

RiverVLN: Phase-Grounded Temporal Vision–Language Navigation for Unmanned Surface Vehicles

Jieling Wu, Yuehao Huang, Jiajun Lv, Tao Huang, Yong Liu, Weiwei Liu

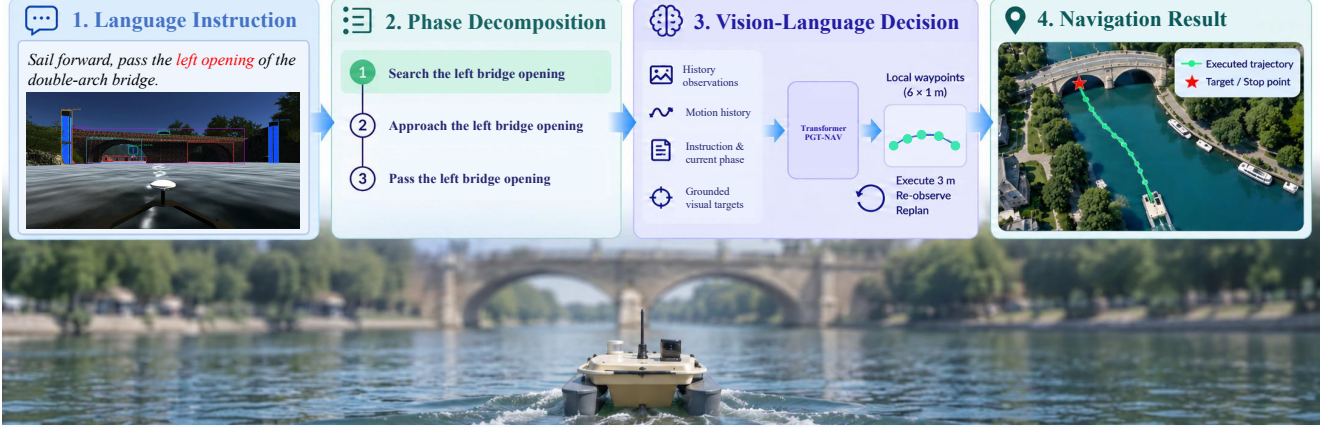


Fig. 1. **Overview of RiverVLN.** A natural-language instruction is decomposed into ordered semantic phases, and phase-grounded visual–motion context is used to predict short-horizon waypoints for closed-loop USV navigation.

Abstract—Vision–language navigation (VLN) has largely been developed for indoor and terrestrial robots, where language can often be treated as a static goal and motion is approximated by discrete or near-instantaneous actions. These assumptions break down for unmanned surface vehicles (USVs): river navigation requires continuous motion under inertia and limited maneuverability, while long-horizon instructions must be executed through sparse and visually ambiguous maritime landmarks. We introduce RiverVLN, to our knowledge the first benchmark designed for long-horizon USV VLN under continuous riverine motion, and PGT-NAV, a phase-grounded temporal navigation framework for USVs. Rather than directly mapping an entire instruction to motion, PGT-NAV converts it into an ordered sequence of visually verifiable semantic phases and maintains the active phase online through grounded visual and motion evidence. This explicit semantic progress state is fused with visual–motion history and phase-specific grounding to predict six local SE(2) pose increments. The resulting trajectory is executed in a predict–execute–re-observe loop, where the vessel executes toward W_3 , updates phase and grounding, and replans through a map-based safety layer. Experiments show that PGT-NAV substantially reduces recursive position and heading drift relative to GNM-style and ViNT-style baselines and achieves an average success rate of 0.79 in Unity–ROS closed-loop navigation. Unseen bridge-opening trials and real-world USV experiments further demonstrate that the phase-grounded representation transfers from controlled evaluation to physical USV deployment.

I. INTRODUCTION

Unmanned surface vehicles (USVs) are increasingly deployed for environmental monitoring, waterway inspection, and infrastructure-related operations [1], [2]. As riverine applications expand, natural-language instructions provide an

intuitive interface for specifying navigation tasks without requiring operators to define low-level waypoints or control commands. Such tasks often require considerably more than reaching a destination: a vessel may be instructed to follow a channel, select one bridge opening among visually similar alternatives, pass semantic landmarks in a specified order, turn relative to a directional cue, and finally stop near a designated target. Successfully executing these instructions therefore requires reasoning about temporal order, object attributes, spatial relations, and terminal conditions, rather than simply navigating toward a fixed goal.

Vision–Language Navigation (VLN) studies how an embodied agent grounds natural-language instructions in first-person visual observations and translates them into navigation decisions. It has been extensively studied in indoor and terrestrial embodied environments [3]–[6] and has also been extended to aerial platforms [7]. Recent embodied vision–language models have further improved instruction following through richer multimodal representations [8]–[12]. These advances provide a strong foundation for language-conditioned navigation, but they do not directly address how semantic instruction progress interacts with the continuous dynamics of a surface vessel.

Extending VLN to river navigation, however, introduces challenges arising from both the characteristics of river environments and the dynamics of USVs. First, river environments often contain sparse semantic landmarks and repetitive visual structure, making visually similar targets difficult to distinguish and a single observation insufficient

to determine whether an instruction-relevant landmark is still being approached or has already been passed. Navigation therefore requires temporal evidence to infer and maintain the current semantic progress of the instruction. Second, USVs exhibit strongly non-instantaneous motion dynamics, including pronounced inertia, lateral drift, delayed stopping, and a finite turning radius. Consequently, an incorrect semantic transition cannot always be corrected by an immediate change of action. For example, prematurely switching from an approach phase to a pass phase, or initiating a terminal maneuver too late, may already place the vessel on an undesirable trajectory. Riverine VLN therefore requires semantic progress estimation and continuous motion prediction to be considered jointly, rather than treating instruction grounding and short-horizon control as largely independent problems.

Studying these coupled challenges requires a benchmark that jointly captures long-horizon language instructions, temporally evolving semantic progress, visual observations, and continuous vessel motion. To this end, we introduce RiverVLN, a Unity–ROS benchmark for long-horizon VLN in continuous river environments. RiverVLN aligns natural-language instructions with visual–motion observations, metric trajectories, phase-level task progress, and instruction-relevant maritime grounding annotations. To our knowledge, it is the first benchmark designed for long-horizon USV VLN under continuous riverine motion. Building on RiverVLN, we propose **PGT-NAV**, a **Phase-Grounded Temporal Navigation** framework. Rather than predicting motion directly from the current observation and the full instruction, PGT-NAV maintains an explicit estimate of the active semantic phase, grounds the phase-relevant target in the current observation, and jointly reasons over visual–motion history to predict a short-horizon metric trajectory.

Our contributions are summarized as follows:

- We introduce RiverVLN, a continuous-river VLN benchmark for USVs that pairs long-horizon navigation instructions with visual–motion observations, metric trajectories, phase-level task progress, and phase-aligned maritime grounding annotations.
- We propose PGT-NAV, a phase-grounded temporal framework that jointly models visual–motion history, explicit semantic progress, and phase-specific grounding to predict short-horizon metric trajectories for continuous USV navigation.
- We evaluate PGT-NAV through offline trajectory prediction, Unity–ROS closed-loop navigation, and real-world USV experiments, demonstrating improved closed-loop performance in simulation and the feasibility of deploying the proposed framework on a real USV.

The dataset, code, and evaluation tools will be publicly released at <https://anonymous.4open.science/r/PGT-NAV-D349>.

II. RELATED WORK

A. Autonomous USV Navigation and Motion Planning

Autonomous USV navigation commonly combines global path planning, collision avoidance, and local control, with

COLREGs-aware risk assessment and replanning central to operation in constrained waterways [1], [2]. Recent systems further account for current fields, energy consumption, and dynamic obstacles through hierarchical planning and optimization [13], [14]. Deterministic methods based on A*, dynamic-window search, and modified artificial potential fields remain attractive because their geometric constraints are interpretable and can be coupled directly with vessel kinematics and safety margins [15], [16]. Learning-based approaches, including deep reinforcement learning, have also been explored for local collision avoidance and COLREGs-compliant decisions [17]. These methods are effective when the navigation objective is represented by waypoints, coordinates, occupancy maps, or explicit rules, but they generally do not model long natural-language instructions containing ordered semantic goals, object attributes, relative directions, and stopping conditions. Our framework therefore does not replace deterministic planning: it retains map-based planning as the safety execution layer and places a language-conditioned semantic navigation layer above it to determine the task-relevant local intent.

B. VLN in Continuous Environments

Vision–language navigation was initially studied on discrete indoor navigation graphs, where agents selected among graph-connected actions. VLN-CE extended instruction following to continuous 3-D environments, exposing the agent to visual interpretation, traversability, and low-level motion execution [3]. Recurrent cross-modal models and history-aware Transformers subsequently improved progress reasoning [4], [5], while waypoint prediction and discrete-to-continuous transfer improved geometric executability [18], [19]. Long-horizon VLN further motivated explicit memory and structured spatial representations: DUET and cross-modal map learning combine local visual reasoning with graph-based or bird’s-eye representations [20], [21], milestone-based navigation decomposes long instructions into ordered intermediate goals [22], and ScaleVLN and GOAT-Bench emphasize data scale, multimodal goals, and long-term memory [23], [24]. Despite these advances, most continuous VLN methods assume terrestrial embodiments whose turning and stopping dynamics differ substantially from those of USVs. River scenes additionally contain sparse maritime landmarks and repetitive structures such as shorelines and bridge openings, making task progress more ambiguous. Our method is related to milestone- and progress-aware VLN, but represents progress explicitly as a semantic phase tied to a grounded maritime target and predicts a short-horizon metric SE(2) trajectory that is repeatedly replanned during execution rather than a sequence of discrete actions.

C. Foundation Models for Language-Guided Navigation

Recent foundation-model-based navigation systems combine pretrained language, vision, and navigation models. LM-Nav couples large language and vision models with a navigation policy for outdoor landmark following [25]. GNM [26] and ViNT [27] learn transferable visual navigation



Fig. 2. **Overview of the RiverVLN benchmark.** (a) Unity river environments and global layouts. (b) Language-grounded trajectories aligned with long-horizon instructions. (c) Phase-aligned visual grounding annotations. (d) Representative tasks including landmark search, bridge selection, and directional navigation.

from heterogeneous robot data, with ViNT using a Transformer to fuse temporal observations and navigation goals. NoMaD extends this family with a diffusion policy for goal reaching and exploration [28]. More recent systems extend beyond image-goal conditioning: NaviLLM, NaVid, and UniNaVid use language- or video-conditioned prediction [8], [9], [29], VLFM couples vision-language semantics with frontier planning [30], and CA-Nav reasons about sub-instruction completion in continuous environments [31]. In contrast, we focus on long-horizon river navigation under USV dynamics and sparse maritime landmarks. We invoke the language model only for initial phase decomposition, while online progress is updated from grounded visual and motion evidence, linking phase-specific grounding and temporal progress tracking to metric waypoint prediction.

III. METHOD

Given a natural-language navigation instruction I , the USV must produce an executable continuous trajectory while respecting the instruction’s semantic order, spatial constraints, and stopping condition. We formulate this as phase-conditioned short-horizon trajectory prediction rather than direct discrete action generation. The overall framework is shown in Fig. 3.

A. RiverVLN Dataset and Task Formulation

Constructing a VLN benchmark for USVs requires addressing three data-level challenges. First, sparse maritime landmarks and language-specified spatial relations make it difficult to associate navigation instructions with the intended visual targets. We address the lack of explicit language-to-target supervision by annotating object identities, bounding boxes, and spatial relations, thereby linking each instruction to its corresponding maritime landmarks. Second, vessel inertia and limited maneuverability require navigation decisions to account for continuous motion rather than isolated observations or discrete actions. We address the lack of motion-aware supervision by recording continuous pose histories under constrained vessel control and deriving metric displacement, heading, and future waypoint labels. Third, long-horizon instructions contain ordered sub-tasks and stopping conditions that cannot be represented by a terminal goal alone. We address the lack of explicit semantic-progress supervision by aligning instructions with ordered phase labels, active targets, and corresponding trajectory segments, including terminal stopping phases.

As shown in Fig. 2, RiverVLN is constructed in Unity-ROS and covers river routes containing bridges with multiple openings, directional signs, colored light gates, shoreline landmarks, and obstacles. Object identities, grounding boxes, and spatial-relation annotations establish the correspondence between language-specified targets and visual observations, including individual bridge openings. A forward-facing RGB camera captures 1280×720 images with a 60° vertical and approximately 91.5° horizontal field of view; images are resized to 512×288 for network input. To retain metric motion information, navigation samples are recorded at approximately 1 m intervals of accumulated travel rather than at fixed time intervals. Reference trajectories are collected using a 20 Hz ROS controller with a maximum commanded speed of 2.0 m/s, a maximum commanded yaw rate of 1.0 rad/s, and a 2.0 m minimum turning-radius constraint during normal forward tracking. Synchronized RGB and pose records preserve the vessel’s motion history and yield body-frame displacement and heading increments, providing continuous SE(2) trajectory supervision. Each route is further paired with natural-language instructions and ordered phase plans. Phase labels, active targets, grounding boxes, spatial relations, and trajectory segments are aligned along the route, identifying the objective of each segment and explicitly representing terminal stopping behavior.

A navigation episode is represented as

$$\mathcal{D}^{(n)} = \left(I^{(n)}, \mathcal{Z}^{(n)}, \{o_t, p_t, m_t, z_t, G_t\}_{t=1}^{T_n} \right), \quad (1)$$

where $I^{(n)}$ and $\mathcal{Z}^{(n)}$ denote the instruction and ordered phase plan, while o_t , p_t , m_t , z_t , and G_t represent the RGB observation, vessel pose, body-frame motion increment, phase label, and phase-specific grounding, respectively. At each decision step, the dataset constructs the PGT-NAV input

$$\mathcal{X}_t = (O_t, \mathcal{M}_t, I, z_t, G_t), \quad (2)$$

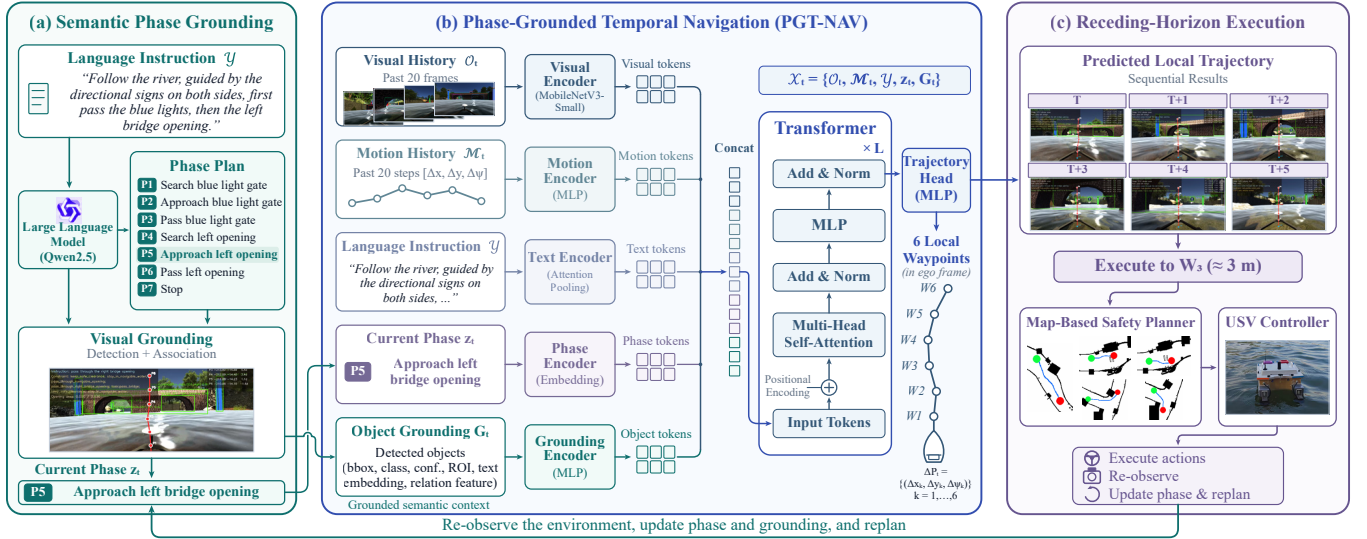


Fig. 3. **Overview of PGT-NAV.** (a) Long-horizon instructions are decomposed into ordered semantic phases and associated with visually grounded targets. (b) Visual–motion history, language, the current phase, and phase-specific grounding cues are fused by a Transformer to predict six local SE(2) waypoints. (c) The controller executes toward W_3 , re-observes the environment, updates phase and grounding, and replans in a receding-horizon loop.

TABLE I
STATISTICS OF THE RIVERVLN BENCHMARK

Annotation Statistics			
Annotation	Count	Annotation	Count
Phase Categories	7	Phase Instances	6,134
Object Categories	16	Grounding Boxes	103,472

Dataset Statistics						
Split	Routes	Samples	RGB	Instr.	Len.	Phases
Train	201	88,388	22,016	804	108.43	5.94
Val	36	13,172	3,277	144	89.97	5.38
Test-Unseen	23	6,776	1,688	92	72.22	6.35
Total	260	108,336	26,981	1,040	102.67	5.90

where O_t and M_t contain the latest $H = 20$ visual and motion observations, with missing history masked. Six future local SE(2) waypoint increments are derived from reference poses as prediction targets. Consequently, each sample jointly provides language-grounded visual targets, continuous motion context, and phase-aligned trajectory supervision, enabling PGT-NAV to learn local navigation behavior conditioned on the current semantic objective. Table I summarizes the Train, Val, and Test-Unseen splits, comprising 260 routes, 1,040 instructions, 108,336 samples, 6,134 phase instances, and 103,472 grounding boxes.

B. Semantic Phase Grounding

Long-horizon river navigation requires the USV to track which sub-task is currently active rather than treating the instruction as a static destination. At the beginning of each episode, Qwen2.5-7B-Instruct is invoked once to decompose instruction I into an ordered phase plan. Its output is restricted to seven phases: *search*, *approach*, *maneuver*, *pass*, *stop*, *stop-before-target*, and *stop-near-target*. The decomposition pre-

serves directional, color, landmark, and stopping constraints without introducing new targets. For example, “pass through the left bridge opening” yields *search* $\rightarrow$ *approach* $\rightarrow$ *pass* for the specified opening. The active phase provides a local semantic objective, preventing intermediate targets from being confused with the terminal destination.

During execution, detections are associated with the active target using object class and language-specified attributes, and only phase-relevant detections form G_t . Rather than switching phases according to a fixed schedule or a single image, a deterministic state machine evaluates recent visual and motion evidence $\mathcal{E}_t$, including target visibility, normalized box area, horizontal image position, and vessel motion:

$$z_{t+1} = \begin{cases} \text{next}(z_t), & C_{z_t}(\mathcal{E}_t) = 1, \\ z_t, & \text{otherwise,} \end{cases} \quad (3)$$

where C_{z_t} is a phase-specific transition predicate. In particular, target disappearance alone cannot confirm passage because it may result from viewpoint changes or detection failures. Passage completion therefore also requires traveled-distance evidence, while directional maneuvers incorporate accumulated heading change.

Search-to-approach transitions use persistent target visibility and scale, whereas approach-to-pass transitions additionally consider image position and vessel motion. To suppress detection noise, ordinary transitions require at least five positive observations among the latest eight and a minimum phase age of 15 observations. Ordering guards prevent future targets from advancing the current phase prematurely. The manager thus updates semantic progress online without repeated LLM calls. Ground-truth phase labels are used for training supervision and evaluation metrics only, never for online phase selection.

C. Phase-Grounded Temporal Trajectory Prediction

Due to vessel inertia and limited maneuverability, identical visual observations may require different trajectories depending on recent motion. Inspired by ViNT [27], we condition local trajectory prediction on visual and motion histories, instruction I , active phase z_t , and phase-specific grounding G_t . The phase specifies the immediate objective, grounding identifies its visual target, and motion history captures recent translation and rotation. These modalities are fused by a Transformer:

$$\begin{aligned} X_t &= [E_v(O_t), E_m(M_t), E_I(I), E_s(z_t), E_g(G_t)], \\ F_t &= \text{Transformer}(X_t), \end{aligned} \quad (4)$$

where the encoders produce modality-specific tokens and O_t, M_t contain the latest $H = 20$ observations. This temporal conditioning allows the policy to account for ongoing vessel motion instead of relying on a single image.

The trajectory head predicts six body-frame SE(2) increments:

$$\hat{\Delta P}_t = f_{\text{traj}}(F_t) = \{(\Delta x_k, \Delta y_k, \Delta \psi_k)\}_{k=1}^6, \quad (5)$$

with approximately 1 m of progress per step. The increments represent a local trajectory independent of global-map coordinates, while their accumulated waypoints describe the short-horizon turning pattern. Unlike predicting isolated positions or discrete steering actions, this representation provides a sequence of displacement and heading changes for subsequent execution. Map-based obstacle handling remains the responsibility of the downstream safety planner.

Training jointly supervises the predicted increments, their accumulated waypoints, and the execution anchor W_3 . Position errors use Smooth-L1 ($\beta = 1$), and heading errors use the circular distance $e_\psi = |\text{atan2}(\sin(\hat{\psi} - \psi), \cos(\hat{\psi} - \psi))|$. The three terms use component weights (1, 3, 12), (1, 3, 8), and (1, 4, 16), respectively. The overall objective is

$$\begin{aligned} \mathcal{L} &= \mathcal{L}_\Delta + 0.5\mathcal{L}_W + 2\mathcal{L}_{\text{exec}} \\ &+ \mathcal{L}_{\text{straight}} + 0.25\mathcal{L}_{\text{relation}} + \mathcal{L}_{\text{stop}}, \end{aligned} \quad (6)$$

where the auxiliary terms regularize near-straight motion, target alignment, and stopping behavior. The execution-anchor loss emphasizes W_3 , which defines the execution target, aligning training with the receding-horizon strategy.

D. Receding-Horizon Closed-Loop Execution

Executing the full predicted trajectory open loop can accumulate motion errors before the USV receives new observations. We therefore execute only part of each prediction and replan from the updated vessel state, allowing the trajectory to adapt to both motion deviations and changes in the active semantic objective.

At each decision step, the six predicted SE(2) increments are converted into local waypoints spanning approximately 6 m. The third waypoint W_3 , approximately 3 m ahead, serves as the execution anchor. The predicted corridor and known map are passed to a safety planner [13], [16], whose path is tracked by the low-level controller. The policy thus provides

language-conditioned motion intentions, while planning and control handle map-based obstacles and path tracking.

After approximately 3 m of execution, the USV acquires new visual and motion observations, updates its phase and target grounding, and predicts a trajectory. The unexecuted waypoints from the previous prediction are discarded. As shown in Fig. 3, this forms a *predict 6 m* $\rightarrow$ *execute 3 m* $\rightarrow$ *re-observe* $\rightarrow$ *replan* loop, continuously revising the trajectory according to updated motion and semantic progress.

IV. EXPERIMENTS

We evaluate PGT-NAV at three complementary levels. Offline recursive rollout measures trajectory and heading stability under repeated predictions, ablations isolate the contributions of phase, temporal history, and grounding, and Unity-ROS closed-loop evaluation and real-world USV trials test whether these gains translate to executable navigation.

A. Experimental Setup and Protocol

Train is used for optimization, Val for checkpoint selection and offline evaluation, and Test-Unseen only for generalization evaluation. In offline rollout, the first state is anchored to ground truth and each subsequent decision uses the predicted W_3 pose as the next anchor at approximately 3 m intervals. Because the RGB stream remains on the recorded reference trajectory, these errors measure sensitivity to recursive pose drift rather than closed-loop navigation success. GNM-style and ViNT-style retain their visual-navigation principles but are adapted to the same history length, decision interval, and six-waypoint output representation; neither receives the semantic phase or phase-specific grounding available to PGT-NAV, and all methods are evaluated on the same Val trajectories and rollout anchors.

For Unity-ROS closed-loop evaluation, the high-level policy runs every 2 s and the low-level controller at 20 Hz, with maximum USV speed limited to 1.5 m/s. The forward RGB camera publishes 1280×720 images at 10 Hz, which are resized to 512×288 before inference. The policy uses 20 history frames, a frozen pretrained MobileNetV3-Small visual backbone, and a 3-layer Transformer with 256-dimensional embeddings and four attention heads. Training uses AdamW for up to 200 epochs with batch size 8, gradient accumulation 4, and learning rate and weight decay of 10^{-4} ; the checkpoint with the best validation performance is used for evaluation.

B. Offline Prediction and Ablation

We report Teacher ADE, Anchor Yaw Error, Recursive ADE/FDE, and Execution Yaw Error. The first measures prediction from a ground-truth anchor; the remaining metrics measure positional and heading drift under recursive rollout. Because the RGB stream remains on the recorded trajectory, recursive errors are not interpreted as closed-loop success.

Table II shows that GNM-style and ViNT-style achieve low teacher-forced error but accumulate substantially larger recursive drift. In contrast, PGT-NAV maintains lower recursive position and heading errors. Removing phase or temporal history raises Recursive ADE/FDE to roughly 13–14 m,

TABLE II
OFFLINE COMPARISON AND ABLATION ON THE RIVERVLN VALIDATION SPLIT

Method	Teacher ADE (m)	Anchor Yaw Err. (°)	Rec. ADE (m)	Rec. FDE (m)	Exec. Yaw Err. (°)
GNM-style [26]	0.0696	7.8054	6.2387	6.5177	8.0644
ViNT-style [27]	0.0476	11.2238	9.8883	10.2982	11.7040
w/o Phase	0.1247	13.9324	13.3150	13.8202	14.4716
w/o Temporal	0.1269	13.9265	13.7390	14.2165	14.4441
w/o Grounding	0.0557	9.1439	6.7696	7.0825	9.5724
PGT-NAV	0.0858	3.2770	2.8171	2.9080	3.4283

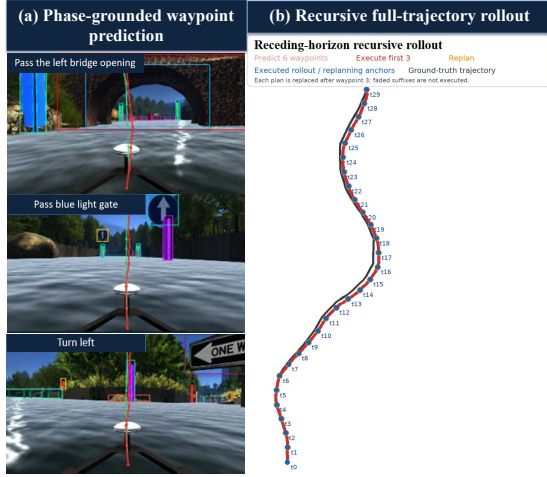


Fig. 4. **Local waypoint prediction and recursive rollout.** The examples on the left show phase-conditioned local predictions; the global rollout on the right visualizes error propagation under repeated prediction and execution anchors.

while removing grounding increases Recursive ADE/FDE to 6.770/7.083 m despite a lower Teacher ADE. The gain therefore lies primarily in maintaining semantic-motion consistency across repeated decisions rather than in minimizing single-step regression error.

Figure 4 illustrates the effect: local predictions remain aligned with the intended bridge, light gate, or turn, whereas recursive rollout exposes accumulated state error. This motivates re-observation after each 3 m execution segment.

C. Closed-Loop Navigation in Unity

TABLE III
UNITY-ROS CLOSED-LOOP PERFORMANCE

Task	SR	SCR	PLR	Coll.	WPR	G-Acc.	P-Acc.
Long	0.60	0.745	1.03	0.40	0.10	0.945	0.65
Bridge	0.95	0.970	0.99	0.00	0.05	0.905	0.95
Left	0.85	0.960	1.06	0.15	0.05	0.980	0.95
Right	0.90	0.980	1.00	0.05	0.05	0.980	0.95
L-Turn	0.70	0.925	0.84	0.05	0.25	0.990	0.78
R-Turn	0.75	0.938	0.97	0.10	0.15	0.990	0.81
Avg.	0.79	0.920	0.98	0.13	0.11	0.965	0.85
Test-Unseen	0.80	0.913	0.95	0.10	0.05	0.95	0.95

Note: Each task is evaluated over 20 trials.

In closed-loop simulation, each predicted local trajectory is executed before the next camera observation is acquired and the phase and grounding states are updated. We report success rate (SR), stage completion rate (SCR), path-length ratio (PLR), collision rate (Coll.), wrong-passage rate (WPR), grounding accuracy (G-Acc.), and phase accuracy (P-Acc.). A trial is successful only when all instruction-critical stages and the terminal condition are satisfied without collision; SCR measures the fraction of completed semantic stages, and PLR is defined as $L_{\text{exec}}/L_{\text{ref}}$, so a value below one may also indicate early termination. WPR records entry into an incorrect bridge opening, gate, or instruction-inconsistent direction.

Table III gives an average SR of 0.79 and SCR of 0.920 across the six closed-loop task categories. Bridge and left/right-opening tasks achieve 0.85–0.95 SR, whereas turning tasks exhibit higher WPR. Long-distance navigation is the most difficult category, with an SR of 0.60. These routes average 206.1 m in length and contain 6.95 phases, exposing the system to more phase transitions and accumulated motion error, although SCR and G-Acc. remain 0.745 and 0.945, respectively. For the unseen closed-loop study, we conduct 20 trials on held-out left/right bridge-opening scenarios using manually specified navigation instructions. These trials form a closed-loop evaluation subset of Test-Unseen, which contains 23 routes with 6.35 phases per route on average. On this evaluation, PGT-NAV reaches 0.80 SR and 0.913 SCR, with both grounding and phase accuracy at 0.95; collision and WPR remain low at 0.10 and 0.05, respectively. Together, the results suggest that maintaining reliable phase transitions and motion execution over long routes remains more challenging than completing individual bridge-opening tasks.

TABLE IV
CLOSED-LOOP COMPARISON AND ABLATION ON LEFT/RIGHT BRIDGE-OPENING TASKS

Method	SR↑	SCR↑	PLR	Coll.↓	WPR↓
GNM-style	0.25	–	0.88	0.70	0.15
ViNT-style	0.10	–	0.97	0.90	0.00
w/o Phase	0.00	–	–	1.00	0.30
w/o Grounding	0.15	0.25	1.21	0.85	0.40
PGT-NAV	0.875	0.97	1.03	0.10	0.05

Table IV compares adapted visual-navigation baselines and key ablations on left/right bridge-opening tasks. “–” denotes an unavailable metric; SCR is omitted for methods without

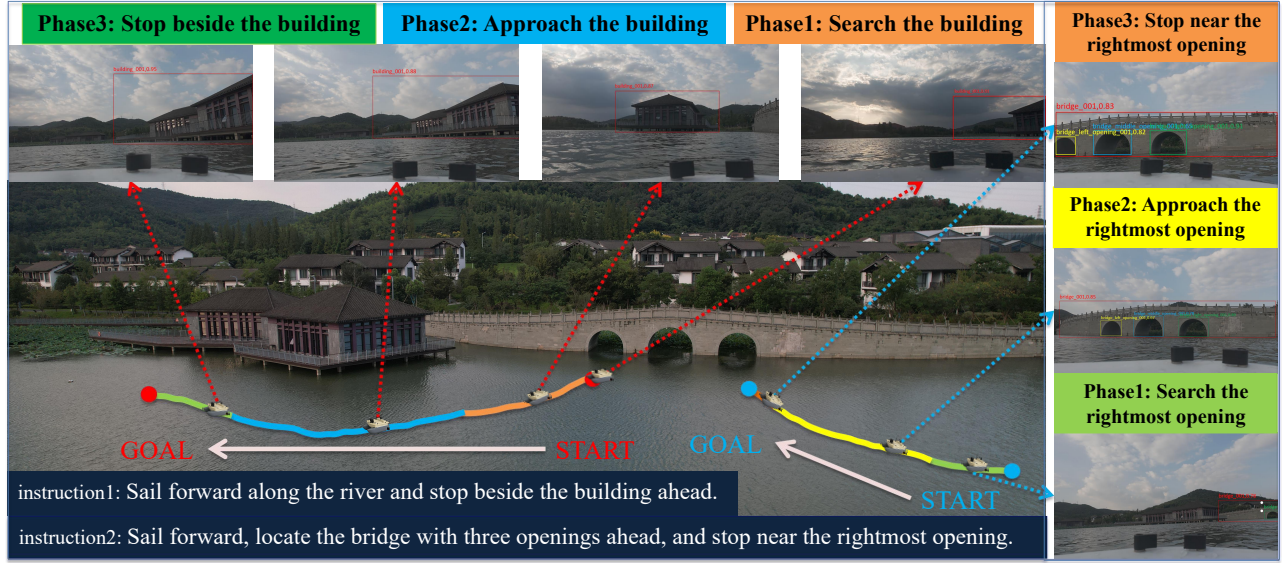


Fig. 5. **Real-world closed-loop navigation of PGT-NAV.** The global trajectories and phase-aligned first-person observations show two representative tasks: stopping beside a building and stopping near the rightmost opening of a three-opening bridge.

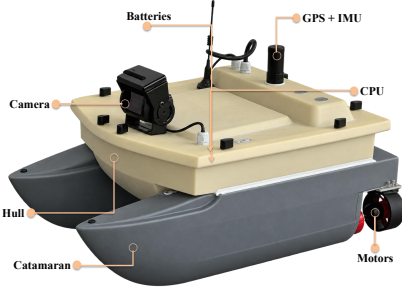


Fig. 6. Physical USV platform developed for real-world PGT-NAV evaluation, integrating an RGB camera, GPS/IMU, onboard CPU, battery system, and dual motors.

explicit phase-state tracking, and PLR for w/o Phase because no trial succeeds. PGT-NAV achieves 0.875 SR with a 0.10 collision rate, compared with 0.25/0.70 for GNM-style and 0.10/0.90 for ViNT-style. ViNT-style’s WPR of 0.00 is not meaningful because most trials end in collision before a wrong passage is recorded. Removing the phase state yields no successful trials, while removing grounding reduces SR/SCR to 0.15/0.25 and raises WPR to 0.40. Together with the offline ablation, these results show that explicit phase progress and phase-target association improve language-conditioned navigation reliability.

D. Real-World USV Experiments

To evaluate deployment beyond simulation, we built the physical catamaran shown in Fig. 6, integrating a forward RGB camera, GPS/IMU, onboard CPU, battery system, and dual motors. The onboard stack retains the same phase definitions and predict-six/execute-three interface used in simulation and runs perception, online phase update, trajectory prediction, and ROS communication without task-specific changes to the high-level policy.

We evaluate two language-conditioned tasks: stopping

beside a specified building and locating a three-opening bridge before stopping near its rightmost opening. Figure 5 shows the executed routes and phase-aligned observations. In both cases, the system progresses from target search to approach and terminal stopping despite reflections, sensor noise, vessel inertia, and control deviations, providing system-level evidence that the semantic phase representation and local trajectory interface can operate on the physical USV without task-specific scripting.

E. Discussion

Overall, the experiments indicate that the main benefit of PGT-NAV lies in maintaining semantic and motion consistency across repeated decisions rather than merely improving single-step waypoint prediction. The gap between teacher-forced and recursive evaluation shows that low local prediction error does not necessarily prevent accumulated position and heading drift. The ablation results further suggest that phase tracking, temporal context, and target grounding make complementary contributions to stable trajectory prediction. Closed-loop evaluation confirms that these improvements translate into higher navigation success and lower collision rates than the adapted visual-navigation baselines. The relatively lower performance on long multi-stage routes also highlights the importance of maintaining semantic progress and motion accuracy over successive replanning cycles.

The real-USV experiments further show that the phase-grounded representation and receding-horizon execution strategy can be transferred from simulation to the physical platform without task-specific changes to the high-level policy. Across the evaluated tasks, the system maintains semantic progress and completes language-conditioned navigation despite reflections, sensing noise, vessel inertia, and imperfect tracking. These results provide system-level evidence that the proposed framework supports continuous-motion navigation

on a physical USV.

V. CONCLUSION

We presented RiverVLN and PGT-NAV for long-horizon vision–language navigation of unmanned surface vehicles in continuous river environments. Rather than mapping the full instruction directly to motion, PGT-NAV decomposes instructions into ordered semantic phases and updates the active phase using visual and motion evidence. By integrating semantic progress, temporal visual–motion context, and phase-specific grounding, the framework predicts short-horizon SE(2) waypoints and executes them through a receding-horizon closed-loop strategy. Experiments demonstrate that PGT-NAV reduces recursive trajectory and heading drift and achieves higher closed-loop navigation success than adapted visual-navigation baselines. Evaluation on unseen bridge-opening scenarios further supports its ability to execute language-conditioned navigation beyond the training routes. Real-world USV trials show that the phase-grounded navigation framework can operate on a physical USV without task-specific changes to the policy, demonstrating its applicability to continuous-motion river navigation.

REFERENCES

- [1] L. Hu, H. Hu, W. Naeem, and Z. Wang, “A review on COLREGS-compliant navigation of autonomous surface vehicles: From traditional to learning-based approaches,” *Journal of Automation and Intelligence*, vol. 1, no. 1, 2022, art. no. 100003.
- [2] Ü. Öztürk, M. Akdağ, and T. Ayabakan, “A review of path planning algorithms in maritime autonomous surface ships: Navigation safety perspective,” *Ocean Engineering*, vol. 251, 2022, art. no. 111010.
- [3] J. Krantz, E. Wijmans, A. Majumdar, D. Batra, and S. Lee, “Beyond the Nav-Graph: Vision-and-language navigation in continuous environments,” in *Computer Vision – ECCV 2020*, 2020, pp. 104–120.
- [4] Y. Hong, Q. Wu, Y. Qi, C. Rodriguez-Opazo, and S. Gould, “VLN BERT: A recurrent vision-and-language BERT for navigation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 1643–1653.
- [5] S. Chen, P.-L. Guhur, C. Schmid, and I. Laptev, “History aware multi-modal Transformer for vision-and-language navigation,” in *Advances in Neural Information Processing Systems*, vol. 34, 2021, pp. 5834–5847.
- [6] H. Zhang, S. Liang, L. Chen *et al.*, “Sparse video generation propels real-world beyond-the-view vision-language navigation,” *arXiv preprint arXiv:2602.05827*, 2026.
- [7] S. Liu, H. Zhang, Y. Qi, P. Wang, Y. Zhang, and Q. Wu, “AerialVLN: Vision-and-language navigation for UAVs,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 15 384–15 394.
- [8] D. Zheng, S. Huang, L. Zhao, Y. Zhong, and L. Wang, “Towards learning a generalist model for embodied navigation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 13 624–13 634.
- [9] J. Zhang, K. Wang, R. Xu *et al.*, “NaVid: Video-based VLM plans the next step for vision-and-language navigation,” in *Proceedings of Robotics: Science and Systems*, 2024.
- [10] Y. Huang, L. Liu, S. Lei *et al.*, “Cogddn: A cognitive demand-driven navigation with decision optimization and dual-process thinking,” in *Proceedings of the 33rd ACM International Conference on Multimedia*, 2025, pp. 5237–5246.
- [11] Y. Huang, Y. Wu, X. Zhang *et al.*, “Wnm-3d: A world navigation model with 3d scene conditioning for closed-loop vln,” *arXiv preprint arXiv:2608.07267*, 2026.
- [12] X. Li, X. Zhang, Y. Huang *et al.*, “Gn0: Toward a unified paradigm for generation, evaluation, and policy learning in visual-language navigation,” *arXiv preprint arXiv:2606.03682*, 2026.
- [13] L. Zhao, Y. Bai, and J. K. Paik, “Global-local hierarchical path planning scheme for unmanned surface vehicles under dynamically unforeseen environments,” *Ocean Engineering*, vol. 280, 2023, art. no. 114750.
- [14] S. Hao, W. Ma, Y. Han, H. Zheng, and D. Ma, “Optimal path planning of unmanned surface vehicle under current environment,” *Ocean Engineering*, vol. 286, 2023, art. no. 115591.
- [15] W. Liu, K. Qiu, X. Yang *et al.*, “COLREGS-based collision avoidance algorithm for unmanned surface vehicles using modified artificial potential fields,” *Physical Communication*, vol. 57, 2023, art. no. 101980.
- [16] S. Hu, S. Tian, J. Zhao, and R. Shen, “Path planning of an unmanned surface vessel based on the improved A-Star and dynamic window method,” *Journal of Marine Science and Engineering*, vol. 11, no. 5, 2023, art. no. 1060.
- [17] N. Wen, Y. Long, R. Zhang, G. Liu, W. Wan, and D. Jiao, “COLREGS-based path planning for USVs using the deep reinforcement learning strategy,” *Journal of Marine Science and Engineering*, vol. 11, no. 12, 2023, art. no. 2334.
- [18] J. Krantz, A. Gokaslan, D. Batra, S. Lee, and O. Maksymets, “Waypoint models for instruction-guided navigation in continuous environments,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 15 162–15 171.
- [19] Y. Hong, Z. Wang, Q. Wu, and S. Gould, “Bridging the gap between learning in discrete and continuous environments for vision-and-language navigation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 15 439–15 449.
- [20] S. Chen, P.-L. Guhur, M. Tapaswi, C. Schmid, and I. Laptev, “Think global, act local: Dual-scale graph Transformer for vision-and-language navigation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 16 537–16 547.
- [21] G. Georgakis, K. Schmeckpeper, K. Wanchoo *et al.*, “Cross-modal map learning for vision and language navigation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 15 460–15 470.
- [22] C. H. Song, J. Kil, T.-Y. Pan, B. M. Sadler, W.-L. Chao, and Y. Su, “One step at a time: Long-horizon vision-and-language navigation with milestones,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 15 482–15 491.
- [23] Z. Wang, J. Li, Y. Hong *et al.*, “Scaling data generation in vision-and-language navigation,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023, pp. 12 009–12 020.
- [24] M. Khanna, R. Ramrakhya, G. Chhablani *et al.*, “GOAT-Bench: A benchmark for multi-modal lifelong navigation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 16 373–16 383.
- [25] D. Shah, B. Osiński, B. Ichter, and S. Levine, “LM-Nav: Robotic navigation with large pre-trained models of language, vision, and action,” in *Proceedings of The 6th Conference on Robot Learning*, vol. 205, 2023, pp. 492–504.
- [26] D. Shah, A. Sridhar, A. Bhorkar, N. Hirose, and S. Levine, “GNM: A general navigation model to drive any robot,” in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 7226–7233.
- [27] D. Shah, A. Sridhar, N. Dashora *et al.*, “ViNT: A foundation model for visual navigation,” in *Proceedings of The 7th Conference on Robot Learning*, vol. 229, 2023, pp. 711–733.
- [28] A. Sridhar, D. Shah, C. Glossop, and S. Levine, “NoMaD: Goal masked diffusion policies for navigation and exploration,” in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 63–70.
- [29] J. Zhang, K. Wang, S. Wang *et al.*, “Uni-NaVid: A video-based vision-language-action model for unifying embodied navigation tasks,” in *Proceedings of Robotics: Science and Systems*, 2025.
- [30] N. Yokoyama, S. Ha, D. Batra, J. Wang, and B. Bucher, “VLFM: Vision-language frontier maps for zero-shot semantic navigation,” in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 42–48.
- [31] K. Chen, D. An, Y. Huang *et al.*, “Constraint-aware zero-shot vision-language navigation in continuous environments,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 47, no. 11, pp. 10 441–10 456, 2025.