR-CE. We report the standard VLN-CE metrics, including Navigation Error (NE), Oracle Success (OS), Success Rate (SR), and Success weighted by Path Length (SPL), and additionally report normalized Dynamic Time Warping (nDTW) on RxR-CE. We treat SR and SPL as the primary metrics.

Implementation Details. ***FutureNav*** instantiates the VLM backbone with Qwen3-VL-4B and Qwen3-VL-8B Bai et al. (2025), together with a frozen spatial encoder for spatial-aware features and future-state targets. During navigation fine-tuning, we optimize the language model, multimodal merger, and world-action heads, while keeping the VLM vision tower and spatial encoder frozen. The residual spatial fusion weight is set to $\alpha = 0.2$. In the 0K-external-data setting (0K), models are trained only with expert trajectories from R2R-CE and RxR-CE, which provide approximately 2.43M training samples, including 631K from R2R-CE and 1.80M from RxR-CE. In the full-data setting, we further add external navigation data from ScaleVLN Wang et al. (2023c) and DAgger rollouts Ross et al. (2011). Unless otherwise specified, the policy loss weight is set to 1.0, and the auxiliary world-action loss weights are set to $\lambda_f = \lambda_i = \lambda_g = 0.1$.

Real-world Deployment. For real-world experiments, we deploy ***FutureNav*** on a remote H20 GPU server connected to a Go2 robot equipped with a D435i camera. At each step, the robot streams egocentric RGB observations to the server, which predicts and returns a low-level action conditioned on the instruction and RGB history. We evaluate the model zero-shot, without real-world fine-tuning.

Table 1: Comparison with prior VLN-CE methods on the R2R-CE and RxR-CE val-unseen splits. ***FutureNav*** uses only a single RGB sensor. StreamVLN* uses EnvDrop as external data. NaVILA* excludes human-following data. † denotes models trained only with VLN-CE expert trajectories and without any additional data. $\Delta(\%)$ rows report relative gains over JanusVLN; NE is computed in the reverse direction since lower is better.

Method	VLM Params	Observation				R2R Val-Unseen				RxR Val-Unseen				External Training Data
		Pano.	Odo.	Depth	S.RGB	NE↓	OS↑	SR↑	SPL↑	NE↓	SR↑	SPL↑	nDTW↑	
HPN+DN[ICCV'21] Krantz et al. (2021)	-	✓	✓	✓		6.31	40.0	36.0	34.0	-	-	-	-	-
CMA[CVPR'22] Hong et al. (2022)	-	✓	✓	✓		6.20	52.0	41.0	36.0	8.76	26.5	22.1	47.0	-
Sim2Sim[ECCV'22] Krantz & Lee (2022)	-	✓	✓	✓		6.07	52.0	43.0	36.0	-	-	-	-	-
VLN⟳BERT[CVPR'22] Hong et al. (2022)	-	✓	✓	✓		5.74	53.0	44.0	39.0	8.98	27.0	22.6	46.7	-
Ego^2-Map[ICCV'23] Hong et al. (2023)	-	✓	✓	✓		5.54	56.0	47.0	41.0	-	-	-	-	-
DreamWalker[ICCV'23] Wang et al. (2023a)	-	✓	✓	✓		5.53	59.0	49.0	44.0	-	-	-	-	-
GridMM[ICCV'23] Wang et al. (2023b)	-	✓	✓	✓		5.11	61.0	49.0	41.0	-	-	-	-	-
Reborn[ICCV'23] An et al. (2022)	-	✓	✓	✓		5.40	57.0	50.0	46.0	5.98	48.6	42.0	63.3	-
InstructNav[CoRL'24] Long et al. (2024)	-	✓	✓	✓		6.89	-	31.0	24.0	-	-	-	-	-
COSMO[ICCV'25] Zhang et al. (2025h)	-	✓				-	56.0	47.0	40.0	-	-	-	-	-
AO-Planner[AAAI'25] Chen et al. (2025)	-	✓		✓		5.55	59.0	47.0	33.0	7.06	43.3	30.5	50.1	-
LAW[EMNLP'21] Raychaudhuri et al. (2021)	-		✓	✓	✓	6.83	44.0	35.0	31.0	10.90	8.0	8.0	38.0	-
MapNav[ACL'25] Zhang et al. (2025b)	-		✓	✓	✓	4.93	53.0	39.7	37.2	-	-	-	-	-
g3D-LF[CVPR'25] Wang & Lee (2025)	-		✓	✓	✓	5.70	59.5	47.2	34.6	-	-	-	-	-
Seq2Seq[ECCV'20] Krantz et al. (2020)	-			✓	✓	7.77	37.0	25.0	22.0	12.10	13.9	11.9	30.8	-
NaVid-4D[ICRA'25] Liu et al. (2025b)	-			✓	✓	5.99	55.7	43.8	37.1	-	-	-	-	-
NavMorph[ICCV'25] Yao et al. (2025)	-			✓	✓	5.75	56.9	47.9	33.2	8.85	30.8	22.8	44.2	-
NaVid[RSS'24] Zhang et al. (2024b)	7B				✓	5.47	49.1	37.4	35.9	-	-	-	-	953K
Sim2Real[CoRL'24] Wang et al. (2024)	-				✓	5.95	55.8	44.9	30.4	8.79	36.7	25.5	18.1	0K
Uni-NaVid[RSS'25] Zhang et al. (2024a)	7B				✓	5.58	53.3	47.0	42.7	6.24	48.7	40.9	-	3577K
InternVLA-N1[arXiv'25] Wei et al. (2025a)	8B				✓	4.89	60.6	55.4	52.1	6.41	49.5	41.8	62.6	51200K
StreamVLN*[ICRA'26] Wei et al. (2025b)	7B				✓	6.05	53.8	45.5	41.6	-	-	-	-	10033K
NaVILA*[RSS'25] Cheng et al. (2024)	7B				✓	5.37	57.6	49.7	45.5	-	-	-	-	12574K
NaVILA[RSS'25] Cheng et al. (2024)	7B				✓	5.22	62.5	54.0	49.0	6.77	49.3	44.0	58.8	13132K
JanusVLN†[ICLR'26] Zeng et al. (2026)	7B				✓	5.17	58.0	52.8	49.2	6.46	51.4	44.3	59.1	0K
***FutureNav*-4B† (Ours)**	**4B**				✓	**5.13**	**61.8**	**55.1**	**50.1**	**5.62**	**54.5**	**46.0**	**61.8**	**0K**
***FutureNav*-8B† (Ours)**	**8B**				✓	**5.15**	**61.6**	**55.5**	**51.4**	**5.93**	**53.3**	**45.4**	**59.8**	**0K**
StreamVLN[ICRA'26] Wei et al. (2025b)	7B				✓	4.98	64.2	56.9	51.9	6.22	52.9	46.0	61.9	∼26330K
NavFoM[ICLR'26] Zhang et al. (2026a)	7B				✓	5.01	64.9	56.2	51.2	5.51	57.4	49.4	60.2	∼10480K
JanusVLN[ICLR'26] Zeng et al. (2026)	7B				✓	4.78	65.2	60.5	56.8	6.06	56.2	47.5	62.1	10692K
***FutureNav*-4B (Ours)**	**4B**				✓	**4.24**	**70.2**	**65.4**	**61.3**	**4.60**	**60.9**	**52.8**	**68.0**	**10457K**
Δ(%) vs. JanusVLN						*+11.3*	*+7.7*	*+8.1*	*+7.9*	*+24.1*	*+8.4*	*+11.2*	*+9.5*	
***FutureNav*-8B (Ours)**	**8B**				✓	**4.29**	**69.2**	**64.3**	**60.3**	**4.26**	**63.9**	**54.8**	**69.1**	**10435K**
Δ(%) vs. JanusVLN						*+10.3*	*+6.1*	*+6.3*	*+6.2*	*+29.7*	*+13.7*	*+15.4*	*+11.3*	

4.2 Main Results

Table 1 compares ***FutureNav*** with prior VLN-CE methods and recent single-RGB VLM navigation models. When trained only on VLN-CE expert trajectories without external data (†), ***FutureNav***-4B improves over JanusVLN by 4.4% SR on R2R-CE and 6.0% SR on RxR-CE, while the 8B model reaches 55.5 SR and 51.4 SPL on R2R-CE. Despite using no external data, these variants outperform many recent methods, including Uni-NaVid, StreamVLN*, NaVILA*, and NaVILA, on R2R-CE SR/SPL and remain competitive on RxR-CE, showing that the gains do not simply arise from additional training data.

With the full training recipe, ***FutureNav*** uses an external-data budget comparable to JanusVLN and NavFoM, and smaller than StreamVLN and InternVLA-N1. Under this setting, both 4B and 8B checkpoints surpass all prior single-RGB SOTA methods. On R2R-CE, ***FutureNav***-4B achieves the best overall results, improving over JanusVLN by *11.3%* in NE, *8.1%* in SR, and *7.9%* in SPL, despite using slightly less external data (10.457M vs. 10.692M). On RxR-CE, ***FutureNav***-8B gives the strongest performance, reducing NE by *29.7%* over JanusVLN and improving SR/SPL/nDTW by *13.7%*/*15.4%*/*11.3%*. Compared with the strongest previous RxR single-RGB results from NavFoM and InternVLA-N1, it also reduces NE by *22.7%* and improves SR, SPL, and nDTW by *11.3%*, *10.9%*, and *10.4%*, respectively. These results indicate that the unified world-action modeling design improves both data efficiency and final navigation accuracy.

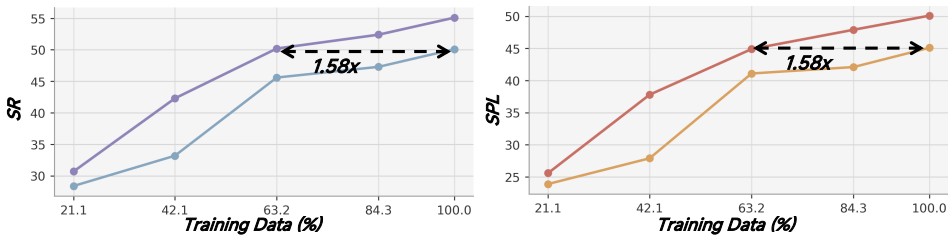

Figure 4: Training efficiency on R2R-CE val-unseen. Under matched training progress, ***FutureNav*** improves SR and SPL faster than the policy-only baseline and achieves higher final performance.

Table 2: Component ablation on the R2R-CE val-unseen split under the 0K setting.

Policy	Inv.	Fwd.	Gen.	NE↓	OS↑	SR↑	SPL↑
✓				6.00	56.5	50.1	45.1
✓	✓			5.82	57.4	50.8	45.8
✓		✓		5.31	59.8	53.0	48.3
✓			✓	5.35	58.3	51.6	46.9
✓	✓	✓	✓	**5.13**	**61.8**	**55.1**	**50.1**

Table 3: Ablation of latent representation on R2R-CE val-unseen split without external data.

Latent Representation	NE↓	OS↑	SR↑	SPL↑
No latent modeling	6.00	56.5	50.1	45.1
VAE latent	5.43	57.8	51.0	46.4
VLM visual latent	5.39	58.1	51.4	46.8
DINO semantic latent	5.34	58.6	51.8	47.1
Spatial latent (Ours)	**5.13**	**61.8**	**55.1**	**50.1**

4.3 Ablation Studies

Effect of World-Action Objectives. Table 2 studies the contribution of each world-action objective under the 0K setting, where all variants are trained only with VLN-CE expert trajectories. The policy-only baseline reaches 50.1 SR and 45.1 SPL, showing that direct action supervision alone already provides a competitive navigation policy. Adding a single world-action objective consistently improves performance. Forward dynamics gives the largest individual gain, increasing SR/SPL to 53.0/48.3, while future generation further improves the baseline to 51.6/46.9. Inverse dynamics brings a smaller but still positive improvement, raising SR/SPL to 50.8/45.8. Combining all three objectives gives the best result, improving over the policy-only baseline by 5.0 SR and 5.0 SPL and reducing NE from 6.00 to 5.13. These results show that the objectives complement action imitation.

Effect of Spatial Latent Representation. Table 3 compares different latent representations for world modeling under the 0K setting. Using VAE latents Kingma & Welling (2014) gives 51.0 SR and 46.4 SPL, while VLM visual latents slightly improve the result to 51.4 SR and 46.8 SPL. DINO semantic latents Caron et al. (2021) perform better, reaching 51.8 SR and 47.1 SPL, suggesting that semantic visual representations provide more useful supervision than generic reconstruction latents or backbone visual features in this setting. Our proposed spatial latent representation performs best, reaching 55.1 SR and 50.1 SPL with the lowest NE of 5.13, demonstrating the benefit of explicitly modeling spatial-aware latent states for future world modeling.

Training Efficiency. Figure 4 compares ***FutureNav*** with a policy-only baseline under the same training schedule. At 42% of training, ***FutureNav*** achieves 42.3 SR and 37.8 SPL, versus 33.2 and 27.9 for the baseline, and reaches the baseline's final SR after only 63% of training. These results show that world-action modeling improves both final performance and training efficiency.

4.4 Qualitative Analysis

Simulated Qualitative Analysis. Figure 5 compares ***FutureNav*** with JanusVLN Zeng et al. (2026) under matched instructions and observations in simulated VLN-CE environments. JanusVLN produces plausible local actions but is more likely to lose spatial progress around landmark-conditioned turns and stopping points. In contrast, ***FutureNav*** maintains a more consistent trajectory, aligns its actions with visual landmarks, and stops closer to the instructed target. This suggests that unified world-action modeling helps preserve spatial context across multiple decision steps rather than merely improving individual action predictions.

Real-world Qualitative Analysis. Figure 6 presents zero-shot real-world navigation results of ***FutureNav*** without additional real-world fine-tuning. The model is evaluated in indoor and outdoor scenes to assess transfer across appearance, lighting, and layout shifts. The examples show that ***FutureNav*** grounds instructions to visible landmarks, maintains the intended route over multiple

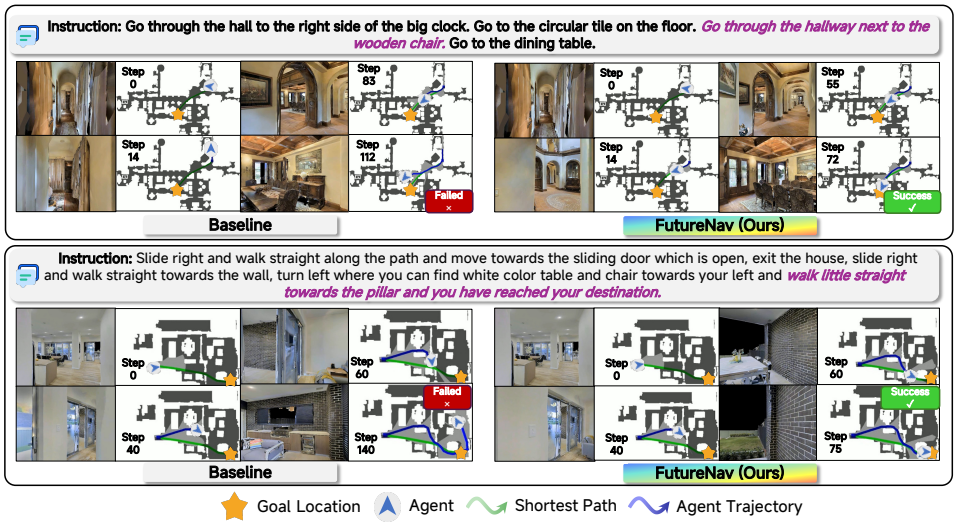

Figure 5: Simulated qualitative comparison on VLN-CE. We compare ***FutureNav*** with JanusVLN Zeng et al. (2026) under matched instructions and observations. JanusVLN tends to make locally plausible but less stable decisions, while ***FutureNav*** better preserves spatial progress, follows landmark-conditioned turns, and stops near the intended target.



Figure 6: Zero-shot real-world qualitative results of ***FutureNav***. Without real-world navigation fine-tuning, the model follows natural language instructions in real indoor and outdoor scenes, grounds navigation decisions to visible landmarks, and stops at the specified destination.

actions, and stops near the specified destination. These results indicate that the learned world-action representations provide useful spatial and temporal structure for zero-shot real-world generalization.

5 Conclusion

We presented ***FutureNav***, a VLM-based unified world-action modeling framework for continuous vision-and-language navigation. Beyond direct action imitation, ***FutureNav*** jointly learns action prediction, inverse dynamics, forward dynamics, and future spatial-state generation within a multimodal action-decoding framework. This design strengthens spatial and transition understanding during training while preserving efficient policy-only inference. Experiments show that ***FutureNav*** achieves state-of-the-art performance with a 4B-scale backbone and outperforms larger VLN foundation models, while ablations confirm the complementary benefits of world-action objectives and spatial latent targets. Our work advances unified world-action modeling for VLN.

AI USE STATEMENT

We did not use generative AI tools to assist with the research or the preparation of this manuscript. The authors take full responsibility for the content, claims, and results presented in this paper.

REFERENCES

Dong An, Zun Wang, Yangguang Li, Yi Wang, Yicong Hong, Yan Huang, Liang Wang, and Jing Shao. 1st place solutions for rxr-habitat vision-and-language navigation competition (cvpr 2022). *arXiv preprint arXiv:2206.11610*, 2022.

Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton Van Den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3674–3683, 2018.

BAAI RoboBrain Team, Mingyu Cao, Huajie Tan, Yuheng Ji, Xiansheng Chen, Minglan Lin, Zhiyu Li, Zhou Cao, Pengwei Wang, Enshen Zhou, et al. Robobrain 2.0 technical report. *arXiv preprint arXiv:2507.02029*, 2025. URL https://arxiv.org/abs/2507.02029.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report, 2025. URL https://arxiv.org/abs/2511.21631.

Amir Bar, Gaoyue Zhou, Danny Tran, Trevor Darrell, and Yann LeCun. Navigation world models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15791–15801, 2025. URL https://arxiv.org/abs/2412.03572.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9650–9660, 2021.

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. *arXiv preprint arXiv:1709.06158*, 2017.

Jiaqi Chen, Bingqian Lin, Xinmin Liu, Lin Ma, Xiaodan Liang, and Kwan-Yee K Wong. Affordances-oriented planning using foundation models for continuous vision-language navigation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 23568–23576, 2025.

An-Chieh Cheng, Yandong Ji, Zhaojing Yang, Zaitian Gongye, Xueyan Zou, Jan Kautz, Erdem Bıyık, Hongxu Yin, Sifei Liu, and Xiaolong Wang. Navila: Legged robot vision-language-action model for navigation. *arXiv preprint arXiv:2412.04453*, 2024.

Haoran Cheng, Enyu Xiao, Yujie Wang, Lingfeng Zhang, Kaidi Xu, Meng Sun, Xiaoshuai Hao, Jinjin Gu, and Renjing Xu. Exploring typographic visual prompts injection threats in cross-modality generation models. In *International Joint Conference on Artificial Intelligence Workshop on Deepfake Detection, Localization and Interpretability*, 2025.

Haoxiang Fu, Lingfeng Zhang, Haoran Li, Rui Hu, Zihan Li, Guoqing Liu, Zeyu Tan, Long Chen, Hang Ye, and Xiaoshuai Hao. Sef-map: Subspace-decomposed expert fusion for robust multimodal hd map prediction. In *IEEE International Conference on Robotics and Automation*, 2026.

Zeying Gong, Rui Li, Tianyu Hu, Ruofei Qiu, Linghe Kong, Lingfeng Zhang, Guoqing Zhao, Yu Ding, and Junwei Liang. Stairway to success: An online floor-aware zero-shot object-goal navigation framework via llm-driven coarse-to-fine exploration. *IEEE Robotics and Automation Letters*, 2026.

Xiaoshuai Hao, Yunfeng Diao, Mengchuan Wei, Yifan Yang, Peng Hao, Rong Yin, Hui Zhang, Weiming Li, Shu Zhao, and Yu Liu. Mapfusion: A novel bev feature fusion network for multi-modal map construction. *Information Fusion*, 119:103018, 2025a.

Xiaoshuai Hao, Yingbo Tang, Lingfeng Zhang, Yuchen Ma, Yuhang Diao, Zihan Jia, Wenbo Ding, Hang Ye, and Long Chen. Roboafford++: A generative ai-enhanced dataset for multimodal affordance learning in robotic manipulation and navigation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems Workshop on RoDGE*, 2025b.

Xiaoshuai Hao, Lei Zhou, Zhijian Huang, Zhiwen Hou, Yingbo Tang, Lingfeng Zhang, Guang Li, Zheng Lu, Shuhuai Ren, et al. Mimo-embodied: X-embodied foundation model technical report. *arXiv preprint arXiv:2511.16518*, 2025c. URL https://arxiv.org/abs/2511.16518.

Xiaoshuai Hao, Haoran Lyu, Lingfeng Zhang, Ruidong Liu, Di Wu, Jing Zhang, and Long Chen. H2r-bm: Can leveraging human videos enhance performance and generalizability in robotic bimanual manipulation? *Pattern Recognition*, pp. 113637, 2026a.

Xiaoshuai Hao, Yingbo Tang, Lingfeng Zhang, Long Chen, Wei Zhou, Jungong Han, Wenbo Ding, and Xiao-Ping Zhang. Embodied spatial affordance: spatial-aware affordance learning for embodied navigation and manipulation. *IEEE Transactions on Image Processing*, 2026b.

Xiaoshuai Hao, Zhihui Zhang, Yingbo Tang, Lingfeng Zhang, Peng Hao, Yunfeng Diao, Guangyin Jin, and Yu Liu. Synergistic prompting for complementarity and consistency in incomplete multi-view clustering. *IEEE Transactions on Image Processing*, 2026c.

Yicong Hong, Zun Wang, Qi Wu, and Stephen Gould. Bridging the gap between learning in discrete and continuous environments for vision-and-language navigation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 15439–15449, 2022.

Yicong Hong, Yang Zhou, Ruiyi Zhang, Franck Dernoncourt, Trung Bui, Stephen Gould, and Hao Tan. Learning navigational visual representations with semantic map supervision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3055–3067, 2023.

Junjun Hu, Jintao Chen, Haochen Bai, Minghua Luo, Shichao Xie, Ziyi Chen, Fei Liu, Zedong Chu, Xinda Xue, Botao Ren, et al. Astranav-world: World model for foresight control and consistency. *arXiv preprint arXiv:2512.21714*, 2025.

Yanjia Huang, Xianshun Jiang, Xiangbo Gao, Mingyang Wu, and Zhengzhong Tu. Vistav2: World imagination for indoor vision-and-language navigation. *arXiv preprint arXiv:2512.00041*, 2025. URL https://arxiv.org/abs/2512.00041.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014.

Jing Yu Koh, Honglak Lee, Yinfei Yang, Jason Baldridge, and Peter Anderson. Pathdreamer: A world model for indoor navigation. *arXiv preprint arXiv:2105.08756*, 2021. URL https://arxiv.org/abs/2105.08756.

Linghe Kong, Sicheng Xie, Zeying Gong, Yuxiang Li, Min Chu, An Liang, Yuhang Dong, Tianyu Hu, Ruofei Qiu, Rui Li, et al. The robosense challenge: Sense anything, navigate anywhere, adapt across platforms. *arXiv preprint*, 2026.

Jacob Krantz and Stefan Lee. Sim-2-sim transfer for vision-and-language navigation in continuous environments. In *European conference on computer vision*, pp. 588–603. Springer, 2022.

Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *European Conference on Computer Vision*, pp. 104–120. Springer, 2020.

Jacob Krantz, Aaron Gokaslan, Dhruv Batra, Stefan Lee, and Oleksandr Maksymets. Waypoint models for instruction-guided navigation in continuous environments. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15162–15171, 2021.

Alexander Ku, Peter Anderson, Roma Patel, Eugene Ie, and Jason Baldridge. Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. *arXiv preprint arXiv:2010.07954*, 2020.

Dong Li, Shuang Ma, Hang Hua, Wei Li, Jian Wang, Chengwei Zhou, Feng Guan, Xin Li, Zhi Yu, Yao Lu, et al. Vquala 2025 challenge on engagement prediction for short videos: Methods and results. In *IEEE/CVF International Conference on Computer Vision Workshop*, 2025.

Haoran Li, Lingfeng Zhang, Ziheng Yang, Li Li, Rui Yin, Xiaoshuai Hao, and Wenbo Ding. Weather-conditioned branch routing for robust lidar-radar 3d object detection. *arXiv preprint arXiv:2604.05405*, 2026a. URL https://arxiv.org/abs/2604.05405.

Pengna Li, Kangyi Wu, Shaoqing Xu, Fang Li, Hanbing Li, Lin Zhao, Kailin Lyu, Long Chen, Zhi-Xin Yang, and Nanning Zheng. Spaact: Spatially-activated transition learning with curriculum adaptation for vision-language navigation. *arXiv preprint arXiv:2604.27620*, 2026b.

Bingqian Lin, Yunshuang Nie, Ziming Wei, Jiaqi Chen, Shikui Ma, Jianhua Han, Hang Xu, Xiaojun Chang, and Xiaodan Liang. Navcot: Boosting llm-based vision-and-language navigation via learning disentangled reasoning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025. URL https://arxiv.org/abs/2403.07376.

Fei Liu, Shichao Xie, Minghua Luo, Zedong Chu, Junjun Hu, Xiaolong Wu, and Mu Xu. Navforesee: A unified vision-language world model for hierarchical planning and dual-horizon navigation prediction. *arXiv preprint arXiv:2512.01550*, 2025a.

Haoran Liu, Weikang Wan, Xiqian Yu, Minghan Li, Jiazhao Zhang, Bo Zhao, Zhibo Chen, Zhongyuan Wang, Zhizheng Zhang, and He Wang. Na vid-4d: Unleashing spatial intelligence in egocentric rgb-d videos for vision-and-language navigation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 10607–10615. IEEE, 2025b.

Jiahao Liu, Haoran Chi, Lingfeng Zhang, Yuxuan Xie, Yanan Wang, Long Chen, Hang Ye, Xiaoshuai Hao, and Wenbo Ding. Thinking in text and images: Interleaved vision-language reasoning traces for long-horizon robot manipulation. *arXiv preprint arXiv:2605.00438*, 2026a. URL https://arxiv.org/abs/2605.00438.

Peng Liu, Qiang Zhang, Di Peng, Lingfeng Zhang, Yiran Qin, Huan Zhou, Jiajun Ma, Renjing Xu, and Yandong Ji. Toponav: Topological graphs as a key enabler for advanced object navigation. In *IEEE International Conference on Robotics and Automation*, 2026b.

Yichen Liu, Peng Sun, Shuo Li, Yuxuan Xie, Lingfeng Zhang, Xingyu Chao, Siyuan Dong, Fei Chen, Xiaoping Zhang, et al. Oa-wam: Object-addressable world action model for robust robot manipulation. *arXiv preprint arXiv:2605.06481*, 2026c. URL https://arxiv.org/abs/2605.06481.

Yuxing Long, Wenzhe Cai, Hongcheng Wang, Guanqi Zhan, and Hao Dong. Instructnav: Zero-shot system for generic instruction navigation in unexplored environment. *arXiv preprint arXiv:2406.04882*, 2024.

Jian Lu, Jian Guan, Zhaorui Huang, Jiacheng Li, Guoqiang Li, Linghe Kong, Yuxiang Li, Haoran Wang, Shiyu Xu, Yifan Luo, Fei Li, et al. Onevl: One-step latent reasoning and planning with vision-language explanation. *arXiv preprint arXiv:2604.18486*, 2026. URL https://arxiv.org/abs/2604.18486.

Yuchen Ma, Fei Luo, Lingfeng Zhang, Chen Zhao, Ming Wang, Yifan Wu, Ziyu Qian, Yao Lu, Long Chen, et al. Reasoning emerges from constrained inference manifolds in large language models. *arXiv preprint arXiv:2605.08142*, 2026. URL https://arxiv.org/abs/2605.08142.

Sonia Raychaudhuri, Saim Wani, Shivansh Patel, Unnat Jain, and Angel Chang. Language-aligned waypoint (law) supervision for vision-and-language navigation in continuous environments. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pp. 4018–4028, 2021.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pp. 627–635. JMLR Workshop and Conference Proceedings, 2011.

Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied ai research. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9339–9347, 2019.

Yingbo Tang, Lingfeng Zhang, Shilong Zhang, Yifan Zhao, and Xiaoshuai Hao. Roboafford: A dataset and benchmark for enhancing object and spatial affordance learning in robot manipulation. In *Proceedings of the ACM International Conference on Multimedia*, 2025.

Hanqing Wang, Wei Liang, Luc Van Gool, and Wenguan Wang. Dreamwalker: Mental planning for continuous vision-language navigation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10873–10883, 2023a.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5294–5306, 2025a.

Shuo Wang, Yongcai Wang, Wanting Li, Yucheng Wang, Maiyue Chen, Kaihui Wang, Zhizhong Su, Xudong Cai, Yeying Jin, Deying Li, and Zhaoxin Fan. Monodream: Monocular vision-language navigation with panoramic dreaming. *arXiv preprint arXiv:2508.02549*, 2025b. URL https://arxiv.org/abs/2508.02549.

Zihan Wang and Gim Hee Lee. g3d-lf: Generalizable 3d-language feature fields for embodied tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14191–14202, 2025.

Zihan Wang, Xiangyang Li, Jiahao Yang, Yeqi Liu, and Shuqiang Jiang. Gridmm: Grid memory map for vision-and-language navigation. In *Proceedings of the IEEE/CVF International conference on computer vision*, pp. 15625–15636, 2023b.

Zihan Wang, Xiangyang Li, Jiahao Yang, Yeqi Liu, and Shuqiang Jiang. Sim-to-real transfer via 3d feature fields for vision-and-language navigation. *arXiv preprint arXiv:2406.09798*, 2024.

Zun Wang, Jialu Li, Yicong Hong, Yi Wang, Qi Wu, Mohit Bansal, Stephen Gould, Hao Tan, and Yu Qiao. Scaling data generation in vision-and-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12009–12020, 2023c.

Meng Wei, Chenyang Wan, Jiaqi Peng, Xiqian Yu, Yuqiang Yang, Delin Feng, Wenzhe Cai, Chenming Zhu, Tai Wang, Jiangmiao Pang, et al. Ground slow, move fast: A dual-system foundation model for generalizable vision-and-language navigation. *arXiv preprint arXiv:2512.08186*, 2025a.

Meng Wei, Chenyang Wan, Xiqian Yu, Tai Wang, Yuqiang Yang, Xiaohan Mao, Chenming Zhu, Wenzhe Cai, Hanqing Wang, Yilun Chen, et al. Streamvln: Streaming vision-and-language navigation via slowfast context modeling. *arXiv preprint arXiv:2507.05240*, 2025b.

Yifan Wu, Haoran Lyu, Yingbo Tang, Lingfeng Zhang, Ziheng Zhang, Wenxuan Zhou, and Shibo Hao. Evaluating gpt-4o's embodied intelligence: A comprehensive empirical study. Technical report, Technical Report, 2025.

Enyu Xiao, Lingfeng Zhang, Yingbo Tang, Haoran Cheng, Renjing Xu, Wenbo Ding, Li Zhou, Long Chen, Hang Ye, et al. Learning to navigate socially through proactive risk perception – technical report for iros 2025 robosense challenge social navigation track. Technical report, IROS 2025 RoboSense Challenge, 2025.

Ning Yang, Yan Huang, Kaiwen Peng, Ziheng He, Kai Wang, Cui Miao, Kailin Lyu, Guo Li, Xiaofeng Wang, Zheng Zhu, Jing Liu, and Nianfeng Liu. Wam-nav: Asymmetric latent world-action modeling for unified visual navigation. *arXiv preprint arXiv:2606.04907*, 2026. URL https://arxiv.org/abs/2606.04907.

Xuan Yao, Junyu Gao, and Changsheng Xu. Navmorph: A self-evolving world model for vision-and-language navigation in continuous environments. *arXiv preprint arXiv:2506.23468*, 2025.

Naoki Yokoyama, Sehoon Ha, Dhruv Batra, Jiuguang Wang, and Bernadette Bucher. VLFM: Vision-language frontier maps for zero-shot semantic navigation. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2024a.

Naoki Yokoyama, Ram Ramrakhya, Abhishek Das, Dhruv Batra, and Sehoon Ha. HM3D-OVON: A dataset and benchmark for open-vocabulary object goal navigation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5543–5550, 2024b. URL https://arxiv.org/abs/2409.14296.

Tianyuan Yuan, Zibin Dong, Yicheng Liu, and Hang Zhao. Fast-wam: Do world action models need test-time future imagination? *arXiv preprint arXiv:2603.16666*, 2026.

Shuang Zeng, Dekang Qi, Xinyuan Chang, Feng Xiong, Shichao Xie, Xiaolong Wu, Shiyi Liang, Mu Xu, and Xing Wei. JanusVLN: Decoupling Semantics and Spatiality with Dual Implicit Memory for Vision-Language Navigation. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=RnuB0Nlbd5.

Jiazhao Zhang, Kunyu Wang, Shaoan Wang, Minghan Li, Haoran Liu, Songlin Wei, Zhongyuan Wang, Zhizheng Zhang, and He Wang. Uni-navid: A video-based vision-language-action model for unifying embodied navigation tasks. *arXiv preprint arXiv:2412.06224*, 2024a.

Jiazhao Zhang, Kunyu Wang, Rongtao Xu, Gengze Zhou, Yicong Hong, Xiaomeng Fang, Qi Wu, Zhizheng Zhang, and He Wang. Navid: Video-based vlm plans the next step for vision-and-language navigation. *arXiv preprint arXiv:2402.15852*, 2024b.

Jiazhao Zhang, Anqi Li, Yunpeng Qi, Minghan Li, Jiahang Liu, Shaoan Wang, Haoran Liu, Gengze Zhou, Yuze Wu, Xingxing Li, Yuxin Fan, Wenjun Li, Zhibo Chen, Fei Gao, Qi Wu, Zhizheng Zhang, and He Wang. Embodied Navigation Foundation Model. In *International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=kkBOIsrCXh.

Lingfeng Zhang, Qiang Zhang, Hao Wang, Erjia Xiao, Zixuan Jiang, Honglei Chen, and Renjing Xu. Trihelper: Zero-shot object navigation with dynamic assistance. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2024c. URL https://arxiv.org/abs/2403.15223.

Lingfeng Zhang, Xiaoshuai Hao, Yingbo Tang, Haoxiang Fu, Xinyu Zheng, Pengwei Wang, Zhongyuan Wang, Wenbo Ding, and Shanghang Zhang. Nava^3: Understanding any instruction, navigating anywhere, finding anything. *arXiv preprint arXiv:2508.04598*, 2025a. URL https://arxiv.org/abs/2508.04598.

Lingfeng Zhang, Xiaoshuai Hao, Qinwen Xu, Qiang Zhang, Xinyao Zhang, Pengwei Wang, Jing Zhang, Zhongyuan Wang, Shanghang Zhang, and Renjing Xu. Mapnav: A novel memory representation via annotated semantic maps for vlm-based vision-and-language navigation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13032–13056, 2025b.

Lingfeng Zhang, Hanqing Wang, Enyu Xiao, Xinyao Zhang, Qiang Zhang, Zihan Jiang, and Renjing Xu. Multi-floor zero-shot object navigation policy. In *IEEE International Conference on Robotics and Automation*, 2025c.

Lingfeng Zhang, Enyu Xiao, Yifan Zhang, Haoxiang Fu, Rui Hu, Yuchen Ma, Wenbo Ding, Long Chen, Hang Ye, et al. Team xiaomi ev-ad vla: Caption-guided retrieval system for cross-modal drone navigation – technical report for iros 2025 robosense challenge track 4. Technical report, IROS 2025 RoboSense Challenge, 2025d.

Lingfeng Zhang, Erjia Xiao, Xiaoshuai Hao, Haoxiang Fu, Zeying Gong, Long Chen, Xiaojun Liang, Renjing Xu, Hangjun Ye, and Wenbo Ding. Socialnav-map: Dynamic mapping with human trajectory prediction for zero-shot social navigation. *arXiv preprint arXiv:2511.12232*, 2025e. URL https://arxiv.org/abs/2511.12232.

Lingfeng Zhang, Haoxiang Fu, Xiaoshuai Hao, Shilong Zhang, Qiang Zhang, Ruidong Liu, Long Chen, and Wenbo Ding. What you see is what you reach: Towards spatial navigation with high-level human instructions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026b.

Lingfeng Zhang, Xiaoshuai Hao, Xinyu Bu, Yingbo Tang, Haoran Li, Jian Lu, Xinyu Wei, Jiajun Ma, Yichen Liu, Jing Zhang, et al. Walk with me: Long-horizon social navigation for human-centric outdoor assistance. *arXiv preprint arXiv:2604.26839*, 2026c. URL https://arxiv.org/abs/2604.26839.

Lingfeng Zhang, Xiaoshuai Hao, Yingbo Tang, Lei Zhou, Shuyi Zhang, Jinkun Liu, Hongsheng Li, Chenhao Zhang, Qiang Zhang, Hangjun Ye, Xiaojun Liang, Long Chen, and Wenbo Ding. Onevla: A unified framework for embodied tasks. *arXiv preprint arXiv:2606.01241*, 2026d. URL https://arxiv.org/abs/2606.01241.

Lingfeng Zhang, Yingbo Tang, Xinyu Zheng, Liang Li, Jinglin Xu, and Xiaoshuai Hao. Embodiedplan-1k: A benchmark for complex navigation-manipulation task planning. 2026e.

Lingfeng Zhang, Yifan Zhang, Haoran Li, Haoxiang Fu, Yingbo Tang, Hang Ye, Long Chen, Xiaojun Liang, Xiaoshuai Hao, et al. Is your vlm sky-ready? a comprehensive spatial intelligence benchmark for uav navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026f.

Qiang Zhang, Ziheng Zhang, Wenxuan Cui, Jiaming Sun, Junyi Cao, Yuxuan Guo, Guang Han, Wen Zhao, Jun Wang, et al. Humanoidpano: Hybrid spherical panoramic-lidar cross-modal perception for humanoid robots. *arXiv preprint arXiv:2503.09010*, 2025f. URL https://arxiv.org/abs/2503.09010.

Qiang Zhang, Jiajun Ma, Peng Liu, Shiyu Shi, Zhiqiang Su, Zhongyuan Wang, Jiaming Sun, Wenxuan Cui, Jia Yu, Guang Han, et al. Meshmimic: Geometry-aware humanoid motion learning through 3d scene reconstruction. *arXiv preprint arXiv:2602.15733*, 2026g. URL https://arxiv.org/abs/2602.15733.

Shilong Zhang, Xiaoshuai Hao, Yingbo Tang, Lingfeng Zhang, Pengwei Wang, Zhongyuan Wang, Huan Ma, and Shanghang Zhang. Video-cot: A comprehensive dataset for spatiotemporal understanding of videos based on chain-of-thought. In *Proceedings of the ACM International Conference on Multimedia*, 2025g.

Siqi Zhang, Yanyuan Qiao, Qunbo Wang, Zike Yan, Qi Wu, Zhihua Wei, and Jing Liu. Cosmo: Combination of selective memorization for low-cost vision-and-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5511–5522, 2025h.

Baining Zhao, Jiacheng Xu, Weicheng Feng, Xin Zhang, Zhaolu Wang, Haoyang Wang, Shilong Ji, Ziyou Wang, Jianjie Fang, Zhiheng Zheng, Weichen Zhang, Yu Shang, Wei Wu, Chen Gao, Xinlei Chen, and Yong Li. Worldvln: Autoregressive world action model for aerial vision-language navigation. *arXiv preprint arXiv:2605.15964*, 2026. URL https://arxiv.org/abs/2605.15964.

Ziyu Zhu, Xilin Wang, Yixuan Li, Zhuofan Zhang, Xiaojian Ma, Yixin Chen, Baoxiong Jia, Wei Liang, Qian Yu, Zhidong Deng, Siyuan Huang, and Qing Li. Move to understand a 3D scene: Bridging visual grounding and exploration for efficient and versatile embodied navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8120–8132, 2025.

APPENDIX

This appendix supplements the main paper with implementation details, quantitative real-world evaluation results, and additional experimental analyses of ***FutureNav***. It is organized as follows:

- Sec. A provides **Implementation Details** on 3D feature construction and fusion, world-action modeling and supervision, and training hyperparameters.
- Sec. B reports the protocol and results of **Quantitative Real-World Evaluation**, comparing success rate and navigation error in indoor and outdoor environments.
- Sec. C presents **Additional Experiments** on open-vocabulary object-goal navigation and auxiliary loss weights, together with controlled ablation protocols for spatial representations and training configurations.
- Sec. C.5 describes the methodology for **Visualization Analysis** of navigation behavior and spatial-state prediction.

A IMPLEMENTATION DETAILS

A.1 3D FEATURE CONSTRUCTION AND FUSION

RGB-derived spatial representation. ***FutureNav*** extracts complementary semantic and spatial representations from RGB observations using the Qwen3-VL vision tower and a frozen VGGT encoder, respectively. Here, "3D features" refer to latent features of the geometry-pretrained encoder, rather than measured depth, reconstructed point clouds, or ground-truth camera poses supplied to the navigation policy. This distinction preserves the single-RGB observation setting. The spatial branch operates in evaluation mode with gradient computation disabled, while the projection that connects it to the language model remains trainable.

Preprocessing and temporal context. The spatial-image loader uses bicubic resizing with a target width of 644 pixels and preserves the aspect ratio, rounding the height to a multiple of the 14-pixel patch size. If the resized height exceeds 644 pixels, it is center-cropped. Thus, the spatial input is not necessarily square. The semantic branch independently follows the VLM image processor and the pixel budgets in Table 4. With the first frame as the temporal reference, spatial frames are processed in chronological order, passing the VGGT key-value cache from one frame to the next. During training, this cache is reset for each sample. The current-frame representation therefore incorporates the preceding observations supplied with that sample.

Patch selection and projection. Let $G_t \in \mathbb{R}^{N_g \times 2048}$ denote the spatial patch features at the current step. The spatial encoder processes the observations sequentially and selects the penultimate returned aggregation feature tensor, excluding its leading non-patch tokens. The implementation retains the current frame's spatial features after processing the observed history; these features carry information from preceding frames through the encoder cache. The patch grid is cropped to dimensions divisible by the VLM spatial merge factor m, and each local $m \times m$ group is packed into one vector. Each 2048-dimensional patch feature is RMS-normalized before packing. A two-layer projection then maps the packed vector to the language-model dimension d:

$$e_j^g = W_2\,\mathrm{GELU}(W_1\,\mathrm{Pack}_m(\mathrm{RMSNorm}(G_t))_j + b_1) + b_2, \tag{12}$$

where the projection widths are $2048m^2 \to 4096 \to d$ under the default merger configuration. If the projected token count differs from the semantic visual-token count, one-dimensional linear interpolation aligns the spatial sequence to the latter, with `align_corners=False`. The aligned spatial tokens are fused by residual addition,

$$E_t^v = E_t^q + \alpha\,\mathrm{Align}(E_t^g), \qquad \alpha = 0.2. \tag{13}$$

The fused embeddings replace the image-placeholder embeddings in the multimodal sequence. This procedure transfers spatial context through latent tokens without introducing explicit geometric measurements into the policy input.

A.2 World-Action Inputs, Outputs, and Supervision

Future-state targets. For a training transition (O_t, a_t, I_{t+1}), the next RGB frame supplies a latent supervision target from the frozen spatial encoder and is not part of the policy's observed input. Let G_{t+1} denote its spatial patch features. Consistent with the main text, the target is a pooled spatial representation,

$$z_{t+1} = \mathrm{stopgrad}[\mathrm{AAP}_{2048}(\mathrm{vec}(G_{t+1}))] \in \mathbb{R}^{2048}, \tag{14}$$

where AAP_{2048} is one-dimensional adaptive average pooling after flattening the spatial feature tensor, producing 2048 output elements. This is not a channel-preserving average over patch positions. The target is computed in floating-point precision and detached from the gradient graph. In the online target path, the current spatial features are retained before the next frame is processed. Both forward dynamics and future generation regress the resulting future target, without inserting that future observation into the policy input.

Action policy and hidden-state extraction. The language-model output head produces vocabulary logits and autoregressively predicts the textual navigation action. Its cross-entropy loss is restricted to unmasked assistant tokens; prompt and observation tokens have label -100. We denote by h_t^{obs} the hidden state at the last image-token position, by h_t^{act} the hidden state at the last unmasked response-token position, and by h_t^{ctx} the state immediately preceding the first unmasked response token. The current visual state $\bar{h}_t^v$ is the mean of the final image-token group's language-model hidden states. Each of these vectors has dimension d.

Forward dynamics and future generation. The forward head takes $[h_t^{\mathrm{obs}}; h_t^{\mathrm{act}}] \in \mathbb{R}^{2d}$ and returns a 2048-dimensional prediction $\hat{z}_{t+1}^{\mathrm{fwd}}$. The future-generation head instead takes $[h_t^{\mathrm{ctx}}; \bar{h}_t^v] \in \mathbb{R}^{2d}$ and predicts $\hat{z}_{t+1}^{\mathrm{gen}}$ without directly receiving the teacher-forced action representation. Each head consists of LayerNorm, a linear map from $2d$ to d, GELU, and a linear map from d to 2048. The implementation uses mean squared error averaged over the valid samples and output coordinates. These outputs are latent-state predictions, not decoded RGB frames, depth maps, or action sequences.

Inverse dynamics. Inverse dynamics recovers the action connecting the previous and current observations. The visual-token hidden states are partitioned into frame groups, and the last two groups are mean-pooled to obtain $\bar{h}_{t-1}^v, \bar{h}_t^v \in \mathbb{R}^d$. These are contextualized language-model states, rather than the vision tower's pre-fusion embeddings. Their difference captures the observed transition. Together with the context hidden state, these vectors form

$$u_t^{\mathrm{inv}} = [h_t^{\mathrm{ctx}}; \bar{h}_{t-1}^v; \bar{h}_t^v; \bar{h}_t^v - \bar{h}_{t-1}^v], \tag{15}$$

$$\ell_t^{\mathrm{inv}} = g_\theta(u_t^{\mathrm{inv}}) \in \mathbb{R}^4. \tag{16}$$

The inverse head consists of LayerNorm over $4d$ channels, a linear map from $4d$ to d, GELU, and a final linear map from d to four action logits. The classes correspond to `STOP`, `MOVE_FORWARD`, `TURN_LEFT`, and `TURN_RIGHT`. Cross-entropy supervision uses the previous-action label, namely the action that leads from the previous observation to the current one. Thus, inverse dynamics operates on an observed transition, whereas forward dynamics and future generation predict the next spatial state. The inverse branch uses neither a next-frame target nor a separate next-state projection.

Transition validity and inference. Inverse supervision requires at least two image-state groups, a valid previous-action label, and an available context state. Forward and generation supervision require a valid future-state target; generation additionally requires the current image-state group and context state. Samples that fail the corresponding checks are excluded from that auxiliary loss, and an auxiliary loss is omitted when no valid samples remain. The target-dependent losses and the inverse loss therefore need not use the same subset of a batch. At inference, only autoregressive policy decoding is required: none of the three auxiliary heads or the next-frame target encoder pass is needed. The frozen spatial encoder and its fusion projection remain part of the policy input pathway.

A.3 Training Configuration

Table 4 summarizes the training configuration for ***FutureNav***-4B, including optimization hyperparameters, preprocessing settings, and auxiliary loss weights. The loss coefficients and world-action

Table 4: Training configuration for ***FutureNav***-4B. Auxiliary loss weights follow the main-text setting. Pixel limits specify processor budgets rather than fixed input resolutions.

Hyperparameter	Value
VLM backbone	Qwen3-VL-4B-Instruct
Spatial encoder	VGGT-1B, frozen
Training dataset identifier	`VLN-CE, ScaleVLN`
Optimizer	AdamW (`adamw_torch`)
Base learning rate	2×10^{-5}
Merger learning rate	1×10^{-5}
Vision-tower learning-rate argument	1×10^{-6} (backbone frozen)
Weight decay	0.01
Training epochs	1
Scheduler / warmup ratio	Cosine / 0.03
Per-device batch / accumulation	1 / 4
Effective global batch	$4N_{\mathrm{GPU}}$
Precision / distributed strategy	BF16 / DeepSpeed ZeRO stage 2
Gradient checkpointing	Enabled
Maximum sequence length	163,840 tokens
Image pixel budget (min, max)	$(16 \times 28^2,\ 576 \times 28^2)$
Video frame count (min, max)	$(4, 8)$
Video frame pixel budget (min, max)	$(256 \times 28^2,\ 1664 \times 28^2)$
Base sampling interval	2
Spatial resize target / patch size	644 / 14 pixels
Spatial feature / target dimension	2048 / 2048
Spatial merger hidden width	4096 (default)
Spatial fusion weight α	0.2 (default)
Temporal reference	First frame
$(\lambda_f, \lambda_i, \lambda_g)$	$(0.1, 0.1, 0.1)$
Policy-loss coefficient	1
Data flattening / modality grouping	Disabled / enabled
Data-loader workers	8 per process
Random seed	42
Logging / checkpoint interval	10 / 1000 steps
Maximum retained checkpoints	2

formulation are consistent with those described in the main text. The language model, visual merger, spatial merger, and auxiliary prediction modules are trainable; the vision-tower backbone and VGGT are frozen. AdamW uses a cosine learning-rate schedule with linear warmup. With one sample per device and four gradient accumulation steps, the effective global batch size is $4N_{\mathrm{GPU}}$, where N_{GPU} is the total number of distributed training processes.

B Quantitative Real-World Evaluation

Platform and evaluation objective. We complement the qualitative deployment described in the main text with a quantitative evaluation on the Go2 platform. The robot's D435i camera supplies egocentric RGB observations to an H20 inference server, which returns navigation actions for execution. We evaluate ***FutureNav*** without additional real-world fine-tuning to assess its generalization to physical indoor and outdoor environments.

Tasks and comparison protocol. We select four real-world scenes, comprising two indoor and two outdoor environments, and construct a set of 25 navigation instructions. Each method executes every instruction four times, yielding 100 trials per method. We compare NaVILA (Cheng et al., 2024), InternVLA-N1 (Wei et al., 2025a), JanusVLN (Zeng et al., 2026), a policy-only variant of ***FutureNav***, and the full ***FutureNav*** model on the same instruction set. The policy-only variant is trained with the action prediction objective, while the full model additionally incorporates the three auxiliary world-action objectives.

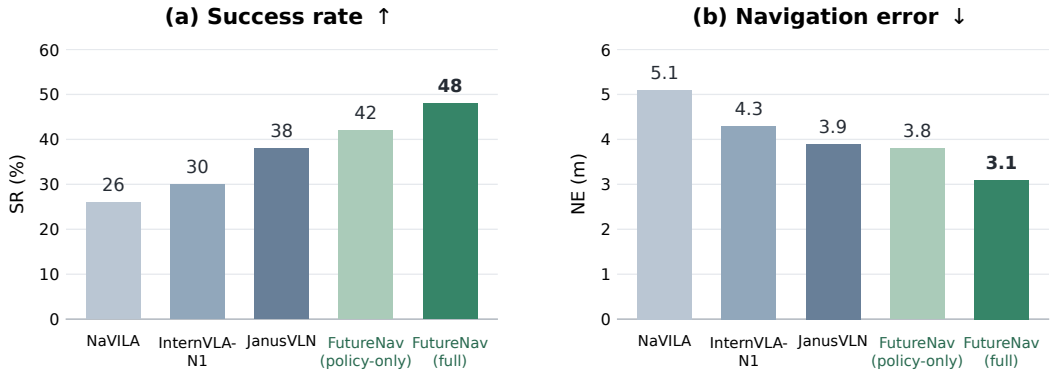

Figure 7: Quantitative real-world navigation results in two indoor and two outdoor scenes. Each method is evaluated on 25 instructions with four repetitions per instruction, totaling 100 trials. The panels report average success rate (SR, higher is better) and navigation error (NE, lower is better), with a 1 m success threshold. We compare three navigation baselines with the policy-only and full variants of ***FutureNav***.

Metrics and reporting. We report two metrics: success rate (SR) and navigation error (NE). A trial is successful if the robot reaches within 1 m of the goal. SR is the percentage of successful trials, and NE is the distance in meters between the robot's final position and the goal. For each metric, we average over the four repetitions of the 25-instruction evaluation, equivalent to aggregating all 100 trials per method. Higher SR and lower NE indicate better navigation performance.

Results. Figure 7 summarizes the quantitative comparison. The full ***FutureNav*** model achieves the highest SR (48%) and the lowest NE (3.1 m) among the evaluated methods. Compared with its policy-only variant, it improves SR by 6 percentage points and reduces NE by 0.7 m. Relative to JanusVLN, it improves SR by 10 percentage points and reduces NE by 0.8 m.

C Additional Experiments

C.1 Open-Vocabulary Object-Goal Navigation

We evaluate ***FutureNav*** on HM3D-OVON (Yokoyama et al., 2024b), which specifies navigation goals through object descriptions rather than route-following instructions. Each object goal is expressed as "Navigate to the [target object] and stop near it." The policy-only and full variants use the same goal prompt and action mapping. Table 5 reports SR and SPL on the validation-unseen split, together with published baseline results from JanusVLN (Zeng et al., 2026). The policy-only model achieves 43.4% SR and 30.2% SPL, while the full ***FutureNav*** model reaches 47.1% SR and 33.3% SPL, improving the two metrics by 3.7 and 3.1 percentage points, respectively. These results demonstrate the benefit of world-action modeling for object-goal navigation.

Table 5: Open-vocabulary object-goal navigation results on HM3D-OVON validation-unseen. Baseline results are taken from Table 6 of JanusVLN (Zeng et al., 2026). Higher SR and SPL indicate better performance. Bold denotes the best result in each column.

Method	SR (%)	SPL (%)
VLFM (Yokoyama et al., 2024a)	35.2	19.6
DAgRL+OD (Yokoyama et al., 2024b)	37.1	19.8
Uni-NaVid (Zhang et al., 2024a)	39.5	19.8
MTU3D (Zhu et al., 2025)	40.8	12.1
JanusVLN (Zeng et al., 2026)	44.9	31.7
Policy-only 4B	43.4	30.2
FutureNav-4B	**47.1**	**33.3**

C.2 Sensitivity to Auxiliary Loss Weights

Beyond the component ablation in the main text, we examine the magnitude and balance of the auxiliary losses while fixing the policy-loss coefficient to one. Table 6 compares five weight configurations using the same 4B backbone, expert-only training data, optimization budget, and R2R-CE validation-unseen episodes. All configurations with auxiliary supervision outperform the policy-only baseline. Among the evaluated settings, $(\lambda_f, \lambda_i, \lambda_g) = (0.1, 0.1, 0.1)$ achieves the lowest NE (5.13 m), highest SR (55.1%), and highest SPL (50.1%), supporting the default weighting used in the main text. Increasing all three coefficients to 0.2 slightly reduces SR and SPL, indicating that stronger auxiliary supervision does not necessarily improve navigation performance.

Table 6: Sensitivity to auxiliary loss weights on R2R-CE validation-unseen with the 4B backbone and expert-only training data. The policy-loss coefficient is fixed to one. Lower NE and higher SR and SPL indicate better performance. Bold denotes the best result in each column.

$(\lambda_f, \lambda_i, \lambda_g)$	NE (m)	SR (%)	SPL (%)
$(0, 0, 0)$	6.00	50.1	45.1
$(0.05, 0.05, 0.05)$	5.35	53.4	48.5
$(0.1, 0.05, 0.1)$	5.20	54.6	49.6
$(0.1, 0.1, 0.1)$	**5.13**	**55.1**	**50.1**
$(0.2, 0.2, 0.2)$	5.29	53.8	48.9

C.3 Spatial Fusion and Target Construction

The ablation protocol separates spatial input fusion from future-state target construction through two complementary comparisons. The spatial-fusion comparison varies the residual weight over $\alpha \in \{0, 0.1, 0.2, 0.5\}$ while keeping auxiliary supervision fixed. The $\alpha = 0$ variant removes spatial features from the policy input but retains the frozen encoder for training targets, isolating the contribution of spatial input fusion. The target-construction comparison contrasts flattened adaptive pooling with channel-preserving averaging over spatial patches. Both constructions produce 2048-dimensional vectors, allowing the prediction-head architecture and loss coefficients to remain fixed. Each variant uses a consistent target definition during training and evaluation. Joint analysis of navigation metrics and latent prediction error relates the quality of future-state prediction to downstream navigation performance.

C.4 Training Data and Optimization Budget

The training-data comparison considers four settings: expert-only training, expert data augmented with ScaleVLN, expert data augmented with DAgger rollouts, and expert data augmented with both sources. The comparison protocol accounts for both the number of unique training samples and the number of optimizer updates. Holding the update budget fixed controls for the additional computation introduced by larger datasets under a fixed epoch count. Training efficiency is assessed through SR and SPL at common fractions of the training schedule for the policy-only and full world-action models. This protocol separates the contribution of training-data composition from that of optimization progress and complements the main-text comparisons.

C.5 Qualitative Experiments

In this section, we present additional visualizations of ***FutureNav*** in both simulated and real-world environments to facilitate a more detailed qualitative analysis.

For the simulated environments shown in Fig. 8, we selected tasks from R2R-CE and RxR-CE that involve complex trajectories and long instructions of increasing difficulty. Observations show that the previous state-of-the-art baseline, JanusVLN Zeng et al. (2026), often misses or overshoots the correct target location. In contrast, by modeling future states, our method follows instructions more effectively and accurately predicts target locations. Thus, our method successfully handles more challenging tasks through unified world-action modeling.

Regarding the real-world scenarios shown in Fig. 9, we deployed the system for VLN tasks across diverse locations. These settings encompass varying lighting conditions, floor colors, and both indoor and outdoor spaces, demonstrating the excellent generalization capabilities of ***FutureNav***.

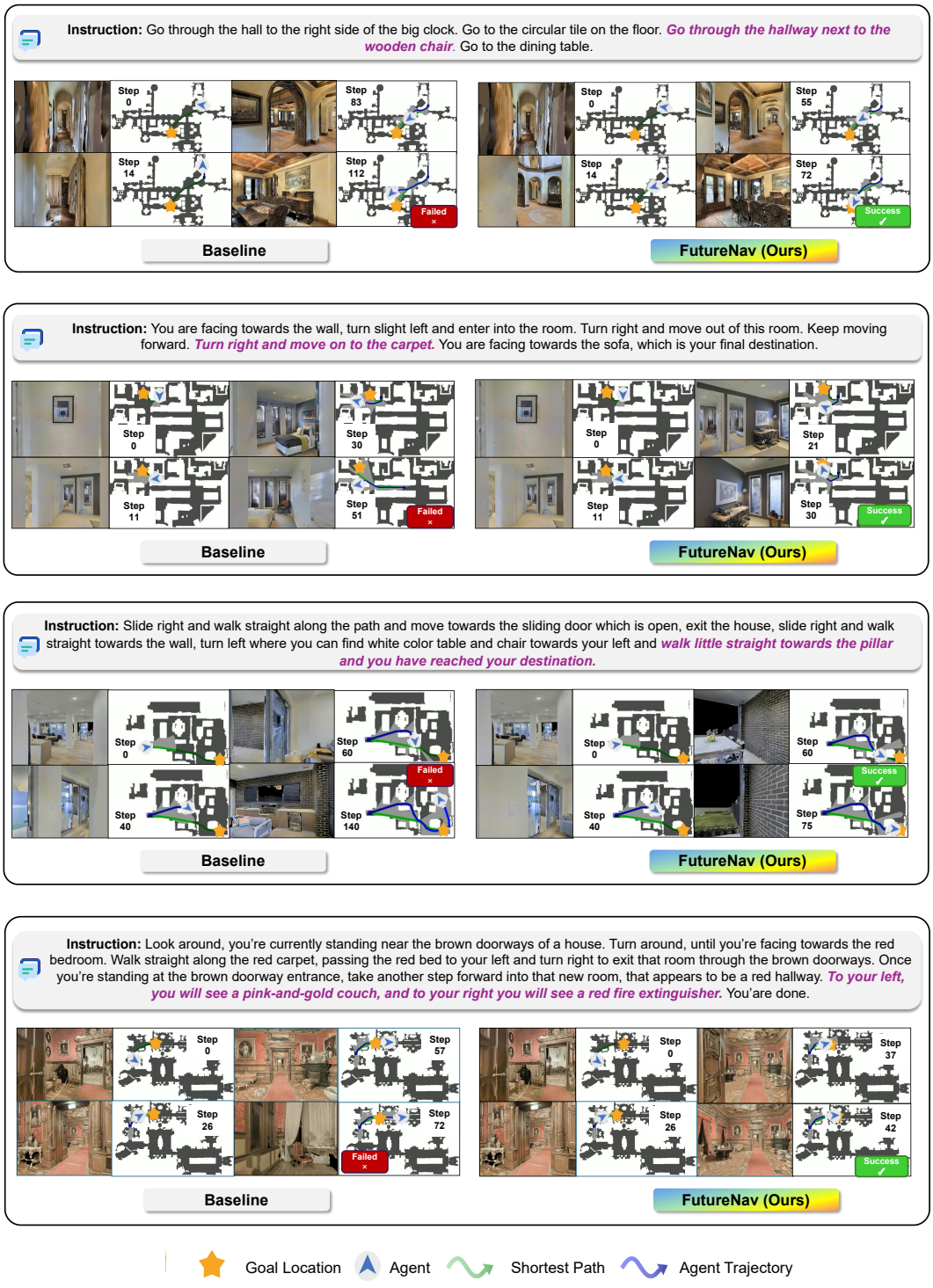

Figure 8: More simulated qualitative comparison with ***FutureNav*** and JanusVLN Zeng et al. (2026).

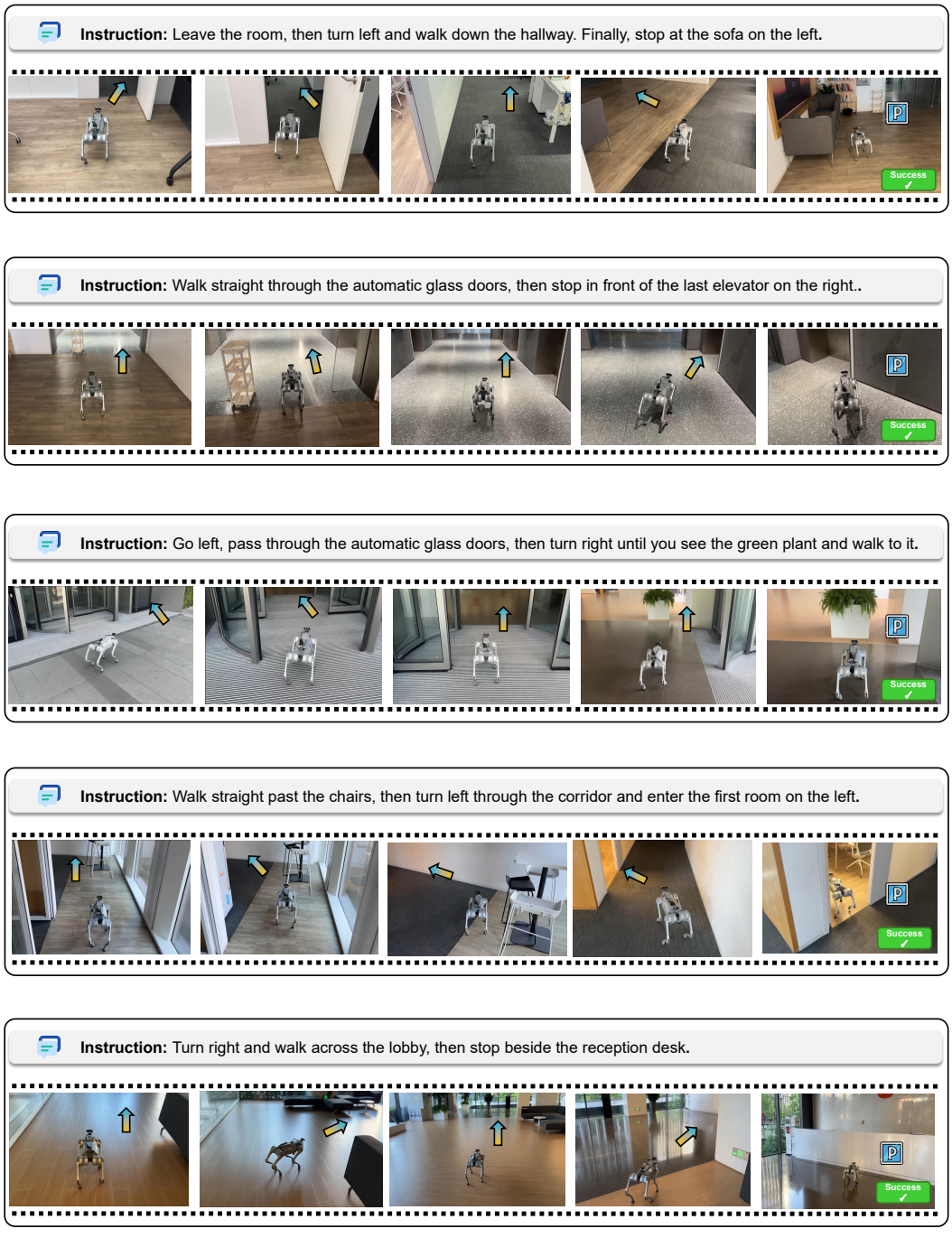

Figure 9: More zero-shot real-world qualitative results of ***FutureNav***.