Reason-WAM: Vision-Language Reasoning for World Action Models in Urban Aerial Goal Navigation

Anonymous ACL submission

Abstract

Goal navigation is a fundamental problem in embodied intelligence, requiring an agent to reach a target specified by a sparse natural-language goal. The task inherently demands both high-level spatial reasoning for goal interpretation and route planning, and low-level action grounding for executable control. World Action Models (WAMs) are an emerging paradigm for embodied control: by first predicting future visual states and then recovering actions through inverse dynamics, they leverage visual pretraining priors for strong spatial action grounding, but are typically limited to simple and short-horizon tasks. We propose **Reason-WAM**, a hierarchical navigation framework in which a VLM performs goal reasoning and route planning, then drives the WAM to transform route-level intents into executable actions. To train this architecture effectively for goal navigation, we introduce a bottom-up training framework that first grounds visual transitions into actions, then learns instruction-conditioned visual dynamics, and finally optimizes the VLM planner with multi-turn GRPO for closed-loop decision-making. Experiments on a public urban aerial goal-navigation benchmark, a particularly challenging setting characterized by large-scale spaces and continuous three-dimensional motion, show that Reason-WAM achieves state-of-the-art performance, demonstrating its ability to combine the high-level reasoning strength of VLMs with the spatial action grounding capability of WAMs. The anonymous project page is available at: https://reason-wam-anonymous.github.io/project-page/.

1 Introduction

A central goal of embodied intelligence is to enable robots to act upon natural-language intentions: a user specifies what should be achieved, and the agent autonomously determines how to achieve it through perception and action (Ku et al., 2020; Gu et al., 2022; Zitkovich et al., 2023). Goal-oriented navigation is a canonical instance of this language-to-action capability (Chaplot et al., 2020a; Majumdar et al., 2022). Given a natural-language goal such as "a nearby convenience store," the agent is not provided with a step-by-step route; instead, it must continuously observe the environment, infer where the target is likely to appear, and move toward it until success (Song et al., 2025; Zhao et al., 2026a). This setting becomes particularly challenging in urban aerial environments, where the agent must operate in large-scale 3D space under partial observation (Fan et al., 2023; Zhang et al., 2025d). We observe that successful goal-oriented aerial navigation requires two complementary capabilities:

- **Spatial reasoning.** Since no route is specified, the agent must infer a goal-directed exploration strategy from visual observations and task context (Zhou et al., 2024; Ji et al., 2026). Such reasoning should guide the agent toward likely target regions, actively reveal informative visual cues, and update the plan as new observations become available. For example, to find a convenience store, an agent may first move toward a road to expose storefronts, inspect shop signs, and then approach the most likely target.

- **Spatial action grounding.** Once an intermediate navigation intent is determined, the agent must realize it as executable actions grounded in the surrounding 3D environment (Kim et al., 2026; Wang et al., 2025a). This requires accounting for viewpoint changes, metric distance, altitude, orientation, local geometry, and the spatial relations between the agent and visible landmarks. For instance, a subgoal such as "move toward the road" must be translated into an action sequence that follows the current visual layout and moves the agent toward the intended region.

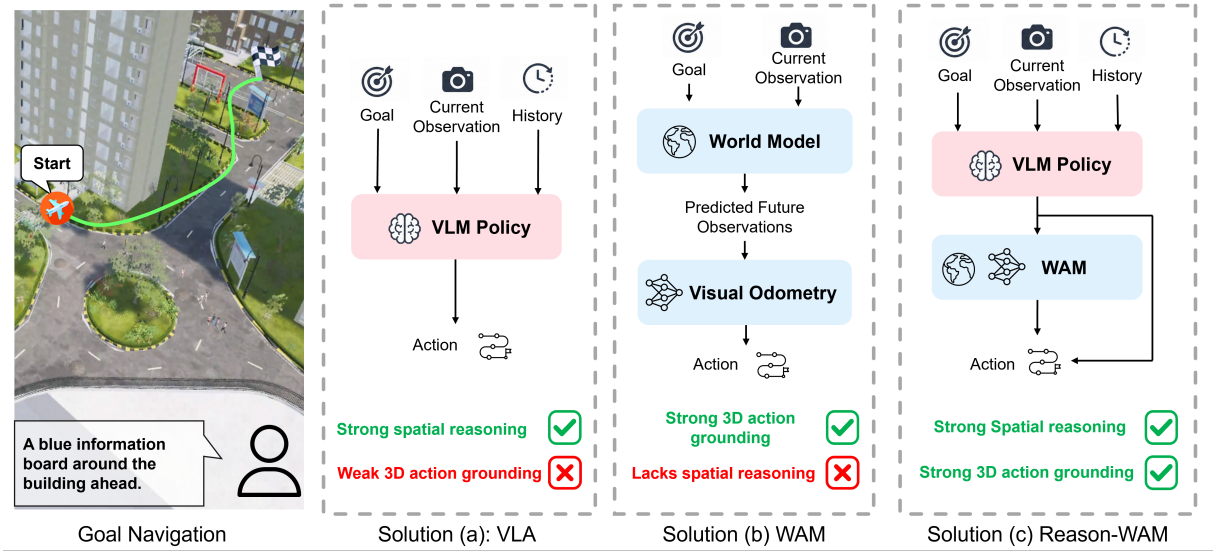

Figure 1: Overview of goal-oriented urban aerial navigation and the core motivation of Reason-WAM. The agent receives only a sparse natural-language goal rather than a step-by-step route, requiring high-level spatial reasoning to infer plausible subgoals and grounded action execution to realize them in continuous 3D space.

Large-scale pretrained models have opened new opportunities for embodied spatial intelligence (Zhang et al., 2024b), yet existing paradigms only partially address these two capabilities. LLM- or VLM-based agents (Shah et al., 2023; Zhang et al., 2025e; Qi et al., 2025) use foundation models as high-level decision makers, enabling semantic interpretation, commonsense reasoning, and flexible planning over navigation goals. However, these agents typically rely on handcrafted prompts or task-specific mapping tools, which limits their generality. Vision-language-action (VLA) models (Zhang et al., 2025b; Gao et al., 2026; Chen et al., 2026b) offer a more direct mapping from observations and language to actions. By operating over semantic tokens, they inherit strong abstract reasoning ability from pretrained vision-language models; nevertheless, deriving precise action tokens from highly compressed semantic representations remains difficult. This compression weakens action grounding in continuous 3D space, where navigation requires fine-grained sensitivity to viewpoint, distance, orientation, altitude, and local scene geometry. More recently, world models and world action models (WAMs) (Li et al., 2026; Ye et al., 2026; Bar et al., 2025) provide a complementary direction by predicting short-horizon visual futures from the current observation and a local instruction, and then inversely decoding the predicted transition into actions. This prediction-and-decoding process makes WAMs well suited for grounding concrete short commands in the surrounding 3D environment (Huang et al., 2026). Yet, WAMs lack high-level task reasoning: they require an appropriate local instruction as input and are not designed to decompose an open-ended navigation goal into a sequence of executable subgoals.

To this end, we propose **Reason-WAM**, a hierarchical architecture that couples VLM-based spatial reasoning with selective WAM-based action grounding, as illustrated in Figure 1. Given a navigation goal, the VLM reasons over current observations, interaction history, and the goal description to iteratively plan the next step. Each step can take one of three forms: acquiring additional surrounding observations at the current location, issuing a primitive control action, or producing a short route-oriented VLN instruction. Reason-WAM directly executes the first two lightweight forms for efficiency, while delegating VLN-instruction steps to the WAM for grounded execution. The world model predicts how the embodied visual state would change if the instruction were executed, and the predicted transition is converted by a action-decoding module into an executable action sequence. To instantiate this architecture, we design a bottom-up training pipeline: we first train the action-decoding module from video-trajectory pairs, then train the world model with short-horizon route-oriented instruction-video pairs, and finally optimize the VLM with multi-turn Group Relative Policy Optimization (GRPO) to learn goal decom-

position and selective WAM invocation.

Our main contributions are as follows:

- We propose **Reason-WAM**, a hierarchical architecture that couples VLM-based spatial reasoning with WAM-based action grounding for goal-oriented urban aerial navigation.
- We develop a bottom-up training framework that learns visual odometry, short-horizon world-action dynamics, and VLM-based goal reasoning with multi-turn GRPO.
- We show that Reason-WAM improves open-ended aerial goal navigation with stronger spatial grounding, more efficient execution, and better task decomposition.

2 Related Work

Vision-Language Navigation (VLN). Vision-language navigation studies closed-loop perception and action under natural-language guidance (Anderson et al., 2018). Existing settings can be broadly divided into route-oriented navigation (Jain et al., 2019; Liu et al., 2023), where instructions provide explicit step-by-step movement cues, and goal-oriented navigation, where only a target description is given. The latter (Zhu et al., 2021; Chaplot et al., 2020b) is more challenging, requiring the agent to use spatial commonsense reasoning to derive feasible plans and translate the action intent of each step into executable actions. These difficulties are further amplified in aerial object navigation, where large-scale urban layouts, continuous 3D flight, altitude-dependent visual appearance, and sparse target cues jointly make both spatial reasoning and action grounding substantially harder (Xiao et al., 2025).

Vision Language Models for VLN. Large language models and vision-language models have been increasingly used in embodied navigation, either as agentic planners that decompose goals and maintain semantic memory (Zhang et al., 2024a, 2025c), or as multimodal backbones in vision-language-action policies that directly predict actions (Zhang et al., 2025a,b). Their strength lies in semantic abstraction and instruction-level reasoning, enabling high-level decisions such as where relevant objects are likely to appear (Qiao et al., 2025). However, such models still struggle with fine-grained spatial action and continuous control, often exhibiting unstable orientation, weak metric awareness, and unreliable conversion from semantic intent to executable 3D actions.

World Action Models for VLN. World Action Models (WAMs) offer a predictive action-grounding paradigm that first emerged in manipulation (Du et al., 2023; Chen et al., 2026a), where action-conditioned future states support planning and inverse dynamics. This idea has recently been extended to navigation by predicting visual futures under candidate instructions or actions and decoding executable movements from the predicted consequences (Huang et al., 2026; Zhao et al., 2026b). Such models provide strong local spatial priors and improve short-horizon action grounding, but remain less suited to open-ended goal navigation: their visual predictive basis lacks abstract chain-of-thought reasoning, while prevailing diffusion-based architectures struggle with long-horizon generation.

3 Method

For goal-oriented navigation, each episode provides a natural-language goal g, an initial embodied observation o_0. At step t, the agent observes visual input o_t, maintains an interaction history h_t, and executes an action a_t that updates its hidden pose by $p_t = F(p_{t-1}, a_t)$. An episode is successful if the final pose reaches the target region $p^\star$, i.e., $\|p_T - p^\star\|_2 \leq \epsilon$. To solve this problem, Reason-WAM combines a hierarchical model architecture with a bottom-up training framework: a VLM performs goal reasoning and step selection, while a WAM grounds route-oriented steps into executable aerial actions, as shown in Figure 2.

3.1 Model Architecture

Reason-WAM factorizes navigation into VLM-based spatial reasoning and WAM-based action grounding. At each decision point, the VLM π_θ conditions on the goal g, the current embodied observation o_t, and the interaction history h_t to reason about an overall plan P_t and instantiate its next executable step s_t:

$$(P_t, s_t) \sim \pi_\theta(\cdot \mid g, o_t, h_t), \quad s_t \in \mathcal{S}_{\text{sense}} \cup \mathcal{S}_{\text{ctrl}} \cup \mathcal{S}_{\text{route}}. \tag{1}$$

Here P_t captures the VLM's current high-level reasoning over the goal and scene, while s_t is the step to be executed at the current turn. The step space is organized by execution interface: $\mathcal{S}_{\text{sense}}$ requests additional surrounding observations when

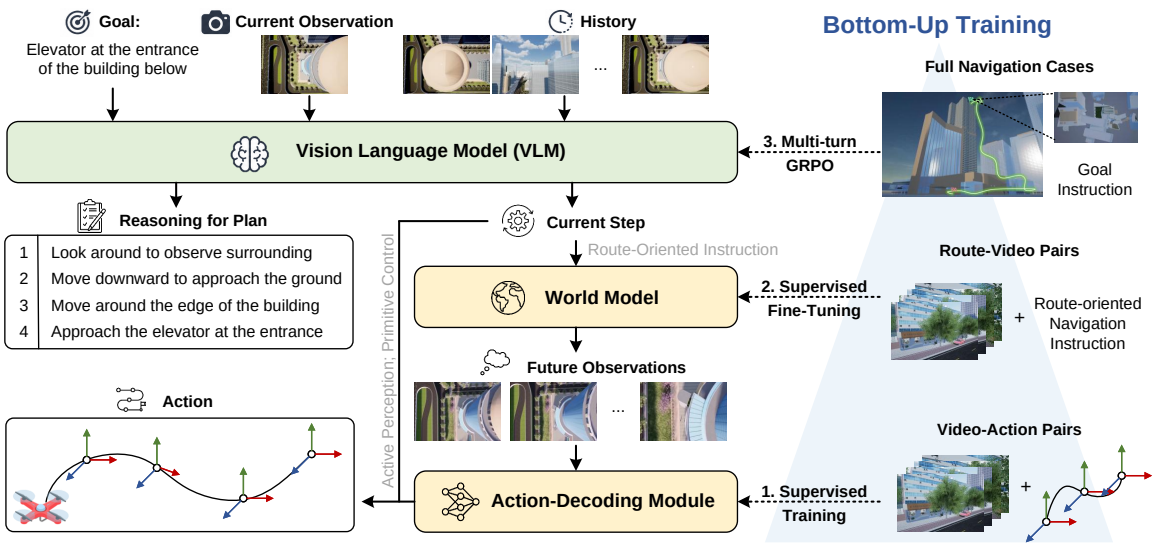

Figure 2: Architecture and bottom-up training pipeline of Reason-WAM. The VLM planner performs goal reasoning and selects sensing, primitive-control, or route-instruction steps; the WAM grounds route-oriented instructions by predicting visual futures, and the action decoder converts predicted transitions into executable aerial actions. Training proceeds bottom-up from action decoding, to instruction-conditioned world modeling, and finally to multi-turn GRPO for closed-loop VLM planning.

the current view is insufficient; $\mathcal{S}_{\text{ctrl}}$ contains primitive controls for immediate local adjustment; and $\mathcal{S}_{\text{route}}$ contains short route-oriented VLN instructions whose spatial intent must be grounded over multiple steps. Although the first two types could also be mediated by a world model, they correspond to relatively fixed sensing or primitive-control operations and are therefore executed directly for efficiency, whereas route instructions are delegated to the WAM for action grounding.

For a VLN-instruction step $s_t \in \mathcal{S}_{\text{route}}$, the step generally corresponds to a route-oriented navigation instruction, denoted as ℓ_t. The world model π_ϕ serves as an intermediate predictive interface: instead of committing to low-level controls immediately, it first forecasts the visual consequences of following the instruction. This converts the abstract route intent into an explicit future-state target, indicating how the embodied observation should change if the instruction is correctly executed. The predicted visual rollout provides dense spatial cues for action grounding, thereby bridging route semantics and low-level motion. The process can be formalized as:

$$\hat{o}_{t+1:t+H} \sim \pi_\phi(\cdot \mid o_t, s_t), \tag{2}$$

where H is the execution horizon, and $\hat{o}_{t+1:t+H}$ denotes a sampled future observation sequence. The resulting visual transition is decoded into executable aerial actions by a learning-based visual odometry model (Wang et al., 2020), which estimates relative motion from o_t and the predicted rollout. Its learned robustness to appearance changes and prediction artifacts makes it suitable for using world-model outputs rather than real future observations:

$$a_{t:t+H-1} = D_\psi(o_t, \hat{\mathbf{o}}_{t+1:t+H}). \tag{3}$$

After execution, the new observation is derived, enabling the VLM to re-plan in a closed loop.

3.2 Bottom-Up Training Framework

We adopt a bottom-up training curriculum that reflects the dependency structure of embodied navigation. Each upper-level module relies on a reliable lower-level abstraction: the VLM manager depends on the world model to translate semantic intent into visual predictions, while the world model depends on the action decoder to connect visual transitions with executable motions. This staged training design decouples metric action grounding from semantic decision making, reduces long-horizon credit assignment, and makes the final policy easier to optimize.

Concretely, suppose that each object navigation case in a training dataset contains a navigation goal

g, a human flight video $o^\star_{0:T}$, and the corresponding action-coordinate sequence $a^\star_{0:T-1}$. The training process is as follows:

Action-decoding supervised training. For a trajectory of length T, we uniformly partition it into segments of length at most H; trajectories shorter than H are kept intact. This yields video–action pairs for training the action decoder. Given a trajectory segment $(o^\star_{i:j}, a^\star_{i:j-1})$, we train the decoder D_ψ as visual odometry or inverse dynamics:

$$\mathcal{L}_{\text{act}}(\psi) = \sum_{(i,j)} \left\| D_\psi(o^\star_i, o^\star_{i+1:j}) - a^\star_{i:j-1} \right\|. \quad (4)$$

World-model supervised fine-tuning. The segmented video $o^\star_{i:j}$ is further relabeled by a powerful VLM into a linguistic route-oriented step $\hat{s}_t$ describing the local movement intent. The route-oriented navigation instruction–video pairs supervise the world model π_ϕ to generate the future visual sequence from the initial observation and route step:

$$\mathcal{L}_{\text{wm}}(\phi) = -\sum_{(i,j)} \log \pi_\phi(o^\star_{i+1:j} \mid o^\star_i, \hat{s}_t). \quad (5)$$

VLM training with multi-turn GRPO. After D_ψ and π_ϕ are trained, we optimize the VLM π_θ with multi-turn GRPO (Guo et al., 2025; Feng et al., 2025) rather than supervised imitation. In open-ended goal navigation, a sparse goal does not specify a unique expert decomposition, and a locally plausible step may become suboptimal after several rounds of execution. We therefore train the VLM through closed-loop rollouts that directly evaluate task-level progress. At the m-th VLM decision, the model observes $c_m = (g, o_m, h_m)$ and samples $y_m = (P_m, s_m) \sim \pi_\theta(\cdot \mid c_m)$; the selected step is executed directly or through the WAM according to its type, updating the observation and history for subsequent decisions. We index the resulting trajectory at the decision level and define the decision-level reward as:

$$r_m = \lambda_s \mathbf{1}[\|p_M - p^\star\|_2 \le \epsilon] + \lambda_p \mathbf{1}[d_m < d_{m-1}], \quad (6)$$

where M is the number of VLM decisions, p_m is the pose after executing the m-th decision, and $d_m = \|p_m - p^\star\|_2$. The final success term is shared by all decisions in the trajectory, while the progress term indicates whether the current decision moves closer to the target. For each training episode, we sample a group of complete trajectories $\{\tau_k\}_{k=1}^K$, compute their rollout scores from the decision-level rewards, normalize these scores within the group to obtain estimated trajectory-level advantages $\hat{A}_k$, and assign each $\hat{A}_k$ to all VLM decisions in τ_k. The VLM is updated with the clipped GRPO objective:

$$\mathcal{L}_{\text{grpo}}(\theta) = -\mathbb{E}_{k,m}\Big[\min\big(\rho_{k,m}\hat{A}_k, \text{clip}(\rho_{k,m}, 1-\eta, 1+\eta)\hat{A}_k\big)\Big], \quad (7)$$

where $\rho_{k,m} = \pi_\theta(y_{k,m} \mid c_{k,m}) / \pi_{\text{old}}(y_{k,m} \mid c_{k,m})$ is computed for the m-th VLM decision in trajectory τ_k.

To sum up, this bottom-up procedure first trains D_ψ to ground visual transitions into actions, then learns instruction-conditioned visual dynamics with π_ϕ, and finally trains π_θ for spatial reasoning ability.

4 Experiments

4.1 Experimental Setup

Dataset and benchmark. We evaluate ReasonWAM on goal-oriented embodied navigation in urban airspace, following the EmbodiedCity (Gao et al., 2024a) simulator and the EmbodiedNavBench (Zhao et al., 2026a) task definition. This benchmark provides a large-scale continuous urban aerial environment with high-quality human-collected trajectories, continuous egocentric visual observations, and natural-language goal specifications, making it well suited for evaluating both long-horizon spatial reasoning and fine-grained action grounding. We adopt 5-fold cross-validation: for each fold, we split the data into disjoint training and test sets, train the compared models on the training split, and repeat evaluation on the corresponding test split.

Evaluation metrics. We follow the evaluation metrics in EmbodiedNav-Bench: Success Rate (SR), Success weighted by Path Length (SPL), and Navigation Error (NE). SR measures the fraction of episodes that terminate within the success radius of the target. SPL jointly measures task success and path efficiency. NE measures the final distance between the agent and the target in meters.

Baselines. We organize baselines into two groups. The first group follows the zero-shot benchmark protocol of EmbodiedNav-Bench, which evaluates general spatial-action ability without task-specific training. It covers direct LMM action prediction in the action-as-language paradigm, including the Qwen2.5-VL series (Bai

Method	Short			Middle			Long			Average		
	SR↑	SPL↑	NE↓	SR↑	SPL↑	NE↓	SR↑	SPL↑	NE↓	SR↑	SPL↑	NE↓
Zero-shot baselines from EmbodiedNav-Bench												
Qwen2.5-VL-3B	9.7	9.2	148.6	3.0	2.8	215.0	0.0	0.0	467.5	4.2	4.0	277.0
Qwen2.5-VL-7B	9.9	9.5	86.2	3.5	3.2	109.9	0.0	0.0	300.7	4.5	4.2	165.6
Qwen-VL-Max	27.3	27.0	78.9	6.4	6.2	148.2	5.3	4.7	344.7	13.0	12.6	190.6
GLM-4.6V	29.3	29.1	62.3	6.2	6.1	115.5	0.0	0.0	278.2	12.0	11.9	151.1
Gemini-2.0-Flash	31.8	29.3	64.4	11.3	10.6	130.8	0.0	0.0	335.7	14.7	13.3	177.0
GPT-4o	34.2	30.5	70.7	9.8	7.9	90.9	0.0	0.0	200.2	14.7	12.8	120.6
GPT-4.1	51.5	45.7	56.8	15.2	15.0	93.0	6.3	5.9	189.3	24.3	22.2	113.0
Qwen3-VL-Plus	30.1	29.7	83.7	8.7	8.5	122.9	0.0	0.0	301.2	12.9	12.7	169.3
Gemini-2.5-Flash	35.5	33.6	71.4	9.1	9.1	95.2	0.0	0.0	223.3	14.9	14.2	129.4
Gemini-2.5-Pro	27.7	24.8	64.0	11.5	10.1	105.5	9.8	8.3	388.2	16.3	14.4	185.9
OpenAI o4-mini	35.6	33.1	51.7	22.4	19.8	91.4	2.7	2.6	224.3	20.2	18.5	122.5
GPT-5.1	**52.9**	**48.6**	60.9	**30.4**	**29.0**	**75.3**	**18.2**	**18.0**	186.1	**34.0**	**32.0**	107.0
SayNav	25.6	24.3	124.0	21.7	19.0	132.6	6.8	5.5	225.0	18.0	16.3	160.5
STMR	35.9	23.8	71.5	15.0	10.6	87.7	3.6	3.1	183.8	18.2	12.5	114.3
PRPSearcher	42.4	24.4	**27.0**	18.2	12.6	**52.0**	2.9	1.9	**170.6**	21.2	13.0	**83.2**
Trained VLM and VLA baselines												
Qwen3-VL (SFT)	48.5	47.9	57.5	27.3	24.7	108.3	8.8	8.8	226.9	28.0	27.0	131.9
Uni-NaVid	39.4	38.5	79.3	24.2	24.2	122.6	8.8	8.8	235.0	24.0	23.7	146.5
OpenVLA-UAV	48.5	48.5	80.3	18.2	18.2	148.9	2.9	2.9	314.7	23.0	23.0	182.6
OpenFly	18.2	18.2	110.5	3.0	3.0	156.9	5.9	5.9	272.6	9.0	9.0	180.9
Reason-WAM	**68.2**	**61.6**	**45.3**	**34.8**	**30.6**	80.4	**20.6**	**20.2**	**126.3**	**41.0**	**37.3**	**84.4**

Table 1: Quantitative results on EmbodiedNav-Bench across short, middle, and long episodes. Higher SR/SPL and lower NE indicate better performance. The best results are marked in red and the second-best results in blue.

et al., 2025b), Qwen-VL-Max and Qwen3-VL-Plus (Cloud, 2025), GLM-4.6V (GLM et al., 2024), Gemini-2.0/2.5 models (Google, 2025), and GPT-4o, GPT-4.1, OpenAI o4-mini, and GPT-5.1 (OpenAI, 2025). It also includes agent-based action-as-reasoning methods adapted to aerial goal navigation, including SayNav (Rajvanshi et al., 2023), STMR (Gao et al., 2024b), and PRPSearcher (Ji et al., 2025). The second group contains trained or navigation-adapted baselines under our cross-validation setting, which we further divide into VLM-based and VLA-based methods. The VLM-based baselines include Qwen3-VL policies that directly output navigation actions. The VLA-based baselines include action-as-token models including OpenFly (Gao et al., 2026), OpenVLA-UAV (Wang et al., 2025b), and Uni-NaVid (Zhang et al., 2025b), which directly learn to predict navigation actions from visual-language inputs. All trainable baselines are trained or fine-tuned on the same training/test splits.

Implementation details. We initialize the VLM planner from Qwen3-VL-8B (Bai et al., 2025a), instantiate the world model from Wan2.1-Fun-V1.1-1.3B-Control-Camera (ignore camera-control and use only its TI2V functionality) (Wan et al., 2025; He et al., 2025), and implement the action decoder following TSFormer-VO (Françani and Maximo, 2025). The TSFormer-VO action decoder is trained on video–pose trajectory segments to regress relative 6-DoF motion with an MSE loss, using Adam with learning rate 1×10^{-5}, per-process batch size 4, and gradient clipping 1.0. The world model is supervised fine-tuned with route-oriented instruction–video pairs relabeled from training trajectories and fine-tuned with AdamW using a learning rate of 1×10^{-5}, weight decay 0.01, bfloat16 mixed precision, and gradient checkpointing. After the WAM is trained, we optimize the VLM planner with multi-turn GRPO in closed-loop navigation roll-

Figure 3: **Qualitative case of VLM action selection in Reason-WAM.** The VLM adaptively chooses among three action types: active perception for acquiring surrounding views, direct primitive actions for short-range local adjustment, and WAM invocation for goal-directed movement.

outs. The GRPO stage uses a trajectory-level advantage estimator, 4 rollouts per episode, training batch size 4, mini-batch size 16, actor learning rate 1×10^{-6}, rollout temperature 1.0, top-p 1.0, and 100 training steps.

4.2 Quantitative Results

Table 1 compares Reason-WAM with zero-shot and trained baselines. Reason-WAM achieves the best average SR and SPL, reaching 41.0% SR and 37.3% SPL, while also obtaining the second-best average NE of 84.4. Compared with the strongest zero-shot baseline in SR/SPL, GPT-5.1, Reason-WAM improves SR from 34.0% to 41.0% and SPL from 32.0% to 37.3%. It also substantially outperforms the strongest trained baseline, Qwen3-VL (SFT), improving SR by 13.0 points, SPL by 10.3 points, and reducing NE from 131.9 to 84.4. These results suggest that neither zero-shot spatial reasoning nor supervised action-token learning is sufficient for reliable aerial goal navigation; effective performance requires coordinated semantic decomposition and metric action grounding.

Across trajectory lengths, Reason-WAM consistently achieves the highest SR and SPL. On short episodes, it reaches 68.2% SR and 61.6% SPL, indicating that WAM-grounded local execution can reliably realize route-level intents within relatively short horizons. On middle episodes, it obtains 34.8% SR and 30.6% SPL, outperforming all baselines under more ambiguous spatial observations. On long episodes, Reason-WAM reaches 20.6% SR and 20.2% SPL, improving over the strongest baseline by 2.4 SR points and 2.2 SPL points, and achieves the lowest NE of 126.3. This shows that coupling VLM planning with WAM execution improves robustness to accumulated reasoning and control errors.

4.3 Case Analysis

Figure 3 illustrates a complete decision process of Reason-WAM in a representative goal-navigation case. Given the sparse goal "Blue board behind," the VLM first determines that the target state cannot be inferred reliably from the initial forward-facing observation alone. It therefore selects an *active perception* action to obtain surrounding views, which provides a broader spatial context for identifying the blue board and reasoning about its behind-view location. After the scene layout becomes clearer, the VLM can issue direct primitive actions for small, locally grounded movements when the next step is visually obvious and does not require long-horizon prediction. Finally, when the agent needs to move toward the target region across a larger spatial gap, the VLM invokes the WAM with a route-level instruction. The WAM predicts short-horizon visual transitions conditioned on the current observation and instruction, and the action decoder converts the imagined transition into executable aerial actions. This case shows that Reason-WAM does not rely on a fixed action interface; instead, the VLM dynamically switches between perception, direct control, and WAM-based execution, enabling the agent to progressively acquire context, make local adjustments, and reach the goal-directed behind-view of the blue board.

4.4 Ablation Study

Table 2 isolates the contribution of the main components. WAM only obtains 26.6% SR, showing that although the executor can ground local instruc-

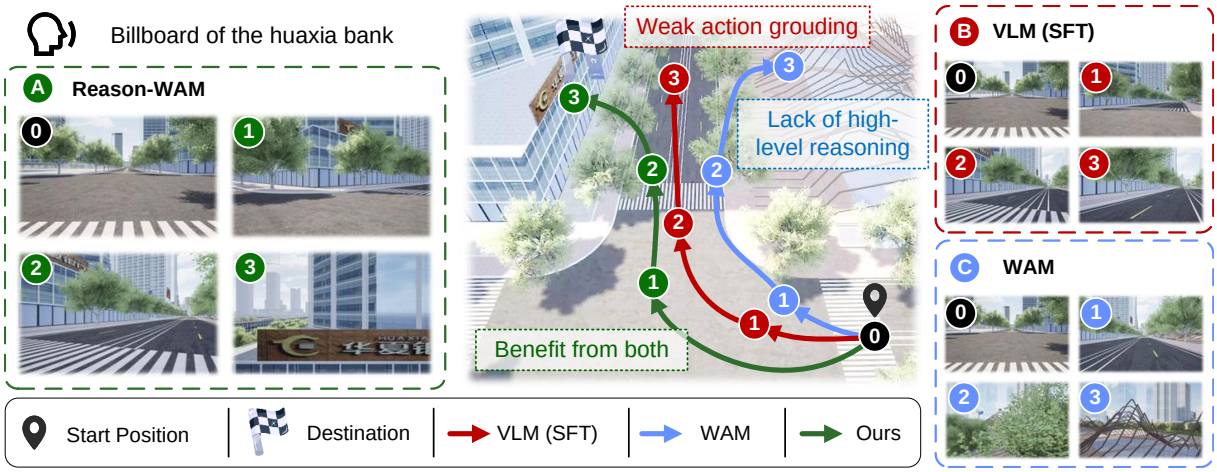

Figure 4: Qualitative comparison of goal-oriented urban aerial navigation. Starting from the same initial position and goal, Reason-WAM successfully reaches the destination by combining high-level spatial reasoning with grounded action execution, whereas the VLM-only and WM-VO-only baselines fail to reach the target. The numbered snapshots denote the temporal order of navigation, and the arrows indicate the corresponding navigation paths.

Setting	SR↑	SPL↑	NE↓
WAM	26.6	26.1	199.6
VLM	15.9	15.5	124.0
VLM (SFT)	28.0	27.0	131.9
VLM (w/o RL) + WAM	32.3	31.4	168.1
Reason-WAM	**41.0**	**37.3**	**84.4**

Table 2: Ablation study on model variants and training stages. Each row reports average SR/SPL/NE. SFT indicates supervised fine-tuning on actions.

tions into motion, it still lacks reliable spatial reasoning ability needed to infer where to go from an open-ended goal. Direct VLM reaches 15.9% SR, while action imitation improves it to 28.0% SR, indicating that VLMs can provide semantic reasoning but still lack reliable action grounding in continuous 3D space. This limitation is difficult to address with supervised imitation because the demonstrations contain state-action trajectories but no reasoning traces, route-step decompositions, or WAM-invocation labels. Adding WAM to the VLM further improves SR to 32.3%, but the higher NE of 168.1 indicates insufficient coordination between high-level decisions and low-level execution: the VLM may invoke the executor at inappropriate times or issue route intents that are locally executable but globally suboptimal. This coordination gap is also reflected qualitatively in Figure 4. Starting from the same initial state and goal, Reason-WAM follows a coherent step-by-step navigation sequence and successfully reaches the destination, whereas the VLM-only baseline makes semantically plausible but weakly grounded decisions, and the WM-VO-only baseline executes locally reasonable motions but lacks the global goal reasoning needed to stay on a correct route. In contrast, Reason-WAM achieves the best overall result with 41.0% SR, 37.3% SPL, and 84.4 NE, showing that the model must still learn when to rely on semantic reasoning and when to invoke WAM for grounded execution and closed-loop RL is necessary for learning selective WAM invocation from task-level feedback.

5 Conclusion

We presented **Reason-WAM**, a hierarchical framework that integrates VLM-based spatial reasoning with WAM-based action grounding for goal-oriented urban aerial navigation. We further introduced a bottom-up training framework that progressively learns visual transition grounding, instruction-conditioned world dynamics, and adaptive VLM planning with multi-turn GRPO. Experiments on the public urban aerial goal-navigation benchmark EmbodiedNav-Bench show that Reason-WAM achieves a state-of-the-art average SR of 41.0%, outperforming existing VLM/VLA methods. Case analysis and ablation studies verify the necessity of combining VLM-based spatial reasoning, WAM-based spatial action grounding, and RL-based coordination. More broadly, Reason-WAM extends the scope of WAMs from short-horizon settings to more complex, long-horizon embodied tasks.

Limitations

This work has two limitations. First, the world model is used without dedicated acceleration or inference optimization, leaving room for improving the overall efficiency of Reason-WAM, particularly in long-horizon episodes that require frequent WAM invocation. Second, the current training pipeline adopts a staged formulation, where the action decoder, world model, and VLM planner are optimized separately. While this design offers a practical bottom-up instantiation of the proposed hierarchy, it does not explicitly optimize the mutual adaptation between VLM-based reasoning and WAM-based execution. Future work may investigate more efficient inference and joint training paradigms that promote tighter coordination between the VLM and WAM.

References

Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton Van Den Hengel. 2018. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3674–3683.

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, and 45 others. 2025a. Qwen3-vl technical report. *Preprint*, arXiv:2511.21631.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, and 1 others. 2025b. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Amir Bar, Gaoyue Zhou, Danny Tran, Trevor Darrell, and Yann LeCun. 2025. Navigation world models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 15791–15801.

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

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

Boyuan Chen, Tianyuan Zhang, Haoran Geng, Caiyi Zhang, Peihao Li, Kiwhan Song, William T. Freeman, Jitendra Malik, Pieter Abbeel, Russ Tedrake, Vincent Sitzmann, and Yilun Du. 2026a. Large video planner enables generalizable robot control. *Preprint*, arXiv:2512.15840.

Jinyu Chen, Hongyu Li, Zongheng Tang, Xiaoduo Li, Wenjun Wu, and Si Liu. 2026b. Aerialvla: A vision-language-action model for aerial navigation with online dialogue. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 18161–18169.

Alibaba Cloud. 2025. Qwen documentation. https://tongyi.aliyun.com/. Accessed: 2025-09-24.

Yilun Du, Sherry Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Josh Tenenbaum, Dale Schuurmans, and Pieter Abbeel. 2023. Learning universal policies via text-guided video generation. *Advances in neural information processing systems*, 36:9156–9172.

Yue Fan, Winson Chen, Tongzhou Jiang, Chun Zhou, Yi Zhang, and Xin Wang. 2023. Aerial vision-and-dialog navigation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 3043–3061.

Lang Feng, Zhenghai Xue, Tingcong Liu, and Bo An. 2025. Group-in-group policy optimization for llm agent training. In *Advances in Neural Information Processing Systems*, volume 38, pages 46375–46408. Curran Associates, Inc.

André O. Françani and Marcos R. O. A. Maximo. 2025. Transformer-based model for monocular visual odometry: A video understanding approach. *IEEE Access*, 13:13959–13971.

Chen Gao, Baining Zhao, Weichen Zhang, Jinzhu Mao, Jun Zhang, Zhiheng Zheng, Fanhang Man, Jianjie Fang, Zile Zhou, Jinqiang Cui, Xinlei Chen, and Yong Li. 2024a. Embodiedcity: A benchmark platform for embodied agent in real-world city environment. *Preprint*, arXiv:2410.09604.

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, and 4 others. 2026. Openfly: A comprehensive platform for aerial vision-language navigation. *Preprint*, arXiv:2502.18041.

Yunpeng Gao, Zhigang Wang, Linglin Jing, Dong Wang, Xuelong Li, and Bin Zhao. 2024b. Aerial vision-and-language navigation via semantic-topo-metric representation guided llm reasoning. *arXiv preprint arXiv:2410.08500*.

Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Dan Zhang, Diego Rojas, Guanyu Feng, Hanlin Zhao, and 1 others. 2024. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *arXiv preprint arXiv:2406.12793*.

Google. 2025. Gemini api. https://ai.google.dev/gemini-api. Accessed: 2025-04-12.

Jing Gu, Eliana Stefani, Qi Wu, Jesse Thomason, and Xin Wang. 2022. Vision-and-language navigation: A survey of tasks, methods, and future directions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7606–7623.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, and 1 others. 2025. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638.

Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. 2025. Cameractrl: Enabling camera control for text-to-video generation. *Preprint*, arXiv:2404.02101.

Xijie Huang, Weiqi Gai, Tianyue Wu, Congyu Wang, Zhiyang Liu, Xin Zhou, Yuze Wu, and Fei Gao. 2026. Navdreamer: Video models as zero-shot 3d navigators. *Preprint*, arXiv:2602.09765.

Vihan Jain, Gabriel Magalhaes, Alexander Ku, Ashish Vaswani, Eugene Ie, and Jason Baldridge. 2019. Stay on the path: Instruction fidelity in vision-and-language navigation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1862–1872.

Yatai Ji, Zhengqiu Zhu, Yong Zhao, Beidan Liu, Chen Gao, Yihao Zhao, Sihang Qiu, Yue Hu, and Quanjun Yin. 2026. Towards autonomous uav visual object search in city space: Benchmark and agentic methodology. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 18342–18350.

Yatai Ji, Zhengqiu Zhu, Yong Zhao, Beidan Liu, Chen Gao, Yihao Zhao, Sihang Qiu, Yue Hu, Quanjun Yin, and Yong Li. 2025. Towards autonomous uav visual object search in city space: Benchmark and agentic methodology. *arXiv preprint arXiv:2505.08765*.

Moo Jin Kim, Yihuai Gao, Tsung-Yi Lin, Yen-Chen Lin, Yunhao Ge, Grace Lam, Percy Liang, Shuran Song, Ming-Yu Liu, Chelsea Finn, and Jinwei Gu. 2026. Cosmos policy: Fine-tuning video models for visuomotor control and planning. *Preprint*, arXiv:2601.16163.

Alexander Ku, Peter Anderson, Roma Patel, Eugene Ie, and Jason Baldridge. 2020. Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4392–4412.

Lin Li, Qihang Zhang, Yiming Luo, Shuai Yang, Ruilin Wang, Fei Han, Mingrui Yu, Zelin Gao, Nan Xue, Xing Zhu, Yujun Shen, and Yinghao Xu. 2026. Causal world modeling for robot control. *Preprint*, arXiv:2601.21998.

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

Arjun Majumdar, Gunjan Aggarwal, Bhavika Devnani, Judy Hoffman, and Dhruv Batra. 2022. Zson: Zero-shot object-goal navigation using multimodal goal embeddings. In *Advances in Neural Information Processing Systems*, volume 35, pages 32340–32352. Curran Associates, Inc.

OpenAI. 2025. Gpt-4o api. https://openai.com/api/. Accessed: 2025-04-12.

Zhangyang Qi, Zhixiong Zhang, Yizhou Yu, Jiaqi Wang, and Hengshuang Zhao. 2025. Vln-r1: Vision-language navigation via reinforcement fine-tuning. *Preprint*, arXiv:2506.17221.

Yanyuan Qiao, Haodong Hong, Wenqi Lyu, Dong An, Siqi Zhang, Yutong Xie, Xinyu Wang, and Qi Wu. 2025. Navbench: Probing multimodal large language models for embodied navigation. In *Advances in Neural Information Processing Systems*, volume 38, pages 95365–95385. Curran Associates, Inc.

Abhinav Rajvanshi, Karan Sikka, Xiao Lin, Bhoram Lee, Han-Pang Chiu, and Alvaro Velasquez. 2023. Saynav: Grounding large language models for dynamic planning to navigation in new environments. *arXiv preprint arXiv:2309.04077*.

Dhruv Shah, Błażej Osiński, Sergey Levine, and 1 others. 2023. Lm-nav: Robotic navigation with large pre-trained models of language, vision, and action. In *Conference on robot learning*, pages 492–504. pmlr.

Xinshuai Song, Weixing Chen, Yang Liu, Weikai Chen, Guanbin Li, and Liang Lin. 2025. Towards long-horizon vision-language navigation: Platform, benchmark and method. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12078–12088.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, and 43 others. 2025. Wan: Open and advanced large-scale video generative models. *Preprint*, arXiv:2503.20314.

Ke Wang, Sai Ma, Junlan Chen, Fan Ren, and Jianbo Lu. 2020. Approaches, challenges, and applications for deep visual odometry: Toward complicated and emerging areas. *IEEE Transactions on Cognitive and Developmental Systems*, 14(1):35–49.

Liuyi Wang, Xinyuan Xia, Hui Zhao, Hanqing Wang, Tai Wang, Yilun Chen, Chengju Liu, Qijun Chen, and Jiangmiao Pang. 2025a. Rethinking the embodied gap in vision-and-language navigation: A holistic study of physical and visual disparities. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9455–9465.

Xiangyu Wang, Donglin Yang, Yue Liao, Wenhao Zheng, wenjun wu, Bin Dai, Hongsheng Li, and Si Liu. 2025b. Uav-flow colosseo: A real-world benchmark for flying-on-a-word uav imitation learning. In *Advances in Neural Information Processing Systems*, volume 38. Curran Associates, Inc.

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

Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, Ayaan Malik, Kyungmin Lee, William Liang, Nadun Ranawaka, Jiasheng Gu, Yinzhen Xu, Guanzhi Wang, Fengyuan Hu, Avnish Narayan, and 17 others. 2026. World action models are zero-shot policies. *Preprint*, arXiv:2602.15922.

Jiazhao Zhang, Anqi Li, Yunpeng Qi, Minghan Li, Jiahang Liu, Shaoan Wang, Haoran Liu, Gengze Zhou, Yuze Wu, Xingxing Li, Yuxin Fan, Wenjun Li, Zhibo Chen, Fei Gao, Qi Wu, Zhizheng Zhang, and He Wang. 2025a. Embodied navigation foundation model. *Preprint*, arXiv:2509.12129.

Jiazhao Zhang, Kunyu Wang, Shaoan Wang, Minghan Li, Haoran Liu, Songlin Wei, Zhongyuan Wang, Zhizheng Zhang, and He Wang. 2025b. Uni-navid: A video-based vision-language-action model for unifying embodied navigation tasks. *Preprint*, arXiv:2412.06224.

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. 2025c. Mem2ego: Empowering vision-language models with global-to-ego memory for long-horizon embodied navigation. *Preprint*, arXiv:2502.14254.

Mike Zhang, Kaixian Qu, Vaishakh Patil, Cesar Cadena, and Marco Hutter. 2024a. Tag map: A text-based map for spatial reasoning and navigation with large language models. *Preprint*, arXiv:2409.15451.

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

Yue Zhang, Tianyi Ma, Zun Wang, Yanyuan Qiao, and Parisa Kordjamshidi. 2025e. Vision-and-language navigation with analogical textual descriptions in llms. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 15028–15036.

Yue Zhang, Ziqiao Ma, Jialu Li, Yanyuan Qiao, Zun Wang, Joyce Chai, Qi Wu, Mohit Bansal, and Parisa Kordjamshidi. 2024b. Vision-and-language navigation today and tomorrow: A survey in the era of foundation models. *Preprint*, arXiv:2407.07035.

Baining Zhao, Ziyou Wang, Jianjie Fang, Zile Zhou, Yanggang Xu, Yatai Ji, Jiacheng Xu, Qian Zhang, Weichen Zhang, Chen Gao, and Xinlei Chen. 2026a. How far are large multimodal models from human-level spatial action? a benchmark for goal-oriented embodied navigation in urban airspace. *Preprint*, arXiv:2604.07973.

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. 2026b. Worldvln: Autoregressive world action model for aerial vision-language navigation. *Preprint*, arXiv:2605.15964.

Gengze Zhou, Yicong Hong, and Qi Wu. 2024. Navgpt: Explicit reasoning in vision-and-language navigation with large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 7641–7649.

Fengda Zhu, Xiwen Liang, Yi Zhu, Qizhi Yu, Xiaojun Chang, and Xiaodan Liang. 2021. Soon: Scenario oriented object navigation with graph-based exploration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12689–12699.

Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, and 1 others. 2023. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, pages 2165–2183. PMLR.

A License For Artifacts

The EmbodiedNav-Bench datasets used in this work are released under the Creative Commons Attribution 4.0 International License (CC BY 4.0), which permits sharing, adaptation, and reuse with appropriate attribution. The pretrained models and other third-party artifacts used in this paper are publicly available and licensed for academic or research use. We comply with the license terms, attribution requirements, and usage restrictions specified by the original providers. We do not use proprietary or privately licensed data. Any released artifacts derived from these resources will retain compatible licensing terms and include appropriate citations and acknowledgments to the original authors.

Our use of these artifacts is consistent with their intended purposes. The datasets used in this work are designed to support research on embodied navigation, visual-spatial reasoning, and goal-conditioned decision making in simulated environments. Accordingly, we use them for benchmarking and analyzing navigation algorithms under controlled experimental settings. The proposed Reason-WAM framework is also aligned with the original purpose of the benchmark: it aims to improve goal-oriented navigation by combining high-level reasoning with grounded action execution, rather than repurposing the data for unrelated applications. We do not use the datasets or models for surveillance, identification, real-world deployment without validation, or any other use beyond academic evaluation of embodied navigation methods.

B Information About Use of AI Assistants

In the preparation of this manuscript, AI Assistant was used solely for writing support, including language polishing and expression refinement.

C Details of Method

C.1 Vision Language Model

After training, the VLM planner operates as the high-level decision module of Reason-WAM. At each navigation step, it receives the sparse natural-language goal, the current egocentric observation, recent interaction history, and optional feedback from the previous step. Following the original problem setting, the planner reasons over an egocentric visual context and may optionally activate a compact six-view rendering for additional spatial evidence when needed. The six-view context, when used, is rendered from the drone's current position in a fixed order: `front`, `down`, `left`, `back`, `right`, and `up`. This order is kept fixed so that the planner can associate each image index with a stable egocentric direction. In addition, the planner is given a compact context: up to three recent history images are appended when available, and each history image is paired with a textual record of the form `selected_view=<view>` and `command=<command or None>`. This history allows the VLM to remember which local evidence was used in recent decisions without requiring a full trajectory memory.

The planner does not output a free-form paragraph or a JSON object. Instead, it follows a structured text protocol with the sections `Thinking`, `Plan`, `Use_Six_View`, `Use_WM_VO`, `Selected_View`, `WM_VO_Prompt`, and `Command`. The `Thinking` and `Plan` fields provide a concise rationale and immediate plan, while the final five fields define the executable interface. The planner first decides whether the six-view visual context should be used for the current step, and then decides whether to invoke the WAM executor. The compact inference-time prompt used to enforce this interface is summarized below:

> You are directly controlling a drone in a goal-oriented navigation setting. The navigation goal is given in natural language. At each step, you may optionally choose to reason with a compact six-view egocentric observation rendered at the drone's current position. When used, the six views follow a fixed order: Image 1: front, Image 2: down, Image 3: left, Image 4: back, Image 5: right, and Image 6: up. Recent history images may also be provided when available.
>
> Default to a direct primitive command. Use WM_VO only for near-target final refinement when the actual target or control surface is clearly visible, close enough for fine positioning, occupies a meaningful part of the selected image, and a single 10-meter primitive command is not precise enough. Do not use WM_VO for exploration, scanning, broad approach, normal progress, or distant visible targets. If one primitive command can make reasonable progress, use the primitive command.
>
> Output exactly the following sections: Thinking, Plan, Use_Six_View, Use_WM_VO, Selected_View, WM_VO_Prompt, and Command. If `Use_Six_View` is `No`, reason only over the current egocentric observation and history. If `Use_Six_View` is `Yes`, use the six-view observation in the fixed order above. If `Use_WM_VO` is `No`, set `WM_VO_Prompt` to `None` and output exactly one primitive command. If `Use_WM_VO` is `Yes`, set `Use_Six_View` to `Yes`, choose one selected view from `front`, `down`, `left`, `back`, `right`, and `up`,

write one executable WM_VO instruction, and set Command to None.

The primitive command set is move_forward, move_back, move_left, move_right, move_up, and move_down. Each primitive corresponds to a fixed 10-meter translation in the drone-local frame: forward/backward update the longitudinal coordinate, left/right update the lateral coordinate, and up/down update the vertical coordinate. For history construction, primitive decisions are also associated with the view most aligned with the executed motion: move_forward maps to front, move_back to back, move_left to left, move_right to right, move_up to up, and move_down to down. This mapping makes the stored history images semantically consistent with both primitive execution and WAM invocation.

C.2 World Action Model

At inference time, the WAM executor is invoked only when the VLM planner outputs Use_WM_VO: Yes. It is therefore a selective local grounding module rather than an always-on controller. The executor receives only the selected view image and the planner's WM_VO_Prompt; it does not receive the full six-view observation. This design forces the VLM to specify which view contains the most relevant local control surface and to translate its high-level navigation reasoning into a concrete local instruction.

The WAM executor contains a short-horizon visual world model and an action decoder. Given the selected image and the local instruction, the world model predicts a short visual rollout that represents the intended local transition. The action decoder then compares the current image with the predicted rollout and recovers an executable relative trajectory. The simulator executes this trajectory and returns a new embodied observation to the VLM planner, closing the loop for the next decision step. Thus, the WAM is responsible for grounding local route instructions into metric motion, while the VLM remains responsible for long-horizon goal reasoning, deciding whether WAM execution is needed, selecting the view, and writing the local instruction.

Before passing an instruction to the WAM, Reason-WAM wraps the planner output with the selected-view metadata:

```
The selected image is Image {index}
({selected_view}). Interpret this image
as the local control frame for WM_VO.
{WM_VO_Prompt}
```

This wrapper makes the control frame explicit. For horizontal views, namely front, left, back, and right, the decoded relative motion is interpreted in the yaw frame of the selected view. For vertical views, namely up and down, the trajectory is interpreted in the current horizontal yaw frame, because these views primarily provide vertical or top-down local alignment cues rather than a new heading direction. In all cases, Reason-WAM preserves the current yaw during execution and uses the decoded trajectory only to update the local position. This convention keeps WAM execution compatible with the VLM's egocentric six-view reasoning and prevents the local executor from taking over the long-horizon orientation policy.

D Details of EmbodiedNav-Bench

EmbodiedNav-Bench is a goal-oriented embodied navigation benchmark in urban airspace. Unlike conventional aerial VLN tasks that provide step-by-step route instructions, this benchmark only provides a sparse natural-language goal, such as a target shop, building entrance, billboard, or rooftop structure. The agent is required to continuously observe the environment from an egocentric perspective, understand the target, infer spatial relations, plan its route, and execute three-dimensional flight actions. Therefore, EmbodiedNav-Bench evaluates not only visual-language understanding, but also long-horizon spatial reasoning, active perception, and closed-loop decision making in open urban environments. The benchmark is built upon the EmbodiedCity simulator, which is constructed from real-world city environments and contains rich urban elements such as buildings, roads, shops, signs, and rooftop structures, providing a complex and reproducible urban aerial testbed.

Another important advantage of EmbodiedNav-Bench lies in its strong 3D characteristics and high-quality human-collected trajectories. The dataset contains 5,037 navigation trajectories, each consisting of a start point, a target location, a natural-language goal instruction, and a corresponding human flight trajectory. These trajectories are collected by experienced UAV operators according to the goal instructions, making them closer to realistic goal-directed UAV navigation behavior. In terms of action distribution, the benchmark is not limited to horizontal movement, but also includes

vertical movement and rotation/viewpoint adjustment. In particular, vertical actions account for a substantial portion of the data, enabling the benchmark to evaluate a model's ability to handle altitude changes, viewpoint shifts, local geometry, and three-dimensional spatial relations.

These properties make EmbodiedNav-Bench well suited for evaluating Reason-WAM. The core objective of Reason-WAM is to combine the high-level spatial reasoning ability of a VLM with the local action grounding ability of a WAM. EmbodiedNav-Bench naturally requires both capabilities: the sparse goal instruction demands goal understanding and long-horizon route decomposition, while the continuous urban aerial environment requires fine-grained three-dimensional action execution. In other words, this benchmark can expose the limitations of pure VLM-based methods in metric spatial awareness and executable control, while also testing whether short-horizon executors alone are insufficient for open-ended goal reasoning. Therefore, it provides a challenging and well-aligned testbed for validating whether Reason-WAM can bridge goal-level reasoning and executable 3D action.

E Details of Baseline Setting

We compare Reason-WAM with three trainable vision-language-action baselines, namely Uni-NaVid, OpenVLA-UAV, and OpenFly. For fair comparison, all trainable baselines are trained or fine-tuned on the same training split and evaluated on the same test split of EmbodiedNav-Bench. All methods operate on the same image root and are evaluated in the same closed-loop simulator with identical success criteria and metric computation.

For Uni-NaVid, we preserve its native action formulation and training interface. Following the original Uni-NaVid setup, each sample consists of a history video clip together with a language instruction, and the model predicts a short sequence of navigation actions in its native action vocabulary. The input videos are constructed from the egocentric image trajectory, and the model is fine-tuned from the public Uni-NaVid checkpoint using its official video-based training entry point. This baseline therefore remains closest to the original Uni-NaVid design, with adaptation mainly limited to dataset conversion and benchmark integration.

For OpenVLA-UAV, we preserve its native continuous control interface. The model takes the current observation, state information, and the language goal as input, and predicts low-level continuous motion commands in its original action parameterization. In our implementation, the supervised target is represented as local relative motion and yaw change, matching the native output format of the OpenVLA-UAV action head. The model is fine-tuned with LoRA from the public OpenVLA-UAV checkpoint, and closed-loop rollout is performed by feeding the predicted native controls into the benchmark execution pipeline. This setup keeps OpenVLA-UAV in its original continuous-action regime rather than converting it into a text-action model.

For OpenFly, we also follow its native action-prediction interface rather than treating it as a free-form text-generation model. OpenFly originally takes a three-image visual stack as input, so we preserve this design by using the current view together with two history views as the visual input. The model predicts native low-level action outputs through its original action head, and these predictions are executed in the same closed-loop environment as the other baselines. We use LoRA fine-tuning on top of the public OpenFly checkpoint and keep the three-image history-aware observation format throughout both training and evaluation, which is closer to the original OpenFly design than a single-frame or open-ended text-decoding setup.

All trainable baselines are integrated into the same evaluation framework and are rolled out in closed loop under identical benchmark conditions. At each step, the policy receives the current egocentric observation and the goal instruction, predicts an action using its native output interface, executes the action in the simulator, and receives the next observation. We report Success Rate (SR), Success weighted by Path Length (SPL), and Navigation Error (NE) for all baselines.

F Navigation Cases

Figure 5 presents several representative navigation cases of Reason-WAM on EmbodiedNav-Bench. These cases cover a variety of urban aerial target types, including a building with distinctive yellow-edged appearance, a rooftop helipad located in the forward overhead direction, the white-colored roof of a blue tower, the atop region of a road overpass, and the center of a crossroad with surrounding green trees. Notably, these goal instructions are provided only in the form of sparse natural-

language descriptions of target regions or target attributes, without any step-by-step action supervision. Therefore, the model must understand the target semantics from continuous egocentric observations, infer the spatial relationship between the target and the current view, and progressively approach the goal in a closed-loop interaction process.

The navigation trajectories in these examples further illustrate that Reason-WAM can maintain relatively stable goal-directed behavior across diverse urban structures. For example, in the case of locating the yellow-edged building, the model needs to identify local visual attributes of the building and continuously adjust its flight direction as the viewpoint changes; in the cases involving the rooftop helipad and the white roof of the blue tower, the model needs to handle altitude variation, rooftop localization, and local geometric relationships; in the cases related to the overpass and the crossroad, the model must further reason over urban spatial cues such as road layouts, bridge structures, and surrounding greenery. These examples indicate that EmbodiedNav-Bench evaluates not only semantic recognition, but also cross-view spatial consistency, three-dimensional localization, and continuous action execution.

These qualitative results also highlight the effectiveness of the hierarchical design of Reason-WAM. Given sparse goal instructions, the VLM planner is responsible for goal understanding, scene analysis, and local route-intent generation, while the WAM executor further transforms route-level intents into executable three-dimensional flight actions. By decoupling high-level spatial reasoning from local action grounding, Reason-WAM is able to progressively reduce the distance to the target region under different target attributes and urban structures, thereby demonstrating strong closed-loop navigation capability in complex urban aerial environments.

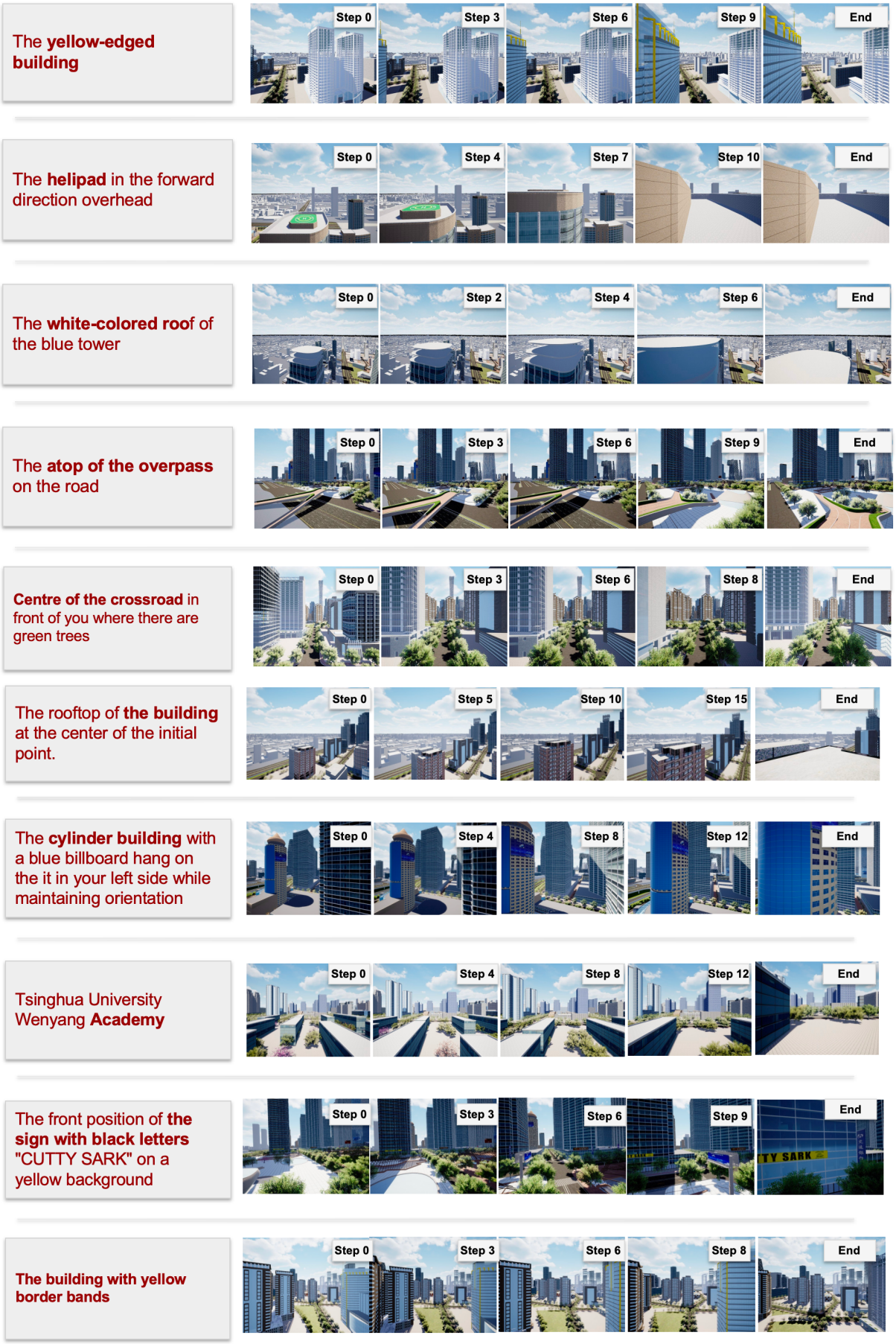

Figure 5: Qualitative navigation cases of Reason-WAM on EmbodiedNav-Bench. Each row shows a goal instruction and the corresponding egocentric observations at representative steps during navigation.