This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

NavMorph: A Self-Evolving World Model for Vision-and-Language Navigation in Continuous Environments

Xuan Yao[1,2], Junyu Gao[1,2*], and Changsheng Xu[1,2,3]
[1]State Key Laboratory of Multimodal Artificial Intelligence Systems (MAIS),
Institute of Automation, Chinese Academy of Sciences (CASIA)
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences (UCAS)
[3]Peng Cheng Laboratory, ShenZhen, China
yaoxuan2022@ia.ac.cn; {junyu.gao, csxu}@nlpr.ia.ac.cn

Abstract

Vision-and-Language Navigation in Continuous Environments (VLN-CE) requires agents to execute sequential navigation actions in complex environments guided by natural language instructions. Current approaches often struggle with generalizing to novel environments and adapting to ongoing changes during navigation. Inspired by human cognition, we present NavMorph, a self-evolving world model framework that enhances environmental understanding and decision-making in VLN-CE. NavMorph employs compact latent representations to model environmental dynamics, equipping agents with foresight for adaptive planning and policy refinement. By integrating a novel Contextual Evolution Memory, NavMorph leverages scene-contextual information to support effective navigation while maintaining online adaptability. Extensive experiments demonstrate that NavMorph achieves notable performance improvements on popular VLN-CE benchmarks. Our Code is available at https://github.com/Feliciaxyao/NavMorph.

1. Introduction

In recent years, Embodied AI [41, 43, 57, 64] has emerged as a pivotal research direction, attracting significant attention across computer vision, natural language processing, and robotics communities due to its interdisciplinary nature. Among its core challenges, Vision-and-Language Navigation (VLN) [5] has gained prominence for its broad applicability in many intriguing real-world applications, such as robotic assistance [7, 14, 15, 37], autonomous navigation [31, 45], and smart home systems [17, 54, 55].

VLN tasks require agents to interpret natural language instructions, process visual information in dynamic environments, and execute sequential actions to reach target locations. To navigate effectively, agents must dynamically build a structured understanding of scenes from current observations and instructions, enabling them to predict the outcomes of their actions and make reasonable inferences about subsequent steps in response to ongoing environmental changes. Inspired by remarkable human capacity to construct mental representations of environments, facilitating deliberate planning for complex tasks like motor control, imagery, inference, and decision-making [23], world models have emerged as promising frameworks that explicitly model environmental dynamics by simulating actions and their effects on world states. By maintaining a compact latent representation, world models encapsulate dynamic and structural characteristics through agent-environment interactions [38], enabling prediction of state transitions and potential future observations. Despite a few pioneering efforts, such as Navigation World Models [9], DreamWalker [56] and PathDreamer [32], have demonstrated the potential of world models for navigation, their application in VLN remains underexplored. Specifically, these approaches either lack the capacity to learn underlying action-state transitions or rely on discrete state dynamics, limiting their ability to capture the continuous nature of spatial-temporal dynamics inherent in VLN tasks. The limitation becomes even evident during online deployment, where agents frequently encounter complex and varied environments with substantial distributional shifts from training data [15, 46, 63]. Such environmental disparities pose risks of performance degradation [16], while existing world model approaches [9, 32, 56]

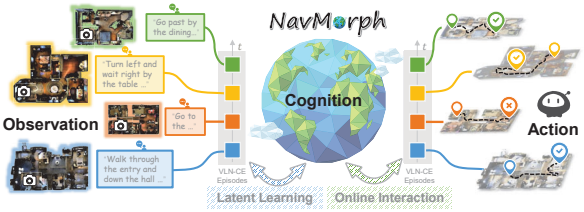


Figure 1. Self-Evolving World Model for Online VLN-CE tasks.

*Corresponding author

focus primarily on the pre-training stage and lack mechanisms for continuous updating, therefore limiting their ability for online adaptation to novel environments.

To address the above issue, we propose NavMorph, a self-evolving world model framework tailored for VLN-CE tasks. As shown in Figure 1, NavMorph constructs a continuously evolving latent space that transforms vision-language inputs (observations) into effective navigation decisions (actions) through online interaction. We design a tailored Recurrent State-Space Model (RSSM) [20] for VLN, which can capture continuous navigational dynamics by explicitly modeling latent action-state transitions. Extending this foundation, NavMorph incorporates self-evolution by progressively enhancing its latent representations, enabling agents to accumulate contextual information. Specifically, the framework comprises two core networks: World-aware Navigator and Foresight Action Planner. The World-aware Navigator infers environmental dynamics from historical context and current observations, constructing robust latent state representations. The Foresight Action Planner then leverages these latent states to learn a navigation policy that outputs control signals, while decoding them into foresight visual embedding, thus providing enriched scene-aware visual information for navigation decisions. To further equip the model with evolving adaptability, we introduce Contextual Evolution Memory (CEM), a simple yet effective memory mechanism seamlessly integrated into two networks of our model. CEM serves as a central enhancement to the recurrent module within RSSM, bolstering foresight planning and plausible navigational inference. Moreover, CEM enables efficient and dynamic updates during learning and online testing, empowering NavMorph with self-evolving abilities.

Our contributions can be summarized as follows:

- Starting from the definition, a novel self-evolving world model is developed for VLN-CE tasks, which adaptively models latent representations of continuous environments, enabling effective foresight-driven decision-making and adaptability in online navigation.
- We design World-aware Navigator to infer latent representations of environmental dynamics and Foresight Action Planner to optimize navigational policy via predictive modeling, integrating seamlessly with Contextual Evolution Memory to accumulate navigation insights for plausible and dynamic action planning in evolving environmental observations.
- Extensive experiments on popular VLN-CE benchmarks demonstrate that NavMorph significantly enhances the performance of several leading models, validating its improved adaptability and generalization.

2. Related Work

VLN in Continuous Environments. Recently, Vision-and-Language Navigation (VLN) tasks [1, 5, 8, 24, 40, 49, 50] have garnered considerable attention among embodied AI, requiring an agent to follow human language instructions to navigate in previously unseen environments using visual observations. To remove the unrealistic assumptions in discrete VLN settings, VLN in continuous environments (VLN-CE) [34] is proposed to extend navigation to continuous spaces. In VLN-CE, agents can freely explore any unobstructed location in 3D environments, using low-level actions (*e.g.,* MOVE-FORWARD 0.25m, TURN-LEFT/RIGHT 15° and STOP) through the Habitat Simulator [53]. Early VLN-CE methods employed end-to-end trained systems to directly predict low-level actions from nature language instructions and visual observations [29, 34]. To alleviate the reliance on costly, data-intensive pre-training, Krantz *et al*. [35] proposed to decouple the task into subgoal planning and low-level control by waypoint predictor. Building on this framework, recent studies have explored waypoint-based approaches by using more robust visual representations pre-trained on large datasets [30], incorporating graph-based models [1], or integrating auxiliary tasks [26] to improve performance. Currently, many of these models rely on panoramic RGB-D inputs [3, 6, 25, 33], but their high cost, bulkiness, and computational overhead hinder deployment on physical robots. Therefore, the monocular configuration—processing single RGB-D frames sequentially—has emerged as a practical alternative, positioning it as a favorable approach for real-world applications in state-of-the-art methods [60, 65].

World Models. World models have emerged as a promising framework for data-efficient learning in simulated environments [27, 42, 48, 58, 62] and video games [22, 66]. These generative models embed observations into latent states, predict future states conditioned on actions, and decode latent predictions back into observation space [38]. As an effective framework for latent dynamics learning, RSSM consists of an Inference Network for estimating latent states and a Predictive Network for forecasting future transitions. Ha *et al*. [19] introduced a prototypical two-part model combining RSSM with a variational autoencoder. Subsequent works like PlaNet [20] and the Dreamer series [21, 22, 44] refined this approach, optimizing policies over predicted latent states. World models enable agents to model environmental dynamics and predict future states without decoding observations, which is crucial for reasoning and planning in VLN tasks. However, their application in VLN remains underexplored. Pathdreamer [32] introduced a visual world model to generate future observations at discrete viewpoints based on prior seen views, using them as augmented data for training. In continuous settings, DREAMWALKER [56] adopted a heuristic search-based world model to generate panoramic images of navigable candidates. Recently, Navigation World Models [9] introduced a controllable video generation model that utilizes

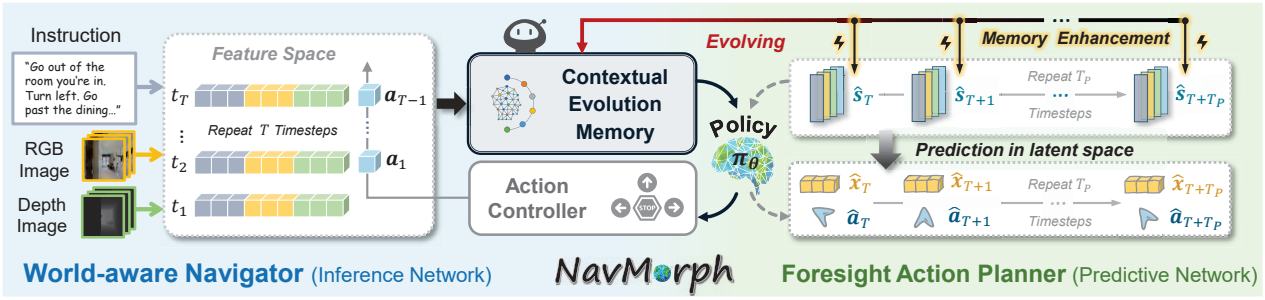


Figure 2. Overall framework of the proposed NavMorph, a self-evolving world model for VLN-CE tasks. World-aware Navigator (Inference Network) infers latent representations by leveraging current observations $\boldsymbol{o}$ and historical context from Contextual Evolution Memory. Foresight Action Planner (Predictive Network) then uses these latent representations $\hat{\boldsymbol{s}}$ to predict future visual embeddings $\hat{\boldsymbol{x}}$, enriching scene semantics for action sequence prediction $\hat{\boldsymbol{a}}$. This integrated framework enables robust navigation and effective policy learning.

learned visual priors to imagine trajectories in unfamiliar environments. Despite notable gains, existing navigational world models encounter challenges in VLN tasks, as they (1) rely on discrete state dynamics, limiting their ability to model continuous action-state transitions, (2) use static pretrained models, restricting adaptability to novel and dynamic environments, and (3) employ pixel-level future prediction that may incur high computational costs. To address these limitations, our proposed world model that models continuous action-state transitions through latent representation learning, adapts dynamically via a self-evolving mechanism, and reconstructs visual embeddings and perform future prediction at feature level to enhance efficiency.

3. Method

In this section, we propose a self-evolving world model tailored for VLN-CE [34], a practical setting where the agent navigates a 3D mesh environment through low-level, parameterized actions. Our model aims to learn and adapt to the spatio-temporal dynamics of continuous environments within a structured latent space, facilitating effective reasoning and action planning during online testing.

Task Definition. VLN-CE [34] utilizes the Habitat Simulator [53] to render observations of environments derived from the Matterport3D dataset [10], providing the agent with RGB-D sensory inputs. Each episode, referring to a complete navigation task, begins with the VLN agent receiving an instruction alongside an initial visual observation, and concludes when the agent either selects the 'stop' action or reaches the maximum step limit. At each timestep t, the agent predicts waypoints [25, 35] from a predefined set of angles and distances, which are subsequently translated into low-level control actions. This navigation progress unfolds over time steps $t \in \mathbb{N}$, with high-dimensional RGB-D images $\boldsymbol{o}_t \in \mathbb{R}^{N_o \times h \times w \times c}$ serving as input observations to infer navigational actions $\boldsymbol{a}_t = \pi(\boldsymbol{a}_t | \boldsymbol{o}_{1:t}, \boldsymbol{a}_{1:t-1})$ under a learnable policy π. The sequence $\boldsymbol{o}_{1:t}$ denotes all observations up to t, while $\boldsymbol{a}_{1:t-1}$ denotes all navigation actions up to $t-1$.

Framework Overview. To enhance the agent's ability to understand and interact with environments for VLN-CE tasks, as illustrated in Figure 2, our proposed world model consists of two primary components: (1) World-Aware Navigator (Inference Network), which continuously constructs environmental representations and infers state transitions in a latent space, and (2) Foresight Action Planner (Predictive Network), which predicts future states to facilitate effective policy learning for strategic navigation.

Within the World-Aware Navigator (parameterized by ϕ), we adopt a structured latent representation following RSSM [20], decomposed into two components: a deterministic history $\boldsymbol{h}_t$ and a stochastic state $\boldsymbol{s}_t$, maintaining consistent navigation progress while incorporating new observations and their potential variations. A visual encoder e_ϕ processes observations and integrates them with these latent variables $(\boldsymbol{h}_{t-1}, \boldsymbol{s}_{t-1})$ to form a comprehensive state representation $\boldsymbol{h}_t$ for navigation decision-making. The Predictive Network (parameterized by θ) then leverages this latent representation to plan foresight actions sequentially. Specifically, it generates the predicted latent stochastic feature $\hat{\boldsymbol{s}}_t$, which encapsulates potential environmental uncertainties in future time steps. Subsequently, a visual decoder d_θ transforms this feature into a predicted visual embedding $\hat{\boldsymbol{x}}_t$, providing a high-level semantic representation of the anticipated navigational scene to guide the prediction of the next action $\hat{\boldsymbol{a}}_{t+1}$. Notably, a recurrent module $f_{\phi/\theta}$ is utilized as the fundamental component of RSSM for modeling latent states, sharing parameter between both above components to ensure consistent temporal dynamics across state inference and future prediction. For notational clarity, we denote it as f. An overview of detailed architecture is shown in Figure 3, with components of our self-evolving world model outlined below:

- **Inference Network**

$$\text{Visual Representation: } \boldsymbol{x}_t = e_\phi(\boldsymbol{o}_t)$$
$$\text{Initial Latent Deterministic State: } \boldsymbol{h}_1 \sim \delta(\boldsymbol{0})$$
$$\text{Recurrent Model: } \boldsymbol{h}_t = f(\boldsymbol{h}_{t-1}, \boldsymbol{s}_{t-1})$$
$$\text{Dynamic Transition Model: } \boldsymbol{s}_t \sim q_\phi(\boldsymbol{s}_t | \boldsymbol{o}_{1:t}, \boldsymbol{a}_{1:t-1})$$

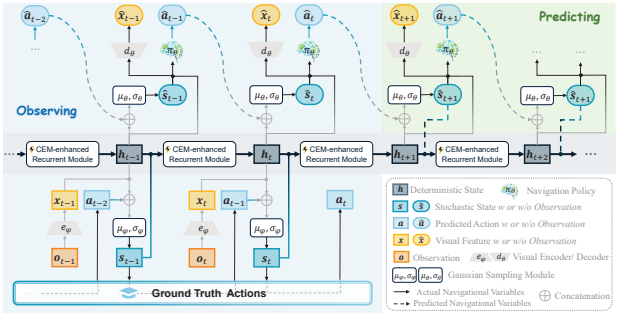


Figure 3. Detailed architecture of our world model.

- **Predictive Network**

$$\text{Initial Latent Stochastic State: } \boldsymbol{s}_1 \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$$
$$\text{Stochastic State Model: } \hat{\boldsymbol{s}}_t \sim p_\theta(\hat{\boldsymbol{s}}_t | \boldsymbol{h}_t, \hat{\boldsymbol{s}}_{t-1})$$
$$\text{Visual Decoder: } \hat{\boldsymbol{x}}_t \sim p_\theta(\hat{\boldsymbol{x}}_t | \boldsymbol{h}_t, \hat{\boldsymbol{s}}_t)$$
$$\text{Action Decoder: } \hat{\boldsymbol{a}}_t \sim p_\theta(\hat{\boldsymbol{a}}_t | \boldsymbol{h}_t, \hat{\boldsymbol{s}}_t)$$

3.1. World-Aware Navigator (Inference Network)

The World-Aware Navigator is a policy-based agent designed to effectively map the cross-modal input onto a sequence of actions within the VLN framework. Actions are pivotal for the world model to anticipate and predict future states. In a typical VLN-CE setting [3, 32], the agent predicts actions by selecting a candidate sub-goal point on a constructed map, along with its corresponding coordinates. To ensure plausible predictions of NavMorph, we define the action at time step t as $\Delta position_t$, which represents the movement of ego location between the previous time step $t-1$ to current time step t. For consistency with operations in the latent space of our world model, a linear transformation maps the action into a d_a-dimensional embedding $\boldsymbol{a}_t \in \mathbb{R}^{d_a}$. After integrating the encoded action with the agent's current state and observation, we sample from $(\boldsymbol{h}_t, \boldsymbol{a}_t, \boldsymbol{x}_t)$ to estimate the Gaussian posterior, yielding a probabilistic representation of the current state:

$$q(\boldsymbol{s}_t | \boldsymbol{o}_{1:t}, \boldsymbol{a}_{1:t-1}) \sim \mathcal{N}(\mu_\phi(\boldsymbol{h}_t, \boldsymbol{a}_{t-1}, \boldsymbol{x}_t), \sigma_\phi(\boldsymbol{h}_t, \boldsymbol{a}_{t-1}, \boldsymbol{x}_t)\boldsymbol{I}), \quad (1)$$

where (μ_ϕ, σ_ϕ) are multi-layer perceptions (MLPs) that parameterize the posterior state distribution. During training, $\boldsymbol{a}_t$ corresponds to ground truth actions.

In our model, the deterministic history should encode both temporal dynamics and accumulated information from past observations and language instructions. A common approach to achieve this is through conventional recurrent modules like RNNs, LSTMs, or transformer-based architectures, which can process current observations alongside previously obtained features to recursively integrate historical information $\boldsymbol{h}_{t-1}$ and stochastic states $\boldsymbol{s}_{t-1}$ into a full history representation $\boldsymbol{h}_t$. However, while effective in capturing short-term dynamics, these approaches struggle with adaptive updates during online testing due to their reliance on gradient descent and lack an explicit memory mechanism to store and utilize novel scene information. Therefore, we introduce a simple yet effective Contextual Evolution Memory (CEM), which enhances the agent's capacity to retain and leverage historical navigation insights. CEM maintains N_m scene-contextual features $\{\boldsymbol{v}_m\}_{m=1}^{N_m}$, where each $\boldsymbol{v}_m \in \mathbb{R}^{d_v}$ captures crucial visual insights from diverse scenarios encountered. To accommodate the dynamic nature of navigation environments, CEM continuously evolves by incorporating new scene features while selectively preserving valuable historical context. CEM is randomly initialized and progressively integrates informative scene-contextual features accumulated throughout learning and online testing. This adaptive mechanism ensures relevance to recent experiences while preserving critical visual-semantic information from past episodes.

Specifically, this evolving CEM supports the enhanced deterministic history $\tilde{\boldsymbol{h}}_t$ by assimilating relevant contextual information from past navigational experiences. Our CEM-enhanced recurrent module retrieves the top-K scene-contextual features from CEM, using them as historical semantic insights to enrich the deterministic state:

$$\{\boldsymbol{v}_k\}_{k=1}^{K} = \arg\text{top}K\left(\mathcal{SIM}(\boldsymbol{h}_t, \boldsymbol{v}_m)\right),$$
$$\tilde{\boldsymbol{h}}_t = (1-\alpha)\boldsymbol{h}_t + \alpha \sum_{k=1}^{K} \mu_k \boldsymbol{v}_k, \quad (2)$$

where the relevance score $\mathcal{SIM}(\boldsymbol{h}_t, \boldsymbol{v}_m)$ is calculated using cosine similarity. The weight μ_k denote the normalized similarity score assigned to the k-th relevant retrieved features, while α serves as an enhancing factor.

To balance the trade-off between previously accumulated experiences and newly observed information, CEM is efficiently evolved by refining the retrieved scene-contextual features through forward updates, rather than using time-consuming gradient-based backpropagation. This evolving process is governed by a factor β and the enriched deterministic state is then used for future predictions:

$$\boldsymbol{v}_k \leftarrow (1-\beta)\boldsymbol{v}_k + \beta \boldsymbol{h}_t, \ \boldsymbol{h}_t \leftarrow \tilde{\boldsymbol{h}}_t. \quad (3)$$

3.2. Foresight Action Planner (Predictive Network)

Empowered by the world model, the agent acquires the ability to anticipate environmental changes resulting from its actions over an extended future period. Based on the state representations learned by the World-Aware Navigator, the Foresight Action Planner is designed to execute future planning by imagining potential scenarios in the latent space. In contrast to future predictions with observed images, we rely solely on the enhanced history states $\boldsymbol{h}_{1:t}$ provided by the CEM-enhanced recurrent module, and the predicted states $\hat{\boldsymbol{s}}_{1:t}$. Similar to inference network, $\hat{\boldsymbol{s}}_t$ is sampled from $(\boldsymbol{h}_t, \hat{\boldsymbol{a}}_t)$ with the Gaussian prior distribution, defined as:

$$p_\theta(\hat{\boldsymbol{s}}_t | \boldsymbol{h}_t, \hat{\boldsymbol{s}}_{t-1}) \sim \mathcal{N}(\mu_\theta(\boldsymbol{h}_t, \hat{\boldsymbol{a}}_{t-1}), \sigma_\theta(\boldsymbol{h}_t, \hat{\boldsymbol{a}}_{t-1})\boldsymbol{I}), \quad (4)$$

where $(\mu_\theta, \sigma_\theta)$ are MLPs that parameterize the prior state distribution. Since the prior does not have access to the ground truth actions, the predicted action $\hat{\boldsymbol{a}}_{t-1}$ is estimated with the learned policy $\pi_\theta(\boldsymbol{h}_{t-1}, \hat{\boldsymbol{s}}_{t-1})$. Building upon this, the Foresight Action Planner extends the prediction of future states over T_p time steps without access to corresponding observations, leveraging the learned latent state and previously inferred actions over previous T timesteps (*i.e.*, observation window). In specific, the planner applies the learned policy to estimate actions $\hat{\boldsymbol{a}}_{T+j} = \pi_\theta(\boldsymbol{h}_{T+j}, \hat{\boldsymbol{s}}_{T+j})$, predicts the next deterministic state $\boldsymbol{h}_{T+j+1} = f_\theta(\boldsymbol{h}_{T+j}, \hat{\boldsymbol{s}}_{T+j})$ and samples from the prior distribution: $\hat{\boldsymbol{s}}_{T+j+1} \sim \mathcal{N}(\mu_\theta(\boldsymbol{h}_{T+j+1}, \hat{\boldsymbol{a}}_{T+j}), \sigma_\theta(\boldsymbol{h}_{T+j+1}, \hat{\boldsymbol{a}}_{T+j})\boldsymbol{I})$, $j \in [1, T_p]$.

Aiming for a comprehensive understanding of environmental dynamics to support action planning, we introduce a reconstruction task that guides foresight action planner to simulate potential future scenarios internally while learning useful latent representations. To achieve this, we reconstruct visual embeddings $\boldsymbol{x}_t \in \mathbb{R}^{d_x}$ derived from input observations $\boldsymbol{o}_t$ through a visual decoder d_θ, rather than direct pixel-level image reconstruction. While pixel-level generation methods [9, 32, 39, 56] are common for future observation prediction, they incur high computational costs and training complexity, while their reliance on pretrained generative models may introduce biases and hinder fair comparisons. This reconstruction objective serves to regularize and enhance the model's future state predictions, with details provided in § 3.3. Additionally, the designed CEM enriches latent representations with contextually relevant historical information, leading to more informed reconstructions and improved planning.

Discussion. The above analysis highlights the crucial role of CEM in our proposed world model, serving as a core component within RSSM to facilitate plausible foresight planning and enable efficient self-evolution. Given the limited exploration of dynamic evolution within existing world models, this paper aims to elucidate the potential of integrating evolutionary strategies into VLN world models. To this end, instead of designing a more complex evolution mechanism, we design a simple, direct, and effective approach that dynamically integrates scene contextual information during learning and online testing. Refer to Sec 4.3 and **Supplementary Material** for further discussion.

3.3. Pre-training Objective

Our evolving world model is trained by maximizing a variational lower bound on the data log-likelihood. This training process involves minimizing losses related to visual reconstruction and action prediction for both past and future states, as well as the divergence between the posterior and prior state distributions. Specifically, the model observes inputs over T timesteps and subsequently forecasts visual embeddings and actions for an additional T_p timesteps. The loss function $\mathcal{L}_W$ is defined as follows:

$$\begin{aligned}\mathcal{L}_W = &\sum_{t=1}^{T} \mathbb{E}\left[\underbrace{-\log p(\boldsymbol{x}_t|\boldsymbol{h}_t, \boldsymbol{s}_t)}_{\text{Reconstruction Loss } \ell_{re}} \underbrace{-\log p(\boldsymbol{a}_t|\boldsymbol{h}_t, \boldsymbol{s}_t)}_{\ell_{ac}}\right] \\ &+ \sum_{j=1}^{T_p} \mathbb{E}\left[\underbrace{-\log p(\boldsymbol{x}_{T+j}|\boldsymbol{h}_T, \boldsymbol{s}_T)}_{\ell_{re}} \underbrace{-\log p(\boldsymbol{a}_{T+j}|\boldsymbol{h}_T, \boldsymbol{s}_T)}_{\text{Action Prediction Loss } \ell_{ac}}\right] \\ &+ \sum_{t=1}^{T} \mathbb{E}\left[\gamma \cdot \underbrace{D_{\text{KL}}\left(q(\boldsymbol{s}_t|\boldsymbol{o}_{1:t}, \boldsymbol{a}_{1:t-1}) \,\|\, p(\boldsymbol{s}_t|\boldsymbol{h}_{t-1}, \boldsymbol{s}_{t-1})\right)}_{\text{Posterior and Prior Matching Loss } \ell_{kl}}\right],\end{aligned} \tag{5}$$

where γ is a weighting coefficient. For simplicity of notation, we use a unified notation $(\boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{x}_t)$ to represent variables across both the observation and prediction phases, omitting the hat notation $(\hat{\boldsymbol{s}}_t, \hat{\boldsymbol{a}}_t, \hat{\boldsymbol{x}}_t)$. Please refer to **Supplementary Material** for the detailed derivation of $\mathcal{L}_W$.

In practice, we implement the log-likelihood term with two components: a visual embedding reconstruction loss ℓ_{re} and an action prediction loss ℓ_{ac}, optimized using stochastic gradient ascent. While point-wise difference metrics (*e.g.*, cross-entropy) provide a direct measure of prediction accuracy, they may not adequately capture the temporal nature of navigation trajectories and potential misalignments between predicted and actual routes. To mitigate this, we incorporate a regularization term based on Normalized Dynamic Time Warping (NDTW) [28], which constrains the predicted visual embeddings and actions. This regularization is specifically formulated as follows:

$$\ell_\star = 1 - \frac{1}{T_p} \sum_{t=1}^{T} \sum_{j=1}^{T_p} \text{NDTW}(\star_{1:t+j}, \hat{\star}_{1:t+j}), \tag{6}$$

where $\star_{1:t+j}$ and $\hat{\star}_{1:t+j}$ denote the ground-truth and predicted sequences, respectively. This NDTW-based regularization is applied to both ℓ_{re} and ℓ_{ac}, encouraging temporal consistency and alignment between predicted and actual navigation trajectories. Additionally, to ensure precise action predictions, we incorporate an L2 loss within ℓ_{ac}, focusing on accurate estimation of $\boldsymbol{a}_t = \Delta position_t$ and leveraging supervision from ground truth actions.

In addition to ensuring foresight latent states $\hat{\boldsymbol{s}}_t$ align with observed data, a Kullback-Leibler (KL) divergence-based loss is employed to align the prior distribution with the posterior. Since both the approximate $q(\boldsymbol{s}_t|\boldsymbol{o}_{1:t}, \boldsymbol{a}_{1:t-1})$ and the prior $p(\boldsymbol{s}_t|\boldsymbol{h}_{t-1}, \boldsymbol{s}_{t-1})$ are modeled as Gaussian distributions (Eq. (1) and Eq. (4)), the posterior and prior matching loss can be computed in closed form. Upon convergence, the KL divergence quantifies the information disparity between the prior and posterior distributions, ensuring the prior models state-action dynamics observed in

actual navigation. Finally, the world model loss $\mathcal{L}_W$ is combined with the imitation learning objective $\mathcal{L}_{IL}$ [3], to form the complete loss function: $\mathcal{L} = \mathcal{L}_W + \mathcal{L}_{IL}$. Note that $\mathcal{L}_{IL} = -\sum_{t=1}^{T} \log p(\boldsymbol{a}^*|\mathcal{I}, \mathcal{O})$, and $\boldsymbol{a}^*$ represents the teacher action node of each step, determined by an interactive demonstrator similar to DAgger algorithm [52].

3.4. Working Modes

Our world model operates in two distinct modes during the training and testing phases.

During training, the model focuses on learning the underlying latent dynamics of environments through future state prediction. It processes sequences of observations and corresponding actions, optimizing loss functions delineated in § 3.3 to update the model via gradient descent.

During testing, our self-evolving world model continually adapts to changing environments via the evolving mechanism. The recurrent module updates CEM in an online manner during testing, similar to the training phase as described in Eq. (3). This adaptive ability allows the model to retain insightful information from recent observations while adjusting to testing scenes, thereby maintaining a contextually enriched and up-to-date comprehension of its surroundings.

4. Experiments

We evaluate our model on R2R-CE and RxR-CE [34] datasets, which transform discrete paths of R2R [5] and RxR [36] datasets into continuous environments. Experiments and ablation studies show our effectiveness.

4.1. Experimental Setup

Datasets. R2R-CE dataset [5, 34] comprises 5,611 shortest-path trajectories, divided across training, validation, and test splits. Each trajectory is paired with approximately three English instructions, with an average path length of 9.89 meters and a mean instruction length of 32 words. Based on similar scene splits, RxR-CE dataset [34, 36] is larger and more challenging, offering substantially more instructions in multiple languages, including English, Hindi, and Telugu, with an average of 120 words per instruction. Notably, agents in R2R-CE have a chassis radius of 0.10 meters and can slide along obstacles during navigation, while agents in RxR-CE, with a larger chassis radius of 0.18 meters, are restricted from sliding thus more prone to collisions.

Evaluation Metrics. We follow previous approaches [4, 5, 28] and adopt the standard metrics for evaluating VLN-CE agents, including TL (Trajectory Length), NE (Navigation Error), OSR (SR given Oracle stop policy), SR (Success Rate), SPL (Success weighted by Path Length), NDTW(Normalized Dynamic Time Warping), and SDTW (Success weighted by NDTW).

Implementation Details. For panoramic settings, we follow standard VLN-CE setups [25, 33], where each location is represented by 12 RGB-D images captured at 30° intervals. We evaluate our approach with ETPNav[3] and HNR[61] to verify our efficacy. While monocular cameras offer advantages in real-world deployment, including lower cost, reduced size, and improved energy efficiency, we validate our method in a more practical setting. Specifically, we adopt VLN-3DFF [60], which adapts the original panoramic observation setting for deployment on real-world monocular robots, leveraging the pretrained 3D Feature Fields model [61]. To align with practical applications, we adhere to the online VLN setting established in [16], setting the batch size to 1 during evaluation. All experiments were conducted using PyTorch framework [47] on a single NVIDIA RTX 3090 GPU. Please refer to **Supplementary Material** for detailed hyper-parameter settings.

4.2. Comparison with State-of-the-art VLN Models

R2R-CE. Table 1 presents a comparative analysis of our self-evolving world model against state-of-the-art methods on R2R-CE dataset. Compared to other monocular methods, our NavMorph consistently achieves strong performance across multiple metrics, particularly in unseen environments. Specifically, on the Val Unseen split, our model achieves remarkable gains of over 4% in both SR and SPL. The model's generalization capabilities are further validated on Test Unseen split, where NavMorph surpasses baseline VLN-3DFF by 1.6% in SPL and approximately 2% in SR.

Notably, our model achieves a marked reduction in trajectory length (TL) alongside increased SR under practical monocular settings, indicating more efficient navigation along optimized paths. This can be attributed to model's robust ability to capture and predict latent environmental dynamics, enabling more informed decision-making through memory-enhanced representations. With panoramic inputs, our model maintains comparable trajectory efficiency while further enhancing SR, likely due to the extensive visual context provided by panoramic views, which supplies sufficient information for base models to select well-informed paths. These results underscore the efficacy of our NavMorph in improving navigation in continuous environments.

RxR-CE. Table 2 compares NavMorph with existing methods on the RxR-CE dataset, illustrating its superior performance in both monocular and panoramic setups. In the monocular setting, NavMorph outperforms the previous state-of-the-art method VLN-3DFF by substantial margins of 4.1% in SR and 2.73% in SPL, highlighting its effectiveness in unseen environments. In panoramic setup, NavMorph achieves 58.02% SR and 48.98% SPL, performing competitively with specialized panoramic methods like ETPNav and HNR. The favorable performance across both NDTW (64.77) and SDTW (48.85) metrics further validates our model's capability to follow instructions faithfully while maintaining path fidelity. These results underscore NavMorph's robustness on the more challenging RxR-CE dataset, with its larger instruction set, multilingual support, and stricter navigation constraints.

Table 1. Experimental results on R2R-CE dataset. Results better than base model are shown in blue. Best results for the panoramic and monocular settings are each highlighted in **bold**. * indicates experimental results that we have reproduced in this work.

Camera	Methods	Val Seen					Val Unseen					Test Unseen				
		TL ↓	NE ↓	OSR	SR	SPL	TL ↓	NE ↓	OSR	SR	SPL	TL ↓	NE ↓	OSR	SR	SPL
Monocular	LAW [51] [EMNLP21]	**9.34**	6.35	49	40	37	**8.89**	6.83	44	35	31	**9.67**	7.69	28	38	25
	CM^2 [18] [CVPR22]	12.05	6.10	50.7	42.9	34.8	11.54	7.02	41.5	34.3	27.6	13.90	7.70	39	31	24
	WS-MGMap [12] [NeurIPS22]	10.12	5.65	51.7	46.9	**43.4**	10.00	6.28	47.6	38.9	34.3	12.30	7.11	45	35	28
	NaVid [65] [RSS24]	-	-	-	-	-	-	**5.47**	49.1	37.4	**35.9**	-	-	-	-	-
	ETPNav/p [60] [CoRL24]	-	-	-	-	-	-	6.81	42.4	32.9	23.1	-	-	-	-	-
	VLN-3DFF [60] [CoRL24]	-	-	-	-	-	-	5.95	55.8	44.9	30.4	-	6.24	54.4	43.7	28.9
	VLN-3DFF*	22.90	4.92	62.1	52.7	36.7	26.16	6.05	54.9	43.8	29.4	26.02	6.22	**54.7**	43.8	28.6
	NavMorph	20.03	**4.58**	**62.7**	**55.8**	38.9	22.54	5.75	**56.9**	**47.9**	33.2	24.75	**6.01**	54.5	**45.7**	**30.2**
Panoramic	Seq2Seq [5] [CVPR18]	**9.26**	7.12	46	37	35	**8.64**	7.37	40	32	30	**8.85**	7.91	36	28	25
	Sim2Sim [33] [ECCV22]	11.18	4.67	61	52	44	10.69	6.07	52	43	36	11.43	6.17	52	44	37
	CWP-CMA [25] [CVPR22]	11.47	5.20	61	51	45	10.90	6.20	52	41	36	11.85	6.30	49	38	33
	CWP-BERT [25] [CVPR22]	12.50	5.02	59	50	44	12.23	5.74	53	44	39	13.51	5.89	51	42	36
	DREAMW [56] [ICCV23]	11.60	4.09	59	66	48	11.30	5.53	49	59	44	11.80	5.48	49	57	44
	GridMM [59] [ICCV23]	12.69	4.21	69	59	51	13.36	5.11	61	49	41	13.31	5.64	56	46	39
	FSTTA [16] [ICML24]	12.39	4.25	69	58	50	11.58	5.27	58	48	42	13.17	5.84	55	46	38
	ETPNav [3] [TPAMI24]	11.78	3.95	72	66	59	11.99	4.71	65	57	49	12.87	5.12	63	55	48
	ETPNav*	11.35	3.93	72	66	59	11.40	4.69	64	57	49	12.72	5.10	63	55	48
	NavMorph	11.43	3.86	73	67	60	11.55	4.62	66	59	50	12.88	4.91	64	57	49
	HNR [61] [CVPR24]	11.79	3.67	76	69	61	12.64	4.42	67	61	51	13.03	4.81	67	58	50
	HNR*	11.84	3.73	76	69	61	12.76	4.57	67	61	51	12.92	4.85	67	58	50
	NavMorph	11.76	**3.66**	**78**	**70**	**62**	12.68	**4.37**	**68**	**64**	**53**	12.69	**4.69**	**68**	**60**	**52**

Note: Following established conventions, we report results with different precision formats across camera configurations: integers for panoramic settings and two decimal places for monocular settings.

Table 2. Experimental results on RxR-CE dataset.

Camera	Methods	Val Unseen				
		NE ↓	SR	SPL	NDTW	SDTW
Monocular	LAW [51]	10.87	8.0	8.0	-	-
	CM^2 [18]	8.98	14.4	9.2	-	-
	WS-MGMap [12]	9.83	15.0	12.1	-	-
	NaVid [65]	**8.41**	23.8	32.2	-	-
	A^2-Nav [13]	-	16.8	6.3	-	-
	VLN-3DFF [60]	8.79	25.5	18.1	-	-
	VLN-3DFF*	9.41	26.66	20.11	42.91	20.36
	NavMorph	8.85	**30.76**	**22.84**	**44.19**	**23.30**
Panoramic	LAW-Pano [51]	11.04	10.0	9.0	-	-
	Seq2Seq [5]	12.1	13.93	11.96	30.86	11.01
	CWP-CMA [25]	8.76	26.59	22.16	47.05	-
	CWP-BERT [25]	8.98	27.08	22.65	46.71	-
	AO-Planner [11]	7.06	43.3	30.5	50.1	-
	Reborn [2]	5.98	48.60	42.05	63.35	41.82
	ETPNav [3]	5.64	54.79	44.89	61.90	45.33
	ETPNav*	5.96	54.83	44.62	61.36	44.87
	NavMorph	5.80	56.23	46.39	63.23	46.98
	HNR [61]	**5.51**	56.39	46.73	63.56	47.24
	HNR*	5.75	56.48	46.62	63.43	47.38
	NavMorph	5.70	**58.02**	**48.98**	**64.77**	**48.85**

4.3. Further Remarks

We perform ablation studies and other in-depth analysis of NavMorph on validation unseen set of R2R-CE dataset under the monocular setting, using VLN-3DFF as base model.

Ablation Study of the proposed World Model. The ablation study presented in Table 3 offers critical insights into the efficacy of our proposed world model. The removal of individual loss terms leads to moderate performance declines, underscoring their collective importance in capturing latent dynamics of the world model. The visual embedding reconstruction loss ℓ_{re} and the action prediction loss ℓ_{ac}, both optimized using NDTW, are crucial for navigation success. Removing ℓ_{re} leads to shorter path length (22.54 $\rightarrow$ 20.25) and reduced OSR (56.88% $\rightarrow$ 54.27%), emphasizing the role of visual consistency in maintaining a temporally stable representation of surroundings. Similarly, omitting ℓ_{ac} degrades multiple metrics, as the constraint on predicted action sequence supports effective policy learning. Moreover, the alignment between posterior and prior distributions is vital for maintaining consistency between learned latent dynamics and practical navigation scenarios, ensuring well-calibrated latent representations. Additionally, to better assess contribution of self-evolution, we evaluate NavMorph *w/o* SE, a variant where NavMorph operates with a fixed CEM module, which remains unchanged and does not apply refinements introduced in Eq. 3. Rows 2-3 of Table 3 highlight the impact of CEM in facilitating online adaptation through dynamic navigation insights aggregation.

Notably, the results validate not only the fundamental efficacy of our world model architecture—as evidenced by substantial performance gains even in the absence of self-evolution—but also complementary benefits of CEM-based adaptation, further enhancing navigation effectiveness. This underscores the collaborative nature of self-evolution and a well-structured world model, where self-evolution enhances adaptability while drawing support from the world model's latent representations. Extended experiments on self-evolution are provided in **Supplymentary Material**.

Ablation Study on CEM. In our model, CEM serves as the core module for self-evolution, enabling the agent to retain and refine historical scene information for improved adaptability in navigation tasks. Unlike conventional recurrent architectures (*e.g.*, RNNs, LSTMs) that rely on

Table 3. Ablation Study of the proposed World Model.

Methods	R2R-CE Val Unseen						
	TL ↓	NE ↓	OSR	SR	SPL	NDTW	SDTW
Base model	26.16	6.05	54.92	43.77	29.39	40.94	29.30
NavMorph	22.54	**5.75**	56.88	**47.91**	**33.22**	44.86	**32.73**
w/o ℓ_{re}	**20.25**	5.85	54.27	45.51	32.38	**45.92**	32.21
w/o ℓ_{ac}	25.14	5.96	56.06	44.81	31.22	43.04	30.49
w/o ℓ_{kl}	25.69	6.30	**57.15**	44.10	30.44	41.20	29.91
NavMorph *w/o* SE	23.34	5.92	55.19	45.08	31.19	42.70	30.85

Note: 'NavMorph *w/o* SE' denotes NavMorph does not perform self-evolution. Last three rows denote NavMorph pretrained with $\mathcal{L}$ without specific loss.

Table 4. Ablation Study on Contextual Evolution Memory (CEM).

Methods		R2R-CE Val Unseen							Testing Time*
		TL ↓	NE ↓	OSR	SR	SPL	NDTW	SDTW	
Base Model		26.16	6.05	54.92	43.77	29.39	40.94	29.30	20.53s
NavMorph	*LSTM-based*	25.17	6.11	54.54	43.67	29.81	41.71	29.83	44.56s
	CEM-based	**22.54**	**5.75**	**56.88**	**47.91**	**33.22**	**44.86**	**32.73**	21.22s

* Note: The last column displays the average execution time of the agnet for a single instruction, calculated on the Validation Unseen set of R2R-CE dataset. Best results are shown in bold.

gradient-based backpropagation to update model parameters for adaptation during online testing, CEM employs a forward iterative update mechanism, efficiently integrating new observations without incurring excessive computational costs. To further validate this design choice, we conduct an ablation study under identical evaluation settings, comparing our *CEM-based* self-evolution mechanism with *LSTM-based* alternative. In the *LSTM-based* mechanism, self-evolution is governed by an LSTM network, where optimization is performed via gradient backpropagation [16] during online testing, and latent states are recursively updated through recurrent processing: $\tilde{\boldsymbol{h}}_t = LSTM(\boldsymbol{h}_t, \boldsymbol{o}_t)$.

As shown in Table 4, our NavMorph with the CEM-based mechanism outperforms the LSTM-based approach across all key metrics, demonstrating superior adaptability in unseen environments. Notably, while both methods improve generalization through self-evolution, the LSTM-based approach incurs a $2.1\times$ increase in testing time due to its reliance on gradient-based optimization. These results underscore the efficacy of our forward-update mechanism in balancing generalization and efficiency in VLN-CE tasks.

Size of Contextual Evolution Memory N_m. We conduct comprehensive ablation studies by varying the memory size N_m within our Contextual Evolution Memory (CEM) from 100 to 5000. As shown in Figure 4, the model achieves optimal performance with $N_m = 1000$, where SR peaks at 47.91% (a 4.14% improvement over baseline), OSR reaches 56.88%, and SPL attains 33.22%. Interestingly, we observe that both smaller and larger memory sizes lead to performance degradation. This suggests that insufficient memory capacity constrains the model's ability to store essential contextual information, while excessive memory potentially introduces noise and redundancy that may interfere with effective decision-making. The results empirically validate a memroy size of 1000 within CEM as the optimal balance between contextual retention and adaptability, enhancing the agent's navigational robustness.

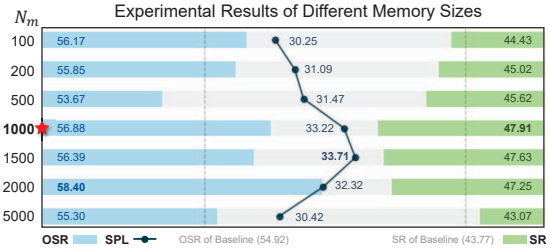


Figure 4. Experimental Results of Different Memory Sizes N_m.

Table 5. Comparison with SOTA Online VLN Method.

Methods	R2R-CE Val Unseen							Time* (s)
	TL ↓	NE ↓	OSR	SR	SPL	NDTW	SDTW	
Base model	26.16	6.05	54.92	43.77	29.39	40.94	29.30	20.53
FSTTA [16]	28.25	6.67	55.41	43.94	29.63	42.76	29.34	27.34
NavMorph	**22.54**	**5.75**	**56.88**	**47.91**	**33.22**	**44.86**	**32.73**	21.22

* Note that the last column displays the average execution time of the VLN agent for a single instruction, calculated on the Validation Unseen set of R2R-CE.

Discussion. Astute readers may notice that our evolving world model accumulates scene-specific information from test environments stored as memory information during online testing. While this memory mechanism operates without access to ground truth actions of testing samples, it may potentially raise concerns about evaluation fairness. Recent research [16] has established the practical significance of online test-time adaptation in VLN tasks, demonstrating its value for real-world deployments where agents must continuously adapt to novel environments. As shown in Table 5, our self-evolving world model, which employs memory-based forward updates, significantly outperforms gradient-based test-time adaptation method, FSTTA [16], in both navigation accuracy and efficiency. In fact, Rows 2-3 of Table 3 also demonstrate the effectiveness of our approach in leveraging accumulated information for efficient adaptation while maintaining robust performance.

5. Conclusion

This paper presents NavMorph, a novel self-evolving world model architecture that addresses the challenges of vision-and-language navigation in continuous environments. At its core, NavMorph integrates World-aware Navigator and Foresight Action Planner to model environmental dynamics in latent space, enabling effective adaptation to diverse online scenarios through cross-episodic learning by the introduced Contextual Evolution Memory mechanism. Extensive experiments demonstrate the effectiveness of our approach, suggesting valuable directions for advancing adaptive navigation tasks in complex dynamic environments.

An open challenge is to design a navigational reward function directly from ground truth, enabling explicit planning within world models to better balance task objectives with environmental constraints, thus enhancing long-term decision-making. Another promising direction lies in developing a modular world model framework with standardized components, enabling flexible composition and evolution of agent capabilities through systematic modular abstractions.

Acknowledgment

This work was supported in part by the National Key Research and Development Plan of China under Grant 2023YFC3310700, in part by the National Natural Science Foundation of China under Grants 62036012, U21B2044, 62236008, 62472422, and U2333215, and in part by Beijing Natural Science Foundation under Grant 4242051.

References

[1] Dong An, Yuankai Qi, Yangguang Li, Yan Huang, Liang Wang, Tieniu Tan, and Jing Shao. Bevbert: Topo-metric map pre-training for language-guided navigation. *arXiv preprint arXiv:2212.04385*, 2022. 2

[2] 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. *arXiv preprint arXiv:2206.11610*, 2022. 7

[3] Dong An, Hanqing Wang, Wenguan Wang, Zun Wang, Yan Huang, Keji He, and Liang Wang. Etpnav: Evolving topological planning for vision-language navigation in continuous environments. *IEEE TPAMI*, 2024. 2, 4, 6, 7

[4] 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*, 2018. 6

[5] 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 *CVPR*, pages 3674–3683, 2018. 1, 2, 6, 7

[6] Peter Anderson, Ayush Shrivastava, Joanne Truong, Arjun Majumdar, Devi Parikh, Dhruv Batra, and Stefan Lee. Sim-to-real transfer for vision-and-language navigation. In *CoRL*, pages 671–681, 2021. 2

[7] Pietro Balatti, Idil Ozdamar, Doganay Sirintuna, Luca Fortini, Mattia Leonori, Juan M Gandarias, and Arash Ajoudani. Robot-assisted navigation for visually impaired through adaptive impedance and path planning. In *ICRA*, pages 2310–2316, 2024. 1

[8] Shurjo Banerjee, Jesse Thomason, and Jason Corso. The robotslang benchmark: Dialog-guided robot localization and navigation. In *CoRL*, pages 1384–1393, 2021. 2

[9] Amir Bar, Gaoyue Zhou, Danny Tran, Trevor Darrell, and Yann LeCun. Navigation world models. *ArXiv*, 2024. 1, 2, 5

[10] 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. 3

[11] Jiaqi Chen, Bingqian Lin, Xinmin Liu, Xiaodan Liang, and Kwan-Yee K Wong. Affordances-oriented planning using foundation models for continuous vision-language navigation. *arXiv preprint arXiv:2407.05890*, 2024. 7

[12] Peihao Chen, Dongyu Ji, Kun-Li Channing Lin, Runhao Zeng, Thomas H. Li, Mingkui Tan, and Chuang Gan. Weakly-supervised multi-granularity map learning for vision-and-language navigation. In *NeurIPS*, pages 38149–38161, 2022. 7

[13] Peihao Chen, Xinyu Sun, Hongyan Zhi, Runhao Zeng, Thomas H Li, Gaowen Liu, Mingkui Tan, and Chuang Gan. A^2 nav: Action-aware zero-shot robot navigation by exploiting vision-and-language ability of foundation models. *arXiv preprint arXiv:2308.07997*, 2023. 7

[14] Junyu Gao, Tianzhu Zhang, and Changsheng Xu. Graph convolutional tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4649–4659, 2019. 1

[15] Junyu Gao, Mengyuan Chen, and Changsheng Xu. Vectorized evidential learning for weakly-supervised temporal action localization. *IEEE transactions on pattern analysis and machine intelligence*, 45(12):15949–15963, 2023. 1

[16] Junyu Gao, Xuan Yao, and Changsheng Xu. Fast-slow test-time adaptation for online vision-and-language navigation. In *ICML*, pages 14902–14919, 2024. 1, 6, 7, 8

[17] Junyu Gao, Mengyuan Chen, and Changsheng Xu. Learning probabilistic presence-absence evidence for weakly-supervised audio-visual event perception. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025. 1

[18] Georgios Georgakis, Karl Schmeckpeper, Karan Wanchoo, Soham Dan, Eleni Miltsakaki, Dan Roth, and Kostas Daniilidis. Cross-modal map learning for vision and language navigation. In *CVPR*, pages 15439–15449, 2022. 7

[19] David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In *NeurIPS*, 2018. 2

[20] Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *ICML*, pages 2555–2565, 2019. 2, 3

[21] Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *ICLR*, 2020. 2

[22] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023. 2

[23] Shibo Hao, Yi Gu, Haodi Ma, Joshua Jiahua Hong, Zhen Wang, Daisy Zhe Wang, and Zhiting Hu. Reasoning with language model is planning with world model. *arXiv preprint arXiv:2305.14992*, 2023. 1

[24] Keji He, Chenyang Si, Zhihe Lu, Yan Huang, Liang Wang, and Xinchao Wang. Frequency-enhanced data augmentation for vision-and-language navigation. In *NeurIPS*, 2024. 2

[25] 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 *CVPR*, pages 15418–15428, 2022. 2, 3, 6, 7

[26] Yicong Hong, Yang Zhou, Ruiyi Zhang, Franck Dernoncourt, Trung Bui, Stephen Gould, and Hao Tan. Learning navigational visual representations with semantic map supervision. In *ICCV*, pages 3055–3067, 2023. 2

[27] Anthony Hu, Gianluca Corrado, Nicolas Griffiths, Zachary Murez, Corina Gurau, Hudson Yeo, Alex Kendall, Roberto Cipolla, and Jamie Shotton. Model-based imitation learning for urban driving. In *NeurIPS*, pages 20703–20716, 2022. 2

[28] Gabriel Ilharco, Vihan Jain, Alexander Ku, Eugene Ie, and Jason Baldridge. General evaluation for instruction conditioned navigation using dynamic time warping. *arXiv preprint arXiv:1907.05446*, 2019. 5, 6

[29] Muhammad Zubair Irshad, Niluthpol Chowdhury Mithun, Zachary Seymour, Han-Pang Chiu, Supun Samarasekera, and Rakesh Kumar. Semantically-aware spatio-temporal reasoning agent for vision-and-language navigation in continuous environments. In *ICPR*, pages 4065–4071, 2022. 2

[30] Aishwarya Kamath, Peter Anderson, Su Wang, Jing Yu Koh, Alexander Ku, Austin Waters, Yinfei Yang, Jason Baldridge, and Zarana Parekh. A new path: Scaling vision-and-language navigation with synthetic instructions and imitation learning. In *CVPR*, pages 10813–10823, 2023. 2

[31] Seita Kayukawa, Daisuke Sato, Masayuki Murata, Tatsuya Ishihara, Hironobu Takagi, Shigeo Morishima, and Chieko Asakawa. Enhancing blind visitor's autonomy in a science museum using an autonomous navigation robot. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pages 1–14, 2023. 1

[32] Jing Yu Koh, Honglak Lee, Yinfei Yang, Jason Baldridge, and Peter Anderson. Pathdreamer: A world model for indoor navigation. In *ICCV*, pages 14738–14748, 2021. 1, 2, 4, 5

[33] Jacob Krantz and Stefan Lee. Sim-2-sim transfer for vision-and-language navigation in continuous environments. In *ECCV*, pages 588–603, 2022. 2, 6, 7

[34] Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *ECCV*, pages 104–120, 2020. 2, 3, 6

[35] Jacob Krantz, Aaron Gokaslan, Dhruv Batra, Stefan Lee, and Oleksandr Maksymets. Waypoint models for instruction-guided navigation in continuous environments. In *ICCV*, pages 15162–15171, 2021. 2, 3

[36] Alexander Ku, Peter Anderson, Roma Patel, Eugene Ie, and Jason Baldridge. Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. In *EMNLP*, pages 4392–4412, 2020. 6

[37] Masaki Kuribayashi, Kohei Uehara, Allan Wang, Daisuke Sato, Simon Chu, and Shigeo Morishima. Memory-maze: Scenario driven benchmark and visual language navigation model for guiding blind people. *arXiv preprint arXiv:2405.07060*, 2024. 1

[38] Yann LeCun. A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. *Open Review*, 62(1):1–62, 2022. 1, 2

[39] Jialu Li and Mohit Bansal. Improving vision-and-language navigation by generating future-view image semantics. In *CVPR*, pages 10803–10812, 2023. 5

[40] Jialu Li and Mohit Bansal. Panogen: Text-conditioned panoramic environment generation for vision-and-language navigation. In *NeurIPS*, 2024. 2

[41] Arjun Majumdar, Anurag Ajay, Xiaohan Zhang, Pranav Putta, Sriram Yenamandra, Mikael Henaff, Sneha Silwal, Paul Mcvay, Oleksandr Maksymets, Sergio Arnaud, et al. Openeqa: Embodied question answering in the era of foundation models. In *CVPR*, pages 16488–16498, 2024. 1

[42] Chen Min, Dawei Zhao, Liang Xiao, Jian Zhao, Xinli Xu, Zheng Zhu, Lei Jin, Jianshu Li, Yulan Guo, Junliang Xing, et al. Driveworld: 4d pre-trained scene understanding via world models for autonomous driving. In *CVPR*, pages 15522–15533, 2024. 2

[43] Yao Mu, Qinglong Zhang, Mengkang Hu, Wenhai Wang, Mingyu Ding, Jun Jin, Bin Wang, Jifeng Dai, Yu Qiao, and Ping Luo. Embodiedgpt: Vision-language pre-training via embodied chain of thought. In *NeurIPS*, 2024. 1

[44] Masashi Okada and Tadahiro Taniguchi. Dreamingv2: Reinforcement learning with discrete world models without reconstruction. In *IROS*, pages 985–991, 2022. 2

[45] Shivam K Panda, Yongkyu Lee, and M Khalid Jawed. Agronav: Autonomous navigation framework for agricultural robots and vehicles using semantic segmentation and semantic line detection. In *CVPR*, pages 6272–6281, 2023. 1

[46] Sang-Min Park and Young-Gab Kim. Visual language navigation: A survey and open challenges. *Artificial Intelligence Review*, 56(1):365–427, 2023. 1

[47] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019. 6

[48] Rudra PK Poudel, Harit Pandya, Stephan Liwicki, and Roberto Cipolla. Recore: Regularized contrastive representation learning of world model. In *CVPR*, pages 22904–22913, 2024. 2

[49] 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 *CVPR*, pages 9982–9991, 2020. 2

[50] 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 TPAMI*, 45(7):8524–8537, 2023. 2

[51] Sonia Raychaudhuri, Saim Wani, Shivansh Patel, Unnat Jain, and Angel X. Chang. Language-aligned waypoint (law) supervision for vision-and-language navigation in continuous environments. In *EMNLP*, pages 4018–4028, 2021. 7

[52] Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *AISTATS*, pages 627–635, 2011. 6

[53] 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 *ICCV*, pages 9339–9347, 2019. 2, 3

[54] Petros Toupas, Georgios Tsamis, Andreas Kargakos, Dimitrios Giakoumis, Konstantinos Votis, and Dimitrios Tzovaras. A framework towards ambient assisted living enhanced by service robots. In *Proceedings of the 16th International Conference on PErvasive Technologies Related to Assistive Environments*, pages 423–428, 2023. 1

[55] Jan Vanus, Radim Hercik, and Petr Bilik. Using interoperability between mobile robot and knx technology for occupancy monitoring in smart home care. *Sensors*, 23(21):8953, 2023. 1

[56] Hanqing Wang, Wei Liang, Luc Van Gool, and Wenguan Wang. Dreamwalker: Mental planning for continuous vision-language navigation. In *ICCV*, pages 10873–10883, 2023. 1, 2, 5, 7

[57] Tai Wang, Xiaohan Mao, Chenming Zhu, Runsen Xu, Ruiyuan Lyu, Peisen Li, Xiao Chen, Wenwei Zhang, Kai Chen, Tianfan Xue, et al. Embodiedscan: A holistic multi-modal 3d perception suite towards embodied ai. In *CVPR*, pages 19757–19767, 2024. 1

[58] Yuqi Wang, Jiawei He, Lue Fan, Hongxin Li, Yuntao Chen, and Zhaoxiang Zhang. Driving into the future: Multiview visual forecasting and planning with world model for autonomous driving. In *CVPR*, pages 14749–14759, 2024. 2

[59] Zihan Wang, Xiangyang Li, Jiahao Yang, Yeqi Liu, and Shuqiang Jiang. Gridmm: Grid memory map for vision-and-language navigation. In *ICCV*, pages 15625–15636, 2023. 7

[60] Zihan Wang, Xiangyang Li, Jiahao Yang, and Shuqiang Jiang. Sim-to-real transfer via 3d feature fields for vision-and-language navigation. In *CoRL*, 2024. 2, 6, 7

[61] 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 *CVPR*, pages 13753–13762, 2024. 6, 7

[62] Philipp Wu, Alejandro Escontrela, Danijar Hafner, Pieter Abbeel, and Ken Goldberg. Daydreamer: World models for physical robot learning. In *CoRL*, pages 2226–2240, 2023. 2

[63] Wansen Wu, Tao Chang, Xinmeng Li, Quanjun Yin, and Yue Hu. Vision-language navigation: a survey and taxonomy. *Neural Computing and Applications*, 36(7):3291–3316, 2024. 1

[64] Yue Yang, Fan-Yun Sun, Luca Weihs, Eli VanderBilt, Alvaro Herrasti, Winson Han, Jiajun Wu, Nick Haber, Ranjay Krishna, Lingjie Liu, et al. Holodeck: Language guided generation of 3d embodied ai environments. In *CVPR*, pages 16227–16237, 2024. 1

[65] Jiazhao Zhang, Kunyu Wang, Rongtao Xu, Gengze Zhou, Yicong Hong, Xiaomeng Fang, Qi Wu, Zhizheng Zhang, and Wang He. Navid: Video-based vlm plans the next step for vision-and-language navigation. In *RSS*, 2024. 2, 7

[66] Weipu Zhang, Gang Wang, Jian Sun, Yetian Yuan, and Gao Huang. Storm: Efficient stochastic transformer based world models for reinforcement learning. In *NeurIPS*, 2024. 2