Let's Reward Step-by-Step: Step-Aware Contrastive Alignment for Vision-Language Navigation in Continuous Environments

Haoyuan Li[1], Rui Liu[1], Hehe Fan[1], and Yi Yang[1†]

College of Computer Science and Technology, Zhejiang University, Hangzhou, China
{haoyuanli,rui.liu,hehefan,yangyics}@zju.edu.cn
https://github.com/lhy-zjut/SACA

Abstract. Vision-Language Navigation in Continuous Environments (VLN-CE) requires agents to learn complex reasoning from long-horizon human interactions. While multi-modal large language models (MLLMs) have driven recent progress, current training paradigms struggle to balance generalization capability, error recovery, and training stability. Specifically, **(i)** policies derived from supervised fine-tuning (SFT) suffer from compounding errors, struggling to recover from out-of-distribution states, and **(ii)** Reinforcement fine-tuning (RFT) methods such as GRPO are bottlenecked by sparse outcome rewards. Their binary feedback fails to assign credit to individual steps, leading to gradient signal collapse in failure-dominant batches. To address these challenges, we introduce Step-Aware Contrastive Alignment (**SACA**), a framework designed to extract dense supervision from imperfect trajectories. At its core, the Perception-Grounded Step-Aware auditor evaluates progress step-by-step, disentangling failed trajectories into valid prefixes and exact divergence points. Leveraging these signals, the Scenario-Conditioned Group Construction mechanism dynamically routes batches to specialized resampling and optimization strategies. Extensive experiments on VLN-CE benchmarks demonstrate that SACA achieves state-of-the-art performance.

Keywords: Vision-Language Navigation · Multi-modal Large Language Models · Reinforcement Fine-Tuning · Embodied Intelligence

1 Introduction

Vision-Language Navigation in Continuous Environments (VLN-CE) requires agents to interpret natural language instructions, process visual streams and execute low-level actions [2, 29]. While Video-based Large Language Models (Video-LLMs) have improved spatial-temporal reasoning and serve as promising foundation models, applying them to continuous navigation is challenging due to trade-offs among generalization, error recovery and training stability [43,58]. The standard adaptation approach for VLN relies on supervised fine-tuning (SFT)

† Corresponding author.

using expert data [12, 58, 70]. Although SFT quickly aligns models' behavior with instructions, policies trained purely by imitation suffer from compounding errors. As shown in Fig. 1, minor deviations push agents into out-of-distribution (OOD) states where policies trained by SFT often fail [43].

Reinforcement fine-tuning (RFT) addresses this by allowing autonomous exploration. Group relative policy optimization (GRPO) [19, 49] is an efficient RFT algorithm that eliminates the need for expensive value functions and critic models. However, standard GRPO struggles in VLN-CE due to sparse navigation rewards. The environment typically provides binary feedback only when the agent executes a `STOP` action. This coarse signal fails at step-level credit assignment, treating immediate failures and near-misses the same. During early exploration, this phenomenon often results in batches where all trajectories fail. In all-failure groups, GRPO's relative advantage vanishes, leading to gradient collapse and wasted computation. While Process Reward Models (PRMs) [13,59,62] can provide dense supervision, training domain-specific PRMs can be costly and prone to reward hacking.

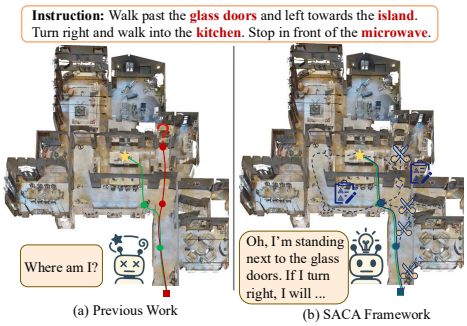

Fig. 1: (a) Previous works discard entire trajectories upon failure due to compounding errors and sparse rewards. **(b)** SACA uses the PGSA auditor to pinpoint the exact Divergence Point ( ). It then uses Repair Resampling ( ) to recover from near-miss trajectories.

We observe that not all failures are equally uninformative. In fact, our experimental data indicates that approximately **73%** of failed episodes successfully execute the initial sub-instructions, yielding valuable Valid Prefixes that standard RL naively discards (details in Appendix A). To extract this dense, step-level supervision from imperfect trajectories without relying on expensive, domain-specific PRMs, we propose Step-Aware Contrastive Alignment (SACA). At its core, SACA employs a Perception-Grounded Step-Aware (PGSA) auditor that evaluates progress step-by-step against landmarks extracted from instructions. Integrating precise spatial grounding with semantic alignment, the PGSA auditor provides a robust continuous Soft Score for trajectory ranking alongside a discrete Structural Mask. This mask explicitly disentangles the trajectory into a Valid Prefix and a Divergence Point, enabling SACA to pinpoint the earliest moment the agent drifted off-course.

Leveraging these granular signals, SACA employs a Scenario-Conditioned Group Construction mechanism to dynamically route batches to specialized optimization strategies. If a sampled group has at least one success (mixed-outcome), outcome rewards drive training: near-miss failures are utilized through Repair Resampling, replanning from the Divergence Point to synthesize corrective demonstrations. Conversely, if all trajectories fail (null-outcome), SACA

triggers All-Failure Rescue. It forms a Reflection Subgroup using the best failure (the Pseudo-Anchor) alongside mined hard negatives, restoring relative supervision via conservative process-aware advantages. Furthermore, fine-grained step-level constraints are applied exclusively to the Pseudo-Anchor: Consistency Alignment reinforces the Valid Prefix through behavior cloning, while Contrastive Correction explicitly penalizes the Divergence Point. These mechanisms effectively convert failed rollouts into trajectory- and step-level supervision.

In summary, the main contributions are three-fold: **(i)** We propose Step-Aware Contrastive Alignment (SACA), a framework that resolves learning-signal collapse in sparse-reward settings by extracting dense supervision from imperfect trajectories. This is achieved through the Perception-Grounded Step-Aware (PGSA) auditor that leverages zero-shot foundation models for precise spatial and semantic tracking, bypassing the need for domain-specific PRMs (§3.1). **(ii)** We introduce a Scenario-Conditioned Group Construction mechanism that dynamically switches between Repair Resampling for mixed-outcome groups and All-Failure Rescue for null-outcome groups (§3.2). We further propose the robust SACA optimization objective that combines trajectory-level advantages with step-level Consistency Alignment and Contrastive Correction (§3.3). **(iii)** Extensive experiments on VLN-CE benchmarks demonstrate that SACA achieves state-of-the-art performance, providing an efficient exploration paradigm for continuous embodied tasks (§4.2).

2 Related Work

Vision-Language Navigation in Continuous Environments (VLN-CE). Vision-Language Navigation (VLN) requires embodied agents to execute natural language instructions [2,42]. While early research focused on discrete topological graphs [10, 14–17, 35–37, 52, 71], recent paradigms have shifted to continuous environments (VLN-CE) [1,8,18,28,46,51,53], demanding fine-grained low-level control. To manage this complexity, many methods rely on simulator-pretrained waypoint predictors [1, 55, 57]. However, these predictors often overfit training layouts, bottlenecking generalization to unseen scenes and highlighting the need for more adaptable, long-horizon navigation architectures.

Navigation with Multi-modal Large Language Models (MLLMs). The rapid evolution of MLLMs has unlocked new avenues for VLN. Some approaches [9, 39, 40, 71] integrate MLLMs as high-level planners to achieve instruction annotation with high linguistic diversity, though they typically underperform task-specific models. Conversely, another line of research [12,64–66] focuses on end-to-end SFT of Video-LLMs [6,11,30,31,33,58,68] to directly parse spatio-temporal dynamics and output low-level control commands.

Reinforcement Fine-Tuning (RFT) in Embodied Intelligence. RFT can enhance the ability of LLMs and MLLMs in reasoning and generalization [19, 23, 24, 34, 49, 50, 63]. To balance alignment with computational efficiency and learning complexity, RFT has rapidly evolved from DPO [45] and PPO [48] to highly scalable, critic-free frameworks, *e.g.,* GRPO [19,49] and GSPO [69]. They

span several paradigms, including offline, online, and test-time RL. While RFT was a common technique in physics-based simulators [25,32], VLA [20,21,61,67] and discrete VLN [5,54], its direct application to VLN-CE is incomplete due to the expanded action space and longer trajectory length. Recently, VLN-R1 [43] pioneered GRPO in VLN-CE. Yet, it remains bottlenecked by sparse outcome supervision. The proposed SACA framework introduces a PGSA auditor to extract dense, step-level signals from imperfect trajectories. Leveraging these signals, the Scenario-Conditioned Group Construction mechanism dynamically reconstructs pseudo-positive samples from failures. By structurally isolating Valid Prefixes and repairing Divergence Points, SACA provides continuous supervision, effectively overcoming the sample inefficiency of sparse rewards.

3 Method

In this section, we present the SACA framework. We first introduce a PGSA auditor, which extracts continuous ranking signals and discrete structural boundaries from uninformative rollouts without requiring a pre-trained reward model (§3.1). Then, we detail the Scenario-Conditioned Group Construction mechanism, dynamically switching between Repair Resampling for mixed-outcome groups and All-Failure Rescue for null-outcome groups (§3.2). Finally, we formulate the robust SACA optimization objective, integrating trajectory-level conservative advantages with step-level corrective constraints (§3.3).

3.1 Perception-Grounded Step-Aware (PGSA) Auditor

To bridge the semantic gap between high-level linguistic instructions I and low-level continuous observations $O = \{o_1, \dots, o_T\}$, the PGSA auditor employs a hierarchical, zero-shot perception pipeline. Given an instruction, we first use a frozen tiny LLM (*e.g.*, Qwen3-0.6B [60]) to parse the instruction into a sequence of intermediate landmarks $L = \{l_1, \dots, l_m\}$. By integrating foundation models (*e.g.*, GroundingDINO [38], SAM3 [3], CLIP [44]), the PGSA auditor synthesizes two complementary supervision signals: a continuous `Soft Score` for fine-grained trajectory ranking, and a discrete `Hard Mask` to structurally isolate the exact Divergence Point where the policy deviates from the optimal path.

Hierarchical Soft Scoring. To formulate a robust evaluation of progress toward a target landmark l_i, we construct a composite Soft Score S_t that progressively fuses global contextual awareness with precise object-level grounding. Specifically, the PGSA auditor first computes a global semantic similarity via CLIP [44]. GroundingDINO [38] acts as a spatial bottleneck, predicting a bounding box and an associated detection confidence c_{det}. To distill pure object-centric semantics and filter out background noise, instances with high detection confidence ($c_{det} \geq \tau_{det}$) are further refined by SAM3 [3], yielding a precise object mask M_{obj} with an IoU score c_{iou}. The integrated reward is formulated as:

$$S_t(o_t, l_i) = w_1 \text{sim}(o_t, l_i) + \mathbb{I}(c_{det} \geq \tau_{det}) \big[w_2 c_{det} + w_3 (c_{iou} \cdot \text{sim}(o_t \odot M_{obj}, l_i)) \big], \quad (1)$$

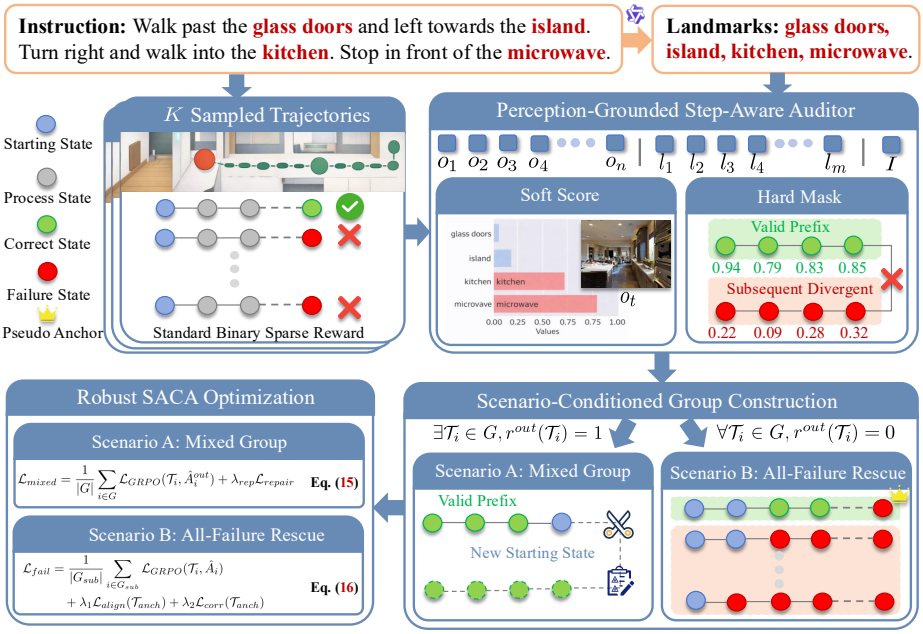

Fig. 2: Overview of proposed SACA framework. The PGSA auditor evaluates K trajectories against instruction landmarks, yielding a Soft Score for ranking and a Hard Mask to isolate the Divergence Point. Based on batch outcomes, a Scenario-Conditioned mechanism dynamically routes to either Repair Resampling (for mixed groups) or All-Failure Rescue (for null-outcome groups), followed by robust optimization.

where $\text{sim}(\cdot,\cdot)$ denotes the cosine similarity between CLIP visual and textual embeddings, and w_1, w_2, w_3 are balancing weights. This progressive formulation ensures that S_t gracefully relies on global context when the target is distant, yet exhibits a sharp, highly informative peak upon precise spatial grounding, providing the agent with dense positive reinforcement signals.

To derive a scale-invariant trajectory-level evaluation and mitigate the length bias inherent in cumulative step-wise rewards, we aggregate the step scores into a normalized process score $R_{proc}(\mathcal{T})$ via a threshold-gated formulation:

$$R_{proc}(\mathcal{T}) = \frac{1}{T}\sum_{t=1}^{T} \text{ReLU}(S_t - \tau_s), \tag{2}$$

where T represents the trajectory length. Here, the ReLU function acts as a noise-filtering gate: it truncates low-confidence step scores below the threshold (τ_s) to zero, ensuring that only valid perception steps strongly aligned with the instruction can accumulate and contribute to the final trajectory score. Please refer to Appendix B for detailed visualizations of the step-level trajectory score.

Structural Hard Masking. Simultaneously, we apply a stricter hard threshold τ_h to generate a step-wise binary mask $M_t = \mathbb{I}(S_t > \tau_h) \in \{0, 1\}$. During navigation, the active landmark l_i advances to the next landmark in L once this hard-match condition is satisfied for c consecutive steps. This mask structurally

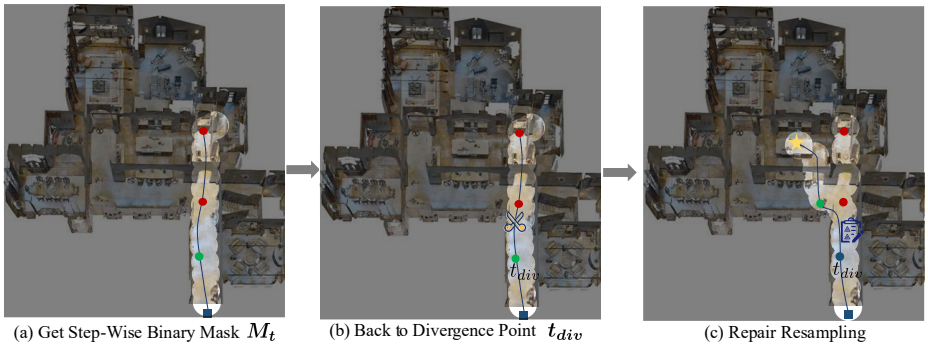

Fig. 3: Illustration of the Repair Resampling process. (a) Extracting the structural mask M_t via the PGSA auditor. **(b)** Backtracking to the Divergence Point t_{div} to prune the erroneous suffix. **(c)** Resampling a corrective path from t_{div}, thereby salvaging the Valid Prefix and providing robust step-level supervision.

disentangles the trajectory by identifying the exact step where the agent deviates from the instruction. We define the Divergence Point t_{div} as:

$$t_{div} = \begin{cases} \min\{t \in \{1, \dots, T\} \mid M_t = 0\}, & \text{if } \exists t \text{ s.t. } M_t = 0, \\ T + 1, & \text{otherwise.} \end{cases} \tag{3}$$

As shown in Fig. 3, the trajectory is partitioned into a Valid Prefix comprising all steps $t < t_{div}$, and a subsequent divergent phase starting at t_{div}. This decoupling enables smooth trajectory-level ranking via R_{proc} while preserving strict discrete boundaries for step-level correction.

3.2 Scenario-Conditioned Group Construction Mechanism

Let $r^{out}(\mathcal{T}) \in \{0, 1\}$ denote the binary outcome reward. For each instruction, the policy model π_θ samples a group of trajectories: $G = \{\mathcal{T}_1, \dots, \mathcal{T}_K\}$, where K denotes the group size.

Scenario A: Mixed Group ($\exists \tau_i \in G, r^{out}(\tau_i) = 1$). When G contains successful trajectories, outcome supervision drives the primary optimization. To further boost sample efficiency, we apply **Repair Resampling** to high-quality "near-miss" failures. We formally define this near-miss subset as trajectories:

$$G_{miss} = \left\{ \mathcal{T}_i \in G \mid r^{out}(\mathcal{T}_i) = 0 \wedge \frac{t_{div}^{(i)}}{T_i} > \eta \right\}, \tag{4}$$

where the valid-prefix ratio exceeds a threshold η. For each $\mathcal{T}_i \in G_{miss}$, we truncate the rollout at the Divergence Point $t_{div}^{(i)}$ and iteratively resample suffixes $\mathcal{T}_{suff}^{(n)} \sim \pi_\theta(\cdot \mid \mathcal{T}_{i,1:t_{div}^{(i)}})$ up to N_{rep} times. We then halt resampling and construct a repaired trajectory via concatenation:

$$\mathcal{T}_i^{rep} = \mathcal{T}_{i,1:t_{div}^{(i)}} \oplus \mathcal{T}_{suff}^{(n)}. \tag{5}$$

These successfully synthesized trajectories form an auxiliary set $\mathcal{T}^{rep}$ to augment the current training update, strictly without recursive resampling. If all N_{rep} attempts fail, the original $\mathcal{T}_i$ is retained.

Scenario B: All-Failure Rescue ($\forall \mathcal{T}_i \in G, r^{out}(\mathcal{T}_i) = 0$). When all K trajectories fail, standard outcome-based GRPO collapses. SACA triggers the All-Failure Rescue by constructing a Reflection Subgroup G_{sub} centered around a Pseudo-Anchor, which isolates the most informative failure and its nearby alternatives for corrective comparison. The Pseudo-Anchor $\mathcal{T}_{anch}$ is used only as the most informative failure, rather than a pseudo-positive target. First, we select the trajectory with the highest process score:

$$\mathcal{T}_{anch} = \underset{\mathcal{T} \in G}{\operatorname{argmax}} R_{proc}(\mathcal{T}). \tag{6}$$

Next, we mine Hard Negatives using a dual-criterion score:

$$S_{neg}(\mathcal{T}) = \lambda \cdot \mathrm{PrefixSim}(\mathcal{T}, \mathcal{T}_{anch}) - (1-\lambda) \cdot (R_{proc}(\mathcal{T}_{anch}) - R_{proc}(\mathcal{T})), \tag{7}$$

where PrefixSim is instantiated by action-level Longest Common Subsequence (LCS), and λ balances prefix similarity and process-score gap. We select the top-m failed trajectories ranked by S_{neg}, excluding $\mathcal{T}_{anch}$, to form G_{neg} and construct $G_{sub} = \{\mathcal{T}_{anch}\} \cup G_{neg}$, with $|G_{sub}| \geq 2$ for stable normalization. Notably, Eq. (8) and Eq. (16) are applied only to G_{sub} in Scenario B, whereas G_{miss} is used only in Scenario A for Repair Resampling.

3.3 Robust SACA Optimization Objective

We use $\mathcal{L}_{GRPO}(\tau_i, \hat{A}_i)$ to denote the standard clipped policy-ratio loss evaluated on trajectory τ_i with advantage $\hat{A}_i$. For all-failure groups, we compute process-aware relative advantages within the Reflection Subgroup G_{sub}. To reduce the risk of wrongful penalization due to noisy visual-process estimates, we introduce a hierarchical conservative estimation mechanism:

$$\hat{A}_i^{raw} = \frac{R_{proc}(\tau_i) - \mu\big(\{R_{proc}(\tau) \mid \tau \in G_{sub}\}\big)}{\sigma\big(\{R_{proc}(\tau) \mid \tau \in G_{sub}\}\big) + \epsilon_0}. \tag{8}$$

Margin-Based Rescue. First, we compute the margin Δ:

$$\Delta = R_{proc}(\mathcal{T}_{anch}) - \mu\big(\{R_{proc}(\mathcal{T}) \mid \mathcal{T} \in G_{neg}\}\big). \tag{9}$$

If the Pseudo-Anchor lacks credibility ($\Delta < \delta$), we shrink the advantages by a temperature factor $\kappa < 1$:

$$\hat{A}'_i = \begin{cases} \kappa \hat{A}_i^{raw}, & \text{if } \Delta < \delta, \\ \hat{A}_i^{raw}, & \text{otherwise.} \end{cases} \tag{10}$$

Negative-Only Scaling. We apply an attenuation factor $s \in (0, 1]$ strictly to the negative advantages:

$$\hat{A}_i = \begin{cases} \hat{A}'_i, & \hat{A}'_i \geq 0, \\ s \cdot \hat{A}'_i, & \hat{A}'_i < 0. \end{cases} \tag{11}$$

This preserves directional preference toward the most informative failure while reducing the risk of over-penalizing plausible alternatives under noisy visual estimates.

Step-Level Constraints. Beyond trajectory-level advantages, we apply fine-grained structural constraints exclusively to the Pseudo-Anchor $\mathcal{T}_{anch}$ to avoid amplifying noisy supervision from low-quality failures. We treat the actions along the Valid Prefix as pseudo-correct decisions and reinforce them via behavior cloning. By definition of t_{div}, all steps $t < t_{div}$ satisfy $M_t = 1$. Contrastive Correction is applied strictly to the Divergence Point t_{div}:

$$\mathcal{L}_{align}(\mathcal{T}_{anch}) = -\sum_{t<t_{div}} \log \pi_\theta(a_t \mid o_t, I), \tag{12}$$

$$\mathcal{L}_{corr}(\mathcal{T}_{anch}) = -\log \frac{\exp(\mathrm{sim}(h_{s_{t_{div}}}, h_{a^+})/\alpha)}{\exp(\mathrm{sim}(h_{s_{t_{div}}}, h_{a^+})/\alpha) + \exp(\mathrm{sim}(h_{s_{t_{div}}}, h_{a^-})/\alpha)}, \tag{13}$$

where $h_{s_{t_{div}}}$ denotes the policy hidden state at the Divergence Point t_{div}. The positive action a^+ is the simulator shortest-path action, while the negative action a^- is the erroneous policy action at t_{div}. Thus, $\mathcal{L}_{corr}$ performs local correction at the detected Divergence Point rather than trajectory-wide imitation.

Summary. SACA dynamically switches objectives based on the group scenario:

- For a Mixed Group, we use the outcome-based advantage $\hat{A}_i^{out}$ and the mixed strategy:

$$\hat{A}_i^{out} = \frac{r^{out}(\mathcal{T}_i) - \mu(\{r^{out}(\mathcal{T}) \mid \mathcal{T} \in G\})}{\sigma(\{r^{out}(\mathcal{T}) \mid \mathcal{T} \in G\}) + \epsilon_0}, \tag{14}$$

$$\mathcal{L}_{mixed} = \frac{1}{|G|}\sum_{i\in G} \mathcal{L}_{GRPO}(\mathcal{T}_i, \hat{A}_i^{out}) - \lambda_{rep}\frac{1}{|\mathcal{T}_{rep}|}\sum_{\mathcal{T}\in\mathcal{T}_{rep}}\sum_{t\in\mathrm{suffix}(\mathcal{T})} \log \pi_\theta(a_t \mid o_t, I), \tag{15}$$

  where λ_{rep} is the repair loss weight, $\mathcal{T}_{rep}$ denotes the set of successfully repaired trajectories, and ϵ_0 is a small constant for numerical stability.
- For an All-Failure Group, we utilize the robust process advantage $\hat{A}_i$ alongside the anchor-specific constraints:

$$\mathcal{L}_{fail} = \frac{1}{|G_{sub}|}\sum_{i\in G_{sub}} \mathcal{L}_{GRPO}(\mathcal{T}_i, \hat{A}_i) + \lambda_1\mathcal{L}_{align}(\mathcal{T}_{anch}) + \lambda_2\mathcal{L}_{corr}(\mathcal{T}_{anch}), \tag{16}$$

  where λ_1, λ_2 are balancing weights.

In each training iteration, SACA first scores sampled trajectories with the PGSA auditor, then routes the group to either Mixed-Group Repair or All-Failure Rescue based on outcome availability, and finally updates the policy with scenario-specific objectives. Every sampled rollout is assigned a usable training role, *e.g.,* successful, repairable, or rescuable. Even failed trajectories contribute structured supervision under severe reward sparsity. Additional step-by-step explanation of the SACA framework is provided in Appendix B.

Method	Observation Encoder				R2R-CE Val-Unseen				RxR-CE Val-Unseen			
	Pano.	Odo.	Depth	S.RGB	NE↓	OS↑	SR↑	SPL↑	NE↓	SR↑	SPL↑	nDTW↑
HPN+DN* [26]	✓	✓	✓		6.31	40.0	36.0	34.0	-	-	-	-
CMA* [22]	✓	✓	✓		6.20	52.0	41.0	36.0	8.76	26.5	22.1	47.0
VLN↻BERT* [22]	✓	✓	✓		5.74	53.0	44.0	39.0	8.98	27.0	22.6	46.7
Sim2Sim* [27]	✓	✓	✓		6.07	52.0	43.0	36.0	-	-	-	-
GridMM* [56]	✓	✓	✓		5.11	61.0	49.0	41.0	-	-	-	-
ETPNav* [1]	✓	✓	✓		4.71	65.0	57.0	49.0	5.64	54.7	44.8	61.9
ScaleVLN* [57]	✓	✓	✓		4.80	–	55.0	51.0	-	-	-	-
InstructNav [39]	-	-	-	-	6.89	–	31.0	24.0	-	-	-	-
AG-CMTP [7]	✓	✓	✓		7.90	39.2	23.1	19.1	-	-	-	-
R2R-CMTP [7]	✓	✓	✓		7.90	38.0	26.4	22.7	-	-	-	-
LAW [46]		✓	✓	✓	6.83	44.0	35.0	31.0	10.90	8.0	8.0	38.0
CM2 [18]		✓	✓	✓	7.02	41.5	34.3	27.6	-	-	-	-
WS-MGMap [8]		✓	✓	✓	6.28	47.6	38.9	34.3	-	-	-	-
ETPNav + FF [55]		✓	✓	✓	5.95	55.8	44.9	30.4	8.79	25.5	18.1	-
Seq2Seq [29]			✓	✓	7.77	37.0	25.0	22.0	12.10	13.9	11.9	30.8
CMA [29]			✓	✓	7.37	40.0	32.0	30.0	-	-	-	-
VLN-R1 [43]				✓	7.00	41.2	30.2	21.8	10.40	22.3	17.5	-
NaVid [65]				✓	5.47	49.1	37.4	35.9	-	-	-	-
MapNav [66]				✓	4.93	53.0	39.7	37.2	-	-	-	-
NaVILA [12]				✓	5.37	57.6	49.7	45.5	-	-	-	-
StreamVLN [58]				✓	5.43	62.5	52.8	47.2	6.72	48.6	42.5	60.2
SACA (Ours)				✓	**4.57**	**64.9**	**60.3**	**55.1**	**4.90**	**60.3**	**49.8**	**62.1**
NaVILA† [12]				✓	5.22	62.5	54.0	49.0	6.77	49.3	44.0	58.8
UniNaVid† [64]				✓	5.58	53.3	47.0	42.7	6.24	48.7	40.9	-
StreamVLN† [58]				✓	4.98	64.2	56.9	51.9	6.22	52.9	46.0	61.9
SACA (Ours)†				✓	**4.19**	**69.3**	**64.7**	**56.9**	**4.75**	**62.1**	**51.7**	**66.0**

Table 1: Comparison with state-of-the-art methods. ∗ indicates methods using the waypoint predictor from [22]. † denotes methods using additional training data beyond the R2R-CE [2] and RxR-CE [29] benchmarks.

4 Experiment

In this section, we comprehensively evaluate the SACA framework. We first outline the experimental setup, detailing the VLN-CE benchmarks, evaluation metrics, and implementation details (§4.1). Building upon this foundation, we compare SACA with existing state-of-the-art methods, demonstrating its superior generalization and robust error-recovery capabilities in both standard and complex long-horizon navigation tasks (§4.2). Finally, we present extensive ablation studies to validate the individual contributions of the PGSA perception modules, the Scenario-Conditioned Group Construction mechanism, and the robust SACA optimization objective (§4.3).

4.1 Experimental Setup

Datasets and Evaluation. We evaluate the SACA framework on two prominent Vision-Language Navigation in Continuous Environments (VLN-CE) benchmarks: R2R-CE [2] and RxR-CE [29]. Both datasets are constructed over the

Fig. 4: Qualitative comparison of navigation trajectories. (a) Egocentric observations. **(b)** Top-down maps. Green and red frames denote correct trajectory and failure trajectory, respectively.

Matterport3D (MP3D) [4] scenes and simulated within the Habitat platform [41, 47]. Following the standard evaluation protocol established by previous VLN-CE works, we assess navigation performance using a comprehensive suite of metrics. These include Trajectory Length (TL), Navigation Error (NE), Oracle Success Rate (OS), Success Rate (SR), and Success weighted by Path Length (SPL). For RxR-CE, we additionally report normalized Dynamic Time Warping (nDTW) to measure the spatial fidelity between the predicted and ground-truth paths. Among these metrics, SR and SPL on the unseen splits remain the most critical indicators of an agent's generalization capability.

Implementation Details. We initialize the policy network from a pre-trained Video-LLM, LLaVA-Video-7B [68], and optimize it in two stages. In the first stage, we perform SFT on navigation demonstrations to align the visual-language representation with low-level navigation actions. SFT is trained on 8 NVIDIA A6000 GPUs with a learning rate of 1×10^{-5}, a cosine learning-rate schedule, and 10% warmup, taking approximately 36 hours. In the second stage, we conduct RFT with the proposed SACA objective. Unless otherwise specified, we use group size $K = 8$, maximum repair attempts $N_{rep} = 3$, and the default PGSA auditor's thresholds reported in Appendix C. During RFT, the learning rate is reduced to 1×10^{-6} with weight decay 0.01, and the KL penalty coefficient is set to $\beta = 0.04$. The PGSA auditor is used only during training to score sampled rollouts and construct step-aware supervision; all its perception modules are frozen off-the-shelf models and introduce no additional inference-time cost. The RFT stage takes about 24 hours. Detailed hyperparameters are provided in Appendix C.

Table 2: Ablation study on the core components of SACA. SS: Soft Score for ranking; **AFR**: All-Failure Rescue (Scenario B); **RR**: Repair Resampling (Scenario A).

Row	Components			R2R-CE Val-Unseen				RxR-CE Val-Unseen			
	SS	**AFR**	**RR**	**NE↓**	**OS↑**	**SR↑**	**SPL↑**	**NE↓**	**SR↑**	**SPL↑**	**nDTW↑**
0	SFT Baseline [58]			5.43	62.5	52.8	47.2	6.72	48.6	42.5	60.2
1				5.39	55.3	54.1	48.8	6.54	49.9	43.0	60.3
2	✓			5.21	63.2	55.4	49.6	6.15	51.8	44.3	60.7
3	✓	✓		4.82	64.1	58.2	52.4	5.48	56.4	47.6	61.3
4	✓	✓	✓	**4.57**	**64.9**	**60.3**	**55.1**	**4.90**	**60.3**	**49.8**	**62.1**

4.2 Comparisons with State-of-the-Art Methods

Table 1 compares SACA with state-of-the-art methods on the val-unseen splits of VLN-CE R2R [2] and RxR [29], with baselines categorized by observation encoders and extra data usage. SACA achieves new state-of-the-art performance across nearly all metrics. In the standard single-RGB setting without extra data, SACA obtains 60.3% SR and 55.1% SPL on R2R-CE, outperforming the previous best StreamVLN [58] by 7.5% SR and 7.9% SPL while reducing NE to 4.57. On the more challenging long-horizon RxR-CE benchmark, SACA reaches 60.3% SR and 49.8% SPL, exceeding prior SOTA by 11.7% SR and 7.3% SPL. With additional ScaleVLN data (†), SACA further improves to 64.7% SR on R2R-CE and 62.1% SR on RxR-CE, surpassing StreamVLN† by 9.2% RxR SR. Qualitative comparisons in Fig. 4 show that, unlike StreamVLN and VLN-R1 [43], SACA better maintains step-level alignment and recovers from local deviations.

4.3 Ablation Study

We conduct extensive ablation studies on the VLN-CE R2R [2] and RxR [29] Val-Unseen splits. The analysis is structured to validate the core macro-architecture (Table 2 and Fig. 5), the specific micro-objective designs (Table 3), and the hyperparameter sensitivity (Table 4).

Training Dynamics and Sample Efficiency. Fig. 5 illustrates SR during the RFT process. Standard GRPO (brown curve) suffers from severe learning-signal collapse and a premature plateau due to its reliance on sparse outcome rewards. Incorporating step-aware dense supervision without the All-Failure Rescue mechanism (SACA w/o AFR, blue curve) accelerates initial bootstrapping but eventually bottlenecks because it still

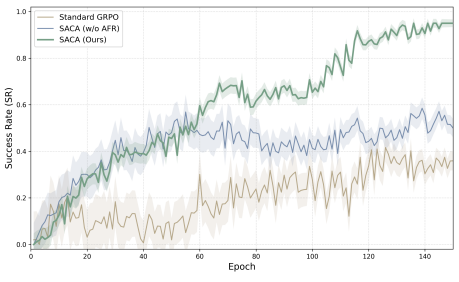

Fig. 5: Illustration of SR curves during RFT training.

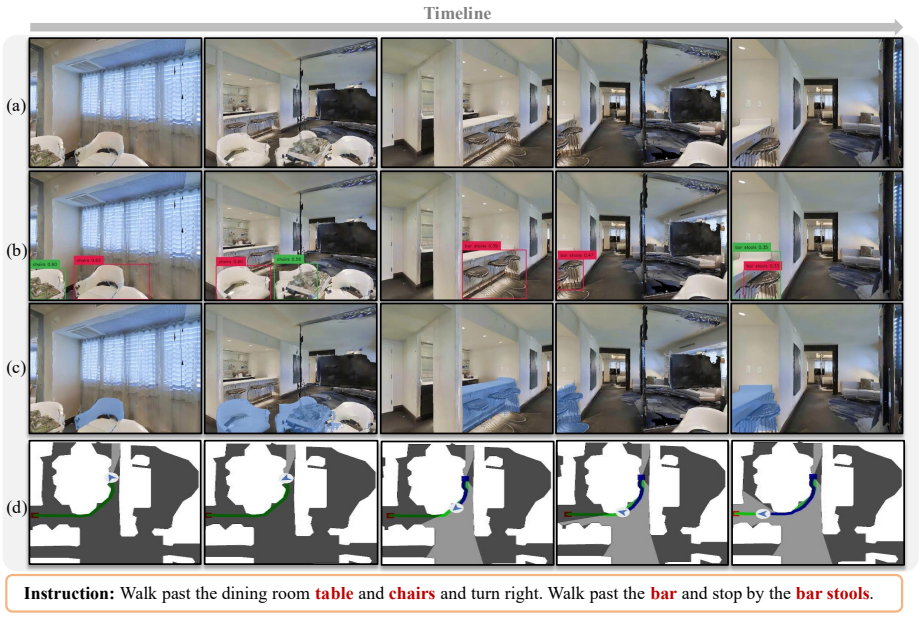

Fig. 6: Qualitative visualization of the Perception-Grounded Step-Aware (PGSA) auditor's cascaded pipeline. (a) Raw continuous egocentric observations o_t during navigation. **(b)** GroundingDINO [38] detects intermediate landmarks. **(c)** SAM3 [3] extracts precise pixel-level masks. **(d)** The corresponding top-down trajectory mapping.

discards null-outcome batches. In contrast, the proposed full SACA framework (green curve) achieves better sample efficiency and stronger asymptotic performance. By dynamically recovering dense supervision from completely failed batches via AFR, SACA maintains stable and effective gradient updates throughout training, alleviating the exploration bottleneck.

Core Components Analysis. Table 2 demonstrates the progressive integration of SACA's core components starting from the SFT Baseline (Row 0). Relying exclusively on sparse outcome rewards, standard GRPO (Row 1) yields only marginal improvements (*e.g.*, +1.3% SR on RxR) and suffers from severe learning-signal collapse, as illustrated by the early plateau of the brown curve in Fig. 5. Incorporating the continuous Soft Score (SS) from the PGSA auditor (Row 2) provides a dense ranking signal crucial for initial bootstrapping, steadily boosting the RxR SR to 51.8%. The subsequent introduction of the All-Failure Rescue (AFR) mechanism (Row 3) brings substantial gains, improving RxR SR by 4.6%. As visually corroborated by Fig. 5, AFR effectively recovers dense supervision from null-outcome batches (green curve), breaking the early performance bottleneck observed when AFR is disabled (blue curve). Finally, integrating Repair Resampling (Row 4) salvages near-miss failures by truncating trajectories at the exact Divergence Point and resampling the suffix.

Objective Constraints & Robustness Mechanisms. Table 3 details the impact of the specific objectives and robust scaling factors. Removing Consistency

Table 3: Ablation study on specific objective designs and robustness mechanisms within SACA. All variants are compared against the full SACA model.

Model Variant	R2R-CE Val-Unseen				RxR-CE Val-Unseen			
	NE↓	**OS↑**	**SR↑**	**SPL↑**	**NE↓**	**SR↑**	**SPL↑**	**nDTW↑**
SACA	**4.57**	**64.9**	**60.3**	**55.1**	**4.90**	**60.3**	**49.8**	**62.1**
Ablation on Objective Constraints								
w/o Consistency Align ($\mathcal{L}_{align}$)	4.89	63.8	58.0	51.6	5.34	57.1	46.2	60.5
w/o Contrastive Correct ($\mathcal{L}_{correct}$)	4.98	63.5	57.3	52.8	5.51	56.4	46.5	59.4
Ablation on Robustness Mechanisms								
w/o Margin-Based Rescue ($\kappa = 1$)	4.85	64.1	58.5	53.0	5.25	57.8	47.9	61.2
w/o Negative-Only Scaling ($s = 1$)	4.76	64.5	59.1	54.0	5.12	58.6	48.5	61.6

Table 4: Ablation study on the group size (K) and maximum repair attempts (N_{rep}).

Group Size (K)	**Max Repairs (N_{rep})**	R2R-CE Val-Unseen				RxR-CE Val-Unseen			
		NE↓	**OS↑**	**SR↑**	**SPL↑**	**NE↓**	**SR↑**	**SPL↑**	**nDTW↑**
4	3	4.92	62.1	56.4	51.2	5.30	55.2	45.3	59.5
8	3	**4.57**	**64.9**	**60.3**	**55.1**	**4.90**	**60.3**	**49.8**	**62.1**
16	3	4.59	64.7	60.1	54.8	4.93	59.9	49.5	61.8
8	1	4.81	63.5	58.1	53.4	5.15	57.5	47.1	60.2
8	5	4.60	64.8	60.0	54.9	4.92	60.1	49.6	62.0

Align ($\mathcal{L}_{align}$) significantly reduces navigation efficiency (*e.g.*, R2R SPL drops from 55.1% to 51.6%), indicating that without behavior cloning on the Valid Prefix, the agent takes longer, sub-optimal paths. Conversely, omitting Contrastive Correct ($\mathcal{L}_{correct}$) heavily degrades long-horizon success (*e.g.*, RxR SR drops to 56.4%), emphasizing the necessity of explicitly penalizing the Divergence Point to prevent repeated mistakes at complex intersections. Furthermore, disabling robustness mechanisms like Margin-Based Rescue ($\kappa = 1$) or Negative-Only Scaling ($s = 1$) degrades overall performance. This demonstrates their crucial roles in filtering noisy gradients from low-confidence pseudo-anchors and acting as regularizers against the over-penalization of plausible alternative paths.

Qualitative Visualization of PGSA Auditor. To intuitively demonstrate the efficacy of the proposed cascaded perception modules, we visualize a navigation episode in Fig. 6. Given an instruction containing intermediate landmarks (*e.g.*, "chairs", "bar stools"), raw visual observations (Row a) lack explicit object-level focus. GroundingDINO [38] (Row b) successfully detects these textual landmarks, yielding bounding boxes. However, standard bounding boxes inevitably encompass noisy background pixels. SAM3 [3] (Row c) refines these detections into precise pixel-level masks, strictly isolating the target objects. This pure spatial-semantic grounding ensures robust local CLIP [44] alignment, allowing the Auditor to accurately track progress on the top-down trajectory (Row d) and pinpoint the exact Divergence Point without any trained reward models.

Table 5: Ablation study on the perception modules within the PGSA Auditor. Global: CLIP [44] Image-Text Similarity; **BBox**: GroundingDINO [38] object detection; **Mask**: SAM3 [3] precise segmentation.

Row	Perception Modules			R2R-CE Val-Unseen				RxR-CE Val-Unseen			
	Global	**BBox**	**Mask**	**NE↓**	**OS↑**	**SR↑**	**SPL↑**	**NE↓**	**SR↑**	**SPL↑**	**nDTW↑**
1	✓			4.95	63.2	57.6	52.1	5.45	56.5	46.2	59.5
2	✓	✓		4.71	64.1	59.2	54.0	5.11	58.7	48.3	61.1
3	✓	✓	✓	**4.57**	**64.9**	**60.3**	**55.1**	**4.90**	**60.3**	**49.8**	**62.1**

Hyperparameter Sensitivity. Table 4 evaluates SACA's sensitivity to group size (K) and maximum repair attempts (N_{rep}). A smaller group ($K = 4$) suffers from high variance in advantage estimation (*e.g.*, RxR SR drops to 55.2%), while a larger group ($K = 16$) introduces computational overhead and noisy long-tail trajectories that degrade performance (R2R SR drops to 60.1%), making $K = 8$ the optimal balance. For repair attempts, $N_{rep} = 1$ provides insufficient exploration depth. Conversely, excessive attempts ($N_{rep} = 5$) cause the policy to overfit to highly improbable, "stitched" trajectories, leading to a slight performance decay. Thus, $N_{rep} = 3$ balances exploration depth and policy stability.

Perception Modules within the PGSA Auditor. Table 5 evaluates the cascaded perception modules of the PGSA auditor. Relying solely on global CLIP similarity (Row 1) yields suboptimal results (*e.g.*, 57.6% SR on R2R). Adding GroundingDINO [38] bounding boxes (Row 2) provides explicit object-level grounding, which increases R2R SR to 59.2%. Finally, incorporating SAM3 [3] masks (Row 3) enables precise pixel-level segmentation. This step filters out background noise to ensure pure local semantic alignment, leading to 60.3% SR, the best result among the compared methods.

5 Conclusion

In this work, we introduce Step-Aware Contrastive Alignment (SACA), a novel reinforcement fine-tuning framework designed to overcome learning-signal collapse in sparse-reward Vision-Language Navigation. By leveraging a zero-shot Perception-Grounded Step-Aware (PGSA) auditor, SACA effectively extracts dense, step-level supervision for both continuous ranking scores and discrete structural boundaries, bypassing the need for costly task-specific Process Reward Models. The Scenario-Conditioned Group Construction mechanism integrates Repair Resampling for near-misses and All-Failure Rescue for entirely failed batches. The robust SACA optimization objective fuses trajectory-level advantages and step-level constraints to extract dense supervision from failed rollouts, preventing RL signal collapse. Extensive experiments on VLN-CE benchmarks demonstrate that SACA establishes a new state-of-the-art performance.

Acknowledgements

This work was supported in part by the "Pioneer" and "Leading Goose" R&D Program of Zhejiang (No. 2025C02032), the National Natural Science Foundation of China (62472381), the Fundamental Research Funds for the Central Universities (226-2025-00080), and the Earth System Big Data Platform of the School of Earth Sciences, Zhejiang University.

References

1. An, D., Wang, H., Wang, W., Wang, Z., Huang, Y., He, K., Wang, L.: Etpnav: Evolving topological planning for vision-language navigation in continuous environments. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **47**(7), 5130–5145 (2025)
2. Anderson, P., Wu, Q., Teney, D., Bruce, J., Johnson, M., Sünderhauf, N., Reid, I., Gould, S., Van Den Hengel, A.: Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3674–3682 (2018)
3. Carion, N., Gustafson, L., Hu, Y.T., Debnath, S., Hu, R., Suris, D., Ryali, C., Alwala, K.V., Khedr, H., Huang, A., Lei, J., Ma, T., Guo, B., Kalla, A., Marks, M., Greer, J., Wang, M., Sun, P., Rädle, R., Afouras, T., Mavroudi, E., Xu, K., Wu, T.H., Zhou, Y., Momeni, L., Hazra, R., Ding, S., Vaze, S., Porcher, F., Li, F., Li, S., Kamath, A., Cheng, H.K., Dollár, P., Ravi, N., Saenko, K., Zhang, P., Feichtenhofer, C.: Sam 3: Segment anything with concepts. arXiv preprint arXiv:2511.16719 (2025)
4. Chang, A., Dai, A., Funkhouser, T., Halber, M., Nießner, M., Savva, M., Song, S., Zeng, A., Zhang, Y.: Matterport3d: Learning from rgb-d data in indoor environments. In: Proceedings of the International Conference on 3D Vision (3DV). pp. 667–676 (2017)
5. Chen, J., Gao, C., Meng, E., Zhang, Q., Liu, S.: Reinforced structured state-evolution for vision-language navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 15450–15459 (2022)
6. Chen, K., Wang, J., Zhang, J., Li, M., Lu, Y., Fan, H.: Scaling video understanding via compact latent multi-agent collaboration. arXiv preprint arXiv:2605.00444 (2026)
7. Chen, K., Chen, J.K., Chuang, J., Vázquez, M., Savarese, S.: Topological planning with transformers for vision-and-language navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11276–11286 (2021)
8. Chen, P., Ji, D., Lin, K., Zeng, R., Li, T., Tan, M., Gan, C.: Weakly-supervised multi-granularity map learning for vision-and-language navigation. Advances in Neural Information Processing Systems (NeurIPS) **35**, 38149–38161 (2022)
9. Chen, P., Sun, X., Zhi, H., Zeng, R., Li, T.H., Liu, G., Tan, M., Gan, C.: a^2 nav: Action-aware zero-shot robot navigation by exploiting vision-and-language ability of foundation models. arXiv preprint arXiv:2308.07997 (2023)

10. Chen, S., Guhur, P.L., Tapaswi, M., Schmid, C., Laptev, I.: 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 (CVPR). pp. 16537–16547 (2022)
11. Chen, X., Tao, K., Shao, K., Wang, H.: Streamingtom: Streaming token compression for efficient video understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 24675–24685 (2026)
12. Cheng, A.C., Ji, Y., Yang, Z., Gongye, Z., Zou, X., Kautz, J., Bıyık, E., Yin, H., Liu, S., Wang, X.: Navila: Legged robot vision-language-action model for navigation. In: Proceedings of Robotics: Science and Systems (RSS) (2025)
13. Cui, G., Yuan, L., Wang, Z., Wang, H., Zhang, Y., Chen, J., Li, W., He, B., Fan, Y., Yu, T., et al.: Process reinforcement through implicit rewards. arXiv preprint arXiv:2502.01456 (2025)
14. Fan, S., Liu, R., Wang, W., Yang, Y.: Navigation instruction generation with bev perception and large language models. In: European Conference on Computer Vision (ECCV). pp. 368–387 (2024)
15. Fan, S., Liu, R., Wang, W., Yang, Y.: Scene map-based prompt tuning for navigation instruction generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6898–6908 (2025)
16. Gao, J., Liu, R., Wang, W.: 3d gaussian map with open-set semantic grouping for vision-language navigation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 9252–9262 (2025)
17. Gao, J., Liu, R., Xu, Y., Cao, T., Zhang, Y., Zhang, Z., Peng, S., Yang, Y., Wang, W.: Uncertainty-aware gaussian map for vision-language navigation. In: International Conference on Learning Representations (ICLR) (2026)
18. Georgakis, G., Schmeckpeper, K., Wanchoo, K., Dan, S., Miltsakaki, E., Roth, D., Daniilidis, K.: Cross-modal map learning for vision and language navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 15460–15470 (2022)
19. Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., et al.: Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. arXiv:2501.12948 (2025)
20. Guo, W., Lu, G., Deng, H., Wu, Z., Tang, Y., Wang, Z.: Vla-reasoner: Empowering vision-language-action models with reasoning via online monte carlo tree search. arXiv preprint arXiv:2509.22643 (2025)
21. Guo, Y., Zhang, J., Chen, X., Ji, X., Wang, Y.J., Hu, Y., Chen, J.: Improving vision-language-action model with online reinforcement learning. In: 2025 IEEE International Conference on Robotics and Automation (ICRA). pp. 15665–15672 (2025)
22. Hong, Y., Wang, Z., Wu, Q., Gould, S.: Bridging the gap between learning in discrete and continuous environments for vision-and-language navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 15439–15449 (2022)
23. Hurst, A., Lerer, A., Goucher, A.P., Perelman, A., Ramesh, A., Clark, A., Ostrow, A., Welihinda, A., Hayes, A., Radford, A., et al.: Gpt-4o system card. arXiv preprint arXiv:2410.21276 (2024)
24. Jaech, A., Kalai, A., Lerer, A., Richardson, A., El-Kishky, A., Low, A., Helyar, A., Madry, A., Beutel, A., Carney, A., et al.: Openai o1 system card. arXiv preprint arXiv:2412.16720 (2024)

25. Koenig, N., Howard, A.: Design and use paradigms for gazebo, an open-source multi-robot simulator. In: 2004 IEEE/RSJ international conference on intelligent robots and systems (IROS). vol. 3, pp. 2149–2154 (2004)
26. Krantz, J., Gokaslan, A., Batra, D., Lee, S., Maksymets, O.: Waypoint models for instruction-guided navigation in continuous environments. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 15162–15171 (2021)
27. Krantz, J., Lee, S.: Sim-2-sim transfer for vision-and-language navigation in continuous environments. In: European Conference on Computer Vision (ECCV). pp. 588–603 (2022)
28. Krantz, J., Wijmans, E., Majundar, A., Batra, D., Lee, S.: Beyond the nav-graph: Vision and language navigation in continuous environments. In: European Conference on Computer Vision (ECCV). pp. 104–120 (2020)
29. Ku, A., Anderson, P., Patel, R., Ie, E., Baldridge, J.: Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. arXiv preprint arXiv:2010.07954 (2020)
30. Li, H., Hu, Z., Wang, J., Fan, H., Yang, Y.: Skill-3d: Evolving scene-aware skills for agentic 3d spatial reasoning. arXiv preprint arXiv:2606.07436 (2026)
31. Li, Y., Wang, C., Jia, J.: Llama-vid: An image is worth 2 tokens in large language models. In: European Conference on Computer Vision (ECCV). pp. 323–340 (2024)
32. Liang, J., Makoviychuk, V., Handa, A., Chentanez, N., Macklin, M., Fox, D.: Gpu-accelerated robotic simulation for distributed reinforcement learning. In: Conference on Robot Learning (CoRL). pp. 270–282. PMLR (2018)
33. Lin, J., Yin, H., Ping, W., Molchanov, P., Shoeybi, M., Han, S.: Vila: On pretraining for visual language models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 26689–26699 (2024)
34. Liu, A., Feng, B., Xue, B., Wang, B., Wu, B., Lu, C., Zhao, C., Deng, C., Zhang, C., Ruan, C., et al.: Deepseek-v3 technical report. arXiv:2412.19437 (2024)
35. Liu, R., Wang, W., Yang, Y.: Vision-language navigation with energy-based policy. Advances in Neural Information Processing Systems (NeurIPS) **37**, 108208–108230 (2024)
36. Liu, R., Wang, W., Yang, Y.: Volumetric environment representation for vision-language navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 16317–16328 (2024)
37. Liu, R., Wang, X., Wang, W., Yang, Y.: Bird's-eye-view scene graph for vision-language navigation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 10968–10980 (2023)
38. Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Jiang, Q., Li, C., Yang, J., Su, H., et al.: Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In: European Conference on Computer Vision (ECCV). pp. 38–55. Springer (2024)
39. Long, Y., Cai, W., Wang, H., Zhan, G., Dong, H.: Instructnav: Zero-shot system for generic instruction navigation in unexplored environment. arXiv preprint arXiv:2406.04882 (2024)
40. Long, Y., Li, X., Cai, W., Dong, H.: Discuss before moving: Visual language navigation via multi-expert discussions. arXiv preprint arXiv:2309.11382 (2023)
41. Puig, X., Undersander, E., Szot, A., Cote, M.D., Partsey, R., Yang, J., Desai, R., Clegg, A.W., Hlavac, M., Min, T., Gervet, T., Vondruš, V., Berges, V.P., Turner, J., Maksymets, O., Kira, Z., Kalakrishnan, M., Malik, J., Chaplot, D.S., Jain, U., Batra, D., Rai, A., Mottaghi, R.: Habitat 3.0: A co-habitat for humans, avatars and robots (2023)

42. Qi, Y., Wu, Q., Anderson, P., Wang, X., Wang, W.Y., Shen, C., Hengel, A.v.d.: Reverie: Remote embodied visual referring expression in real indoor environments. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 9982–9991 (2020)
43. Qi, Z., Zhang, Z., Yu, Y., Wang, J., Zhao, H.: Vln-r1: Vision-language navigation via reinforcement fine-tuning. arXiv preprint arXiv:2506.17221 (2025)
44. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning (ICML). pp. 8748–8763. PMLR (2021)
45. Rafailov, R., Sharma, A., Mitchell, E., Manning, C.D., Ermon, S., Finn, C.: Direct preference optimization: Your language model is secretly a reward model. Advances in Neural Information Processing Systems (NeurIPS) **36**, 53728–53741 (2023)
46. Raychaudhuri, S., Wani, S., Patel, S., Jain, U., Chang, A.: Language-aligned waypoint (law) supervision for vision-and-language navigation in continuous environments. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 4018–4028 (2021)
47. Savva, M., Kadian, A., Maksymets, O., Zhao, Y., Wijmans, E., Jain, B., Straub, J., Liu, J., Koltun, V., Malik, J., et al.: Habitat: A platform for embodied ai research. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 9339–9347 (2019)
48. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347 (2017)
49. Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., Zhang, H., Zhang, M., Li, Y., Wu, Y., et al.: Deepseekmath: Pushing the limits of mathematical reasoning in open language models. arXiv:2402.03300 (2024)
50. Singh, A., Fry, A., Perelman, A., Tart, A., Ganesh, A., El-Kishky, A., McLaughlin, A., Low, A., Ostrow, A., Ananthram, A., et al.: Openai gpt-5 system card. arXiv preprint arXiv:2601.03267 (2025)
51. Wang, H., Liang, W., Van Gool, L., Wang, W.: Dreamwalker: Mental planning for continuous vision-language navigation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 10873–10883 (2023)
52. Wang, H., Wang, W., Liang, W., Xiong, C., Shen, J.: Structured scene memory for vision-language navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 8455–8464 (2021)
53. Wang, X., Zhu, W., Wang, T., Geng, T., Zhang, Z., Qi, Z., Yang, J., Zheng, F.: Livevln: Breaking the stop-and-go loop in vision-language navigation. arXiv preprint arXiv:2604.19536 (2026)
54. Wang, X., Xiong, W., Wang, H., Wang, W.Y.: Look before you leap: Bridging model-free and model-based reinforcement learning for planned-ahead vision-and-language navigation. In: European Conference on Computer Vision (ECCV). pp. 37–53 (2018)
55. Wang, Z., Li, X., Yang, J., Liu, Y., Hu, J., Jiang, M., Jiang, S.: Lookahead exploration with neural radiance representation for continuous vision-language navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 13753–13762 (2024)
56. Wang, Z., Li, X., Yang, J., Liu, Y., Jiang, S.: Gridmm: Grid memory map for vision-and-language navigation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 15625–15636 (2023)

57. Wang, Z., Li, J., Hong, Y., Wang, Y., Wu, Q., Bansal, M., Gould, S., Tan, H., Qiao, Y.: Scaling data generation in vision-and-language navigation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 12009–12020 (2023)
58. Wei, M., Wan, C., Yu, X., Wang, T., Yang, Y., Mao, X., Zhu, C., Cai, W., Wang, H., Chen, Y., Liu, X., Pang, J.: Streamvln: Streaming vision-and-language navigation via slowfast context modeling. arXiv preprint arXiv:2507.05240 (2025)
59. Wu, Z., Hu, Y., Shi, W., Dziri, N., Suhr, A., Ammanabrolu, P., Smith, N.A., Ostendorf, M., Hajishirzi, H.: Fine-grained human feedback gives better rewards for language model training. Advances in Neural Information Processing Systems (NeurIPS) **36**, 59008–59033 (2023)
60. Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al.: Qwen3 technical report. arXiv preprint arXiv:2505.09388 (2025)
61. Ye, A., Zhang, Z., Wang, B., Wang, X., Zhang, D., Zhu, Z.: Vla-r1: Enhancing reasoning in vision-language-action models. arXiv preprint arXiv:2510.01623 (2025)
62. Yin, Z., Sun, Q., Zeng, Z., Cheng, Q., Qiu, X., Huang, X.: Dynamic and generalizable process reward modeling. In: Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 4203–4233 (2025)
63. Zhai, S., Bai, H., Lin, Z., Pan, J., Tong, P., Zhou, Y., Suhr, A., Xie, S., LeCun, Y., Ma, Y., et al.: Fine-tuning large vision-language models as decision-making agents via reinforcement learning. Advances in Neural Information Processing Systems (NeurIPS) **37**, 110935–110971 (2024)
64. Zhang, J., Wang, K., Wang, S., Li, M., Liu, H., Wei, S., Wang, Z., Zhang, Z., Wang, H.: Uni-navid: A video-based vision-language-action model for unifying embodied navigation tasks. In: Proceedings of Robotics: Science and Systems (RSS) (2025)
65. Zhang, J., Wang, K., Xu, R., Zhou, G., Hong, Y., Fang, X., Wu, Q., Zhang, Z., Wang, H.: Navid: Video-based vlm plans the next step for vision-and-language navigation. In: Proceedings of Robotics: Science and Systems (RSS) (2024)
66. Zhang, L., Hao, X., Xu, Q., Zhang, Q., Zhang, X., Wang, P., Zhang, J., Wang, Z., Zhang, S., Xu, R.: Mapnav: A novel memory representation via annotated semantic maps for vlm-based vision-and-language navigation. In: Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 13032–13056 (2025)
67. Zhang, R., Zhao, C., Du, H., Niyato, D., Wang, J., Sawadsitang, S., Shen, X., Kim, D.I.: Embodied ai-enhanced vehicular networks: An integrated vision language models and reinforcement learning method. IEEE Transactions on Mobile Computing (2025)
68. Zhang, Y., Wu, J., Li, W., Li, B., Ma, Z., Liu, Z., Li, C.: Video instruction tuning with synthetic data. arXiv preprint arXiv:2410.02713 (2024)
69. Zheng, C., Liu, S., Li, M., Chen, X.H., Yu, B., Gao, C., Dang, K., Liu, Y., Men, R., Yang, A., et al.: Group sequence policy optimization. arXiv preprint arXiv:2507.18071 (2025)
70. Zheng, D., Huang, S., Zhao, L., Zhong, Y., Wang, L.: Towards learning a generalist model for embodied navigation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 13624–13634 (2024)
71. Zhou, G., Hong, Y., Wu, Q.: Navgpt: Explicit reasoning in vision-and-language navigation with large language models. In: Proceedings of the AAAI Conference on Artificial Intelligence (AAAI). pp. 7641–7649 (2024)