# VoLN: Vision-Only Long-Horizon Navigation—Paradigm, Benchmark, and Method

Jiabin Lou[1,2], Haopeng Wang[1,2], Yuanshuai Wang[1], Xinyu Liu[1], Xuxin Lv[1], Yuxin Guo[1], Lei Huang[1], Rongye Shi[1,2], and Wenjun Wu[1,2,*]

[1]Beihang University, Beijing 100191, China

[2]Hangzhou International Innovation Institute, Beihang University, Hangzhou 311115, China

[*]Corresponding author: Wenjun Wu

loujiabin@buaa.edu.cn; wwj09315@buaa.edu.cn

arXiv:2607.21400v1 [cs.RO] 23 Jul 2026

## Abstract

Vision-and-Language Navigation (VLN) enables embodied agents to follow natural-language instructions. However, route-level instructions commonly encode spatial priors, such as orientation, distance, and layout, that are not explicitly available from onboard sensing at deployment in open, GPS-denied environments. Benchmark performance under such interfaces therefore jointly reflects visual navigation ability and the use of route structure explicitly supplied by the task description. As a complementary formulation, we propose Vision-Only Long-Horizon Navigation (VoLN), which shifts route-relevant information from externally supplied instructions and global guidance to locally observable in-scene cues. In VoLN, goal views specify the destination, while route-relevant information is available only through locally observable in-scene cues that the agent must detect, interpret, and select online. We instantiate VoLN for aerial navigation through VoLN-UAV, a 7,210-episode benchmark that combines long-horizon goal-directed flight, continuous 3D motion, large viewpoint changes, and context-dependent beacon selection. We further provide VoLN-MLLM as an initial reference baseline. It aligns self-supervised visual features with a structured semantic space and predicts short-horizon waypoint segments from observation history, goal views, retrieved visual–semantic tokens, and proprioception. On the five-environment Test-Unseen split, it obtains success rates of 7.4%, 4.5%, and 1.8% on Easy, Normal, and Hard episodes, respectively. These results provide an initial evaluation of VoLN and reveal substantial remaining challenges in long-horizon evidence integration, cross-view goal matching, and closed-loop stability. Project page: https://admire-ljb.github.io/VoLN-UAV/.

## 1 Introduction

Vision-and-Language Navigation (VLN) maps high-level semantic instructions into physical actions. Much of its early progress centered on indoor, ground-level agents operating in bounded spaces that are readily described in language and represented by maps or topological graphs. Applying the same instruction interface to aerial agents introduces a different operating regime: navigation unfolds in open 3D space, destinations frequently lie outside the current field of view, and long-range trajectories involve substantial changes in position, altitude, and viewpoint. Route instructions in such settings are commonly authored from trajectories planned with global scene knowledge and encode absolute orientation, metric distance, or route structure. These quantities provide an effective means of specifying a route, but they are not directly observed through onboard sensing at the corresponding decision points. Benchmark performance therefore reflects a combination of visual perception, language grounding, and the use of route structure conveyed by the instruction, making their respective contributions difficult to disentangle.

Figure 1: The instruction-based setting exposes route-level information through language and global guidance, whereas VoLN specifies the destination visually and presents route-relevant cues within the observable scene.

To address this issue, we introduce Vision-Only Long-Horizon Navigation (VoLN). During execution, VoLN removes externally supplied task-level route instructions and global navigation signals, including GPS, global maps, and shortest-path

This work was supported by the National Key Research and Development Program of China under Grant 2025YFF1505704.

annotations, from the policy interface. As illustrated in Fig. 1, goal views specify the destination, whereas route-relevant information is encountered only as locally observable in-scene cues, including semantic beacons. The agent must detect these cues from egocentric observations, interpret their meaning, and select those relevant to the current task online, while proprioception provides onboard motion state.

We instantiate VoLN in aerial navigation, where an Unmanned Aerial Vehicle (UAV) operates in continuous 3D space under substantial viewpoint and scale variation and flight-dynamics constraints. This setting stresses cross-view re-identification, cue selection, and closed-loop control simultaneously. Our benchmark, VoLN-UAV, spans diverse simulated environments and embeds the cue-discrimination problem in the scene itself: active beacons provide route-relevant guidance, while passive beacons with similar visual forms provide structured distractors. Evaluation measures goal convergence, trajectory quality, and closed-loop reliability.

To provide an initial solution, we introduce VoLN-MLLM, a two-stage visual–semantic planning framework. The first stage aligns self-supervised visual features with a structured semantic space, providing comparable representations for observations, goal views, and visible scene cues. The second stage integrates the aligned visual evidence with goal views and proprioception to generate short-horizon UAV trajectories in closed loop. The resulting experiments provide an initial benchmark reference and highlight recurring challenges under viewpoint change, visually similar distractors, and long-horizon execution.

Our contributions are:

- **Task formulation.** We formulate VoLN as a long-horizon navigation paradigm in which goal views specify the destination, while the agent infers route-relevant information online from locally observable scene cues.
- **Benchmark.** We introduce VoLN-UAV, a 7,210-episode benchmark for long-horizon aerial navigation in continuous 3D environments, featuring active and passive semantic beacons and dedicated evaluation splits for seen and unseen environments.
- **Method.** We present VoLN-MLLM, a two-stage visual–semantic planning framework that first aligns observations and goal views with a structured semantic space and then generates short-horizon trajectories through cue-conditioned closed-loop planning.

# 2 Related Work

## 2.1 Navigation task interfaces

A navigation benchmark is shaped by its task interface, the channel through which intent reaches the agent. Language remains the dominant choice in VLN. NavGPT exemplifies explicit language-model reasoning for sequential action prediction Zhou, Hong, and Wu (2024). In aerial multi-agent autonomy, TALKER uses language task descriptions to activate and plan over a learned action-primitive library Lou et al. (2025). MapGPT adds an explicit topological memory for long-horizon planning Chen et al. (2024). Reinforcement post-training for continuous control is explored in VLN-R1 Qi et al. (2025). VLNVerse provides systematic evaluation across models and datasets Lin et al. (2025). NavFoM learns transferable navigation priors from large-scale language supervision Zhang et al. (2026).

Under this interface, benchmark performance jointly reflects perception, language grounding, and route-level information expressed in the instruction. A complementary line of work specifies the navigation goal visually. End-to-end policies learn cross-view correspondence for image-goal reaching Bono et al. (2024), transformer architectures strengthen sequential decision making Pelluri (2024), GaussNav grounds the goal in an explicit 3D Gaussian scene representation Lei et al. (2025), IGL-Nav performs incremental 3D Gaussian localization Guo et al. (2025), and NavigateDiff introduces diffusion-based prediction Qin et al. (2025). These methods primarily study terminal visual-goal grounding. Long-horizon navigation in which route-relevant information must be detected, interpreted, and selected online from locally observable in-scene cues remains comparatively less explored.

## 2.2 Aerial navigation

Aerial navigation research spans two related levels: motion planning and control in open 3D space, and semantic task execution under continuous flight. At the motion level, optimization-based methods explicitly model terrain, obstacle, and flight constraints. HHPSO uses heuristic hybrid particle swarm optimization for real-time quadcopter path planning and validates the resulting trajectories in simulation and real-flight experiments Lou, Ding, and Wu (2024). Learning-based control provides a complementary direction. Swift combines simulation-trained deep reinforcement learning with onboard sensing for agile real-world flight Kaufmann et al. (2023). Air Learning provides an open simulation and gym environment for deep reinforcement learning in resource-constrained visual UAV navigation Krishnan et al. (2021). Air-M further provides a visual-reality many-agent reinforcement learning platform for large-scale training and sim-to-real evaluation of aerial systems Lou et al. (2023).

At the task level, aerial VLN studies how UAVs interpret semantic instructions and execute them through onboard perception and control. AerialVLN provides an early city-scale formulation and data-construction pipeline Liu et al. (2023); OpenUAV emphasizes high-fidelity flight control and assistant-guided evaluation Wang et al. (2025); and OpenFly scales the collection of outdoor instruction–trajectory data Gao et al. (2026). Corresponding methods include the end-to-end multimodal policy of UAV-VLN Saxena, Raghuvanshi, and Goveas (2025), the hierarchical planning and global memory of CityNavAgent Zhang et al. (2025b), and the staged training and interpretable reasoning of FlightGPT Cai et al. (2025).

Overall, aerial navigation has advanced substantially. Exist-

ing aerial VLN benchmarks, however, commonly provide route information explicitly through natural-language instructions or other task-level guidance. Long-horizon aerial navigation in open 3D environments, with locally observable in-scene cues serving as en-route guidance, has received limited attention.

### 2.3 Visual–semantic alignment and planning

Acting on a visually specified goal requires semantic grounding, memory, and foresight. Pretrained vision–language models map observations into shared semantic spaces that support planning: VLFM constructs vision–language value maps for zero-shot target search Yokoyama et al. (2024), PixelNav specifies targets directly in pixel space Cai et al. (2024), and Find Everything balances multiple targets through score-map inference Choi et al. (2025). To maintain evidence over long horizons, Tag Map stores explicit text-based maps Zhang et al. (2025a), E2Map updates maps from experience Kim et al. (2025), and ReMEmbR retrieves from spatio-temporal memory Anwar et al. (2025). Predictive methods add foresight: Imagine-Before-Go completes unseen semantic regions Zhang et al. (2024), while WMNav Nie et al. (2025), ForesightNav Shah et al. (2025), and VISTA Huang et al. (2025) plan over imagined futures. At the system level, AERIS coordinates language-model-based planning and control at runtime Lou et al. (2026).

These studies provide useful foundations for semantic grounding, memory, and predictive planning. However, long-horizon closed-loop navigation remains less explored when the destination is specified visually and route-relevant information must be recovered from locally observable in-scene cues.

## 3 The VoLN Paradigm

We formulate VoLN as a goal-directed, long-horizon embodied navigation paradigm. At execution time, the policy receives no externally supplied task-level route instructions or global navigation signals. Instead, each episode provides a visual goal set $\mathcal{V}$ composed of images captured near the destination, while route-relevant information is available only through locally observable in-scene cues encountered through $o_t$ (Fig. 2). The agent must ground the goal views across changes in viewpoint and detect, interpret, and select relevant cues online during closed-loop interaction.

We model each episode as a partially observable sequential decision process with latent state $s_t \in \mathcal{S}$. At time step $t$, the agent receives an observation $x_t = (o_t, p_t)$, where $o_t$ denotes the egocentric RGB observation and $p_t$ denotes proprioception, implemented as platform-provided onboard signals such as IMU measurements, altitude, velocity, and orientation; GPS and world-frame position are excluded. The agent outputs an action $a_t \in \mathcal{A}$, which may be continuous or discrete and includes an explicit stop decision, and the environment evolves according to the transition kernel $P$ and observation function $\Omega$:

$$s_{t+1} \sim P(\cdot \mid s_t, a_t), \qquad x_{t+1} \sim \Omega(\cdot \mid s_{t+1}). \tag{1}$$

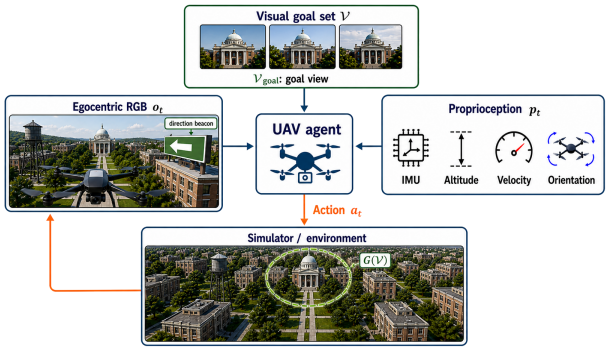


Figure 2: The VoLN interaction. The policy maps goal views $\mathcal{V}$, observations $o_t$, and proprioception $p_t$ to closed-loop actions $a_t$; $G(\mathcal{V})$ serves only for evaluation.

The policy conditions on the interaction history $h_t = (x_0, a_0, \ldots, x_t)$ and selects actions according to

$$a_t \sim \pi(\cdot \mid h_t, \mathcal{V}). \tag{2}$$

VoLN targets episodes in which the destination remains outside the current view over substantial portions of the trajectory and route-relevant cues are encountered at multiple decision points. The policy therefore uses $h_t$ to integrate evidence across the trajectory. As the agent moves, $o_t$ reveals semantic beacons and naturally occurring landmarks at different decision points. The policy interprets these observations in relation to $\mathcal{V}$ and the accumulated interaction context, selecting cues that are relevant to the current task. For evaluation, each task instance associates $\mathcal{V}$ with a goal region $G(\mathcal{V}) \subset \mathcal{S}$. Starting from a designated initial pose, the agent has at most $T$ steps and succeeds by issuing the stop action inside $G(\mathcal{V})$.

## 4 The VoLN-UAV Benchmark

### 4.1 Simulation Environments

VoLN-UAV is built with Unreal Engine and Microsoft AirSim, which provide high-fidelity rendering and UAV simulation across the benchmark environments. The environment pool contains 17 distinct environments drawn from selected scenes adapted from existing open-source aerial VLN benchmarks, including AerialVLN Liu et al. (2023) and OpenUAV Wang et al. (2025), together with additional custom-built environments, $\mathcal{E} = \mathcal{E}^{\text{open}} \cup \mathcal{E}^{\text{custom}}$. This hybrid design preserves compatibility with existing aerial benchmarks while expanding scene diversity. As shown in Fig. 3, the benchmark spans natural and built environments, from deserts, forests, and mountains to urban canyons, tunnels, and industrial corridors, with substantial variation in layout, visibility, altitude change, and landmark density.

### 4.2 Benchmark Construction Pipeline

We instantiate VoLN-UAV through the trajectory-centric pipeline illustrated in Fig. 4. The following paragraphs detail

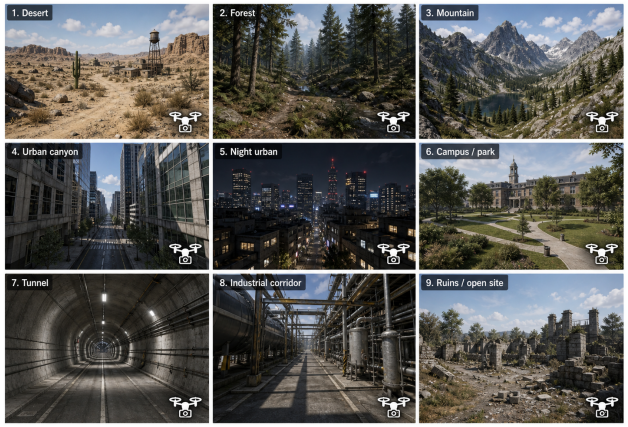


Figure 3: Representative VoLN-UAV environments.

its construction stages and resulting data organization.

**Trajectory collection.** Panels (a) and (e) relate reference-route collection to the resulting path-length distribution. For each episode, we sample a scene from $\mathcal{E}$ and select a reference route from the corresponding route pool. The pool contains predefined routes from existing datasets and custom routes collected by trained human operators following a standardized recording protocol. The predefined trajectories follow the route and goal settings of the source datasets, while the custom trajectories expand coverage of underrepresented flight patterns, including long corridors, sharp turns, altitude transitions, and ambiguous junctions. VoLN-UAV stratifies episodes by accumulated reference-path length, $L_{\text{ref}}(\xi) = \sum_{t=1}^{T} \|\mathbf{r}_t - \mathbf{r}_{t-1}\|_2$, where $\mathbf{r}_t$ denotes the reference position at step $t$. Episodes are labeled **Easy** ($L_{\text{ref}} < L_1$), **Normal** ($L_1 \le L_{\text{ref}} < L_2$), or **Hard** ($L_{\text{ref}} \ge L_2$), with $L_1 = 300\,\text{m}$ and $L_2 = 450\,\text{m}$ across all splits.

**Beacon augmentation and cue categories.** Panels (b) and (f) show beacon placement and the corresponding cue categories. Each reference trajectory is augmented with three to five active beacons, sparsely placed at decision points as task-relevant cues. Each environment is additionally populated with approximately 150 passive beacons that remain fixed across episodes and provide semantic clutter. The beacons span four semantic categories: directional guidance, warning cues related to feasible flight, environmental distractors, and contextual cues whose relevance depends on the current task. During execution, the UAV accesses beacons only through its egocentric observation $o_t$, which may contain both task-relevant active beacons and passive beacons present in the scene.

**Multimodal rollout and annotation schema.** Panels (c) and (g) present the synchronized multimodal rollout and its step-level annotation schema. For each reference trajectory $\xi = \{s_0, \ldots, s_T\}$, the simulator records synchronized egocentric RGB observations $\{o_0, \ldots, o_T\}$ and proprioceptive states $p_t$ at a fixed interval $\Delta t = 2\,\text{s}$. The final three RGB observations form the episode-level visual goal set:

$$\mathcal{V}(\xi) = \{o_{T-2}, o_{T-1}, o_T\}. \tag{3}$$

For each step $t$, the observation and proprioceptive state are paired with the episode-level goal set $\mathcal{V}(\xi)$ to form a training sample, while the following $H$ states along the reference route define the short-horizon waypoint target $W_{t:t+H}$.

**Dataset split and statistics.** Panels (d) and (h) summarize the scene-source split and episode distribution. The dataset contains 7,210 episodes over 17 distinct environments: **Train** contains 5,047 episodes from 12 environments; **Validation-Seen** (VS) contains 1,082 episodes whose trajectories are disjoint from training but are drawn from 5 environments within the training pool; and **Test-Unseen** (TU) contains 1,081 episodes from 5 additional environments belonging to a held-out scene source. The splits correspond to an approximately 70%/15%/15% episode ratio, and the aggregate difficulty mix is 52% Easy, 36% Normal, and 12% Hard.

# 5 The VoLN-MLLM Method

VoLN-UAV couples visual–semantic grounding with closed-loop trajectory generation: the agent must interpret locally observed route cues in relation to the goal views and convert this evidence into executable waypoint segments. We address these requirements with **VoLN-MLLM**, a two-stage visual–semantic planning framework (Fig. 5). Phase I aligns DINO visual features with CLIP's joint image–text space, allowing observations to retrieve relevant concepts from a fixed semantic bank. Phase II employs a pretrained language-model planner. Aligned visual features from the recent observation history and goal views, retrieved visual–semantic tokens, and proprioception are projected into the planner's embedding space and jointly encoded for short-horizon waypoint and stopping prediction. The predicted segment is executed by a low-level flight controller, and the model replans from the subsequent observation. The semantic bank is constructed offline and shared across episodes, while execution follows the VoLN interface without externally supplied task-level route instructions or global navigation signals.

**Visual–Semantic Alignment.** Given an observation $o_t$, a frozen DINO backbone extracts a visual representation. A lightweight trainable adapter, following the cross-space alignment principle of Talking to DINO Barsellotti et al. (2025), maps this representation into the CLIP image-embedding space, producing a normalized student embedding $E_{\text{stu}}$. During training of the adapter, a frozen CLIP image encoder provides the normalized teacher embedding $E_{\text{clip}}$ for the same image. The adapter is optimized by a distillation objective:

$$\mathcal{L}_{\text{distill}} = \ell(E_{\text{stu}}, E_{\text{clip}}), \tag{4}$$

where $\ell(\cdot,\cdot)$ is instantiated as cosine distance. This stage updates only the adapter parameters and keeps the DINO backbone and CLIP encoders fixed.

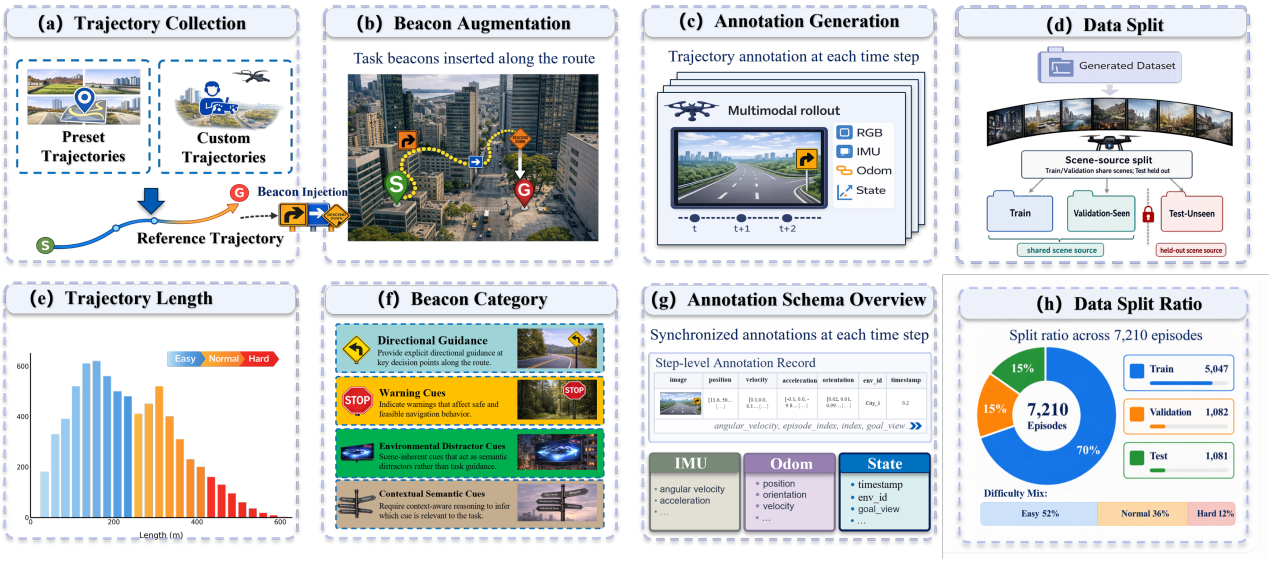


Figure 4: Overview of VoLN-UAV. Panels (a)–(d) show trajectory collection, beacon augmentation, annotation generation, and scene-source splitting; panels (e)–(h) summarize trajectory length, beacon categories, annotation fields, and split statistics.

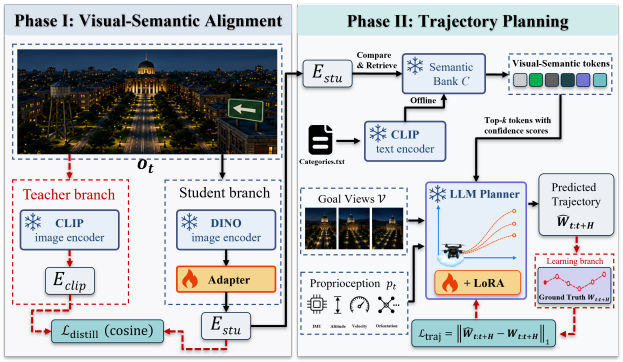


Figure 5: VoLN-MLLM overview. Phase I learns visual–semantic alignment; Phase II predicts short-horizon waypoints and stopping decisions. Dashed arrows indicate training branch; the stopping head is omitted for clarity.

**Visual–semantic tokenization.** We construct a fixed semantic bank $\mathcal{C}$ whose entries are textual category descriptors encoded offline by a frozen CLIP text encoder. At each time step, the aligned visual embedding is compared with the bank embeddings using cosine similarity, and the top-$k$ entries are retained. Their category identifiers are encoded with the planner tokenizer, while their similarity scores are mapped to learned confidence embeddings. The resulting visual–semantic tokens are passed to the planner.

**Trajectory decoding.** At each decision step, the aligned features of a fixed window of recent observations and the goal views are mapped to the planner's embedding dimension by a shared visual projector. A separate state projector maps proprioception $p_t$ to a state token. These embeddings are concatenated with the retrieved visual–semantic tokens and learned structural tokens marking the goal, history, semantics, state, and planning fields. The language-model planner jointly encodes this sequence, and the hidden state of the planning token is passed to a trajectory head and a binary stopping head. The two heads predict a short-horizon sequence of relative waypoints $\hat{W}_{t:t+H}$ in the current UAV body frame and stopping probability $\hat{z}_t$, respectively. The predicted sequence is tracked by the shared low-level flight controller, after which the next observation is acquired and the planner is invoked again.

**Planner adaptation and supervision.** We keep the pretrained language-model backbone frozen and adapt its attention and feed-forward projections using low-rank adaptation (LoRA) Hu et al. (2022). The visual and state projectors, trajectory head, and stopping head are trained jointly with the LoRA parameters. At each step, the trajectory head predicts a segment of length $H$, supervised against demonstrations with an $\ell_1$ loss:

$$\mathcal{L}_{\text{traj}} = \left\| \hat{W}_{t:t+H} - W_{t:t+H} \right\|_1, \tag{5}$$

where $\hat{W}_{t:t+H}$ denotes the predicted waypoint sequence and $W_{t:t+H}$ denotes the corresponding demonstrated sequence. Let $z_t \in \{0, 1\}$ indicate whether the reference state lies inside the success region. The stopping head is trained with binary cross-entropy, and the complete objective is

$$\mathcal{L} = \mathcal{L}_{\text{traj}} + \lambda_{\text{stop}} \, \text{BCE}(\hat{z}_t, z_t). \tag{6}$$

At inference time, the policy stops when $\hat{z}_t$ exceeds a threshold $\tau$ selected on Validation-Seen; otherwise, it executes the predicted waypoint segment and replans.

Table 1: Results on Validation-Seen (VS) and Test-Unseen (TU) across difficulty levels.

| Method | Split | NE/m↓ | | | SR/%↑ | | | OSR/%↑ | | | nDTW/%↑ | | | SPL/%↑ | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Easy | Normal | Hard | Easy | Normal | Hard | Easy | Normal | Hard | Easy | Normal | Hard | Easy | Normal | Hard |
| Random | VS | 268.5 | 308.7 | 398.9 | 0.6 | 0.0 | 0.0 | 1.8 | 0.8 | 0.2 | 28.3 | 20.0 | 12.1 | 0.4 | 0.0 | 0.0 |
| Seq2Seq-VG | VS | 205.8 | 251.6 | 307.4 | 1.2 | 0.5 | 0.1 | 5.4 | 2.9 | 1.0 | 30.1 | 21.0 | 11.8 | 0.9 | 0.3 | 0.1 |
| CMA-VG | VS | 170.2 | 211.9 | 261.3 | 1.9 | 0.9 | 0.2 | 7.6 | 4.3 | 1.9 | 34.5 | 25.2 | 16.1 | 1.3 | 0.6 | 0.1 |
| LAG-VG | VS | 118.7 | 154.9 | 203.6 | 2.8 | 1.5 | 0.5 | 7.8 | 4.6 | 1.9 | 29.7 | 20.4 | 12.7 | 1.9 | 1.0 | 0.3 |
| **VoLN-MLLM** | VS | **92.4** | **126.8** | **171.5** | **8.7** | **5.4** | **2.1** | **13.4** | **10.6** | **4.1** | **54.8** | **40.9** | **25.7** | **6.5** | **3.8** | **1.4** |
| Random | TU | 270.1 | 310.4 | 395.2 | 0.4 | 0.0 | 0.0 | 1.4 | 0.6 | 0.2 | 30.1 | 22.7 | 15.1 | 0.3 | 0.0 | 0.0 |
| Seq2Seq-VG | TU | 208.6 | 254.8 | 309.9 | 1.0 | 0.4 | 0.1 | 4.8 | 2.5 | 0.9 | 28.9 | 21.4 | 13.0 | 0.7 | 0.3 | 0.0 |
| CMA-VG | TU | 174.5 | 216.8 | 266.1 | 1.6 | 0.8 | 0.2 | 6.5 | 3.9 | 1.7 | 33.2 | 26.4 | 18.5 | 1.1 | 0.6 | 0.1 |
| LAG-VG | TU | 122.4 | 158.3 | 206.7 | 2.3 | 1.2 | 0.4 | 6.4 | 3.8 | 1.7 | 28.1 | 20.5 | 14.0 | 1.5 | 0.7 | 0.2 |
| **VoLN-MLLM** | TU | **97.1** | **131.4** | **176.8** | **7.4** | **4.5** | **1.8** | **14.6** | **10.1** | **4.5** | **53.1** | **41.2** | **28.0** | **5.7** | **3.2** | **1.3** |

# 6 Experiments

## 6.1 Experimental Setup

**Implementation details.** All experiments follow the VoLN-UAV protocol (Sec. 4): at each decision step the agent receives only an egocentric RGB observation, proprioception, and the visual goal specification $\mathcal{V}$. World-frame poses are used only for supervision and evaluation and are not exposed to the policy. The shared action interface is waypoint-based: at each decision step the policy emits a segment of $H = 8$ relative three-dimensional waypoints together with a stop signal, a low-level controller tracks the segment, and an episode terminates when the policy stops or the step budget of $T = 128$ decision steps is exhausted. Our reference baseline, VoLN-MLLM (Sec. 5), consists of a frozen DINOv3 ViT-B16 visual backbone, a lightweight adapter that aligns DINO features to the CLIP ViT-B/16 image-embedding space, and a frozen Vicuna-7B-v1.5 planning backbone Zheng et al. (2023). Learned visual and proprioceptive projectors map the permitted task inputs to the Vicuna embedding dimension, and LoRA modules of rank 16 adapt its attention and feed-forward projections. A trajectory head predicts eight such waypoints, while a binary stopping head produces the stop signal. All methods operate in closed loop and share this action interface and stopping criterion.

**Baselines.** We compare VoLN-MLLM with a random policy and three visual-goal (VG) variants of representative instruction-following architectures, replacing their language inputs with the VoLN interface. All learned methods receive the same observation history, goal views, proprioception, and semantic tokens retrieved from the frozen bank, and share the same visual encoder, waypoint action space, and training targets. **Random** samples feasible actions uniformly. **Seq2Seq-VG**, based on Anderson et al. (2018b), uses a recurrent encoder–decoder over visual–proprioceptive history, the goal representation, and retrieved semantic tokens. **CMA-VG**, based on Krantz et al. (2020), attends from the current visual state to history, goal, semantic tokens, and proprioception. **LAG-VG**, based on Liu et al. (2023), separately attends to observation history and semantic tokens before waypoint prediction.

**Evaluation metrics.** Following common practice Krantz et al. (2020), we report Success Rate (SR, stopping inside the $\epsilon = 4\,\mathrm{m}$ goal region), Oracle Success Rate (OSR, entering the success region at any point), Navigation Error (NE, the final Euclidean distance to the goal), normalized Dynamic Time Warping (nDTW) Ilharco et al. (2019), and Success weighted by Path Length (SPL) Anderson et al. (2018a):

$$\mathrm{SPL} = \frac{1}{N}\sum_{i=1}^{N} S_i \cdot \frac{l_i}{\max(p_i, l_i)}\,, \tag{7}$$

where $S_i \in \{0, 1\}$ indicates whether episode $i$ is successful, $l_i$ denotes the shortest-path distance from the start position to the goal (computed offline for evaluation only), and $p_i$ is the actual path length executed by the agent.

## 6.2 Main Results

Table 1 compares VoLN-MLLM with the baselines on Validation-Seen and Test-Unseen across the Easy, Normal, and Hard subsets.

**Overall performance.** VoLN-MLLM yields the highest reported point estimate for every metric in Table 1. On Test-Unseen, its SR reaches 7.4%, 4.5%, and 1.8% across the three difficulty levels, compared with 2.3%, 1.2%, and 0.4% for the strongest baseline, LAG-VG. Performance declines with increasing route difficulty for all methods, but the relative advantage of VoLN-MLLM persists. Nevertheless, the low absolute SR, particularly on the Hard subset, shows that long-horizon visual-only navigation remains challenging.

**Trajectory quality and efficiency.** Beyond terminal success, VoLN-MLLM has lower reported NE, indicating that its final positions are closer to the goal. Its higher reported nDTW corresponds to greater agreement between the executed and reference trajectories. These improvements are observed across

Figure 6: **Successful and failed rollouts.** Top: the trajectory passes the active-beacon locations and enters the goal region. Bottom: the trajectory diverges near the passive billboards and terminates outside the goal region.

Figure 7: Physical testbed and a representative VoLN rollout.

both splits and all three path-length strata. VoLN-MLLM also records the highest SPL, reflecting stronger combined performance in navigation success and path efficiency.

## 6.3 Supplementary Analyses

In the ablation study, we additionally report cycle time (CT), measured from observation input to action output, and execution error rate (EER), defined as the percentage of planning cycles that exceed the time budget or produce invalid outputs.

Table 2: Ablation results on the Test-Unseen split.

| Variant | CT (s)↓ | EER (%)↓ | NE↓ | SR↑ | OSR↑ | nDTW↑ | SPL↑ |
|---|---|---|---|---|---|---|---|
| VoLN-MLLM | 1.42 | **0.5** | **119.0** | **5.7** | **11.8** | **45.8** | **4.3** |
| – No-Align | **1.36** | 0.9 | 162.8 | 2.3 | 7.1 | 29.6 | 1.2 |
| – No-LoRA | 1.45 | 5.8 | 176.9 | 2.8 | 7.8 | 27.2 | 1.4 |
| – CLIP-Input | 1.98 | 1.5 | 158.6 | 2.9 | 8.2 | 30.9 | 1.6 |

**Ablation studies.** On Test-Unseen, we ablate three components. No-Align retains the dimensional projection but removes CLIP-teacher supervision. No-LoRA freezes the planner backbone and removes the LoRA branch, while retaining the trained input projectors and prediction heads. CLIP-Input replaces the DINO backbone with CLIP image encoder. All variants use the same training split, seed pool, and evaluation budget. As shown in Table 2, No-Align causes the largest drop in SR ($5.7 \rightarrow 2.3$), while nDTW decreases from $45.8$ to $29.6$, highlighting the importance of compatibility between visual representations and the semantic space for effective grounding. Removing the LoRA branches yields the highest EER ($5.8\%$) and the lowest nDTW ($27.2$), suggesting that planner adaptation improves output reliability and trajectory fitting. CLIP-Input increases the cycle time from $1.42$ s to $1.98$ s and reduces SR to $2.9\%$. Overall, the results highlight the contributions of visual–semantic alignment to grounding quality, planner adaptation to output reliability and trajectory fitting, and robust visual representations to navigation success and efficient inference.

**Success and failure cases.** Fig. 6 presents one successful and one failed navigation rollout under the same goal exemplar and initial context. The selected frames correspond to comparable stages of the two rollouts. In the successful case, the trajectory passes the active-beacon locations and eventually enters the goal region. In the failed case, the trajectory deviates near the passive billboards and terminates outside the goal region. Together, the two cases qualitatively illustrate trajectory divergence at a corresponding intermediate stage in a beacon-rich environment.

**Physical testbed demonstration.** We conduct a preliminary physical demonstration of the VoLN task interface in a controlled indoor testbed. The testbed comprises a scaled urban scene with roads, a roundabout, building clusters, vegetated terrain, and miniature directional beacons (Fig. 7). At the navigation-policy level, the UAV receives goal views, onboard RGB observations, and proprioception, following the same input interface used in simulation. In one representative rollout, the UAV traverses the roundabout and changes direction near the beacon-marked junction toward the hilltop goal region. This demonstration provides qualitative evidence that the VoLN task interface and closed-loop navigation pipeline can be instantiated on a controlled physical platform.

# 7 Conclusion

This work introduces VoLN, a vision-only long-horizon closed-loop navigation paradigm. During execution, VoLN removes externally supplied task-level route instructions and global navigation signals from the policy interface. Goal views specify the destination, while route-relevant information is available only through locally observable in-scene cues that the agent must detect, interpret, and select online. We instantiate this formulation in VoLN-UAV, a long-horizon aerial navigation benchmark featuring active and passive semantic beacons, continuous 3D control, and a scene-source-held-out test split. VoLN-MLLM provides an initial reference baseline for this task. It maps self-supervised visual features into the semantic space defined by a fixed semantic bank and predicts short-horizon waypoint segments and stopping decisions from observation history, goal views, and proprioception. It produces the highest reported point estimates among the adapted baselines across both evaluation splits and all three difficulty levels. Nevertheless, its success rates on Test-Unseen remain 7.4%, 4.5%, and 1.8% for Easy, Normal, and Hard episodes, respectively. The low success rates show that reliable long-horizon navigation remains unresolved under this interface. Progress requires integrating observational evidence over time, assessing the route relevance of in-scene cues, and limiting the accumulation of local errors during closed-loop execution. Finally, a controlled physical testbed demonstration provides a proof-of-concept instantiation of the VoLN interface beyond simulation.

# References


Anderson, P.; Chang, A.; Chaplot, D. S.; Dosovitskiy, A.; Gupta, S.; Koltun, V.; Kosecka, J.; Malik, J.; Mottaghi, R.; Savva, M.; and Zamir, A. R. 2018a. On Evaluation of Embodied Navigation Agents. arXiv:1807.06757.

Anderson, P.; Wu, Q.; Teney, D.; Bruce, J.; Johnson, M.; Sünderhauf, N.; Reid, I.; Gould, S.; and van den Hengel, A. 2018b. Vision-and-Language Navigation: Interpreting Visually-Grounded Navigation Instructions in Real Environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3674–3683.

Anwar, A.; Welsh, J.; Biswas, J.; Pouya, S.; and Chang, Y. 2025. ReMEmbR: Building and Reasoning over Long-Horizon Spatio-Temporal Memory for Robot Navigation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, 2838–2845. IEEE.

Barsellotti, L.; Bianchi, L.; Messina, N.; Carrara, F.; Cornia, M.; Baraldi, L.; Falchi, F.; and Cucchiara, R. 2025. Talking to DINO: Bridging Self-Supervised Vision Backbones with Language for Open-Vocabulary Segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 22025–22035.

Bono, G.; Antsfeld, L.; Chidlovskii, B.; Weinzaepfel, P.; and Wolf, C. 2024. End-to-End (Instance)-Image Goal Navigation through Correspondence as an Emergent Phenomenon. In *International Conference on Learning Representations (ICLR)*.

Cai, H.; Dong, J.; Tan, J.; Deng, J.; Li, S.; Gao, Z.; Wang, H.; Su, Z.; Sumalee, A.; and Zhong, R. 2025. FlightGPT: Towards Generalizable and Interpretable UAV Vision-and-Language Navigation with Vision-Language Models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 6659–6676. Association for Computational Linguistics.

Cai, W.; Huang, S.; Cheng, G.; Long, Y.; Gao, P.; Sun, C.; and Dong, H. 2024. Bridging Zero-Shot Object Navigation and Foundation Models through Pixel-Guided Navigation Skill. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 5228–5234.

Chen, J.; Lin, B.; Xu, R.; Chai, Z.; Liang, X.; and Wong, K.-Y. K. 2024. MapGPT: Map-Guided Prompting with Adaptive Path Planning for Vision-and-Language Navigation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 9796–9810. Association for Computational Linguistics.

Choi, D.; Fung, A.; Wang, H.; and Tan, A. H. 2025. Find Everything: A General Vision Language Model Approach to Multi-Object Search. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 19936–19943.

Gao, Y.; Li, C.; You, Z.; Liu, J.; Li, Z.; Chen, P.; Chen, Q.; Tang, Z.; Wang, L.; Yang, P.; Tang, Y.; Tang, Y.; Liang, S.; Zhu, S.; Xiong, Z.; Su, Y.; Ye, X.; Li, J.; Ding, Y.; Wang, D.; Wang, Z.; Zhao, B.; and Li, X. 2026. OpenFly: A Comprehensive Platform for Aerial Vision-Language Navigation. In *International Conference on Learning Representations (ICLR)*.

Guo, W.; Xu, X.; Yin, H.; Wang, Z.; Feng, J.; Zhou, J.; and Lu, J. 2025. IGL-Nav: Incremental 3D Gaussian Localization for Image-Goal Navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 6808–6817.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations (ICLR)*.

Huang, Y.; Wu, M.; Li, R.; and Tu, Z. 2025. VISTA: Generative Visual Imagination for Vision-and-Language Navigation. arXiv:2505.07868.

Ilharco, G.; Jain, V.; Ku, A.; Ie, E.; and Baldridge, J. 2019. General Evaluation for Instruction Conditioned Navigation Using Dynamic Time Warping. In *NeurIPS Visually Grounded Interaction and Language (ViGIL) Workshop*.

Kaufmann, E.; Bauersfeld, L.; Loquercio, A.; Müller, M.; Koltun, V.; and Scaramuzza, D. 2023. Champion-Level Drone Racing Using Deep Reinforcement Learning. *Nature*, 620(7976): 982–987.

Kim, C.; Kim, K.; Oh, M.; Baek, H.; Lee, J.; Jung, D.; Woo, S.; Woo, Y.; Tucker, J.; Firoozi, R.; Seo, S.-W.; Schwager, M.; and Kim, S.-W. 2025. E2Map: Experience-and-Emotion Map for Self-Reflective Robot Navigation with Language Models. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, 12811–12817.

Krantz, J.; Wijmans, E.; Majumdar, A.; Batra, D.; and Lee, S. 2020. Beyond the Nav-Graph: Vision-and-Language Navigation in Continuous Environments. In *Computer Vision – ECCV 2020*, volume 12373 of *Lecture Notes in Computer Science*, 104–120. Springer.

Krishnan, S.; Boroujerdian, B.; Fu, W.; Faust, A.; and Reddi, V. J. 2021. Air Learning: A Deep Reinforcement Learning Gym for Autonomous Aerial Robot Visual Navigation. *Machine Learning*, 110(9): 2501–2540.

Lei, X.; Wang, M.; Zhou, W.; and Li, H. 2025. GaussNav: Gaussian Splatting for Visual Navigation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(5): 4108–4121.

Lin, S.; Li, Z.; Zhao, X.; Zhou, G.; Wang, L.; Wei, R.; Tang, R.; Li, J.; Wang, H.; Pang, J.; van den Hengel, A.; Liu, J.; and Wu, Q. 2025. VLNVerse: A Benchmark for Vision-Language Navigation with Versatile, Embodied, Realistic Simulation and Evaluation. arXiv:2512.19021.

Liu, S.; Zhang, H.; Qi, Y.; Wang, P.; Zhang, Y.; and Wu, Q. 2023. AerialVLN: Vision-and-Language Navigation for UAVs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 15384–15394.

Lou, J.; Ding, R.; and Wu, W. 2024. HHPSO: A Heuristic Hybrid Particle Swarm Optimization Path Planner for Quadcopters. *Drones*, 8(6): 221.

Lou, J.; Shi, R.; Lin, Y.; Wang, Q.; and Wu, W. 2025. TALKER: A Task-Activated Language Model Based Knowledge-Extension Reasoning System. *IEEE Robotics and Automation Letters*, 10(2): 1026–1033.

Lou, J.; Wang, H.; Liu, X.; Zhang, Y.; Shi, R.; and Wu, W. 2026. AERIS: Aerial-Edge Role-Driven Intelligence at Runtime via Orchestrated Language-Model Swarm. arXiv:2606.30151.

Lou, J.; Wu, W.; Liao, S.; and Shi, R. 2023. Air-M: A Visual Reality Many-Agent Reinforcement Learning Platform for Large-Scale Aerial Unmanned System. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 5598–5605. IEEE.

Nie, D.; Guo, X.; Duan, Y.; Zhang, R.; and Chen, L. 2025. WMNav: Integrating Vision-Language Models into World Models for Object Goal Navigation. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2392–2399.

Pelluri, N. 2024. Transformers for Image-Goal Navigation. arXiv:2405.14128.

Qi, Z.; Zhang, Z.; Yu, Y.; Wang, J.; and Zhao, H. 2025. VLN-R1: Vision-Language Navigation via Reinforcement Fine-Tuning. arXiv:2506.17221.

Qin, Y.; Sun, A.; Hong, Y.; Wang, B.; and Zhang, R. 2025. NavigateDiff: Visual Predictors are Zero-Shot Navigation Assistants. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, 12002–12009. IEEE.

Saxena, P.; Raghuvanshi, N.; and Goveas, N. 2025. UAV-VLN: End-to-End Vision Language Guided Navigation for UAVs. In *2025 European Conference on Mobile Robots (ECMR)*, 1–6.

Shah, H.; Xing, J.; Messikommer, N.; Sun, B.; Pollefeys, M.; and Scaramuzza, D. 2025. ForesightNav: Learning Scene Imagination for Efficient Exploration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 5236–5245.

Wang, X.; Yang, D.; Wang, Z.; Kwan, H.; Chen, J.; Wu, W.; Li, H.; Liao, Y.; and Liu, S. 2025. Towards Realistic UAV Vision-Language Navigation: Platform, Benchmark, and Methodology. In *International Conference on Learning Representations (ICLR)*.

Yokoyama, N.; Ha, S.; Batra, D.; Wang, J.; and Bucher, B. 2024. VLFM: Vision-Language Frontier Maps for Zero-Shot Semantic Navigation. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 42–48.

Zhang, J.; Li, A.; Qi, Y.; Li, M.; Liu, J.; Wang, S.; Liu, H.; Zhou, G.; Wu, Y.; Li, X.; Fan, Y.; Li, W.; Chen, Z.; Gao, F.; Wu, Q.; Zhang, Z.; and Wang, H. 2026. Embodied Navigation Foundation Model. In *International Conference on Learning Representations (ICLR)*.

Zhang, M.; Qu, K.; Patil, V.; Cadena, C.; and Hutter, M. 2025a. Tag Map: A Text-Based Map for Spatial Reasoning and Navigation with Large Language Models. In *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, 2120–2146. PMLR.

Zhang, S.; Yu, X.; Song, X.; Wang, X.; and Jiang, S. 2024. Imagine Before Go: Self-Supervised Generative Map for Object Goal Navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 16414–16425.

Zhang, W.; Gao, C.; Yu, S.; Peng, R.; Zhao, B.; Zhang, Q.; Cui, J.; Chen, X.; and Li, Y. 2025b. CityNavAgent: Aerial Vision-and-Language Navigation with Hierarchical Semantic Planning and Global Memory. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 31292–31309. Association for Computational Linguistics.

Zheng, L.; Chiang, W.-L.; Sheng, Y.; Zhuang, S.; Wu, Z.; Zhuang, Y.; Lin, Z.; Li, Z.; Li, D.; Xing, E. P.; Zhang, H.; Gonzalez, J. E.; and Stoica, I. 2023. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems*, volume 36, 46595–46623. Curran Associates, Inc.

Zhou, G.; Hong, Y.; and Wu, Q. 2024. NavGPT: Explicit Reasoning in Vision-and-Language Navigation with Large Language Models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(7): 7641–7649.