Specialist Foundation Models as Tools for Generalist Navigation

Anonymous authors
Paper under double-blind review

Abstract

Generalist navigation requires diverse perceptual and reasoning capabilities to support tasks such as instruction following and object search. Recent generalist navigation models typically learn these capabilities within an end-to-end policy, so adding or improving individual capabilities can require further training of the navigation model. We argue that a navigation decision model need not master every capability itself; it can instead use independently pretrained specialist models as external tools to expand navigation capabilities without retraining the decision model. We present Perceive–Imagine–Verify (PIV), a loosely coupled framework that augments a shared decision model trained across navigation tasks with specialist foundation models as external tools. Perception tools recover missing depth and reconstruct panoramic context, foresight tools generate future observations conditioned on candidate moves, and verification tools ground targets to validate the decision model's stopping proposals. This separation offers two benefits. First, compatible tool backends can be replaced or upgraded while keeping the decision model fixed, providing a path to benefiting from advances in specialist models. Second, perception tools enable one checkpoint to operate across panoramic and forward-facing RGB and RGB-D configurations without sensor-specific fine-tuning. PIV achieves state-of-the-art performance in both vision-and-language navigation and object-goal navigation with one shared decision model. Specifically, it attains success rates of 71.6% on R2R-CE val-unseen and 56.7% on HM3D-OVON val-unseen.

1 Introduction

Generalist embodied navigation aims to support vision-and-language and object-goal navigation across sensing conditions within a shared system (Anderson et al., 2018; Batra et al., 2020; Yokoyama et al., 2024b). It requires understanding the surroundings, anticipating what a move may reveal, and recognizing goal completion despite differences in camera coverage and depth availability. Recent generalist models learn these capabilities within unified perception-to-action policies trained across tasks, enabling knowledge sharing and broader task coverage (Zhang et al., 2024a; Xue et al., 2026; Chu et al., 2026; Zhang et al., 2026b). However, learning individual capabilities within the navigation policy means that adding or improving them can require further training of the navigation model (Figure 1a).

The extended mind thesis proposes that cognitive processes can extend beyond an individual through reliable interactions with external resources (Clark & Chalmers, 1998). Inspired by this view, we argue that a navigation decision model need not master every capability itself; it can instead use independently pretrained specialist models as external tools to expand navigation capabilities without retraining the decision model.

We present Perceive–Imagine–Verify (PIV), a loosely coupled framework that pairs a shared navigation decision model trained across tasks with specialist foundation models as external tools (Figure 1b). Perception, foresight, and verification tools provide complementary information for navigation. Perception tools recover missing depth and reconstruct panoramic context, providing compatible observations across sensing conditions. Foresight tools generate future observations conditioned on candidate moves, allowing the decision model to

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

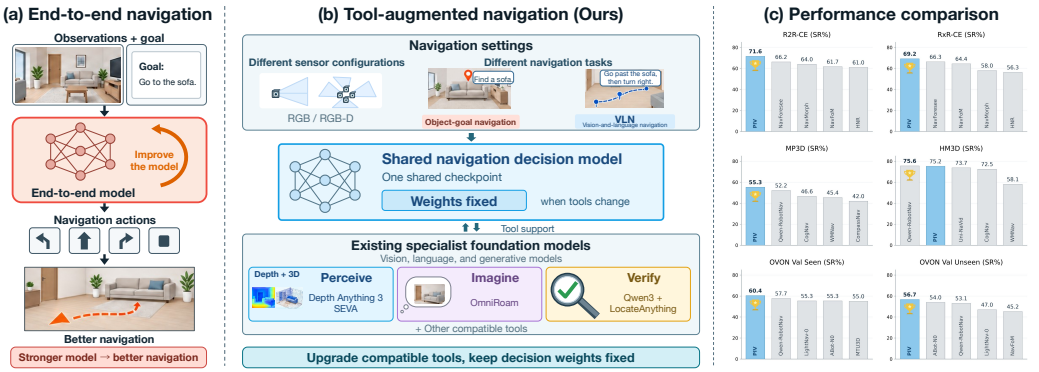

Figure 1: **Method and performance comparisons.** (a) End-to-end navigation. (b) PIV combines a shared navigation decision model with external tools, allowing compatible tools to change while keeping decision weights fixed. (c) Performance comparison. Blue bars denote Our PIV, whereas gray bars denote prior methods.

compare their anticipated outcomes before execution. The decision model uses the goal, observation history, and available perceptual and predictive information to select a movement or propose stopping. Verification tools then ground the target to validate stopping proposals before navigation terminates.

By maintaining consistent interfaces, PIV allows compatible tool backends to be replaced or upgraded without updating the navigation decision model, providing a route to benefiting from independent advances in specialist models. Perception tools further adapt observations across panoramic and forward-facing RGB and RGB-D configurations, enabling the same checkpoint to operate without sensor-specific fine-tuning.

We evaluate PIV on vision-and-language navigation using R2R-CE and RxR-CE (Krantz et al., 2020; Ku et al., 2020), closed-vocabulary object-goal navigation using MP3D and HM3D (Chang et al., 2017; Yadav et al., 2023), and open-vocabulary object-goal navigation using HM3D-OVON (Yokoyama et al., 2024b). With one shared checkpoint and no benchmark-specific fine-tuning, PIV achieves competitive performance across these task families (Figure 1c). Tool ablations on R2R-CE and evaluations across sensor configurations further characterize how external tools support the shared navigator.

Our contributions are threefold:

- First, inspired by the extended mind thesis, we propose separating navigation decision-making from the acquisition of specialist capabilities, allowing generalist navigation capabilities to be jointly realized by a shared decision model and independently pretrained external tools.
- Second, we develop PIV, a loosely coupled framework that connects perception, foresight, and verification tools with a shared navigation decision model through compatible interfaces, supporting independent tool replacement and adaptation to different sensor configurations.
- Last, we demonstrate competitive performance across vision-and-language and closed- and open-vocabulary object-goal navigation with one checkpoint, together with adaptation across panoramic and forward-facing RGB and RGB-D configurations without sensor-specific fine-tuning.

2 Related Work

2.1 Generalist Embodied Navigation

Embodied navigation encompasses diverse tasks, including vision-and-language navigation and object-goal navigation (Anderson et al., 2018; Batra et al., 2020; Yokoyama et al.,

2024b). This diversity has motivated a transition from task-specific systems toward generalist navigation models. Uni-NaVid (Zhang et al., 2024a) unifies multiple tasks through a video-based vision-language-action policy, while NavFoM (Zhang et al., 2026a) extends shared navigation learning across tasks and embodiments. OmniNav (Xue et al., 2026) and ABot-N0 (Chu et al., 2026) combine high-level reasoning with trajectory generation, and Qwen-RobotNav (Zhang et al., 2026b) develops a scalable navigation model through joint vision-language and trajectory training. These efforts advance knowledge sharing and broaden the range of tasks and environments supported by a common navigation model. However, adding or improving a capability learned within the navigation policy can require further training of that policy. This raises a complementary question: once a shared navigation model has been trained, how can it benefit from independently pretrained specialist models without updating its decision weights? PIV investigates this direction by separating navigation decisions from the acquisition of specialist capabilities, combining a shared cross-task decision model with external perception, foresight, and verification tools.

2.2 Specialist Foundation Models

Large-scale pretraining has produced models with strong capabilities in visual geometry, controllable visual generation, and language-conditioned localization. Depth Anything 3 (Lin et al., 2025) recovers spatially consistent geometry from single or multiple images, providing depth information when direct measurements are unavailable. Generative models offer complementary priors for views beyond the current observation: Stable Virtual Camera (SEVA) (Zhou et al., 2025) synthesizes novel views under camera control, while OmniRoam (Liu et al., 2026b) generates panoramic videos conditioned on visual observations and camera trajectories. Navigation World Models (Bar et al., 2025) further demonstrate how action-conditioned predictions of future observations can support navigation planning and the evaluation of candidate trajectories. For language-conditioned localization, LocateAnything (Wang et al., 2026b) grounds target descriptions in images by predicting bounding boxes. Together, these advances provide reusable geometric, predictive, and grounding capabilities learned independently of a navigation policy. PIV draws on these capabilities through external tools that support a shared navigation decision model.

3 Method

3.1 Framework and Shared Navigation Decision Model

PIV formulates embodied navigation as hierarchical decision making over a shared waypoint interface. At timestep t, the agent uses an egocentric observation O_t, a language goal L, and history H_t to choose a locally reachable waypoint or Stop. Figure 2 summarizes the shared decision model and its external tools. PIV expresses heterogeneous navigation goals through language. For VLN, L is a route instruction; for ObjectNav and OVON, it specifies a target category. These tasks share one prompt structure and waypoint output interface and differ primarily in the instruction text. The navigation harness first converts the raw sensor stream into a heading-indexed spatial observation

$$\widetilde{O}_t = \left\{ \left(\widetilde{I}_t^k, \widetilde{D}_t^k, C_t^k \right) \right\}_{k=0}^{N-1}, \qquad \alpha_k = \frac{2\pi k}{N}, \tag{1}$$

where view 0 is aligned with the agent heading, α_k is the relative yaw of view k, and C_t^k optionally records view validity. A fixed waypoint proposer predicts a polar traversability map from $\widetilde{O}_t$ and extracts feasible candidates using circular non-maximum suppression:

$$\mathcal{W}_t = \{w_t^i\}_{i=1}^{M_t}, \qquad w_t^i = (\phi_t^i, d_t^i). \tag{2}$$

Candidate locations are projected onto four cardinal RGB views and marked with identifiers such as `A`, `B`, and `C`. These identifiers link the visual prompt and action interface to explicit spatial targets. Rather than exposing model-specific APIs, PIV organizes foundation capa-

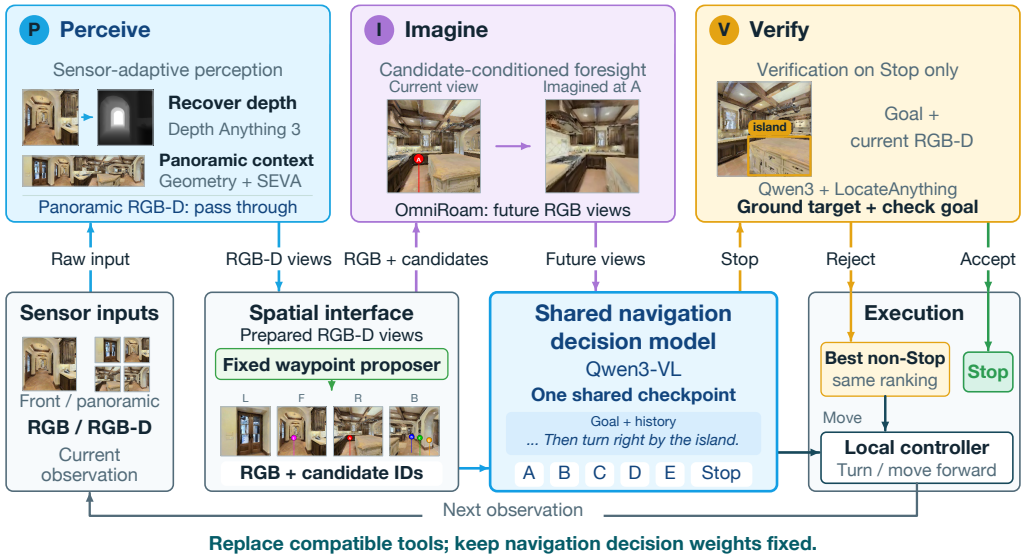

Figure 2: **Overview of PIV.** Perception tools adapt sensor inputs into RGB-D observations. A fixed waypoint proposer supplies feasible candidates, while foresight generates candidate-conditioned future views. The shared decision model combines RGB observations, candidate identifiers, the goal, history, and foresight evidence to propose a move or STOP. Verification grounds targets to validate stopping proposals. After training, compatible tools can change while decision weights remain fixed.

bilities through a small set of shared tool interfaces:

$$\mathcal{T}_{\text{per}} = \{\texttt{complete_panorama}, \texttt{recover_depth}\}, \tag{3}$$
$$\mathcal{T}_{\text{fore}} = \{\texttt{imagine}(i)\}, \tag{4}$$
$$\mathcal{T}_{\text{ver}} = \{\texttt{verify}(L)\}, \tag{5}$$
$$\mathcal{T}_{\text{act}} = \{\texttt{move_to}(i), \texttt{stop}()\}. \tag{6}$$

Perception tools normalize observations, foresight imagines candidate outcomes, verification provides target evidence, and action tools execute moves or terminate the episode. Perception is routed by the sensor configuration, and foresight can be enabled to inform waypoint selection. Compatible tool backends can be replaced through these interfaces while navigation decision weights remain unchanged.

Shared decision model. We instantiate $\mathcal{S}_\theta$ with Qwen3-VL (Bai et al., 2025):

$$u_t = \mathcal{S}_\theta\left(L, H_t, \widetilde{O}_t^{\text{RGB}}, \mathcal{W}_t, E_t^{\text{fore}}\right) \in \mathcal{W}_t \cup \{\text{STOP}\}. \tag{7}$$

Here, $\widetilde{O}_t^{\text{RGB}}$ denotes the annotated cardinal RGB views; depth supports waypoint proposal, target localization, and local control rather than entering the decision model directly. The history H_t contains observations from previous waypoint boundaries, and E_t^{fore} pairs queried candidate identifiers with imagined future views when foresight is enabled and is empty otherwise. The model outputs a candidate identifier or STOP, rather than an open-ended plan. One checkpoint is trained with token-level supervision on mixed navigation tasks and reused across supported tasks, sensor configurations, and tool settings.

Action execution and stopping. If $u_t = w_t^i$, the harness invokes

$$\texttt{move_to}(i) \rightarrow a_{t:t+\tau} = \pi_{\text{local}}(O_t, w_t^i). \tag{8}$$

The local controller turns toward the waypoint and advances until it is reached or replanning is required, with τ denoting the execution duration. If $u_t =$ STOP and verification is

enabled, the harness invokes `verify`(L): acceptance invokes `stop`() and ends the episode, while rejection executes the highest-scoring non-STOP waypoint from the same decision output. When verification is disabled, STOP directly ends the episode.

3.2 PERCEPTION TOOLS

In practice, robots differ in their sensing capabilities. Perception tools enable PIV to reuse the same navigation weights across these configurations without retraining. Panorama reconstruction provides panoramic context from limited-field-of-view observations, while depth recovery estimates missing depth. Native panoramic RGB-D observations are used directly to form $\widetilde{O}_t$; otherwise, the harness invokes only the required perception tools.

Panorama reconstruction. This tool reconstructs panoramic RGB-D observations from a single forward-facing RGB-D camera. PIV fuses posed RGB-D observations into a colored point cloud and renders $N = 12$ heading-indexed views at the current position, retaining the current front view. Geometric rendering supplies RGB, nearest-surface depth, and validity masks C_t^k; its implementation is detailed in Appendix B. When coverage is insufficient, an in-place active scan captures additional views, which are fused before rendering again. SEVA (Zhou et al., 2025) can complete residual RGB gaps using its pretrained generative prior, without changing the validity of unobserved geometry.

Depth recovery. When depth measurements are unavailable, PIV uses DA3 (Lin et al., 2025) to jointly predict relative depth maps $\{\widehat{D}_t^k\}_{k=0}^{N-1}$ from the N panoramic RGB views in a single forward pass. A shared scale derived from the median valid depth across the panorama avoids independent normalization across headings (Appendix B). The normalized maps replace the depth inputs to the frozen waypoint predictor, while the selector retains the original RGB observations.

3.3 FORESIGHT TOOL

The foresight tool imagines the front RGB view at a candidate waypoint before execution. We instantiate $\mathcal{F}$ with OmniRoam (Liu et al., 2026b), a pretrained camera-controlled panoramic video model. Let $\mathcal{V}_t$ denote the observed RGB frames used for conditioning. For candidate w_t^i, the harness derives a camera trajectory Γ_t^i from the current camera pose and waypoint. OmniRoam conditions generation on $\mathcal{V}_t$ and the trajectory's motion directions and displacement scale. The interface returns

$$\texttt{imagine}(i) \rightarrow \widehat{I}_{t+1}^i = \Pi_{\text{front}}^i \left[\mathcal{F}(\mathcal{V}_t, \Gamma_t^i)\right], \tag{9}$$

where Π_{front}^i extracts a perspective view from the generated panorama at the trajectory endpoint, aligned with the expected arrival heading. For queried candidates $\mathcal{I}_t$, the evidence $E_t^{\text{fore}} = \{(i, \widehat{I}_{t+1}^i)\}_{i \in \mathcal{I}_t}$ pairs each imagined view with its waypoint index. These views provide semantic evidence for comparing candidate moves.

3.4 VERIFICATION TOOL

The verification tool converts the instruction into a grounding query and localizes the target in the current observation through the following interface:

$$\texttt{verify}(L) \rightarrow E_t^{\text{ver}} = \text{Ground}\left(\widetilde{O}_t, \text{TargetParser}(L)\right). \tag{10}$$

A Qwen3-based parser extracts the target object and its descriptive constraints from L. LocateAnything (Wang et al., 2026b) then grounds the query in the current cardinal views and returns a target location and confidence. Combining the grounded image location with aligned depth yields the target-relative bearing and range. This evidence supports the stopping rule in Section 3.1.

3.5 Data Collection and Training

Data Collection. We collect navigation records from R2R-CE (Krantz et al., 2020), RxR-CE (Ku et al., 2020), ScaleVLN (Wang et al., 2023), ObjectNav (HM3D/MP3D) (Yadav et al., 2023; Chang et al., 2017), and OVON (Yokoyama et al., 2024b). Native panoramic RGB-D inputs provide canonical observations for the fixed waypoint proposer during decision-model training. Foresight outputs are included in decision-model training. Perception and verification tools are integrated afterward without further training of the decision model. The target is a teacher-selected candidate identifier or STOP, determined from the reference path or navigation goal.

Training Scheme. We train the Qwen3-VL navigation decision model $\mathcal{S}_\theta$ in two stages, both using token-level supervision of the target action on eight NVIDIA H20 GPUs. **Stage 1** fine-tunes the pretrained model on 1.1M standard training samples for 26 hours. Based on this model, we employ the DAgger (Ross et al., 2011) algorithm to collect 1.7M additional training samples. **Stage 2** jointly trains on the combined 2.8M standard and DAgger training samples for 60 hours. This stage restarts from the original pretrained Qwen3-VL checkpoint rather than continuing from the Stage 1 weights.

4 Experiments

Table 1: Comparison on R2R-CE and RxR-CE Val-Unseen, grouped by input observation setting. Baseline results follow the cited publications and the comparison tables in Qwen-RobotNav (Zhang et al., 2026b); NavMorph and NavForesee results are from their original papers. Bold and underlining denote the best and second-best reported results, respectively, within each observation group.

		R2R-CE Val-Unseen				RxR-CE Val-Unseen		
Method	Publication	NE↓	OS↑	SR↑	SPL↑	NE↓	SR↑	SPL↑
Monocular								
NaVid (Zhang et al., 2024b)	RSS24	5.72	49.2	41.9	36.5	<u>5.72</u>	45.7	38.2
Uni-NaVid (Zhang et al., 2024a)	RSS25	5.58	53.3	47.0	42.7	6.24	48.7	40.9
NaVILA (Cheng et al., 2024)	RSS25	5.22	62.5	54.0	49.0	6.77	49.3	44.0
NavMorph (Yao et al., 2025)	ICCV25	5.75	56.9	47.9	33.2	8.85	30.8	22.8
StreamVLN (Wei et al., 2025)	ICRA26	4.98	64.2	56.9	51.9	6.22	52.9	46.0
JanusVLN (Zeng et al., 2026)	ICLR26	<u>4.78</u>	<u>65.2</u>	<u>60.5</u>	<u>56.8</u>	6.06	<u>56.2</u>	<u>47.5</u>
PIV (ours)	–	**3.95**	**73.6**	**66.8**	**58.5**	**4.67**	**66.6**	**49.2**
Panoramic								
ETPNav (An et al., 2024)	TPAMI25	4.71	65.0	57.0	49.0	5.64	54.7	44.8
HNR (Wang et al., 2024)	CVPR24	4.42	67.0	61.0	51.0	5.50	56.3	46.7
NavMorph$^\diamond$ (Yao et al., 2025)	ICCV25	4.37	68.0	64.0	53.0	5.70	58.0	49.0
NavFoM (Zhang et al., 2026a)	ICLR26	4.61	72.1	61.7	55.3	4.74	64.4	**56.2**
NavForesee (Liu et al., 2026a)	CVPR26	<u>3.94</u>	**78.4**	<u>66.2</u>	<u>59.7</u>	<u>4.20</u>	<u>66.3</u>	53.2
PIV (ours)	–	**3.45**	<u>77.5</u>	**71.6**	**63.1**	**4.07**	**69.2**	<u>54.5</u>

4.1 Experimental Setup

Benchmarks and metrics. We evaluate vision-and-language navigation on the Val-Unseen splits of R2R-CE (Krantz et al., 2020) and RxR-CE (Ku et al., 2020), closed-vocabulary ObjectNav on MP3D (Chang et al., 2017) and HM3D (Yadav et al., 2023), and open-vocabulary ObjectNav on the Seen, Synonyms, and Unseen validation splits of HM3D-OVON (Yokoyama et al., 2024b). For VLN, we report navigation error (NE), success rate (SR), and success weighted by path length (SPL), with oracle success rate (OS) additionally reported on R2R-CE. ObjectNav uses SR and SPL. NE is measured in meters; all other metrics are percentages. Among these, SR and SPL are widely regarded as the primary metrics, reflecting task completion and path efficiency.

Implementation details. We initialize the shared high-level selector from Qwen3-VL-4B-Instruct (Bai et al., 2025) and use a frozen pretrained RGB-D waypoint predictor (Shi et al., 2025) to construct the candidate decision set. We freeze the Qwen3-VL visual backbone while fine-tuning the complete language model together with the visual merger/projector modules. All trainable parameters use a single peak learning rate of 2×10^{-5}; we do not use a separate lower learning rate for the projector. Training uses AdamW with $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$, weight decay 0.05, a linear learning-rate schedule, and a 10% warmup ratio. We train on eight H20 GPUs with four examples per GPU and four gradient-accumulation steps, giving an effective batch size of 128. We use bfloat16, FlashAttention-2, DeepSpeed ZeRO-2, and gradient checkpointing.

4.2 Comparison with the State of the Art

We compare PIV with task-specific navigation systems and generalist models, including Uni-NaVid (Zhang et al., 2024a), NavFoM (Zhang et al., 2026a), and Qwen-RobotNav (Zhang et al., 2026b). The comparisons additionally cover navigation with future prediction through NavMorph (Yao et al., 2025) and NavForesee (Liu et al., 2026a), universal goal navigation through UniGoal (Yin et al., 2025), decision learning through CompassNav (Li et al., 2026), and open-vocabulary navigation through OVSegDT (Zemskova et al., 2025) and JanusVLN (Zeng et al., 2026).

Vision-and-language navigation. Table 1 groups methods by monocular and panoramic observations. With panoramic input, PIV achieves 71.6% SR on R2R-CE and 69.2% on RxR-CE, exceeding NavForesee by 5.4 and 2.9 percentage points, respectively, with higher SPL on both benchmarks. PIV retains one shared decision checkpoint across tasks, although NavForesee has higher R2R-CE OS and NavFoM has higher RxR-CE SPL. These system-level comparisons do not isolate the contribution of foresight, which we examine through ablations.

Table 2: Closed-vocabulary ObjectNav results on MP3D and HM3D. Results follow the evaluation settings of the original papers. Bold and underlining denote the best and second-best results, respectively, in each column; – denotes an unreported result.

		MP3D		HM3D	
Method	Publication	SR↑	SPL↑	SR↑	SPL↑
VLFM‡ (Yokoyama et al., 2024a)	ICRA24	36.4	<u>17.5</u>	52.5	30.4
OpenFMNav‡ (Kuang et al., 2024)	NAACL24	37.2	15.7	52.5	24.1
SG-Nav‡ (Yin et al., 2024)	NeurIPS24	40.2	16.0	54.0	24.9
TriHelper‡ (Zhang et al., 2024c)	IROS24	–	–	56.5	25.3
UniGoal‡ (Yin et al., 2025)	CVPR25	41.0	16.4	54.5	25.1
WMNav‡ (Nie et al., 2025)	IROS25	45.4	17.2	58.1	31.2
CogNav‡ (Cao et al., 2025)	ICCV25	46.6	16.1	72.5	26.2
Uni-NaVid (Zhang et al., 2024a)	RSS25	–	–	73.7	<u>37.1</u>
CompassNav† (Li et al., 2026)	ICLR26	42.0	<u>17.5</u>	56.6	27.6
Qwen-RobotNav-4B (Zhang et al., 2026b)	arXiv26	<u>52.2</u>	16.0	**75.6**	30.6
PIV (ours)	–	**55.3**	**28.3**	<u>75.2</u>	**45.6**

‡ HM3D v1 results; PIV is evaluated on HM3D v2. † CompassNav results are from Table 1 of the original paper; its HM3D version is unspecified.

Closed-vocabulary object-goal navigation. In Table 2, PIV achieves the highest reported MP3D SR and SPL, exceeding Qwen-RobotNav-4B by 3.1 and 12.3 percentage points, respectively. On HM3D, its SR is 0.4 points lower than Qwen-RobotNav-4B, while its SPL is 15.0 points higher, indicating more efficient successful navigation under the reported evaluation settings.

Table 3: Open-vocabulary ObjectNav results on the HM3D-OVON validation splits (Yokoyama et al., 2024b). Bold and underlining denote the best and second-best results, respectively, in each column; – denotes an unreported result. † marks baseline results reported with depth or odometry. OVSegDT and JanusVLN use monocular RGB input; JanusVLN reports only the Unseen split.

		Val Seen		Val Synonyms		Val Unseen	
Method	Publication	SR↑	SPL↑	SR↑	SPL↑	SR↑	SPL↑
VLFM† (Yokoyama et al., 2024a)	ICRA24	35.2	18.6	32.4	17.3	35.2	19.6
DAgRL+OD† (Yokoyama et al., 2024b)	IROS24	38.5	21.1	39.0	21.4	37.1	19.8
MTU3D† (Zhu et al., 2025)	ICCV25	55.0	23.6	45.0	14.7	40.8	12.1
Uni-NaVid (Zhang et al., 2024a)	RSS25	41.3	21.1	43.9	21.8	39.5	19.8
OVSegDT (Zemskova et al., 2025)	CVPR26	43.6	20.1	40.1	17.9	44.7	20.6
JanusVLN (Zeng et al., 2026)	ICLR26	–	–	–	–	44.9	31.7
NavFoM (4-view) (Zhang et al., 2026a)	ICLR26	40.1	27.1	45.4	32.6	45.2	<u>31.9</u>
LightNav-0 (Wang et al., 2026a)	arXiv26	55.3	31.2	54.6	29.6	47.0	24.2
ABot-N0 (Chu et al., 2026)	arXiv26	55.3	<u>32.1</u>	55.4	<u>33.2</u>	<u>54.0</u>	30.5
Qwen-RobotNav-4B (Zhang et al., 2026b)	arXiv26	<u>57.7</u>	24.4	**60.1**	25.1	53.1	20.9
PIV (ours)	–	**60.4**	**36.4**	<u>59.2</u>	**35.8**	**56.7**	**33.2**

Open-vocabulary object-goal navigation. PIV achieves the highest SPL on all three HM3D-OVON splits and the highest SR on Seen and Unseen (Table 3). On Unseen, it exceeds ABot-N0 by 2.7 percentage points in both SR and SPL; on Synonyms, its SR is lower than Qwen-RobotNav-4B (59.2% vs. 60.1%). Together with the VLN and closed-vocabulary results, these comparisons support competitive performance across tasks with one shared decision checkpoint. The following ablations examine individual tools within this complete system.

4.3 Ablation Study

Table 4: Tool ablations and sensor adaptation on R2R-CE Val-Unseen using the Stage 1 checkpoint. Upper: native panoramic RGB-D, bypassing perception tools. Lower: foresight and verification disabled, with w/o both as the reference. Front denotes a single forward-facing camera; – denotes a disabled or unnecessary tool.

	Tools				R2R-CE Val-Unseen			
Setting	Depth recovery	Panorama reconstruction	Foresight	Verification	NE↓	OS↑	SR↑	SPL↑
(a) Foresight and verification: panoramic RGB-D								
Full PIV	–	–	✓	✓	3.87	72.60	69.03	63.53
w/o verification	–	–	✓	–	4.00	72.32	68.62	64.17
w/o foresight	–	–	–	✓	4.05	71.17	66.47	62.12
w/o both	–	–	–	–	4.23	70.85	66.01	61.88
(b) Sensor adaptation: foresight and verification disabled								
Panoramic RGB-D	–	–	–	–	4.23	70.85	66.01	61.88
Panoramic RGB	✓	–	–	–	4.12	71.29	65.91	61.60
Front RGB-D	–	✓	–	–	4.46	68.41	62.70	58.66
Front RGB	✓	✓	–	–	4.95	62.28	57.37	52.32

All ablations use the same fixed Stage 1 checkpoint, trained without DAgger data. Table 4 reports tool ablations and four sensor configurations on R2R-CE Val-Unseen, with the waypoint proposer, local controller, evaluation episodes, and action budgets held fixed.

Foresight and verification. Table 4 shows that foresight alone improves R2R-CE SR by 2.61 percentage points, compared with 0.46 for verification. Combining both achieves the highest SR of 69.03%, but lowers SPL relative to foresight alone, indicating a trade-off between task completion and path efficiency. In contrast, the ObjectNav ablations (Table 5)



Figure 3: **Qualitative navigation rollout of PIV.** Six selected steps show waypoint navigation, a `Ground` call, and stopping. Each panel contains four directional RGB views and a top-down map. Lettered markers denote candidate waypoints; green and blue lines indicate the reference and executed trajectories, respectively, and the red star marks the goal. The cyan cross and diamond in panel (e) denote the tool's image-space detection and estimated 3D target, respectively.

show that verification alone improves HM3D-OVON Val-Unseen SR by 1.56 percentage points, while foresight alone has little effect (+0.05 points).

Adaptation to sensor settings. With foresight and verification disabled (Table 4(b)), replacing sensed depth with DA3 (Lin et al., 2025) predictions under panoramic RGB input reduces SR and SPL by only 0.10 and 0.28 percentage points, respectively, largely preserving navigation performance. With front-facing RGB-D input, geometric panorama reconstruction and SEVA (Zhou et al., 2025) completion enable the same checkpoint to operate, but SR remains 3.31 points below the panoramic RGB-D reference. Removing depth as well widens this gap to 8.64 points. Thus, perception tools support all four sensor configurations without sensor-specific fine-tuning, but do not fully compensate for limited visual coverage.

4.4 Qualitative Results

Figure 3 illustrates how PIV combines waypoint-based navigation with the Verification Tool (`Ground`) to follow a multi-stage instruction. The agent exits the kitchen, turns right, and approaches the specified bedroom doorway before invoking `Ground` at Step 06 and stopping at Step 07. Its trajectory broadly follows the reference path, with local deviations in the hallway, and ends 0.215 m from the navigation goal.

5 Conclusion

We presented Perceive–Imagine–Verify (PIV), a loosely coupled framework that augments a shared navigation decision model with independently pretrained specialist models as external tools. Perception, foresight, and verification provide complementary capabilities for adapting observations, generating candidate-conditioned future views, and validating stopping proposals. This separation allows compatible tool backends to be replaced or upgraded while keeping the navigation decision weights fixed, providing a path to benefiting from advances in specialist models. Evaluations show competitive performance across vision-and-language and closed- and open-vocabulary object-goal navigation with one shared checkpoint. Perception tools also enable this checkpoint to operate across panoramic and forward-facing RGB and RGB-D configurations without sensor-specific fine-tuning.

AI Use Statement

We used generative AI tools to assist with manuscript drafting and revision, discussion of how to frame and present the work, identification and organization of relevant literature and comparison baselines, and preparation of scientific figures. The authors designed the navigation method, implemented the research code, and conducted the experiments. The authors take responsibility for the final manuscript, including all AI-assisted text, references, and figures.

References

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*, 47(7): 5130–5145, 2024.

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 *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 3674–3683. IEEE, 2018.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Amir Bar, Gaoyue Zhou, Danny Tran, Trevor Darrell, and Yann LeCun. Navigation world models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15791–15801. IEEE, 2025.

Dhruv Batra, Aaron Gokaslan, Aniruddha Kembhavi, Oleksandr Maksymets, Roozbeh Mottaghi, Manolis Savva, Alexander Toshev, and Erik Wijmans. Objectnav revisited: On evaluation of embodied agents navigating to objects. *arXiv preprint arXiv:2006.13171*, 2020.

Yihan Cao, Jiazhao Zhang, Zhinan Yu, Shuzhen Liu, Zheng Qin, Qin Zou, Bo Du, and Kai Xu. Cognav: Cognitive process modeling for object goal navigation with llms. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9550–9560. IEEE, 2025.

Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *arXiv preprint arXiv:1709.06158*, 2017.

An-Chieh Cheng, Yandong Ji, Zhaojing Yang, Zaitian Gongye, Xueyan Zou, Jan Kautz, Erdem Bıyık, Hongxu Yin, Sifei Liu, and Xiaolong Wang. Navila: Legged robot vision-language-action model for navigation. *arXiv preprint arXiv:2412.04453*, 2024.

Zedong Chu, Shichao Xie, Xiaolong Wu, Yanfen Shen, Minghua Luo, Zhengbo Wang, Fei Liu, Xiaoxu Leng, Junjun Hu, Mingyang Yin, et al. Abot-n0: Technical report on the vla foundation model for versatile embodied navigation. *arXiv preprint arXiv:2602.11598*, 2026.

Andy Clark and David Chalmers. The extended mind. *analysis*, 58(1):7–19, 1998.

Ruiyan Gong, Yingnan Guo, Junjun Hu, Jintao Kong, Xiaoxu Leng, Tianlun Li, Weize Li, Fei Liu, Zhicheng Liu, Jia Lu, et al. Abot-n1: Toward a general visual language navigation foundation model. *arXiv preprint arXiv:2607.10383*, 2026.

Wenxuan Guo, Xiuwei Xu, Yichen Liu, Xiangyu Li, Hang Yin, Huangxing Chen, Wenzhao Zheng, Jianjiang Feng, Jie Zhou, and Jiwen Lu. Awarevln: Reasoning with self-awareness for vision-language navigation. *arXiv preprint arXiv:2605.22816*, 2026.

Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *European Conference on Computer Vision*, pp. 104–120. Springer, 2020.

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

Yuxuan Kuang, Hai Lin, and Meng Jiang. Openfmnav: Towards open-set zero-shot object navigation via vision-language foundation models. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 338–351, 2024.

LinFeng Li, Jian Zhao, Yuan Xie, Xin Tan, and Xuelong Li. Compassnav: Steering from path imitation to decision understanding in navigation. In *International Conference on Learning Representations*, volume 2026, pp. 44744–44768, 2026.

Haotong Lin, Sili Chen, Junhao Liew, Donny Y Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*, 2025.

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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 32431–32440, 2026a.

Yuheng Liu, Xin Lin, Xinke Li, Baihan Yang, Chen Wang, Kalyan Sunkavalli, Yannick Hold-Geoffroy, Hao Tan, Kai Zhang, Xiaohui Xie, et al. Omniroam: World wandering via long-horizon panoramic video generation. *arXiv preprint arXiv:2603.30045*, 2026b.

Dujun Nie, Xianda Guo, Yiqun Duan, Ruijun Zhang, and Long Chen. Wmnav: Integrating vision-language models into world models for object goal navigation. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2392–2399. IEEE, 2025.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 627–635. JMLR Workshop and Conference Proceedings, 2011.

Xiangyu Shi, Zerui Li, Wenqi Lyu, Jiatong Xia, Feras Dayoub, Yanyuan Qiao, and Qi Wu. Smartway: Enhanced waypoint prediction and backtracking for zero-shot vision-and-language navigation. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 16923–16930. IEEE, 2025.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Shaoan Wang, Aocheng Luo, Fei Huang, Jingyi Xu, Xiaoyang Wang, Yueyu Wang, Qianli Ma, Fan Yang, Ran Mei, Jia Wei, et al. Lightnav-0: Eliciting vlm spatial intelligence for generalist embodied navigation. *arXiv preprint arXiv:2608.30935*, 2026a.

Shihao Wang, Shilong Liu, Yuanguo Kuang, Xinyu Wei, Yangzhou Liu, Zhiqi Li, Yunze Man, Guo Chen, Andrew Tao, Guilin Liu, et al. Locateanything: Fast and high-quality vision-language grounding with parallel box decoding. In *European Conference on Computer Vision*, pp. 336–357. Springer, 2026b.

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

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 *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11975–11986. IEEE, 2023.

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.

Meng Wei, Chenyang Wan, Peng Peng, Xiqian Yu, Yuqiang Yang, Delin Feng, Wenzhe Cai, Chenming Zhu, Tai Wang, Jiangmiao Pang, et al. Ground slow, move fast: A dual-system foundation model for generalizable vision-language navigation. In *International Conference on Learning Representations*, volume 2026, pp. 12380–12396, 2026.

Xinda Xue, Junjun Hu, Minghua Luo, Shichao Xie, Jintao Chen, Zixun Xie, Quan Kuichen, Wei Guo, Zedong Chu, Mu Xu, et al. Omninav: A unified framework for prospective exploration and visual-language navigation. In *International Conference on Learning Representations*, volume 2026, pp. 106216–106231, 2026.

Karmesh Yadav, Ram Ramrakhya, Santhosh Kumar Ramakrishnan, Theo Gervet, John Turner, Aaron Gokaslan, Noah Maestre, Angel Xuan Chang, Dhruv Batra, Manolis Savva, et al. Habitat-matterport 3d semantics dataset. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4927–4936. IEEE, 2023.

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

Hang Yin, Xiuwei Xu, Zhenyu Wu, Jie Zhou, and Jiwen Lu. Sg-nav: Online 3d scene graph prompting for llm-based zero-shot object navigation. *Advances in neural information processing systems*, 37:5285–5307, 2024.

Hang Yin, Xiuwei Xu, Linqing Zhao, Ziwei Wang, Jie Zhou, and Jiwen Lu. Unigoal: Towards universal zero-shot goal-oriented navigation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19057–19066. IEEE, 2025.

Naoki Yokoyama, Sehoon Ha, Dhruv Batra, Jiuguang Wang, and Bernadette Bucher. Vlfm: Vision-language frontier maps for zero-shot semantic navigation. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 42–48. IEEE, 2024a.

Naoki Yokoyama, Ram Ramrakhya, Abhishek Das, Dhruv Batra, and Sehoon Ha. Hm3d-ovon: A dataset and benchmark for open-vocabulary object goal navigation. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5543–5550. IEEE, 2024b.

Tatiana Zemskova, Aleksei Staroverov, Dmitry Yudin, and Aleksandr Panov. Ovsegdt: segmenting transformer for open-vocabulary object goal navigation. *arXiv preprint arXiv:2508.11479*, 2025.

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*, volume 2026, pp. 33001–33026, 2026.

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. *arXiv preprint arXiv:2412.06224*, 2024a.

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. *arXiv preprint arXiv:2402.15852*, 2024b.

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. In *International Conference on Learning Representations*, volume 2026, pp. 127293–127322, 2026a.

Jiazhao Zhang, Gengze Zhou, Hale Yin, Yiyang Huang, Zixing Lei, Qihang Peng, Haoqi Yuan, Jie Zhang, Xudong Guo, Xiaoyue Chen, et al. Qwen-robotnav technical report: A scalable navigation model designed for an agentic navigation system. *arXiv preprint arXiv:2606.18112*, 2026b.

Lingfeng Zhang, Qiang Zhang, Hao Wang, Erjia Xiao, Zixuan Jiang, Honglei Chen, and Renjing Xu. Trihelper: Zero-shot object navigation with dynamic assistance. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 10035–10042. IEEE, 2024c.

Jensen Zhou, Hang Gao, Vikram Voleti, Aaryaman Vasishta, Chun-Han Yao, Mark Boss, Philip Torr, Christian Rupprecht, and Varun Jampani. Stable virtual camera: Generative view synthesis with diffusion models. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 12405–12414. IEEE, 2025.

Ziyu Zhu, Xilin Wang, Yixuan Li, Zhuofan Zhang, Xiaojian Ma, Yixin Chen, Baoxiong Jia, Wei Liang, Qian Yu, Zhidong Deng, et al. Move to understand a 3d scene: Bridging visual grounding and exploration for efficient and versatile embodied navigation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8120–8132. IEEE, 2025.

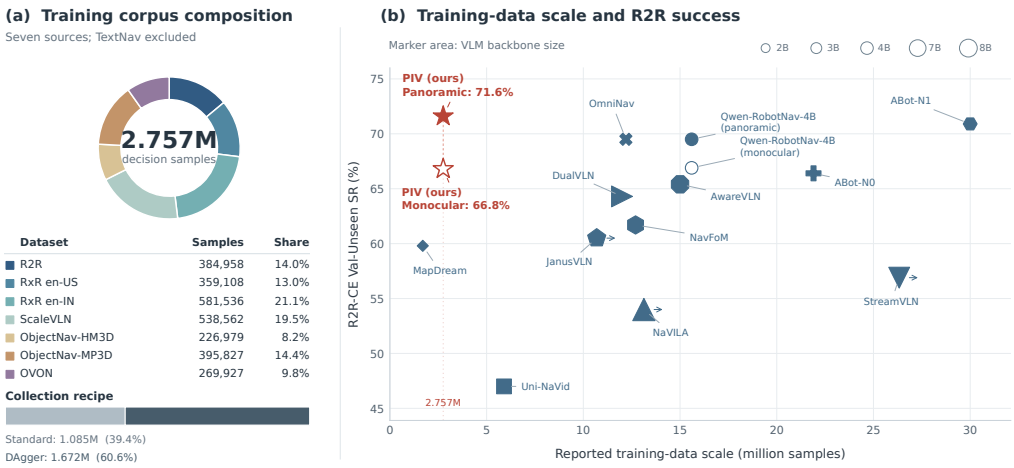

Figure 4: **Training-data composition and reported data–performance comparison.** (a) Composition of PIV's 2.757M navigation decision samples by source and collection strategy. (b) R2R-CE Val-Unseen SR against reported training-data scale. Marker shapes identify methods; enclosed marker area is proportional to the nominal parameter count of the main navigation VLM, excluding auxiliary models and external tools. Filled and open markers distinguish panoramic and monocular settings for PIV and Qwen-RobotNav-4B, respectively; PIV uses the same checkpoint for both. † denotes an external-data count, from the comparison in Zeng et al. (2026), with a rightward arrow indicating that the full total is larger; StreamVLN's count is approximate. ‡ denotes a cross-paper estimate for AwareVLN or DualVLN (Guo et al., 2026; Wei et al., 2026). § ABot-N1's approximately 30M samples exclude 0.5M separately reported GRPO episodes Gong et al. (2026)

.

A Training data and navigation performance.

Figure 4 summarizes the composition of our training corpus and places its scale in the context of reported R2R-CE Val-Unseen success rates and navigation backbone sizes.

Our training corpus contains 2,756,897 navigation decision samples after excluding TextNav from both the standard and DAgger collections. Standard training data contribute 1,085,121 samples (39.36%), while DAgger contributes 1,671,776 (60.64%). The two RxR subsets together account for 34.12% of the corpus, followed by ScaleVLN (19.54%), ObjectNav-MP3D (14.36%), R2R (13.96%), OVON (9.79%), and ObjectNav-HM3D (8.23%). This mixture combines instruction-following and object-goal supervision with a substantial amount of policy-collected data. Samples reused across training stages are not added again.

Using a 4B decision backbone, PIV achieves 71.6% SR with panoramic observations and 66.8% with monocular observations. Compared with Qwen-RobotNav-4B (Zhang et al., 2026b), PIV improves panoramic SR by 2.1 percentage points and obtains a monocular SR only 0.1 point lower, with a corpus size equal to 17.67% of its reported 15.6M samples. Relative to ABot-N0 (Chu et al., 2026), PIV's panoramic SR is 5.2 points higher, with 12.59% of its reported 21.9M samples. PIV also exceeds the reported 70.9% SR of ABot-N1 (Gong et al., 2026), which uses approximately 30M pretraining samples and additional reinforcement-learning episodes. These comparisons show a favorable combination of reported corpus size and navigation performance. They do not constitute a controlled data-efficiency experiment: sample granularity, sensing, auxiliary supervision, pretrained components, and training procedures differ across methods.

B Tool Implementation Details

This section provides implementation details for PIV's perception, foresight, and verification tools. We describe panorama reconstruction and depth normalization, the generation of candidate-aligned future views, and the use of grounded target evidence to validate or revise stopping decisions.

B.1 Geometric Panorama Reconstruction

PIV accumulates posed RGB-D observations into a colored point cloud. For a valid pixel $\mathbf{u} = (u, v)$ in frame s, its world coordinate is

$$\mathbf{X}_{s,\mathbf{u}} = \mathbf{R}_s D_s(\mathbf{u}) \begin{bmatrix} (u - c_x)/f \\ (v - c_y)/f \\ 1 \end{bmatrix} + \mathbf{t}_s, \tag{11}$$

where $(\mathbf{R}_s, \mathbf{t}_s)$ is the camera-to-world pose and (f, c_x, c_y) are the camera intrinsics. The valid points and their RGB values from recent frames form $\mathcal{P}_t = \{(\mathbf{X}_j, \mathbf{c}_j)\}_j$.

At the current position, $\mathcal{P}_t$ is reprojected into $N = 12$ views with headings $\alpha_k = 2\pi k/N$. The projected location of point j in view k is $\mathbf{u}_j^k = \pi(\mathbf{R}_{t,k}^\top(\mathbf{X}_j - \mathbf{p}_t))$, where $\mathbf{R}_{t,k}$ is the camera-to-world rotation of the target view and π uses the same camera intrinsics and coordinate convention as the back-projection above. Gaussian-weighted point splatting computes

$$w_j^k(\mathbf{u}) = m_j^k(\mathbf{u}) \exp\left(-\frac{\|\mathbf{u} - \mathbf{u}_j^k\|_2^2}{2\sigma^2}\right), \widehat{I}_t^k(\mathbf{u}) \quad = \frac{\sum_j w_j^k(\mathbf{u})\mathbf{c}_j}{\sum_j w_j^k(\mathbf{u})}. \tag{12}$$

Here, σ is a fixed kernel bandwidth in pixels, and $m_j^k(\mathbf{u})$ is a binary mask that restricts contributions to the local splat footprint and enforces nearest-surface depth filtering. The RGB expression is evaluated only when $\sum_j w_j^k(\mathbf{u}) > 0$; pixels with zero accumulated weight are marked invalid by C_t^k. Depth at each pixel is taken from the nearest valid projected point. The current front RGB-D view is retained, while the remaining eleven views are reconstructed. Insufficient coverage triggers an active scan. SEVA (Zhou et al., 2025) fills residual RGB gaps without changing depth or geometric validity.

B.2 Depth Normalization

Given the relative depth maps $\{\widehat{D}_t^k\}_{k=0}^{N-1}$ predicted jointly by DA3 (Lin et al., 2025), we normalize them for the frozen waypoint predictor. A shared scale is applied across all views:

$$\widetilde{D}_t^k(\mathbf{u}) = \mathrm{clip}_{[0,1]}\left(\frac{d_{\mathrm{ref}}}{d_{\max}} \frac{\widehat{D}_t^k(\mathbf{u})}{m_t}\right), \tag{13}$$

where m_t is the median of all valid predicted depths across the panorama, d_{ref} is a fixed reference median, and $d_{\max}$ is the maximum depth range.

B.3 Waypoint-Conditioned Foresight

OmniRoam Liu et al. (2026b) predicts one future view for each motion candidate, with at most five candidates per step. The four cardinal RGB views are converted to a 960×480 equirectangular panorama, with gray padding for the unobserved top and bottom faces. For candidate $w_t^i = (\phi_t^i, d_t^i)$, we rotate the panorama to align its bearing with the forward direction. We then construct 21 fixed-orientation camera poses with uniformly spaced translations from zero to d_t^i meters.

The aligned panorama is repeated over 81 conditioning frames. We use OmniRoam Preview with the Wan2.1-T2V-1.3B (Wan et al., 2025) backbone, 30 denoising steps, guidance scale 5.0, and seed 0. The generator receives RGB and camera poses, without depth. A 224×224, 90° forward crop from the final frame is paired with the corresponding candidate identifier; Stop has no prediction. Generation is precomputed offline, and the selector loads the candidate-aligned images. Simulator-rendered future views are used only for diagnostic checks, never as substitutes for generated predictions.

B.4 Stop-Time Goal Verification

Verification runs after the selector proposes Stop. An external parser extracts the target description, which LocateAnything (Wang et al., 2026b) grounds in the four cardinal RGB views. By default, the highest-confidence valid grounding supplies an image point or box center. Its bearing and range are recovered from the grounded pixel and the median depth in a 7×7 patch, using a default depth-to-meter scale of 10. Filtering requires confidence of at least 0.02 and four valid depth samples within $[0.1, 8.0]$ meters.

Depending on the configured mode, valid evidence can retain Stop, select an existing waypoint compatible with the target semantics and bearing, or trigger a bounded docking correction. The default R2R confirmation range is 3.0 m. Waypoint continuation preserves the selector's original ranking among eligible candidates. The optional docking mode defaults to a 0.75 m stand-off, a 2.0 m correction cap, 0.25 m motion increments, and one attempt. Without temporal rechecking, the next decision is forced to Stop. Failed grounding or consistency checks retain the original Stop. Verification never reads the evaluator's ground-truth goal location or success metric.

C More Ablations

C.1 Foresight and Verification on OVON

Table 5 evaluates foresight and verification on HM3D-OVON Val-Unseen. All variants use the same fixed Stage 1 checkpoint and native panoramic RGB-D inputs, bypassing perception tools. The waypoint proposer, local controller, evaluation episodes, and action budgets are held fixed. This comparison examines whether tool benefits depend on the navigation task.

Table 5: Foresight and verification ablations on HM3D-OVON Val-Unseen using the same fixed Stage 1 checkpoint. All variants use native panoramic RGB-D inputs and bypass perception tools. – denotes a disabled or unnecessary tool.

	Tools				OVON Val-Unseen	
Setting	Depth recovery	Panorama reconstruction	Foresight	Verification	SR↑	SPL↑
(a) Foresight and verification: panoramic RGB-D						
Full PIV	–	–	✓	✓	51.16	33.51
w/o verification	–	–	✓	–	49.72	32.95
w/o foresight	–	–	–	✓	51.23	33.34
w/o both	–	–	–	–	49.67	32.64

C.2 Training Data and Training Stages

Table 6 reports the training-data composition and performance of four configurations. T1–T3 use standard supervised data. T4 adds DAgger data and follows the Stage 2 procedure, restarting from the pretrained Qwen3-VL (Bai et al., 2025) checkpoint rather than continuing from T3. Its results therefore reflect the complete second-stage training recipe, including additional data and a longer training schedule.

Adding ScaleVLN (Wang et al., 2023) improves VLN performance (T1→T2). Further adding ObjectNav (Chang et al., 2017; Yadav et al., 2023) and OVON data (Yokoyama et al., 2024b) yields larger gains on the object-goal benchmarks, while also slightly improving VLN SR and SPL (T2→T3). These results support joint training across navigation tasks.

Table 6: Training-data and stage ablations. The training mixtures are cumulative. SR, SPL, and OS are percentages; NE is in meters. OS denotes oracle success rate. T4† reproduces the main-paper results, using the panoramic VLN setting.

Model	Training data	SR↑	SPL↑	OS↑	NE↓
R2R-CE Val-Unseen					
T1	R2R + RxR	59.7	55.7	65.4	4.80
T2	T1 data + ScaleVLN	66.6	62.5	71.3	4.11
T3	T2 data + ObjectNav + OVON	69.0	63.5	72.6	3.87
T4†	T3 data + DAgger	71.6	63.1	77.5	3.45
RxR-CE Val-Unseen					
T1	R2R + RxR	58.3	47.6	65.1	5.95
T2	T1 data + ScaleVLN	62.6	51.5	70.4	5.47
T3	T2 data + ObjectNav + OVON	63.3	52.5	70.3	5.21
T4†	T3 data + DAgger	69.2	54.5	78.8	4.07
ObjectNav-HM3D Val					
T1	R2R + RxR	50.6	31.7	56.4	3.44
T2	T1 data + ScaleVLN	53.4	35.0	58.0	3.41
T3	T2 data + ObjectNav + OVON	62.4	41.6	65.4	2.89
T4†	T3 data + DAgger	75.2	45.6	78.8	2.02
HM3D-OVON Val-Unseen					
T1	R2R + RxR	45.2	28.4	54.0	4.05
T2	T1 data + ScaleVLN	45.2	29.1	52.5	4.03
T3	T2 data + ObjectNav + OVON	51.2	33.5	58.9	3.61
T4†	T3 data + DAgger	56.7	33.2	66.1	3.02

D Failure Analysis

Beyond aggregate evaluation, we inspect individual trajectories to identify limitations in instruction following. Figure 5 presents a qualitative failure of PIV in which the agent misses a landmark-conditioned right turn despite having an appropriate waypoint candidate.

Missing the right turn beside the island. The instruction requires the agent to pass through the left doorway and then turn right by the island. After entering the kitchen, the agent has a right-side candidate A at Step 03 but selects the forward candidate E instead. It consequently passes the turning point, increasing its geodesic distance to the goal from 1.386 to 2.495 m. Another forward move increases this distance to 3.973 m. At this decision point, the error is therefore not the absence of a right-turn candidate, but the failure to select it according to the instruction. The subsequent docking query at Step 06 still refers to "the doorway on your left," an earlier instruction clause, and the agent stops at Step 07 without recovering the omitted turn. This behavior suggests insufficient alignment between instruction progress and waypoint selection, motivating explicit checks of unfulfilled directional clauses before both movement and termination.

Instruction fidelity versus reported success. The replay reports `Strict success = 1`, with a final goal distance of 2.795 m and an nDTW of 0.364. We therefore characterize this example as an instruction-following failure, rather than a failure under the replay's reported success criterion: ending near the goal does not establish that the required turn was executed at the correct landmark.

E Limitations and Future Work

PIV relies on a compact Qwen3-VL model fine-tuned for navigation and a predefined execution harness, with limited support for open-ended planning and adaptive tool use. Future work will explore more adaptive agent architectures built around larger foundation models to better support task planning, tool orchestration, and navigation decisions. External

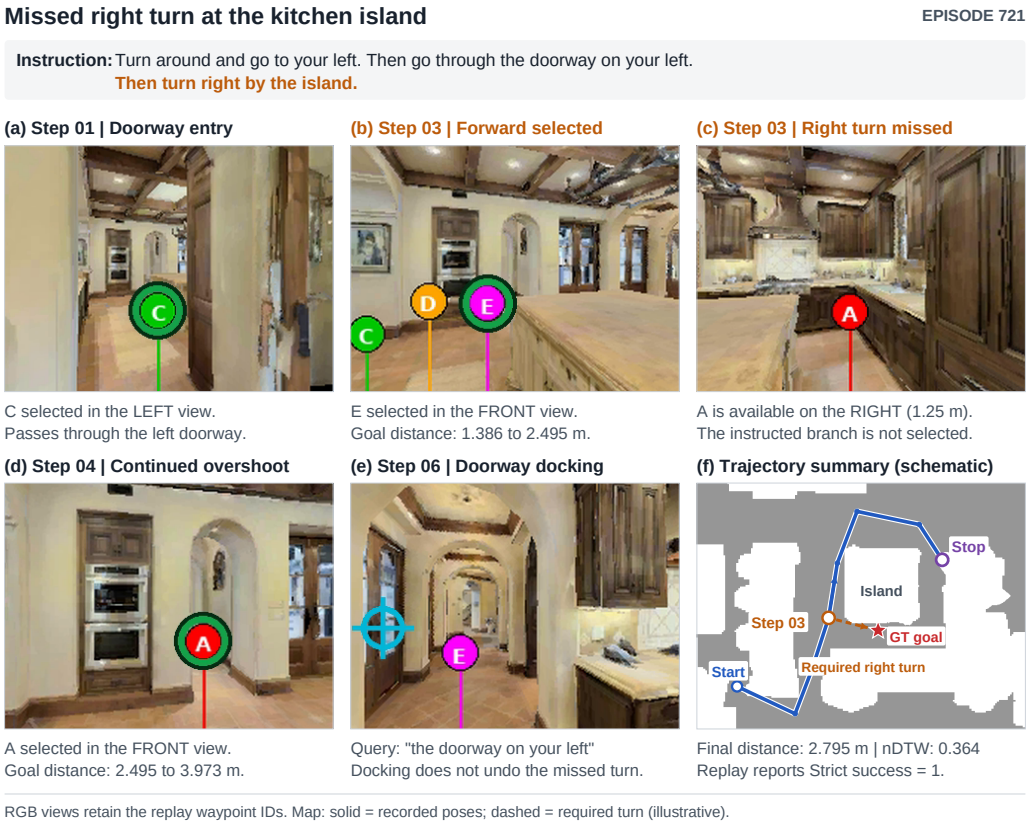

Figure 5: **A missed-right-turn case in PIV (Episode 721).** (a) The agent passes through the left doorway. (b–c) At Step 03, it selects the forward waypoint E rather than the available right-side waypoint A, missing the instruction "turn right by the island." (d) It continues past the turning point. (e) A later docking action queries the earlier doorway clause and does not recover the omitted turn. (f) The schematic connects recorded decision-step positions; the dashed arrow illustrates the required turn, not a ground-truth trajectory. RGB observations and waypoint IDs are retained from the replay. Although the replay reports success, the trajectory does not faithfully execute the instruction.

tools can also introduce unreliable predictions, while stop-time verification does not explicitly track instruction progress and may leave earlier route errors uncorrected. We will investigate uncertainty-aware evidence integration and trajectory-level verification to support more reliable decisions and timely recovery. Finally, our evaluation is simulation-based, with foresight precomputed offline. Efficient online tool execution, selective tool invocation, and physical-robot evaluation remain important directions for practical deployment.