AIRGROUNDVLN: A LARGE-SCALE BENCHMARK FOR GOAL-ORIENTED AIR-GROUND COLLABORATIVE VISION-AND-LANGUAGE NAVIGATION

Anonymous authors
Paper under double-blind review



Figure 1: **Overview of AirGroundVLN**. The benchmark contains 10,281 navigation episodes, 955 target instances, and 19 diverse scenes for goal-oriented air–ground collaborative VLN. It supports evaluation across aerial-visible/aerial-occluded target conditions and seen/unseen environments. AG-CoNAV is provided as a trainable reference framework for this setting.

ABSTRACT

Goal-oriented Vision-and-Language Navigation (VLN) requires agents to locate and reach targets described in natural language without prescribed routes. Air–ground collaboration is valuable for tasks requiring both wide-area search and fine-grained localization. However, systematic study of goal-oriented air–ground collaborative VLN remains limited by the lack of large-scale, diverse benchmarks and two core challenges: 1) substantial differences between aerial and ground views, together with useful observations becoming unavailable as navigation proceeds, make it difficult to maintain spatially consistent context across platforms and over time; and 2) asymmetric spatial observability makes ground perception locally detailed but spatially limited and aerial perception broad but locally coarse, limiting the reliability of single-platform planning. To address these limitations, we introduce **AirGroundVLN**, a benchmark containing ***10,281*** navigation episodes and ***955*** target instances across ***19*** Unreal Engine environments, with ***seen/unseen*** splits and an ***aerial-visibility*** protocol for systematic evaluation. Alongside the benchmark, we propose AG-CoNAV, a trainable reference framework comprising two key components: Spatiotemporally Anchored Collaborative Memory (SACM) and Aerial-Guided Regional-to-Local Planning (AGRLP). SACM maintains and retrieves spatially consistent historical context across aerial and ground observations. Meanwhile, AGRLP combines regional aerial guidance with fine-grained ground navigation. Extensive experiments demonstrate the effectiveness of AG-CoNAV and establish AirGroundVLN as a comprehensive benchmark for future exploration.

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.

1 INTRODUCTION

Goal-oriented vision-and-language navigation (VLN) is a core problem in embodied intelligence (Qi et al., 2020; Zhu et al., 2021), requiring an agent to interpret a language-specified target, search for it, identify the correct instance, and reach it without prescribed intermediate routes. Existing goal-oriented VLN studies predominantly consider two types of embodied platforms: (1) **Ground Platforms**, such as wheeled Unmanned Ground Vehicles (UGVs), quadruped robots, and humanoid robots (Krantz et al., 2020; Li et al., 2025). (2) **Aerial Platforms**, typically Unmanned Aerial Vehicles (UAVs) (Liu et al., 2023; Wang et al., 2025; Xiao et al., 2025; Gao et al., 2026). In large-scale outdoor environments, ground platforms provide fine-grained observations of local scene structure and target appearance and can physically approach ground-level targets. In contrast, aerial platforms provide broader spatial coverage and global scene context, enabling efficient wide-area search and route-level reasoning. As two popular embodied platforms, substantial research has focused on separately expanding their perception (Huang et al., 2023), reasoning and planning (Ahn et al., 2022; Song et al., 2023), and execution capabilities (Zhou et al., 2020; Kumar et al., 2021).

But consider such a real mission in a large industrial complex, where robots are instructed to ***rescue a worker in a red jacket near a damaged warehouse***. In this mission, an aerial platform can rapidly survey the area and search for the worker from a global viewpoint, but its observations may be obstructed by trees, roofs, or collapsed structures, and it cannot provide physical rescue. Meanwhile, a ground platform can closely approach the worker, inspect regions hidden from the aerial viewpoint, and provide assistance, but damaged roads, debris, and collapsed buildings may render routes infeasible. Such a mission calls for aerial and ground platforms to work together, combining wide-area search with fine-grained target localization and physical access. This complementary division of capabilities motivates heterogeneous air–ground collaboration for goal-oriented navigation.

However, systematic study of goal-oriented air–ground collaborative VLN remains limited by three key issues: (1) Although recent efforts have explored air–ground VLN (Li et al., 2026; Zhang et al., 2026b), they remain limited in scale and diversity, with AirGroundBench and AGC-VLN evaluated on only 115 and 100 episodes, respectively. To our knowledge, a large-scale benchmark for goal-oriented air–ground collaborative VLN is still lacking, limiting systematic evaluation. (2) Beyond benchmark limitation, aerial and ground observations differ in viewpoint, while useful observations from either platform may become unavailable during navigation. Therefore, effective collaboration requires spatially consistent context across views and time. (3) Aerial and ground platforms exhibit asymmetric spatial observability: ground perception is locally detailed but spatially limited, whereas aerial perception covers broader regions but provides less fine-grained obstacle information, making either platform alone insufficient for reliable navigation.

To address this gap, we first introduce **AirGroundVLN**, a large-scale benchmark for goal-oriented air–ground collaborative VLN. AirGroundVLN contains **10,281** navigation episodes, **955** target instances, and **19** Unreal Engine environments, covering urban areas, towns, industrial areas, and other complex outdoor environments. Each episode includes aerial and ground observations together with a goal-oriented instruction describing the target appearance and its surrounding environmental context, rather than prescribing intermediate navigation actions. We provide seen and unseen environment splits to evaluate cross-environment generalization. An ***aerial-visibility protocol*** is further defined to distinguish targets visible from above and those occluded by overhead structures, enabling evaluation under different target-visibility conditions.

Alongside the benchmark, we develop **AG-CoNAV**, a trainable reference framework addressing the above two key challenges of air–ground collaborative VLN. First, to maintain spatially consistent spatiotemporal context across heterogeneous aerial and ground observations, we introduce the Spatiotemporally Anchored Collaborative Memory (SACM), which stores and retrieves historical evidence by associating observations of physically related regions over time. Second, to overcome limitations of single-platform planning under asymmetric spatial observability, we introduce the Aerial-Guided Regional-to-Local Planning module (AGRLP), which combines regional aerial guidance with locally executable ground planning. In addition, to establish reference baselines for AirGroundVLN, we construct training-free and trainable methods for air–ground collaborative VLN and evaluate them alongside AG-CoNAV across seen/unseen environments and aerial-visible/aerial-occluded conditions. These evaluations establish reference performance for AirGroundVLN and demonstrate the effectiveness of AG-CoNAV. Our main contributions are summarized as follows:

- We introduce **AirGroundVLN**, a large-scale benchmark for goal-oriented air–ground collaborative Vision-and-Language Navigation. It contains **10,281** navigation episodes and **955** target instances across **19** Unreal Engine environments, together with goal-oriented instructions, seen/unseen splits, and an *aerial-visibility* protocol for systematic evaluation.
- We develop **AG-CoNAV**, a trainable reference framework for heterogeneous air–ground collaborative VLN. It incorporates a Spatiotemporally Anchored Collaborative Memory (SACM) for spatiotemporally consistent context association and historical retrieval, together with an Aerial-Guided Regional-to-Local Planning module (AGRLP) that coordinates regional aerial guidance with fine-grained ground execution.
- We provide a diverse suite of training-free and trainable baselines for AirGroundVLN, establishing a comprehensive evaluation suite for this new setting. Systematic evaluations across environment generalization and aerial-visibility conditions provide reference performance for the benchmark and validate the effectiveness of AG-CoNAV.

2 Related Work

2.1 Vision-and-Language Navigation Benchmarks

Early VLN benchmarks such as R2R and R4R focus on route following in indoor environments (Anderson et al., 2018b; Jain et al., 2019), while Touchdown and Talk2Nav extend language-guided navigation to outdoor scenes (Chen et al., 2019; Vasudevan et al., 2021). VLN-CE moves navigation from discrete panoramic graphs to continuous environments (Krantz et al., 2020), and AerialVLN further extends route-oriented VLN to UAV navigation in large-scale outdoor environments (Liu et al., 2023). Beyond route following, REVERIE and SOON study goal-oriented navigation toward language-described targets (Qi et al., 2020; Zhu et al., 2021), while TravelUAV, UAV-ON, and CityNav extend goal-oriented language guidance to aerial platforms (Wang et al., 2025; Xiao et al., 2025; Lee et al., 2025). Despite these advances, most benchmarks still focus on a single embodied platform. AirGroundBench begins to study heterogeneous UAV–UGV perception through 62K diagnostic VQA instances and 115 closed-loop VLN episodes (Li et al., 2026), but large-scale goal-oriented air–ground collaborative VLN remains underexplored.

2.2 Multi-Agent Collaborative Navigation

Multi-agent navigation studies focus on homogeneous teams of ground robots (Burgard et al., 2000; Gao et al., 2022; Tan et al., 2022) or aerial agents (Westheider et al., 2023). Vision-language research extends collaboration to language-conditioned navigation, with CoNavBench studying VLN among ground robots (Wang et al., 2026), and AeroDuo and CoNav-UAV exploring complementary viewpoints within multi-UAV teams (Wu et al., 2025; Song et al., 2026). Moving toward heterogeneous collaboration, AGOS-Bench studies closed-loop UAV–UGV cooperative object search with visual-reference goals (Yu et al., 2026), while AGC-VLN introduces language-conditioned air–ground navigation through a shared bird's-eye map linking aerial context with ground planning (Zhang et al., 2026b). However, dedicated trainable air–ground collaborative VLN remains underexplored.

3 AirGroundVLN Benchmark

In this section, we first define the goal-oriented air–ground collaborative VLN task and its evaluation protocol (Sec. 3.1), then present the construction pipeline of AirGroundVLN (Sec. 3.2), and finally analyze its scale, diversity, and evaluation splits (Sec. 3.3).

3.1 Task Definition

Task Formulation. AirGroundVLN considers a closed-loop navigation episode in which an Unmanned Ground Vehicle (UGV) and an Unmanned Aerial Vehicle (UAV) cooperate to locate and reach a language-specified target instance in a large-scale outdoor environment. The UGV serves as the primary target-reaching platform, while the UAV provides complementary aerial perception and global spatial context. At the beginning of each episode, the two platforms are initialized at poses P_0^g and P_0^a, respectively, and receive a goal-oriented instruction I. The instruction describes

the target through its semantic category, visual attributes, surrounding environmental context, and coarse relative direction, without prescribing intermediate navigation actions. At navigation step t, the UGV and UAV receive their current visual observations o_t^g and o_t^a, together with historical context available at the current step. A collaborative navigation system π produces platform-specific navigation decisions (*e.g.*, waypoints or actions) and a termination signal:

$$(u_t^g, u_t^a, q_t) = \pi\left(I, o_t^g, o_t^a, H_t\right), \qquad q_t \in \{0, 1\}, \tag{1}$$

where H_t denotes the historical information maintained by the navigation system at step t. For AirGroundVLN, the variables u_t^g and u_t^a denote the predicted waypoint decisions for the UGV and UAV, respectively, and $q_t = 1$ indicates termination of the episode. An episode terminates at $T_{\text{end}} = \min\{t \mid q_t = 1 \text{ or } t = T_{\max}\}$, where $T_{\max}$ is the maximum allowable navigation budget.

Evaluation Protocol. Given these asymmetric roles, we evaluate performance on the UGV using four standard VLN metrics (Anderson et al., 2018a): Success Rate (SR), Success weighted by Path Length (SPL), UGV Navigation Error (NE_{UGV}), and Oracle Success Rate (OSR). For the UAV, we report UAV Navigation Error (NE_{UAV}), defined as its final horizontal distance to the target, as a *diagnostic* metric. Evaluation protocol and metric definitions are provided in Appendix A.

3.2 Benchmark Construction



Figure 2: **Overview of AirGroundVLN data collection and an illustrative episode**. The collection process includes target and environment description, start-location selection, goal-oriented instruction generation, and dense air–ground waypoint generation.

AirGroundVLN comprises 19 outdoor scenes in Unreal Engine, covering urban areas, towns, industrial areas, and other complex outdoor environments. The simulation environment is built in Unreal Engine using AirSim (Shah et al., 2017) interfaces for vehicle control and sensing, with CARLA-Air interface compatibility (Zeng et al., 2026). The benchmark construction pipeline consists of four components: **(1) Goal-Oriented Instruction Generation.** As illustrated in Fig. 2, for each target selected by annotators, we collect multi-view observations to describe its appearance and surrounding environment, and derive a coarse relative direction for each start location. **(2) Air–Ground Trajectory Generation.** Following prior aerial VLN benchmark construction (Gao et al., 2026), we generate platform-specific reference trajectories using A*-based planning (Hart et al., 1968). For the UGV, annotators specify sparse key waypoints that respect road topology and traversability, between which a modified surface-based A* planner generates the ground trajectory. UAV reference trajectories are generated directly using A* from the start pose to the end pose. **(3) Visibility Annotation.** In goal-oriented air–ground collaborative navigation, target instances may be occluded from

the aerial viewpoint by overhead structures. To systematically evaluate collaborative navigation under different aerial target-visibility conditions, we categorize target instances into ***aerial-visible*** and ***aerial-occluded*** subsets. **(4) Human Quality Assurance.** Human verification is incorporated throughout construction. Notably, all candidate episodes are checked for consistency across target instances, instructions, air–ground trajectories, and aerial-visibility labels. Recoverable inconsistencies are manually corrected, while ambiguous or invalid samples are discarded, followed by a final comprehensive review to obtain the resulting benchmark. Detailed construction procedures and quality-control statistics are provided in Appendix B.

3.3 Dataset Analysis

Table 1: Comparison with representative VLN benchmarks. N_{traj} denotes the number of navigation episodes (paired air–ground trajectories for AirGroundVLN), while L_{traj} and L_{instr} denote average trajectory length (m) and instruction length (words), respectively. "–" indicates statistics not officially reported by the corresponding benchmark.

Benchmark	Ground	Air	Simulator	$\mathbf{N}_{\text{traj}}$	$\mathbf{L}_{\text{traj}}^{\text{ground}}$	$\mathbf{L}_{\text{traj}}^{\text{aerial}}$	$\mathbf{L}_{\text{instr}}$	$\mathbf{N}_{\text{vocab}}$
R2R (Anderson et al., 2018b)	√	×	Matterport3D	7,189	10.0	×	29	3.1K
REVERIE (Qi et al., 2020)	√	×	Matterport3D	7,234	10.0	×	18	1.6K
RxR (Ku et al., 2020)	√	×	Matterport3D	13,992	14.9	×	129	7.0K
VLN-CE (Krantz et al., 2020)	√	×	Habitat	4,475	11.1	×	–	4.3K
AerialVLN (Liu et al., 2023)	×	√	AirSim	8,446	×	661.8	83	4.5K
TravelUAV (Wang et al., 2025)	×	√	OpenUAV	12,149	×	255.0	104	10.8K
CityNav (Lee et al., 2025)	×	√	CityFlight	32,637	×	545.0	26	6.4K
UAV-ON (Xiao et al., 2025)	×	√	AirSim	11,000	×	–	–	–
HaL-13k (Wu et al., 2025)	×	√	OpenUAV	13,838	×	208.4	–	–
AirGroundBench (Li et al., 2026)	√	√	AirSim	115	–	–	–	–
AGC-VLN (Zhang et al., 2026b)	√	√	CARLA-Air	100	–	–	–	–
AirGroundVLN (Ours)	√	√	AirSim/CARLA-Air	**10,281**	**90.9**	**79.0**	**82**	**2.5K**

As summarized in Table 1, AirGroundVLN extends existing VLN benchmarks to large-scale goal-oriented navigation with air–ground agents. Most prior datasets focus on single-platform ground or aerial navigation, while existing air–ground benchmarks lack scale and comprehensive evaluation. By contrast, AirGroundVLN provides 10,281 air–ground collaborative navigation episodes across 19 simulation scenes, covering 955 target instances. This benchmark enables systematic evaluation of how heterogeneous aerial and ground observations can be jointly exploited for goal-oriented navigation. We further analyze the key characteristics of the benchmark in Fig. 3.

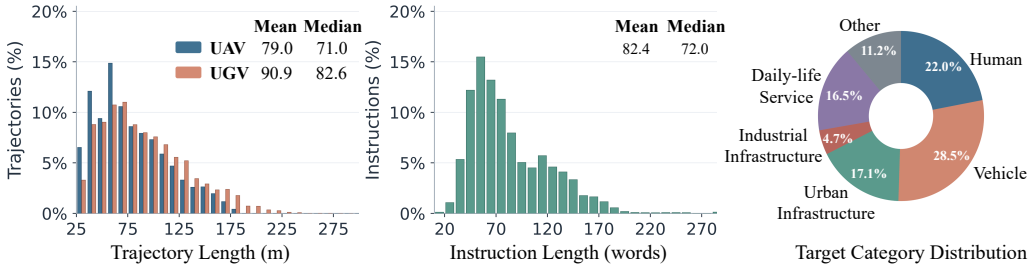


Figure 3: **Statistics of AirGroundVLN**. (a) Distribution of UAV and UGV reference trajectory lengths. (b) Distribution of navigation instruction lengths. (c) Distribution of target instances across six major semantic categories.

Trajectory, Language, and Target Diversity. As shown in Fig. 3 (a), the average length of UAV reference trajectories in AirGroundVLN is 79.0 m , whereas the corresponding UGV trajectories average 90.9 m. Ground trajectories are longer on average, reflecting the different motion constraints of the two platforms: UAV trajectories are less constrained by ground topology, whereas UGVs must follow traversable surfaces and road structures. As shown in Fig. 3 (b), navigation instructions in AirGroundVLN average 82 words. Furthermore, as depicted in Fig. 3 (c), the 955 target instances span six broad semantic categories, demonstrating substantial semantic diversity across vehicles, humans, urban infrastructure, daily-life services, industrial infrastructure, and other objects.

Data Splits and Visibility. AirGroundVLN is divided into *train*, *val-seen*, *test-seen*, and *test-unseen* splits with 7,538, 933, 962, and 848 episodes, respectively, where *test-unseen* evaluates generalization to unseen environments. To evaluate air–ground collaborative VLN under different aerial target-visibility conditions, we categorize test episodes as *aerial-visible* or *aerial-occluded* based on visibility from the aerial viewpoint. Further benchmark statistics are provided in Appendix C (*e.g.*, word clouds of target names and visual attributes, and aerial-visibility distributions).

4 METHOD

We propose **AG-CoNAV**, a trainable framework for goal-oriented air–ground collaborative VLN designed to tackle cross-view spatiotemporal context inconsistency and planning under asymmetric spatial observability. As illustrated in Fig. 4, given a goal-oriented instruction, aerial–ground observations, and a UAV-maintained fused BEV map constructed by projecting and fusing sequential UAV RGB-D observations, the Spatiotemporally Anchored Collaborative Memory (SACM) (Sec. 4.1) maintains and retrieves spatially consistent context across views and time. This retrieved context is then incorporated into multimodal waypoint queries to predict UAV and UGV waypoint distributions. Finally, the Aerial-Guided Regional-to-Local Planning module (AGRLP) (Sec. 4.2) uses these predictions to coordinate regional aerial guidance with fine-grained ground navigation.

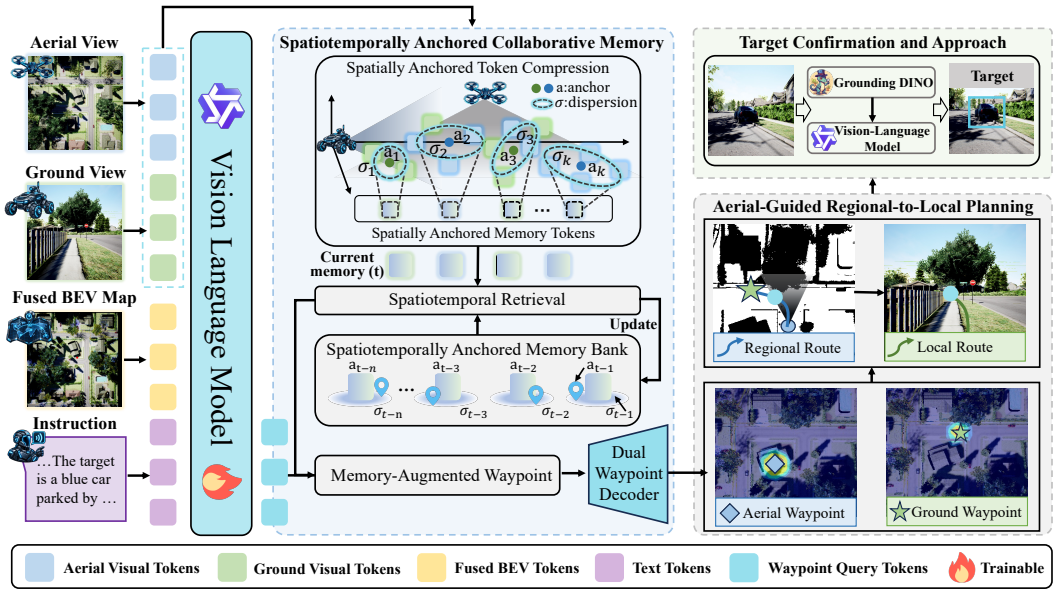


Figure 4: **Overview of the AG-CoNAV framework**. Given goal-oriented instructions, aerial–ground observations, and a UAV-maintained fused BEV map, SACM maintains and retrieves spatially consistent context across views and time, while AGRLP coordinates regional aerial guidance with fine-grained ground execution.

4.1 SPATIOTEMPORALLY ANCHORED COLLABORATIVE MEMORY

Aerial and ground observations offer complementary information but differ in viewpoint, scale, and spatial coverage. Moreover, evidence observed at one navigation step may become unavailable in later observations due to occlusion and changing viewpoints. Effective collaboration therefore requires preserving complementary evidence across views and time while maintaining spatial consistency. To this end, we introduce SACM, which compresses observations into spatially characterized memory tokens and retrieves historical evidence via dispersion-aware spatiotemporal matching.

Spatially Anchored Memory Construction. To preserve historical evidence efficiently, we compress visual features into a compact set of memory tokens. Conventional feature-only compression discards the physical origin of the aggregated evidence. Associating each compressed token with a single coordinate partially restores this information, but cannot distinguish tokens concentrated

around a local region from those aggregating evidence over a much broader area. *This distinction is particularly important for air–ground collaboration, where aerial and ground observations naturally differ in spatial coverage.* We therefore characterize each compressed token with both a spatial anchor and a spatial dispersion to regulate historical retrieval. For each platform $v \in \{g, a\}$, let $X_t^v \in \mathbb{R}^{N_v \times d}$ denote its visual tokens at step t. Using camera parameters and depth, each visual token is projected onto the shared ground plane with metric coordinate $\mathbf{c}_{t,i}^v \in \mathbb{R}^2$. View-specific learned queries compress X_t^v into K memory tokens $M_t^v \in \mathbb{R}^{K \times d}$. Let $\alpha_{t,k,i}^v$ denote the normalized cross-attention weight ($\sum_i \alpha_{t,k,i}^v = 1$). Each compressed token is characterized by a spatial anchor $\mathbf{a}_{t,k}^v$ and dispersion $\sigma_{t,k}^v$:

$$\mathbf{a}_{t,k}^v = \sum_i \alpha_{t,k,i}^v \mathbf{c}_{t,i}^v, \qquad \sigma_{t,k}^v = \sqrt{\sum_i \alpha_{t,k,i}^v \left\| \mathbf{c}_{t,i}^v - \mathbf{a}_{t,k}^v \right\|_2^2}. \tag{2}$$

Thus, $\mathbf{a}_{t,k}^v$ captures where the compressed evidence is centered, whereas $\sigma_{t,k}^v$ captures how broadly that evidence is distributed in physical space. The resulting tokens and their metadata are stored in a shared air–ground memory bank.

Spatiotemporal Memory Retrieval. Historical evidence is retrieved through cross-attention augmented with spatial and temporal priors, allowing current tokens to aggregate complementary evidence across aerial and ground views. Let $\mathcal{B}_{t-1} = \{(\mathbf{m}_j, \mathbf{a}_j, \sigma_j, \tau_j)\}_j$ denote the historical memory bank accumulated from both platforms, where $\mathbf{m}_j$, $\mathbf{a}_j$, σ_j, and τ_j are the feature, spatial anchor, dispersion, and timestamp of the j-th memory token, respectively. For each current–historical token pair, the semantic attention score incorporates spatial and temporal priors. Let $\Delta t_j = t - \tau_j$ denote the relative age of historical token j. The retrieval score is

$$S_{ij} = \frac{\mathbf{q}_i^\top \mathbf{k}_j}{\sqrt{d}} - \lambda_{\text{spa}} \frac{\|\mathbf{a}_i - \mathbf{a}_j\|_2^2}{\rho^2 + \sigma_i^2 + \sigma_j^2} - \lambda_{\text{tmp}} \log(1 + \Delta t_j), \tag{3}$$

where d is the feature dimension, ρ defines a base spatial tolerance, and λ_{spa} and λ_{tmp} are learnable weights for the spatial and temporal priors, respectively. The dispersion-aware spatial term favors closer anchor alignment for spatially concentrated evidence, while allowing greater positional variation for tokens spanning broader regions. Thus, retrieval accounts for both where the evidence is located and how much physical space it represents. The retrieved historical evidence is aggregated into a spatiotemporally consistent context F_t^{mem}, while the current memory tokens and their spatial metadata are appended to update the memory bank. Finally, waypoint queries from the multimodal backbone attend to F_t^{mem}, yielding the memory-augmented waypoint queries Q_t^{wp} used by AGRLP.

4.2 Aerial-Guided Regional-to-Local Planning

While ground observations ensure locally safe navigation, their limited spatial scope can lead to short-sighted planning. To overcome this asymmetric perception, AGRLP couples regional aerial guidance with hierarchical ground planning, leveraging broad spatial context while preserving locally executable navigation.

Dual Waypoint Prediction. Given the memory-augmented waypoint queries Q_t^{wp}, following heatmap-based waypoint prediction (Hong et al., 2022), we model aerial and ground waypoints as spatial distributions. The waypoints are obtained as $(\hat{\mathbf{p}}_t^g, \hat{\mathbf{p}}_t^a) = \Phi(Q_t^{\text{wp}})$, where Φ denotes the waypoint prediction and decoding function. We jointly supervise the predicted distributions and their spatial centers, with the training loss detailed in Appendix D.1.

Regional-to-Local Planning and Target Approach. The UAV maintains a regional occupancy grid Occ_t^a from depth observations while exploring toward $\hat{\mathbf{p}}_t^a$. Given the current UGV position $\mathbf{x}_t^g$ and predicted ground waypoint $\hat{\mathbf{p}}_t^g$, the regional route is computed as $P_t^{\text{regional}} = \text{A}^*(\mathbf{x}_t^g, \hat{\mathbf{p}}_t^g; \text{Occ}_t^a)$, providing long-range navigation guidance. Meanwhile, the UGV maintains a local occupancy grid Occ_t^g from its RGB-D observations, selects a reachable sub-goal along P_t^{regional} within the local traversable space, and computes a collision-free local trajectory toward it using A* search. Following (Wang et al., 2025), we detect target candidates with Grounding DINO (Liu et al., 2024) and verify their appearance and surrounding context against the instruction. Upon confirmation, the UGV approaches the candidate's position estimated from RGB-D observations (Zhang et al., 2026a) and terminates the episode (Appendix D.2).

5 EXPERIMENTS

5.1 EXPERIMENTAL SETUP

AG-CoNAV uses Qwen3.5-2B (Team, 2026) as the vision-language backbone. We freeze the pre-trained backbone parameters and train LoRA adapters (Hu et al., 2021), while jointly optimizing the memory and waypoint prediction modules. Implementation details are provided in Appendix E. All experiments follow the evaluation protocol in Sec. 3.1. The success distance threshold is set to 5 m.

5.2 BASELINES

Random Policy. We include a random navigation policy as a lower-bound reference. At each step, the UGV randomly samples and follows a waypoint within its local collision-free space.

Training-Free Baselines. We evaluate several representative Vision-Language Models (VLMs), including Qwen3.5-2B/4B (Team, 2026), Gemini-3.6-Flash (Team et al., 2023), and GPT-5.5 (OpenAI, 2026), under a unified zero-shot air–ground navigation pipeline. The evaluated models span high-capacity proprietary VLMs and compact open-source VLMs. We also include AGC-VLN (Zhang et al., 2026b), a training-free air–ground collaborative VLN method. Details of the unified training-free navigation pipeline are provided in Appendix E.

Training-Based Baselines. To examine the effects of global spatial context and heterogeneous air–ground collaboration, we construct three training-based baselines. **(1) UGV Only**: This baseline predicts ground waypoints solely from ground observations without aerial assistance. **(2) UGV + Global BEV Map**: This baseline incorporates an aerially derived global BEV map into ground waypoint prediction to compare static global context with active collaboration. **(3) AeroDuo**: To our knowledge, no existing trainable framework is designed for heterogeneous air–ground collaborative VLN. We therefore adapt AeroDuo (Wu et al., 2025) to the UAV–UGV setting of AirGroundVLN.

5.3 MAIN RESULTS

Table 2: Comparison with baselines on AirGroundVLN. Results are reported across seen and unseen environments and under different aerial visibility conditions.

Method	Seen					Unseen				
	SR↑	OSR↑	SPL↑	NE_{UGV}↓	NE_{UAV}↓	SR↑	OSR↑	SPL↑	NE_{UGV}↓	NE_{UAV}↓
Random	0.52	0.52	0.34	79.66	–	0.47	0.59	0.23	72.43	–
Training-Free Methods										
Air-Ground (Qwen3.5-2B)	4.05	4.26	2.38	81.08	100.85	1.30	1.77	0.89	69.60	92.12
Air-Ground (Qwen3.5-4B)	13.10	15.59	9.54	63.33	68.47	10.14	11.56	6.75	54.96	62.58
AGC-VLN (Qwen3.5-4B)	13.41	16.63	8.42	65.75	63.99	10.50	12.38	4.95	54.37	51.16
Air-Ground (Gemini-3.6-Flash)	22.25	22.77	20.51	47.80	50.59	17.45	18.99	15.79	46.41	51.82
Air-Ground (GPT-5.5)	24.43	25.99	21.59	48.77	54.78	19.69	22.05	17.25	**44.21**	48.07
Training-Based Methods										
UGV Only	16.84	20.89	10.86	62.92	–	13.33	17.69	8.38	57.43	–
UGV + Global BEV Map	20.06	24.53	18.81	53.13	–	16.86	20.28	15.66	45.21	–
AeroDuo	17.46	22.45	14.00	54.71	56.27	15.45	19.69	13.79	46.17	55.88
AG-CoNAV (Qwen3.5-2B)	**32.64**	**35.65**	**29.85**	**39.72**	**44.81**	**20.05**	**24.06**	**18.46**	44.43	**44.04**
Method	**Visible**					**Occluded**				
	SR↑	OSR↑	SPL↑	NE_{UGV}↓	NE_{UAV}↓	SR↑	OSR↑	SPL↑	NE_{UGV}↓	NE_{UAV}↓
Random	0.64	0.71	0.37	78.41	–	0.00	0.00	0.00	68.96	–
Training-Free Methods										
Air-Ground (Qwen3.5-2B)	3.14	3.43	1.93	77.83	96.77	1.46	1.95	0.83	68.41	96.71
Air-Ground (Qwen3.5-4B)	12.57	14.57	9.00	61.29	67.79	8.78	10.73	5.60	52.96	58.60
AGC-VLN (Qwen3.5-4B)	12.86	14.86	6.88	61.97	61.88	9.27	13.90	6.50	55.10	44.64
Air-Ground (Gemini-3.6-Flash)	21.43	22.36	19.76	46.67	51.42	15.12	16.34	13.32	48.78	50.33
Air-Ground (GPT-5.5)	23.29	24.86	20.58	47.38	53.12	18.54	21.71	16.05	44.09	46.60
Training-Based Methods										
UGV Only	16.36	20.64	10.07	61.35	–	11.22	15.12	8.42	56.90	–
UGV + Global BEV Map	19.64	24.36	18.89	49.73	–	14.88	16.34	12.03	48.35	–
AeroDuo	17.36	22.71	14.84	51.29	56.64	13.66	15.85	10.68	48.75	54.21
AG-CoNAV (Qwen3.5-2B)	**28.86**	**31.86**	**26.54**	**42.12**	**45.21**	**19.51**	**24.63**	**17.59**	**41.25**	**41.84**

Performance Analysis. Table 2 compares training-free and training-based baselines on AirGround-VLN. AG-CoNAV achieves the best SR across all splits, demonstrating its effectiveness for air–ground collaborative VLN. Compared with the training-free Air-Ground model using the same

Qwen3.5-2B backbone, AG-CoNAV substantially improves SR from 4.05% to 32.64% and SPL from 2.38% to 29.85% on the seen split, highlighting the advantage of task-specific training. Compared with *UGV + Global BEV Map*, AG-CoNAV consistently achieves higher success rates under both visible and occluded conditions, demonstrating the benefit of actively acquired aerial context over static global information. Further ***qualitative trajectory comparisons*** are provided in Appendix F. An animated ***simulation demo of AG-CoNAV*** showcasing the closed-loop air–ground navigation trajectories is provided in the supplementary material.

5.4 Ablation Studies

In this section, we ablate SACM and AGRLP to evaluate the effectiveness of each component in AG-CoNAV.

Influence of Main Modules. We first evaluate the contribution of each module in AG-CoNAV. The variant without SACM and AGRLP serves as the baseline. As shown in Table 3, introducing AGRLP improves the baseline SR from 25.68% to 26.30% and SPL from 22.63% to 24.03% on the seen split, demonstrating the benefit of aerial guidance for ground navigation. Adding SACM alone further increases the SR to 31.39% and SPL to 27.79%, indicating the importance of spatially consistent historical context retrieval. Combining SACM and AGRLP achieves the best performance, with SR and SPL reaching 32.64% and 29.85%, respectively, showing that the two components provide complementary benefits for air–ground collaborative navigation.

Influence of SACM Priors. We further investigate the contribution of the temporal prior and the dispersion-aware spatial prior in SACM. As shown in Table 4, removing either prior degrades performance compared with the complete SACM. Specifically, using only the temporal prior achieves 29.83% SR on the seen split, while using only the spatial prior achieves 31.19% SR. Combining both priors further increases the SR to 32.64% and consistently improves other navigation metrics. These results demonstrate that temporal continuity and dispersion-aware spatial context provide complementary information for retrieving relevant historical observations across views and time.

Table 3: Ablation of SACM and AGRLP components in AG-CoNAV on seen/unseen environments.

SACM	AGRLP	Seen					Unseen				
		SR↑	OSR↑	SPL↑	NE_{UGV}↓	NE_{UAV}↓	SR↑	OSR↑	SPL↑	NE_{UGV}↓	NE_{UAV}↓
✗	✗	25.68	27.65	22.63	48.59	55.21	17.10	19.93	15.61	50.02	49.30
✗	✓	26.30	29.42	24.03	46.71	54.49	17.92	20.64	16.18	48.59	48.76
✓	✗	31.39	33.47	27.79	41.93	45.62	19.22	22.29	17.56	45.76	45.13
✓	✓	**32.64**	**35.65**	**29.85**	**39.72**	**44.81**	**20.05**	**24.06**	**18.46**	**44.43**	**44.04**

Table 4: Detailed ablation of the temporal and dispersion-aware spatial priors in SACM on seen and unseen environments. AGRLP is enabled for all variants.

Temporal Prior	Spatial Prior	Seen					Unseen				
		SR↑	OSR↑	SPL↑	NE_{UGV}↓	NE_{UAV}↓	SR↑	OSR↑	SPL↑	NE_{UGV}↓	NE_{UAV}↓
✓	✗	29.83	31.91	26.75	43.92	47.03	18.99	21.82	16.92	47.14	46.19
✗	✓	31.19	34.10	28.56	41.37	45.92	19.69	23.11	17.53	45.61	44.86
✓	✓	**32.64**	**35.65**	**29.85**	**39.72**	**44.81**	**20.05**	**24.06**	**18.46**	**44.43**	**44.04**

6 Conclusion

We introduce **AirGroundVLN**, a large-scale benchmark for goal-oriented air–ground collaborative Vision-and-Language Navigation, containing 10,281 episodes and 955 target instances across 19 Unreal Engine environments, equipped with an aerial-visibility evaluation protocol. Alongside the benchmark, we propose **AG-CoNAV**, a framework addressing heterogeneous collaboration via Spatiotemporally Anchored Collaborative Memory (SACM) and Aerial-Guided Regional-to-Local Planning (AGRLP). Extensive evaluations validate AG-CoNAV and establish strong baselines. We hope AirGroundVLN can serve as a useful testbed for the community and facilitate future research on heterogeneous air–ground collaborative VLN toward more capable embodied navigation systems.

AI USE STATEMENT

In this work, we used generative AI tools to assist with instruction generation during dataset construction. All generated instructions were manually reviewed and verified by the authors (Sec. 3.2). We also used generative AI tools to check grammatical errors and improve the clarity and readability of the manuscript. We take responsibility for the final content of this work.

REFERENCES

Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, et al. Do as i can, not as i say: Grounding language in robotic affordances. *arXiv preprint arXiv:2204.01691*, 2022.

Peter Anderson, Angel Chang, Devendra Singh Chaplot, Alexey Dosovitskiy, Saurabh Gupta, Vladlen Koltun, Jana Kosecka, Jitendra Malik, Roozbeh Mottaghi, Manolis Savva, et al. On evaluation of embodied navigation agents. *arXiv preprint arXiv:1807.06757*, 2018a.

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, 2018b.

Wolfram Burgard, Mark Moors, Dieter Fox, Reid Simmons, and Sebastian Thrun. Collaborative multi-robot exploration. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, volume 1, pp. 476–481. IEEE, 2000.

Howard Chen, Alane Suhr, Dipendra Misra, Noah Snavely, and Yoav Artzi. Touchdown: Natural language navigation and spatial reasoning in visual street environments. In *2019 IEEE/CVF conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12530–12539. IEEE, 2019.

Yuman Gao, Yingjian Wang, Xingguang Zhong, Tiankai Yang, Mingyang Wang, Zhixiong Xu, Yongchao Wang, Yi Lin, Chao Xu, and Fei Gao. Meeting-merging-mission: A multi-robot coordinate framework for large-scale communication-limited exploration. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 13700–13707. IEEE, 2022.

Yunpeng Gao, Chenhui Li, Zhongrui You, Junli Liu, Li Zhen, Pengan Chen, Qizhi Chen, Zhonghan Tang, Liansheng Wang, Yiwen Tang, et al. Openfly: A comprehensive platform for aerial vision-language navigation. In *International Conference on Learning Representations*, volume 2026, pp. 101858–101871, 2026.

Peter E Hart, Nils J Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968.

Yicong Hong, Zun Wang, Qi Wu, and Stephen Gould. Bridging the gap between learning in discrete and continuous environments for vision-and-language navigation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15418–15428. IEEE, 2022.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Chenguang Huang, Oier Mees, Andy Zeng, and Wolfram Burgard. Visual language maps for robot navigation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 10608–10615. IEEE, 2023.

Vihan Jain, Gabriel Magalhaes, Alexander Ku, Ashish Vaswani, Eugene Ie, and Jason Baldridge. Stay on the path: Instruction fidelity in vision-and-language navigation. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 1862–1872, 2019.

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.

Ashish Kumar, Zipeng Fu, Deepak Pathak, and Jitendra Malik. RMA: Rapid Motor Adaptation for Legged Robots. In *Proceedings of Robotics: Science and Systems*, 2021. doi: 10.15607/RSS.2021.XVII.011.

Jungdae Lee, Taiki Miyanishi, Shuhei Kurita, Koya Sakamoto, Daichi Azuma, Yutaka Matsuo, and Nakamasa Inoue. Citynav: A large-scale dataset for real-world aerial navigation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5912–5922. IEEE, 2025.

Haotian Li, Yida Wang, Leyuan Wang, Jinshan Lai, Keyang Wang, Zonghao Guo, Qiang Ma, Liuyu Xiang, Jianwei Hu, and Zhaofeng He. Airgroundbench: Probing spatial intelligence in multimodal large models under heterogeneous multi-view embodied collaboration. *arXiv preprint arXiv:2606.28049*, 2026.

Zerui Li, Gengze Zhou, Haodong Hong, Yanyan Shao, Wenqi Lyu, Yanyuan Qiao, and Qi Wu. Ground-level viewpoint vision-and-language navigation in continuous environments. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5266–5273. IEEE, 2025.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European Conference on Computer Vision*, pp. 38–55. Springer, 2024.

Shubo Liu, Hongsheng Zhang, Yuankai Qi, Peng Wang, Yanning Zhang, and Qi Wu. Aerialvln: Vision-and-language navigation for uavs. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15338–15348. IEEE, 2023.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

OpenAI. GPT-5.5: A new class of intelligence for real work, April 2026. URL `https://openai.com/index/introducing-gpt-5-5/`.

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

Shital Shah, Debadeepta Dey, Chris Lovett, and Ashish Kapoor. Airsim: High-fidelity visual and physical simulation for autonomous vehicles. In *Field and service robotics: Results of the 11th international conference*, pp. 621–635. Springer, 2017.

Chan Hee Song, Jiaman Wu, Clayton Washington, Brian M Sadler, Wei-Lun Chao, and Yu Su. Llm-planner: Few-shot grounded planning for embodied agents with large language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2998–3009, 2023.

Junru Song, Wenhao Zhang, Yang Yang, Xuekai Qiu, Feifei Wang, Weien Zhou, Tingsong Jiang, Ying Wen, Yang Li, and Wen Yao. Conav-uav: Cooperative dual-altitude aerial navigation via stackelberg learning. *arXiv preprint arXiv:2608.01802*, 2026.

Aaron Hao Tan, Federico Pizarro Bejarano, Yuhan Zhu, Richard Ren, and Goldie Nejat. Deep reinforcement learning for decentralized multi-robot exploration with macro actions. *IEEE Robotics and Automation Letters*, 8(1):272–279, 2022.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

Arun Balajee Vasudevan, Dengxin Dai, and Luc Van Gool. Talk2nav: Long-range vision-and-language navigation with dual attention and spatial memory. *International Journal of Computer Vision*, 129(1):246–266, 2021.

Tianhang Wang, Xinhai Li, Fan Lu, Tianshi Gong, Jiankun Dong, Weiyi Xue, Sanqing Qu, Chenjia Bai, and Guang Chen. Conavbench: Collaborative long-horizon vision-language navigation benchmark. In *The Fourteenth International Conference on Learning Representations*, 2026.

Xiangyu Wang, Donglin Yang, Ziqin Wang, Hohin Kwan, Jinyu Chen, Hongsheng Li, Yue Liao, Si Liu, et al. Towards realistic uav vision-language navigation: Platform, benchmark, and methodology. In *International Conference on Learning Representations*, volume 2025, pp. 7292–7310, 2025.

Jonas Westheider, Julius Rückin, and Marija Popović. Multi-uav adaptive path planning using deep reinforcement learning. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 649–656. IEEE, 2023.

Ruipu Wu, Yige Zhang, Jinyu Chen, Linjiang Huang, Shifeng Zhang, Xu Zhou, Liang Wang, and Si Liu. Aeroduo: Aerial duo for uav-based vision and language navigation. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 2576–2585, 2025.

Jianqiang Xiao, Yuexuan Sun, Yixin Shao, Boxi Gan, Rongqiang Liu, Yanjin Wu, Weili Guan, and Xiang Deng. Uav-on: A benchmark for open-world object goal navigation with aerial agents. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 13023–13029, 2025.

Boao Yu, Zimo Chen, Junreng Rao, Yue Hu, Zhengqiu Zhu, Yong Zhao, and Rusheng Ju. Towards embodied air-ground cooperative object search: Benchmark, dataset and agentic method. *arXiv preprint arXiv:2609.08402*, 2026.

Tianle Zeng, Yanci Wen, and Hong Zhang. Carla-air: Fly drones inside a carla world–a unified infrastructure for air-ground embodied intelligence. *arXiv preprint arXiv:2603.28032*, 2026.

Daoxuan Zhang, Ping Chen, Xiaobo Xia, Xiu Su, Ruichen Zhen, Jianqiang Xiao, and Shuo Yang. Apex: A decoupled memory-based explorer for asynchronous aerial object goal navigation. *arXiv preprint arXiv:2602.00551*, 2026a.

Shuning Zhang, Liang Li, Yunheng Wang, Tao Wang, Yihang Kang, and Renjing Xu. Air-ground collaborative vision-and-language navigation via shared bird's-eye maps. *arXiv preprint arXiv:2609.03483*, 2026b.

Xin Zhou, Zhepei Wang, Hongkai Ye, Chao Xu, and Fei Gao. Ego-planner: An esdf-free gradient-based local planner for quadrotors. *IEEE Robotics and Automation Letters*, 6(2):478–485, 2020.

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

APPENDIX

A EVALUATION PROTOCOL AND METRICS

Metrics. Consider an evaluation split containing N episodes. For the i-th episode, let $\mathbf{x}_{i,t}^{g} \in \mathbb{R}^3$ and $\mathbf{x}_{i,t}^{a} \in \mathbb{R}^3$ denote the positions of the UGV and UAV at navigation step t, respectively, and let $\mathbf{x}_{i}^{\text{target}} \in \mathbb{R}^3$ denote the target position. Let $T_i \leq T_{\max}$ denote the terminal navigation step of the

episode. The UGV-to-target distance at step t is defined as $d^g_{i,t} = \left\|\mathbf{x}^g_{i,t} - \mathbf{x}^{\text{target}}_i\right\|_2$. The success indicator $S_i \in \{0, 1\}$ is defined as

$$S_i = \mathbb{I}\left[q_{i,T_i} = 1 \ \wedge \ d^g_{i,T_i} \le d_s\right], \tag{4}$$

where $q_{i,T_i} = 1$ indicates that the collaborative navigation system outputs the termination signal at step T_i, and $d_s = 5\,\text{m}$ denotes the success distance threshold. Based on these episode-level definitions, we compute the following evaluation metrics over the N episodes:

- **Success Rate (SR).** SR measures the fraction of successful episodes:

$$\text{SR} = \frac{1}{N}\sum_{i=1}^{N} S_i. \tag{5}$$

- **Oracle Success Rate (OSR).** OSR measures the fraction of episodes in which the UGV reaches within the success distance threshold at any point along its trajectory:

$$\text{OSR} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}\left[\min_{0\le t\le T_i} d^g_{i,t} \le d_s\right]. \tag{6}$$

- **Success weighted by Path Length (SPL).** Let ℓ_i and p_i denote the shortest traversable path length and the actual path length of the UGV, respectively. SPL is defined as

$$\text{SPL} = \frac{1}{N}\sum_{i=1}^{N} S_i \frac{\ell_i}{\max(\ell_i, p_i)}. \tag{7}$$

- **UGV Navigation Error (NE_{UGV}).** NE_{UGV} measures the average 3D distance between the UGV and the target at the terminal navigation step:

$$\text{NE}_{\text{UGV}} = \frac{1}{N}\sum_{i=1}^{N} d^g_{i,T_i}. \tag{8}$$

- **UAV Navigation Error (NE_{UAV}).** As a diagnostic metric for the UAV, NE_{UAV} measures the average horizontal distance between the UAV and the target at the terminal navigation step. Let $\Pi_{xy}(\cdot)$ denote projection onto the horizontal plane:

$$\text{NE}_{\text{UAV}} = \frac{1}{N}\sum_{i=1}^{N} \left\|\Pi_{xy}\left(\mathbf{x}^a_{i,T_i}\right) - \Pi_{xy}\left(\mathbf{x}^{\text{target}}_i\right)\right\|_2. \tag{9}$$

Evaluation Splits. The seen/unseen partition is defined at the environment level. The seen split contains 17 environments used during training, while the unseen split contains 2 held-out environments. Seen evaluation uses new episodes from the training environments with different start poses and relative target directions, whereas unseen evaluation assesses generalization to held-out environments and unseen target instances. To evaluate air–ground collaboration under different aerial visibility conditions, we further categorize test episodes into *aerial-visible* and *aerial-occluded* subsets based on whether the target is visible from the aerial viewpoint.

B Benchmark Construction Details

B.1 Instruction Construction

The goal-oriented instruction generation component consists of five stages: *Target Selection*, *Target Instance Description*, *Environmental Context Description*, *Coarse Direction Annotation*, and *Instruction Synthesis*. We detail each stage below.

(1) Target Selection. We select target instances with distinctive visual attributes or contextual cues. Each candidate target is then reviewed by multiple annotators to verify that it can be consistently described and identified from language instructions.

Target Instance Description Generation

Role : *You are a visual target description generator for an air-ground navigation dataset.*

Task : *Given multiple close-up views of the same target instance and its target name, generate a concise and discriminative description using only visually supported intrinsic attributes.*

Inputs :
- *Multiple close-up RGB images of the same target instance.*
- *Target name (e.g., "man", "pickup truck", "small booth", "traffic cone").*

Processing Steps :
- *Identify the selected target consistently across all input views using the provided target name.*
- *Extract distinctive intrinsic visual attributes of the target, including category, color, shape, texture, material, clothing, accessories, text, or other visible characteristics when applicable.*
- *Remove attributes that are uncertain, ambiguous, weakly visible, redundant, or not useful for distinguishing the target.*
- *Combine the remaining attributes into one concise and natural sentence.*

Constraints :
- *Describe only the target itself.*
- *Use only visually supported information.*
- *Do not infer invisible or uncertain properties.*
- *Do not describe surrounding objects, environmental context, location, or spatial direction.*
- *Avoid redundant and low-information details.*

Output Format : *Output exactly one concise sentence describing the target.*

Examples
- *A man with short blond hair, wearing a light green sweater, beige pants, and red shoes.*
- *A red pickup truck with a black front grille and an open cargo bed.*
- *A small white metal booth with dark windows and a flat black roof.*
- *A yellow traffic cone with two reflective white bands.*

Figure 5: Prompt for target instance description.

Target Surroundings Description Generation

Role : *You are a target-centric scene description generator for an air-ground navigation dataset.*

Inputs :
- *Target Name (e.g., car, building, person, etc.).*
- *Multiple RGB images showing the same selected target and its surrounding environment from different viewpoints.*

Task : *Given the name of the selected target, multiple views of the target, and its surrounding environment, generate a concise description of the target's immediate surroundings using visually distinctive and localization-relevant environmental cues.*

Processing Steps :
- *Locate the specified target consistently across all provided views using the target name and set it as the reference point for scene analysis.*
- *Identify nearby and visually distinctive environmental elements, including buildings, facilities, landmarks, people, vehicles, vegetation, roads, or other relevant objects.*
- *Retain only environmental cues that are spatially close to the target, visually salient, and useful for target localization.*
- *Remove distant, generic, redundant, uncertain, or low-information background details.*
- *Combine the remaining cues into one concise and natural sentence.*

Constraints :
- *Describe only the target's immediate surroundings.*
- *Use only visually supported information.*
- *Prioritize nearby and distinctive reference elements.*
- *Do not repeat the target's intrinsic appearance attributes.*
- *Avoid unnecessary background details.*

Output Format : *Output exactly one concise sentence describing the target's surroundings.*

Examples
- *The target is beside a small white booth, with another person nearby and several trucks parked around it.*
- *The target is near a bus shelter at a crossroads, with a pizza shop and several pedestrians nearby.*
- *The target is beside a red-brick building, with parked cars and trees along the roadside.*
- *The target is near a fountain in an open plaza, surrounded by benches and several people.*
- *The target is beside a loading area, with stacked containers and multiple trucks nearby.*

Figure 6: Prompt for environmental context description.

(2) Target Instance Description. For each selected target, we collect target-centric multi-view observations and use GPT-5.5 to generate a concise description of its semantic category and distinctive visual attributes. The description focuses on visually supported attributes of the target itself, while excluding surrounding environmental context. The corresponding prompt is shown in Fig. 5.

(3) Target Surroundings Description. We further collect multi-view observations of the environment surrounding the target from multiple directions. GPT-5.5 summarizes nearby objects, structures, and other localization-relevant environmental cues while avoiding redundant descriptions of the target itself. The corresponding prompt is shown in Fig. 6.

Navigation Instruction Synthesis

Role : *You are a navigation instruction synthesizer for an air-ground navigation task.*

Task : *Combine the provided target direction, target description, and surroundings description into one complete and natural navigation instruction.*

Inputs :
- *Direction: the target's direction relative to the drone.*
- *Target Description: the target category and its distinctive visual attributes.*
- *Surroundings Description: nearby environmental cues useful for target localization.*

Processing Steps :
- *Preserve the provided directional expression exactly.*
- *Use the target description to specify the target category and its most distinctive visual attributes.*
- *Append the most informative environmental cues from the surroundings description.*
- *Remove redundant or low-information details.*
- *Generate one complete navigation instruction following the required template.*

Constraints :
- *Preserve the direction exactly as provided.*
- *Describe the target before describing its surroundings.*
- *Do not introduce information that is absent from the inputs.*
- *Do not change or infer additional spatial relations.*
- *Avoid redundant descriptions.*
- *The final sentence must end with: Please help me navigate to the location of [target].*

Output Format : *The target is located to the [direction] of the drone. The description of the target and its surroundings is shown below. [Target description and surroundings description]. Please help me navigate to the location of [target].*

Examples :
The target is located to the south of the drone. The description of the target and its surroundings is shown below. The target is a yellow-and-blue SMOOTHIES & BUBBLE TEA food cart located on the sidewalk near a T-junction, immediately adjacent to the entrance of a park. A tree and a street lamp are nearby. Please help me navigate to the location of this food cart.

Figure 7: Prompt for goal-oriented instruction synthesis.

(4) Coarse Direction Annotation. For each target, annotators select multiple episode start locations from different directions. Since valid ground start locations must satisfy local traversability constraints, we use human selection rather than automatic sampling to ensure high-quality episode initialization. Given the start and target coordinates, we automatically compute the coarse relative direction from the start location to the target and quantize it into eight compass directions: north, northeast, east, southeast, south, southwest, west, and northwest.

(5) Navigation Instruction Synthesis. GPT-5.5 integrates the coarse relative direction, target description, and environmental context into the goal-oriented navigation instruction, without prescribing intermediate actions. The corresponding prompt is shown in Fig. 7.

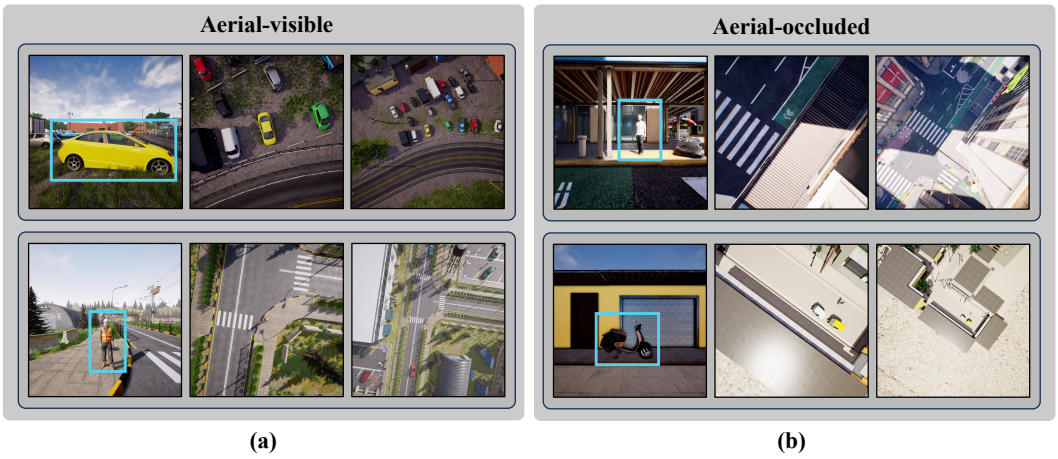


Figure 8: Representative aerial-visible and aerial-occluded targets.

B.2 Aerial-Visibility Annotation.

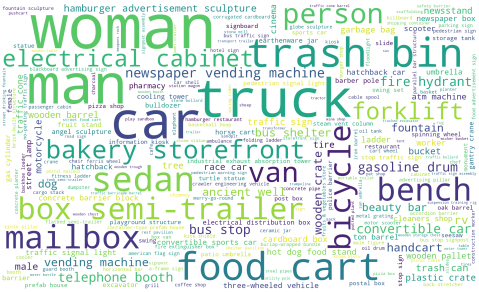
To evaluate collaborative navigation under different aerial target-visibility conditions, we partition all target instances in the test sets into *aerial-visible* and *aerial-occluded* subsets. Since current

vision-language models remain unreliable for fine-grained spatial occlusion reasoning, we manually annotate each target instance to ensure annotation quality. A target is labeled *aerial-visible* if it is directly observable from the top-down aerial viewpoint, and *aerial-occluded* if it is hidden from this viewpoint by roofs, overhangs, tree canopies, or other scene elements. Representative examples are shown in Fig. 8.

B.3 Human Quality Assurance.

We adopt a multi-stage human verification process to ensure the quality of the constructed episodes. Each candidate episode is reviewed for consistency across the target instance, goal-oriented instruction, air–ground reference trajectories, and aerial-visibility annotation. Reviewers verify that the target is uniquely identifiable, the instruction accurately describes the intended target and its surrounding context, and both aerial and ground trajectories are valid and executable. Episodes with correctable issues are manually revised and then cross-checked by a different annotator before being retained. Samples with ambiguous targets, invalid trajectories, or annotation errors that cannot be reliably resolved are discarded. For uncertain cases or disagreements between reviewers, an additional annotator examines the episode and determines the final annotation. After episode-level verification, all retained samples undergo a final dataset-level review to identify remaining inconsistencies across annotations and trajectories. Of the 14,697 initially constructed episodes, 6,375 were accepted without modification, 3,906 were retained after correction and re-verification, and 4,416 were discarded, resulting in the final 10,281 episodes.

C Additional Dataset Statistics



Target Name Word Cloud
(a)

Visual Attribute Word Cloud
(b)

Figure 9: Additional statistics of AirGroundVLN. (a) Target name word cloud. (b) Visual attribute word cloud.

Target and Visual Attribute Statistics. The word cloud in Fig. 9 (a) further highlights the semantic diversity of target entities, ranging from common vehicles and pedestrians to infrastructure-related objects such as fire hydrants, electrical cabinets, and traffic signal lights. Meanwhile, Fig. 9 (b) shows that target descriptions contain diverse fine-grained visual attributes, including color, shape, material, texture, and scale. Together, these statistics characterize the category-level semantics and instance-level appearance cues represented in AirGroundVLN.

Aerial Visibility Distribution. The test episodes are categorized into *aerial-visible* and *aerial-occluded* cases, with 77.3% and 22.7% of episodes falling into each category, respectively.

D Method Details

D.1 Training Objective.

For platform $v \in \{g, a\}$, let $Z_t^v \in \mathbb{R}^{H \times W}$ denote the predicted waypoint logits. The supervision target $Y_t^v \in \mathbb{R}^{H \times W}$ is constructed as a normalized Gaussian heatmap centered at the ground-truth

future waypoint of platform v. The predicted waypoint distribution is $P_t^v = \mathrm{Softmax}(Z_t^v)$. We combine distribution-level supervision with an auxiliary coordinate-level objective:

$$\mathcal{L}_v = D_{\mathrm{KL}}\left(Y_t^v \parallel P_t^v\right) + \lambda_c \,\mathrm{SmoothL1}\left(\mu(P_t^v), \mu(Y_t^v)\right), \tag{10}$$

where $\mu(\cdot)$ denotes the differentiable expected spatial coordinate under a probability map. Let $\mathbf{r}_{ij} \in \mathbb{R}^2$ denote the spatial coordinate associated with grid cell (i, j). The spatial expectation is computed as $\mu(P) = \sum_{i=1}^{H}\sum_{j=1}^{W} P_{ij}\mathbf{r}_{ij}$. The KL divergence supervises the predicted waypoint distribution with the Gaussian supervision distribution, while the coordinate-level term further constrains its expected spatial coordinate. The aerial and ground objectives are jointly optimized as

$$\mathcal{L}_{\mathrm{total}} = \mathcal{L}_g + \lambda_a \mathcal{L}_a, \tag{11}$$

where λ_c controls the coordinate-level supervision and λ_a balances the aerial objective relative to the ground objective.

D.2 Target Approach and Termination

Once a target candidate is confirmed, its 3D position $\hat{\mathbf{x}}^{\mathrm{target}}$ is estimated from synchronized RGB-D observations in the same coordinate frame as the UGV. Since the detected target may not lie on the traversable surface, only the horizontal coordinates $\hat{\mathbf{x}}_{xy}^{\mathrm{target}}$ are used as the approach goal. The UGV then executes a local planner to approach this estimated target position (Zhang et al., 2026a). No additional VLM-based waypoint prediction is performed during this final approach stage. The approach radius is set to $d_{\mathrm{stop}} = 1.5\,\mathrm{m}$, measured in the horizontal plane, and the maximum approach budget is $T_{\mathrm{app}} = 20$ iterations. The approach terminates when the UGV is within d_{stop} of $\hat{\mathbf{x}}_{xy}^{\mathrm{target}}$ or when the execution budget is exhausted. The stopping radius is a controller termination condition and is distinct from the benchmark success criterion, which evaluates the final UGV pose against the ground-truth target position.

E Implementation Details

AG-CoNAV is built upon the pretrained Qwen3.5-2B (Team, 2026) vision-language backbone. We keep the pretrained backbone weights frozen and train LoRA adapters (Hu et al., 2021). We use AdamW (Loshchilov & Hutter, 2017) with an initial learning rate of 2×10^{-5} and a cosine learning-rate decay schedule. For waypoint supervision, we set the coordinate-level loss weight to $\lambda_c = 10.0$ and the aerial loss weight to $\lambda_a = 0.5$. The SACM memory bank stores up to $M = 16$ historical air–ground entries. The maximum navigation budget is set to $T_{\max} = 200$ steps. During navigation, the UGV uses only a front-facing camera. After executing each step, it performs a 360° sweep using 6 camera angles for target verification. Once a target candidate is confirmed, the UGV enters the final approach stage, for which we set the stopping radius to $d_{\mathrm{stop}} = 1.5\,\mathrm{m}$ and the maximum number of approach steps to $T_{\mathrm{app}} = 20$. All methods use the same target verification and final approach procedure to ensure a fair comparison.

Training-Free Baseline Setup. For the VLM-based training-free baselines, the model receives the goal-oriented instruction together with the current ground and aerial BEV observations and predicts the next UGV and UAV waypoints. The overall navigation procedure is summarized in Algorithm 1. The waypoint generation model $\mathcal{M}_{\mathrm{VLM}}$ uses the prompt shown in Fig. 10 to jointly predict the next UGV navigation waypoint $\hat{w}_t^g$ and UAV observation waypoint $\hat{w}_t^a$ from the instruction and current air–ground observations. The target verifier $\mathcal{V}$ uses the prompt shown in Fig. 11 to determine whether the current observations match the instructed target by jointly considering candidate appearance attributes and surrounding environmental context. It outputs a confidence score s_t used for termination.

F Qualitative Navigation Analysis

To demonstrate the navigation behavior of different approaches, we present qualitative comparisons across multiple scenarios in AirGroundVLN. First, Fig. 12 provides bird's-eye-view (BEV) trajectory visualizations across three distinct navigation episodes, including ***two episodes from unseen environments***, comparing our proposed AG-CoNAV against representative training-free baselines

Algorithm 1 Training-Free Air–Ground Navigation

Require: instruction I, initial poses (P_0^g, P_0^a), maximum navigation budget $T_{\max}$, verification threshold τ, stopping radius d_{stop}, maximum approach steps T_{app}

Initialize the UGV and UAV at (P_0^g, P_0^a) and set $t \leftarrow 0$.
while $t < T_{\max}$ **do**
  Acquire the current ground and aerial BEV observations (o_t^g, o_t^a).
  Verify the target using appearance and surrounding context matching: $(b_t, s_t) \leftarrow \mathcal{V}(I, o_t^g, o_t^a)$, where b_t indicates whether the target is verified and s_t denotes the target-context matching confidence.
  if $b_t \wedge (s_t \geq \tau)$ **then**
    Estimate the confirmed target position $\hat{\mathbf{x}}^{\text{target}}$ from RGB-D observations.
    Execute final approach toward $\hat{\mathbf{x}}^{\text{target}}$ with stopping radius d_{stop} and step limit T_{app}.
    Terminate the episode.
    break
  end if
  Predict air–ground waypoints $(\hat{w}_t^g, \hat{w}_t^a) \leftarrow \mathcal{M}_{\text{VLM}}(I, o_t^g, o_t^a)$.
  Generate navigation trajectories $\hat{u}_t^g \leftarrow \mathcal{P}_g(P_t^g, \hat{w}_t^g)$ and $\hat{u}_t^a \leftarrow \mathcal{P}_a(P_t^a, \hat{w}_t^a)$.
  Execute $(\hat{u}_t^g, \hat{u}_t^a)$ and update (P_{t+1}^g, P_{t+1}^a).
  $t \leftarrow t + 1$.
end while

Air-Ground Training-Free Baseline Prompt

Role : *You are an Air-Ground Navigation Agent.*

Task : *Given a natural-language task instruction describing a target, an aerial top-down RGB image, and a ground front-facing RGB image, simultaneously predict:1. The UGV's next navigation waypoint (ground_waypoint).2. The UAV's next observation waypoint (air_waypoint).*

Input and coordinate rules:
1. The two input images are provided in a fixed order:
- Image 1 is the complete current UAV top-down bird's-eye RGB image.
- Image 2 is the current UGV front RGB image.
2. Image 1 is the only coordinate reference for both waypoints. Image 2 provides target appearance and ground-level environmental context only. Never output coordinates relative to Image 2 or directly transfer pixel locations from Image 2 to Image 1.
3. Both waypoints must use normalized pixel coordinates relative to Image 1:
- the coordinate origin is the top-left corner;
- x is the horizontal coordinate in [0, 1], increasing to the right;
- y is the vertical coordinate in [0, 1], increasing downward.

Waypoint selection rules: :
1. Ground_waypoint is the UGV's next navigation target.
2. Air_waypoint is the UAV's next horizontal observation position.
3. The two waypoints are independent and may differ.

Output Format : *{ "ground_waypoint": { "x": 0.0, "y": 0.0 }, "air_waypoint": { "x": 0.0, "y": 0.0 } }*

Example: *{ "ground_waypoint": { "x": 0.62, "y": 0.48 }, "air_waypoint": { "x": 0.55, "y": 0.41 } }*

Figure 10: Air–ground waypoint generation prompt.

(Qwen3.5-2B, GPT-5.5) and a training-based baseline (*UGV + Global BEV Map*), alongside ground-truth (GT) paths. As shown in Fig. 12, AG-CoNAV consistently reaches the target locations across all three scenarios with reasonable path execution. In contrast, the zero-shot baselines (Qwen3.5-2B and GPT-5.5) often select incorrect directions at crucial intersections due to the difficulty of long-range spatial reasoning without task-specific training. Meanwhile, the *UGV + Global BEV Map* baseline, constrained by a static global map and a lack of fine-grained local details, frequently mistargets inaccurate goals or fails during execution. Furthermore, Fig. 13 and Fig. 14 detail step-by-step navigation visualizations in *seen-visible* and *unseen-occluded* environments, respectively. Notably, AG-CoNAV maintains accurate target localization and executable path planning even in ***unseen*** environments with ***aerial occlusions***.

Air-Ground Target Verification Prompt

Role : *You are a visual target verification module for an air-ground navigation system.*

Task : *Determine whether the current ground observations contain the target described by the navigation instruction and identify the best matching detection candidate.*

Inputs :
- *Six directional RGB views captured by the ground robot.*
- *A target description containing the target's intrinsic visual attributes.*
- *A navigation instruction containing the target's surrounding environmental context.*
- *Candidate target crops produced by Grounding DINO.*

Processing Steps :
- *Analyze all six directional views and evaluate how well the observed scene matches the target surroundings described in the navigation instruction.*
- *Examine all candidate crops and compare them with the target description using visible attributes such as category, color, shape, material, text, clothing, accessories, or other distinctive characteristics.*
- *Select the candidate that best matches the target description.*
- *Determine whether the selected candidate provides sufficient visible evidence to verify the target.*
- *Estimate a normalized scene-matching confidence score in [0,1] representing the consistency between the observed surroundings and the environmental context described in the instruction.*

Decision Rules :
- *Use all six directional views for surroundings matching.*
- *Use candidate crops only for target appearance verification.*
- *Rely only on visible evidence.*
- *Do not treat Grounding DINO confidence scores or predicted phrases as proof of target identity.*
- *If no candidate reliably matches the target description, set is_target to false and best_detection_index to -1.*
- *confidence represents surroundings consistency, not the Grounding DINO detection confidence.*

Output Constraints :
- *is_target must be either true or false.*
- *confidence must be a number in [0,1].*
- *best_detection_index must be the selected candidate index, or -1 if no candidate is verified.*
- *reason must contain only 1–5 words.*
- *Output only the required JSON object without Markdown or additional explanations.*

Output Format : *{ "is_target": true, "confidence": 0.0, "best_detection_index": 0, "reason": "matching visual attributes" }*

Figure 11: Target verification prompt.

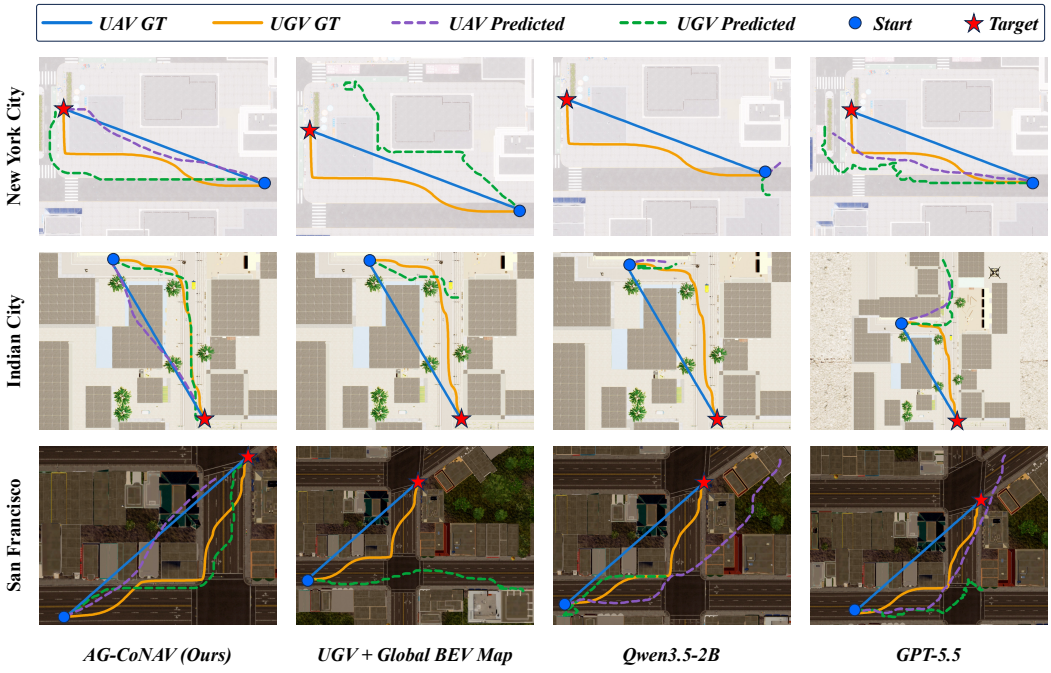


Figure 12: **Bird's-eye-view trajectory comparisons on AirGroundVLN.** Joint UAV and UGV trajectories from four methods (Qwen3.5-2B, GPT-5.5, *UGV + Global BEV Map*, and AG-CoNAV) are visualized across three representative episodes alongside ground-truth (GT) paths.

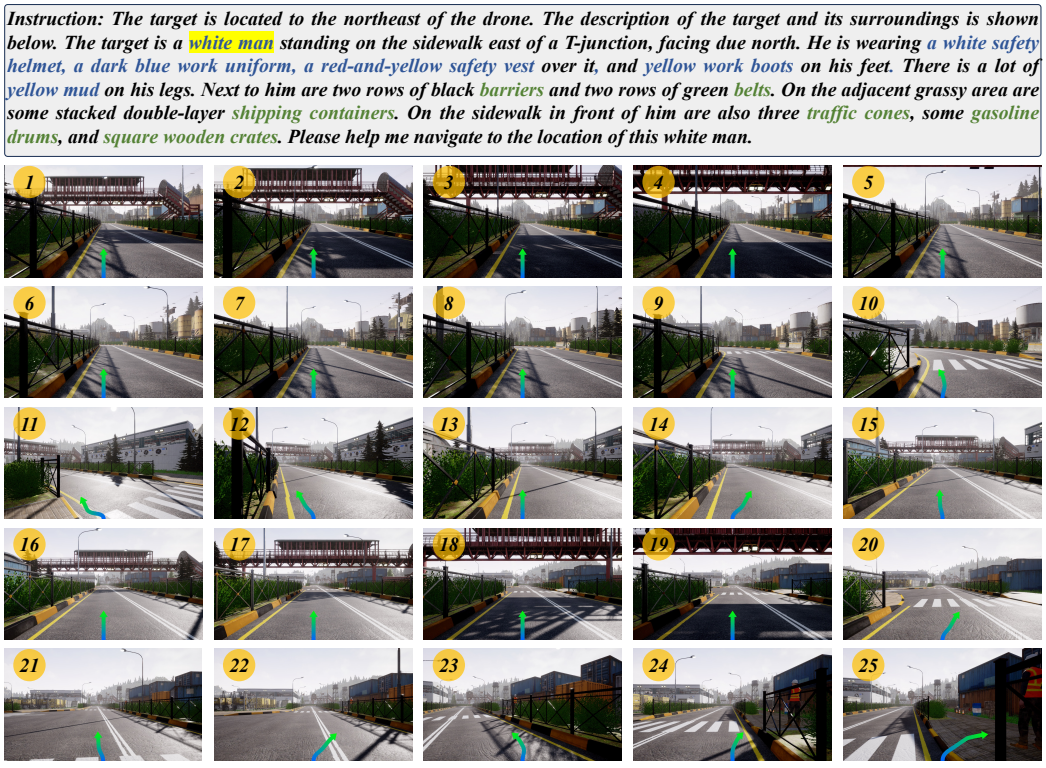


Figure 13: **Qualitative navigation trajectory visualization on AirGroundVLN.** Navigation trajectories toward language-specified targets are visualized in ***seen*** environments with ***visible*** aerial observations.

Instruction: *The target is located to the northwest of the drone. The description of the target and its surroundings is shown below. The target is a dark-skinned woman with golden curly hair, wearing a green strapless sleeveless dress and sandals. She stands facing west on the stone-paved sidewalk immediately adjacent to the northern edge of a T-intersection, with a yellow tactile paving strip right beside her left foot. To her northeast is the "BLACK & WHITE" store. Across the road to the west are two red "SocaSola" signs mounted on a storefront and the "MAXD" store. To the southeast, along the roadside, stand one orange and one blue metal mailbox. To the west lies an east-west pedestrian crossing. On the tactile paving to the north-northwest stands a man facing west, with golden curly hair, facial stubble, fair complexion, wearing a white-and-green collared shirt, dark trousers, and black cloth shoes. A green non-motorized lane runs along the road to the north-northwest. Please help me navigate to the location of this woman.*

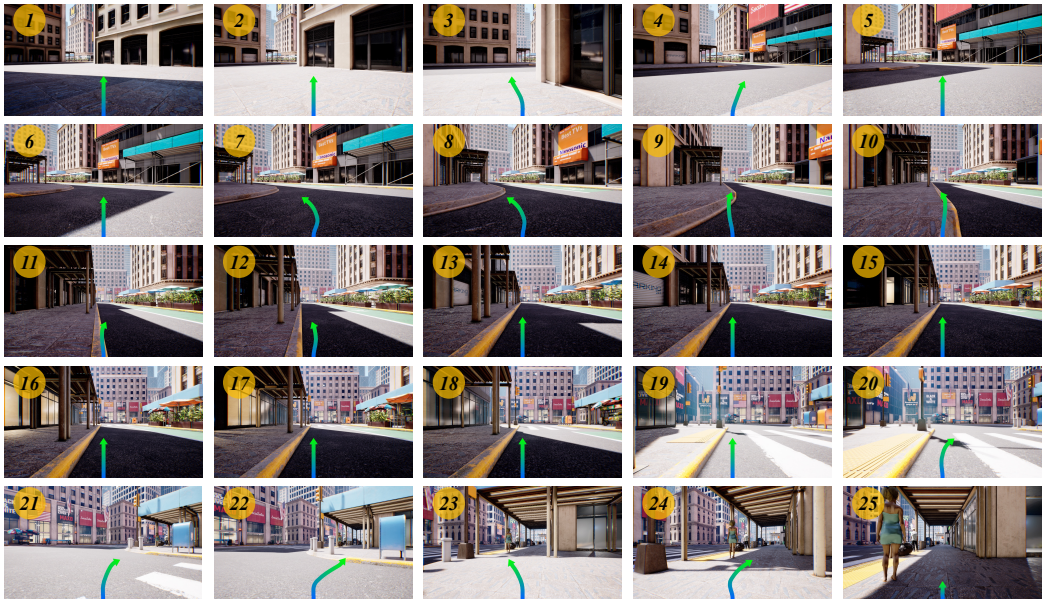


Figure 14: **Qualitative navigation trajectory visualization on AirGroundVLN.** Navigation trajectories toward language-specified targets are visualized in ***unseen*** environments with ***occluded*** aerial observations.