Trajectory-Diversity-Driven Robust Vision-and-Language Navigation

Jiangyang Li[1] Cong Wan[4] Songlin Dong[2,3*] Chenhao Ding[4] Qiang Wang[1]
Zhiheng Ma[2,3] Yihong Gong[1]
[1]State Key Laboratory of Human-Machine Hybrid Augmented Intelligence, National Engineering Research Center for Visual Information and Applications, and Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University
[2]Faculty of Computility Microelectronics, Shenzhen University of Advanced Technology
[3]Guangdong Provincial Key Laboratory of Computility Microelectronics
[4]School of Software Engineering, Xi'an Jiaotong University

Abstract

*Vision-and-Language Navigation (VLN) requires agents to navigate photo-realistic environments following natural language instructions. Current methods predominantly rely on imitation learning, which suffers from limited generalization and poor robustness to execution perturbations. We present **NavGRPO**, a reinforcement learning framework that learns goal-directed navigation policies through Group Relative Policy Optimization. By exploring diverse trajectories and optimizing via within-group performance comparisons, our method enables agents to distinguish effective strategies beyond expert paths without requiring additional value networks. Built on ScaleVLN, **NavGRPO** achieves superior robustness on R2R and REVERIE benchmarks with **+3.0%** and **+1.71%** SPL improvements in unseen environments. Under extreme early-stage perturbations, we demonstrate **+14.89% SPL gain** over the baseline, confirming that goal-directed RL training builds substantially more robust navigation policies. The code is available at* https://github.com/cocoastar/NavGRPO.

1. Introduction

Vision-and-Language Navigation (VLN) [4] is a core task in Embodied AI, requiring agents to autonomously navigate to target locations based on natural language instructions. This task is highly challenging as agents must ground abstract linguistic concepts to visual observations and perform multi-step reasoning to achieve navigation goals. Additionally, VLN tasks typically require agents to execute instructions in unseen environments, posing stringent tests on the model's generalization capability.

*Corresponding author: Songlin Dong (dongsl@suat-sz.edu.cn)

Currently, the mainstream paradigm for solving VLN tasks is imitation learning (IL), which employs DAgger-based [11] techniques to train agents to mimic expert demonstrations through supervised learning. Although these methods have achieved significant results in seen environments, their optimization objective is to imitate expert behavior rather than learn goal-directed reasoning. Therefore, two fundamental limitations exist: (i) Limited generalization capability: IL agents rely on expert trajectories in training data, leading to overfitting and difficulty generalizing to unseen environments. (ii) Poor robustness to perturbations: IL methods suffer from the inherent distribution shift problem. When agents deviate from expert paths due to perturbations, they enter unseen state distributions. Without effective recovery policies, such deviations often prevent task completion.

Figure 1 intuitively shows that in unseen environments, when IL agents are perturbed and deviate from expert paths, they often have to take significant detours due to a lack of effective recovery strategies. This raises the core question of this paper: *Can we train a VLN agent that not only overcomes the limited generalization capability of IL, but also possesses effective recovery and replanning capabilities when perturbed and deviating from expert paths?*

We hypothesize that robust learning requires exposing agents to **diverse navigation trajectories**, including both successes and failures. By learning relative distinctions across all outcomes, rather than relying solely on the absolute correctness of expert demonstrations, an agent can extract a richer learning signal that generalizes beyond simple expert replication.

To realize this hypothesis, we propose NavGRPO, a reinforcement learning framework designed for robust generalization built on three key design choices. (1) Group Relative Policy Optimization [20, 58]. GRPO samples

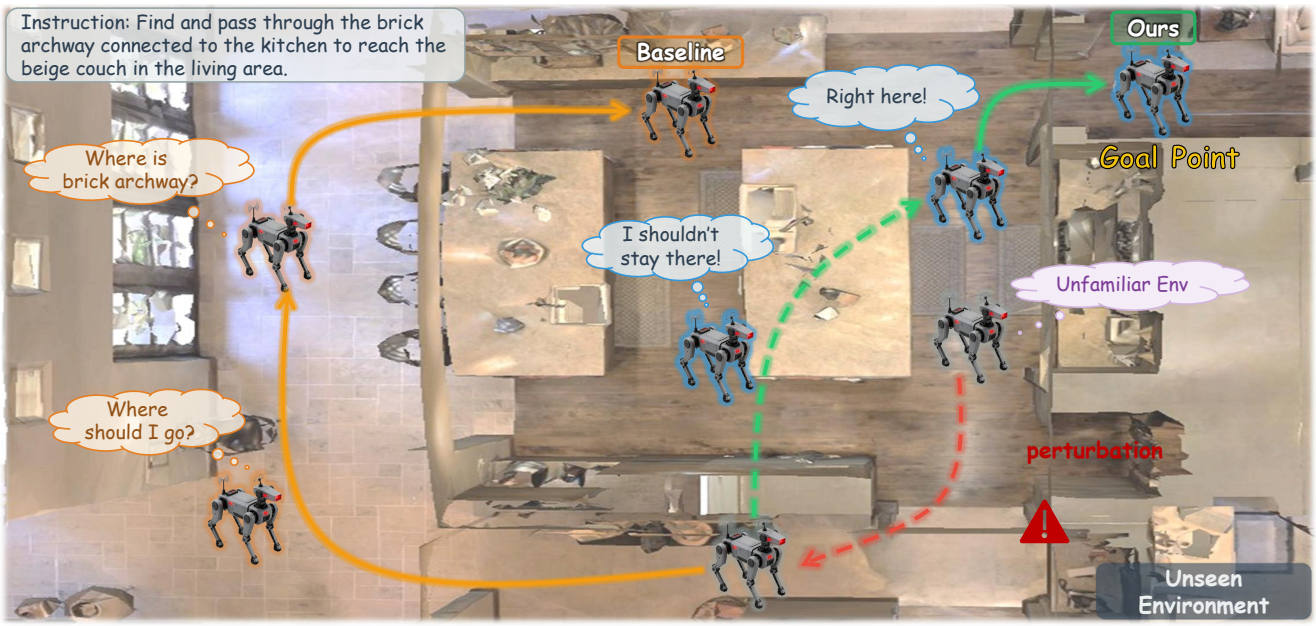

Figure 1. Navigation behavior under early-stage perturbations in unfamiliar environments. The baseline IL agent struggles to recover from errors due to limited exposure to failed trajectories, often detouring or failing to reach the goal. Our NavGRPO agent learns from diverse rollouts, enabling robust error correction and successful navigation despite perturbations.

multiple trajectories per instruction and optimizes policies through within-group performance comparisons. Unlike value-based RL methods [10] that require auxiliary critic networks, GRPO directly learns from trajectory-level rewards by ranking outcomes within each group. This design naturally encourages exploration of diverse navigation strategies while avoiding the instability of value network training, enabling agents to discover effective recovery behaviors beyond expert demonstrations. (2) Goal-oriented trajectory reward. Unlike prior RL methods using step-wise rewards [6], our reward directly evaluates complete trajectories based on goal achievement and path efficiency, providing clearer learning signals. (3) Adaptive training strategy. We incorporate targeted supervised fine-tuning on challenging instructions to prevent catastrophic forgetting and complement reinforcement learning with behavioral guidance.

On R2R [4] and REVERIE [53] validation unseen splits, we achieve +3% and +1.71% SPL improvements over the ScaleVLN baseline. More importantly, under extreme perturbations where agents are forced off-path in early steps, we demonstrate superior robustness with **+14.89% SPL** improvement compared to ScaleVLN, showing effective robustness. Our contributions are: (1) We establish an effective framework for VLN that integrates GRPO with a multi-level reward function and adaptive training strategy, applicable to multiple baseline models. (2) Perturbation experiments show that agents trained with GRPO exhibit stronger robustness to perturbations present in navigation. (3) On standard R2R and REVERIE unseen benchmarks, we provide empirical evidence that goal-directed RL fine-tuning improves generalization to unseen environments.

2. Related Work

Vision-and-Language Navigation (VLN). VLN requires embodied agents to follow natural language instructions to reach target locations in photo-realistic environments. The task was formalized with the Matterport3D simulator [5], and various datasets have since been introduced to address different aspects of embodied navigation [4, 33, 53, 82], covering diverse scenarios from fine-grained object interaction to long-horizon instruction following across indoor and outdoor scenes. Early approaches employed sequence-to-sequence architectures with imitation learning [4, 17, 39], which were later complemented by diverse training strategies including reinforcement learning [59, 64, 65], adversarial training [18, 42, 74], generative modeling [34], curriculum learning [72], and cycle-consistent learning [60]. The integration of vision-language pre-training brought substantial improvements through large-scale offline pre-training [13, 21, 22, 26], auxiliary task design [41, 50, 81], and regularization techniques [52, 62, 67] for more stable and less biased training. Architectural innovations evolved from recurrent encoders to sophisticated representations, including attention-based mechanisms [10, 24], graph-based topological reasoning [8, 11, 14], and scene representations [6, 45, 61, 69], with recent works designing flexi-

ble action spaces for efficient exploration and backtracking [11, 19, 28]. To enhance cross-modal understanding [15, 16], researchers have pursued finer-grained visual feature extraction [27, 44, 51] and textual decomposition [12, 36, 43, 75], while incorporating external knowledge from large language models [7, 78–80], vision-language models [38], and structured knowledge bases [40]. Parallel efforts in data augmentation have explored observation perturbation [23, 29, 37], automatic trajectory annotation via speaker-follower frameworks [17, 30, 70], and large-scale scene generation [35, 46, 47, 68]. Despite these advances, existing methods predominantly rely on imitation learning from expert demonstrations, limiting exposure to diverse navigation scenarios. This results in poor generalization to novel configurations and fragility under execution deviations. We address these limitations with NavGRPO, a trajectory-level reinforcement learning approach that learns from diverse navigation experiences to develop robust policies.

Reinforcement Learning for VLN. While imitation learning dominates VLN research, several efforts have explored reinforcement learning to address generalization challenges. Early work by Wang et al. [65] introduced RCM, which employs a matching critic to provide intrinsic rewards for vision-language alignment. Their method further incorporated Self-Imitation Learning to exploit successful trajectories in unseen environments, demonstrating improved generalization on R2R. Subsequent approaches focused on reward engineering: SERL [59] proposed soft expert reward learning to distill expert demonstrations into dense reward signals, avoiding manual reward shaping while maintaining strong supervision from human annotations. More recent methods have integrated RL into their training pipelines, with HAMT [10] applying policy gradient fine-tuning after pretraining, and SEvol [6] using RL to refine graph-based scene representations. However, these methods face critical limitations: step-level sparse rewards lead to severe credit assignment issues in long-horizon navigation, while learned value networks struggle in VLN's high-dimensional action space. We adopt Group Relative Policy Optimization (GRPO) [20], which eliminates both issues through trajectory-level geometric rewards and advantage estimation via relative ranking, enabling robust learning from diverse trajectories including both successful and failed attempts.

3. Method

Problem Definition. In the standard VLN setup, the environment is represented as an undirected navigation graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V} = \{V_i\}_{i=1}^{K}$ denotes a set of K navigable locations and $\mathcal{E}$ represents the connectivity edges between locations. Each location in the graph corresponds to a specific viewpoint in the physical environment, and the agent can only traverse along the edges defined by the graph structure. Given a natural language instruction $\mathcal{W} = \{w_1, w_2, ..., w_L\}$ composed of L words, at each navigation timestep t, the agent is situated at location $V_t \in \mathcal{V}$ and perceives the surrounding environment through panoramic visual observations $\mathcal{O}_t = \{o_i\}_{i=1}^{n}$ captured at the current location, consisting of n view images. The agent also observes a set of reachable neighboring nodes $\mathcal{N}(V_t)$. Based on the instruction and current observations, the agent must select an action a_t from the available action space $\mathcal{A}_t$. The action space includes navigating to a neighboring location $V_{t+1} \in \mathcal{N}(V_t)$ connected to the current position, or issuing a stop signal to terminate the navigation episode.

Overview. NavGRPO trains navigation policies through Group Relative Policy Optimization, which learns from diverse trajectories sampled during training. We describe the GRPO framework in Sec. 3.1, our geometric reward design in Sec. 3.2, and the training pipeline in Sec. 3.3.

3.1. NavGRPO for VLN

We adopt GRPO [20], which estimates advantages through group-based reward comparison without requiring a separate value network.

Policy Architecture. The proposed navigation policy π_θ is parameterized by the vision-language transformer network. At each timestep t, given the instruction $\mathcal{W}$, the current panoramic observations $\mathcal{O}_t$, and the graph map representation $\mathcal{M}_t$ that encodes spatial topology and historical navigation information, the policy outputs a probability distribution over the action space:

$$\pi_\theta(a_t|\mathcal{W}, \mathcal{O}_t, \mathcal{M}_t) = \text{softmax}(f_\theta(\mathcal{W}, \mathcal{O}_t, \mathcal{M}_t)) \quad (1)$$

where f_θ represents the network's logit output for each candidate action in $\mathcal{A}_t$.

Group-based Trajectory Sampling. For each instruction $\mathcal{W}_i$ sampled from the instruction set $\mathcal{D}_\mathcal{B}$, which denotes a mini-batch of B instructions, we sample a group of K independent trajectories by rolling out the current policy π_θ. Each trajectory $\tau_k = (s_{k,0}, a_{k,0}, s_{k,1}, a_{k,1}, ..., s_{k,T})$ represents a complete navigation episode, where $s_{k,t} = (V_{k,t}, \mathcal{O}_{k,t}, \mathcal{M}_{k,t})$ denotes the state at timestep t. During sampling, actions are drawn stochastically from $a_t \sim \pi_\theta(\cdot|s_{k,t})$, and we store the corresponding log probability $\log \pi_\theta(a_{k,t}|s_{k,t})$ as $p_{k,t}^{\text{old}}$ for later policy updates.

Debiased Group Relative Advantage Estimation. For each instruction $\mathcal{W}$ in the training batch, we sample K trajectories and compute their rewards $\{r_1, r_2, ..., r_K\}$ using the reward function detailed in Section 3.2. Following Dr.GRPO [49], we estimate advantages through group-relative comparison without variance normalization:

$$\hat{A}_k = r_k - \text{mean}(\{r_1, ..., r_K\}) \quad (2)$$

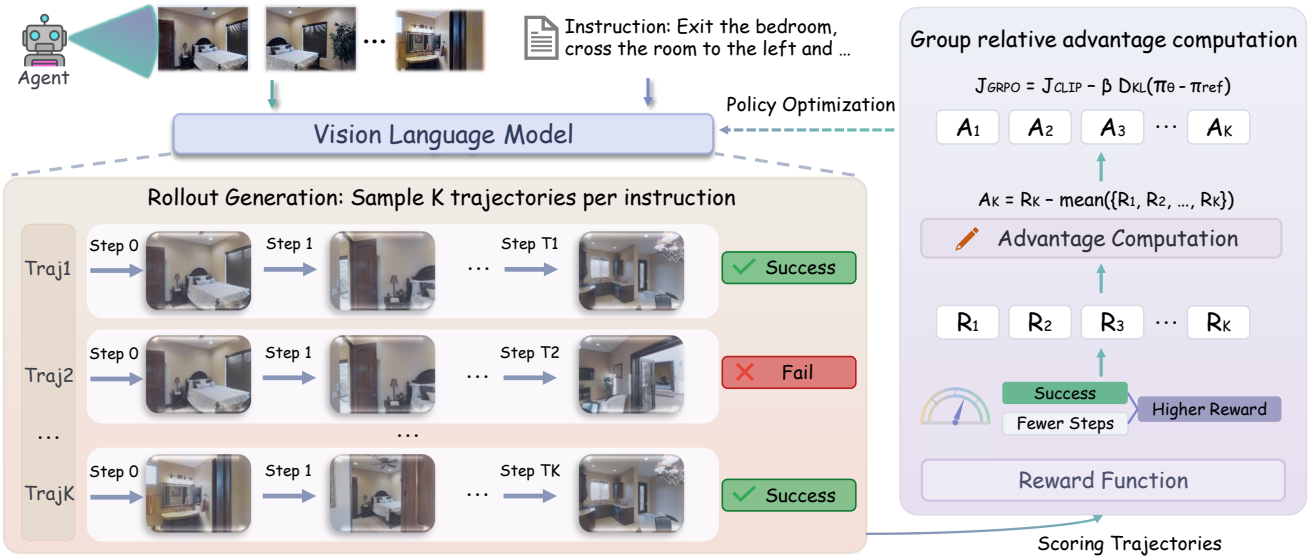

Figure 2. Overview of our NavGRPO training framework for vision-language navigation. For each instruction, we sample K diverse trajectories through policy rollout, compute rewards using trajectory-level and step-level signals, estimate group relative advantages by comparing within instruction groups, and optimize the policy through debiased advantage estimation without value networks.

This uses the empirical group mean as a natural baseline, eliminating the need for separate value function approximation. Removing variance normalization avoids amplifying noise in low-diversity groups, making the learning signal more robust. The trajectory-level advantage is broadcast to all timesteps during policy updates.

Policy Optimization Objective. For each state-action transition $(s_{k,t}, a_{k,t})$ in trajectory τ_k, we compute the probability ratio:

$$\rho_{k,t} = \frac{\pi_\theta(a_{k,t}|s_{k,t})}{\pi_{\theta_{\text{old}}}(a_{k,t}|s_{k,t})} = \exp\left(\log \pi_\theta(a_{k,t}|s_{k,t}) - p_{k,t}^{\text{old}}\right) \tag{3}$$

where $\pi_{\theta_{\text{old}}}$ represents the behavior policy with frozen parameters. We incorporate the step-level progress coefficient $\gamma_{k,t}$ from Section 3.2 to modulate the advantage signal. Following PPO [57], the clipped surrogate objective becomes:

$$\mathcal{J}_{\text{clip}}(k,t) = \min\left(\gamma_{k,t}\hat{A}_k\rho_{k,t},\ \gamma_{k,t}\hat{A}_k \cdot \text{clip}(\rho_{k,t}, 1-\delta, 1+\delta)\right) \tag{4}$$

The clipping mechanism constrains $\rho_{k,t}$ within $[1-\delta, 1+\delta]$ to prevent excessively large policy updates. The complete optimization objective is:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{\mathcal{W}\sim\mathcal{D}_\mathcal{B},\{\tau_k\}_{k=1}^K\sim\pi_{\theta_{\text{old}}}(\cdot|\mathcal{W})} \left[\frac{1}{K}\sum_{k=1}^{K}\sum_{t=1}^{|\tau_k|}\left(\mathcal{J}_{\text{clip}}(k,t) - \beta\mathbb{D}_{\text{KL}}[\pi_\theta\|\pi_{\text{ref}}]\right)\right] \tag{5}$$

We aggregate across all timesteps without length normalization to avoid introducing bias. The KL divergence term regularizes the policy against the reference policy π_{ref}, which is the fixed policy before RL training begins, weighted by β to prevent excessive deviation.

3.2. Reward Function for NavGRPO

The reward function guides the RL training process by providing learning signals at both trajectory and step levels. We design a composite reward function with trajectory-level rewards for overall navigation quality and step-level rewards for step-wise progress.

Navigation Success Reward. Let V_T^k denote the final viewpoint of trajectory τ_k and V^* represent the target destination. The navigation error is $d_k = d_{\text{shortest}}(V_T^k, V^*)$. The success reward uses exponential decay:

$$R_{\text{nav}}(d_k) = \exp\left(-\frac{d_k^2}{2\epsilon^2}\right) \cdot \mathbb{I}(d_k < \epsilon) \tag{6}$$

Where ϵ is the success threshold. This provides smooth decay that emphasizes precise goal reaching while maintaining differentiability.

Path Efficiency Reward. To discourage inefficient behaviors such as backtracking and detours, we penalize excessive path length:

$$R_{\text{path}}(L_k, L^*) = -\max(L_k - L^*, 0)/L^* \tag{7}$$

where L_k is the actual path length and L^* is the shortest path length.

Total Reward Function. The trajectory-level reward combines navigation success and path efficiency:

$$r_k = R_{\text{nav}}(d_k) + \alpha \cdot R_{\text{path}}(L_k, L^*) \tag{8}$$

Table 1. Robustness analysis under stochastic perturbations on R2R Val Unseen. Left: Global perturbation samples from policy distribution with probability p. Right: Early perturbation selects the least probable action for the first N steps. ΔSPL shows SPL degradation from the unperturbed setting (prob=0 or Steps=0) for each method. NavGRPO demonstrates superior robustness across all perturbation levels.

Method	Global Perturbation						Early Perturbation					
	prob	OSR↑	NE↓	SR↑	SPL↑	ΔSPL	Steps	OSR↑	NE↓	SR↑	SPL↑	ΔSPL
ScaleVLN	0.0	87.48	2.34	79.40	69.97	-0.00	0	87.48	2.34	79.40	69.97	-0.00
NavGRPO	0.0	**89.18**	**2.19**	**81.88**	**72.65**	**-0.00**	0	**89.18**	**2.19**	**81.88**	**72.65**	**-0.00**
ScaleVLN	0.2	87.14	2.43	78.97	67.80	-2.17	1	83.06	2.94	74.76	59.06	-10.91
NavGRPO	0.2	**88.75**	**2.21**	**81.27**	**72.11**	**-0.54**	1	**85.05**	**2.67**	**77.44**	**67.04**	**-5.61**
ScaleVLN	0.4	87.19	2.42	78.67	65.90	-4.07	2	83.61	2.86	75.69	54.25	-15.72
NavGRPO	0.4	**88.79**	**2.22**	**81.06**	**71.69**	**-0.96**	2	**83.85**	**2.70**	**76.50**	**65.10**	**-7.55**
ScaleVLN	0.8	87.06	2.52	77.61	61.94	-8.03	3	83.52	3.04	74.16	47.77	-22.20
NavGRPO	0.8	**88.49**	**2.27**	**79.95**	**69.98**	**-2.67**	3	**82.96**	**2.82**	**75.35**	**62.66**	**-9.99**

where α balances the contribution of path efficiency.

Step-level Progress Coefficient. Navigation tasks possess well-defined spatial metrics that allow quantifying progress at each decision step. For trajectory τ_k at timestep t, we define a progress coefficient that modulates the advantage signal:

$$\gamma_{k,t} = 1 + \text{sign}(\hat{A}_k) \cdot \frac{d_{t-1} - d_t}{L^*} \tag{9}$$

where d_t is the distance to the goal at step t and L^* is the ground-truth shortest path length. This coefficient ensures that steps approaching the goal ($d_{t-1} > d_t$) yield stronger learning signals $\gamma_{k,t} \cdot \hat{A}_k$ in magnitude, while steps deviating from the goal receive attenuated signals, allowing the agent to distinguish productive actions even in failed trajectories.

3.3. Adaptive Training with Hard Case Replay

While our method enables learning from diverse trajectories, certain instructions remain persistently challenging despite extensive sampling. We address this by periodically applying supervised refinement on hard cases identified during RL training.

Hard Case Identification. For each instruction $\mathcal{W}_i$ in the training batch, we sample K trajectories and identify it as a hard case when all sampled trajectories fail:

$$\text{Hard}(\mathcal{W}_i) = \mathbb{I}\left(\sum_{k=1}^{K} \mathbb{I}(d_k < \epsilon) = 0\right) \tag{10}$$

These instructions are stored in buffer $\mathcal{B}_{\text{hard}}$.

Supervised Refinement. When $|\mathcal{B}_{\text{hard}}|$ reaches threshold M, we perform supervised updates on expert trajectories:

$$\mathcal{L}_{\text{hard}}(\theta) = -\frac{1}{|\mathcal{B}_{\text{hard}}|} \sum_{\mathcal{W}_i \in \mathcal{B}_{\text{hard}}} \sum_{t=0}^{T_i} \log \pi_\theta(a_t^* | s_t^*, \mathcal{W}_i) \tag{11}$$

The buffer is then cleared. This adds negligible cost since the compute saved from fewer RL rollouts more than compensates for the supervised updates on hard cases.

4. Experiment

4.1. Experimental Setup

Datasets. We evaluate on three widely-used VLN benchmarks: R2R [4] provides fine-grained step-by-step navigation instructions; REVERIE [53] presents high-level goal-oriented instructions specifying target rooms and objects; R2R-CE [32] extends R2R to continuous environments using Habitat simulator [56], where agents navigate continuously rather than between discrete viewpoints.

Evaluation Metrics. We adopt standard VLN evaluation metrics [4]. Navigation Error (NE) measures the average distance in meters between the agent's final position and the goal location. Success Rate (SR) computes the percentage of episodes where the agent stops within 3 meters of the target. Success Rate penalized by Path Length (SPL) penalizes SR by the ratio of the shortest path length to the actual trajectory length, rewarding both accuracy and efficiency. Oracle Success Rate (OSR) measures the success rate under an ideal stop policy.

Implementation Details. We maintain the same setup as baseline methods [11, 68]. All models are trained for 200k steps with learning rate 1×10^{-5}. After supervised warm-up (30k steps), we transition to GRPO with batch size $B = 8$ and $K = 8$ trajectories per instruction. Following standard PPO settings [57], we set clipping threshold $\delta = 0.2$ and KL penalty $\beta = 0.01$. For reward function, we use $\alpha = 0.25$. For hard case replay, we set buffer size M = 200. As training progresses and the policy improves, the frequency of hard case updates naturally decreases. All results are averaged over three random seeds.

4.2. Robustness to Action Noise

Motivation. Real-world deployment faces execution noise from sensor errors and actuation imprecision. While imitation learning optimizes for near-optimal trajectories, it provides limited exposure to noisy action sequences during

Table 2. Comparison with SoTA methods on R2R and REVERIE datasets. †: Methods using reinforcement learning for policy optimization.

Methods	R2R Val Unseen			R2R Test Unseen			REVERIE Val Unseen			REVERIE Test Unseen		
	NE↓	SR↑	SPL↑	NE↓	SR↑	SPL↑	OSR↑	SR↑	SPL↑	OSR↑	SR↑	SPL↑
RCM† [65]	5.88	43	-	6.12	43	38	14.23	9.29	6.97	11.68	7.84	6.67
SERL† [59]	4.74	56	48	5.63	53	49	-	-	-	-	-	-
VLN⟳BERT [24]	3.93	63	57	4.09	63	57	35.02	30.67	24.90	32.91	29.61	23.99
HAMT† [10]	3.65	66	61	3.93	65	60	36.84	32.95	30.20	33.41	30.40	26.67
SEvol† [6]	3.99	62	57	4.13	62	57	-	-	-	-	-	-
HOP+ [54]	3.49	67	61	3.71	66	60	40.04	36.07	31.13	35.81	33.82	28.24
BEVBert [2]	2.81	75	64	3.13	73	62	56.40	51.78	36.37	57.26	52.81	36.41
LANA [66]	-	68	62	-	65	60	38.54	34.00	29.26	36.41	33.50	26.89
KERM [40]	3.22	72	61	3.61	70	59	55.21	50.44	35.38	57.58	52.43	39.21
GridMM [69]	2.83	75	64	3.35	73	62	57.48	51.37	36.47	59.55	53.13	36.60
NaviLLM [78]	3.51	67	59	3.71	68	60	52.27	42.15	35.68	51.75	39.80	32.33
NavGPT-2 [80]	2.84	74	61	3.33	72	60	-	-	-	-	-	-
VER [48]	2.80	76	65	2.74	76	66	61.09	55.98	39.66	62.22	56.82	38.76
MAGIC-L [63]	2.22	79	70	2.75	77	69	-	-	-	-	-	-
GOAT [62]	2.40	78	68	3.04	75	65	-	53.37	36.70	-	57.72	40.53
NavQ [71]	3.06	73	63	3.30	72	63	60.47	53.22	38.89	60.39	53.29	39.50
COSMO [73]	3.15	73	61	3.43	71	58	56.09	50.81	35.93	59.33	52.53	36.12
DUET [11]	3.31	72	60	3.65	69	59	51.07	46.98	33.73	56.91	52.51	36.06
DUET-NavGRPO	3.18	74	63	3.39	71	62	53.17	49.47	35.32	58.32	54.49	38.16
ScaleVLN [68]	2.34	79	70	2.73	77	68	63.85	56.97	41.84	62.65	56.13	39.52
ScaleVLN-NavGRPO	**2.19**	**82**	**73**	**2.52**	**79**	**70**	**65.19**	**58.91**	**43.55**	**64.21**	**58.25**	**41.34**

Table 3. Navigation performance on the R2R-CE dataset. †: Methods that apply candidate waypoint predictor to support high-level action space.

Methods	Val Unseen			Test Unseen		
	NE↓	SR↑	SPL↑	NE↓	SR↑	SPL↑
LAW [55]	6.83	35	31	-	-	-
Sim2Sim [31]	6.07	43	36	6.17	44	37
MGMap [9]	6.28	39	34	7.11	35	28
CMA† [25]	6.20	41	36	6.30	38	33
VLN⟳BERT† [25]	5.74	44	39	5.89	42	36
GridMM† [69]	5.11	49	41	5.64	46	39
Ego^2-Map†	4.94	52	46	5.54	47	41
Reborn [1]	5.40	50	46	5.55	49	45
ETPNAV [3]	4.71	57	49	5.12	55	48
COSMO† [73]	-	47	40	-	47	40
DUET† [11]	5.26	47	39	5.82	42	36
DUET-NavGRPO†	5.02	50	42	5.64	44	38
ScaleVLN†	4.80	55	51	5.11	55	50
ScaleVLN-NavGRPO†	**4.69**	**57**	**53**	**5.01**	**57**	**52**

training. We evaluate whether RL exploration, by experiencing diverse state-action distributions, enables the agent to better handle stochastic perturbations at inference time.

Experimental Setup. We design two perturbation strategies to evaluate robustness: (1) *Global perturbation* mimics real-world random noise—at each step with probability $p \in \{0.0, 0.2, 0.4, 0.8\}$, the agent samples an action from its policy distribution $\pi(\cdot|s)$; otherwise it takes the argmax action. (2) *Early perturbation* tests recovery from initial mistakes—the first $N \in \{1, 2, 3\}$ steps select the *least probable* action from the policy distribution, while remaining steps use argmax actions. This simulates scenarios where agents start from poor decisions due to uncertain states, such as imprecise localization, but must recover to reach the goal.

Results and Analysis. Table 1 demonstrates superior robustness of our method across both perturbation scenarios. Under global perturbation at $p = 0.4$, our method degrades by only 0.96 SPL compared to the baseline's 4.07 degradation, representing 4.2× better robustness. At $p = 0.8$, this advantage amplifies to 3.0× with degradations of 2.67 versus 8.03 SPL. Notably, our method maintains 69.98 SPL at $p = 0.8$, exceeding the baseline's unperturbed performance of 69.97. For early perturbation, perturbing the first three steps causes our method to degrade by 9.99 SPL while the baseline degrades by 22.20 SPL, yielding a 2.2× robustness advantage. These results indicate that GRPO training enables effective recovery from execution perturbations through exposure to diverse trajectories.

4.3. Main Results

R2R. Table 2 shows results on the R2R dataset. Our method consistently outperforms DUET, achieving 2% higher SR and 3% higher SPL on both val unseen and test unseen splits. When applied to ScaleVLN, our approach achieves 3% SPL improvement over the baseline and outperforms GOAT by 5% on val unseen.

Table 4. Ablation study on reward function design. Experiments are conducted on R2R val unseen split.

Method #	Reward Components			R2R Val Unseen		
	R_{nav}	R_{path}	R_{step}	NE↓	SR↑	SPL↑
1	✓			2.31	80.14	71.07
2	✓	✓		2.26	81.02	71.88
3	✓		✓	2.25	81.29	71.93
4	✓	✓	✓	**2.19**	**81.88**	**72.65**

Table 5. Comparison of group-based policy optimization variants on R2R Val Unseen split. All methods use the same reward function and sampling strategy.

Method	OSR↑	NE↓	SR↑	SPL↑
w/o Group	87.48	2.34	79.40	69.97
GRPO [20]	88.15	2.26	80.15	71.23
GSPO [77]	88.62	2.23	80.52	71.68
GMPO [76]	88.56	2.24	80.47	71.62
Dr.GRPO [49]	**89.02**	**2.19**	**81.88**	**72.65**

REVERIE. On the REVERIE dataset, which features high-level goal-oriented instructions requiring longer-horizon planning, our approach outperforms DUET by 2.49% in SR and 1.59% in SPL on val unseen. When built upon ScaleVLN, our method further improves SR by 1.94% and SPL by 1.71%, surpassing GOAT by 6.48% in SPL. The gains are validated on test unseen, where we achieve 1.98% higher SR and 2.10% higher SPL compared to DUET. Notably, our approach substantially outperforms traditional RL-based methods such as RCM and HAMT, which face challenges from step-level sparse rewards and credit assignment in long-horizon navigation. The consistent improvements across both datasets demonstrate generalization to diverse navigation scenarios, from fine-grained step-by-step instructions to high-level goal-oriented tasks.

R2R-CE. Table 3 presents results on the continuous R2R-CE benchmark. Despite training in discrete environments, both DUET-GRPO and ScaleVLN-GRPO show consistent improvements over their respective baselines, demonstrating effective transfer to continuous navigation scenarios.

4.4. Ablation Studies and Analysis

Impact of Reward Function Design. To understand the contribution of different reward components, we conduct ablation studies on the R2R val unseen split, as shown in Table 4. Using only the navigation success reward provides a basic foundation with 71.07% SPL, demonstrating that sparse trajectory-level signals alone can guide policy learning. Incorporating the path efficiency reward improves SPL to 71.88%, indicating that penalizing unnecessarily long trajectories encourages more compact navigation behaviors. Alternatively, adding the step-level progress reward achieves 71.93% SPL, showing that fine-grained intermediate guidance helps the agent make better local decisions. Combining all three components yields the best performance at 72.65% SPL and 81.88% SR, demonstrating that trajectory-level and step-level rewards provide complementary supervision—the former guides overall navigation quality while the latter refines individual action selections.

Table 6. Analysis of sampling trajectory number on R2R Val Unseen split.

Trajectories	OSR↑	NE↓	SR↑	SPL↑
$K = 2$	88.28	2.31	79.98	71.02
$K = 4$	88.62	2.28	80.37	71.98
$K = 8$	89.02	2.19	81.88	72.65
$K = 16$	**89.12**	**2.15**	**81.96**	**72.87**

Impact of Group-Based Policy Optimization. We compare different variants of group-based policy optimization to validate our design choices, as shown in Table 5. Without grouping, the agent achieves 69.97% SPL on val unseen, establishing a baseline for individual trajectory optimization. Standard GRPO [20] introduces group-wise advantage normalization, improving performance to 71.23% SPL through better calibrated gradients. We evaluate three advanced variants adapted from recent LLM alignment literature: GSPO [77] shifts importance sampling and clipping to the sequence level, GMPO [76] adopts geometric mean for step-level reward aggregation, and Dr.GRPO [49] removes length and variance normalization to mitigate length bias. Dr.GRPO achieves the strongest performance with 72.65% SPL on val unseen. We adopt Dr.GRPO's debiased advantage estimation in our framework, which removes normalization constraints and reduces hyperparameter sensitivity. This design choice reduces hyperparameter sensitivity and proves effective for generalizing across different VLN benchmarks and base models. The consistent gains across different optimization variants confirm the robustness of this approach.

Analysis of Sampling Trajectory Number. We investigate the impact of sampling trajectory number K during training, where K trajectories are sampled for each instruction to form a group for relative advantage computation. Table 6 shows the results on R2R val unseen split with different values of K. Using fewer trajectories provides limited diversity for relative comparison, resulting in suboptimal performance. Increasing K to 8 substantially improves both SR and SPL by 1.51% and 1.63% respectively compared to $K = 4$, as the agent benefits from richer comparative signals within each group. Further increasing K to 16 yields only marginal gains of 0.08% in SR and 0.22% in SPL while doubling the training time and memory cost. Therefore, we adopt $K = 8$ as our default setting to balance performance and computational efficiency.

Comparison with Alternative RL Methods. We apply

Table 7. Comparison with alternative RL methods on R2R Val Unseen split.

Method	OSR↑	NE↓	SR↑	SPL↑
SFT only	87.48	2.34	79.40	69.97
SFT+REINFORCE	87.35	2.36	79.28	69.85
SFT+A2C	87.72	2.32	79.65	70.25
SFT+PPO	88.15	2.29	79.95	70.68
SFT+GRPO	**89.02**	**2.19**	**81.88**	**72.65**

Table 8. Analysis of different training strategies on R2R Val Unseen split.

Training Strategy	OSR↑	NE↓	SR↑	SPL↑
SFT only	87.48	2.34	79.40	69.97
RL only	74.52	4.30	64.30	59.82
Sequential SFT-RL	88.95	2.22	81.25	72.08
Ours	**89.02**	**2.19**	**81.88**	**72.65**

different RL algorithms to further optimize the IL-finetuned model in Table 7 under a unified training framework. Our GRPO uses trajectory-level rewards measuring navigation success and path efficiency, which cannot be decomposed into step-level signals required for A2C and PPO's value bootstrapping. REINFORCE, though compatible with both reward types, still fails to improve performance due to its inherently high gradient variance. Therefore, following established practices [10], we provide classical methods with step-level rewards including distance progress and orientation alignment. REINFORCE degrades performance slightly, while A2C achieves modest gains of 0.28% SPL, and PPO improves the SPL metric to 70.68%. In contrast, our GRPO achieves 72.65% SPL, outperforming PPO by a margin of 1.97%. This result demonstrates that comparing trajectories within instruction groups yields substantially more informative learning signals than optimizing trajectories independently.

Analysis of Training Strategy. Table 8 compares different training strategies to understand the contribution of each component. All methods start from the same pretrained vision-language model. Training with RL alone performs poorly without navigation-specific initialization. SFT-only training establishes a solid baseline by learning from expert demonstrations. Sequential SFT-RL improves performance to 81.25% SR and 72.08% SPL, demonstrating that RL optimization can surpass IL-finetuned models. Our approach with hard case replay further enhances results to 81.88% SR and 72.65% SPL, providing modest but consistent gains. Importantly, hard case replay is triggered only when the entire sampled group fails, thereby avoiding redundant RL exploration on persistently challenging instructions. This adaptive strategy stabilizes training and prevents catastrophic forgetting, balancing broad policy exploration with targeted supervision.

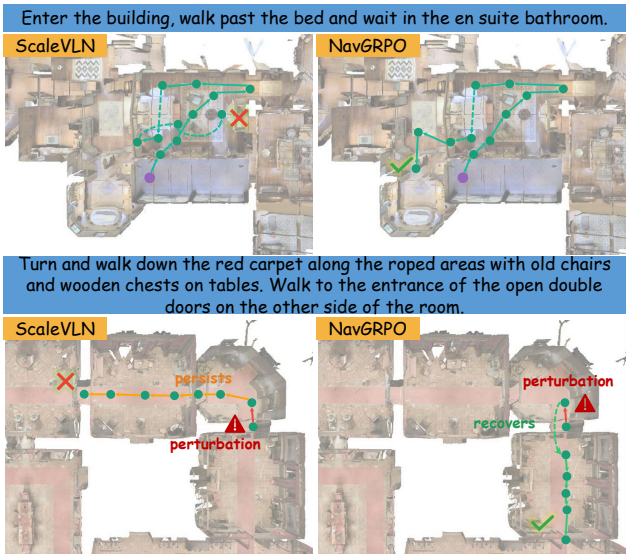

Figure 3. Qualitative comparison on challenging instructions under normal conditions (top) and initial perturbations (bottom). ScaleVLN fails to recover from errors in both scenarios. Our GRPO-trained agent successfully completes tasks and demonstrates robust error correction under perturbations.

4.5. Qualitative Analysis

Figure 3 presents qualitative comparisons on spatially ambiguous instructions. Under standard conditions, ScaleVLN commits navigation errors from which it cannot recover, while our GRPO-trained agent makes correct decisions at critical waypoints to reach target locations. Under adversarial perturbations, ScaleVLN persists along erroneous trajectories initiated by the perturbation. In contrast, our method demonstrates error-correction capabilities: in the first case, it recognizes spatial deviation and backtracks to the correct path; in the second case, it adjusts mid-trajectory despite initial disruption. This demonstrates that group-based trajectory optimization enables both improved spatial reasoning and robust recovery from navigational errors.

5. Conclusion

We present NavGRPO, a reinforcement learning framework for VLN that learns from diverse trajectories through Group Relative Policy Optimization. By comparing complete navigation rollouts, our method achieves stable policy updates without additional value networks. Experiments on several benchmarks show consistent improvements over imitation learning baselines, with substantial gains under perturbations, demonstrating that goal-directed RL training builds more robust navigation policies.

Acknowledgement

This work is supported by the National Natural Science Foundation of China under Grant No.U21B2048 and No.62576224, the Shenzhen Key Technical Projects under Grant CJGJZD20220517141605013, JCYJ20220818101406014, JSGG20220831105801004, and Guangdong Provincial Key Laboratory of Computility Microelectronics with Grant 2024B1212010007.

References

[1] Dong An, Zun Wang, Yangguang Li, Yi Wang, Yicong Hong, Yan Huang, Liang Wang, and Jing Shao. 1st place solutions for rxr-habitat vision-and-language navigation competition (cvpr 2022). *arXiv preprint arXiv:2206.11610*, 2022. 6

[2] Dong An, Yuankai Qi, Yangguang Li, Yan Huang, Liang Wang, Tieniu Tan, and Jing Shao. Bevbert: Multimodal map pre-training for language-guided navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2737–2748, 2023. 6

[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 Transactions on Pattern Analysis and Machine Intelligence*, 2024. 6

[4] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton van den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3674–3683, 2018. 1, 2, 5

[5] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niebner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. In *2017 International Conference on 3D Vision (3DV)*, pages 667–676. IEEE, 2017. 2

[6] Jinyu Chen, Chen Gao, Erli Meng, Qiong Zhang, and Si Liu. Reinforced structured state-evolution for vision-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15450–15459, 2022. 2, 3, 6

[7] Jiaqi Chen, Bingqian Lin, Ran Xu, Zhenhua Chai, Xiaodan Liang, and Kwan-Yee K Wong. Mapgpt: Map-guided prompting with adaptive path planning for vision-and-language navigation. *arXiv preprint arXiv:2401.07314*, 2024. 3

[8] Kevin Chen, Junshen K Chen, Jo Chuang, Marynel Vázquez, and Silvio Savarese. Topological planning with transformers for vision-and-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11276–11286, 2021. 2

[9] Peihao Chen, Dongyu Ji, Kunyang Lin, Runhao Zeng, Thomas H Li, Mingkui Tan, and Chuang Gan. Weakly-supervised multi-granularity map learning for vision-and-language navigation. *arXiv preprint arXiv:2210.07506*, 2022. 6

[10] Shizhe Chen, Pierre-Louis Guhur, Cordelia Schmid, and Ivan Laptev. History aware multimodal transformer for vision-and-language navigation. *Advances in Neural Information Processing Systems*, 34:5834–5847, 2021. 2, 3, 6, 8

[11] Shizhe Chen, Pierre-Louis Guhur, Makarand Tapaswi, Cordelia Schmid, and Ivan Laptev. Think global, act local: Dual-scale graph transformer for vision-and-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16537–16547, 2022. 1, 2, 3, 5, 6

[12] Wenhao Cheng, Xingping Dong, Salman Khan, and Jianbing Shen. Learning disentanglement with decoupled labels for vision-language navigation. In *European Conference on Computer Vision*, pages 309–329. Springer, 2022. 3

[13] Yibo Cui, Liang Xie, Yakun Zhang, Meishan Zhang, Ye Yan, and Erwei Yin. Grounded entity-landmark adaptive pre-training for vision-and-language navigation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12043–12053, 2023. 2

[14] Zhiwei Deng, Karthik Narasimhan, and Olga Russakovsky. Evolving graphical planner: Contextual global planning for vision-and-language navigation. *Advances in Neural Information Processing Systems*, 33:20660–20672, 2020. 2

[15] Songlin Dong, Yihong Gong, Jingang Shi, Miao Shang, Xiaoyu Tao, Xing Wei, Xiaopeng Hong, and Tiangang Zhou. Brain cognition-inspired dual-pathway cnn architecture for image classification. *IEEE Transactions on Neural Networks and Learning Systems*, 35(7):9900–9914, 2023. 3

[16] Songlin Dong, Yingjie Chen, Yuhang He, Yuhan Jin, Alex C Kot, and Yihong Gong. Analogical augmentation and significance analysis for online task-free continual learning. *IEEE Transactions on Multimedia*, 27:3370–3382, 2025. 3

[17] Daniel Fried, Ronghang Hu, Volkan Cirik, Anna Rohrbach, Jacob Andreas, Louis-Philippe Morency, Taylor Berg-Kirkpatrick, Kate Saenko, Dan Klein, and Trevor Darrell. Speaker-follower models for vision-and-language navigation. In *Advances in Neural Information Processing Systems*, pages 3314–3325, 2018. 2, 3

[18] Tsu-Jui Fu, Xin Eric Wang, Matthew F Peterson, Scott T Grafton, Miguel P Eckstein, and William Yang Wang. Counterfactual vision-and-language navigation via adversarial path sampler. In *European Conference on Computer Vision*, pages 71–86. Springer, 2020. 2

[19] Chen Gao, Xingyu Peng, Mi Yan, He Wang, Lirong Yang, Haibing Ren, Hongsheng Li, and Si Liu. Adaptive zone-aware hierarchical planner for vision-language navigation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14911–14920, 2023. 3

[20] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025. 1, 3, 7

[21] Weituo Hao, Chunyuan Li, Xiujun Li, Lawrence Carin, and Jianfeng Gao. Towards learning a generic agent for vision-

and-language navigation via pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13137–13146, 2020. 2

[22] Weituo Hao, Chunyuan Li, Xiujun Li, Lawrence Carin, and Jianfeng Gao. Towards learning a generic agent for vision-and-language navigation via pre-training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13137–13146, 2020. 2

[23] Keji He, Chenyang Si, Zhihe Lu, Yan Huang, Liang Wang, and Xinchao Wang. Frequency-enhanced data augmentation for vision-and-language navigation. *Advances in neural information processing systems*, 36:4351–4364, 2023. 3

[24] Yicong Hong, Qi Wu, Yuankai Qi, Cristian Rodriguez-Opazo, and Stephen Gould. A recurrent vision-and-language bert for navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1643–1653, 2021. 2, 6

[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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15439–15449, 2022. 6

[26] Haoshuo Huang, Vihan Jain, Harsh Mehta, Alexander Ku, Gabriel Magalhaes, Jason Baldridge, and Eugene Ie. Transferable representation learning in vision-and-language navigation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7404–7413, 2019. 2

[27] Jingyang Huo, Qiang Sun, Boyan Jiang, Haitao Lin, and Yanwei Fu. Geovln: Learning geometry-enhanced visual representation with slot attention for vision-and-language navigation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 23212–23221, 2023. 3

[28] Minyoung Hwang, Jaeyeon Jeong, Minsoo Kim, Yoonseon Oh, and Songhwai Oh. Meta-explore: Exploratory hierarchical vision-and-language navigation using scene object spectrum grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6683–6693, 2023. 3

[29] Jing Yu Koh, Harsh Agrawal, Dhruv Batra, Richard Tucker, Austin Waters, Honglak Lee, Yinfei Yang, Jason Baldridge, and Peter Anderson. Simple and effective synthesis of indoor 3d scenes. *arXiv preprint arXiv:2204.02960*, 2022. 3

[30] Xianghao Kong, Jinyu Chen, Wenguan Wang, Hang Su, Xiaolin Hu, Yi Yang, and Si Liu. Controllable navigation instruction generation with chain of thought prompting. *arXiv preprint arXiv:2407.07433*, 2024. 3

[31] Jacob Krantz and Stefan Lee. Sim-2-sim transfer for vision-and-language navigation in continuous environments. In *European Conference on Computer Vision*, pages 588–603. Springer, 2022. 6

[32] Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII 16*, pages 104–120. Springer, 2020. 5

[33] Alexander Ku, Peter Anderson, Roma Patel, Eugene Ie, and Jason Baldridge. Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4392–4412, 2020. 2

[34] Shuhei Kurita and Kyunghyun Cho. Generative language-grounded policy in vision-and-language navigation with bayes' rule. *arXiv preprint arXiv:2009.07783*, 2020. 2

[35] Jialu Li and Mohit Bansal. Panogen: Text-conditioned panoramic environment generation for vision-and-language navigation. *arXiv preprint arXiv:2305.19195*, 2023. 3

[36] Jialu Li, Hao Tan, and Mohit Bansal. Improving cross-modal alignment in vision language navigation via syntactic information. *arXiv preprint arXiv:2104.09580*, 2021. 3

[37] Jialu Li, Hao Tan, and Mohit Bansal. Envedit: Environment editing for vision-and-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15407–15417, 2022. 3

[38] Mingxiao Li, Zehao Wang, Tinne Tuytelaars, and Marie-Francine Moens. Layout-aware dreamer for embodied visual referring expression grounding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1386–1395, 2023. 3

[39] Xiujun Li, Chunyuan Li, Qiaolin Xia, Yonatan Bisk, Asli Celikyilmaz, Jianfeng Gao, Noah Smith, and Yejin Choi. Robust navigation with language pretraining and stochastic sampling. *arXiv preprint arXiv:1909.02244*, 2019. 2

[40] Xiangyang Li, Zihan Wang, Jiahao Yang, Yaowei Wang, and Shuqiang Jiang. Kerm: Knowledge enhanced reasoning for vision-and-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2583–2592, 2023. 3, 6

[41] Xiwen Liang, Fengda Zhu, Yi Zhu, Bingqian Lin, Bing Wang, and Xiaodan Liang. Contrastive instruction-trajectory learning for vision-language navigation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1592–1600, 2022. 2

[42] Bingqian Lin, Yi Zhu, Yanxin Long, Xiaodan Liang, Qixiang Ye, and Liang Lin. Adversarial reinforced instruction attacker for robust vision-language navigation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10): 7175–7189, 2021. 2

[43] Bingqian Lin, Yi Zhu, Zicong Chen, Xiwen Liang, Jianzhuang Liu, and Xiaodan Liang. Adapt: Vision-language navigation with modality-aligned action prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15396–15406, 2022. 3

[44] Bingqian Lin, Yi Zhu, Xiaodan Liang, Liang Lin, and Jianzhuang Liu. Actional atomic-concept learning for demystifying vision-language navigation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1568–1576, 2023. 3

[45] Chuang Lin, Yi Jiang, Jianfei Cai, Lizhen Qu, Gholamreza Haffari, and Zehuan Yuan. Multimodal transformer with variable-length memory for vision-and-language navigation. In *Computer Vision–ECCV 2022: 17th European Confer-*

ence, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXVI, pages 380–397. Springer, 2022. 2

[46] Kunyang Lin, Peihao Chen, Diwei Huang, Thomas H Li, Mingkui Tan, and Chuang Gan. Learning vision-and-language navigation from youtube videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8317–8326, 2023. 3

[47] Chong Liu, Fengda Zhu, Xiaojun Chang, Xiaodan Liang, Zongyuan Ge, and Yi-Dong Shen. Vision-language navigation with random environmental mixup. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1644–1654, 2021. 3

[48] Rui Liu, Wenguan Wang, and Yi Yang. Volumetric environment representation for vision-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16317–16328, 2024. 6

[49] Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*, 2025. 3, 7

[50] Chih-Yao Ma, Jiasen Lu, Zuxuan Wu, Ghassan AlRegib, Zsolt Kira, Richard Socher, and Caiming Xiong. Self-monitoring navigation agent via auxiliary progress estimation. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019. 2

[51] Abhinav Moudgil, Arjun Majumdar, Harsh Agrawal, Stefan Lee, and Dhruv Batra. Soat: A scene-and object-aware transformer for vision-and-language navigation. *Advances in Neural Information Processing Systems*, 34:7357–7367, 2021. 3

[52] Amin Parvaneh, Ehsan Abbasnejad, Damien Teney, Javen Qinfeng Shi, and Anton Van den Hengel. Counterfactual vision-and-language navigation: Unravelling the unseen. *Advances in neural information processing systems*, 33:5296–5307, 2020. 2

[53] Yuankai Qi, Qi Wu, Peter Anderson, Xin Wang, William Yang Wang, Chunhua Shen, and Anton van den Hengel. Reverie: Remote embodied visual referring expression in real indoor environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9982–9991, 2020. 2, 5

[54] Yanyuan Qiao, Yuankai Qi, Yicong Hong, Zheng Yu, Peng Wang, and Qi Wu. Hop+: History-enhanced and order-aware pre-training for vision-and-language navigation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 6

[55] 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. *arXiv preprint arXiv:2109.15207*, 2021. 6

[56] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied ai research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9339–9347, 2019. 5

[57] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 4, 5

[58] Cong Wan, Zeyu Guo, Jiangyang Li, SongLin Dong, Yifan Bai, Lin Peng, Zhiheng Ma, and Yihong Gong. Remot: Reinforcement learning with motion contrast triplets. *arXiv preprint arXiv:2603.00461*, 2026. 1

[59] Hu Wang, Qi Wu, and Chunhua Shen. Soft expert reward learning for vision-and-language navigation. In *European Conference on Computer Vision*, pages 126–141. Springer, 2020. 2, 3, 6

[60] Hanqing Wang, Wei Liang, Jianbing Shen, Luc Van Gool, and Wenguan Wang. Counterfactual cycle-consistent learning for instruction following and generation in vision-language navigation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15471–15481, 2022. 2

[61] Liuyi Wang, Zongtao He, Jiagui Tang, Ronghao Dang, Naijia Wang, Chengju Liu, and Qijun Chen. A dual semantic-aware recurrent global-adaptive network for vision-and-language navigation. *arXiv preprint arXiv:2305.03602*, 2023. 2

[62] Liuyi Wang, Zongtao He, Ronghao Dang, Mengjiao Shen, Chengju Liu, and Qijun Chen. Vision-and-language navigation via causal learning. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2, 6

[63] Liuyi Wang, Zongtao He, Mengjiao Shen, Jingwei Yang, Chengju Liu, and Qijun Chen. Magic: Meta-ability guided interactive chain-of-distillation for effective-and-efficient vision-and-language navigation. *arXiv preprint arXiv:2406.17960*, 2024. 6

[64] Xin Wang, Wenhan Xiong, Hongmin Wang, and William Yang Wang. Look before you leap: Bridging model-free and model-based reinforcement learning for planned-ahead vision-and-language navigation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 37–53, 2018. 2

[65] Xin Wang, Qiuyuan Huang, Asli Celikyilmaz, Jianfeng Gao, Dinghan Shen, Yuan-Fang Wang, William Yang Wang, and Lei Zhang. Reinforced cross-modal matching and self-supervised imitation learning for vision-language navigation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6629–6638, 2019. 2, 3, 6

[66] Xiaohan Wang, Wenguan Wang, Jiayi Shao, and Yi Yang. Lana: A language-capable navigator for instruction following and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19048–19058, 2023. 6

[67] Xin Eric Wang, Vihan Jain, Eugene Ie, William Yang Wang, Zornitsa Kozareva, and Sujith Ravi. Environment-agnostic multitask learning for natural language grounded navigation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIV 16*, pages 413–430. Springer, 2020. 2

[68] Zun Wang, Jialu Li, Yicong Hong, Yi Wang, Qi Wu, Mohit Bansal, Stephen Gould, Hao Tan, and Yu Qiao. Scaling data generation in vision-and-language navigation. In

Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pages 12009–12020, 2023. 3, 5, 6

[69] Zihan Wang, Xiangyang Li, Jiahao Yang, Yeqi Liu, and Shuqiang Jiang. Gridmm: Grid memory map for vision-and-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15625–15636, 2023. 2, 6

[70] Zun Wang, Jialu Li, Yicong Hong, Songze Li, Kunchang Li, Shoubin Yu, Yi Wang, Yu Qiao, Yali Wang, Mohit Bansal, et al. Bootstrapping language-guided navigation learning with self-refining data flywheel. *arXiv preprint arXiv:2412.08467*, 2024. 3

[71] Peiran Xu, Xicheng Gong, and Yadong Mu. Navq: Learning a q-model for foresighted vision-and-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6327–6341, 2025. 6

[72] Jiwen Zhang, Jianqing Fan, Jiajie Peng, et al. Curriculum learning for vision-and-language navigation. *Advances in Neural Information Processing Systems*, 34:13328–13339, 2021. 2

[73] Siqi Zhang, Yanyuan Qiao, Qunbo Wang, Zike Yan, Qi Wu, Zhihua Wei, and Jing Liu. Cosmo: Combination of selective memorization for low-cost vision-and-language navigation. *arXiv preprint arXiv:2503.24065*, 2025. 6

[74] Weixia Zhang, Chao Ma, Qi Wu, and Xiaokang Yang. Language-guided navigation via cross-modal grounding and alternate adversarial learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(9):3469–3481, 2020. 2

[75] Yue Zhang and Parisa Kordjamshidi. Vln-trans: Translator for the vision and language navigation agent. *arXiv preprint arXiv:2302.09230*, 2023. 3

[76] Yuzhong Zhao, Yue Liu, Junpeng Liu, Jingye Chen, Xun Wu, Yaru Hao, Tengchao Lv, Shaohan Huang, Lei Cui, Qixiang Ye, et al. Geometric-mean policy optimization. *arXiv preprint arXiv:2507.20673*, 2025. 7

[77] Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, et al. Group sequence policy optimization. *arXiv preprint arXiv:2507.18071*, 2025. 7

[78] Duo Zheng, Shijia Huang, Lin Zhao, Yiwu Zhong, and Liwei Wang. Towards learning a generalist model for embodied navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13624–13634, 2024. 3, 6

[79] Gengze Zhou, Yicong Hong, and Qi Wu. Navgpt: Explicit reasoning in vision-and-language navigation with large language models. *arXiv preprint arXiv:2305.16986*, 2023.

[80] Gengze Zhou, Yicong Hong, Zun Wang, Xin Eric Wang, and Qi Wu. Navgpt-2: Unleashing navigational reasoning capability for large vision-language models. In *European Conference on Computer Vision*, pages 260–278, 2024. 3, 6

[81] Fengda Zhu, Yi Zhu, Xiaojun Chang, and Xiaodan Liang. Vision-language navigation with self-supervised auxiliary reasoning tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10012–10022, 2020. 2

[82] Fengda Zhu, Xiwen Liang, Yi Zhu, Qizhi Yu, Xiaojun Chang, and Xiaodan Liang. Soon: Scenario oriented object navigation with graph-based exploration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12689–12699, 2021. 2