VIREO: LEARNING HOW TO LOOK AND WHAT TO TRUST FOR AERIAL OBJECT NAVIGATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Aerial object navigation (ObjectNav) requires an unmanned aerial vehicle (UAV) to search for a target specified by a textual description using visual observations. Existing methods mainly rely on spatial search signals to guide navigation, while overlooking that target recognition can vary substantially across observation viewpoints. Thus, approaching a promising candidate does not necessarily ensure reliable recognition. Even at similar distances, the same target can provide substantially different visual evidence from different viewing directions. To bridge this gap, we propose **V**iewpoint-**I**nformed **R**ecognition and **E**vidence **O**ptimization (**VIREO**), a viewpoint-aware framework for aerial ObjectNav. To identify viewpoints that better support target recognition during navigation, VIREO models how recognition outcomes for target and non-target candidates vary across observation viewpoints. Based on these response patterns, VIREO constructs a viewpoint information field to guide the UAV toward viewpoints with higher expected information gain. To address evidence accumulation across viewpoints, VIREO introduces viewpoint-aware evidence aggregation, which adjusts each observation's contribution according to the similarity to previous viewpoints. Experiments on UAV-ON show that VIREO surpasses the best baseline.[1]

1 INTRODUCTION

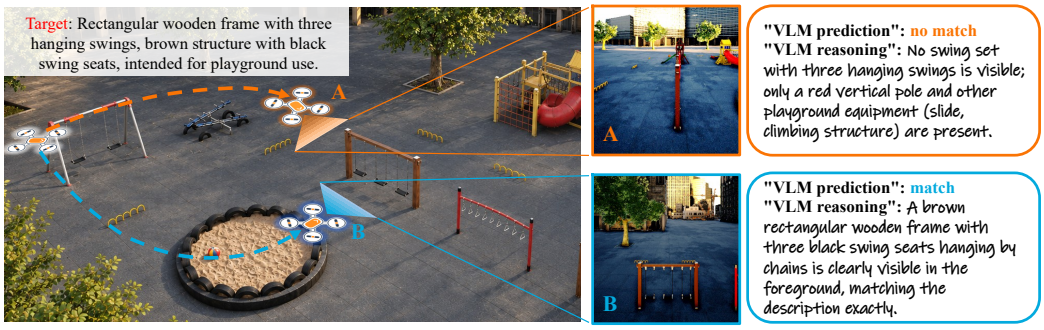

Figure 1: Viewpoint-dependent recognition of the same swing-set target. The VLM predicts no match from viewpoint A and a match from viewpoint B under the same target description.

Aerial object navigation (ObjectNav) requires an unmanned aerial vehicle (UAV) to search for a target specified by a textual description in an unknown environment using onboard visual observations (Xiao et al., 2025). This task requires the UAV to recognize targets across changes in altitude and viewing direction while searching large outdoor environments (Lee et al., 2025; Chu et al., 2024). The resulting diversity of observations and extended search trajectories place demands on both target recognition and the integration of historical observations for navigation (Zhang et al., 2025b;a).

Existing approaches primarily address **where to look** by using exploration progress and target-related cues to select promising regions for the UAV to search (Xiao et al., 2025; Zhang et al., 2026; Zhou et al., 2026). However, approaching a promising candidate does not necessarily provide

[1]The code will be released upon acceptance.

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.

sufficient visual evidence for reliable recognition. As illustrated in Fig. 1, observations of the same target from different viewing directions can lead to different vision-language model (VLM) judgments (Ruan et al., 2024). Such viewpoint-dependent judgments can produce conflicting recognition evidence along a trajectory, potentially prompting unnecessary movements or premature abandonment of a valid candidate. While searching for the target, geometrically similar viewpoints may produce correlated responses (Velez et al., 2012), which can lead to overconfident beliefs if treated as independent evidence (Marques et al., 2024). These challenges highlight the need to decide both **how to look and what to trust** during navigation.

To address these challenges, we propose **V**iewpoint-**I**nformed **R**ecognition and **E**vidence **O**ptimization (**VIREO**), a viewpoint-aware framework for aerial ObjectNav. VIREO guides viewpoint selection during navigation through a viewpoint information field and integrates recognition evidence collected along the trajectory through viewpoint-aware evidence aggregation. These mechanisms form a feedback loop: new observations update the belief that the current candidate matches the target description, and this belief guides subsequent viewpoint selection and decisions to continue verification, accept the candidate, or resume searching.

Specifically, to determine **how to look**, VIREO learns viewpoint-dependent recognition response patterns for matching and non-matching candidates from offline observations in the training environments. The viewpoint information field combines these patterns with the current candidate belief to estimate the expected information gain about the candidate's match status, guiding the UAV toward informative viewpoints for further verification. To determine **what to trust**, VIREO uses the same response model to quantify the evidence each new observation provides for or against a match. Viewpoint-aware evidence aggregation then weights this evidence by the geometric novelty of the executed viewpoint relative to previous viewpoints for the same candidate, reducing the influence of repetitive observations on the accumulated belief. Our main contributions are summarized as follows:

- Unlike existing aerial ObjectNav methods focused on spatial search, we account for the fact that proximity alone does not ensure reliable recognition across viewpoints. We therefore propose VIREO, a viewpoint-aware framework that couples spatial search with active viewpoint planning, using accumulated recognition evidence to guide UAV motion and candidate decisions.
- To guide the UAV toward more informative viewpoints, we introduce a viewpoint information field that estimates the expected information gain about a candidate's match status using recognition response models for matching and non-matching candidates.
- To reduce belief overconfidence caused by treating correlated recognition responses as independent evidence, we introduce viewpoint-aware evidence aggregation that weights each observation's contribution by the geometric novelty of its viewpoint, reducing the influence of repetitive observations on candidate belief updates.
- We evaluate VIREO on the UAV-ON benchmark and show that it outperforms existing baselines.

2 Related Work

2.1 Aerial Object-Goal Navigation

ObjectNav requires an agent to locate and reach an object of a specified category in an unknown environment (Chaplot et al., 2020). Foundational work focuses on indoor platforms and datasets, including AI2-THOR (Kolve et al., 2017), Gibson (Xia et al., 2018), RoboTHOR (Deitke et al., 2020), and HM3D (Ramakrishnan et al., 2021). Approaches range from semantic exploration and potential-based planning (Chaplot et al., 2020; Ramakrishnan et al., 2022) to imitation and reinforcement learning (Ramrakhya et al., 2023) and language-driven zero-shot search (Gadre et al., 2023). VLFM uses vision-language similarity to guide frontier selection (Yokoyama et al., 2024). These methods connect visual observations and goal information with exploration decisions in indoor scenes. Aerial navigation introduces outdoor search with greater variation in altitude and viewpoint. AerialVLN studies route-instruction following (Liu et al., 2023), whereas UAV-ON establishes target-description-driven aerial ObjectNav and introduces AOA for semantic goal-directed

exploration (Xiao et al., 2025). APEX combines spatial-semantic mapping, action selection, and target grounding (Zhang et al., 2026); OctMem-Agent uses adaptive octree memory to retrieve scene and exploration information (Zhou et al., 2026). These aerial ObjectNav approaches emphasize spatial search but fail to account for the sensitivity of target evidence to viewing direction. VIREO addresses this gap by actively selecting informative viewpoints to improve the efficiency and stability of aerial object navigation.

2.2 Scene Representations for Navigation

Navigation agents construct scene representations to preserve observations and support target search. Semantic maps enable goal-conditioned exploration and frontier selection (Chaplot et al., 2020; Ramakrishnan et al., 2022), while generative maps predict unobserved regions from available context (Zhang et al., 2024b). Open vocabulary methods ground visual-language features in spatial maps (Huang et al., 2023) or continuous fields (Shafiullah et al., 2023), allowing language queries to identify navigation goals. Object centric and hierarchical scene graphs further organize these features into objects, regions, and their spatial relationships (Gu et al., 2024; Werby et al., 2024). Historical observations are also aggregated to retain information relevant to navigation. GridMM (Wang et al., 2023) weights features within each grid cell according to instruction relevance, whereas VER (Liu et al., 2024) integrates multi-view features into a volumetric representation through cross-view attention. Graph-based memories organize historical observations and semantic relationships to guide navigation. BRM (Wu et al., 2019) updates probabilistic relations between semantic entities, and MemoNav (Li et al., 2024) combines selective forgetting with global memory to retain goal-relevant scene features. These representations support decisions about both exploration and target approach. DAT (Dang et al., 2023) maintains a target-oriented memory graph to guide navigation after the target has been observed. MapGPT (Chen et al., 2024) expresses topological connectivity and navigation history as prompts for path planning. L3MVN (Yu et al., 2023) uses semantic context to rank exploration frontiers, while VoroNav (Wu et al., 2024) combines topological paths with scene descriptions to select waypoints. These approaches primarily organize historical observations into scene representations to support navigation. In contrast, VIREO uses the geometric relationships between current and historical observation poses to weight recognition evidence for candidate verification. By downweighting responses from similar viewpoints, VIREO reduces the influence of repetitive observations on candidate belief updates.

3 Method

3.1 Problem Formulation

Given a textual description g, a UAV searches for the target without access to its ground-truth location. At high-level planning step t, it receives multi-view RGB-D observations O_t and state s_t, including its pose. RGB supports semantic planning, while depth supports low-level obstacle avoidance. The agent maintains the interaction history

$$H_t = \{(O_k, s_k, z_k, \mathbf{a}_k)\}_{k=1}^{t-1}, \tag{1}$$

and selects a high-level decision according to

$$z_t = \pi_{\text{plan}}(g, O_t, s_t, H_t), \tag{2}$$

where π_{plan} selects a motion goal or termination decision z_t. The low-level controller executes

$$\mathbf{a}_t = (a_{t,1}, \ldots, a_{t,m_t}), \qquad a_{t,j} \in \mathcal{A}, \tag{3}$$

before the next planning step or termination. Here, $\mathcal{A}$ includes *Stop*, and m_t counts executed actions under the episode-level budget:

$$\sum_{k=1}^{t} m_k \le B_{\max}, \qquad B_{\max} = 150. \tag{4}$$

The episode ends on *Stop*, collision, or budget exhaustion. Under UAV-ON (Xiao et al., 2025), success requires stopping within 20 m of the target without collision.

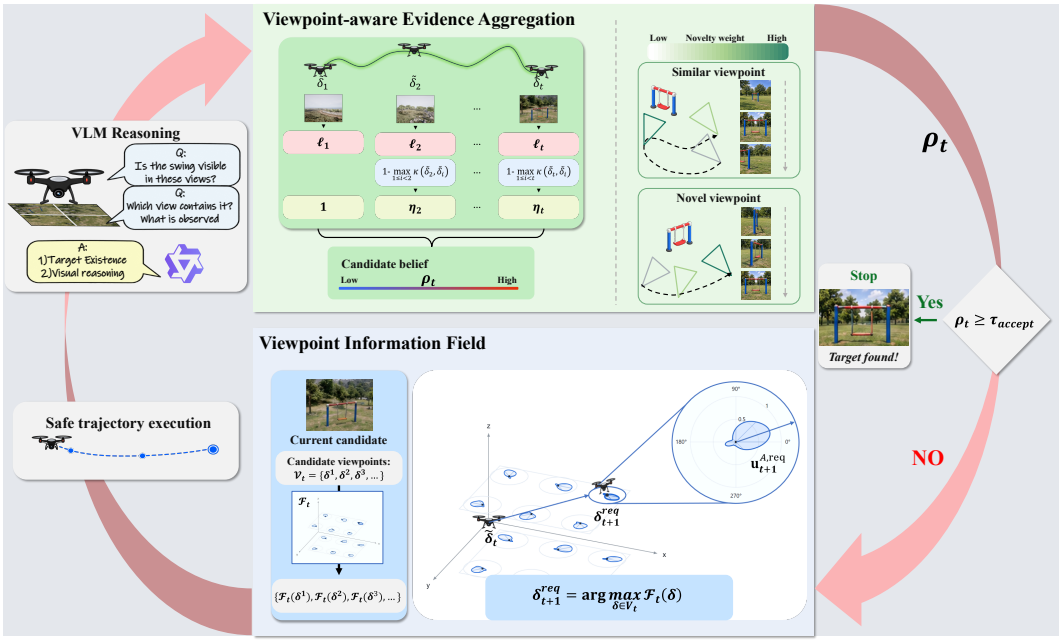

Figure 2: Core planning and evidence aggregation mechanism of VIREO. We use a viewpoint information field to select eligible viewpoints with maximum expected information gain. After motion execution, viewpoint-aware evidence aggregation weights the resulting recognition evidence by geometric novelty to update candidate belief. The updated belief guides subsequent viewpoint selection and stopping decisions.

3.2 Overview of VIREO

VIREO treats aerial ObjectNav as a joint spatial search and active viewpoint planning problem. The UAV explores to discover candidate objects, then accumulates recognition evidence to decide whether to stop, gather further observations, or resume spatial search.

As illustrated in Fig. 2, VIREO consists of two core components: a viewpoint information field and viewpoint-aware evidence aggregation. The viewpoint information field guides the UAV toward viewpoints with high expected information gain about a candidate's match status. Viewpoint-aware evidence aggregation uses geometric similarity to historical viewpoints to downweight repetitive evidence in candidate belief updates. The updated belief guides subsequent viewpoint selection and candidate decisions. To construct these two components, we first establish an observation likelihood model that relates VLM judgments to observation viewpoints and candidate match status.

3.3 Observation Likelihood Model

A matching candidate may receive different VLM judgments across viewpoints (Ruan et al., 2024), while a non-matching candidate may also elicit a positive response (Guan et al., 2024). We therefore model how the probability of a positive response varies with viewpoint under each hypothesis. These likelihoods are used to compute expected information gain for viewpoint selection and recognition evidence for candidate belief updates.

Let $\xi \in \{0, 1\}$ denote a candidate's underlying match status and $y \in \{0, 1\}$ the VLM's judgment from a single image and g, with 1 indicating a match in each case. We fix a local reference frame A at the camera pose $(\mathbf{p}_0, \mathbf{R}_0)$ of the candidate's initially selected view, avoiding the need for target coordinates. Superscript A denotes this frame, and t indexes planning steps. For camera position $\mathbf{p}$ and unit viewing direction $\mathbf{f}$, the relative viewpoint is

$$\delta = \left(\Delta\mathbf{p}^A, \mathbf{u}^A\right), \qquad \Delta\mathbf{p}^A = \mathbf{R}_0^\top(\mathbf{p} - \mathbf{p}_0), \qquad \mathbf{u}^A = \mathbf{R}_0^\top \mathbf{f}. \tag{5}$$

Thus, $\delta \in \Omega^A \subseteq \mathbb{R}^3 \times \mathbb{S}^2$ represents the camera displacement and viewing direction relative to the initial observation. Using labeled single-image observations from the training environments, we fit a viewpoint-conditioned response model:

$$r_\xi(\delta) := P_\theta(y = 1 \mid \xi, \delta) = \sigma\left(\mathbf{w}_\xi^\top \boldsymbol{\phi}(\delta) + b_\xi\right). \tag{6}$$

Here, σ is the logistic sigmoid, $\boldsymbol{\phi}(\delta)$ contains standardized geometric features, and $\theta = \{\mathbf{w}_0, b_0, \mathbf{w}_1, b_1\}$ denotes the model parameters. Offline viewpoints are expressed relative to each reference object's first collected camera pose. We fit the parameters using L_2-regularized binary cross-entropy, pooling observations across target descriptions.

The conditional probabilities $r_1(\delta)$ and $r_0(\delta)$ describe positive responses under the matching and non-matching hypotheses, respectively. They define the response likelihood

$$p_\theta(y \mid \xi, \delta) = r_\xi(\delta)^y \left[1 - r_\xi(\delta)\right]^{1-y}, \tag{7}$$

which supports both prospective viewpoint scoring and evidence accumulation during navigation.

3.4 Viewpoint Information Field

Using the observation likelihood model, we evaluate how much a response from each prospective viewpoint is expected to reduce uncertainty about a candidate's match status (Rainforth et al., 2024). Let $\rho_t \in (0, 1)$ denote the approximate belief that the active candidate matches g after processing the recognition response at planning step t. For a prospective viewpoint δ, the predicted positive-response probability is

$$q_t(\delta) = \rho_t r_1(\delta) + (1 - \rho_t) r_0(\delta). \tag{8}$$

We define the viewpoint information field as the expected information gain about the candidate label (Denzler & Brown, 2002; Foster et al., 2021):

$$\mathcal{F}_t(\delta) = H_\mathrm{b}(q_t(\delta)) - \rho_t H_\mathrm{b}(r_1(\delta)) - (1 - \rho_t) H_\mathrm{b}(r_0(\delta)), \tag{9}$$

where $H_\mathrm{b}(p) = -p \log p - (1 - p) \log(1 - p)$. Under the fitted model and current belief, $\mathcal{F}_t(\delta)$ is the mutual information between the candidate label and a prospective response, measured in nats. When $r_1(\delta) = r_0(\delta)$, the response cannot distinguish the matching and non-matching hypotheses, and $\mathcal{F}_t(\delta) = 0$.

To select the next viewpoint, we evaluate this field at the predicted camera viewpoints of proposed UAV motions. For each motion, the camera pose is computed from the proposed UAV pose and camera extrinsics, with the forward camera oriented toward the estimated candidate direction by default. We filter options using similarity to previous viewpoints, a minimum information score, the standardized feature range, and limits on consecutive requests using the same onboard camera. The similarity filter uses the kernel in Eq. 12. Let $\mathcal{V}_t$ denote the remaining viewpoints. When this set is nonempty, VIREO selects

$$\delta_{t+1}^{\mathrm{req}} = \arg\max_{\delta \in \mathcal{V}_t} \mathcal{F}_t(\delta) \tag{10}$$

and passes the corresponding spatial goal to the low-level controller. The direction component of $\delta_{t+1}^{\mathrm{req}}$ is denoted by $\mathbf{u}_{t+1}^{A,\mathrm{req}}$ and is expressed in frame A. If no option passes the filters, local motion planning returns to the base planner, while evidence accumulation and candidate decisions remain active. The next valid response updates the belief used for subsequent viewpoint selection.

3.5 Viewpoint-Aware Evidence Aggregation

After acquiring a new observation, we update the candidate belief using the VLM response and actual camera viewpoint. Similar viewpoints can yield redundant evidence despite viewpoint filtering, causing overconfidence if treated as independent. We therefore weight each valid response's log-likelihood contribution by its viewpoint's geometric novelty relative to previous observations of the same candidate.

For each valid recognition response y_t, we use the actual camera viewpoint $\widetilde{\delta}_t$ and the response model from Section 3.3 to compute the log-likelihood ratio:

$$\ell_t = \log \frac{p_\theta\left(y_t \mid \xi = 1, \widetilde{\delta}_t\right)}{p_\theta\left(y_t \mid \xi = 0, \widetilde{\delta}_t\right)}. \tag{11}$$

Positive values support the target hypothesis, while negative values support the non-target hypothesis. To determine the weight assigned to this evidence, we compare the current viewpoint with previous viewpoints of the same candidate using camera position and viewing direction. For viewpoints $\delta_i = (\Delta\mathbf{p}_i^A, \mathbf{u}_i^A)$ and $\delta_j = (\Delta\mathbf{p}_j^A, \mathbf{u}_j^A)$, we define geometric similarity as

$$\kappa(\delta_i, \delta_j) = \exp\left(-\frac{\|\Delta\mathbf{p}_i^A - \Delta\mathbf{p}_j^A\|_2^2}{2\sigma_p^2} - \frac{d_{\text{ang}}^2(\mathbf{u}_i^A, \mathbf{u}_j^A)}{2\sigma_u^2}\right), \tag{12}$$

where $d_{\text{ang}}(\mathbf{u}, \mathbf{v}) = \arccos(\text{clip}(\mathbf{u}^\top\mathbf{v}, -1, 1))$. The parameters σ_p and σ_u control the spatial and angular scales. This similarity serves as a proxy for observation redundancy. Let $\mathcal{I}_t$ contain the planning-step indices of all previous valid recognition responses for the active candidate:

$$\mathcal{I}_t = \{i \mid t_c \le i < t,\ y_i \text{ is valid for the active candidate}\}.$$

The evidence weight is

$$\eta_t = \begin{cases} 1, & \mathcal{I}_t = \varnothing, \\ 1 - \max\limits_{i \in \mathcal{I}_t} \kappa\left(\widetilde{\delta}_t, \widetilde{\delta}_i\right), & \mathcal{I}_t \neq \varnothing. \end{cases} \tag{13}$$

For a candidate first proposed at step t_c, initialize its log-odds as $L_{t_c-1} = \log[\rho_{\text{init}}/(1-\rho_{\text{init}})]$. At a step with a valid recognition response, we update the log-odds and candidate belief as follows

$$\begin{aligned} L_t &= L_{t-1} + \eta_t \ell_t, \\ \rho_t &= \sigma(L_t). \end{aligned} \tag{14}$$

Geometric novelty scales each log-likelihood contribution, yielding a weighted belief update related to generalized Bayesian inference (Bissiri et al., 2016). Invalid responses leave the belief unchanged.

The updated belief guides viewpoint selection and candidate decisions within the framework of active sequential hypothesis testing (Naghshvar & Javidi, 2013). VIREO issues *Stop* when $\rho_t \ge \tau_{\text{accept}}$ and releases the candidate when $\rho_t \le \tau_{\text{reject}}$. If neither threshold is reached, the agent continues gathering observations unless the valid-response budget K_{obs} or viewpoint-request budget K_{req} is exhausted, in which case the candidate is released. Stopping depends on accumulated evidence without a separate target-distance check; navigation success is evaluated using the UAV-ON criterion in Section 3.1.

4 Experiments

4.1 Experimental Setting

Dataset. We evaluate VIREO on all test tasks across the 14 outdoor environments of UAV-ON (Xiao et al., 2025). The benchmark contains 1,270 annotated target objects, each associated with a textual description. The UAV receives front, left, right, and downward camera observations. Observation roles, the episode-level action budget, and the success criterion follow Section 3.1.

Baselines. We compare VIREO with eight methods evaluated on UAV-ON. Random, CLIP-H, and AOA-F are benchmark baselines (Xiao et al., 2025). AerialVLN (Liu et al., 2023), NaVid (Zhang et al., 2024a), and OpenFly (Gao et al., 2026) represent vision-language navigation approaches evaluated in aerial settings. APEX (Zhang et al., 2026) and OctMem-Agent (Zhou et al., 2026) are recent aerial ObjectNav methods. For AerialVLN, NaVid, and OpenFly, we report the results of their implementations on UAV-ON evaluated by Zhou et al. (2026). Results for APEX and OctMem-Agent are taken directly from their respective original papers.

Evaluation Metrics. Performance is measured by Success Rate (SR), Oracle Success Rate (OSR), and Success weighted by Path Length (SPL) (Anderson et al., 2018). SR measures the percentage of episodes in which the UAV explicitly stops within 20 m of the target without collision. OSR measures the percentage of episodes in which the UAV enters this success radius at any point and incurs no collision during the episode. The OSR–SR gap represents episodes that reach the success region without collision but do not terminate successfully. SPL combines navigation success with path efficiency relative to the reference path length, assigning zero to unsuccessful episodes. All metrics are computed over the full evaluation set and reported as percentages.

Table 1: Overall and size-stratified performance on the UAV-ON benchmark.

Method	Small			Medium			Large			Total		
	SR↑	OSR↑	SPL↑	SR↑	OSR↑	SPL↑	SR↑	OSR↑	SPL↑	SR↑	OSR↑	SPL↑
Random	4.14	7.80	2.80	3.33	8.10	3.05	2.48	8.07	1.62	3.70	8.00	2.66
CLIP-H	2.86	8.43	1.51	10.95	16.67	7.17	13.04	19.25	10.53	6.20	11.90	4.15
AOA-F	4.45	16.38	1.61	10.48	17.62	6.36	14.29	21.74	10.66	7.30	17.50	4.06
AerialVLN	4.13	19.87	4.01	4.76	17.62	4.90†	1.86	19.25	1.86	3.90	19.30	3.72
NaVid	10.02	29.55	4.41	11.43	26.44	5.95	17.39	30.73	15.03	11.50	29.10	6.44
OpenFly	12.40	26.23	6.35	10.00	20.48	4.19	13.04	31.65	7.56	12.00	25.90	6.09
APEX	–	–	–	–	–	–	–	–	–	13.33	20.00	10.14
OctMem-Agent	18.91	29.57	6.03	18.09	26.66	6.27	23.60	31.67	7.80	19.50	29.30	6.37
VIREO	**22.44**	**39.56**	**13.67**	**29.01**	**37.04**	**21.44**	**26.45**	**33.88**	**20.92**	**24.56**	**38.06**	**16.59**

† Zhou et al. (2026) report an SPL of 4.90% for AerialVLN on medium targets, exceeding the corresponding SR of 4.76%. We retain the published value for traceability; this entry remains unverified and is excluded from comparative claims.

Implementation Details. We use Qwen3-VL-plus (Bai et al., 2025) with temperature zero. The offline collection contains 424 objects from the training environments, each observed from 12 viewpoints and paired with matching and non-matching descriptions. We allocate 298 objects to response-model fitting and 126 to policy calibration, keeping each object's views and queries in the same partition. The response model uses eight geometric features standardized on the fitting split. We set $\sigma_p = 12\,\mathrm{m}$, $\sigma_u = \pi/3$, and $K_{\mathrm{obs}} = K_{\mathrm{req}} = 12$. Complete parameter settings are listed in Table 5; feature definitions, viewpoint filters, and execution rules are provided in Appendix A.

4.2 Performance Comparison on UAV-ON

As shown in Table 1, VIREO achieves the highest reported SR, OSR, and SPL across small, medium, and large targets. Compared with OctMem-Agent and APEX, which guide spatial search through octree memory and spatial-semantic maps, respectively, VIREO uses its viewpoint information field to guide motion toward views expected to be informative for candidate recognition. Its viewpoint-aware evidence aggregation weights recognition evidence according to viewpoint novelty, reducing the contribution of repetitive observations to candidate belief updates. On the full test set, VIREO achieves relative gains of 25.95% in SR and 29.90% in OSR over OctMem-Agent, and a relative SPL gain of 63.61% over APEX. These results suggest that the proposed framework contributes to improved navigation success and efficiency in aerial ObjectNav.

Table 2: Performance of VIREO on seen and unseen objects.

Split	SR↑	OSR↑	SPL↑
Seen	**25.35**	35.92	**17.32**
Unseen	24.05	**39.42**	16.12
Total	24.56	38.06	16.59

Table 2 compares VIREO's navigation performance on seen and unseen target objects. VIREO maintains comparable success rates across the two groups, achieving 25.35% SR on seen objects and 24.05% on unseen objects. This small performance gap suggests that VIREO generalizes to target objects absent from the dataset's training split.

4.3 Recognition Analysis and Component Ablation

Viewpoint Sensitivity Analysis. We examine how VLM recognition responses vary across the 12 viewpoints of each object-query pair in the offline observation set. Table 3 summarizes these patterns separately for matching and non-matching pairs. Mixed denotes both positive and negative responses; All positive and All negative denote uniform responses, while Any positive denotes at least one positive response. These patterns suggest that seeking a positive response can lead to accepting an incorrect candidate, while abandoning a candidate after a negative response can discard a valid target. They motivate selecting viewpoints for their expected ability to distinguish matching from non-matching candidates and using accumulated evidence to inform candidate decisions.

Online Component Ablation. We evaluate the contributions of Viewpoint Planning (VP) and Evidence Aggregation (EA) through a 2×2 ablation. VP selects viewpoints using the viewpoint information field, while EA combines novelty-weighted belief updates with belief-based candidate acceptance and release. The variant w/o VP & EA uses the base-agent planning and candidate-decision mechanisms, recovering the Base Agent. The variant w/o EA enables VP and maintains a unit-weight internal belief for viewpoint scoring and determining when to return control to the base planner; candidate acceptance and release follow the base-agent rules. The variant w/o VP enables EA while retaining the base agent's local observation planner. Full VIREO uses the novelty-weighted belief for both viewpoint selection and candidate decisions.

All four variants are evaluated on the same 200-task UAV-ON subset, with equal representation across scenes and a target-size distribution of 120 small, 48 medium, and 32 large targets. They share the perception pipeline, global exploration mechanism, viewpoint candidate generation, low-level controller, and action budget.

Table 3: Viewpoint sensitivity.

Pair type	Responses	Ratio (%)
Matching	Mixed	54.72
	All positive	34.43
	All negative	10.85
Non-matching	Any positive	28.77

Table 4: Online component ablation.

Variant	SR↑	OSR↑	SPL↑
w/o VP & EA	13.50	30.50	6.89
w/o EA	22.00	36.00	15.12
w/o VP	19.00	33.50	13.59
VIREO	**25.50**	**37.00**	**17.41**

Table 4 shows that enabling VP improves both target-region reachability and task completion, supporting the benefit of guiding UAV movements according to the expected information provided by future observations. EA narrows the OSR–SR gap under both planning strategies, suggesting that accumulated recognition evidence helps translate reaching the target region into successful termination. Full VIREO achieves the highest SR, OSR, and SPL, supporting the complementary contributions of informative viewpoint selection and evidence-based candidate decisions to navigation performance.

4.4 Qualitative Navigation Examples

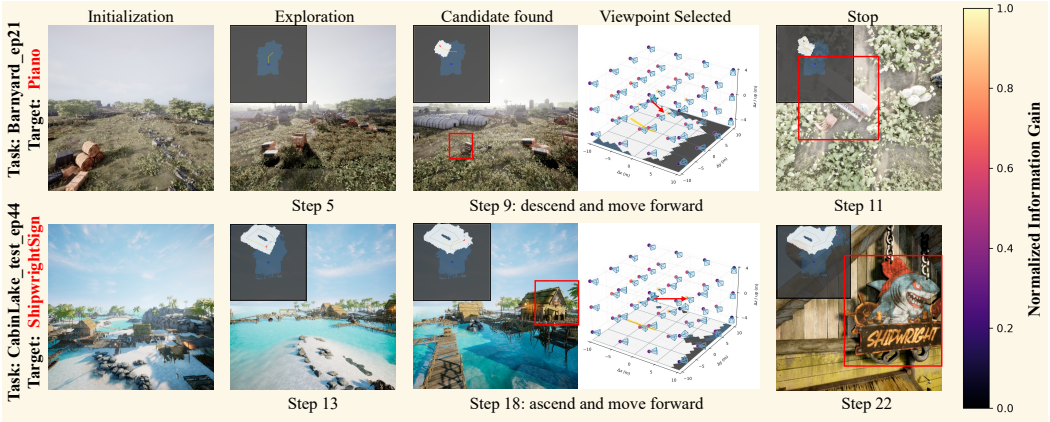

Figure 3: VIREO navigation for a piano in Barnyard (top) and a Shipwright sign in CabinLake (bottom). Images, maps, and viewpoint plots trace exploration, candidate discovery, viewpoint selection, and stopping. Red boxes mark candidates; lighter viewpoint colors indicate higher normalized information gain.

Fig. 3 illustrates viewpoint selection between candidate discovery and stopping. For the Barnyard piano, VIREO descends and moves forward at step 9 before stopping at step 11; for the CabinLake Shipwright sign, it ascends and moves forward at step 18 before stopping at step 22. The viewpoint fields show the information gain associated with these motion options. The field values depend on the current candidate belief and quantify how much each prospective observation is expected to

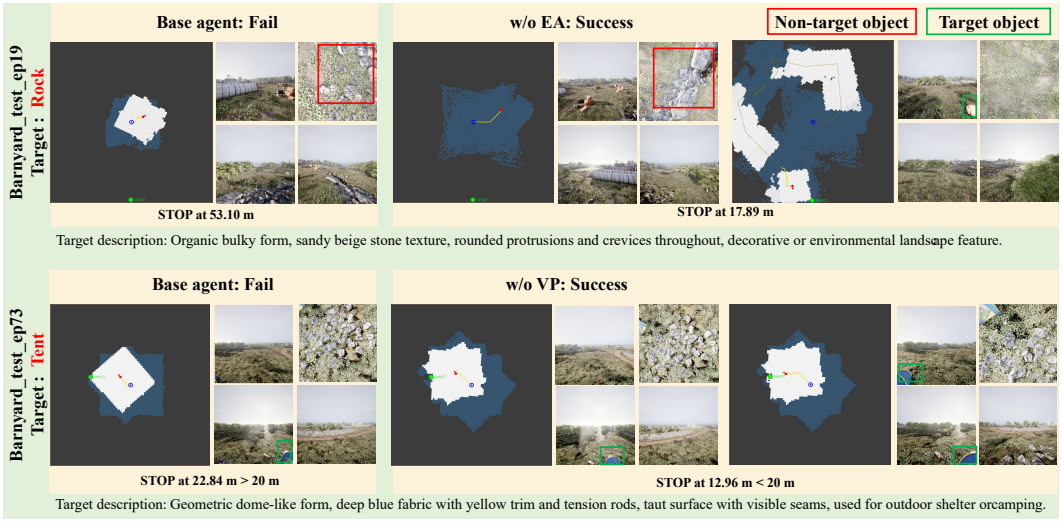

Figure 4: Qualitative ablations on Barnyard rock (top) and tent (bottom) tasks. The Base Agent fails in both cases; w/o EA succeeds on the rock task, and w/o VP succeeds on the tent task. Red and green boxes mark non-target and target objects, respectively. Distances are measured to the benchmark target locations. Ground-truth locations in the maps are shown for visualization only.

reduce uncertainty about the candidate's match status. After the selected motion is executed, the new recognition response contributes to the belief update that informs subsequent decisions.

Fig. 4 compares the Base Agent with variants using VP or EA individually. The VP-only variant retains the Base Agent's candidate-decision rules, while the EA-only variant retains its local observation planner. In the rock task, the Base Agent stops at a visually similar non-target location 53.10 m away, whereas w/o EA (VP only) continues searching and succeeds at 17.89 m. This illustrates how VP can help avoid premature acceptance based on category-level resemblance. In the tent task, both agents observe the target early, but the Base Agent stops outside the success radius at 22.84 m. With EA, w/o VP collects further evidence before stopping at 12.96 m. EA delays premature termination, allowing the base planner's continued approach to satisfy the spatial success criterion in this example.

5 Conclusion

This paper presents VIREO, a framework that couples spatial search with active viewpoint planning for aerial ObjectNav. Using viewpoint-conditioned recognition likelihoods, the viewpoint information field guides UAV motion toward viewpoints expected to reduce uncertainty about a candidate's match status. Viewpoint-aware evidence aggregation weights recognition evidence by geometric novelty, reducing the influence of repetitive observations on candidate beliefs and stopping decisions. Experiments on UAV-ON show improved navigation success and path efficiency over the compared baselines across target sizes, while ablations support the contributions of both components.

Limitations and Future Work. VIREO relies on VLM calls for perception and planning, and variability in model responses can lead to inconsistent navigation decisions. Future work could incorporate task-specific VLM training to improve the consistency of recognition and planning. For example, supervised fine-tuning on labeled recognition examples and expert navigation demonstrations could better align model outputs with task requirements. Consistency regularization could discourage contradictory recognition judgments for identical observations and semantically equivalent target descriptions. Reinforcement learning with navigation feedback could further align planning decisions with task success and motion efficiency.

AI USE STATEMENT

In this work, we used generative AI tools to assist with manuscript editing and language polishing, including improvements to grammar, clarity, readability, and phrasing. We reviewed and revised the AI-assisted text, checking its consistency with our methods, experimental results, and intended scientific claims. We take responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Peter Anderson, Angel Chang, Devendra Singh Chaplot, Alexey Dosovitskiy, Saurabh Gupta, Vladlen Koltun, Jana Kosecka, Jitendra Malik, Roozbeh Mottaghi, Manolis Savva, and Amir R. Zamir. On evaluation of embodied navigation agents. *arXiv preprint arXiv:1807.06757*, 2018. URL https://arxiv.org/abs/1807.06757.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025. URL https://arxiv.org/abs/2511.21631.

Pier Giovanni Bissiri, Chris C Holmes, and Stephen G Walker. A general framework for updating belief distributions. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 78 (5):1103–1130, 2016.

Devendra Singh Chaplot, Dhiraj Prakashchand Gandhi, Abhinav Gupta, and Russ R Salakhutdinov. Object goal navigation using goal-oriented semantic exploration. *Advances in Neural Information Processing Systems*, 33:4247–4258, 2020.

Jiaqi Chen, Bingqian Lin, Ran Xu, Zhenhua Chai, Xiaodan Liang, and Kwan-Yee Wong. 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)*, pp. 9796–9810. Association for Computational Linguistics, 2024.

Meng Chu, Zhedong Zheng, Wei Ji, Tingyu Wang, and Tat-Seng Chua. Towards natural language-guided drones: GeoText-1652 benchmark with spatial relation matching. In *Computer Vision – ECCV 2024*, pp. 213–231. Springer, 2024.

Ronghao Dang, Liuyi Wang, Zongtao He, Shuai Su, Jiagui Tang, Chengju Liu, and Qijun Chen. Search for or navigate to? dual adaptive thinking for object navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8250–8259, 2023.

Matt Deitke, Winson Han, Alvaro Herrasti, Aniruddha Kembhavi, Eric Kolve, Roozbeh Mottaghi, Jordi Salvador, Dustin Schwenk, Eli VanderBilt, Matthew Wallingford, Luca Weihs, Mark Yatskar, and Ali Farhadi. RoboTHOR: An open simulation-to-real embodied AI platform. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3164–3174, 2020.

Joachim Denzler and Christopher M. Brown. Information theoretic sensor data selection for active object recognition and state estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(2):145–157, 2002.

Tom Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.

Adam Foster, Desi R. Ivanova, Ilyas Malik, and Tom Rainforth. Deep adaptive design: Amortizing sequential Bayesian experimental design. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 3384–3395. PMLR, 2021.

Samir Yitzhak Gadre, Mitchell Wortsman, Gabriel Ilharco, Ludwig Schmidt, and Shuran Song. CoWs on pasture: Baselines and benchmarks for language-driven zero-shot object navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23171–23181, 2023.

Yunpeng Gao, Chenhui Li, Zhongrui You, Junli Liu, Zhen Li, 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*, 2026.

Qiao Gu, Alihusein Kuwajerwala, Sacha Morin, Krishna Murthy Jatavallabhula, Bipasha Sen, Aditya Agarwal, Corban Rivera, William Paul, Kirsty Ellis, Rama Chellappa, Chuang Gan, Celso Miguel de Melo, Joshua B. Tenenbaum, Antonio Torralba, Florian Shkurti, and Liam Paull. ConceptGraphs: Open-vocabulary 3D scene graphs for perception and planning. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5021–5028, 2024.

Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, Dinesh Manocha, and Tianyi Zhou. HallusionBench: An advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14375–14385, 2024.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1321–1330. PMLR, 2017.

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, 2023. doi: 10.1109/ICRA48891.2023.10160969.

Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Matt Deitke, Kiana Ehsani, Daniel Gordon, Yuke Zhu, Aniruddha Kembhavi, Abhinav Gupta, and Ali Farhadi. AI2-THOR: An interactive 3D environment for visual AI. *arXiv preprint arXiv:1712.05474*, 2017.

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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5912–5922, 2025.

Hongxin Li, Zeyu Wang, Xu Yang, Yuran Yang, Shuqi Mei, and Zhaoxiang Zhang. MemoNav: Working memory model for visual navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17913–17922, 2024.

Rui Liu, Wenguan Wang, and Yi Yang. Volumetric environment representation for vision-language navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16317–16328, 2024.

Shubo Liu, Hongsheng Zhang, Yuankai Qi, Peng Wang, Yanning Zhang, and Qi Wu. AerialVLN: Vision-and-language navigation for UAVs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15384–15394, 2023.

Joao Marcos Correia Marques, Albert J. Zhai, Shenlong Wang, and Kris Hauser. On the overconfidence problem in semantic 3d mapping. In *2024 IEEE International Conference on Robotics and Automation*, pp. 11095–11102, 2024.

Matthias Minderer, Josip Djolonga, Rob Romijnders, Frances Hubis, Xiaohua Zhai, Neil Houlsby, Dustin Tran, and Mario Lucic. Revisiting the calibration of modern neural networks. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Mohammad Naghshvar and Tara Javidi. Active sequential hypothesis testing. *The Annals of Statistics*, 41(6):2703–2738, 2013.

Tom Rainforth, Adam Foster, Desi R. Ivanova, and Freddie Bickford Smith. Modern Bayesian experimental design. *Statistical Science*, 39(1):100–114, 2024.

Santhosh Kumar Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alexander Clegg, John Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel Chang, Manolis Savva, Yili Zhao, and Dhruv Batra. Habitat-Matterport 3D Dataset (HM3D): 1000 large-scale 3D environments for embodied AI. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021.

Santhosh Kumar Ramakrishnan, Devendra Singh Chaplot, Ziad Al-Halah, Jitendra Malik, and Kristen Grauman. Poni: Potential functions for objectgoal navigation with interaction-free learning. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18868–18878. IEEE, 2022.

Ram Ramrakhya, Dhruv Batra, Erik Wijmans, and Abhishek Das. PIRLNav: Pretraining with imitation and RL finetuning for ObjectNav. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17896–17906, 2023.

Shouwei Ruan, Yinpeng Dong, Hanqing Liu, Yao Huang, Hang Su, and Xingxing Wei. Omniview-Tuning: Boosting viewpoint invariance of vision-language pre-training models. In *Computer Vision – ECCV 2024*, pp. 309–327. Springer, 2024.

Nur Muhammad Mahi Shafiullah, Chris Paxton, Lerrel Pinto, Soumith Chintala, and Arthur Szlam. CLIP-Fields: Weakly supervised semantic fields for robotic memory. In *Proceedings of Robotics: Science and Systems*, 2023.

Javier Velez, Garrett Hemann, Albert S Huang, Ingmar Posner, and Nicholas Roy. Modelling observation correlations for active exploration and robust object detection. *Journal of Artificial Intelligence Research*, 44:423–453, 2012.

Zihan Wang, Xiangyang Li, Jiahao Yang, Yeqi Liu, and Shuqiang Jiang. GridMM: Grid memory map for vision-and-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15625–15636, 2023.

Abdelrhman Werby, Chenguang Huang, Martin Büchner, Abhinav Valada, and Wolfram Burgard. Hierarchical open-vocabulary 3D scene graphs for language-grounded robot navigation. In *Proceedings of Robotics: Science and Systems*, 2024.

Pengying Wu, Yao Mu, Bingxian Wu, Yi Hou, Ji Ma, Shanghang Zhang, and Chang Liu. VoroNav: Voronoi-based zero-shot object navigation with large language model. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 53757–53775. PMLR, 2024.

Yi Wu, Yuxin Wu, Aviv Tamar, Stuart Russell, Georgia Gkioxari, and Yuandong Tian. Bayesian relational memory for semantic visual navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2769–2779, 2019.

Fei Xia, Amir R. Zamir, Zhiyang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. Gibson Env: Real-world perception for embodied agents. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 9068–9079, 2018.

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.

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

Bangguo Yu, Hamidreza Kasaei, and Ming Cao. L3MVN: Leveraging large language models for visual target navigation. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3554–3560, 2023.

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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15232–15242, 2026.

Jiazhao Zhang, Kunyu Wang, Rongtao Xu, Gengze Zhou, Yicong Hong, Xiaomeng Fang, Qi Wu, Zhizheng Zhang, and He Wang. NaVid: Video-based VLM plans the next step for vision-and-language navigation. In *Proceedings of Robotics: Science and Systems*, 2024a.

Lingfeng Zhang, Xiaoshuai Hao, Qinwen Xu, Qiang Zhang, Xinyao Zhang, Pengwei Wang, Jing Zhang, Zhongyuan Wang, Shanghang Zhang, and Renjing Xu. MapNav: A novel memory representation via annotated semantic maps for VLM-based vision-and-language navigation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13032–13056. Association for Computational Linguistics, 2025a.

Sixian Zhang, Xinyao Yu, Xinhang Song, Xiaohan Wang, and Shuqiang Jiang. Imagine before go: Self-supervised generative map for object goal navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16414–16425, 2024b.

Weichen Zhang, Chen Gao, Shiquan Yu, Ruiying Peng, Baining Zhao, Qian Zhang, Jinqiang Cui, Xinlei Chen, and Yong Li. 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)*, pp. 31292–31309, 2025b.

Jiacong Zhou, Jiaxu Miao, Yourun Lin, Xianyun Wang, Jun Xiao, and Jun Yu. Memory-augmented scene understanding and exploration for open-world aerial object-goal navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21616–21626, 2026.

APPENDIX

This appendix provides implementation details, additional validation, and diagnostic analyses, organized as follows:

- **Appendix A: Implementation and Experimental Details.** Evaluation settings, geometric features, exploration planning, response-model fitting, policy calibration, VLM interfaces and prompts, and qualitative navigation examples.
- **Appendix B: Candidate Belief Calibration.** Evaluation of final candidate beliefs in offline replay using Brier score, ECE, and ROC AUC, with a paired bootstrap analysis of calibration differences.
- **Appendix C: Offline Candidate Verification.** Held-out-scene comparisons of verification policies in terms of candidate acceptance, observation efficiency, and geometric movement cost.

A IMPLEMENTATION AND EXPERIMENTAL DETAILS

Repeated evaluation. We evaluate our method on the same test tasks for ten runs, the same as APEX (Zhang et al., 2026). For each metric M, we first compute the metric over the complete task set within each run and then report its arithmetic mean across runs:

$$\overline{M} = \frac{1}{10} \sum_{r=1}^{10} M^{(r)}. \tag{15}$$

Repeated executions of the same task are treated as repeated measurements rather than additional distinct benchmark tasks. Results reproduced from prior publications retain their original reporting protocols.

Evaluation time. A complete run with a single evaluation worker takes approximately 11.4 days. The ten-run evaluation was executed with four concurrent evaluation workers and took approximately 28.5 days of elapsed wall-clock time. The corresponding serial workload is approximately 114 worker-days. These timings describe the end-to-end evaluation workflow rather than isolated VLM inference time.

A.1 Geometric Features

We express each observation relative to a fixed camera reference frame A established for the candidate. Let $\mathbf{p}_0$ and $\mathbf{R}_0$ denote the reference camera position and orientation. The columns of $\mathbf{R}_0$ are its forward, right, and down axes expressed in world coordinates. For a camera at position $\mathbf{p}$ with unit viewing direction $\mathbf{f}$, we compute

$$\Delta\mathbf{p}^A = \mathbf{R}_0^\top(\mathbf{p} - \mathbf{p}_0), \qquad \mathbf{u}^A = \mathbf{R}_0^\top \mathbf{f}. \tag{16}$$

This representation uses camera poses without requiring the ground-truth target position.

Let $\Delta\mathbf{p}^A = (d_1, d_2, d_3)^\top$ and $\mathbf{u}^A = (u_1, u_2, u_3)^\top$. We compute

$$r_\perp = \sqrt{d_1^2 + d_2^2}, \qquad h = d_3, \qquad \alpha = \arccos(\mathrm{clip}(u_1, -1, 1)),$$

and

$$c_{\mathrm{align}} = \begin{cases} (\mathbf{u}^A)^\top \dfrac{-\Delta\mathbf{p}^A}{\|\Delta\mathbf{p}^A\|_2}, & \|\Delta\mathbf{p}^A\|_2 > \epsilon, \\ 1, & \|\Delta\mathbf{p}^A\|_2 \le \epsilon, \end{cases} \qquad \epsilon = 10^{-9}\,\mathrm{m}.$$

Here, $r_\perp$ measures displacement in the reference forward–right plane, h is displacement along the reference down axis, α is the angular deviation from the reference forward direction, and c_{align} measures alignment with the direction back to the reference position. The coordinate h is relative to the reference camera and does not represent height above the ground.

The raw feature vector is

$$\boldsymbol{\psi}(\delta) = [r_\perp, r_\perp^2, h, h^2, \alpha, \alpha^2, c_{\mathrm{align}}, c_{\mathrm{align}}^2]^\top.$$

Each component is standardized using the fitting partition:

$$\phi_j(\delta) = \frac{\psi_j(\delta) - \mu_j}{\sigma_{\psi,j}}, \qquad j = 1, \ldots, 8.$$

The fitted means μ_j and standard deviations $\sigma_{\psi,j}$ remain fixed during navigation. At the reference position, the convention $c_{\mathrm{align}} = 1$ also applies to subsequent observations with zero displacement.

A.2 Exploration Planner

When no current target-specific cue is available and no unresolved candidate requires investigation, VIREO uses coverage-driven exploration to propose additional observation viewpoints.

The planner maintains a spatial search-completion map $c_t(x) \in [0, 1]$ over two-dimensional map cells. For a candidate viewpoint v, let $\mathcal{F}_t(v)$ denote its estimated sparse observation footprint, and let $\hat{q}_t(x \mid v) \in [0, 1]$ denote the predicted observation quality for cell x. When multiple cameras cover the same cell, their contributions are combined by retaining the maximum predicted quality. The normalized residual coverage gain is

$$\bar{G}_t(v) = \frac{1}{|\mathcal{F}_t(v)|} \sum_{x \in \mathcal{F}_t(v)} \big(1 - c_t(x)\big)\hat{q}_t(x \mid v), \tag{17}$$

with $\bar{G}_t(v) = 0$ for an empty footprint. This quantity is a heuristic estimate of additional effective search coverage, rather than a calibrated probability of finding the target.

Exploration options are ranked using

$$U_t^{\mathrm{exp}}(v) = \lambda_G \bar{G}_t(v) - C_t(v), \tag{18}$$

where $C_t(v)$ combines normalized horizontal, vertical, and yaw motion costs, an estimated risk penalty, and a revisit penalty. Among the enabled options compatible with the base plan's height action, the preferred option is

$$v_t^\star \in \arg\max_{v \in \mathcal{C}_t^{\exp}} U_t^{\exp}(v). \tag{19}$$

The preferred option replaces the base exploration plan only when the exploration eligibility checks are satisfied and its utility improvement exceeds a prescribed margin; otherwise, the base plan is retained. The selected goal is passed to the low-level executor for motion execution and obstacle avoidance. Search completion is updated using observations associated with the actual reached poses.

Exploration coverage and candidate verification serve different purposes. Coverage guides where to search when no candidate requires investigation, whereas the viewpoint information field guides the inspection of an active candidate. Candidate-directed inspection is not determined solely by exploration coverage, and temporary loss of visual support may trigger reinspection before returning to global exploration.

A.3 Response-Model Fitting and Policy Calibration

The offline collection contains 424 physical objects from the training environments, with 12 viewpoints per object. Each object is paired with one matching and one non-matching target description, yielding 848 candidate sequences and 10,176 cached binary responses. The paired queries share the viewpoint images. We allocate 298 objects to response-model fitting and 126 objects to policy calibration, keeping all viewpoints and both queries of each object in the same partition. The resulting partitions contain 7,152 and 3,024 responses, respectively.

For each hypothesis $\xi \in \{0, 1\}$, we fit a separate logistic response model

$$r_\xi(\delta) = \sigma\big(\mathbf{w}_\xi^\top \boldsymbol{\phi}(\delta) + b_\xi\big), \tag{20}$$

using binary cross-entropy with an L_2 penalty $\frac{\lambda}{2}\|\mathbf{w}_\xi\|_2^2$, where $\lambda = 10^{-4}$. The prediction target is the VLM response y, while ξ specifies whether the object matches the query. The VLM remains frozen.

Policy calibration searches observation budgets $\{4, 6, 8, 12\}$, position scales $\{1, 2, 4, 6, 8, 12\}$ m, direction scales $\{10, 20, 30, 45, 60\}^\circ$, acceptance thresholds $\{0.6, 0.7, 0.8, 0.9, 0.95\}$, and rejection thresholds $\{0.05, 0.1, 0.15, 0.2, 0.3, 0.4\}$. The selection criterion prioritizes true-candidate acceptance subject to an empirical false-acceptance rate of at most 5% on the calibration partition. The selected settings are summarized in Table 5. The initial belief is $\rho_{\text{init}} = 0.5$, reflecting the balanced offline candidate set rather than an estimated online proposal frequency.

This final fitting/calibration split is distinct from the five-fold held-out-scene evaluation in Appendix C. Each outer fold fits its own response models and calibrates its policies using the designated fitting and calibration scenes. Its selected policy parameters need not equal the final settings in Table 5.

A.4 VLM Interfaces and Prompts

Qwen3-VL-plus is used with temperature zero. The four online ablation groups share the recognition interfaces, base-agent prompt templates, camera-selection rules, and call-eligibility rules. Their trajectories, dynamic prompt context, and realized numbers of calls may differ.

Only the single-image recognition call supplies the binary response y used by the calibrated response model and memory. It reuses the offline collection template reproduced below. Multi-view Describe+Match supports scene interpretation, candidate proposals, and reference-camera selection. Waypoint planning proposes base-agent actions, while candidate comparison supports association and recovery using a historical reference. None of these three calls supplies an additional calibrated response or substitutes for an invalid single-image recognition result.

Response-model evaluation, viewpoint information scores, likelihood ratios, geometric weights, and belief updates are computed in code without additional VLM calls. A high-level planning step, a recognition response, a VLM call, and a physical action are therefore distinct counting units.

Table 5: Implementation settings.

Setting	Value
VLM	Qwen3-VL-plus
Decoding temperature	0
Response feature dimension	8
L_2 coefficient λ	10^{-4}
Initial candidate belief ρ_{init}	0.5
Acceptance / rejection thresholds	0.95 / 0.05
Position kernel scale σ_p	12 m
Direction kernel scale σ_u	$\pi/3$ rad
Viewpoint similarity exclusion threshold	0.90
Minimum viewpoint information score	0.01 nats
Maximum absolute standardized feature	3
Consecutive requests with the same expected camera	At most 3
Valid-response budget K_{obs}	12 per candidate
Viewpoint-request budget K_{req}	12 per candidate
Physical-action budget	150 per episode
Physical-action limit per waypoint	10

Shared single-image recognition prompt. Both offline collection and online single-image recognition use one RGB image followed by the text below, with no additional system prompt, historical images, maps, conversation history, or current candidate belief. We use `qwen3-vl-plus` with temperature 0.0. The target name and simulator actor identifier are withheld.

```
Determine whether the requested target object is visible in
this image.  Weak but object-specific evidence may count as
a match; generic scene similarity does not.  Return JSON
only with target_match (boolean), confidence (0 to 1), and
reason (short string).
Target name:  (withheld)
Target description:  {target_description}
Target size:  {target_size}
Camera view:  {camera_label}
```

Here, {`target_description`} is the target description from the dataset. The field {`target_size`} preserves the original dataset size string, such as `mid(3.75*2=7.5 squares)`, rather than reducing it to a categorical label. The field {`camera_label`} identifies the camera used to capture the image, for example `front`. The following JSON illustrates the requested output format; its values are examples rather than experimental results:

```
{
  "target_match": true,
  "confidence": 0.8,
  "reason": "Short explanation of the visual evidence."
}
```

For a parsed response with a native JSON boolean `target_match`, the binary response label used for fitting and online evidence updates is

$$y = \text{int}\big(\texttt{parsed["target_match"]}\big).$$

The `confidence` and `reason` fields are stored, but neither determines this binary label. In particular, no confidence threshold is applied to construct y. The response label y records the VLM judgment and is distinct from the candidate match label ξ.

Camera selection and response handling. The camera is selected before the single-image recognition request, and its image is paired with the actual camera pose at acquisition. The system does not query several single images and then select a positive or high-confidence response. A multi-view proposal may initiate base-agent investigation; activation of the calibrated candidate requires a valid

positive single-image response from the selected camera. That response receives unit weight as the first evidence update.

For an active calibrated candidate, each eligible high-level observation step issues at most one single-image recognition request. Request failures, parsing failures, non-boolean `target_match` values, and unavailable images or poses produce no evidence update. Planner cues and multi-view visibility judgments do not replace a missing response. A valid negative response remains negative evidence, including when it reflects insufficient visibility or detail. Candidate comparison does not impose an additional `same_candidate=yes` gate on evidence updates.

Shared multi-view Describe+Match prompts. This call uses the four current images labeled FRONT, LEFT, RIGHT, and DOWN for a joint scene-level assessment. Its visibility output supports candidate proposals rather than serving as y or an accumulated candidate belief. The system prompt is:

```
You are a UAV visual observation assistant.  Describe the
supplied current camera images and assess whether they
contain an object matching the requested target.  Use only
these images and the supplied target description.  Output
exactly one valid JSON object.
```

The user prompt is:

```
The current images are individually labeled FRONT, LEFT,
RIGHT, and DOWN.
Read each image according to its label.

Target description:  {target_description}
Target size:  {target_size}

Describe the objects, shapes, colors, materials, and
approximate apparent sizes visible in each image.

Set target_visible=true when an object reasonably matches
the target category, appearance, and supplied size
information.  Set it false when no such match is supported
by the current images.  A partial target-like cue may still
be worth investigating even when target_visible=false.

Do not infer an object's physical dimensions directly from
its pixel size without supporting scale information.
Generic scene similarity is insufficient to establish a
target match.
Do not invent fine details that cannot be resolved in the
image.

When target_visible=true, set best_view to the image
containing the clearest supporting evidence.
When target_visible=false, set best_view to null.
Use matched_features to describe the supporting evidence,
the unresolved details, or the specific mismatch.

This is a joint assessment of the current images.  It is not
a navigation-success decision or an accumulated candidate
belief.

Return a JSON object with exactly these fields:
- FRONT_desc:  string
- LEFT_desc:  string
- RIGHT_desc:  string
- DOWN_desc:  string
- target_visible:  boolean
- best_view:  "FRONT", "LEFT", "RIGHT", "DOWN", or null
- confidence:  number in [0, 1]
- matched_features:  string
```

Shared base-agent planning prompts. The planner receives the current observations, available auxiliary maps, enabled actions, shared search history, and current multi-view summary. Its output is a proposal subject to the active viewpoint-selection and candidate-decision modules. The system prompt is:

```
You are the shared base planner of a UAV Object Navigation
agent.
Propose the next horizontal waypoint, altitude change, and
final viewing direction using the supplied observations,
search memory, and enabled options.
You are not the low-level flight controller.
Do not output primitive actions, velocities, or arbitrary
coordinates.
The controller may replace your local viewpoint proposal or
handle candidate acceptance and release separately.
Output exactly one valid JSON object.
```

The user prompt is:

```
TASK
Find the object described below.
Target description: {target_description}
Target size: {target_size}
The target's ground-truth position and simulator instance
identifier are not provided.

CURRENT STATE
- Planning step: {step_num}
- Position relative to start: dx={pos_dx} m, dy={pos_dy} m
- Absolute NED z magnitude: {altitude_m} m; this is not
height above local ground
- Local height and clearance information:
{local_height_clearance_summary}
- Search-area boundary information: {boundary_summary}
- Recent waypoint, height, and heading choices:
{recent_motion_summary}
- Exploration coverage: {waypoint_exploration_summary}

HISTORICAL SEARCH CONTEXT
{agent_memory_section}
{base_candidate_state_section}

Historical context records previous observations and search
hypotheses.
It may guide where to investigate, but it does not establish
current visibility.

CURRENT VISUAL INPUTS
{available_image_description}

FRONT, LEFT, RIGHT, and DOWN, when listed above, are current
onboard RGB views.
Use these images for the current target_visible and
target_cue_visible judgments.
Auxiliary maps provide planning context; do not use a map
alone to assert a target match or successful arrival.

CURRENT MULTI-VIEW RECOGNITION SUMMARY
{current_multiview_response_summary}

This summary describes the current image group.
```

```
A negative match does not prove that the whole area contains
no target or that a partially visible object is irrelevant.
When you disagree with the summary, identify specific
evidence in a current onboard image.
Your judgment is a planning cue and does not overwrite the
controller's recognition label.

AVAILABLE ACTION OPTIONS
Horizontal waypoints:
{enabled_waypoint_list}

Height options:
{height_actions_text}

Final-heading options:
{yaw_actions_text}

PLANNING PRIORITIES
1.  Respect enabled actions, search boundaries, and the
supplied safety constraints.
2.  Investigate a current object-specific target cue or an
active unresolved candidate.
3.  When no such investigation is warranted, explore areas
with useful unobserved coverage.

CURRENT-VIEW JUDGMENTS
- target_cue_visible=true requires an object-specific cue in
a current onboard image.
- target_visible=true requires a reasonably supported match
to the target description in a current onboard image.
- A partial cue may justify investigation without a
confirmed match.
- Historical belief, a previous sighting, and a high
exploration ratio do not determine either current-view
field.
- State unresolved attributes or details in reason instead
of inventing them.

VIEWPOINT PROPOSALS
- Propose an enabled waypoint and viewing direction that may
improve the observation of a relevant candidate.
- Consider framing, occlusion, visible detail, and
approximate scale jointly.
- A closer location is not automatically a more informative
view.
- Revisiting an observed region is allowed when a current
cue or unresolved candidate motivates it.
- Exploration coverage helps prioritize search; it is not
evidence that objects in a region are non-targets.
- Avoid repeated requests with no expected change in
observation.
- W4 may be used for a useful heading or height change, or
for a base-policy stop proposal.

PLAN MODES
- explore:  seek new coverage when there is no candidate
requiring investigation.
- approach:  move to investigate an object-specific current
cue.
- verify:  obtain a better observation of an unresolved
candidate.
```

```
- stop: propose termination to the base decision mechanism;
this does not bypass the controller's candidate-decision
policy.

For a stop proposal, use waypoint=W4, height_action=KEEP,
and yaw_action=KEEP.
Do not infer that the UAV is within a ground-truth success
radius from visual confidence alone.

OUTPUT
Return exactly these fields:
- waypoint: one enabled waypoint identifier
- height_action: one enabled height action
- yaw_action: one enabled heading action
- plan_mode: "explore", "approach", "verify", or "stop"
- target_cue_visible: boolean
- target_visible: boolean
- best_view: "FRONT", "LEFT", "RIGHT", "DOWN", or null
- confidence: number in [0, 1]
- reason: a short explanation grounded in current evidence
and the chosen planning objective
```

The current multi-view summary is inserted using the following format. It contains only the valid current-step result; an unavailable result is marked `unavailable` rather than converted to a negative visibility judgment.

```
Source: current-step multi-view Describe+Match
target_visible: {dm_target_visible}
best_view: {dm_best_view}
confidence: {dm_confidence}
matched_features: {dm_matched_features}
```

Shared candidate-comparison prompts. The first image is a historical reference, followed by the available current camera images with explicit labels. This call supports base-agent candidate association and qualitative visual proximity comparison. It produces neither an additional evidence sample nor a candidate acceptance or rejection decision. The system prompt is:

```
You are the shared visual candidate-comparison module of a
UAV Object Navigation agent.

You receive one historical image labeled
BEST_REFERENCE_<VIEW> and up to four current images labeled
CURRENT_FRONT, CURRENT_LEFT, CURRENT_RIGHT, and CURRENT_DOWN.
The historical image is a reference for a candidate, not
proof that the candidate matches the task or that the
reference pose is a successful stopping location.

First, judge whether a current image contains the same
physical candidate shown in the reference.
Use object-specific appearance and structure. Allow changes
in viewpoint, camera, scale, and partial occlusion.
Background similarity alone is insufficient.
Return uncertain when the available images do not support a
reliable identity comparison.

If same_candidate=yes, compare the apparent proximity of the
current and reference observations.
Consider the candidate's apparent scale together with
perspective, framing, and occlusion.
Sharper detail or better framing alone does not establish a
smaller physical distance.
```

```
Use current_closer or best_closer only when there is clear
visual support; otherwise use same or uncertain.
This is a qualitative visual comparison, not a metric
distance estimate.

Use only the supplied images and target description.
Do not infer hidden target coordinates or task success from
the comparison.
Do not output a stop, acceptance, or rejection decision.

Return exactly one JSON object with these fields:
- same_candidate: "yes", "no", or "uncertain"
- relative_proximity: "current_closer", "best_closer",
"same", or "uncertain"
- current_best_view: "FRONT", "LEFT", "RIGHT", "DOWN", or an
empty string
- candidate_grid_cell: a two-integer array [row, col], with
each integer in {0, 1, 2}, or null
- apparent_scale: "tiny", "small", "medium", "large",
"frame_filling", or "unknown"
- confidence: number in [0, 1]
- reason: a short explanation of the visual evidence

When same_candidate is no or uncertain, set
relative_proximity to uncertain.
When no current image supports a candidate correspondence,
use an empty current_best_view and a null
candidate_grid_cell.
```

The user prompt is:

```
Target description: {target_description}
Target size: {target_size}
Identity reference: step S{best_step}, view {best_view}
Current observation: step S{current_step}

Compare the candidate in the historical reference with the
objects in the labeled current images.
The candidate may appear in a different camera after UAV
motion.
Follow the required output schema.
```

Module authority across ablations. The Base Agent uses the shared planning and candidate-management interfaces without the added calibrated memory. Base Agent + VP maintains a unit-weight internal belief for information scoring and for leaving VP control; candidate decisions remain with the base agent. Base Agent + EA retains base-agent viewpoint proposals and uses geometrically weighted evidence for candidate decisions. VIREO combines VP selection with EA decisions.

While EA is active, a planner stop proposal cannot bypass its candidate-decision rule. Falling back from VP preserves EA authority for the active candidate. Collision, total-budget exhaustion, and execution failures remain separate task-termination conditions. The recognition template and interface rules are shared across all four groups, including the Base Agent.

A.5 Qualitative Navigation Examples

Fig. 5–7 present three additional navigation examples that complement Section 4.4. Each example combines a top-down trajectory visualization with selected onboard observations and the target description. The examples cover a playground in CityStreet, a satellite dish in WesternTown, and a road barrier in WinterTown, illustrating the different scene layouts and viewing conditions encountered during aerial target search.



CityStreet_test_ep37
Target : Playground
Description: Target visual description: Multi-level structure with yellow framework, red tube slide, rope climbing net, and ladder, designed for children's recreational play.

Figure 5: Navigation toward the playground target in CityStreet, episode 37. Left: top-down trajectory. Right: selected onboard observations. The target description appears beneath the image panels.



WesternTown_test_ep2
Target : SatDish
Description: Concave circular shape, light gray metallic surface, mounted support arm with central feed horn, used for receiving satellite signals.

Figure 6: Navigation toward the satellite-dish target in WesternTown, episode 2. The top-down trajectory and selected onboard views show the approach through the surrounding buildings and terrain; the target description appears beneath the image panels.

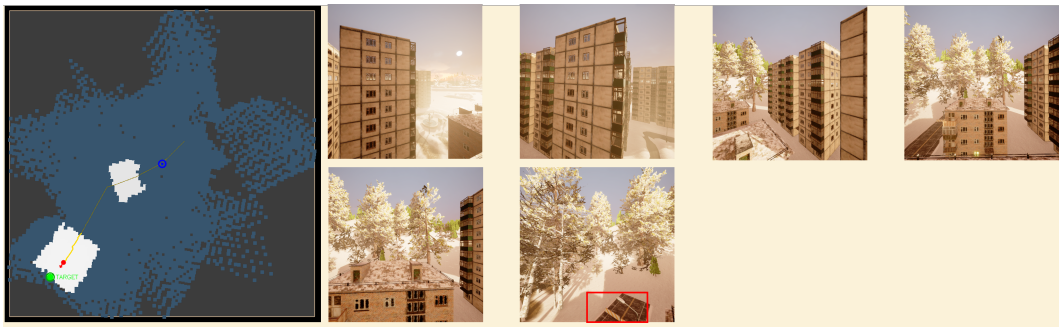

Figure 7: Navigation toward the road-barrier target in WinterTown, episode 7. The top-down trajectory and selected onboard views show the viewing conditions around buildings and trees; the target description appears beneath the image panels.

B CANDIDATE BELIEF CALIBRATION

Evaluation Protocol. We evaluate whether the final candidate belief agrees with the empirical frequency of matching candidates. The prediction target is the candidate label ξ, and the predicted probability is the final belief ρ.

We use the held-out-scene offline replay results for 848 candidate sequences, comprising 424 matching and 424 non-matching queries. Each sequence contributes its final belief once, including sequences that remain undecided. Intermediate beliefs from the same sequence are not treated as separate evaluation samples. The evaluated policies share the cached response pool, while their observation selection, calibrated parameters, and stopping times may differ. This analysis therefore compares the final beliefs produced by complete verification policies.

We report Brier score, ECE (Guo et al., 2017; Minderer et al., 2021), and ROC AUC (Fawcett, 2006) using ρ as the predicted probability and ξ as the binary label. ECE uses ten equal-width probability bins. These metrics assess candidate identity beliefs on the balanced offline query set; the labels do not indicate whether the UAV is within the 20 m navigation success radius.

Overall Calibration. Table 6 reports the probability quality of the final candidate beliefs. The full offline policy achieves the lowest Brier score and ECE, and the highest AUC, among the four evaluated policies. Compared with the belief-only baseline, its Brier score decreases from 0.14285 to 0.13368, ECE decreases from 0.11830 to 0.08373, and AUC increases from 0.83762 to 0.86516. The corresponding point estimates also improve over the information-only policy.

Table 6: Final candidate-belief quality in offline replay.

Offline policy	Brier↓	ECE↓	AUC↑
Belief-only	0.14285	0.11830	0.83762
Fixed order	0.14288	0.09116	0.82642
Information-only	0.14230	0.10278	0.83414
Full (offline)	**0.13368**	**0.08373**	**0.86516**

Paired Comparison. We quantify the ECE difference between the full offline policy and the belief-only baseline using 5,000 paired bootstrap resamples at the physical-object level. Each sampled object contributes both its matching and non-matching queries, and the same sampled objects are used for both policies. ECE is recomputed in each resample using ten fixed equal-width probability bins.

The ECE difference, defined as full minus belief-only, is -0.03457, with a 95% bootstrap interval of $[-0.04514, -0.01455]$. This interval supports improved aggregate calibration of the complete offline policy relative to this baseline. It is a nominal interval without correction for multiple comparisons and is conditional on the fitted models, selected parameters, and collected scenes. The interval applies to this ECE comparison; the other differences in Table 6 are reported as point estimates.

C OFFLINE CANDIDATE VERIFICATION

C.1 EVALUATION PROTOCOL

We evaluate candidate verification through offline replay of cached responses from 424 physical objects, each observed from 12 viewpoints with matching and non-matching queries, yielding 848 candidate sequences. Evaluation uses five outer scene folds, each comprising six scenes for response-model fitting, two for policy calibration, and two for testing. All views and both queries of each object remain in the same split.

C.2 POLICIES AND METRICS

Table 7 compares six policies on the same cached responses, with each policy calibrated separately on the calibration scenes. TAR and FAR denote the acceptance rates of true and false candidates,

respectively; Views measures the mean number of observations per candidate, and Distance is a geometric motion proxy between queried viewpoints.

C.3 RESULT

Table 7: Offline candidate verification on held-out scenes.

Offline policy	TAR (%)	FAR (%)	Views	Distance (m)
Belief-only	55.19	5.42	2.442	7.65
Fixed order	54.25	5.19	1.945	4.78
Random	57.19	4.82	3.012	20.84
Geometry-diverse	54.95	6.13	2.097	13.91
Information-only	53.30	4.01	2.088	5.82
Full (offline)	57.55	4.72	2.156	10.04

Compared with random selection, it requires 28.42% fewer observations, with a paired difference of -0.856 (95% object-grouped bootstrap interval: $[-0.960, -0.746]$), while TAR and FAR point estimates remain close. This result supports improved observation efficiency under the offline replay protocol. Compared with information-only selection, however, the full policy achieves higher TAR at a greater geometric motion proxy distance, revealing a trade-off between candidate acceptance and movement cost.