ANCHORS-GUIDED LATENT WORLD MODEL FOR LONG-HORIZON VISION-AND-LANGUAGE NAVIGATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision-and-Language Navigation (VLN) requires agents to ground language instructions into actions under partial observability. The task remains challenging, particularly in long-horizon settings, where accumulated trajectory errors during planning lead to progressive deviation from the goal, degrading navigation efficiency. Recent methods introduce world models to anticipate future navigation states and improve candidate action evaluation, which can be broadly grouped into two classes. One line uses future-state predictions to refine tentative policy decisions, while the other explicitly searches over alternative trajectories to select actions. However, these methods struggle to balance rollout depth and candidate-action coverage under a limited computational budget. To address this challenge, we propose AnchorWorld, which uses sparse prior anchors, explicitly supervised by future navigation states, as conditioning information to guide the WM in generating a compact latent rollout for each candidate action. This provides route-level guidance for candidate evaluation without explicit tree search. Extensive experiments on the R2R and R4R datasets demonstrate the effectiveness of the proposed method. The advantages are most pronounced in long-horizon navigation, demonstrating improved efficiency and robustness in long-term decision-making.

1 INTRODUCTION

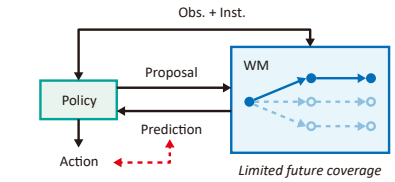
Vision-and-Language Navigation (VLN) (Anderson et al., 2018b; Krantz et al., 2020; Zhang et al., 2024) requires an agent to follow natural language instructions and navigate to a target location in previously unseen environments under partial observability. While significant advances have been made on standard datasets, long-horizon navigation remains a persistent challenge. Accumulated trajectory errors compound over time, causing progressive deviation from the intended route and degrading navigation efficiency. A key challenge is that standard VLN agents lack the ability to anticipate future consequences before committing to an action, making long-horizon planning brittle.

World models (WMs) (Ha & Schmidhuber, 2018; Ding et al., 2025; Baldassarre et al., 2025; Huang et al., 2026; Liu et al., 2025; Wang et al., 2023a) offer a principled way to support such foresight by predicting future navigation states. Their usefulness, however, depends not only on how far they predict, but also on how candidate futures are constructed for decision-making. From this perspective, we distinguish two representative ways of organizing WM-based navigation: *prediction-based action refinement* and *search-based trajectory planning*.

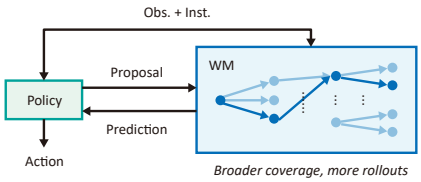
In *prediction-based action refinement*, illustrated in Figure 1, a navigation policy first proposes a tentative action. The WM predicts the resulting state, and this prediction is fed back to the policy to refine its decision (Huang et al., 2026; Li & Bansal, 2023a; Yao et al., 2025). Although the policy can subsequently choose a different action, alternatives that were not initially selected do not receive their own future-state predictions. The agent therefore lacks direct predictive evidence for comparing these alternatives. This can be limiting on long routes, where the usefulness of a direction may become clear only after several subsequent actions.

In *search-based trajectory planning*, shown in Figure 1, the agent uses WM predictions to explore and compare alternative action sequences (Koh et al., 2021; Wang et al., 2023a; Xu et al., 2025). Through procedures such as beam search or Monte Carlo tree search, it expands candidate paths,

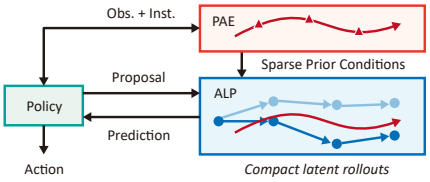
Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping 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.



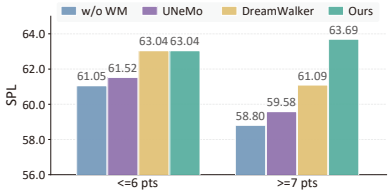
(a) Prediction-based action refinement.



(b) Search-based trajectory planning.



(c) Our anchor-guided imagination.



(d) Performance in long-horizon navigation.

Figure 1: Comparison of representative WM-based navigation designs for VLN. (a) **Prediction-based action refinement** predicts the outcome of a tentative action and feeds the prediction back to the policy to refine its decision. (b) **Search-based trajectory planning** explicitly expands and compares candidate trajectories, requiring additional WM evaluations to explore more paths or predict farther ahead. (c) **AnchorWorld** uses sparse prior anchors as conditioning information to guide the WM in generating a fixed-length latent rollout for each candidate action. (d) SPL on R2R episodes grouped by reference-trajectory length. UNeMo and DreamWalker represent prediction-based action refinement and search-based trajectory planning, respectively. Both baselines obtain lower SPL on trajectories exceeding six waypoints, while AnchorWorld achieves higher SPL on these longer trajectories.

predicts the states along them, and evaluates their outcomes before selecting the next action. A navigation policy may still guide path expansion, while explicit search enables comparison across different continuations. This allows the agent to examine consequences that emerge several steps ahead. However, evaluating more paths or extending them farther requires additional WM predictions. A limited computational budget therefore constrains both the number of alternatives explored and the depth of their evaluation.

These limitations highlight a trade-off between candidate coverage and rollout cost in long-horizon navigation. As shown in Figure 1d, both the refinement-based and search-based baselines achieve lower SPL on R2R (Anderson et al., 2018b) trajectories with more than six waypoints than on shorter ones. We therefore argue that effective long-horizon foresight requires structured, route-level guidance for candidate-conditioned prediction. A sparse representation of the remaining route can help focus imagination on task-relevant futures, while action conditioning allows the consequences of different candidates to be modeled separately. Together, these properties provide a basis for compact, goal-directed imagination.

Motivated by this analysis, we propose AnchorWorld, a latent WM framework for *anchor-guided imagination*, as illustrated in Figure 1c. AnchorWorld comprises two components. First, the *Prior Anchor Extractor* (PAE) infers sparse latent anchors from navigation history and instructions, supervised by future states uniformly sampled from the shortest-path trajectory. Second, the *Anchor-based Latent Planner* (ALP) generates a fixed-length latent rollout for each candidate action, conditioned on local observations and these global anchors. The resulting candidate-specific future representations support policy learning without explicit tree search. After each executed action, PAE refreshes the anchors using the updated navigation history to track the agent's progress.

To summarize, our contributions are as follows: (1) We propose AnchorWorld, which introduces sparse prior anchors to guide candidate-conditioned imagination for efficient long-horizon VLN. (2) We develop PAE to learn route-level priors under shortest-path supervision and ALP to generate action-conditioned latent futures under their guidance, providing compact predictive representations for policy learning. (3) Extensive experiments on R2R and R4R demonstrate state-of-the-art performance, with notable gains in long-horizon navigation efficiency.

2 Related Works

Vision-and-Language Navigation. VLN requires an agent to follow natural language instructions to navigate in previously unseen environments (Anderson et al., 2018b; Krantz et al., 2020; Zhang et al., 2024). Existing work primarily focuses on improving representation learning and policy learning to enhance navigation performance. On the representation side, existing methods learn robust and generalizable multimodal representations via structured scene and history modeling (Chen et al., 2021b; Liu et al., 2023; 2024b; Wang et al., 2023b), data augmentation (Li & Bansal, 2023b; Li et al., 2022; Lin et al., 2023; Zhan et al., 2023; Wang et al., 2023c; 2024b), and large-scale multimodal pretraining (Chen et al., 2022; Wu et al., 2024; An et al., 2023; Hong et al., 2021b; Hao et al., 2020). On the policy side, prior work improves navigation decision-making by exploring more effective learning strategies (Chen et al., 2021a; Zhao et al., 2023; Wang et al., 2023b; 2018; 2019; Liu et al., 2024a), refining decision paradigms (Wang et al., 2020; He et al., 2021; Gao et al., 2023), and introducing auxiliary tasks (Li et al., 2023; Zhu et al., 2020; Ma et al., 2019) to stabilize training and enhance exploration and planning capabilities. In this paper, we propose an anchor-guided future prediction model to enhance environmental perception and support long-horizon policy learning.

World Models. World models learn predictive environment dynamics from high-dimensional experience and enable agents to reason about future consequences before acting (Ding et al., 2025). Recent studies can be broadly divided into two paradigms (Baldassarre et al., 2025). The first paradigm aims to reconstruct future visual observations (Zhen et al., 2024; Zhao et al., 2024; Wang et al., 2025; Bar et al., 2025; Hafner et al., 2019b), providing interpretable predictions while incurring substantial computational overhead for multi-step, multi-candidate rollout due to dense visual decoding. The second paradigm models dynamics in a compact latent or representation space (Assran et al., 2023; 2025; Bu et al., 2025; Yao et al., 2025; Hafner et al., 2019a; 2023; 2020; Maes et al., 2026; Zhou et al., 2024), learning transitions between latent states and leveraging imagination for policy learning, thereby avoiding explicit reconstruction of visual observations at each planning step. Inspired by the second paradigm, our work performs future prediction in a latent navigation state space, while further introducing instruction-grounded priors to guide long-horizon rollout.

World Model for VLN. Recent VLN methods incorporate world models to anticipate future navigation states (Li & Bansal, 2023a;b; Wang et al., 2018; Xu et al., 2025; Huang et al., 2026; Koh et al., 2021; Wang et al., 2023a; 2024a; Liu et al., 2025). Depending on how candidate futures are organized for decision-making, these methods can be broadly categorized into prediction-based action refinement and search-based trajectory planning. Refinement-based methods predict the outcome of a tentative action, either as future visual observations (Li & Bansal, 2023a) or as compact latent states (Huang et al., 2026; Yao et al., 2025), and feed this prediction back to the policy to refine its decision. However, unselected alternatives receive no predicted futures of their own, which leaves the agent without direct evidence to compare them on long routes. In contrast, search-based methods roll out and compare alternative action sequences. They either synthesize future observations along candidate paths in pixel or feature space (Koh et al., 2021; Wang et al., 2023a; 2024a) or estimate long-term outcomes in a learned latent space (Wang et al., 2018; Xu et al., 2025; Liu et al., 2025). Although such rollouts can reveal consequences several steps ahead, each additional candidate or rollout step requires further world-model predictions. A limited computational budget therefore constrains both the breadth and the depth of the search. Different from prior methods, we extract priors from navigation history and instructions to guide multi-step latent space prediction, improving policy learning for long-horizon navigation.

3 Methods

3.1 Problem Formulation

Problem Setup. In the VLN tasks (Jain et al., 2019; Chen et al., 2022; Anderson et al., 2018b; Ku et al., 2020; Qi et al., 2020; Zhu et al., 2021), the environment is represented as an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ denotes navigable viewpoints and $\mathcal{E}$ their connectivity. Given a natural language instruction $\mathcal{X}$, let $\mathcal{W} = \{w_i\}_{i=1}^{L}$ denote its contextualized representations with L words. The agent is initialized at v_0 and incrementally explores the environment by maintaining a partial graph $\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}_t) \subseteq \mathcal{G}$. At each step t, the agent at v_t receives a panoramic observation $\mathcal{R}_t = \{r_{t,i}\}_{i=1}^{n}$, where n is typically set to 36 in discrete VLN settings, and each view encodes visual appearance and

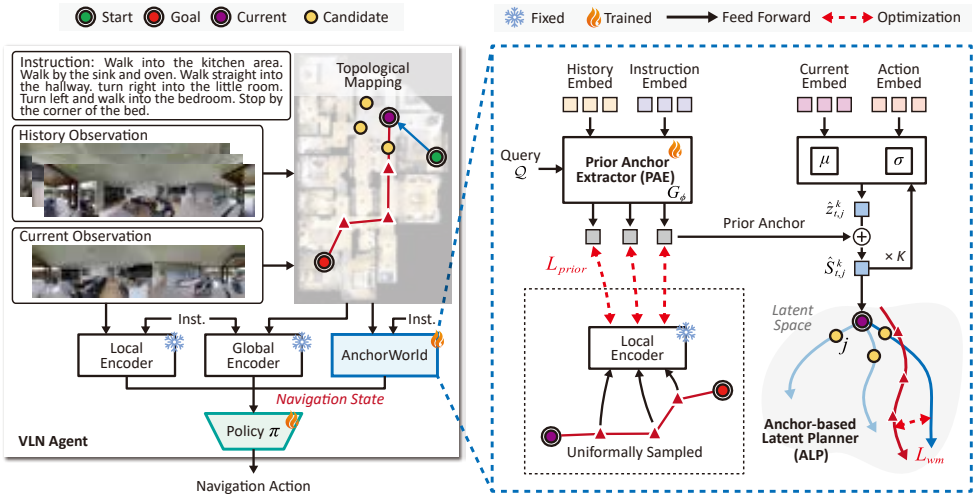


Figure 2: The proposed AnchorWorld framework for long-horizon VLN. We introduce sparse prior anchors to encode global navigation structure from history and language instructions. PAE predicts these anchors in a latent space with future node state supervision, while ALP leverages them to imagine multi-step candidate-action futures. The resulting future-aware representations are fused with topological and panoramic features for policy learning. During testing, AnchorWorld relies only on predicted anchors and observed navigation context.

orientation information. The action space is defined as $\mathcal{A}_t = \{\text{stop}\} \cup \mathcal{N}(v_t)$, where $\mathcal{N}(v_t)$ denotes the set of navigable neighboring viewpoints of v_t.

Graph-based Navigator. We build on DUET (Chen et al., 2022), a graph-based policy widely adopted in recent VLN work (Xu et al., 2025; Lin et al., 2023; Wang et al., 2023c). At step t, the agent maintains a topological map $\mathcal{G}_t$ with candidate nodes $\{v_{t,j}\}_{j=1}^{N_t}$ and representations $\mathcal{S}_t = \{s_{t,j}\}_{j=1}^{N_t}$. It also observes a panorama $\mathcal{R}_t$ at its current node. Conditioned on the instruction embedding $\mathcal{W}$, a global branch reasons over $\mathcal{G}_t$, and a local branch grounds actions in $\mathcal{R}_t$:

$$\ell_t^g = f_g(\mathcal{G}_t, \mathcal{W}), \qquad \ell_t^l = f_l(\mathcal{R}_t, \mathcal{W}). \tag{1}$$

The local logits are mapped onto graph nodes and fused with the global logits by a learned gate ψ:

$$\pi(a \mid \mathcal{W}, \mathcal{G}_t) = \text{softmax}_{a \in \mathcal{A}_t}\big(\psi(\ell_t^g(a),\, \ell_t^l(a))\big), \qquad a_t = \arg\max_{a \in \mathcal{A}_t} \pi(a \mid \mathcal{W}, \mathcal{G}_t). \tag{2}$$

Overview. To improve efficiency in long-horizon navigation, we propose AnchorWorld. As illustrated in Figure 2, a Prior Anchor Extractor (PAE) infers sparse anchors to capture global navigation structure from history and instructions, while an Anchor-based Latent Planner (ALP) performs anchor-conditioned rollout to model future actions. The resulting representations are fused with topological and panoramic features for policy learning.

3.2 Prior Anchor Extractor

In long-horizon VLN, to mitigate semantic drift during execution, we introduce an explicit global-aware representation to guide planning. Such a representation is required to (i) provide a coarse global perception of the remaining trajectory for directional planning, and (ii) encode future structure in a sparse, learnable form instead of dense future reconstruction, reducing computational cost and overfitting risk.

To this end, we propose a Prior Anchor Extractor (PAE) that performs world modeling over historical context and instructions and predicts an ordered sequence of sparse future anchors as an explicit global representation. For supervision, we uniformly sample anchor points along the shortest path between the current and goal nodes, encouraging the predicted anchors to align with the global trajectory geometry. Supervision uses latent node-state features extracted from the local branch. This

design aligns the learned prior with the navigation environment in a shared latent space, enabling direct integration into trajectory rollout and planning.

Specifically, the historical state sequence is defined as $\mathcal{H}_t = (s_{t-L_h+1}, \ldots, s_t)$, where $s_t \in \mathbb{R}^d$ is the current node state and L_h is the maximum history window length. These historical states are the representations of visited nodes maintained in the global branch's topological map. We introduce K learnable anchor queries $\mathcal{Q} \in \mathbb{R}^{K\times d}$ and denote PAE by G_θ with learnable parameters θ. It consists of a two-layer Transformer encoder that models a navigational prior world and predicts sparse future trajectory anchors:

$$\hat{\mathcal{P}}_t = G_\theta\big(\mathcal{H}_t, \mathcal{W}, \mathcal{Q}\big), \tag{3}$$

where $\hat{\mathcal{P}}_t = (\hat{p}_t^1, \ldots, \hat{p}_t^K)$ denotes a set of K sparse prior anchors summarizing the remaining trajectory from node v_t, providing coarse-grained directional guidance for action generation.

Let $\mathcal{Y}_t^\star = (y_t^{1\star}, \ldots, y_t^{K\star})$ denote ground-truth anchors obtained via uniform sampling along the shortest path. The PAE module is optimized with an MSE objective, resulting in $\mathcal{L}_{\mathrm{PAE}}$. Please refer to the appendix for detailed explanation.

3.3 Anchor-based Latent Planner

Prior WM-based navigation planners typically perform rollouts in the observation space to generate future predictions for action planning (Bar et al., 2025; Zhao et al., 2024; Wang et al., 2025; Li & Bansal, 2023b; Koh et al., 2021; Wang et al., 2023a), which is computationally expensive. Inspired by latent world models (Maes et al., 2026; Zhou et al., 2024), we predict future node states on a topological graph, avoiding pixel-level generation while preserving local scene structure. To mitigate semantic drift during rollout, we introduce an Anchor-based Latent Planner (ALP), where explicit anchor-based priors provide global directional guidance at each step, thereby improving long-horizon state consistency.

The module is trained with a Conditional Variational Autoencoder (CVAE) (Sohn et al., 2015) objective to learn a continuous latent space for navigation. At time step t, PAE produces the ordered prior-anchor sequence $\hat{\mathcal{P}}_t = (\hat{p}_t^k)_{k=1}^K$, which provides guidance for a K-step rollout. Conditioned on $\hat{\mathcal{P}}_t$, the latent world model predicts future states for each candidate trajectory, enabling trajectory evaluation in a unified semantic space. For each candidate node $v_{t,j} \in \mathcal{C}_t$, let $c_{t,j}$ denote its action embedding. The latent world model is formulated as

$$\hat{\mathcal{S}}_{t,j} \sim p_\psi(\cdot \mid s_t, c_{t,j}, \hat{\mathcal{P}}_t), \tag{4}$$

where $\hat{\mathcal{S}}_{t,j} = (\hat{s}_{t,j}^1, \ldots, \hat{s}_{t,j}^K)$ is the future node-state sequence predicted for candidate j. Each candidate receives one K-step latent rollout, with step k conditioned on anchor $\hat{p}_t^k$.

At the k-th rollout step, we encode the candidate action $c_{t,j}$, previous state $\hat{s}_{t,j}^{k-1}$, and prior constraint $\hat{p}_t^k$ into a unified context feature $u_{t,j}^k = \phi_{\mathrm{ctx}}([c_{t,j}, \hat{s}_{t,j}^{k-1}, \hat{p}_t^k])$, and model the future trajectory rollout as a Gaussian distribution with mean $\boldsymbol{\mu}_{t,j}^k = \phi_\mu(u_{t,j}^k)$ and variance $(\boldsymbol{\sigma}_{t,j}^k)^2 = \exp(\phi_\sigma(u_{t,j}^k))$, where

$$p_\psi(z_{t,j}^k \mid c_{t,j}, \hat{s}_{t,j}^{k-1}, \hat{p}_t^k) = \mathcal{N}\big(\boldsymbol{\mu}_{t,j}^k, \mathrm{diag}((\boldsymbol{\sigma}_{t,j}^k)^2)\big). \tag{5}$$

The decoder samples $z_{t,j}^k = \boldsymbol{\mu}_{t,j}^k + \boldsymbol{\sigma}_{t,j}^k \odot \boldsymbol{\epsilon}$, $\boldsymbol{\epsilon} \sim \mathcal{N}(0, I)$ Kingma & Welling (2013), and recursively updates $\hat{s}_{t,j}^k = \phi_{\mathrm{state}}([\hat{s}_{t,j}^{k-1}, c_{t,j}, \hat{p}_t^k, z_{t,j}^k])$ for K steps.

During training, we optimize the ALP with an MSE loss between predicted and ground-truth node states, resulting in $\mathcal{L}_{\mathrm{ALP}}$.

3.4 Policy Learning

Prior work has shown that predicting future states can refine graph node representations and thereby facilitate policy learning Huang et al. (2026). However, long-horizon prediction is inherently prone to error accumulation, so future information should not be imposed directly on decision making. Instead, it should serve as a learnable auxiliary signal Xu et al. (2025).

Table 1: Performance comparison on the R2R dataset. * indicates reproduced results.

Methods	Val Seen					Val Unseen					Test Unseen				
	TL	NE↓	OSR↑	SR↑	SPL↑	TL	NE↓	OSR↑	SR↑	SPL↑	TL	NE↓	OSR↑	SR↑	SPL↑
Seq2Seq (Anderson et al., 2018b)	11.33	6.01	53	39	–	8.39	7.81	28	21	–	8.13	7.85	27	20	–
RCM (Wang et al., 2019)	10.65	3.53	75	67	–	11.46	6.09	50	43	–	11.97	6.12	50	43	38
PREVALENT (Hao et al., 2020)	10.32	3.67	–	69	65	10.19	4.71	–	58	53	10.51	5.30	–	54	51
HAMT (Chen et al., 2021b)	11.15	2.51	–	76	72	11.46	3.62	–	66	61	12.27	3.93	–	65	60
DUET (Chen et al., 2022)	12.32	2.28	86	79	73	13.94	3.31	81	72	60	14.73	3.65	76	69	59
BSG (Liu et al., 2023)	–	–	–	–	–	14.90	**2.89**	–	74	62	14.86	3.19	–	73	62
Lily (Lin et al., 2023)	–	–	–	–	–	14.58	2.90	–	74	62	16.13	3.44	–	72	60
ENP (Liu et al., 2024a)	–	–	–	–	–	14.23	3.00	–	74	60	14.27	3.39	–	71	60
PanoGen (Li & Bansal, 2023b)	–	–	–	–	–	13.40	3.03	–	**74.0**	**64.3**	14.38	3.31	–	71.7	61.9
VLN-SIG (Li & Bansal, 2023a)	–	–	–	–	–	–	–	–	72	62	–	–	–	72	60
UNeMo (Huang et al., 2026)	–	–	–	–	–	13.10	3.00	79.6	72.1	61.1	13.90	3.10	78.4	72.5	61.3
UNeMo* (Huang et al., 2026)	11.76	1.94	88.1	81.9	77.0	13.25	3.20	78.8	71.0	60.9	14.39	3.47	77.4	71.0	60.1
DreamWalker* (Wang et al., 2023a)	11.99	2.03	88.0	81.6	76.8	13.49	3.07	81.0	72.7	62.4	14.38	3.17	79.0	73.2	62.8
NavQ (Xu et al., 2025)	–	–	–	–	–	13.80	3.06	–	73	63	14.41	3.3	–	72	63
AnchorWorld (Ours)	11.14	**1.73**	**88.3**	**83.1**	**79.4**	13.13	2.90	**81.0**	72.7	63.3	13.59	**3.08**	**79.1**	**73.8**	**65.0**

To this end, we first aggregate the predicted future states of each candidate node with temporal attention. We then fuse the aggregated context into the node representation $s_{t,j}$ through a learnable gated residual. In this way, future states act not only as an auxiliary supervisory signal but also as feedback propagated into the topological map for node refinement and action inference, which provides more stable guidance for policy learning.

Specifically, for the j-th candidate node $v_{t,j}$ at time step t, we denote its predicted future state sequence as $\hat{\mathcal{S}}_{t,j} = (\hat{s}^1_{t,j}, \dots, \hat{s}^K_{t,j})$. To obtain a compact representation of the future trajectory, we apply a temporal attention mechanism to aggregate the sequence as

$$\bar{s}_{t,j} = \sum_{k=1}^{K} \frac{\exp(\mathbf{w}_p^\top \hat{s}^k_{t,j})}{\sum_{k'=1}^{K} \exp(\mathbf{w}_p^\top \hat{s}^{k'}_{t,j})} \hat{s}^k_{t,j}. \tag{6}$$

The aggregated future context is then integrated with the current node representation via a learnable weighted residual fusion $\tilde{s}_{t,j} = s_{t,j} + \sigma(f_\lambda(\cdot)) \cdot \bar{s}_{t,j}$. This yields an updated set of node representations $\tilde{\mathcal{S}}_t = \{\tilde{s}_{t,j}\}_{j=1}^{N_t}$, from which we construct an updated graph $\tilde{\mathcal{G}}_t$. The resulting graph encodes future-aware node representations, enabling representation refinement with a soft inductive bias from long-horizon predictions rather than directly dominating the decision process. We further use $\tilde{s}_{t,j}$ as a future-aware context to reparameterize the panoramic observation $\mathcal{R}_t$. For each $r_{t,i} \in \mathcal{R}_t$ that leads to $v_{t,j}$, we obtain $\tilde{r}_{t,i} = \phi(r_{t,i}, \tilde{s}_{t,j})$ and $\tilde{\mathcal{R}}_t = \{\tilde{r}_{t,i}\}_{i=1}^{|\mathcal{R}_t|}$, where $\phi(\cdot)$ is a learnable future-conditioned adapter. Following Eqs. 1 and 2, the policy becomes

$$\begin{aligned} \tilde{\ell}^g_t = f_g(\tilde{\mathcal{G}}_t, \mathcal{W}), \quad \tilde{\ell}^l_t = f_l(\tilde{\mathcal{R}}_t, \mathcal{W}), \\ \pi(a \mid \mathcal{W}, \tilde{\mathcal{G}}_t) = \mathrm{softmax}_{a \in \mathcal{A}_t} \left(\psi(\tilde{\ell}^g_t(a),\, \tilde{\ell}^l_t(a))\right). \end{aligned} \tag{7}$$

Following the convention Chen et al. (2022); Huang et al. (2026); Xu et al. (2025), we combine behavior cloning and DAgger (Ross et al., 2011):

$$\mathcal{L}_{\text{nav}} = -\lambda \sum_{t=1}^{T} \log \pi(a^*_t \mid \mathcal{W}, \tilde{\mathcal{G}}_t) - \sum_{t=1}^{T'} \log \pi(a'^*_t \mid \mathcal{W}, \tilde{\mathcal{G}}'_t), \tag{8}$$

where a^*_t is the ground-truth action along the demonstration. $\tilde{\mathcal{G}}'_t$ is refined from the graph $\mathcal{G}'_t$ built along a trajectory sampled by the current policy, and a'^*_t is the corresponding pseudo-expert action. T and T' are the trajectory lengths, and λ balances the two terms. The full objective, which jointly trains navigation, anchor prediction, and latent rollout, is $\mathcal{L} = \mathcal{L}_{\text{nav}} + \lambda_p \mathcal{L}_{\text{PAE}} + \lambda_a \mathcal{L}_{\text{ALP}}$, where λ_p and λ_a are loss weights.

4 Experiments

4.1 Experimental Settings

Datasets and Metrics. We evaluate our method on Room-to-Room (R2R) (Anderson et al., 2018b) and Room-for-Room (R4R) (Jain et al., 2019), representing standard and long-horizon VLN datasets.

R2R is built on Matterport3D (Chang et al., 2017) with 90 indoor environments, while R4R extends R2R by concatenating trajectories to form longer and more complex navigation paths for evaluating long-horizon planning. We report five standard VLN metrics, including (i) Trajectory Length (TL); (ii) Navigation Error (NE); (iii) Success Rate (SR); (iv) Oracle Success Rate (OSR); and (v) Success weighted by Path Length (SPL) (Anderson et al., 2018a). Among all metrics, except TL and NE where lower values indicate better performance, higher values represent superior results. Following the literature (Chen et al., 2022; Li & Bansal, 2023b), we consider SPL as the main metric for the R2R and R4R datasets.

Implementation Details. We build our model upon DUET (Chen et al., 2022) while keeping its original navigation backbone unchanged. To ensure a controlled comparison, all pre-trained weights and reported results of the baseline are fully preserved, and our method is introduced exclusively during the fine-tuning stage. The PAE is implemented as a two-layer Transformer. The ALP is formulated within a CVAE framework (Sohn et al., 2015). All models are optimized by the AdamW optimizer with a learning rate of 1e-5 and a batch size of 8. All experiments are conducted on a single NVIDIA A100 GPU. Additional implementation details and analyses are provided in the appendix.

4.2 Comparison with State of the Art

R2R results. We compare AnchorWorld with extensive SOTA VLN methods: (i) Classic Methods Anderson et al. (2018b); Wang et al. (2019); Hao et al. (2020); Chen et al. (2021b; 2022); Liu et al. (2023); Lin et al. (2023); Liu et al. (2024a); Zhang & Kordjamshidi (2023; 2022); Li & Bansal (2023b); (ii) VLN with Prediction-based action refinement Li & Bansal (2023a); Huang et al. (2026); (iii) VLN with search-driven imagination Wang et al. (2023a); Xu et al. (2025).

Table 1 reports the performance comparison on R2R, where * denotes the reproduced results. Our method achieves SOTA performance on the Test Unseen split. Despite slightly lower Validation Unseen SR and SPL than PanoGen Li & Bansal (2023b), possibly due to PanoGen's training augmentation, AnchorWorld achieves lower NE, shorter trajectories, and higher SPL on Test unseen splits, supporting the value of global route guidance for efficient navigation. Compared with prediction-based action refinement methods, AnchorWorld achieves better results on all reported Test Unseen metrics, with SPL gains of 5.0 points over VLN-SIG Li & Bansal (2023a) and 3.7 and 4.9 points over the reported and reproduced UNeMo Huang et al. (2026) results, respectively. It also surpasses search-driven imagination methods, improving SPL by 2.2 points over DreamWalker Wang et al. (2023a) and 2.0 points over NavQ Xu et al. (2025), while achieving higher SR and lower NE. These gains support the effectiveness of route-guided candidate prediction for successful and efficient navigation without explicit tree search. These results support our hypothesis that VLN benefits from explicitly modeled and dynamically updated prior anchors, which provide global guidance for action generation within the latent world model while mitigating redundant visual distractions, thereby improving navigation efficiency.

R4R results. To further evaluate long-horizon navigation performance, we conduct experiments on R4R and report SR, SPL, and NE as the primary efficiency metrics. As shown in Table 4, AnchorWorld achieves the highest SR and SPL of 51.4% and 47.6%, respectively, and the lowest NE of 5.53 m among the compared methods on the R4R Validation Unseen split. Although PanoGen produces shorter trajectories, AnchorWorld improves SR and SPL by 3.6 and 3.3 percentage points, respectively, demonstrating greater success and path efficiency. Compared with the search-based DreamWalker Wang et al. (2023a) , AnchorWorld improves SR and SPL by 1.4 and 1.8 percentage points while shortening trajectories from 20.81 to 20.08 m. These results further support our hypothesis that explicit prior anchors mitigate locally optimal yet inefficient exploration in long-horizon navigation and improve navigation efficiency.

Table 2: Ablation Study on Model Components on the R2R Validation Unseen split.

Index	PAE	ALP	TL	NE↓	OSR↑	SR↑	SPL↑
C1	×	×	13.94	3.31	81	72	60
C2	✓	×	13.57	3.19	79.8	71.8	61.7
C3	×	✓	13.83	3.11	80.0	72.0	61.6
C4	✓	✓	13.13	**2.90**	**81.0**	**72.7**	**63.3**

Table 3: Ablation study on sampling strategies on the R2R Validation Unseen split.

Methods	TL	NE↓	OSR↑	SR↑	SPL↑
random sampling	13.06	3.17	80.1	71.5	61.4
uniform sampling	13.13	**2.90**	**81.0**	**72.7**	**63.3**

Table 4: Performance comparison on the R4R dataset.

Methods	Val Unseen			
	TL	NE↓	SR↑	SPL↑
DUET-CLIP (Chen et al., 2022)	21.04	6.06	46.6	41.9
LOVIS (Zhang & Kordjamshidi, 2022)	–	6.07	45.0	35.0
VLN-Trans (Zhang & Kordjamshidi, 2023)	–	5.87	46.0	36.0
PanoGen (Li & Bansal, 2023b)	18.62	6.02	47.8	44.3
VLN-SIG (Li & Bansal, 2023a)	–	–	45.8	–
DreamWalker* (Wang et al., 2023a)	20.81	5.58	50.0	45.8
AnchorWorld (Ours)	20.08	**5.53**	**51.4**	**47.6**

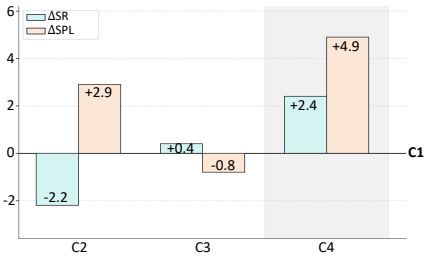


Figure 3: Ablation study of model components on the R2R Val Unseen split with ground-truth trajectory lengths ≥ 7.

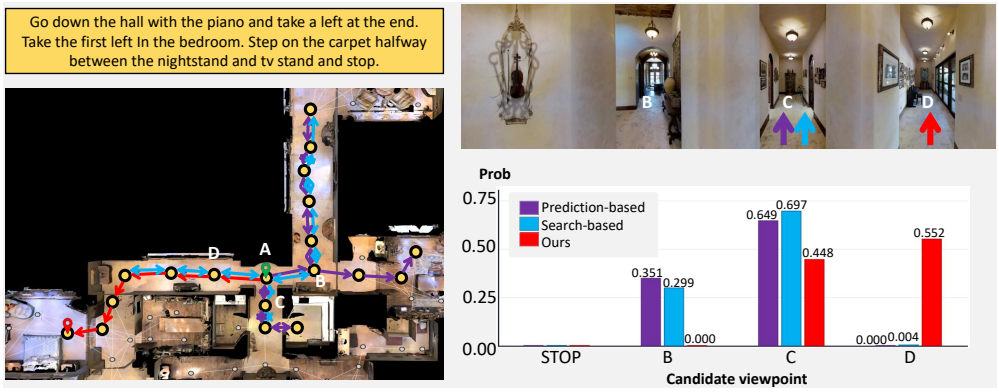


Figure 4: Predicted trajectories of three models on the R2R Val Unseen split, where the green and red landmarks denote the start and target locations. Subfigure A illustrates the distinct actions taken by the models at key locations, while the bar chart displays the probability distributions of the three models across the three candidate viewpoints B, C, and D.

Qualitative examples. Figure 4 shows a representative long-horizon example. Compared to existing WM-based VLN methods, AnchorWorld leverages prior anchors encoding global navigation intent to guide the agent toward the correct region (D). At the ambiguous decision point (A), the candidate directions (B-D) are semantically ambiguous under local perception (see the top-right subfigure). As a result, the other methods tend to select C, driven by stronger local visual affordances (e.g., a visible left-turn opening consistent with the instruction "turn left"), which leads to trajectory deviation, as shown in the bottom-right subfigure. In contrast, AnchorWorld reshapes the action distribution via anchor-based global guidance, suppressing locally attractive but globally suboptimal choices and steering the agent toward D, enabling more consistent goal progression.

Figure 5 illustrates navigation trajectories of three representative methods under different scenarios on the R2R Val Unseen split. Benefiting from prior anchors, AnchorWorld achieves higher navigation efficiency and success rate, while effectively reducing unnecessary exploration.

4.3 Ablation Studies

Ablation Study on Model Components. Table 2 summarizes the contributions of the two key components on the R2R Validation Unseen split, where PAE denotes the Prior Anchor Extractor and ALP denotes the Anchor-based Latent Planner. The compared architectures include: (1) C1, a variant without PAE and ALP; (2) C2, a variant utilizing PAE but not ALP; (3) C3, a variant utilizing ALP but not PAE; and (4) C4, the full AnchorWorld.

As shown in Table 2, adding PAE to the DUET baseline (C2) improves SPL by 1.7%, indicating that prior anchors derived from historical perception provide global directional guidance and reduce redundant exploration. However, SR and OSR do not improve consistently, suggesting that the prior alone may cause premature stopping. Adding only ALP (C3) yields limited gains and produces longer trajectories than C2. This implies that latent future prediction benefits local decisions but cannot suppress ineffective exploration without a global prior. The full model (C4) performs best on all metrics and substantially improves SR and SPL over C3 in unseen environments, confirming

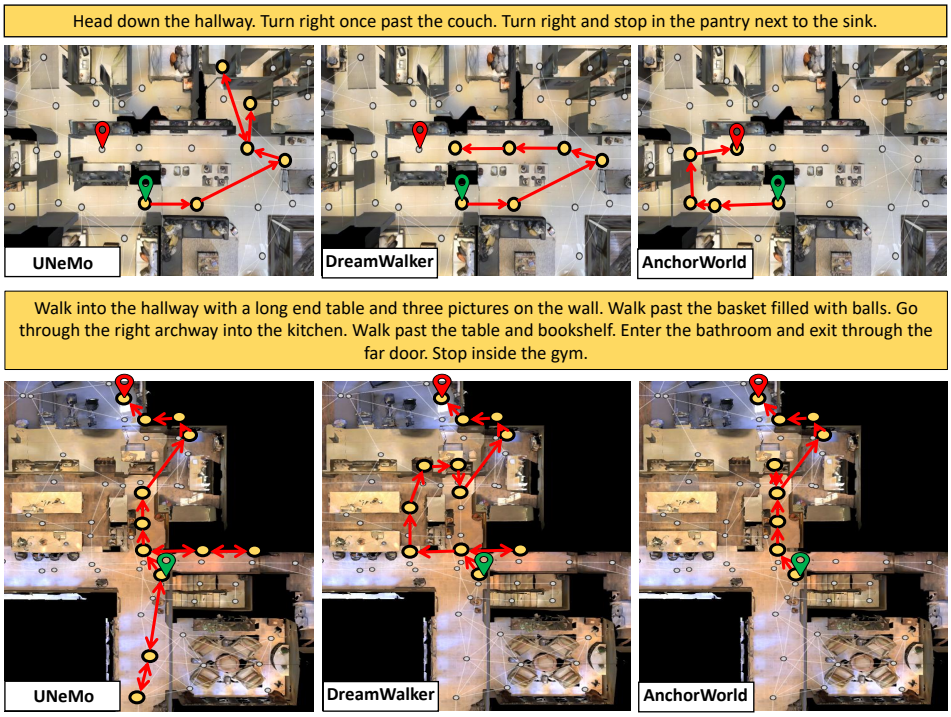


Figure 5: Qualitative comparison with two representative agents. Green and red markers denote the start and goal locations, respectively. Up: Short-horizon comparison; Bottom: long-horizon comparison. AnchorWorld follows the instructions accurately and efficiently, while the other agents either fail or take inefficient detours.

that prior anchors guide latent action generation toward more efficient navigation. Furthermore, the gains of AnchorWorld are more pronounced on long-horizon episodes. As shown in Figure 3, the full model's improvements over DUET on these episodes are substantially larger than those on the full validation split. The two modules are also complementary: PAE improves SPL but slightly lowers SR, whereas ALP improves SR at the expense of SPL. Combining the global directional guidance from PAE with the refined local decisions from ALP improves long-horizon navigation performance.

Ablation Study on Prior Sampling Strategy. Table 3 compares different prior sampling strategies for anchor supervision. Uniform sampling outperforms random sampling on all metrics. This suggests that uniformly distributed anchors provide more stable global guidance and maintain better alignment with navigation progress. These results further validate our design choice of using sparse yet uniformly structured priors to guide future trajectory inference.

Table 5: Ablation study on the number of prior anchors on the R4R Validation Unseen split.

K	NE↓	SR↑	SPL↑
3	5.76	49.7	45.7
4	5.41	50.7	45.8
5	5.53	**51.4**	**47.6**
10	**5.34**	50.6	45.4

Ablation Study on the Number of Prior Anchors. Table 5 shows that increasing the number of prior anchors K from 3 to 5 consistently improves performance on R4R Validation Unseen split, while larger K leads to unstable gains. Although NE peaks at $K = 10$, overly dense anchors introduce redundancy and increased rollout uncertainty, degrading long-horizon robustness. These results suggest that sparse prior anchors provide better global guidance for long-horizon navigation.

5 Conclusion

This paper proposes AnchorWorld, a anchor guided latent world model for long-horizon VLN that learns sparse prior anchors to capture global route structure. PAE supervises anchors with future shortest-path states, while ALP uses them to guide multi-step latent rollouts for goal-directed action generation. Experiments on R2R and R4R datasets demonstrate SOTA performance, with notable gains in navigation efficiency and long-horizon robustness.

AI USE STATEMENT

Generative AI tools were used during the preparation of this work to assist with language editing and improving the clarity and organization of the manuscript, as well as for literature search and summarization. They were also used to provide feedback on the presentation of mathematical formulations and experimental methodology. All research ideas, methodological decisions, experimental results, and conclusions were independently examined and validated by the authors. Relevant literature suggested by AI tools was manually verified against the original sources, and all AI-assisted text and technical content were reviewed and revised by the authors. The authors take full responsibility for the final content of this work, including all text, claims, results, and artifacts produced with the aid of generative AI.

REFERENCES

Dong An, Yuankai Qi, Yangguang Li, Yan Huang, Liang Wang, Tieniu Tan, and Jing Shao. Bevbert: Multimodal map pre-training for language-guided navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2737–2748, 2023.

Peter Anderson, Angel Chang, Devendra Singh Chaplot, Alexey Dosovitskiy, Saurabh Gupta, Vladlen Koltun, Jana Kosecka, Jitendra Malik, Roozbeh Mottaghi, Manolis Savva, et al. On evaluation of embodied navigation agents. *arXiv preprint arXiv:1807.06757*, 2018a.

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, 2018b.

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 15619–15629, 2023.

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, et al. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

Federico Baldassarre, Marc Szafraniec, Basile Terver, Vasil Khalidov, Francisco Massa, Yann LeCun, Patrick Labatut, Maximilian Seitzer, and Piotr Bojanowski. Back to the features: Dino as a foundation for video world models. *arXiv preprint arXiv:2507.19468*, 2025.

Amir Bar, Gaoyue Zhou, Danny Tran, Trevor Darrell, and Yann LeCun. Navigation world models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 15791–15801, 2025.

Qingwen Bu, Yanting Yang, Jisong Cai, Shenyuan Gao, Guanghui Ren, Maoqing Yao, Ping Luo, and Hongyang Li. Univla: Learning to act anywhere with task-centric latent actions. *arXiv preprint arXiv:2505.06111*, 2025.

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.

Kevin Chen, Junshen K Chen, Jo Chuang, Marynel Vázquez, and Silvio Savarese. Topological planning with transformers for vision-and-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11276–11286, 2021a.

Shizhe Chen, Pierre-Louis Guhur, Cordelia Schmid, and Ivan Laptev. History aware multimodal transformer for vision-and-language navigation. *Advances in neural information processing systems*, 34:5834–5847, 2021b.

Shizhe Chen, Pierre-Louis Guhur, Makarand Tapaswi, Cordelia Schmid, and Ivan Laptev. Think global, act local: Dual-scale graph transformer for vision-and-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16537–16547, 2022.

Yibo Cui, Liang Xie, Yakun Zhang, Meishan Zhang, Ye Yan, and Erwei Yin. Grounded entity-landmark adaptive pre-training for vision-and-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12043–12053, 2023.

Jingtao Ding, Yunke Zhang, Yu Shang, Yuheng Zhang, Zefang Zong, Jie Feng, Yuan Yuan, Hongyuan Su, Nian Li, Nicholas Sukiennik, et al. Understanding world or predicting future? a comprehensive survey of world models. *ACM Computing Surveys*, 58(3):1–38, 2025.

Chen Gao, Xingyu Peng, Mi Yan, He Wang, Lirong Yang, Haibing Ren, Hongsheng Li, and Si Liu. Adaptive zone-aware hierarchical planner for vision-language navigation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14911–14920, 2023.

Pierre-Louis Guhur, Makarand Tapaswi, Shizhe Chen, Ivan Laptev, and Cordelia Schmid. Airbert: In-domain pretraining for vision-and-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1634–1643, 2021.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019a.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pp. 2555–2565. PMLR, 2019b.

Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering atari with discrete world models. *arXiv preprint arXiv:2010.02193*, 2020.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

Weituo Hao, Chunyuan Li, Xiujun Li, Lawrence Carin, and Jianfeng Gao. Towards learning a generic agent for vision-and-language navigation via pre-training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 13137–13146, 2020.

Keji He, Yan Huang, Qi Wu, Jianhua Yang, Dong An, Shuanglin Sima, and Liang Wang. Landmark-rxr: Solving vision-and-language navigation with fine-grained alignment supervision. *Advances in Neural Information Processing Systems*, 34:652–663, 2021.

Y Hong, Q Wu, Y Qi, C Rodriguez-Opazo, and S Gould. A recurrent vision-and-language bert for navigation. arxiv 2021. *arXiv preprint arXiv:2011.13922*, 1, 2021a.

Yicong Hong, Qi Wu, Yuankai Qi, Cristian Rodriguez-Opazo, and Stephen Gould. Vln bert: A recurrent vision-and-language bert for navigation. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pp. 1643–1653, 2021b.

Changxin Huang, Lv Tang, Zhaohuan Zhan, Lisha Yu, Runhao Zeng, Zun Liu, Zhengjie Wang, and Jianqiang Li. Unemo: Collaborative visual-language reasoning and navigation via a multimodal world model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 18315–18323, 2026.

Vihan Jain, Gabriel Magalhaes, Alexander Ku, Ashish Vaswani, Eugene Ie, and Jason Baldridge. Stay on the path: Instruction fidelity in vision-and-language navigation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 1862–1872, 2019.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Jing Yu Koh, Honglak Lee, Yinfei Yang, Jason Baldridge, and Peter Anderson. Pathdreamer: A world model for indoor navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 14738–14748, 2021.

Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII 16*, pp. 104–120. Springer, 2020.

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.

Jialu Li and Mohit Bansal. Improving vision-and-language navigation by generating future-view image semantics. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10803–10812, 2023a.

Jialu Li and Mohit Bansal. Panogen: Text-conditioned panoramic environment generation for vision-and-language navigation. *Advances in neural information processing systems*, 36:21878–21894, 2023b.

Jialu Li, Hao Tan, and Mohit Bansal. Envedit: Environment editing for vision-and-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15407–15417, 2022.

Xiangyang Li, Zihan Wang, Jiahao Yang, Yaowei Wang, and Shuqiang Jiang. Kerm: Knowledge enhanced reasoning for vision-and-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2583–2592, 2023.

Kunyang Lin, Peihao Chen, Diwei Huang, Thomas H Li, Mingkui Tan, and Chuang Gan. Learning vision-and-language navigation from youtube videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8317–8326, 2023.

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*, 2025.

Rui Liu, Xiaohan Wang, Wenguan Wang, and Yi Yang. Bird's-eye-view scene graph for vision-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10968–10980, 2023.

Rui Liu, Wenguan Wang, and Yi Yang. Vision-language navigation with energy-based policy. *Advances in Neural Information Processing Systems*, 37:108208–108230, 2024a.

Rui Liu, Wenguan Wang, and Yi Yang. Volumetric environment representation for vision-language navigation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16317–16328, 2024b.

Chih-Yao Ma, Jiasen Lu, Zuxuan Wu, Ghassan AlRegib, Zsolt Kira, Richard Socher, and Caiming Xiong. Self-monitoring navigation agent via auxiliary progress estimation. *arXiv preprint arXiv:1901.03035*, 2019.

Lucas Maes, Quentin Le Lidec, Damien Scieur, Yann LeCun, and Randall Balestriero. Leworld-model: Stable end-to-end joint-embedding predictive architecture from pixels. *arXiv preprint arXiv:2603.19312*, 2026.

Yuankai Qi, Qi Wu, Peter Anderson, Xin Wang, William Yang Wang, Chunhua Shen, and Anton van den Hengel. Reverie: Remote embodied visual referring expression in real indoor environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9982–9991, 2020.

Yanyuan Qiao, Yuankai Qi, Yicong Hong, Zheng Yu, Peng Wang, and Qi Wu. Hop+: History-enhanced and order-aware pre-training for vision-and-language navigation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(7):8524–8537, 2023.

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.

Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. *Advances in neural information processing systems*, 28, 2015.

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.

Teng Wang, Xinxin Zhao, Wenzhe Cai, and Changyin Sun. Imaginenav++: Prompting vision-language models as embodied navigator through scene imagination. *arXiv preprint arXiv:2512.17435*, 2025.

Xin Wang, Wenhan Xiong, Hongmin Wang, and William Yang Wang. Look before you leap: Bridging model-free and model-based reinforcement learning for planned-ahead vision-and-language navigation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 37–53, 2018.

Xin Wang, Qiuyuan Huang, Asli Celikyilmaz, Jianfeng Gao, Dinghan Shen, Yuan-Fang Wang, William Yang Wang, and Lei Zhang. Reinforced cross-modal matching and self-supervised imitation learning for vision-language navigation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6629–6638, 2019.

Xin Wang, Qiuyuan Huang, Asli Celikyilmaz, Jianfeng Gao, Dinghan Shen, Yuan-Fang Wang, William Yang Wang, and Lei Zhang. Vision-language navigation policy learning and adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 43(12):4205–4216, 2020.

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, Junjie Hu, Ming Jiang, and Shuqiang Jiang. Lookahead exploration with neural radiance representation for continuous vision-language navigation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 13753–13762, 2024a.

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.

Zun Wang, Jialu Li, Yicong Hong, Songze Li, Kunchang Li, Shoubin Yu, Yi Wang, Yu Qiao, Yali Wang, Mohit Bansal, et al. Bootstrapping language-guided navigation learning with self-refining data flywheel. *arXiv preprint arXiv:2412.08467*, 2024b.

Siying Wu, Xueyang Fu, Feng Wu, and Zheng-Jun Zha. Vision-and-language navigation via latent semantic alignment learning. *IEEE Transactions on Multimedia*, 26:8406–8418, 2024.

Peiran Xu, Xicheng Gong, and Yadong Mu. Navq: Learning a q-model for foresighted vision-and-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6327–6341, 2025.

Xuan Yao, Junyu Gao, and Changsheng Xu. Navmorph: A self-evolving world model for vision-and-language navigation in continuous environments. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5536–5546, 2025.

Zhaohuan Zhan, Liang Lin, and Guang Tan. Object-aware navigation for remote embodied visual referring expression. *Neurocomputing*, 515:68–78, 2023.

Yue Zhang and Parisa Kordjamshidi. Lovis: Learning orientation and visual signals for vision and language navigation. In *Proceedings of the 29th International Conference on Computational Linguistics*, pp. 5745–5754, 2022.

Yue Zhang and Parisa Kordjamshidi. Vln-trans: Translator for the vision and language navigation agent. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13219–13233, 2023.

Yue Zhang, Ziqiao Ma, Jialu Li, Yanyuan Qiao, Zun Wang, Joyce Chai, Qi Wu, Mohit Bansal, and Parisa Kordjamshidi. Vision-and-language navigation today and tomorrow: A survey in the era of foundation models. *arXiv preprint arXiv:2407.07035*, 2024.

Chongyang Zhao, Yuankai Qi, and Qi Wu. Mind the gap: Improving success rate of vision-and-language navigation by revisiting oracle success routes. In *Proceedings of the 31st ACM International Conference on Multimedia*, pp. 4349–4358, 2023.

Xinxin Zhao, Wenzhe Cai, Likun Tang, and Teng Wang. Imaginenav: Prompting vision-language models as embodied navigator through scene imagination. *arXiv preprint arXiv:2410.09874*, 2024.

Haoyu Zhen, Xiaowen Qiu, Peihao Chen, Jincheng Yang, Xin Yan, Yilun Du, Yining Hong, and Chuang Gan. 3d-vla: A 3d vision-language-action generative world model. *arXiv preprint arXiv:2403.09631*, 2024.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. Dino-wm: World models on pre-trained visual features enable zero-shot planning. *arXiv preprint arXiv:2411.04983*, 2024.

Fengda Zhu, Yi Zhu, Xiaojun Chang, and Xiaodan Liang. Vision-language navigation with self-supervised auxiliary reasoning tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10012–10022, 2020.

Fengda Zhu, Xiwen Liang, Yi Zhu, Qizhi Yu, Xiaojun Chang, and Xiaodan Liang. Soon: Scenario oriented object navigation with graph-based exploration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12689–12699, 2021.

A Appendix

A.1 Details of Figure 1d

For the experiments in Figure 1d, we evaluate DUET (Chen et al., 2022), UNeMo (Huang et al., 2026), DreamWalker (Wang et al., 2023a), and our proposed method on a subset of the R2R dataset where the number of ground-truth trajectory waypoints is ≥ 7. Following the convention (Huang et al., 2026), we regard this subset as a representative long-horizon setting within R2R. The results show that all three categories of methods exhibit varying degrees of performance degradation on this subset, particularly in terms of navigation efficiency. This observation motivates us to further investigate how to maintain stable and efficient navigation behavior in long-horizon navigation.

A.2 Details of Datasets and Metrics

Room-to-Room (R2R) consists of approximately 22K human-annotated navigation instructions and 7.2K expert trajectories built on Matterport3D. Each instruction is paired with a trajectory spanning multiple rooms. On average, each instruction contains 32 words, while the corresponding ground-truth path consists of about six waypoints.

Room-for-Room (R4R) is constructed by concatenating adjacent trajectories from the dataset. Unlike shortest-path supervision, its ground-truth trajectories are not guaranteed to be optimal in length. This property motivates us to evaluate navigation performance in long-horizon settings using SR, SPL, and NE, which more directly reflect success, efficiency, and localization accuracy, rather than metrics tailored to instruction adherence.

We evaluate performance using standard metrics, including (i) Trajectory Length (TL), which measures the length of the navigated path; (ii) Navigation Error (NE), which computes the distance between the final predicted position and the ground-truth goal; (iii) Success Rate (SR), which reports the percentage of episodes where the agent successfully reaches the goal; (iv) Oracle Success Rate (OSR), which measures whether any position along the trajectory is within the success threshold of the goal; and (v) Success weighted by Path Length (SPL) (Anderson et al., 2018a), which jointly

considers success and path efficiency by penalizing unnecessarily long trajectories. Among all metrics, except TL and NE where lower values indicate better performance, higher values represent superior results. Following the literature (Chen et al., 2022; Li & Bansal, 2023b), we consider SPL as the main metric for the R2R and R4R datasets.

A.3 Details of Training

Viewpoint state Ground-Truth Construction To supervise PAE and ALP learning, each sampled viewpoint is projected into the same latent state space used by the navigation policy through the node-state encoder, yielding

$$y_t^{k*} = f_l(\mathcal{R}_t^{k*}, \mathcal{W}), \tag{9}$$

where $f_l(\cdot)$ is defined in Eq. 1. Therefore, both PAE and ALP are optimized with an MSE objective, resulting in $\mathcal{L}_{\text{PAE}}$ and $\mathcal{L}_{\text{ALP}}$:

$$\mathcal{L}_{\text{PAE}} = \frac{1}{K}\sum_{k=1}^{K} \left\|\hat{p}_t^k - y_t^{k*}\right\|_2^2 \tag{10}$$

$$\mathcal{L}_{\text{ALP}} = \frac{1}{K}\sum_{k=1}^{K} \left\|\hat{s}_t^k - y_t^{k*}\right\|_2^2 \tag{11}$$

Training Details During the pre-training stage, we strictly follow the official DUET (Chen et al., 2022) configuration, and incorporate our model only during the policy fine-tuning stage.

During the policy fine-tuning stage, we directly reuse the encoders from DUET, as defined in Eq. 1. Following (Huang et al., 2026), we adopt a phased training strategy. The planner-related losses are activated only during a scheduled portion of fine-tuning, preventing the auxiliary latent reasoning task from overwhelming navigation policy learning. In our default setting, the planner is activated for the first 20% of each training phase on R2R, and for the first 30% on R4R. DUET is initialized from the official pre-trained checkpoint and continues to be fine-tuned together with our modules. We use AdamW with a learning rate of 1×10^{-5} and a batch size of 8. A full list of hyperparameters is provided in

Table 6: Detailed hyperparameters used in our experiments. For attributes with two values, the first corresponds to R2R and the second to R4R.

Attribute	Value	Description
d	768	hidden size of the node state representations
L_h	4	history window length for PAE
K	3 / 5	number of prior anchors and the latent rollout horizon for R2R / R4R
d_z	256	dimension of the CVAE latent variable
$N_{\phi_{\text{ctx}}}$	1	number of latent transition layers
$N_{\phi_{\text{state}}}$	1	number of rollout state update layers
λ_p	0.1	weight of the PAE MSE loss
λ_a	0.1	weight of the ALP MSE loss
p_{drop}	0.5	dropout rate
p_{feat}	0.4	feature dropout rate
η	1×10^{-5}	learning rate of AdamW
B	8	batch size
T_{train}	200k	total number of fine-tuning iterations
r_{planner}	0.2 / 0.3	planner activation ratio for R2R / R4R

A.4 More Results

R2R results. Table 7 shows the comparison with additional classical methods. Our approach outperforms all baselines on the R2R Test Unseen split, further demonstrating the effectiveness and soundness of our design.

Table 8: Performance comparison with the classic methods on the REVERIE Validation Unseen split.

	Val Unseen				
	SR↑	OSR↑	SPL↑	RGS↑	RGSPL↑
DUET (Chen et al., 2022)	46.98	51.07	33.73	32.15	23.03
HAMT (Chen et al., 2021b)	32.95	36.84	30.20	18.92	17.28
Lily (Lin et al., 2023)	48.11	53.71	34.43	32.15	23.43
PanoGen (Li & Bansal, 2023b)	51.18	–	34.99	33.26	22.99
UNeMo (Huang et al., 2026)	49.36	–	32.84	34.28	22.90
Ours	**51.38**	**53.91**	**36.71**	**34.54**	**24.70**

Table 7: Performance comparison with the classic methods on the R2R Test Unseen split.

Methods	Test Unseen			
	TL	NE↓	SR↑	SPL↑
Classic Methods for VLN (w/o WM)				
PREVALENT (Hao et al., 2020)	10.51	5.30	54	51
AirBert (Guhur et al., 2021)	12.41	4.13	62	57
VLN⟳BERT (Hong et al., 2021a)	12.35	4.09	63	57
HAMT (Chen et al., 2021b)	12.27	3.93	65	60
HOP+ (Qiao et al., 2023)	12.67	3.71	66	60
DUET (Chen et al., 2022)	14.73	3.65	69	59
EnvEdit (Li et al., 2022)	11.90	3.59	68	64
GELA (Cui et al., 2023)	12.99	3.59	67	62
VLN-Trans (Zhang & Kordjamshidi, 2023)	–	3.94	66	60
BEVBert (An et al., 2023)	15.87	3.13	73	62
GridMM (Wang et al., 2023b)	14.43	3.35	73	62
BSG (Liu et al., 2023)	14.86	3.19	73	62
Lily (Lin et al., 2023)	16.13	3.44	72	60
ENP (Liu et al., 2024a)	14.27	3.39	71	60
Ours	13.59	**3.08**	**74**	**65**

REVERIE results. Table 8 reports results on REVERIE (Qi et al., 2020), whose concise, object-oriented instructions provide little step-by-step guidance, so the agent must infer the intermediate route on its own.

As shown in Table 8, our model outperforms the DUET baseline on all five metrics of the Val Unseen split. It improves SR by 4.40 and SPL by 2.98 points, and the grounding-aware RGS and RGSPL by 2.39 and 1.67 points, respectively. These gains show that our method remains effective when instructions are sparse and the agent must reason over long horizons, improving both navigation and object grounding. We will add this experiment and discussion to the revised paper.

Results with Other Backbone. AnchorWorld is a **modular enhancement** that can be integrated with any baseline method that encodes node-level latent states from historical observations, since the PAE and ALP modules operate purely on such latent states and are attached only at the fine-tuning stage. In the main text, we mainly adopt DUET as the baseline. In accordance with the reviewers' suggestions, here we explore an alternative backbone, HAMT (Chen et al., 2021b), which encodes the trajectory as a long-horizon history sequence with a hierarchical vision-language transformer. As shown in Table 9, incorporating the prior-anchor encoder (PAE) and the anchor-guided latent prediction (ALP) into HAMT leads to notable improvements on most evaluation metrics, demonstrating the generalizability of the proposed method to some extent.

Computational Cost and Navigation Performance. Table 10 examines the trade-off between computational cost and navigation performance underlying our motivation for compact, candidate-

Table 9: The results on R2R with HAMT as backbone.

Split	Method	TL	NE↓	SR↑	SPL↑
Val Seen	HAMT (baseline)	11.15	2.51	76.00	72.00
	HAMT + AnchorWorld	10.89	**2.48**	**77.86**	**74.35**
Val Unseen	HAMT (baseline)	11.46	3.62	66.00	61.00
	HAMT + AnchorWorld	11.36	**3.53**	**66.62**	**61.28**

Table 10: Computational cost and SPL on the R2R Test Unseen split.

Method	Params (M)	Memory	Time per inference (ms)	SPL↑
DUET (Chen et al., 2022)	181	690 MB	0.18	59.0
UNeMo (Huang et al., 2026)	216	826 MB	0.22	60.1
DreamWalker (Wang et al., 2023a)	325	1.12 GB	0.89	62.8
AnchorWorld	211	807 MB	0.23	**65.0**

conditioned foresight. Relative to DUET, AnchorWorld improves SPL by 6.0 percentage points with 30M additional parameters, 117 MB more memory, and 0.05 ms higher inference latency.

Compared with UNeMo's prediction-based action refinement, AnchorWorld generates an anchor-guided latent rollout for each candidate and improves SPL by 4.9 percentage points at comparable latency (0.23 vs. 0.22 ms), while using fewer parameters and less memory. Compared with DreamWalker's search-based trajectory planning, AnchorWorld improves SPL by 2.2 percentage points while reducing latency from 0.89 to 0.23 ms and the parameter count from 325M to 211M. These results are consistent with the motivation in the introduction: sparse route anchors guide compact, candidate-specific rollouts, yielding higher SPL with latency close to the refinement baseline and lower computational cost than the search-based planner. This supports the use of route-level priors for effective candidate evaluation without explicit tree search.

A.5 Additional visualization examples

More visual comparison results are presented in Figure 6.

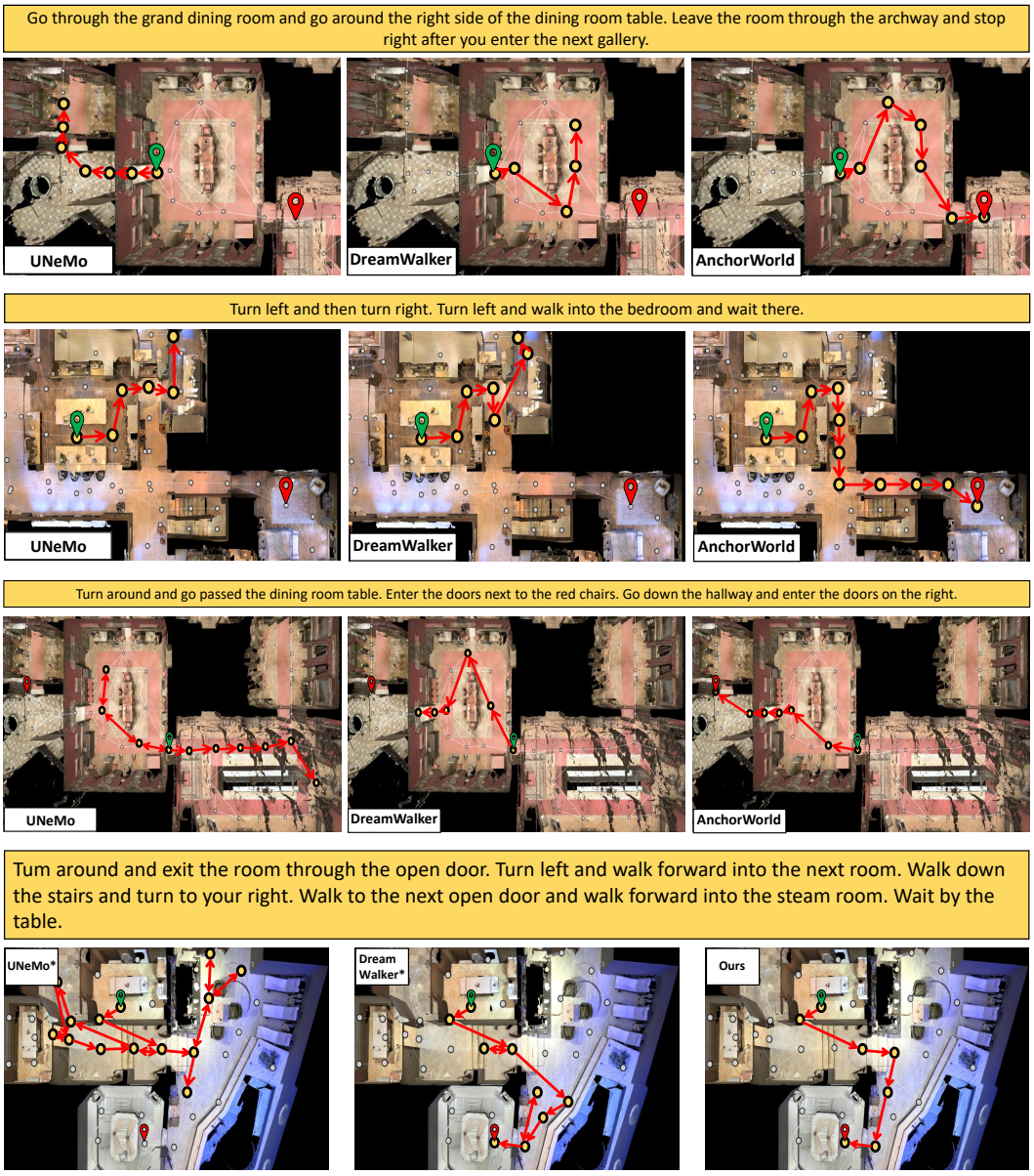


Figure 6: Additional visualization examples for short-horizon and long-horizon comparison on R2R Val Unseen split. Green and red markers denote the start and goal locations, respectively.