ADAFLY: ADAPTIVE TEMPORAL MEMORY FOR ONBOARD AERIAL VISION-LANGUAGE NAVIGATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Many aerial VLN systems rely on large multimodal models, making fully onboard deployment difficult on resource-constrained UAVs. Therefore, we propose **AdaFly**, a lightweight onboard aerial VLN approach that combines a compact 545M-parameter policy with adaptive temporal perception. Our design is motivated by a controlled analysis of 490,090 AeroBrain decisions, which reveals that the value of additional visual history is strongly stage-dependent. Extending history from two to four observations reduces action prediction error by 12.29% near upcoming turns and stops, but increases it by 2.72% during non-critical motion. AdaFly exploits this asymmetry with two components. An *Adaptive Temporal History Selector* determines whether additional older observations should be encoded before action-chunk generation, while a *Context-Conditioned Temporal Memory Compressor* condenses the selected history into a fixed set of task-relevant memory tokens. On AeroBrain, AdaFly achieves a 44.94% navigation success rate, representing a **69.0% relative improvement** over a representative high-performing aerial VLN baseline despite using a substantially smaller model. The proposed memory compressor reduces the four-frame historical representation from 144 visual tokens to only **16** memory tokens, an **88.9% reduction**, while improving navigation performance over uncompressed history. We further deploy the complete system fully onboard a DJI Matrice 4TD with a Manifold 3 (Jetson Orin NX) and validate it across 40 real-world flight trajectories without offboard policy inference.

1 INTRODUCTION

Recent advances in vision-language navigation (VLN) have enabled embodied agents to follow natural-language instructions in increasingly complex environments Liu et al. (2023); Wang et al. (2025); Gao et al. (2026); Zhang et al. (2025d). Extending these capabilities to UAVs is particularly challenging: aerial agents must reason over large-scale 3D scenes, generate flight actions, and operate under strict onboard computational constraints. Many recent aerial VLN systems rely on increasingly capable multimodal backbones to improve perception and reasoning, but their multi-billion-parameter scale and associated inference cost make fully onboard deployment difficult. Recent efforts have therefore begun to explore more efficient or onboard navigation systems Xu et al. (2026b); Zheng et al. (2026). Nevertheless, achieving strong navigation performance under the limited computation available on onboard UAV hardware remains a fundamental challenge.

A straightforward route toward onboard deployment is to reduce the scale of the navigation policy. However, aggressively shrinking the model also limits its representation and reasoning capacity, making it difficult to preserve the navigation performance of larger systems. Rather than recovering capability by increasing the backbone size again, a complementary direction is to exploit temporal context. Historical observations provide additional geometric, motion, and task-relevant cues beyond a single image, and have been widely used through historical frames, keyframes, structured memories, and compressed temporal representations Zhang et al. (2025a;b); He et al. (2025); Jiang et al. (2025). Temporal information is therefore particularly attractive for a lightweight policy: it can enrich the observation available to the model without requiring a substantially larger backbone.

This benefit, however, introduces a new efficiency bottleneck. Processing more historical observations requires additional visual encoding, while the resulting visual tokens enlarge the multimodal

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.

context processed by the policy. Using a long, fixed history at every decision can therefore offset the computational advantage of a lightweight backbone. More importantly, our analysis shows that the benefit of additional history is itself highly decision-dependent. Across 490,090 paired decisions on AeroBrain, extending history from two to four observations substantially improves action prediction near upcoming turns and stops, but degrades prediction during non-critical motion. This suggests that lightweight aerial VLN should not simply use *more* temporal context, but should allocate temporal computation according to the current navigation demand.

Motivated by this observation, we propose **AdaFly**, a lightweight aerial VLN approach that uses temporal information adaptively at both the image and token levels. An *Adaptive Temporal History Selector* (ATHS) estimates the current maneuver demand and determines whether additional historical observations should be encoded. A *Context-Conditioned Temporal Memory Compressor* (CTMC) then compresses the selected history into a fixed set of memory tokens conditioned on the current observation, navigation instruction, and UAV state. ATHS therefore controls the image-level temporal budget, while CTMC bounds the token-level cost of incorporating the selected history. The resulting 545M-parameter policy achieves strong navigation performance on AeroBrain and can be executed entirely onboard a DJI Matrice 4TD equipped with a Manifold 3 (Jetson Orin NX), without offboard policy inference.

Overall, our contributions can be summarized as follows:

- We propose **AdaFly**, a lightweight aerial VLN approach for fully onboard navigation, achieving approximately **270 ms** action-chunk inference on a 100-TOPS Jetson Orin NX without offboard computation.
- Based on our finding that visual history has decision-dependent utility, we introduce an *Adaptive Temporal History Selector* (ATHS), which reduces the average history image budget by **29.5%** through adaptive history allocation.
- We develop a *Context-Conditioned Temporal Memory Compressor* (CTMC) that compresses the selected visual history from **144** to a fixed set of **16** task-relevant memory tokens conditioned on the current navigation context.

2 Related Work

2.1 Vision-Language Navigation for UAVs

Vision-language navigation has recently expanded from ground-based agents to UAVs in large-scale 3D environments. AerialVLN Liu et al. (2023) established an early benchmark for language-guided aerial navigation, while OpenUAV Wang et al. (2025) introduced more realistic continuous-control settings. Subsequent work explored hierarchical semantic planning and global memory Zhang et al. (2025d), minimalist end-to-end control Xu et al. (2026a), and prediction-driven world-action modeling Zhao et al. (2026). OpenFly Gao et al. (2026) further provides a large-scale benchmark across diverse aerial scenes and rendering domains. While these advances improve the scale, realism, and reasoning capability of aerial VLN, computational cost has received comparatively less attention. As aerial VLN moves toward physical deployment, efficient perception and policy inference become increasingly important for resource-constrained onboard platforms.

2.2 Lightweight and Onboard Aerial Navigation

The growing capability of aerial VLN systems is often accompanied by substantial model and inference cost, motivating more efficient navigation methods. AeroDPO Xu et al. (2026b) studies the trade-off between model scale and visual fidelity, showing that compact models with high-resolution perception can remain competitive with larger alternatives. OnFly Zheng et al. (2026) targets fully onboard aerial VLN through shared perception and hardware-aware optimization, while VLN on the Fly Tayar et al. (2026) develops a modular stack separating language grounding, planning, and control. These efforts improve deployability through model, architecture, and system-level optimization. However, lightweight policies may still repeatedly encode and process multiple historical observations, leaving temporal computation as an important source of onboard cost.

2.3 Efficient Temporal Memory for Vision-Language Navigation

Historical observations are essential for long-horizon navigation, but long visual histories introduce computational and representational overhead. Memory-Adaptive VLN He et al. (2024) filters detrimental history, while Mem2Ego Zhang et al. (2025c) retrieves task-relevant cues from global memory. Uni-NaVid Zhang et al. (2025a) merges redundant visual tokens, whereas MapNav Zhang et al. (2025b) and Mem4Nav He et al. (2025) organize history using structured spatial memories. In aerial VLN, OpenFly Gao et al. (2026) uses historical keyframes, LongFly Jiang et al. (2025) compresses multi-view history into fixed-length representations, and HETT Ding et al. (2026) aggregates historical features into structured grid memory. These methods improve how navigation history is selected, stored, retrieved, or represented, but the decision-dependent utility of processing *additional* history for lightweight onboard computation remains less explored.

3 Method

3.1 Problem Formulation and Policy Overview

We consider aerial vision-language navigation conditioned on a natural-language instruction ℓ. At each replanning step t, the policy receives the current RGB observation $\mathbf{I}_t$, a set of recent historical observations $\mathcal{H}_t$, and the current UAV state $\mathbf{s}_t$. The policy predicts a chunk of K future actions,

$$\pi_\theta(\mathbf{I}_t, \mathcal{H}_t, \ell, \mathbf{s}_t) \rightarrow \mathbf{A}_t = [\mathbf{a}_t, \ldots, \mathbf{a}_{t+K-1}], \tag{1}$$

where each action contains a three-dimensional local displacement and a stop signal. This formulation allows the policy to combine the current observation, recent temporal context, and short-horizon action prediction.

For onboard deployment, computational efficiency becomes a critical consideration. Even with a lightweight policy such as SmolVLA Shukor et al. (2025), repeatedly processing a fixed visual history still introduces substantial temporal overhead. Increasing the history length requires additional image encoding, while retaining the resulting visual tokens enlarges the multimodal context processed during action generation. To address these limitations, we propose **AdaFly**, a lightweight aerial VLN approach with adaptive temporal memory. AdaFly operates at two levels: an *Adaptive Temporal History Selector* dynamically allocates the history-image budget, while a *Context-Conditioned Temporal Memory Compressor* transforms the selected history into a fixed-size memory representation.

3.2 Adaptive Temporal History Selection

The utility of visual history varies across a navigation trajectory. Richer temporal context is particularly useful when the UAV is approaching maneuver-critical stages such as turns or stops, whereas processing the same extended history during regular motion introduces unnecessary computation. Therefore, we introduce an *Adaptive Temporal History Selector* (ATHS) to dynamically select between short and extended history budgets.

At each replanning step, ATHS first takes the current image $\mathbf{I}_t$ together with the two most recent historical images $\mathbf{I}_{t-2}$ and $\mathbf{I}_{t-1}$. These images are encoded by the shared vision encoder, and their visual tokens are mean-pooled into frame-level features $\mathbf{v}_{t-2}$, $\mathbf{v}_{t-1}$, and $\mathbf{v}_t$. Based on these frame-level features, ATHS constructs two types of visual cues. The pooled features themselves provide the recent visual appearance, while their consecutive differences capture short-term visual change:

$$\Delta\mathbf{v}_{t-1} = \mathbf{v}_{t-1} - \mathbf{v}_{t-2}, \qquad \Delta\mathbf{v}_t = \mathbf{v}_t - \mathbf{v}_{t-1}. \tag{2}$$

To complement the visual cues, the selector also incorporates language and motion information. These appearance, visual-change, language, and motion cues are projected and concatenated into a unified representation:

$$\mathbf{z}_t = \Big[\phi_v(\mathbf{v}_{t-2}), \phi_v(\mathbf{v}_{t-1}), \phi_v(\mathbf{v}_t), \phi_\Delta(\Delta\mathbf{v}_{t-1}), \phi_\Delta(\Delta\mathbf{v}_t), \phi_l(\mathbf{l}_t), \phi_m(\mathbf{m}_t)\Big], \tag{3}$$

where $\mathbf{l}_t$ is the pooled representation of the navigation instruction, $\mathbf{m}_t$ denotes the short-term relative motion feature derived from the corresponding UAV states, and ϕ_v, ϕ_Δ, ϕ_l, and ϕ_m denote lightweight feature projections.

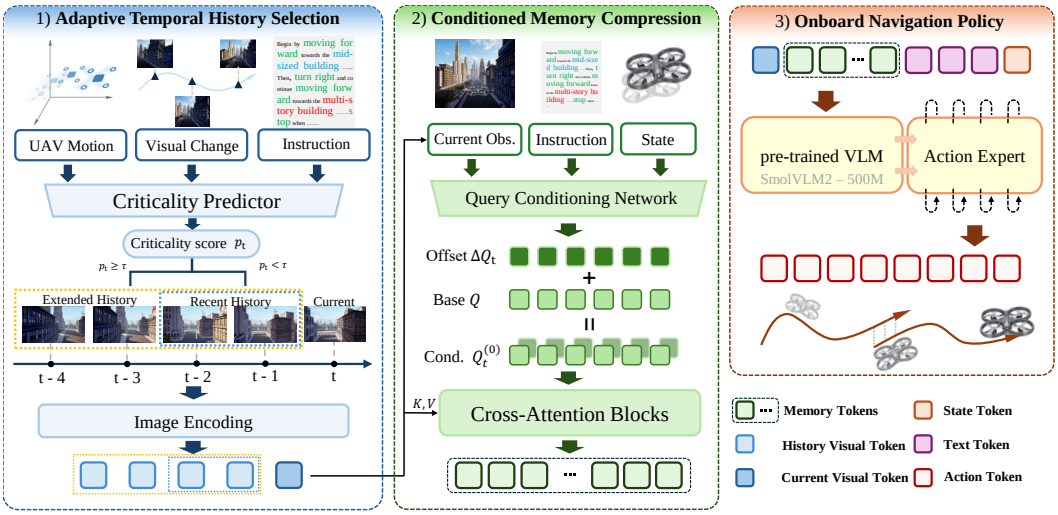


Figure 1: Overview of the proposed temporal perception framework. The adaptive temporal history selector selects a two- or four-image history budget, which determines whether the two older images are encoded. The context-conditioned temporal memory compressor summarizes the selected history into a fixed number of memory tokens. Its queries are conditioned on the current observation, navigation instruction, and UAV state. The resulting visual prefix conditions the VLN policy for action prediction.

Then the fused representation $\mathbf{z}_t$ is fed into two prediction heads, which estimate the probabilities of an upcoming turn and stop:

$$p_t^{\text{turn}} = \sigma(f_{\text{turn}}(\mathbf{z}_t)), \qquad p_t^{\text{stop}} = \sigma(f_{\text{stop}}(\mathbf{z}_t)). \tag{4}$$

These probabilities are finally used to determine the history budget

$$h_t = \begin{cases} 4, & p_t^{\text{turn}} \geq \tau_{\text{turn}} \ \vee \ p_t^{\text{stop}} \geq \tau_{\text{stop}}, \\ 2, & \text{otherwise.} \end{cases} \tag{5}$$

Here, h_t counts historical observations and excludes the current image. When $h_t = 2$, only the two most recent historical frames are used. When $h_t = 4$, two additional older adjacent frames are encoded.

To learn this adaptive history-allocation policy, ATHS is trained to anticipate whether the upcoming action chunk will encounter a maneuver-critical event. During training, turn and stop events are identified from expert trajectories, and a replanning decision is labeled as turn- or stop-critical if its upcoming K-step action horizon overlaps the corresponding event window. This chunk-aware supervision encourages ATHS to predict criticality before the maneuver is reached, rather than only recognizing it from the current frame. At inference time, ATHS is evaluated before the next action chunk is generated: it first determines whether short context is sufficient or whether additional older observations should be encoded, after which the selected history is passed to CTMC and the navigation policy for action generation.

3.3 Context-Conditioned Temporal Memory Compression

Adaptive history selection reduces the number of historical images that need to be encoded, but the resulting visual tokens can still enlarge the multimodal context processed by the policy. We therefore develop a *Context-Conditioned Temporal Memory Compressor* (CTMC) to transform the variable-length history selected by ATHS into a fixed-size temporal memory before action generation.

Given the history budget h_t selected by ATHS, the corresponding historical observations are

$$\mathcal{H}_t^{(h_t)} = \{\mathbf{I}_{t-h_t}, \ldots, \mathbf{I}_{t-1}\}. \tag{6}$$

Here, $\mathcal{H}_t^{(h_t)}$ contains either the short or extended history selected at the current replanning step. After visual encoding, each historical frame produces visual tokens $\mathbf{E}_i$, which are augmented with learned temporal embeddings $\mathbf{T}_i$ and concatenated as

$$\mathbf{X}_t^{\text{hist}} = \text{Concat}\left(\mathbf{E}_{t-h_t} + \mathbf{T}_{t-h_t}, \ldots, \mathbf{E}_{t-1} + \mathbf{T}_{t-1}\right). \tag{7}$$

The resulting $\mathbf{X}_t^{\text{hist}}$ preserves the chronological order of the selected history and serves as the source sequence for memory compression. It is projected into keys and values as

$$(\mathbf{K}_t, \mathbf{V}_t) = \text{Proj}_{K,V}\left(\mathbf{X}_t^{\text{hist}}\right). \tag{8}$$

To compress this variable-length history into a fixed token budget, CTMC maintains M learnable memory queries

$$\mathbf{Q} \in \mathbb{R}^{M \times D}. \tag{9}$$

These queries act as a fixed set of temporal memory slots that retrieve information from the selected history. Rather than using the same queries for every navigation decision, we adapt them to the current navigation context,

$$\mathbf{c}_t = \left[\text{Pool}(\mathbf{X}_t^{\text{cur}}); \text{Pool}(\mathbf{X}_t^{\text{lang}}); \mathbf{x}_t^{\text{state}}\right]. \tag{10}$$

Here, $\mathbf{X}_t^{\text{cur}}$ denotes the current-image tokens, $\mathbf{X}_t^{\text{lang}}$ denotes the instruction tokens, and $\mathbf{x}_t^{\text{state}}$ is the projected UAV-state feature. The context feature $\mathbf{c}_t$ is then used to generate query-specific offsets,

$$\Delta\mathbf{Q}_t = \alpha_q g_\phi(\mathbf{c}_t), \qquad \mathbf{Q}_t^{(0)} = \mathbf{Q} + \Delta\mathbf{Q}_t. \tag{11}$$

Here, $\Delta\mathbf{Q}_t$ is the context-dependent offset applied to the base memory queries $\mathbf{Q}$, and $\mathbf{Q}_t^{(0)}$ denotes the resulting context-conditioned queries for the current replanning step. This allows the compressor to emphasize historical information that is most relevant to the current observation, instruction, and UAV state. Then the conditioned queries retrieve temporal information from the selected history through two cross-attention blocks,

$$\mathbf{M}_t = \text{LN}\left(\text{CrossAttnBlocks}\left(\mathbf{Q}_t^{(0)}, \mathbf{K}_t, \mathbf{V}_t\right)\right). \tag{12}$$

The resulting $\mathbf{M}_t \in \mathbb{R}^{M \times D}$ is the fixed-size temporal memory used by the navigation policy. In our implementation, $M = 16$, so both short and extended histories are represented by the same number of memory tokens.

4 Experiments

4.1 Experimental Setup

Benchmark and metrics. We evaluate AdaFly on the AeroBrain aerial vision-language navigation benchmark, covering 12 test scenes with both seen and unseen environments. We follow the standard closed-loop evaluation protocol and report Success Rate (SR) and Success weighted by Path Length (SPL). For temporal-efficiency analysis, we additionally report the average number of encoded historical images per replanning decision.

Implementation details. All controlled variants use the same lightweight navigation policy described in Sec. 3.1. RGB observations are resized to 384×384, and the policy predicts a chunk of $K = 10$ actions at each replanning step. The short and extended temporal contexts contain two and four historical observations, respectively. The Context-Conditioned Temporal Memory Compressor (CTMC) compresses the selected history into $M = 16$ memory tokens. The AdaFly policy is first trained with expert-derived history allocation, after which it is frozen and the Adaptive Temporal History Selector is trained separately using the action-chunk-aware turn and stop labels described in Sec. 3.2.

Table 1: Navigation performance on the AeroBrain benchmark.

	Seen		Unseen		Overall	
Method	SR ↑	SPL ↑	SR ↑	SPL ↑	SR ↑	SPL ↑
AeroVLA	24.00%	22.61%	**27.67%**	**26.85%**	24.78%	23.51%
WorldVLN	7.13%	5.36%	4.33%	3.77%	6.54%	5.02%
DualVLN	16.62%	16.24%	22.33%	21.89%	17.83%	17.43%
OpenUAV	12.43%	12.43%	26.00%	26.00%	15.30%	15.30%
OmniNav	27.12%	26.12%	24.64%	23.59%	26.60%	25.59%
AdaFly	**49.63%**	**47.32%**	26.89%	25.83%	**44.94%**	**42.95%**

4.2 Overall Performance and Efficiency

Navigation performance. We first compare AdaFly with representative aerial VLN methods on the AeroBrain benchmark. Table 1 reports performance on seen and unseen environments as well as the overall results. AdaFly achieves the strongest overall navigation performance, reaching 44.94% SR and 42.95% SPL. Compared with OmniNav, this corresponds to improvements of 18.34 and 17.36 percentage points in SR and SPL, respectively. The gain is particularly pronounced on seen environments, where AdaFly reaches 49.63% SR. On unseen environments, AdaFly remains competitive with AeroVLA, trailing it by only 0.78 points in SR while substantially outperforming the remaining baselines. These results indicate that our model preserves strong navigation capability despite its smaller parameter count, supported by AdaFly's adaptive history selection and context-conditioned temporal memory.

Model scale and deployment efficiency. Table 2 compares the deployment characteristics of representative navigation systems, while Fig. 2 summarizes the relationship between reported model scale and overall navigation success rate. Since prior methods use different architectures, hardware, and inference pipelines, their reported runtime numbers are provided only as deployment references rather than direct latency comparisons. AdaFly contains 545M parameters in total, substantially below the 3B–8B backbone scale commonly used by the compared systems. This compact scale also enables the complete policy to execute locally on the Jetson Orin NX of the DJI Manifold 3 without offboard policy inference. Among the compared systems, OnFly likewise demonstrates fully onboard execution on an Orin NX, whereas WorldVLN and DualVLN rely on remote-server inference for real-world deployment. In our real-flight service logs, AdaFly requires approximately 270 ms to generate an action chunk; with asynchronous chunk execution, the flight-control loop operates at 20 Hz despite the lower replanning frequency. Detailed onboard profiling is provided in Sec. 4.4.

Table 2: Reported deployment characteristics of representative aerial VLN systems. N/R denotes not reported. For DualVLN, 0.7 s refers to System 2 trajectory-token inference and 0.03 s to System 1 generation of 32 trajectories with TensorRT. Runtime and rate values are not directly comparable across different inference pipelines and hardware configurations.

Method	Compute Platform	Fully Onboard	Real-World Flight	Runtime / Rate
AeroVLA	Workstation	N/R	N/R	N/R
WorldVLN	Remote server	No	Yes	N/R
DualVLN	RTX 4090 server	No	N/R	0.7 / 0.03 s
OpenUAV	N/R	N/R	N/R	N/R
OmniNav	N/R	N/R	Yes	up to 5 Hz
OnFly	Jetson Orin NX	Yes	Yes	0.81 s / decision
AdaFly	**Jetson Orin NX**	**Yes**	**Yes**	**$\sim$270 ms / chunk**

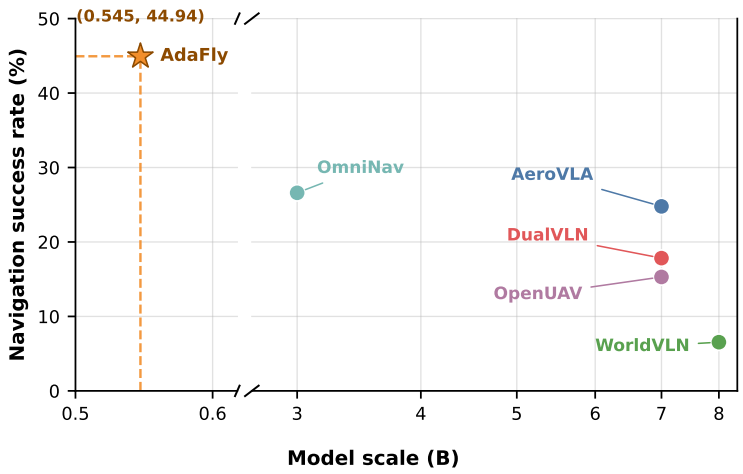


Figure 2: Overall navigation success rate versus reported model scale. Prior methods use the reported VLM/LLM backbone scale when complete system parameter counts are unavailable, whereas AdaFly uses the total checkpoint parameter count.

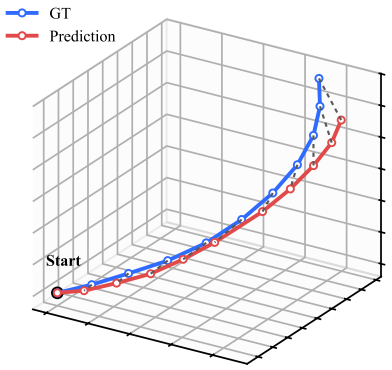
4.3 Temporal Memory Utility and Adaptation

We next analyze why adaptive memory and temporal perception are effective and how the two components of AdaFly contribute to its performance.

Temporal History Utility. We first isolate the marginal effect of extending the history budget from H2 to H4. For each decision, the same policy checkpoint is evaluated twice with identical current observation, instruction, UAV state, target action chunk, and initial flow noise; only the active history budget is changed. This paired analysis covers 490,090 decisions from 7,082 episodes across all 12 AeroBrain test scenes. We define *Move Error* as the mean of the point-wise Euclidean displacement errors over the ten predicted positions in an action chunk. Table 3 reveals a clear decision-dependent asymmetry in the value of temporal context. Extending the history from H2 to H4 reduces *Move Error* by 12.29% at maneuver-critical decisions. In contrast, the same additional history increases the error by 2.72% during non-critical motion. The overall gain of 7.46% therefore masks substantially different behavior across navigation stages. These results motivate allocating additional temporal computation selectively rather than processing an extended history at every decision.

Table 3: Paired action-prediction error under H2 and H4 history budgets. Move Error is the mean point-wise Euclidean displacement over the ten positions in each action chunk (right); lower is better. Positive relative gain favors H4.

	Move Error ↓		
Subset	H2	H4	Rel. gain ↑
Overall	0.24 m	**0.22 m**	**+7.46%**
Non-critical	**0.13 m**	0.14 m	−2.72%
Critical	0.37 m	**0.32 m**	**+12.29%**
Turn	0.32 m	**0.28 m**	+12.28%
Stop	0.41 m	**0.36 m**	+12.55%



Adaptive Temporal History Selection. We next evaluate whether ATHS can exploit this asymmetry in closed-loop navigation. Random 2/4 selects H2 or H4 without using navigation context while maintaining a similar average history budget to ATHS, providing a context-agnostic baseline. As shown in Table 4(a), we compare ATHS against fixed H2, fixed H4, and Random 2/4 to isolate the benefit of adaptive history allocation. The learned ATHS achieves 44.94% SR while using only 2.82 historical images per decision on average. Under a nearly matched history budget, it outperforms Random 2/4 by 4.16 percentage points in SR and 4.09 points in SPL, showing that effective allocation depends on where the additional temporal computation is spent. Compared with Fixed H4, ATHS improves SR while reducing the average history-image budget by 29.5%.

Table 4: Ablations of history allocation and temporal-memory compression. Results are averaged over three runs.

(a) History allocation

Method	SR ↑	SPL ↑	Avg. Hist. ↓
Fixed H2	38.44%	36.66%	2.00
Fixed H4	43.22%	42.23%	4.00
Random 2/4	40.78%	38.86%	2.82
ATHS	**44.94%**	**42.95%**	**2.82**

(b) Temporal-memory compression

Method	SR ↑	SPL ↑	Hist. Tokens ↓
Raw H4	26.78%	25.76%	144
Uncond. CTMC	40.25%	38.41%	16
Cond. CTMC	**45.03%**	**43.11%**	**16**

Context-Conditioned Temporal Memory Compression. Finally, we isolate the effect of temporal memory compression under a fixed H4 history budget. Table 4(b) shows that both CTMC variants compress the 144 historical visual tokens to 16 memory tokens, an 88.9% reduction. Compared with raw H4 history, the conditioned CTMC yields a 68.15% relative improvement in SR and a 67.35% relative improvement in SPL. More importantly, conditioning the memory queries on the current navigation context further improves SR by 11.88% and SPL by 12.24% relative to the unconditional CTMC. These results show that compact temporal memory is most effective when its representation is formed according to the current visual, linguistic, and motion context.

4.4 Onboard Efficiency and Real-World Deployment

Onboard efficiency. We deploy the complete AdaFly policy on a DJI Manifold 3 equipped with an NVIDIA Jetson Orin NX. All visual processing, temporal-memory computation, multimodal reasoning, and action-chunk generation are executed locally on the onboard computer without offboard policy inference. Table 5 summarizes the deployment characteristics of the complete system.

Table 5: Onboard efficiency and deployment footprint of AdaFly on the DJI Manifold 3 (Jetson Orin NX).

(a) Runtime efficiency

Metric	Fixed-H2	Fixed-H4	**AdaFly**
History budget	2	4	Adaptive
Mean latency	253.13 ms	297.24 ms	**268.44 ms**
P95 latency	257.49 ms	303.01 ms	**272.91 ms**
Flight-control rate	–	–	**20 Hz**

(b) Deployment footprint

Metric	**AdaFly**
Parameters	545M
Policy precision	FP16
Peak PyTorch alloc.	1.17 GiB
Peak unified mem. (**43.8%**)	**6.50**/14.83 GB

The 545M-parameter policy runs in FP16 and requires approximately **270 ms** to generate an action chunk according to our onboard flight logs. Despite the lower policy replanning frequency, asynchronous chunk execution maintains a **20 Hz** flight-control loop, allowing continuous control while the next action chunk is being generated. Together with the model-scale comparison in Table 2, these results demonstrate that AdaFly provides a practical lightweight alternative to the multi-billion-parameter models commonly used in aerial VLN.

Real-world deployment. Beyond simulation, we deploy AdaFly on a DJI Matrice 4TD with the Manifold 3 mounted onboard and evaluate the complete navigation pipeline in real outdoor envi-

ronments. For each trial, the UAV is initialized at the designated starting location and receives a natural-language navigation instruction. RGB observations and UAV states are processed online by AdaFly, and the predicted action chunks are executed directly by the flight-control interface. No external workstation or remote inference server is used during navigation. We conduct 40 real-world language-conditioned flight trials covering straight flight, directional changes, target approach, and stopping. AdaFly achieves an overall success rate of approximately 30% in these physical UAV experiments. Figure 3 shows representative successful rollouts, with first-person observations sampled along each trajectory. These examples demonstrate that the onboard policy can interpret language-conditioned visual observations, execute multi-stage maneuvers, and terminate near the instructed target. All perception and policy inference are executed locally onboard, providing an end-to-end validation of AdaFly on a physical UAV without offboard inference.

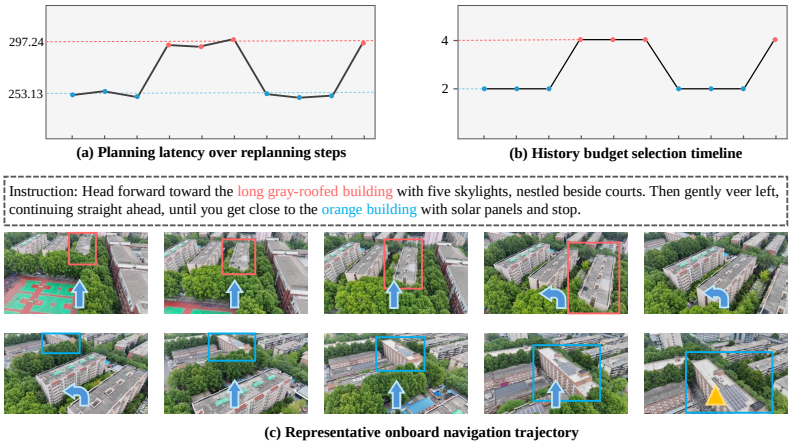


Figure 3: Representative fully onboard real-world flight with AdaFly. (a) Planning latency over replanning steps. (b) History budget selection timeline. (c) Chronologically sampled onboard observations along the corresponding trajectory. In (a) and (b), the horizontal axis denotes replanning steps, while the vertical axes denote planning latency (ms) and the selected history budget, respectively.

5 Conclusion

We presented AdaFly, a lightweight approach for fully onboard aerial vision-language navigation. Our study shows that the value of additional temporal context is strongly decision-dependent: extended visual history is most beneficial near maneuver-critical decisions, while unnecessary history can introduce redundant computation and even degrade prediction. Motivated by this observation, AdaFly combines an *Adaptive Temporal History Selector* (ATHS) to dynamically allocate the history-image budget with a *Context-Conditioned Temporal Memory Compressor* (CTMC) to represent the selected history under a fixed token budget. Built on a lightweight 545M-parameter policy, AdaFly achieves strong navigation performance while substantially reducing temporal visual processing. We further demonstrate the complete system on a DJI Matrice 4TD with all policy inference executed locally on a Jetson Orin NX, validating the practicality of adaptive temporal memory for onboard aerial VLN.

AI Use Statement

Generative AI tools were used to assist with literature retrieval and discovery, discussion and refinement of research ideas and experimental design, drafting parts of the manuscript, and language editing and polishing. We reviewed and verified all AI-assisted content and independently validated the reported experiments, results, and technical claims. We take full responsibility for the final content of this work.

REFERENCES

Xichen Ding, Jianzhe Gao, Cong Pan, Wenguan Wang, and Jie Qin. History-enhanced two-stage transformer for aerial vision-and-language navigation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 18225–18233, 2026. doi: 10.1609/aaai.v40i22.38885.

Yunpeng Gao, Chenhui Li, Zhongrui You, Junli Liu, Zhen Li, Pengan Chen, Qizhi Chen, Zhonghan Tang, Liansheng Wang, Penghui Yang, Yiwen Tang, Yuhang Tang, Shuai Liang, Songyi Zhu, Ziqin Xiong, Yifei Su, Xinyi Ye, Jianan Li, Yan Ding, Dong Wang, Xuelong Li, Zhigang Wang, and Bin Zhao. OpenFly: A comprehensive platform for aerial vision-language navigation. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=OKm3w71ymP.

Keji He, Ya Jing, Yan Huang, Zhihe Lu, Dong An, and Liang Wang. Memory-adaptive vision-and-language navigation. *Pattern Recognition*, 153:110511, 2024. doi: 10.1016/j.patcog.2024.110511.

Lixuan He, Haoyu Dong, Zhenxing Chen, Yangcheng Yu, Jie Feng, and Yong Li. Mem4Nav: Boosting vision-and-language navigation in urban environments with a hierarchical spatial-cognition long-short memory system. *arXiv preprint arXiv:2506.19433*, 2025. doi: 10.48550/arXiv.2506.19433. URL https://arxiv.org/abs/2506.19433.

Wen Jiang, Li Wang, Kangyao Huang, Wei Fan, Jinyuan Liu, Shaoyu Liu, Hongwei Duan, Bin Xu, and Xiangyang Ji. LongFly: Long-horizon UAV vision-and-language navigation with spatiotemporal context integration. *arXiv preprint arXiv:2512.22010*, 2025. doi: 10.48550/arXiv.2512.22010. URL https://arxiv.org/abs/2512.22010.

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, October 2023. URL https://openaccess.thecvf.com/content/ICCV2023/html/Liu_AerialVLN_Vision-and-Language_Navigation_for_UAVs_ICCV_2023_paper.html.

Mustafa Shukor, Dana Aubakirova, Francesco Capuano, Pepijn Kooijmans, Steven Palma, Adil Zouitine, Michel Aractingi, Caroline Pascal, Martino Russi, Andres Marafioti, Simon Alibert, Matthieu Cord, Thomas Wolf, and Remi Cadene. SmolVLA: A vision-language-action model for affordable and efficient robotics. *arXiv preprint arXiv:2506.01844*, 2025. doi: 10.48550/arXiv.2506.01844. URL https://arxiv.org/abs/2506.01844.

Marco S. Tayar, Felipe Tommaselli, Gianluca Capezutto, Pedro Antonio Rabelo Saraiva, Pedro H. V. de Freitas, Lucas Kido, Guilherme Sonego, Ricardo V. Godoy, and Marcelo Becker. VLN on the fly: An onboard vision-language navigation stack for aerial robots. In *International Micro Air Vehicle Conference and Competition (IMAV)*, 2026.

Xiangyu Wang, Donglin Yang, Ziqin Wang, Hohin Kwan, Jinyu Chen, Wenjun Wu, Hongsheng Li, Yue Liao, and Si Liu. Towards realistic UAV vision-language navigation: Platform, benchmark, and methodology. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/15ce8e7afe5ee95bad56e3b9be28d3d1-Paper-Conference.pdf.

Peng Xu, Zhengnan Deng, Jiayan Deng, Zonghua Gu, and Shaohua Wan. AeroVLA: A vision-language-action model for UAV navigation via minimalist end-to-end control. In *European Conference on Computer Vision (ECCV)*, pp. 289–305. Springer, 2026a.

Peng Xu, Chengcheng Wang, and Shaohua Wan. AeroDPO: Unleashing lightweight UAV navigation with high-fidelity perception and automated preference optimization. *arXiv preprint arXiv:2608.07557*, 2026b. doi: 10.48550/arXiv.2608.07557. URL https://arxiv.org/abs/2608.07557.

Jiazhao Zhang, Kunyu Wang, Shaoan Wang, Minghan Li, Haoran Liu, Songlin Wei, Zhongyuan Wang, Zhizheng Zhang, and He Wang. Uni-NaVid: A video-based vision-language-action model for unifying embodied navigation tasks. In *Robotics: Science and Systems*, 2025a. URL https://www.roboticsproceedings.org/rss21/p013.html.

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, Vienna, Austria, July 2025b. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.638. URL https://aclanthology.org/2025.acl-long.638/.

Lingfeng Zhang, Yuecheng Liu, Zhanguang Zhang, Matin Aghaei, Yaochen Hu, Hongjian Gu, Mohammad Ali Alomrani, David Gamaliel Arcos Bravo, Raika Karimi, Atia Hamidizadeh, Haoping Xu, Guowei Huang, Zhanpeng Zhang, Tongtong Cao, Weichao Qiu, Xingyue Quan, Jianye Hao, Yuzheng Zhuang, and Yingxue Zhang. Mem2Ego: Empowering vision-language models with global-to-ego memory for long-horizon embodied navigation, 2025c.

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. Association for Computational Linguistics, 2025d. doi: 10.18653/v1/2025.acl-long.1511.

Baining Zhao, Jiacheng Xu, Weicheng Feng, Xin Zhang, Zhaolu Wang, Haoyang Wang, Shilong Ji, Ziyou Wang, Jianjie Fang, Zhiheng Zheng, Weichen Zhang, Yu Shang, Wei Wu, Chen Gao, Xinlei Chen, and Yong Li. WorldVLN: Autoregressive world action model for aerial vision-language navigation, 2026. URL https://arxiv.org/abs/2605.15964.

Guiyong Zheng, Yueting Ban, Mingjie Zhang, Juepeng Zheng, and Boyu Zhou. OnFly: Onboard zero-shot aerial vision-language navigation toward safety and efficiency, 2026. URL https://arxiv.org/abs/2603.10682.