What Limits Vision-and-Language Navigation ?

Yunheng Wang[1], Yuetong Fang[1], Taowen Wang[1], Lusong Li[2], Kun Liu[2], Junzhe Xu[1,2], Zizhao Yuan[1], Yixiao Feng[1], Jiaxi Zhang[1], Wei Lu[1], Zecui Zeng[2,†], Renjing Xu[1,†]

[1]HKUST(GZ) [2]JD Explore Academy

Abstract

Vision-and-Language Navigation (VLN) is a cornerstone of embodied intelligence. However, current agents often suffer from significant performance degradation when transitioning from simulation to real-world deployment, primarily due to perceptual instability (e.g., lighting variations and motion blur) and under-specified instructions. While existing methods attempt to bridge this gap by scaling up model size and training data, we argue that the bottleneck lies in the lack of robust spatial grounding and cross-domain priors. In this paper, we propose StereoNav, a robust Vision-Language-Action framework designed to enhance real-world navigation consistency. To address the inherent gap between synthetic training and physical execution, we introduce Target-Location Priors as a persistent bridge. These priors provide stable visual guidance that remains invariant across domains, effectively grounding the agent even when instructions are vague. Furthermore, to mitigate visual disturbances like motion blur and illumination shifts, StereoNav leverages stereo vision to construct a unified representation of semantics and geometry, enabling precise action prediction through enhanced depth awareness. Extensive experiments on R2R-CE and RxR-CE demonstrate that StereoNav achieves state-of-the-art egocentric RGB performance, with SR and SPL scores of 81.1% and 68.3%, and 67.5% and 52.0%, respectively, while using significantly fewer parameters and less training data than prior scaling-based approaches. More importantly, real-world robotic deployments confirm that StereoNav substantially improves navigation reliability in complex, unstructured environments. Project page: Link.

1 Introduction

Vision-and-Language Navigation (VLN) [1] is a cornerstone of embodied intelligence, requiring agents to ground natural language instructions into sequential egocentric observations [2]. Ideally, a robust VLN system should serve as a dependable primitive for advanced downstream tasks, such as mobile manipulation [3] and loco-manipulation [4], where success critically depends on reaching a target with precise spatial alignment. However, despite significant progress in simulation, current VLN agents often exhibit brittle behavior and execution inconsistency when deployed in real-world environments. They struggle to maintain performance amidst the complexities of physical reality, such as varying lighting, camera shake, and unstructured layouts.

A prevailing consensus in the community attributes these failures to insufficient cross-modal understanding, leading to a paradigm of "scaling": upgrading to larger and stronger Vision-Language Model (VLM) backbones [4, 5] and expanding training corpora with auxiliary data [6]. As illustrated in Fig. 1, while these strategies have pushed the state-of-the-art, they are reaching a point of diminishing returns. Models built on comparable backbones show saturated performance, and the correlation between training data scale and success remains surprisingly weak. This suggests that the primary bottleneck may not be "understanding" alone, but rather the lack of robust structural priors that can bridge the gap between abstract instructions and noisy physical perceptions.

Preprint.

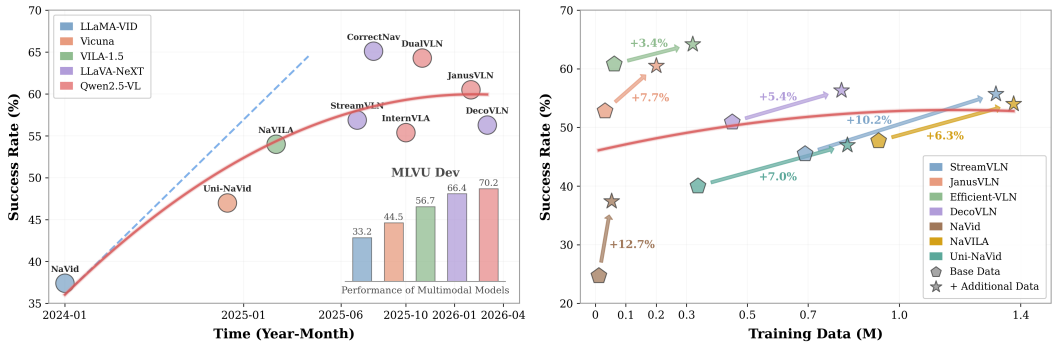

Figure 1: **Performance under different backbones and training data scales.** (a) **Left:** The red trend line shows that success rate increases over time with diminishing returns. The inset reports the MLVU-Dev [7] scores of the adopted backbone VLMs, indicating that stronger backbones improve early performance, but their gains gradually saturate. (b) **Right:** Arrows indicate the performance gains from increased training data for each method, while the red trend line suggests that training data scale has only a weak overall correlation with navigation performance across methods.

In this paper, we argue that real-world performance degradation stems from two neglected factors: environmental perception instability and instructional under-specification. In physical settings, egocentric observations are frequently corrupted by motion blur and illumination shifts, which undermine action prediction in models trained on pristine synthetic data. Furthermore, natural language instructions are inherently sparse, providing insufficient guidance for long-range navigation. While humans often navigate using a mental coarse map alongside visual cues, current agents are forced to infer implicit spatial goals solely from ambiguous text, a task that remains difficult to scale through data alone.

We propose StereoNav, a stereo Vision-Language-Action framework designed for robust embodied navigation. Addressing the challenge of under-specified instructions, we introduce Target-Point Priors that are derived from the coarse or precise knowledge typically available in practical scenarios. By rendering these priors as persistent visual cues, StereoNav equips the agent with global information guidance, ensuring the goal remains spatially anchored despite incomplete linguistic descriptions. Seeking to mitigate perceptual instability (e.g., lighting variations and motion blur), our framework moves beyond conventional monocular perception by incorporating stereo observations. This approach enables unified semantic, structural, and geometric modeling, allowing the agent to maintain stable perception and reliable decision-making through joint action and depth prediction. Our contributions are threefold:

- We identify perceptual instability (e.g., lighting and motion blur) and instruction under-specification as key factors that hinder the Sim-to-Real transfer of VLN agents.
- We propose StereoNav, which integrates target-point-prior-guided navigation to provide persistent global guidance and stereo-based geometric modeling to ensure robust environmental perception.
- We demonstrate that StereoNav achieves state-of-the-art performance on competitive benchmarks while exhibiting significantly higher deployment reliability and execution consistency in real-world robotic tasks compared to monocular alternatives.

2 Related Work

2.1 Vision-and-Language Navigation

Vision-and-Language Navigation (VLN) studies how an embodied agent executes natural-language instructions within complex, visually grounded environments [1]. Early research focused on discrete graph-based formulations [8, 9], which have since evolved toward more realistic continuous settings [10, 11]. A predominant trend in the field emphasizes enhancing the agent's cross-modal understanding as the primary path to performance gains. Consequently, recent state-of-the-art methods have leveraged increasingly powerful pre-trained backbones [12–15], massive-scale navigation and auxiliary datasets [4, 6, 16, 17], and sophisticated temporal modeling [18–21]. While these advancements have significantly pushed the performance frontier, it is increasingly recognized that

understanding ability is not the sole determinant of success in VLN. Recent studies have begun to highlight critical external factors that pose significant challenges to even the most capable agents. Specifically, natural-language instructions often contain inherent ambiguities [22, 23], and agents frequently exhibit sensitivity to visual disturbances or environmental shifts [24, 25]. Despite their importance, these factors—instruction ambiguity and visual robustness—are often treated as anecdotal observations rather than being systematically analyzed. Their individual and joint impacts on navigation performance remain insufficiently quantified. Motivated by this, we conduct analysis studies to show how these factors constrain current VLN agents, aiming to provide a more nuanced understanding of the remaining bottlenecks in the field.

2.2 Vision-Language-Action Models for VLN

Recent advances in large-scale Vision-Language Models (VLMs) [26–30] have inspired a growing body of work that adapts them to Vision-and-Language Navigation. A representative direction is to fine-tune pre-trained VLMs to directly map observations and language instructions to low-level navigation actions in an end-to-end manner, forming Vision-Language-Action models for VLN [4–6, 17, 18, 31]. Benefiting from the broad visual-semantic priors learned from large-scale internet data, this paradigm provides a promising route for policy learning in complex environments [32]. Early efforts such as NaVid [5] demonstrated the feasibility of monocular RGB-based end-to-end navigation, while later methods such as StreamVLN [6] further improved performance with stronger foundation models and larger-scale training data. However, existing methods still suffer from degraded navigation performance and reduced execution consistency in realistic deployment settings In particular, their sensitivity to visual disturbances and instruction ambiguity often results in unstable cross-scene transfer and unsatisfactory accuracy, even in relatively controlled settings [1, 2]. Motivated by these limitations, we introduce StereoNav, which leverages clear or fuzzy target-point priors to mitigate instruction ambiguity, while combining stereo-based unified understanding and joint prediction modeling to improve robustness under visual perturbations and enhance navigation accuracy.

3 Revisiting the Limitations of VLN

To understand the performance degradation in physical VLN deployments, we identify two primary bottlenecks: perception instability (3.1) and instructional under-specification (3.2). We analyze their impact below to provide the empirical motivation for our design.

3.1 The Impact of Visual Uncertainty

While VLN methods excel in in-domain evaluations, their performance often degrades in real-world deployments due to perception instability. As shown in Figure 2, this instability commonly manifests as: lighting perturbation, height fluctuation, motion blur, and viewpoint oscillation [33, 34]. We categorize the former two as mild perturbations and the latter as severe disturbances. Through pilot

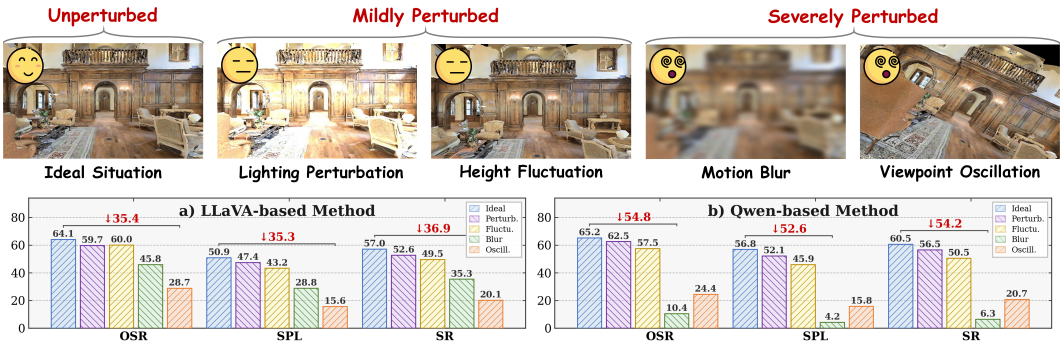

Figure 2: **Impact of visual uncertainty on VLN agents.** (a) **Top:** Visual examples of four common perturbations during embodied navigation. (b) **Bottom:** Performance degradation of representative open-source VLN methods, where the LLaVA-based and Qwen-based methods correspond to StreamVLN [6] and JanusVLN [18], respectively. Although existing agents perform competitively in the ideal setting, their navigation performance degrades markedly under severe visual perturbations.

studies (see Appendix A.1), we systematically examine how these factors constrain agent performance to motivate our design.

As shown in Fig. 2, current VLN agents exhibit a significant lack of robustness to perception instability. Under mild perturbations, including lighting changes and height fluctuations, the degradation remains relatively contained, performance degradation remains relatively contained, with key metrics like Success Rate (SR) and Success weighted by Path Length (SPL) typically dropping by around 10% compared to the ideal situation. However, severe disturbances lead to a dramatic performance collapse. For the LLaVA-based method, SR plummets from 57.0% to 35.3% under motion blur, and further to 20.1% under viewpoint oscillation. The Qwen-based method proves even more vulnerable; notably, its SPL drops from 56.8% to a mere 4.2% under motion blur—less than 8% of its original performance. In real-world deployments, changes in visual perception due to environmental variations (such as lighting disturbances) are very common. However, existing VLN models are extremely sensitive to such changes, necessitating a solution to ensure reliable practical applications.

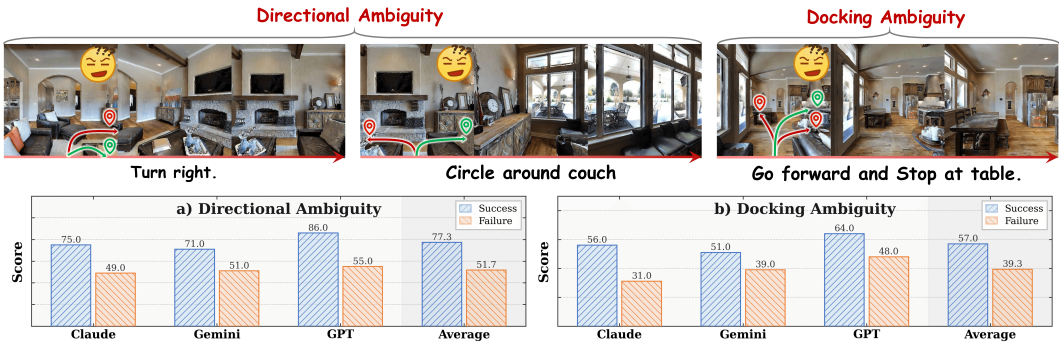

Figure 3: **Impact of instructional under-specification on VLN agents.** (a) **Top:** Representative cases of Directional Ambiguity, where under-specified route or orientation cues permit multiple feasible paths, and Docking Ambiguity, where vague goal descriptions permit multiple plausible stopping targets. (b) **Bottom:** Distributions of ambiguity scores across representative VLMs, where lower scores indicate stronger ambiguity effects.

3.2 The Impact of Instruction Ambiguity

Human-provided instructions serve as the primary guidance for VLN, yet their inherent instructional under-specification often creates a challenging one-to-many mapping between a single utterance and multiple plausible navigation paths. As shown in Figure 3, insufficient instruction description manifests as two different types of decision uncertainty: *Directional Ambiguity*, which complicates path selection during intermediate navigation phases; and *Docking Ambiguity*, which disrupts target localization at termination. Our cross-model analysis reveals a strong correlation between the degree of under-specification and navigation failure, identifying these ambiguities as fundamental failure modes (detailed in Appendix A.2).

As shown in Fig. 3, instruction ambiguity is strongly associated with navigation failure, where lower scores indicate a more severe influence from the corresponding ambiguity type. Specifically, failed cases are more strongly affected by directional ambiguity than successful ones, with average scores of 51.7 and 77.3, respectively, indicating that under-specified path or orientation cues can easily mislead the agent onto an unintended route, thereby causing task failure. Docking ambiguity is also evident. Failed cases score 17.7 points lower than successful ones. This indicates that docking ambiguity is widespread in VLN instructions, while failed cases are often affected by more severe ambiguity around the stopping location or target object. Overall, these results show that instruction ambiguity directly increases the likelihood of navigation failure.

4 Methodology

In this section, we first define the task in Sec. 4.1, and then introduce *StereoNav*. The proposed method alleviates instruction under-specification by incorporating precise or coarse target-point priors (Sec. 4.2), while leveraging stereo-based unified understanding modeling (Sec. 4.3) and joint

prediction modeling (Sec. 4.4) to improve robustness against visual disturbances and achieve more accurate navigation.

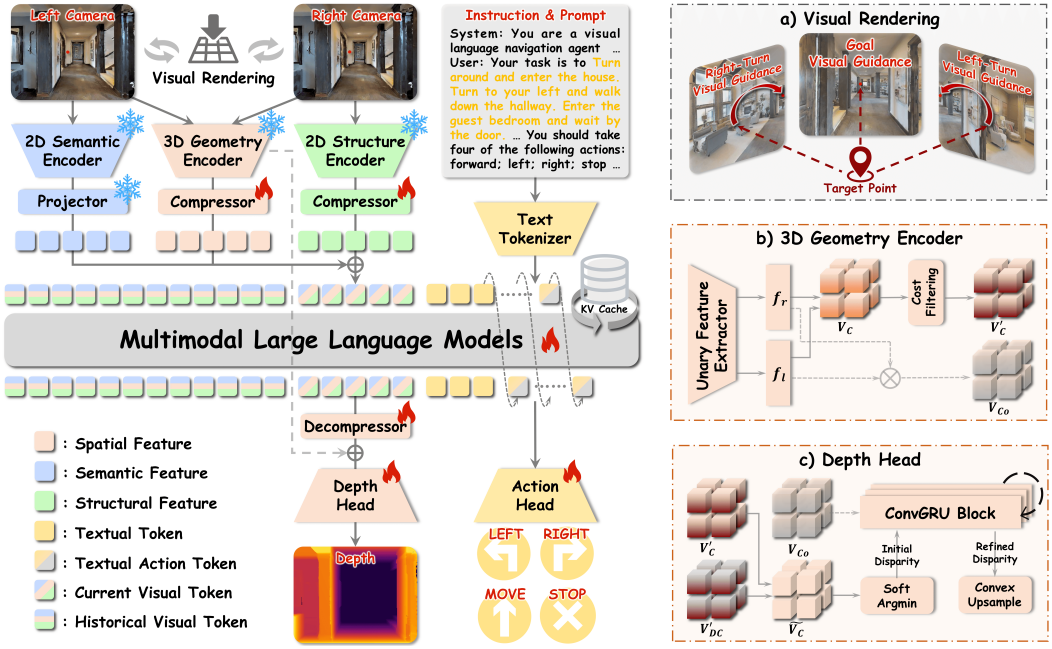

Figure 4: **Overview of StereoNav.** StereoNav takes stereo RGB observations, a navigation instruction, and a target-location prior as input. The target prior is rendered as persistent visual guidance, while stereo observations are encoded into unified semantic, structural, and geometric tokens through 2D semantic, 2D structural, and 3D geometry encoders. These tokens are then processed by the MLLM for joint action and depth prediction. The right panels illustrate detailed designs of selected modules.

4.1 Problem Formulation

We consider Vision-and-Language Navigation in Continuous 3D Environments. In our setting, the agent is given a natural-language instruction I and a fuzzy or precise target-location cue p for global guidance. At each time step t, it receives stereo RGB observations o_t^l and o_t^r, together with the history of past stereo observations $\mathcal{H}_{t-1} = \{(o_1^l, o_1^r), \ldots, (o_{t-1}^l, o_{t-1}^r)\}$. Based on these inputs, the agent jointly predicts a navigation action a_t and a dense depth estimate $\hat{d}_t$:

$$(a_t, \hat{d}_t) = \pi(o_t^l, o_t^r, I, p, \mathcal{H}_{t-1}), \quad t = 1, 2, \ldots, T. \tag{1}$$

The action space is $\mathcal{A} = \{\texttt{move forward}, \texttt{turn left}, \texttt{turn right}, \texttt{stop here}\}$, corresponding to a forward translation of 0.25 m, 15° left/right rotations, and episode termination, respectively. Action prediction is the primary objective, while depth prediction serves as an auxiliary geometric objective for improving spatial perception, robustness, and navigation accuracy.

4.2 Visual Rendering

Existing VLN agents rely heavily on natural-language instructions [4–6, 17], which are inherently prone to instructional under-specification (Sec. 3) and often lead to navigation failure. We observe that in practical human–robot interaction, speakers typically possess varying degrees of prior knowledge regarding the goal's location. Motivated by this, we explicitly leverage such spatial priors. Unlike conventional point-goal methods [35, 36] that treat target coordinates as symbolic inputs, we project the spatial prior directly into the agent's egocentric visual space, transforming abstract coordinates into intuitive visual guidance that facilitates more effective decision-making.

As illustrated in Fig. 4a, the agent is initialized with both a language instruction and a target-location prior (ranging from fuzzy to precise). This prior is dynamically updated based on the current pose and rendered onto the stereo observations: visible targets are marked with a red point, while out-of-view targets are indicated by boundary-aware cues. This design provides a lightweight yet persistent

global directional signal, grounding the agent's target awareness while preserving its capacity for fine-grained local action prediction. By bridging the gap left by ambiguous instructions, this mechanism mitigates both Directional and Docking Ambiguity, thereby enhancing navigation robustness.

4.3 Unified Understanding Modeling

Although existing end-to-end VLN methods [5, 6, 16] commonly inherit the visual encoder of the underlying MLLM, these representations are mainly optimized for cross-modal semantic alignment rather than the structural and geometric perception required for embodied navigation. This makes them vulnerable to visual perturbations and can undermine reliable navigation. To address this limitation, we propose Unified Understanding Modeling, which leverages stereo observations to jointly capture semantic, structural, and geometric cues. By exploiting their complementarity and binocular redundancy, our design improves perceptual stability and supports more robust navigation.

To achieve Unified Understanding Modeling, we construct a multimodal 2D–3D representation that synergizes semantic, structural, and geometric cues. For 2D features, we employ InternViT to maintain semantic grounding from large-scale pretraining and DINOv2 to capture fine-grained structural patterns like layouts and contours. Applying these encoders to stereo views inherently preserves binocular disparity, enhancing the agent's spatial discrimination. To incorporate explicit depth priors, we introduce a 3D geometry encoder inspired by FoundationStereo (Fig. 4b). This module processes stereo pairs into a 4D cost volume V_c, which is refined through attentive hybrid filtering (V_c') and tokenized for fusion. The resulting unified tokens T_t are formulated as:

$$T_t = \alpha P(F_t^{\text{sem}}) + \beta C(F_t^{\text{str}}) + \gamma G(V'_{c,t}), \tag{2}$$

where F_t^{sem} and F_t^{str} represent the semantic and structural features, while $P(\cdot)$, $C(\cdot)$, and $G(\cdot)$ denote their respective projection and tokenization modules. α, β, and γ are learnable fusion weights. This integration yields a compact yet comprehensive representation, ensuring robust perception for embodied navigation.

4.4 Joint Prediction Modeling

Previous studies suggest that explicit depth guidance [37–39] and auxiliary future observation prediction [19, 21, 40] can improve navigation and representation learning. However, these designs remain costly and only partially effective for robust embodied navigation: depth sensing increases hardware dependency and is vulnerable to environmental interference, while future prediction introduces extra computation and mainly provides indirect supervision. To address these limitations, we propose joint action-depth prediction, where depth prediction serves as geometric supervision for implicit 3D perception and further supports more robust action decisions.

Concretely, we instantiate this objective with two lightweight prediction heads: an action head for future action generation and a depth head for stereo disparity estimation. For action prediction, we follow prior VLM-based navigation methods [5, 6]. At each time step, the model autoregressively generates the textualized action sequence for the next four steps, conditioned on the system prompt, historical observations, the current observation, and the navigation instruction. The detailed prompt template used for action generation is provided in Appendix B.2:

$$x = \underbrace{[S_{\text{sys}}]\,\tilde{s}\,[E_{\text{sys}}]\,[S_{\text{usr}}]\,\tilde{o}_{1:t-1},\ \tilde{o}_t,\ \tilde{l}\,[E_{\text{usr}}]}_{\text{Input}}\ \underbrace{[S_{\text{asst}}]\,\tilde{a}_{t:t+3}\,[E_{\text{asst}}]}_{\text{Generation}}, \tag{3}$$

where S_* and E_* denote the segment boundary tokens, and $\tilde{a}_{t:t+3}$ denotes the generated four-step action sequence. Meanwhile, the depth head adopts a coarse-to-fine stereo matching strategy, as shown in Fig. 4 c). It takes the filtered stereo cost volume V_C' from the 3D Geometry Encoder and the decoded cost volume V_{DC}' reconstructed from current-view tokens, and estimates the final disparity through cost-volume fusion, Soft Argmin initialization, recurrent refinement, and convex upsampling:

$$\hat{d} = \text{Upsample}\big(\text{ConvGRU}\big(\sum_{\delta=0}^{D/4-1} \delta \cdot \text{Softmax}(V_C' + V_{DC}')(\delta), V_{co}\big)\big), \tag{4}$$

Here, V_{co} denotes the geometry-aware contextual feature from the 3D Geometry Encoder, which guides ConvGRU-based disparity refinement.

Figure 5: **Qualitative examples of StereoNav in real-world.** From top to bottom: Outdoor, Office, Lobby, and Gym. The results demonstrate StereoNav's reliability across diverse scenes. Note that these examples are visualized from a third-person perspective; details regarding the actual sensor inputs used for navigation are provided in Section C.2.

5 Experiment

We evaluate StereoNav from multiple perspectives. We first present the experimental setup in Sec. 5.1; then report comprehensive comparisons in simulated and real-world environments in Sec. 5.2; analyze robustness and reliability in Sec. 5.3; and provide ablation studies in Sec. 5.4.

5.1 Experimental Details

Implementation Details. StereoNav is built upon InternVL-3.5-2B [52]. The 2D semantic encoder is initialized from the InternViT visual encoder of InternVL-3.5, while the 2D structure encoder is initialized from DINOv2 [53]; both the 3D geometry encoder and the depth head are initialized from FoundationStereo [54]. During training, stereo images are resized to 448×448, and 8 historical frames are uniformly sampled as visual context. StereoNav autoregressively predicts the next 4 navigation actions and is trained following a two-stage strategy, with the 2D semantic, 2D structure, and 3D geometry branches fused using weights of 1.0, 0.6, and 0.2, respectively. Additional implementation details are provided in Appendix B.3.

Evaluation Benchmarks. In simulated environments, we evaluate StereoNav in the Habitat simulator [11] on Matterport3D [9] scenes, following the Val-Unseen of R2R-CE and RxR-CE. Following standard practice [5], we report Navigation Error (NE), Success Rate (SR), Oracle Success Rate (OSR) and Success weighted by Path Length (SPL), which together measure goal attainment, trajectory efficiency, and path fidelity. In real-world environments, we use Success Rate as the primary metric and evaluate StereoNav on a Unitree G1 robot equipped with a ZED Mini stereo camera across four representative navigation scenarios: Office, Gym, Lobby, and Outdoor. A trial is considered successful if the robot stops within 1.5 meters of the goal.

5.2 Comprehensive Performance Evaluation

We comprehensively evaluate StereoNav in both simulated and real-world environments. In simulation, we compare it with existing methods on standard VLN-CE benchmarks under established

Table 1: **Comparison with SOTA methods on the R2R/RxR-CE Val-Unseen.** Light and dark blue rows indicate StereoNav trained with standard navigation data only and with additional external data, respectively. Top-three results are marked by **bold**, underline, and dotted underline. Parentheses report the signed difference from the best metric value among Egocentric RGB Agent baselines.

Method	Size	Observation			Prediction		R2R-CE Val-Unseen				RXR-CE Val-Unseen		
		Pano.	Depth	RGB	Extra	Action	NE↓	OSR↑	SR↑	SPL↑	NE↓	SR↑	SPL↑
Panoramic RGB-D Agent													
ETPNav [41]	-	✓	✓	✗	✗	✓	4.7	65.0	57.0	49.0	5.6	54.8	44.9
CLASH [42]	-	✓	✓	✗	✗	✓	4.1	73.0	65.0	55.0	-	-	-
D3D-VLP [43]	2B	✓	✓	✗	✗	✓	4.7	67.2	61.3	56.1	-	-	-
ETP-R1 [44]	-	✓	✓	✗	✗	✓	3.9	72.0	65.0	56.0	5.2	59.9	49.0
P^3Nav [45]	-	✓	✓	✗	✗	✓	4.4	69.0	62.0	52.0	5.4	58.0	47.9
Panoramic RGB Agent													
AO-Planner [46]	-	✓	✗	✗	✗	✓	5.6	59.0	47.0	33.0	7.1	43.3	30.5
NavFoM [16]	7B	✓	✗	✗	✗	✓	4.6	72.1	61.7	55.3	4.7	64.4	56.2
ABot-N0 [47]	4B	✓	✗	✗	✗	✓	3.8	70.8	66.4	63.9	**3.8**	69.3	60.0
NavForesee [19]	3B	✓	✗	✗	✓	✓	3.9	78.4	66.2	59.7	4.2	66.3	53.2
SPAN-Nav [21]	-	✓	✗	✗	✓	✓	4.1	75.3	66.3	59.3	4.2	**69.7**	60.1
Egocentric RGB-D Agent													
NaVid-4D [39]	7B	✗	✓	✓	✗	✓	6.0	55.7	43.8	37.1	-	-	-
Dynam3D [37]	7B	✗	✓	✓	✗	✓	5.3	62.1	52.9	45.7	-	-	-
NavMorph [40]	-	✗	✓	✓	✓	✓	5.8	56.9	47.9	33.2	8.9	30.8	22.8
InternVLA-N1 [15]	8B	✗	✓	✓	✗	✓	4.8	63.3	58.2	54.0	5.9	53.5	46.1
AgentVLN [38]	3B	✗	✓	✓	✗	✓	3.9	73.5	67.2	64.7	3.9	69.5	**61.3**
Egocentric RGB Agent													
NaVid [5]	7B	✗	✗	✓	✗	✓	5.5	49.1	37.4	35.9	-	-	-
Uni-NaVid [17]	7B	✗	✗	✓	✗	✓	5.6	53.3	47.0	42.7	6.2	48.7	40.9
NaVILA [4]	8B	✗	✗	✓	✗	✓	5.2	62.5	54.0	49.0	6.8	49.3	44.0
StreamVLN [6]	7B	✗	✗	✓	✗	✓	5.0	64.2	56.9	51.9	6.2	52.9	46.0
InternVLA-N1 [15]	8B	✗	✗	✓	✗	✓	4.9	60.6	55.4	52.1	6.4	49.5	41.8
NavFoM [16]	7B	✗	✗	✓	✗	✓	5.0	64.9	56.2	51.2	5.5	57.4	49.4
DualVLN [12]	8B	✗	✗	✓	✗	✓	4.1	70.7	64.3	58.5	4.6	61.4	51.8
Efficient-VLN [48]	4B	✗	✗	✓	✗	✓	4.2	73.7	64.2	55.9	3.9	67.0	54.3
JanusVLN [18]	8B	✗	✗	✓	✗	✓	4.8	65.2	60.5	56.8	6.1	56.2	47.5
PROSPECT [13]	9B	✗	✗	✓	✓	✓	4.9	65.2	58.9	54.0	5.7	54.6	46.2
SACA [49]	8B	✗	✗	✓	✗	✓	4.2	69.3	64.7	56.9	4.8	62.1	51.7
NaVIDA [50]	3B	✗	✗	✓	✗	✓	4.3	69.5	61.4	54.7	5.2	57.4	49.6
DyGeoVLN [14]	9B	✗	✗	✓	✓	✓	4.4	70.1	60.8	55.8	-	-	-
DecoVLN [51]	7B	✗	✗	✓	✗	✓	5.0	63.5	56.3	50.5	5.7	54.2	46.3
StereoNav	3B	✗	✗	✓	✓	✓	3.0 **(-1.1)**	76.6 **(+2.9)**	72.8 **(+8.1)**	56.4 **(+2.1)**	5.9 **(+2.0)**	58.0 **(-9.0)**	43.5 **(-8.5)**
StereoNav	3B	✗	✗	✓	✓	✓	**2.1 (-2.0)**	**82.4 (+8.7)**	**81.1 (+16.4)**	**68.3 (+9.8)**	4.6 **(+0.7)**	67.5 **(+0.5)**	52.0 **(-2.3)**

protocols. In the real world, we assess its practical navigation capability across representative physical scenarios. Detailed settings are provided in Appendix C.

Evaluation in Simulated Environments. Table 1 compares StereoNav with recent SOTA methods on R2R/RxR-CE Val-Unseen. On R2R-CE, StereoNav achieves the best overall performance across all observation settings with only a lightweight 3B egocentric RGB agent, improving over the strongest prior result by 13.9% in SR and 3.6% in SPL, while reducing NE by 1.7m. Under the same egocentric RGB setting, the gains further increase to 16.4% in SR, 9.8% in SPL, 8.7% in OSR, and 2.0m in NE. Notably, even without external data, StereoNav still achieves SOTA performance among egocentric RGB agents on R2R-CE, improving OSR, SR, and SPL by 2.9%, 8.1%, and 2.1%, respectively. On RxR-CE, while StereoNav does not surpass panoramic or RGB-D agents, it achieves the highest SR of 67.5% within the egocentric RGB setting. These results demonstrate the effectiveness of our visual rendering, stereo-based unified understanding, and joint prediction designs.

Evaluation in Real-world Environments. We further evaluate real-world deployment. Each scenario includes three simple and two complex instructions, and each instruction is repeated three times. As shown in Table 2 (a), StereoNav consistently surpasses both zero-shot VLN methods [55, 56] and supervised egocentric baselines [5, 6, 18]. Macro-averaged over all scenario-difficulty settings, StereoNav achieves a success rate of 60.6%, markedly higher than the strongest supervised baseline StreamVLN (24.3%) and the strongest zero-shot baseline DreamNav (22.1%). More importantly, this advantage persists under complex instructions and more challenging deployment scenarios, such as Lobby and Outdoor, where most baselines degrade to near-zero success. These results demonstrate that StereoNav provides stronger cross-scene generalization and more reliable real-world navigation. Qualitative examples are shown in Fig. 6.

5.3 Robustness and Reliability Evaluation

We further assess StereoNav under deployment-oriented conditions, evaluating its robustness to severe visual disturbances and its ability to terminate accurately near the target. Additional experiments and detailed settings are provided in Appendix D.

Table 2: **Real-world performance and architectural ablation.** (a) **Left:** Success rates of different methods across four real-world scenarios with simple and complex instructions. (b) **Right:** Ablation of architectural designs in StereoNav, reporting navigation performance under model variants.

Method	Office		Gym		Lobby		Outdoor	
	Sim.	Com.	Sim.	Com.	Sim.	Com.	Sim.	Com.
Open-Nav [55]	0.33	0.00	0.22	0.00	0.22	0.00	0.33	0.17
DreamNav [56]	0.44	0.00	0.33	0.17	0.22	0.00	0.44	0.17
NaVid [5]	0.33	0.00	0.33	0.00	0.11	0.00	0.00	0.00
StreamVLN [6]	0.56	0.17	0.44	0.33	0.33	0.00	0.11	0.00
JanusVLN [18]	0.22	0.00	0.11	0.17	0.33	0.00	0.22	0.00
StereoNav	**0.67**	**0.50**	**0.56**	**0.83**	**0.78**	**0.17**	**0.67**	**0.67**

Architecture				Views		R2R-CE	
3D Geo.	2D Stru.	2D Sema.	De. Head	Stereo	Rend.	SR↑	SPL↑
✗	✗	✓	✗	L/-	✗	34.4	29.5
✗	✗	✓	✗	L/-	✓	52.3	46.7
✗	✓	✓	✗	R/R	✓	48.3	37.0
✗	✓	✓	✗	L/R	✓	63.4	51.7
✓	✗	✓	✗	L/R	✓	51.3	46.0
✓	✗	✓	✓	L/R	✓	58.4	50.6
✓	✓	✓	✓	L/R	✓	**72.8**	**56.4**

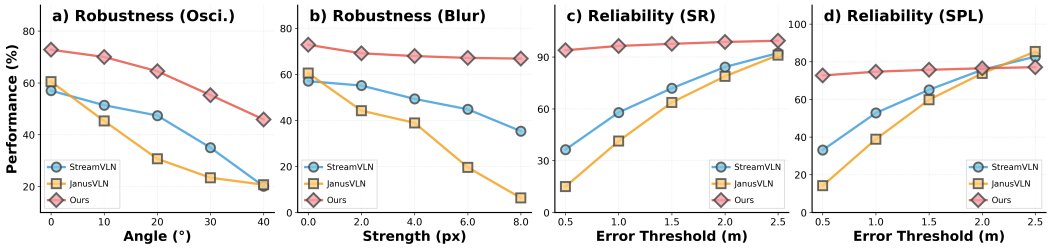

Figure 6: **Robustness and reliability evaluation of StereoNav.** (a–b) Robustness under viewpoint oscillation and motion blur, where StereoNav shows smaller performance degradation as perturbation severity increases. (c–d) Reliability in goal stopping under different stopping-error thresholds, where StereoNav achieves higher SR and SPL, especially under strict target-neighborhood constraints.

As shown in Fig. 6, StereoNav consistently outperforms prior VLN agents in robustness and reliability. Under the strongest viewpoint oscillation, it incurs a 37.0% relative degradation, much lower than StreamVLN (64.7%) and JanusVLN (65.8%); under motion blur, the degradation is only 8.2%, compared with 38.1% and 89.6%. These results show that stereo-based unified understanding and joint action-depth prediction improve perceptual stability under corrupted observations. StereoNav also yields more reliable goal stopping: under a strict 0.5 m threshold, 93.9% of successful episodes remain within the target neighborhood, with 72.7% SPL, substantially surpassing StreamVLN (36.4%/33.1%) and JanusVLN (15.0%/14.2%) in SR/SPL. This confirms that the rendered target prior provides persistent visual guidance for resolving ambiguous routes and stopping decisions.

5.4 Ablation Study

We conduct ablation studies on a subset of the R2R-CE evaluation split to analyze the contribution of each key component in StereoNav. More detailed ablation results are provided in Appendix E.

Ablation Study on Different Model Designs. Table 2 (b) shows that the performance gains of StereoNav arise from the coordinated design of visual guidance, Unified Understanding Modeling, and Joint Prediction Modeling. The rendered target-point prior improves SR/SPL by 17.9%/17.2% over the monocular semantic baseline, confirming that persistent visual guidance helps mitigate instruction ambiguity. With this guidance, Unified Understanding Modeling further improves navigation by integrating semantic, structural, and geometric cues, and the full model achieves the best performance of 72.8% SR and 56.4% SPL. Further comparisons show that stereo-based 2D modeling contributes 15.1%/14.7% gains in SR/SPL, while depth-supervised geometry adds another 7.1%/4.6%, indicating that binocular cues and joint action-depth prediction improve navigation-relevant spatial reasoning.

6 Conclusion

In this work, we revisit the limitations of Vision-and-Language Navigation beyond the conventional focus on stronger model understanding, identifying visual uncertainty and instructional underspecification as two key bottlenecks. We introduce StereoNav, a stereo Vision-Language-Action framework that renders target-location priors as persistent visual guidance, unifies semantic, structural, and geometric cues for robust spatial understanding, and couples action generation with depth prediction for geometry-aware decision making. Evaluations across simulation, real-world deployment, robustness, and reliability demonstrate that StereoNav achieves state-of-the-art performance. However, StereoNav currently relies on explicit or implicit target-oriented input and cannot yet handle cases where such information is absent. Future work will explore multimodal grounding and spatial estimation to infer target-location priors directly from observations and language.

References

[1] Jing Gu, Eliana Stefani, Qi Wu, Jesse Thomason, and Xin Eric Wang. Vision-and-language navigation: A survey of tasks, methods, and future directions. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 7606–7623, 2022.

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

[3] Sriram Yenamandra, Arun Ramachandran, Karmesh Yadav, Austin Wang, Mukul Khanna, Theophile Gervet, Tsung-Yen Yang, Vidhi Jain, Alexander William Clegg, John Turner, Zsolt Kira, Manolis Savva, Angel Chang, Devendra Singh Chaplot, Dhruv Batra, Roozbeh Mottaghi, Yonatan Bisk, and Chris Paxton. Homerobot: Open-vocabulary mobile manipulation. *arXiv preprint arXiv:2306.11565*, 2024.

[4] An-Chieh Cheng, Yandong Ji, Zhaojing Yang, Xueyan Zou, Jan Kautz, Erdem Biyik, Hongxu Yin, Sifei Liu, and Xiaolong Wang. Navila: Legged robot vision-language-action model for navigation. In *Robotics: Science and Systems (RSS)*, 2025.

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

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

[7] Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, Shitao Xiao, Xi Yang, Yongping Xiong, Bo Zhang, Tiejun Huang, and Zheng Liu. Mlvu: A comprehensive benchmark for multi-task long video understanding. *arXiv preprint arXiv:2406.04264*, 2024.

[8] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sunderhauf, Ian Reid, Stephen Gould, and Anton van den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3674–3683, 2018.

[9] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from RGB-D data in indoor environments. In *International Conference on 3D Vision (3DV)*, 2017.

[10] Jacob Krantz, Erik Wijmans, Arjun Majundar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision and language navigation in continuous environments. In *European Conference on Computer Vision (ECCV)*, pages 104–120, 2020.

[11] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A platform for embodied ai research. In *IEEE International Conference on Computer Vision (ICCV)*, pages 9339–9347, 2019.

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

[13] Zehua Fan, Wenqi Lyu, Wenxuan Song, Linge Zhao, Yifei Yang, Xi Wang, Junjie He, Lida Huang, Haiyan Liu, Bingchuan Sun, Guangjun Bao, Xuanyao Mao, Liang Xu, Yan Wang, and Feng Gao. Prospect: Unified streaming vision-language navigation via semantic–spatial fusion and latent predictive representation. *arXiv preprint arXiv:2603.03739*, 2026.

[14] Xiangchen Liu, Hanghan Zheng, Jeil Jeong, Minsung Yoon, Lin Zhao, Zhide Zhong, Haoang Li, and Sung-Eui Yoon. Dygeovln: Infusing dynamic geometry foundation model into vision-language navigation. *arXiv preprint arXiv:2603.21269*, 2026.

[15] InternNav Team. Internvla-n1: An open dual-system navigation foundation model with learned latent plans. 2025.

[16] Jiazhao Zhang, Anqi Li, Yunpeng Qi, Minghan Li, Jiahang Liu, Shaoan Wang, Haoran Liu, Gengze Zhou, Yuze Wu, Xingxing Li, et al. Embodied navigation foundation model. *arXiv preprint arXiv:2509.12129*, 2025.

[17] Jiazhao Zhang, Kunyu Wang, Shaoan Wang, Minghan Li, Haoran Liu, Songlin Wei, Zhongyuan Wang, Zhizheng Zhang, and He Wang. Uni-navid: A video-based vision-language-action model for unifying embodied navigation tasks. In *Robotics: Science and Systems (RSS)*, 2024.

[18] Shuang Zeng, Dekang Qi, Xinyuan Chang, Feng Xiong, Shichao Xie, Xiaolong Wu, Shiyi Liang, Mu Xu, and Xing Wei. Janusvln: Decoupling semantics and spatiality with dual implicit memory for vision-language navigation. In *International Conference on Learning Representations (ICLR)*, 2026.

[19] Fei Liu, Shichao Xie, Minghua Luo, Zedong Chu, Junjun Hu, Xiaolong Wu, and Mu Xu. Navforesee: A unified vision-language world model for hierarchical planning and dual-horizon navigation prediction. *arXiv preprint arXiv:2512.01550*, 2026.

[20] Junjun Hu, Jintao Chen, Haochen Bai, Minghua Luo, Shichao Xie, Ziyi Chen, Fei Liu, Zedong Chu, Xinda Xue, Botao Ren, Xiaolong Wu, Mu Xu, and Shanghang Zhang. Astranav-world: World model for foresight control and consistency. *arXiv preprint arXiv:2512.21714*, 2025.

[21] Jiahang Liu, Tianyu Xu, Jiawei Chen, Lu Yue, Jiazhao Zhang, Zhiyong Wang, Minghan Li, Qisheng Zhao, Anqi Li, Qi Su, Zhizheng Zhang, and He Wang. Span-nav: Generalized spatial awareness for versatile vision-language navigation. *arXiv preprint arXiv:2603.09163*, 2026.

[22] Qiaolin Xia, Xiujun Li, Chunyuan Li, Yonatan Bisk, Zhifang Sui, Jianfeng Gao, Yejin Choi, and Noah A. Smith. Multi-view learning for vision-and-language navigation. *arXiv preprint arXiv:2003.00857*, 2020.

[23] Haodong Hong, Sen Wang, Zi Huang, Qi Wu, and Jiajun Liu. Why only text: Empowering vision-and-language navigation with multi-modal prompts. *arXiv preprint arXiv:2406.02208*, 2024.

[24] Yubo Zhang, Hao Tan, and Mohit Bansal. Diagnosing the environment bias in vision-and-language navigation. *arXiv preprint arXiv:2005.03086*, 2020.

[25] Liuyi Wang, Xinyuan Xia, Hui Zhao, Hanqing Wang, Tai Wang, Yilun Chen, Chengju Liu, Qijun Chen, and Jiangmiao Pang. Rethinking the embodied gap in vision-and-language navigation: A holistic study of physical and visual disparities. In *IEEE International Conference on Computer Vision (ICCV)*, pages 9455–9465, 2025.

[26] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, et al. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[27] Yanwei Li, Chengyao Wang, and Jiaya Jia. Llama-vid: An image is worth 2 tokens in large language models. In *European Conference on Computer Vision (ECCV)*, pages 323–340, 2024.

[28] Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. Llava-video: Video instruction tuning with synthetic data. *arXiv preprint arXiv:2410.02713*, 2025.

[29] Ji Lin, Hongxu Yin, Wei Ping, Yao Lu, Pavlo Molchanov, Andrew Tao, Huizi Mao, Jan Kautz, Mohammad Shoeybi, and Song Han. Vila: On pre-training for visual language models. *arXiv preprint arXiv:2312.07533*, 2023.

[30] Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. Instruction tuning with gpt-4. *arXiv preprint arXiv:2304.03277*, 2023.

[31] Zhuoyuan Yu, Yuxing Long, Zihan Yang, Chengyan Zeng, Hongwei Fan, Jiyao Zhang, and Hao Dong. Correctnav: Self-correction flywheel empowers vision-language-action navigation model. *arXiv preprint arXiv:2508.10416*, 2025.

[32] Taj Jones-McCormick, Aukosh Jagannath, and Subhabrata Sen. Provable benefits of unsupervised pre-training and transfer learning via single-index models. *arXiv preprint arXiv:2502.16849*, 2025.

[33] Shunxin Wang, Raymond Veldhuis, Christoph Brune, and Nicola Strisciuglio. A survey on the robustness of computer vision models against common corruptions. *arXiv preprint arXiv:2305.06024*, 2024.

[34] Prithvijit Chattopadhyay, Judy Hoffman, Roozbeh Mottaghi, and Ani Kembhavi. Robustnav: Towards benchmarking robustness in embodied navigation. In *IEEE International Conference in Computer Vision (ICCV)*, pages 15691–15700, 2021.

[35] Ajay Sridhar, Dhruv Shah, Catherine Glossop, and Sergey Levine. Nomad: Goal masked diffusion policies for navigation and exploration. *arXiv preprint arXiv:2310.07896*, 2023.

[36] Wenzhe Cai, Jiaqi Peng, Yuqiang Yang, Yujian Zhang, Meng Wei, Hanqing Wang, Yilun Chen, Tai Wang, and Jiangmiao Pang. Navdp: Learning sim-to-real navigation diffusion policy with privileged information guidance. *arXiv preprint arXiv:2505.08712*, 2025.

[37] Zihan Wang, Seungjun Lee, and Gim Hee Lee. Dynam3d: Dynamic layered 3d tokens empower vlm for vision-and-language navigation. *arXiv preprint arXiv:2505.11383*, 2025.

[38] Zihao Xin, Wentong Li, Yixuan Jiang, Ziyuan Huang, Bin Wang, Piji Li, Jianke Zhu, Jie Qin, and Sheng-Jun Huang. Agentvln: Towards agentic vision-and-language navigation. *arXiv preprint arXiv:2603.17670*, 2026.

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

[40] Junyu Gao Xuan Yao and Changsheng Xu. Navmorph: A self-evolving world model for vision-and-language navigation in continuous environments. In *IEEE International Conference in Computer Vision (ICCV)*, pages 5536–5546, 2025.

[41] 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. In *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2024.

[42] Liuyi Wang, Zongtao He, Jinlong Li, Ruihao Xia, Mengxian Hu, Chenpeng Yao, Chengju Liu, Yang Tang, and Qijun Chen. Clash: Collaborative large-small hierarchical framework for continuous vision-and-language navigation. *arXiv preprint arXiv:2512.10360*, 2025.

[43] Zihan Wang, Seungjun Lee, Guangzhao Dai, and Gim Hee Lee. D3d-vlp: Dynamic 3d vision-language-planning model for embodied grounding and navigation. *arXiv preprint arXiv:2512.12622*, 2025.

[44] Shuhao Ye, Sitong Mao, Yuxiang Cui, Xuan Yu, Shichao Zhai, Wen Chen, Shunbo Zhou, Rong Xiong, and Yue Wang. Etp-r1: Evolving topological planning with reinforcement fine-tuning for vision-language navigation in continuous environments. *arXiv preprint arXiv:2512.20940*, 2025.

[45] Tianfu Li, Wenbo Chen, Haoxuan Xu, Xinhu Zheng, and Haoang Li. P^3nav: End-to-end perception, prediction and planning for vision-and-language navigation. *arXiv preprint arXiv:2603.17459*, 2026.

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

[47] Zedong Chu, Shichao Xie, Xiaolong Wu, Yanfen Shen, Minghua Luo, Zhengbo Wang, Fei Liu, Xiaoxu Leng, Junjun Hu, Mingyang Yin, Jia Lu, Yingnan Guo, Kai Yang, Jiawei Han, Xu Chen, et al. Abot-n0: Technical report on the vla foundation model for versatile embodied navigation. *arXiv preprint arXiv:2602.11598*, 2026.

[48] Duo Zheng, Shijia Huang, Yanyang Li, and Liwei Wang. Efficient-vln: A training-efficient vision-language navigation model. *arXiv preprint arXiv:2512.10310*, 2025.

[49] Haoyuan Li, Rui Liu, Hehe Fan, and Yi Yang. Let's reward step-by-step: Step-aware contrastive alignment for vision-language navigation in continuous environments. *arXiv preprint arXiv:2603.09740*, 2026.

[50] Weiye Zhu, Zekai Zhang, Xiangchen Wang, Hewei Pan, Teng Wang, Tiantian Geng, Rongtao Xu, and Feng Zheng. Navida: Vision-language navigation with inverse dynamics augmentation. *arXiv preprint arXiv:2601.18188*, 2026.

[51] Zihao Xin, Wentong Li, Yixuan Jiang, Bin Wang, Runmin Cong, Jie Qin, and Shengjun Huang. Decovln: Decoupling observation, reasoning, and correction for vision-and-language navigation. *arXiv preprint arXiv:2603.13133*, 2026.

[52] Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, Zhaokai Wang, Zhe Chen, Hongjie Zhang, Ganlin Yang, Haomin Wang, et al. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

[53] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2024.

[54] Bowen Wen, Matthew Trepte, Joseph Aribido, Jan Kautz, Orazio Gallo, and Stan Birchfield. Foundationstereo: Zero-shot stereo matching. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5249–5260, 2025.

[55] Yanyuan Qiao, Wenqi Lyu, Hui Wang, Zixu Wang, Zerui Li, Yuan Zhang, Mingkui Tan, and Qi Wu. Open-nav: Exploring zero-shot vision-and-language navigation in continuous environment with open-source llms. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 6710–6717, 2025.

[56] Yunheng Wang, Yuetong Fang, Taowen Wang, Yixiao Feng, Yawen Tan, Shuning Zhang, Peiran Liu, Yiding Ji, and Renjing Xu. Dreamnav: A trajectory-based imaginative framework for zero-shot vision-and-language navigation. *arXiv preprint arXiv:2509.11197*, 2025.

[57] Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, Akshay Nathan, Alan Luo, Alec Helyar, Aleksander Madry, Aleksandr Efremov, et al. Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*, 2025.

[58] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, Luke Marris, Sam Petulla, Colin Gaffney, Asaf Aharoni, Nathan Lintz, et al. Gemini 2.5: Pushing the frontier with advanced reasoning. *arXiv preprint arXiv:2507.06261*, 2025.

[59] Anthropic. System card: Claude opus 4 and claude sonnet 4. `https://www.anthropic.com/claude-4-system-card`.

A Details of the Pilot Studies

A.1 Experimental Setup for Visual Uncertainty

To examine whether the performance degradation of current VLN agents is associated with visual uncertainty, we conduct controlled pilot studies on two representative open-source VLM-based navigation methods. Specifically, we choose StreamVLN [6] as the LLaVA-based method [28], which adopts LLaVA-NeXT as its backbone, and JanusVLN [18] as the Qwen-based method, which is built upon Qwen2.5-VL [26]. These two methods represent recent VLN agents based on widely used multimodal backbones, allowing us to evaluate whether visual disturbances consistently affect different VLM-based navigation pipelines. To isolate the effect of visual uncertainty, we keep the navigation instructions, evaluation episodes, action space, and evaluation metrics unchanged, and only modify the visual observations received by the agent during inference.

We manually reproduce four common forms of visual uncertainty in embodied navigation: lighting perturbation, height fluctuation, motion blur, and viewpoint oscillation. Lighting perturbation is implemented by post-processing the RGB observations with a brightness scaling factor of -0.8, which darkens the image by approximately 80% and simulates under-exposed visual inputs. Height fluctuation is implemented by increasing the height of the RGB camera sensors by 0.6 m, thereby changing the egocentric viewpoint while keeping the navigation task unchanged. Motion blur is applied before image resizing with a blur strength of 8.0, simulating image degradation caused by robot motion or camera shake. Viewpoint oscillation is implemented by temporarily perturbing the camera orientation in the simulator and alternately rotating the egocentric view to the left and right by 40° at consecutive steps. These perturbation levels are intentionally set as challenging stress-test conditions, so that the resulting performance changes can directly reflect the robustness of existing VLN agents under deployment-oriented visual disturbances.

A.2 Experimental Setup for Instruction Ambiguity

To analyze the relationship between instruction ambiguity and navigation performance, we construct a diagnostic set from the R2R-CE Val-Unseen split using StreamVLN [6] rollouts. Specifically, we randomly sample 100 successful episodes and 100 failed episodes. For each episode, we extract the original navigation instruction and the visual observations collected along the expert trajectory, so that ambiguity can be assessed with respect to the intended route rather than the agent's executed

You are evaluating ambiguity in a Vision-Language Navigation (VLN) task.

Input:
- A human-written navigation instruction.
- A set of uniformly sampled observation frames from the ground-truth trajectory of a successful navigation episode.

Objective:
Your goal is to assess whether the task itself contains instruction ambiguity, not whether the trajectory is correct or successful. Even though the provided episode is a ground-truth successful example, the instruction may still be ambiguous.

Ambiguity types:
1. **Directional Ambiguity**
   This occurs if, at some intermediate stage of navigation, the instruction does not uniquely specify the intended path or orientation, such that multiple plausible route choices, branches, turns, or movement directions could reasonably satisfy the instruction.
2. **Docking Ambiguity**
   This occurs if, near the stopping stage, the instruction does not uniquely specify the final stopping location or target object, such that multiple nearby endpoints or candidate targets could reasonably satisfy the instruction.

Evaluation criteria:
- Judge ambiguity based on the instruction together with the observed scene context.
- Evaluate whether a reasonable agent could form more than one plausible interpretation.
- Do not judge based on quality, policy quality, or trajectory correctness.
- Do not assume that a successful ground-truth trajectory means the instruction is unambiguous.
- Do not treat sparse observations as "lacking evidence"; instead, evaluate ambiguity with respect to the instruction and scene context.
- Mark ambiguity only when there is a genuine and reasonable alternative interpretation supported by the instruction and scene context.
- Directional Ambiguity concerns ambiguity during movement before the final stopping stage.
- Docking Ambiguity concerns ambiguity about the final stopping place or target at the end.
- The two types can both exist in the same episode.

Instruction:
{instruction}

Respond with exactly two integers separated by a comma:
- First integer: 1 if Directional Ambiguity exists, 0 otherwise
- Second integer: 1 if Docking Ambiguity exists, 0 otherwise

Example output:
1,0

Figure 7: **Prompt template for instruction ambiguity assessment.** Given a navigation instruction and sampled observation frames, the evaluator identifies whether Directional Ambiguity or Docking Ambiguity exists and returns binary labels for the two ambiguity types.

trajectory. To reduce temporal redundancy while preserving the main route context, we uniformly sample 10 frames from each observation sequence and pair them with the corresponding instruction. The sampled frames and instruction are then organized into a unified evaluation prompt, as shown in Fig. 7, where the evaluator is asked to determine whether the instruction is affected by Directional Ambiguity or Docking Ambiguity.

We perform the ambiguity assessment with three strong multimodal models, including GPT-5 [57], Gemini-2.5-Pro [58], and Claude-Opus-4.7 [59]. Each model independently judges the presence of the two ambiguity types for every sampled episode. For each ambiguity type and each outcome group, we first compute the percentage of episodes judged as being affected by that ambiguity, denoted as r. We then define the ambiguity score as $100 - r$, where a lower score indicates that a larger proportion of episodes is affected by the corresponding ambiguity type. The final results are reported for each evaluator and further averaged across the three models, providing a more robust estimate of how directional and docking ambiguity correlate with navigation success and failure.

B Details of the StereoNav

B.1 Visual Rendering

We provide the implementation details of target-prior rendering in Algorithm 1. At each time step, StereoNav first transforms the target prior from the world frame to each stereo camera frame using the current agent pose and the corresponding camera offset. The target is then projected onto the image plane with the camera intrinsics. We define a valid image region and a relaxed boundary region to handle near-boundary projections. If the projected point is valid and falls within the relaxed region, it is clipped to the image range and rendered as the target cue. Otherwise, the cue is placed at the center of the left or right image boundary according to the horizontal direction of the target in the camera frame. This procedure is applied to both stereo views, producing target-aware observations that are consistent across data generation and evaluation.

Algorithm 1: Visual Rendering

Input : Stereo observations (o_t^l, o_t^r), target-location prior p, agent pose $\mathcal{P}_t = (\mathbf{x}_t, \mathbf{q}_t)$, camera intrinsics K, image size (W, H), stereo camera offsets (Δ^l, Δ^r), and relaxation factor ϵ.

Output : Rendered stereo observations $(\tilde{o}_t^l, \tilde{o}_t^r)$.

Define 1: $\Omega = [0, W-1] \times [0, H-1]$, the valid image domain.
Define 2: $\Omega_\epsilon = [-\epsilon W, (1+\epsilon)W - 1] \times [-\epsilon H, (1+\epsilon)H - 1]$, the relaxed projection domain.

foreach *view* $v \in \{l, r\}$ **do**
  Compute the camera center: $\mathbf{c}_t^v = \mathbf{x}_t + R(\mathbf{q}_t)\Delta^v$
  Transform the target prior into the camera coordinate frame: $\mathbf{s}_t^v = (X_t^v, Y_t^v, Z_t^v) = R(\mathbf{q}_t)^\top (p - \mathbf{c}_t^v)$
  if $Z_t^v > 0$ **then**
    Project the p onto the image plane: $\boldsymbol{\pi}_t^v = \Pi_K(\mathbf{s}_t^v)$
  else
    Set $\boldsymbol{\pi}_t^v = \varnothing$
  end
  if $\boldsymbol{\pi}_t^v = \varnothing$ *and* $\boldsymbol{\pi}_t^v \in \Omega_\epsilon$ **then**
    Clip the projection to the valid image domain: $\boldsymbol{\rho}_t^v \leftarrow \mathrm{clip}(\boldsymbol{\pi}_t^v, \Omega)$
  else
    if $X_t^v < 0$ **then**
      $\boldsymbol{\rho}_t^v \leftarrow (0, H/2)$
    else
      $\boldsymbol{\rho}_t^v \leftarrow (W-1, H/2)$
    end
  end
  Render a red circular marker centered at $\boldsymbol{\rho}_t^v$ on o_t^v to obtain $\tilde{o}_t^v$
end
return $(\tilde{o}_t^l, \tilde{o}_t^r)$

StereoNav Prompt

system

You are a visual language navigation model. Your goal is to reach the target location and complete the given task. Compare the observations with the instruction to infer your current progress, then choose the most appropriate action to move toward the target and finish the task.

user

{history_video}{current_image}

These images are your historical observations and your current observation.

Your task is to: {instruction}

Choose one action from the following candidates:

- move forward
- turn left
- turn right
- stop here

assistant

{action}

Instruction Parsing

system

You are a navigation assistant. Extract structured information from the user's natural language navigation command.

The robot uses a coordinate system where:

- X axis: forward direction (positive = forward)
- Y axis: left direction (positive = left, negative = right)
- Origin: robot's current position

Extract:

1. instruction: a clean English navigation instruction suitable for a VLN model
2. target_x: estimated target X coordinate in meters (forward distance)
3. target_y: estimated target Y coordinate in meters (left offset, negative if right)

Input / Output

User input: {user_input}

Respond with ONLY a JSON object, no markdown: {{"instruction": "... ", "target_x": <float>, "target_y": <float>}}

Figure 8: **Prompt templates for StereoNav model input and real-world instruction preprocessing.** (a) **Top:** The StereoNav model prompt formats observations and instructions as inputs, with the assistant response as the action label. (b) **Bottom:** The real-world preprocessing prompt converts user commands into the structured instruction and target-location prior required by StereoNav.

B.2 Prompt Template

We provide the detailed prompt template used by StereoNav in Fig. 8(a). The prompt follows a conversation-style format and consists of three parts: a system message that defines the navigation role and action-selection objective, a user message that provides historical observations, the current observation, and the navigation instruction, and an assistant response that corresponds to the target action. In this formulation, visual inputs are organized as historical frames and the current frame, while the language instruction specifies the navigation goal. The model is required to select one action from a fixed action space, including moving forward, turning left, turning right, and stopping. This prompt design keeps the navigation interface simple and consistent across training and inference, allowing StereoNav to learn action prediction from multimodal context in a unified textual format.

B.3 Training Details and Configurations

To facilitate reproducibility, we provide the detailed training configurations of StereoNav in Table 3. The training procedure follows a two-stage design. Stage I focuses on establishing the basic target-aware stereo navigation capability using standard VLN-CE training data, where the model learns to associate rendered target priors, stereo observations, and language instructions with navigation actions. Stage II further initializes from the Stage-I checkpoint and incorporates additional trajectory data to improve data coverage and policy robustness. The table summarizes the key implementation choices across the two stages, including the basic training setup, model input and prediction settings, depth-related configuration, and optimization objectives. These details are intended to clarify the training protocol without overloading the main paper with implementation-specific hyperparameters.

C Details of Comprehensive Performance Evaluation

C.1 Simulation Evaluation

We conduct all simulation experiments using Habitat-Lab with Habitat-Sim as the underlying simulator, following the standard VLN-CE evaluation protocol. The agent operates in continuous 3D

Table 3: **Training configurations of StereoNav.** The table summarizes the main settings used in the two-stage training procedure, including training setup, model configuration, and optimization details.

Configuration	Stage I	Stage II
Basic Training Setup		
Precision	bfloat16	bfloat16
Global Batch Size	512	512
Training Schedule	1 epoch	1 epoch
Training Data	R2R, RxR	R2R, RxR, DAgger, ScaleVLN
Sliding Window	24 frames, β=8	24 frames, β=6
Training Cost	176 GPU hours	688 GPU hours
Model Setup		
Input Resolution	448×448	448×448
History Length	8 frames	8 frames
Prediction Horizon	4 actions	4 actions
Maximum Token Length	4096	4096
Depth Configuration	5 m range, 4 iterations	5 m range, 4 iterations
Fusion Weights	1.0, 0.6, 0.2	1.0, 0.6, 0.2
Initialization	InternVL3.5, DINOv2, FoundationStereo	Stage-I checkpoint
Optimization and Loss Setup		
Optimizer	AdamW	AdamW
Gradient Clipping	0.5	0.5
Base Learning Rate	2.0×10^{-5}	2.0×10^{-5}
Depth Learning Rate	2.0×10^{-6}	2.0×10^{-6}
Scheduler	warm up 0.05, min lr 0.9	warm up 0.05, min lr 0.5
Loss Weights	λ_{lang}=1.0, λ_{depth}=0.1	λ_{lang}=1.0, λ_{depth}=0.1

environments while executing a discrete action space, including moving forward, turning left, turning right, and stopping. In the simulator, the agent is represented as a cylindrical collision body with a height of 1.5 m and a radius of 0.1 m. Stereo RGB observations are captured by two pinhole cameras mounted at a height of 1.25 m, with a stereo baseline of 0.1 m. During inference, we adopt deterministic decoding with `temperature`=0.0 and `top_p`=1.0. Each episode terminates when the agent predicts the stop action or reaches the maximum episode length of 500 steps. At the top of Fig. 10, we further present first-person views from a representative virtual environment. We also visualize the corresponding depth estimation results in Fig. 9. Under this unified simulation protocol, we further evaluate StereoNav with controlled target-location deviations to analyze its robustness under fuzzy goal priors.

Table 4: **Performance of StereoNav under controlled target-location deviations on the R2R-CE Val-Unseen split.** The preset value denotes the maximum radius used to randomly sample a perturbed target point around the ground-truth goal, while the actual value reports the average Euclidean deviation of the sampled targets. The first row corresponds to the clean setting, and values in parentheses indicate the absolute performance difference from this setting.

Method	Deviation		R2R-CE Val-Unseen			
	Preset	Actual	NE↓	OSR↑	SR↑	SPL↑
StereoNav	0.0	0.0	3.0	76.6	72.8	56.4
	1.0	0.7	3.2 (↑0.2)	74.7 (↓1.9)	70.3 (↓2.5)	50.7 (↓5.7)
	2.0	1.4	4.0 (↑1.0)	72.4 (↓4.2)	65.4 (↓7.4)	42.2 (↓14.2)
	3.0	2.0	4.9 (↑1.9)	71.5 (↓5.1)	50.6 (↓22.2)	33.0 (↓23.4)

To evaluate StereoNav under fuzzy target-location priors, we conduct controlled target-location deviation experiments on the R2R-CE Val-Unseen split. Given a preset deviation radius, we randomly sample a perturbed target point within the circle centered at the ground-truth goal. The "Preset" column denotes the maximum sampling radius for generating the noisy prior, while "Actual" reports the average Euclidean deviation of the sampled targets from the ground-truth location. The first row represents the clean setting, where the provided target point is perfectly aligned with the true goal. As shown in Table 4, StereoNav remains effective as the target-location prior becomes increasingly imprecise. In the clean setting, it achieves 3.0 NE, 76.6 OSR, 72.8 SR, and 56.4 SPL. With a 1.0 m preset deviation and a 0.7 m actual deviation, the model shows only mild degradation: NE increases

by 0.2 m, while OSR, SR, and SPL decrease by 1.9, 2.5, and 5.7 points, respectively. When the preset deviation increases to 2.0 m, corresponding to a 1.4 m actual deviation, the drops become larger but remain moderate, with OSR, SR, and SPL decreasing by 4.2, 7.4, and 14.2 points. Under the most challenging 3.0 m preset deviation, where the actual deviation reaches 2.0 m, StereoNav still achieves 71.5 OSR, 50.6 SR, and 33.0 SPL. Notably, although SR and SPL drop by 22.2 and 23.4 points, OSR decreases by only 5.1 points, indicating that the agent can still approach the target region in many episodes despite substantial prior noise. These results suggest that the target-location prior serves as complementary spatial guidance rather than an exact coordinate constraint, allowing StereoNav to integrate coarse goal information with egocentric observations, language instructions, and stereo-based spatial understanding for robust navigation under approximate goal cues.

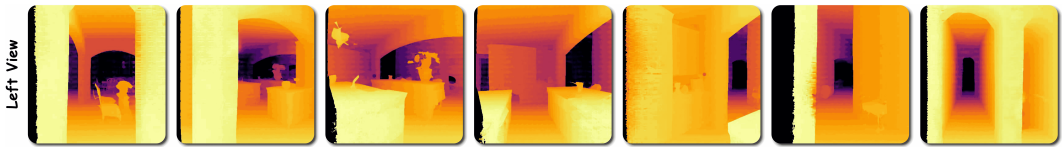

Figure 9: **Visualization of depth estimation results in a representative virtual environment.** StereoNav leverages stereo disparity between the left and right first-person views to produce stable depth estimates, providing reliable geometric cues for robust navigation under approximate goal priors.

Figure 10: **First-person stereo observations in simulation and real-world deployment.** (a) **Top:** First-person left- and right-view observations from a representative virtual environment, where the rendered target-location prior provides persistent visual guidance during navigation. (b) **Bottom:** First-person left- and right-view observations from a representative real-world Gym scenario, demonstrating the deployment of StereoNav on the Unitree G1 robot under physical navigation conditions.

C.2 Real-World Evaluation

We conduct real-world experiments on a Unitree G1 humanoid robot equipped with a ZED Mini stereo camera. The stereo camera is rigidly mounted using the open-source TWIST2 neck firmware and head-mounted structure(`https://yanjieze.com/TWIST2/`), which provides stable egocentric stereo observations during navigation. StereoNav is deployed with a cloud-inference architecture: the G1 robot is directly connected to a local laptop via Ethernet, and the laptop bridges communication

between the robot platform and the remote inference server. At each step, the laptop streams stereo RGB observations, the navigation instruction, and the rendered target prior to the server, and then sends the predicted high-level action back to the robot for execution through the G1 control interface. To support natural human-robot interaction, free-form user commands are first processed by a GPT-based instruction parser, as shown in Fig. 8 (b). The parser converts a raw user command into a clean VLN-style instruction and extracts the corresponding target-location prior required by StereoNav. We further show first-person views from a representative Gym scenario at the bottom of Fig. 10.

D Details of Robustness and Reliability Evaluation

In the main paper, we primarily report robustness results under severe visual disturbances, including motion blur and viewpoint oscillation, as these conditions more directly reflect challenging deployment scenarios that can induce substantial perception instability. Here, we provide complementary results under milder but commonly encountered conditions, including lighting perturbations and camera-height fluctuations. For a conservative comparison, we evaluate the StereoNav model trained only on R2R-CE and RxR-CE, and compare it with StreamVLN and JanusVLN using their full models trained with diverse navigation data. This setting allows us to examine whether the robustness advantage of StereoNav also holds under less severe yet practically relevant observation changes.

Table 5: **Robustness under lighting perturbations and camera-height fluctuations on the R2R-CE Val-Unseen split.** We report Success Rate (SR, %) under lighting perturbations ($-0.8/+0.8$) and camera-height fluctuations ($-0.6/+0.6$ m). Values in parentheses denote the absolute SR drop relative to each method's clean setting.

Method	Perturbation		Fluctuation	
	-0.8	+0.8	-0.6	+0.6
StreamVLN	52.6 (↓4.3)	53.7 (↓3.2)	52.6 (↓4.3)	49.5 (↓7.4)
JanusVLN	56.5 (↓4.0)	57.0 (↓3.5)	49.9 (↓10.6)	50.5 (↓10.0)
StereoNav	69.0 (↓3.8)	69.6 (↓3.2)	69.4 (↓3.4)	67.5 (↓5.3)

As shown in Table 5, lighting perturbations and height fluctuations cause smaller degradation than the severe disturbances reported in the main paper, but clear robustness differences remain across methods. Under lighting perturbations, StreamVLN drops by 4.3 and 3.2 points, while JanusVLN drops by 4.0 and 3.5 points. StereoNav exhibits comparable or smaller degradation, with drops of only 3.8 and 3.2 points, while maintaining higher absolute SR values of 69.0 and 69.6. Under camera-height fluctuations, the advantage of StereoNav becomes more evident. StreamVLN decreases by 4.3 and 7.4 points, and JanusVLN suffers larger drops of 10.6 and 10.0 points, whereas StereoNav decreases by only 3.4 and 5.3 points. Across all four mild settings, StereoNav consistently achieves the highest SR, ranging from 67.5 to 69.6, and shows the smallest or tied-smallest performance degradation. These results indicate that mild lighting and viewpoint-height changes have limited impact on StereoNav, further supporting the effectiveness of stereo-based semantic, structural, and geometric modeling for stable navigation under realistic observation variations.

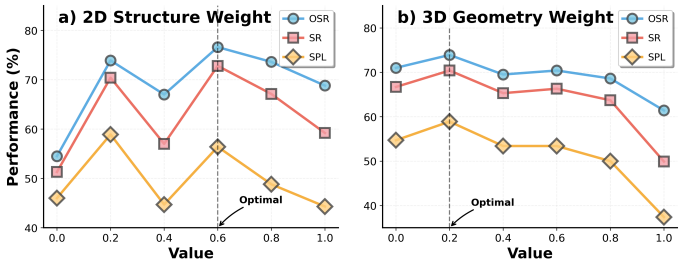

Figure 11: **Ablation study on fusion weights in Unified Understanding Modeling**. The results show that moderate structural guidance and lightweight geometric guidance lead to the best overall performance.

E Additional Ablation Studies

We further study the fusion weights used in Unified Understanding Modeling. As shown in Fig. 11, both the 2D structural branch and the 3D geometric branch are beneficial, but their contributions should be carefully balanced with the language-aligned semantic representation. For the 2D structural branch, a moderate weight of 0.6 yields the best overall performance, indicating that structural cues such as scene layout, object boundaries, and local spatial relations provide strong complementary information for navigation. However, further increasing this weight leads to performance degradation, suggesting that excessive structural emphasis may weaken semantic grounding. For the 3D geometric branch, the best performance is achieved with a smaller weight of 0.2. This shows that geometry is most effective as auxiliary depth-aware guidance, while over-weighting geometric tokens can introduce low-level matching noise and disturb high-level action prediction. Based on these observations, we adopt 1.0, 0.6, and 0.2 as the fusion weights for the semantic, structural, and geometric branches, respectively, which preserves semantic grounding as the dominant signal while incorporating sufficient structural and geometric cues for robust navigation.