DETRACK: A BENCHMARK AND ALTITUDE-AWARE DUAL WORLD MODEL FOR DRONE-EMBODIED TRACKING

Anonymous authors
Paper under double-blind review

ABSTRACT

Aerial object tracking has wide applications in public safety, emergency rescue, wildlife monitoring, etc. However, existing aerial object tracking is mainly limited to the passive tracking paradigm with 2D video sequences pre-recorded in fixed camera locations or along pre-defined flight paths, lacking drone-embodied scene perception, egocentric interaction, and movement control in dynamic 3D scenes. In this paper, we define a novel drone-embodied tracking (DeTrack) task and build a large-scale benchmark with evaluation metrics for this task. It refers to tracking targets in interactive 3D scenes with online drone-egocentric observation and active control in a closed loop. The DeTrack benchmark differs from conventional passive aerial tracking in five characteristics: (1) drone-embodied closed-loop tracking, (2) target tracking coupled with obstacle avoidance, (3) spatio-temporally composite occlusion in 3D scenes, (4) altitude-mediated contradiction between visibility and safety, (5) extensive and extreme scale variation. Besides, the drone-embodied tracking faces intrinsic altitude-mediated contradiction between visibility and safety: the view field is wide while the target detail is weak at high flight altitude, and the target detail is sufficient while the obstacles are dense at low flight altitude. Therefore, we further propose a novel altitude-aware dual worlds (AaDWorlds) for drone-embodied tracking. It consists of an altitude-aware perception (AaP) module and dual world models (DWM). Specifically, the altitude-aware perception module adaptively encodes, couples, and decodes altitude-aware representations. The dual world models imagine future states under both high- and low-altitude regimes. Eventually, the imagined future states (from DWM) and pseudo egocentric observations (from AaP) at both high and low altitudes effectively complement the original drone-egocentric observation, simultaneously facilitating the view field, target detail, and obstacle avoidance during drone-embodied tracking.

1 INTRODUCTION

Object tracking has wide applications in the fields of emergency rescue, traffic inspection, ecological protection, anti-drone confrontation, and so on (Wu et al., 2022; Azar et al., 2021; Chan et al., 2024; Liu et al., 2022). With the rapid growth of low-altitude economy and unmanned aerial vehicles (UAVs), aerial object tracking has achieved significant progress. Representative aerial object tracking benchmarks, such as UAV123 (Mueller et al., 2016), UAVDT (Du et al., 2018), and VisDrone (Zhu et al., 2018; 2019), have promoted standardized evaluation on aerial videos. The classical aerial object tracking approach has evolved from correlation-filter (Henriques et al., 2015), Siamese-based trackers (Bertinetto et al., 2016; Li et al., 2019), and more recent Transformer-based trackers (Chen et al., 2021; Yan et al., 2021; Ye et al., 2022).

However, existing works and benchmarks for aerial object tracking are mainly confined to the passive tracking paradigm (Fig. 1a), where object trackers are evaluated on 2D aerial video sequences recorded from fixed locations or along pre-defined flight paths. As a result, drones are merely treated as passive cameras without active scene perception, motion control, and environment interaction. However, the real-world operating environment for drones is typically wide, dynamic, and complicated (obstacle-dense, occlusion-frequent), which requires active abilities of environment perception and movement control and urgently calls for a new drone-embodied tracking paradigm (Fig. 1b).

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.

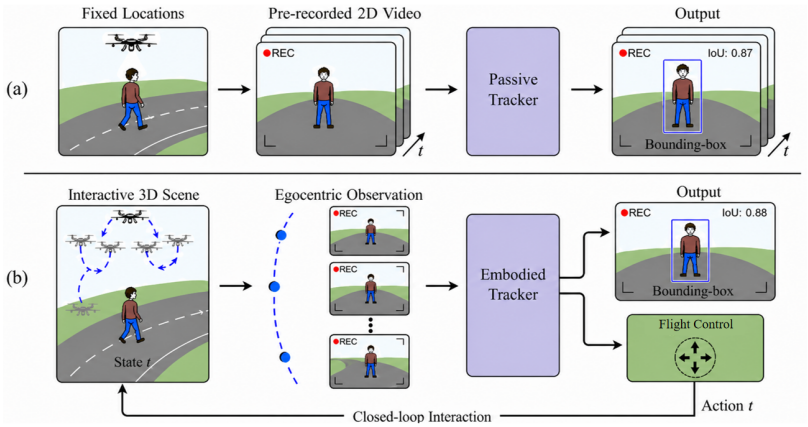


Figure 1: Paradigm comparison between passive aerial tracking and drone-embodied tracking. (a) Passive tracking relies on pre-recorded aerial videos with fixed or pre-defined camera trajectories. (b) Drone-embodied tracking performs online target tracking and flight control through active interaction with a 3D environment.

To activate and advance the research and application of real-world drone object tracking, we define a novel drone-embodied tracking task and build the first large-scale benchmark with evaluation metrics for this task. The DeTrack benchmark is built in an interactive 3D engine and contains 11,368 target trajectories with an average length of 80.09 m for tracking. The drone-embodied tracking task requires the embodied drone to track these targets in interactive 3D scenes with online egocentric observation and active movement control. As illustrated in Fig. 2, the DeTrack task and benchmark differ from conventional passive aerial tracking by five characteristics. (1) *drone-embodied closed-loop tracking*, where historical actions shape future observations and the observations shape the following action in turn; (2) *target tracking coupled with obstacle avoidance*, which requires simultaneous target following and safe navigation; (3) *spatio-temporally composite occlusion in 3D scenes*, arising from both static scene structures and dynamic distractors; (4) *altitude-mediated contradiction between visibility and safety*, which couples target observability with obstacle interactions; (5) *extensive and extreme scale variation*, caused by continuous altitude changes during flight. Taken together, these task characteristics make DeTrack a challenging task and benchmark in interactive 3D environments.

Besides, the 3D drone-embodied tracking task faces an intrinsic altitude-mediated contradiction of visibility and safety. Specifically, at high flight altitudes, the drone enjoys a wider view and fewer obstacle interactions, but the target becomes small and visually ambiguous. At low flight altitudes, the target appears with richer details and better visibility, but the drone faces denser obstacles, more frequent occlusions, and higher collision risk. To alleviate this intrinsic contradiction, we further propose a novel altitude-aware dual worlds framework for drone-embodied tracking (AaDWorlds), which consists of an altitude-aware perception module (AaP) and dual world models (DWM). Specifically, the AaP is designed to encode, couple, and decode altitude-aware observation representations, while the DWM imagines future states in both high- and low-altitude regimes. During inference, with the aid of imagined future states from DWM and pseudo egocentric observations from AaP at both high and low altitudes, our AaDWorlds simultaneously facilitates the view field, target detail, and obstacle avoidance, effectively alleviating the intrinsic altitude-mediated contradiction between visibility and safety during drone-embodied tracking.

In summary, the main contributions of this paper are 4-folds. (1) We define a novel drone-embodied tracking task that tracks targets in interactive 3D scenes with online egocentric observation and active movement control, which differs from conventional passive aerial tracking in five characteristics. (2) We build the first large-scale benchmark and corresponding evaluation metrics for drone-embodied tracking, termed DeTrack, which provides an initial benchmark and evaluation protocols to activate and advance this task. (3) We propose a novel altitude-aware dual worlds approach for the drone-embodied tracking task, which effectively tackles the intrinsic altitude-mediated contradiction between visibility and safety in closed-loop drone-embodied tracking. (4) The proposed AaDWorlds approach achieves promising performance on the DeTrack benchmark across all evaluation metrics, demonstrating the effectiveness and superiority of our approach.

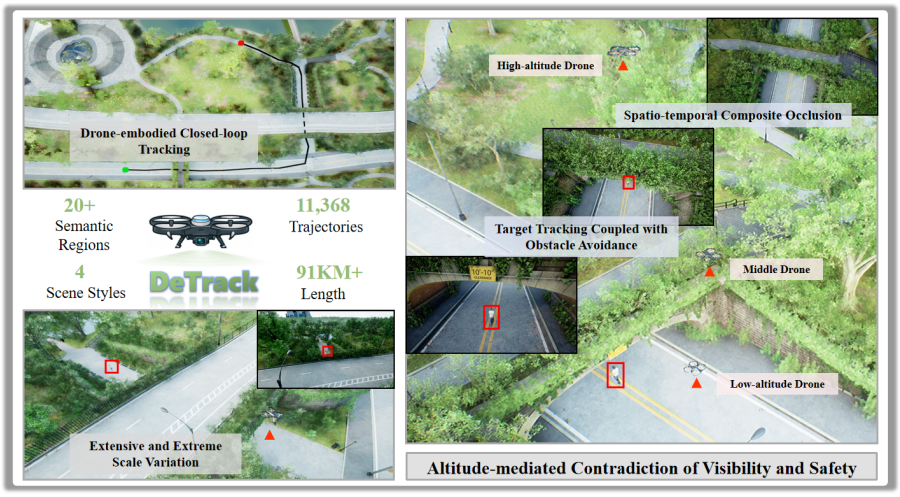


Figure 2: Overview statistics and typical challenges of the proposed drone-embodied tracking (DeTrack) benchmark.

2 Related Works

2.1 Passive Drone Tracking

Most existing drone object tracking studies are based on passive and frame-based benchmarks. Representative datasets such as UAV123 (Mueller et al., 2016), UAVDT (Du et al., 2018), and VisDrone (Zhu et al., 2018; 2020) provide pre-recorded aerial videos for offline tracking evaluation, while recent benchmarks further improve scene scale and realism (Wang et al., 2023; Wu et al., 2022; Zhao et al., 2022). On these benchmarks, drone tracking methods have evolved from correlation-filter and Siamese-based trackers (Henriques et al., 2015; Bertinetto et al., 2016; Li et al., 2019) to recent Transformer-based trackers such as TransT (Chen et al., 2021), STARK (Yan et al., 2021), and OSTrack (Ye et al., 2022). Although these methods achieve strong target localization performance, they are mainly designed for passive tracking on prerecorded videos, where the camera trajectory is predetermined and the drone does not actively interact with the environment. Consequently, they do not explicitly consider how drone motion and viewpoint changes affect future target visibility and tracking performance.

2.2 Embodied Perception

Embodied perception enables an agent to actively adjust its motion or viewpoint for improved perception and decision-making. In object tracking, prior works have explored active tracking with learned camera control (Luo et al., 2018), and subsequent studies extend this paradigm to 3D environments and aerial platforms using reinforcement learning for drone navigation and vision-based target following (Nguyen et al., 2023; Huang et al., 2019; Azar et al., 2021; Feng et al., 2025). Meanwhile, high-fidelity simulators such as Habitat (Savva et al., 2019; Kolve et al., 2017; Xia et al., 2018) and AirSim-based platforms (Shah et al., 2018b; Krishnan et al., 2021; Chan et al., 2024) provide interactive environments for navigation, obstacle avoidance, and active perception. However, existing embodied benchmarks mainly focus on navigation, exploration, or short-horizon active tracking, and seldom provide a standardized drone-specific tracking setting with unified closed-loop evaluation.

3 Drone-embodied Tracking

3.1 Task Formulation

Conventional passive aerial object tracking tracks the target in pre-recorded 2D video sequences at fixed camera locations or along pre-defined flight paths. In contrast, the proposed drone-embodied tracking task (DeTrack) tracks targets in large interactive 3D scenes with online drone-egocentric observations $(o_{t-K+1:t})$ and active flight control in a closed loop. Specifically, assume that a target moves along a trajectory (τ) in an interactive 3D scene. The drone-embodied agent exploits the initial target template (x^{tmp}) and online egocentric observations $(o_{t-K+1:t})$ to continuously control the drone, follow the target, and predict the target bounding box (B_t^{trk}). It is a challenging task that involves scene perception, flight control, passive tracking, etc. To evaluate drone-embodied tracking

under closed-loop interaction, we use four metrics: visible rate (VR), mean IoU (mIoU), tracking rate (TR), and trajectory success rate (SR), as summarized in Table 1. Detailed definitions of the metrics, notation, and thresholds are provided in Appendix B.2.

Table 1: Evaluation metrics for the drone-embodied tracking task.

Metric	Definition	**Metric**	Definition
VR	$\frac{1}{T}\sum_{t=1}^{T}\mathbb{I}\big(p_t^{\text{vis}} \geq \tau_{\text{vis}}\big)$	**mIoU**	$\frac{1}{T}\sum_{t=1}^{T}\text{IoU}_t$
TR	$\frac{1}{T}\sum_{t=1}^{T}\mathbb{I}\big(p_t^{\text{vis}} \geq \tau_{\text{vis}}, \text{IoU}_t > \tau_{\text{trk}}\big)$	**SR**	$\frac{1}{N_{\text{traj}}}\sum_{j=1}^{N_{\text{traj}}}\mathbb{I}\Big(d_{xy,T_j}^{(j)} \leq \tau_d\Big)$

3.2 DeTrack Benchmark

The DeTrack benchmark for drone-embodied tracking is built in an interactive Unreal Engine 4.27 simulator (Epic Games, 2021) with the AirSim plugin (Microsoft Research, 2021; Shah et al., 2018a). It contains 11,368 target trajectories with an average length of 80.09 m, covering diverse motion scales and tracking horizons in interactive 3D environments. The benchmark contains four representative scenes, including Town, Rural, Park, and City, and provides diverse scene layouts, semantic regions, rendering conditions, and moving distractors for closed-loop drone-embodied tracking. Detailed environment configurations and trajectory statistics are provided in Appendix A.

3.3 DeTrack Challenges

The DeTrack benchmark is tailored for closed-loop drone-embodied tracking in dynamic 3D environments. In contrast to conventional passive aerial tracking, it introduces several task and data challenges (as illustrated in Fig. 2). (1) Drone-embodied closed-loop tracking: unlike passive object tracking benchmarks, the observations and viewpoints (locations, angles, etc.) of the drone-embodied agent are non-stationary. Its historical actions shape its future egocentric observations, making long-horizon following and target re-acquisition much more challenging. (2) Target tracking coupled with obstacle avoidance: the drone-embodied agent must simultaneously coordinate target tracking and obstacle avoidance, i.e., keep the target observable and followed while avoiding collisions with surrounding obstacles for safety constraints. (3) Spatio-temporally composite occlusion: complicated occlusions are composed of static elements (such as buildings, vegetation, and traffic infrastructure) and dynamic entities (vehicles and pedestrians). These occlusions exhibit a mixed nature in spatial visibility and temporal duration, ranging from transient partial obstructions to abrupt and complete disappearance, as well as sustained visibility interruptions over extended durations. (4) Altitude-mediated contradiction of visibility and safety: the drone flight altitude mediates the difficulty of target perception and obstacle avoidance, inducing an intrinsic contradiction between visibility and safety. In detail, flying at high altitude provides wider views and fewer obstacles, but weakens target visibility and details. In contrast, flying at low altitude provides richer target details and better visibility, but faces more occlusion and obstacles. (5) Extensive and extreme scale variation: the target sizes in egocentric observations change with the drone altitude substantially and continuously. Low-altitude observations provide relatively large targets, while high-altitude observations even shrink the target to only a few pixels. See Appendix A for more challenge details.

4 Method

4.1 Overview

As shown in Fig. 3, the overall pipeline of our method consists of a reinforcement drone-embodied tracking (ReDeT) baseline and the altitude-aware dual worlds (AaDWorlds). Specifically, we formulate the drone-embodied tracking task into a reinforcement decision-making process (ReDeT). The ReDeT takes the egocentric drone observations to simultaneously predict the bounding-box of target via a conventional passive tracker and the flight action of drone via a reinforcement learning agent. Besides, the AaDWorlds consists of dual world models (DWM) and an altitude-aware perception (AaP) module. The DWM is equipped with a high-altitude world model and a low-altitude world model, which predict auxiliary imagined future observation features under both high- and low-altitude regimes. The AaP adaptively encodes, couples, and decodes altitude-aware representations at different altitudes. As a result, the imagined future states from DWM and pseudo egocentric observations from AaP at both high and low altitudes complement original drone-egocentric observation and provide more comprehensive clues for current decision-making. The auxiliary clues at

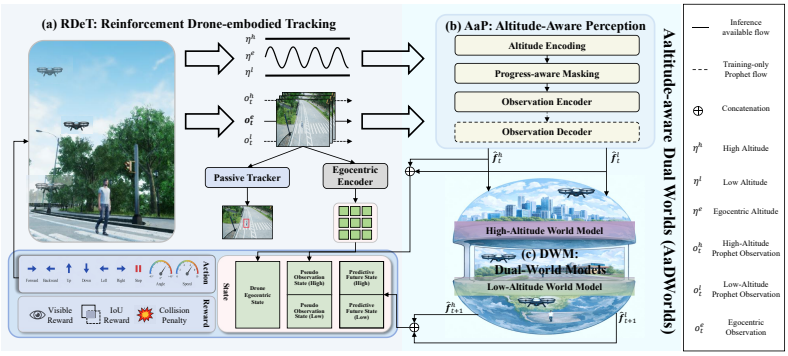


Figure 3: Overview of the proposed AaDWorlds framework, consisting of the ReDeT baseline, altitude-aware perception (AaP), and dual world models (DWM). AaP provides pseudo altitude-aware observations, while DWM predicts future states for closed-loop tracking.

high and low altitudes simultaneously facilitate the view field, target detail, and obstacle avoidance, thus effectively alleviating the intrinsic altitude-mediated contradiction between visibility and safety.

To construct the dual world models (DWM), we always accompany the *real* egocentric observation ($\boldsymbol{o}_t^e$) with a *prophet* high-altitude observation (o_t^h) and a *prophet* low-altitude observation (o_t^l) respectively at high and low altitudes (η^h and η^l) observation during training stage, while we remove prophet observations during inference stage. As shown in Fig. 3, the dashed line indicates training-only observations/structure and will be padded/removed during inference. Therefore, the (AaP) module is proposed to adaptively encode and couple altitude-aware representations at different altitudes, which will be used to decode *pseudo* low-altitude and high-altitude representations ($\hat{\boldsymbol{f}}_t^h$ and $\hat{\boldsymbol{f}}_t^l$) during inference. In the following, we will introduce the details of ReDeT, DWM and AaP.

4.2 Reinforcement Drone-embodied Tracking (ReDeT)

In this paper, we formulate the drone-embodied tracking into a technical paradigm of decision-making process and construct a reinforcement baseline that combines reinforcement learning with passive tracking, termed reinforcement drone-embodied tracking (ReDeT). As shown in Fig. 3a, the passive tracker tracks the target in current drone egocentric observations o_t. Simultaneously, the drone agent takes the latest K drone egocentric observations $o_{t-K+1:t}$ and encodes them into the drone egocentric state (s_t^e). It predicts the next flight action (a_t) according to the current state (s_t). The action space includes 7 discrete movement commands (forward, backward, upward, downward, leftward, rightward, and stop) and two continuous adjustment commands for yaw-angle and speed, respectively. Thereafter, the drone agent moves in interactive 3D scene according to the flight action (a_t), and the spatial position and egocentric observation of drone agent refreshes accordingly.

The total reward of the ReDeT consists of 3 parts in consideration of bounding-box accuracy, target visibility, and collision punishment, *i.e.*, $r = r_{\text{iou}} + \lambda_{\text{vis}} r_{\text{vis}} + \lambda_{\text{col}} r_{\text{col}}$. Specifically, $r_{\text{iou}} = \text{IoU}_t$, which is the IoU between the predicted and ground-truth bounding-boxes. $r_{\text{vis}} = v_t$, which is the visibility status of the target in current drone egocentric observation. $r_{\text{col}} = -\mathbb{I}(\Delta c_t = 1)$, which reflects the punishment for unwished collision, where $c_t \in \{0, 1\}$ indicates collision status and $\Delta c_t = 1$ indicates a collision-rising event. λ_{vis} and λ_{col} are degree controlling hyperparameters. Eventually, we adopt the above ReDeT as baseline framework for the proposed drone-embodied tracking task, and enhance it with auxiliary pseudo observation state ($\hat{\boldsymbol{f}}_t^h$ and $\hat{\boldsymbol{f}}_t^l$) and predictive future state ($\hat{\boldsymbol{f}}_{t+1}^h$ and $\hat{\boldsymbol{f}}_{t+1}^l$) during inference.

4.3 Dual World Models (DWM)

To alleviate intrinsic altitude-mediated contradiction of visibility and safety in the drone-embodied tracking task, we build dual world models (DWM) to respectively simulate low-altitude and high-altitude evolution dynamics (Fig. 4). During inference, they are utilized to simultaneously imagine future states ($\hat{\boldsymbol{f}}_{t+1}^h, \hat{\boldsymbol{f}}_{t+1}^l$) at both low and high flight altitudes serving as auxiliary state clues. Specifically, the DWM consists of two altitude-specific world models ($\mathcal{W}^h$ and $\mathcal{W}^l$) for the high- and low-altitude evolution dynamics, respectively. For each altitude regime $j \in \{h, l\}$, the world

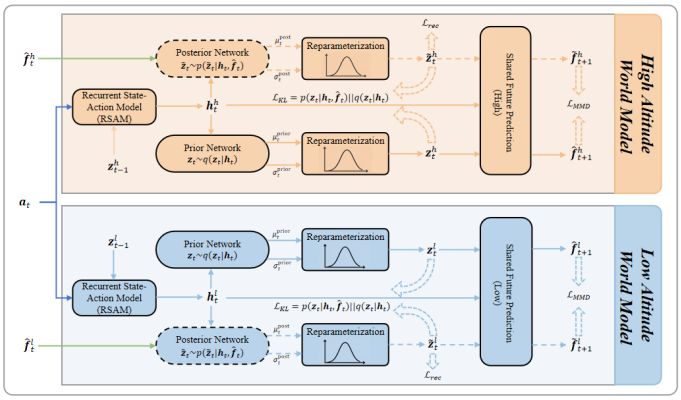


Figure 4: Detailed structure of the proposed Dual World Models (DWM). The high- and low-altitude world models learn altitude-specific dynamics through recurrent state-action modeling and future prediction for closed-loop drone-embodied tracking.

model $\mathcal{W}^j$ consists of a recurrent state-action model (RSAM), a posterior network, a prior network, and a future prediction head. Specifically, each model $\mathcal{W}^j$ maintains a recurrent hidden state $\boldsymbol{z}_t^j$ and recurrent output state $\boldsymbol{h}_t^j$ to summarize the historical state-action dynamics for world modeling and future prediction. Given the current action $\boldsymbol{a}_t$ and the observation feature $\hat{\boldsymbol{f}}_t^j$ at altitude regime j, the posterior network evolves via both the observation feature $\hat{\boldsymbol{f}}_t^j$ and recurrent output state $\boldsymbol{h}_t^j$, while the prior network evolves purely via recurrent output state $\boldsymbol{h}_t^j$, *i.e.*,

$$\boldsymbol{h}_t^j = \mathcal{G}^j(\boldsymbol{z}_{t-1}^j, \boldsymbol{a}_t), \quad \tilde{\boldsymbol{z}}_t^j \sim p^j(\tilde{\boldsymbol{z}}_t^j \mid \boldsymbol{h}_t^j, \hat{\boldsymbol{f}}_t^j), \quad \boldsymbol{z}_t^j \sim q^j(\boldsymbol{z}_t^j \mid \boldsymbol{h}_t^j) \tag{1}$$

where $\mathcal{G}^j(\cdot)$ denotes the recurrent state-action transition function. For simplicity, we assume the recurrent hidden state is in a form of Gaussian distribution. The posterior network (or prior network) predicts the mean and variance (μ, σ) of its recurrent hidden state, and then generates a state from its distribution with the reparameterization trick (Kingma & Welling, 2014). Finally, the shared future prediction head ($\mathcal{D}^j(\cdot)$) imagines corresponding future state feature $(\tilde{\boldsymbol{f}}_{t+1}^j, \hat{\boldsymbol{f}}_{t+1}^j)$ the recurrent hidden state and recurrent output state, *i.e.*,

$$\tilde{\boldsymbol{f}}_{t+1}^j = \mathcal{D}^j(\boldsymbol{h}_t^j, \tilde{\boldsymbol{z}}_t^j), \quad \hat{\boldsymbol{f}}_{t+1}^j = \mathcal{D}^j(\boldsymbol{h}_t^j, \boldsymbol{z}_t^j). \tag{2}$$

During training stage, the posterior network is trained to model evolution dynamics at altitude regime j with external action-observation pairs, which is optimized by approaching prophet observation future $\boldsymbol{f}_{t+1}^j$. The prior network is trained to mimic the evolution dynamics in the posterior network whose input is from external real-world action-observation pairs. During inference stage, the posterior network branch is removed and the prior network will only evolve with the external action input and historical hidden state without external observation feature anymore. The optimization objective for altitude regime j is as follows:

$$\mathcal{L}_{\mathrm{DWM}}^j = \mathrm{KL}\Big(p^j(\tilde{\boldsymbol{z}}_t^j \,|\, \boldsymbol{h}_t^j, \hat{\boldsymbol{f}}_t^j) \,\|\, q^j(\boldsymbol{z}_t^j \,|\, \boldsymbol{h}_t^j)\Big) + \lambda_{\mathrm{mmd}}\mathrm{MMD}\Big(\tilde{\boldsymbol{f}}_{t+1}^j, \hat{\boldsymbol{f}}_{t+1}^j\Big) + \lambda_{\mathrm{rec}}\left\|\tilde{\boldsymbol{z}}_t^j - \boldsymbol{z}_t^j\right\|_2^2. \tag{3}$$

where the KL and MMD respectively denote Kullback–Leibler divergence and maximum mean discrepancy, which respectively decrease the latent distribution distance and predictive representation distance between posterior and prior networks. The latent reconstruction loss is an ℓ_2 loss between $\tilde{\boldsymbol{z}}_t^j$ and $\boldsymbol{z}_t^j$, which governs the evolution dynamics learning. Eventually, the overall optimization objective for the DWM is $\mathcal{L}_{\mathrm{DWM}} = \mathcal{L}_{\mathrm{DWM}}^h + \mathcal{L}_{\mathrm{DWM}}^l$.

4.4 Altitude-aware Perception (AAP)

The dual world models (DWM) require *prophet* observations at high-altitude and low-altitudes (see Fig. 3 and Fig. 4) to model evolution dynamics during training, but the drone agent can only access a single real egocentric observation and these additional observations are unavailable during inference. To bridge this gap and adapt the inference workflow of DWM, we propose an altitude-aware

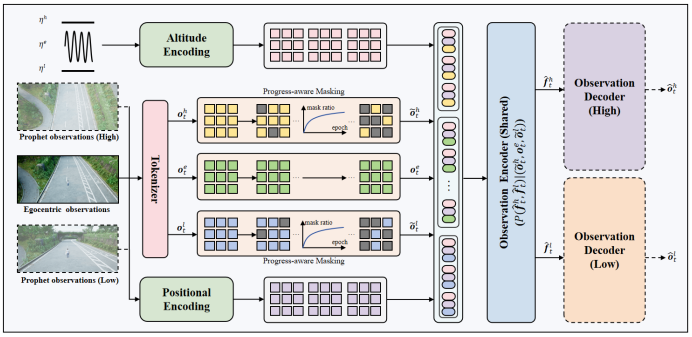


Figure 5: Detailed structure of the proposed Altitude-aware Perception (AaP) module. AaP uses real and prophet observations with progress-aware masking to learn pseudo high- and low-altitude representations, while training-only decoders provide reconstruction supervision.

perception (AaP) module in Fig. 5 adaptively encode and couple altitude-aware representations at different altitudes. It could be used to transform real egocentric observation into *pseudo* low-altitude and high-altitude representations ($\hat{\boldsymbol{f}}_t^h$ and $\hat{\boldsymbol{f}}_t^l$) during inference.

Overall, the AaP is a form of masked autoencoders. During training, the prophet high-altitude observation o_t^h and low-altitude observation o_t^l first undergo a progress-aware masking (PaM) strategy while the real egocentric observation $\boldsymbol{o}_t^e$ stays complete. The masking ratio ($m(n)$) in the PaM gradually increases from 0 to 1 as training proceeds (see Fig. 5 for details), so it just converges to the inference paradigm (no prophet observations) when training is finished. Then, they will be fed into a shared observation encoder for representation encoding and coupling, *i.e.*

$$(\hat{\boldsymbol{f}}_t^h, \hat{\boldsymbol{f}}_t^l) \sim P_\theta(\hat{\boldsymbol{f}}_t^h, \hat{\boldsymbol{f}}_t^l \mid \bar{\boldsymbol{o}}_t^h, \boldsymbol{o}_t^e, \bar{\boldsymbol{o}}_t^l), \quad \bar{\boldsymbol{o}}_t^j = \mathcal{M}_{m(n)}(\boldsymbol{o}_t^j),\ j \in \{h, l\} \tag{4}$$

where $P_\theta(\cdot)$ denotes the shared observation encoder that couples the masked high-altitude prophet observation $\bar{\boldsymbol{o}}_t^h$, the complete real egocentric observation $\boldsymbol{o}_t^e$, and the masked low-altitude prophet observation $\bar{\boldsymbol{o}}_t^l$. $\mathcal{M}_{m(n)}(\cdot)$ is the progress-aware masking operation with masking ratio $m(n)$. The outputs $\hat{\boldsymbol{f}}_t^h$ and $\hat{\boldsymbol{f}}_t^l$ denote pseudo high- and low-altitude current features used by the dual world models. As a result, the AaP encodes and couples altitude-specific observations, and gradually learns to derive *pseudo* high and low altitude representations from real egocentric observation. Finally, two training-only observation decoders respectively decode observations at the high and low altitudes ($\hat{\boldsymbol{o}}_t^h, \hat{\boldsymbol{o}}_t^l$) forming altitude-specific supervisions, *i.e.*

$$\mathcal{L}_{\text{AaP}} = \sum\nolimits_{j \in \{h,l\}} \left\| \hat{\boldsymbol{o}}_t^j - \boldsymbol{o}_t^j \right\|_2^2. \tag{5}$$

where $\hat{\boldsymbol{o}}_t^j$ denotes the reconstructed observation decoded by the training-only observation decoder for altitude regime j, and $\boldsymbol{o}_t^j$ denotes the corresponding prophet observation. The objective encourages the reconstructed high- and low-altitude observations to approach their corresponding prophet observations. As a result, the AaP is capable of generating pseudo altitude-specific representations ($\hat{\boldsymbol{f}}_t^h, \hat{\boldsymbol{f}}_t^l$) during inference without any prophet observation for dual world models (DWM).

5 Experiments

5.1 Experimental Setup

We use OSTrack with a Transformer visual encoder and a ResNet-18 policy backbone. AaP and DWM are optimized with AdamW, and the final ReDeT policy is trained with PPO for 320,000 simulator steps. The altitude-aware prophet observations are rendered at 3 m and 20 m. Complete optimization, camera, input-resolution, and control settings are provided in Appendix B.

5.2 Experimental Results

Baseline and comparing methods. To establish a benchmark for the embodied tracking task in aerial interactive environments, we conduct performance comparison experiments based on several configurations including standard passive trackers, rule-based policies, reinforcement learning (RL)

based policies, and the proposed AaDWorlds framework. These configurations reflect a logical progression of tracking paradigms. We start with traditional passive tracking methods represented by MOSSE (Bolme et al., 2010) and OSTrack (Ye et al., 2022) which focus solely on target localization within pre-recorded frames. We then introduce an active but random baseline where a passive tracker (OSTrack) is combined with a random movement strategy to demonstrate the impact of stochastic egocentric transitions on tracking stability. Beyond random exploration, we also incorporate a heuristic-based policy that utilizes a rule-based controller to adjust the drone position for target centering. Furthermore, we evaluate more sophisticated closed-loop systems that integrate specialist trackers with reinforcement learning policies (such as AC (Konda & Tsitsiklis, 2000) and PPO (Schulman et al., 2017) architectures) to establish a competitive benchmark for drone-embodied tracking. Building upon the baseline (PPO + OSTrack), the proposed AaDWorlds framework further integrates altitude aware perception and dual world model strategies to handle complex 3D dynamic environments and provide predictive pseudo observations. Finally, we introduce a prophet policy as the upper bound method, which leverages prophet ground-truth trajectory information to maintain near-optimal positioning without any perception noise. This upper bound allows us to quantify the performance gap and evaluate how effectively our tracking policy approximates optimal tracking behavior in complex environments. We evaluate the performance of these methods on the DeTrack benchmark across all 4 primary evaluation metrics in Table 1.

Table 2: Experimental results on the DeTrack benchmark (%).

Paradigm	Method	VR↑	mIoU↑	TR↑	SR↑
Passive	MOSSE	9.32	7.17	6.31	1.41
	OSTrack	9.34	7.39	6.32	1.41
Rule-based	Random + OSTrack	10.35	8.32	8.42	3.42
	Heuristic + OSTrack	12.32	10.25	11.21	7.42
RL-based	AC + OSTrack	24.58	14.34	18.64	12.97
	PPO + OSTrack	25.88	15.67	19.07	14.08
	AaDWorlds	**29.00**	**17.06**	**22.57**	**17.66**
Upper-bound	Prophet + OSTrack	96.28	77.28	87.06	95.73

Table 3: Ablation studies on the DeTrack benchmark (%).

Base	AaP	DWM	VR↑	mIoU↑	TR↑	SR↑
✓	×	×	25.88	15.67	19.07	14.08
✓	✓	×	27.79	15.78	19.42	14.13
✓	×	✓	27.64	16.24	20.15	14.61
✓	✓	✓	**29.00**	**17.06**	**22.57**	**17.66**

Performance comparison on the DeTrack benchmark. (1) *Performance of passive trackers degrades significantly on the DeTrack benchmark*: Although conventional passive trackers (such as MOSSE (Bolme et al., 2010) and OSTrack (Ye et al., 2022)) exhibit great performance on the pre-recorded tracking datasets (e.g., OSTrack reports 70.7% AUC on UAV123 (Mueller et al., 2016) and 83.9% AUC on TrackingNet (Muller et al., 2018)), they achieve very poor tracking performance on our DeTrack benchmark tailored for drone-embodied tracking task (Table 2). The performance degradation is primarily attributed to the intrinsic characteristics of the DeTrack benchmark. These challenges collectively exacerbate the difficulty of target localization and flight control and eventually constitute the fundamental reasons for the observed performance degradation. (2) *Rule-based policy brings some but limited improvement*: With the aid of the heuristically rule-based closed-loop drone controlling policy, the performance improves by some degree over passive tracker OSTrack (Ye et al., 2022), indicating the importance of closed-loop drone controlling during the drone-embodied tracking. (3) *RL-based policies bring promising improvement and our AaDWorlds achieves the best performance*. The reinforcement learning policies (AC (Konda & Tsitsiklis, 2000) and PPO (Schulman et al., 2017)) achieves much promising improvement for passive trackers and the PPO policy performs better than the AC policy, where PPO+OSTrack is chosen as the baseline of our AaDWorlds framework. Eventually, our AaDWorlds achieves best performance across all evaluation metrics that demonstrates the stronger adaptability during drone-embodied tracking. In detail, it improves the VR from 25.88% to 29.00%, the mIoU from 15.67% to 17.06% , the TR from 19.07% to 22.57%, and the SR from 14.08% to 17.66%. The performance gains attribute to our altitude aware perception module and dual world model strategy. The dual world models predict imagined future features under both high- and low-altitude regimes to complement drone-egocentric observation, which simultaneously facilitates the view field, target details, and obstacle avoidance during drone-embodied tracking.

5.3 Ablation Studies

To analyze the effectiveness of each primary component in our AaDWorlds framework, we conduct ablation studies on the DeTrack benchmark. Starting from the PPO + OSTrack baseline, we progressively introduce the Altitude-aware Perception (AaP) module and the Dual World Models (DWM)

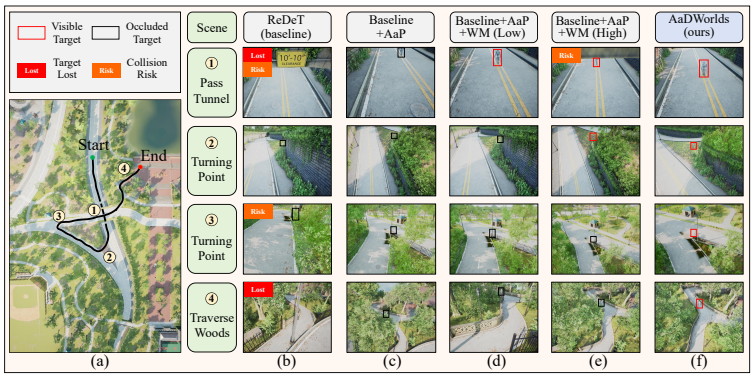


Figure 6: Qualitative comparison of target visibility and flight safety across four representative scenes. (a) Target trajectory overview. (b)–(f) Comparison between AaDWorlds and four method variants.

under different combinations. As shown in Table 3, each module brings performance improvements across all metrics. The AaP improves the representation of altitude-aware current features under extensive variation, while the DWM further enhances decision-making by providing imagined future features for both high- and low-altitude regimes. When combining the AaP and DWM together, the model achieves the best performance, demonstrating the effectiveness of our AaDWorlds framework. The 3 m/20 m altitude pair achieves the best performance, with detailed sensitivity analysis provided in Appendix B.3.

5.4 Qualitative Visualization

As shown in Fig. 6(b), the ReDeT baseline performs worse since the current egocentric observation is insufficient for reliable closed-loop tracking. For example, in the Pass Tunnel and Traverse Woods scenes, the ReDeT baseline encounters target loss and unsafe close-range flight around tunnel structures and dense vegetation (see Fig. 6(b-1) and Fig. 6(b-4)). In the two Turning Point scenes, the drone adjusts its viewpoint only after the target direction has changed, leading to delayed direction following and unstable target observability (see Fig. 6(b-2) and Fig. 6(b-3)). After introducing the proposed AaP module, the variant Baseline + AaP in Fig. 6(c) improves the perception of target scale and local scene structure through altitude-aware current features. Compared with the results from ReDeT baseline in Fig. 6(b), the target can be kept more observable in several scenes. However, since this variant still lacks future dynamics, it remains difficult to anticipate turning changes and near-obstacle interactions before the viewpoint becomes degraded. Furthermore, the proposed world models at low and high altitudes effectively alleviate this issue and lead to different scene-dependent behaviors. On the one hand, aiding with the low-altitude world model (Baseline+AaP+WM (Low)), Fig. 6(d-1) and Fig. 6(d-4) preserve more target-detail clues in the Pass Tunnel and Traverse Woods scenes, but the drone in this variant tends to stay closer to tunnel structures, road boundaries, and vegetation, leaving limited safety margins. On the other hand, aiding with the high-altitude world model (Baseline+AaP+WM (High)), Fig. 6(e-2) and Fig. 6(e-3) show better responses in the Turning Point scenes, because the wide-view imagined states at high altitude provide more global motion context for earlier direction adjustment, but this variant weakens local target-detail and obstacle-structure cues, and thus still suffers from risky behavior in narrow or cluttered areas, such as the Pass Tunnel scene in Fig. 6(e-1). Eventually, our AaDWorlds in Fig. 6(f) combines world models from both low and high altitudes, allowing the agent to use both future target-detail clues in the low altitude and future wide-view context in the high altitude. Compared with the variant in Fig. 6(d) and Fig. 6(e), our AaDWorlds keeps the target observable and simultaneously maintains safer distances from surrounding structures across all 4 scenes, effectively alleviating the intrinsic altitude-mediated contradiction between target visibility and flight safety during drone-embodied tracking.

6 Conclusion

In this paper, we introduce a benchmark (DeTrack) for closed-loop drone-embodied tracking in interactive 3D environments, and propose an altitude-aware dual-world (AaDWorlds) framework for this task. The DeTrack differs from conventional passive aerial tracking by considering perception-action coupling, mixed occlusions, altitude-dependent visibility-safety trade-offs, and joint tracking-obstacle avoidance. To address the intrinsic altitude-mediated contradiction, the AaDWorlds com-

bines dual world models with an altitude-aware perception module, providing altitude-aware current features and imagined future features for action prediction. Extensive experimental results on the DeTrack benchmark show that our AaDWorlds framework significantly improves the performance across all metrics.

AI USE STATEMENT

Generative AI tools were used for language polishing, readability improvement, and minor formatting assistance. All scientific content, experimental results, analyses, and conclusions were developed and verified by the authors. The authors reviewed all AI-assisted content and take full responsibility for the final manuscript.

REFERENCES

Ahmad Taher Azar, Anis Koubaa, Nada Ali Mohamed, Habiba A. Ibrahim, Zahra Fathy Ibrahim, Muhammad Kazim, Adel Ammar, Bilel Benjdira, Alaa M. Khamis, Ibrahim A. Hameed, and Gabriella Casalino. Drone deep reinforcement learning: A review. *Electronics*, 10(9):999, 2021.

Luca Bertinetto, Jack Valmadre, João F. Henriques, Andrea Vedaldi, and Philip H. S. Torr. Fully-convolutional siamese networks for object tracking. In *European Conference on Computer Vision (ECCV) Workshops*, 2016.

David S. Bolme, J. Ross Beveridge, Bruce A. Draper, and Yui Man Lui. Visual object tracking using adaptive correlation filters. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2544–2550, 2010.

Jun Hoong Chan, Kai Liu, Yu Chen, A. S. M. Sharifuzzaman Sagar, and Yong-Guk Kim. Reinforcement learning-based drone simulators: Survey, practice, and challenge. *Artificial Intelligence Review*, 57:281, 2024.

Xin Chen, Bin Yan, Jiawen Zhu, Dong Wang, Xiaoyun Yang, and Huchuan Lu. TransT: Transformer tracking. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Dawei Du, Yuankai Qi, Hongyang Yu, Yifan Yang, Kaiwen Duan, Guorong Li, Weigang Zhang, Qingming Huang, and Qi Tian. The unmanned aerial vehicle benchmark: Object detection and tracking. In *European Conference on Computer Vision (ECCV)*, pp. 370–386, 2018.

Epic Games. Unreal Engine 4.27 Documentation. `https://dev.epicgames.com/documentation/en-us/unreal-engine?application_version=4.27`, 2021. Official documentation.

Zhihui Feng, Xitai Na, Shiji Hai, Qingbin Sun, and Jinshuo Shi. Deep reinforcement learning for UAV target search and continuous tracking in complex environments with gaussian process regression and prior policy embedding. *Electronics*, 14(7):1330, 2025.

João F. Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista. High-speed tracking with kernelized correlation filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(3): 583–596, 2015.

Hongji Huang, Yuchun Yang, Hong Wang, Zhiguo Ding, Hikmet Sari, and Fumiyuki Adachi. Deep reinforcement learning for UAV navigation through massive MIMO technique. *arXiv preprint arXiv:1901.10832*, 2019.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014.

Eric Kolve, Roozbeh Mottaghi, Daniel Gordon, Yuke Zhu, Abhinav Gupta, and Ali Farhadi. AI2-THOR: An interactive 3d environment for visual AI. *arXiv preprint arXiv:1712.05474*, 2017.

Vijay R. Konda and John N. Tsitsiklis. Actor-critic algorithms. In *Advances in Neural Information Processing Systems*, 2000.

Srivatsan Krishnan, Behzad Boroujerdian, William Fu, Aleksandra Faust, and Vijay Janapa Reddi. Air learning: A deep reinforcement learning gym for autonomous aerial robot visual navigation. *Machine Learning*, 110(9):2501–2540, 2021.

Bo Li, Wei Wu, Qiang Wang, Fangyi Zhang, Junliang Xing, and Junjie Yan. SiamRPN++: Evolution of siamese visual tracking with very deep networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4282–4291, 2019.

Meng-Shi Liu, Jin-Quan Gao, Gu-Yue Hu, Guang-Fu Hao, Tian-Zi Jiang, Chen Zhang, and Shan Yu. Monkeytrail: A scalable video-based method for tracking macaque movement trajectory in daily living cages. *Zoological Research*, 43(3):343–351, 2022. doi: 10.24272/j.issn.2095-8137.2021.353.

Wenhan Luo, Peng Sun, Fangwei Zhong, Wei Liu, Tong Zhang, and Yizhou Wang. End-to-end active object tracking via reinforcement learning. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 3286–3295, 2018.

Microsoft Research. AirSim. `https://microsoft.github.io/AirSim/`, 2021. Official documentation.

Matthias Mueller, Neil Smith, and Bernard Ghanem. UAV123: A benchmark and simulator for UAV tracking. In *European Conference on Computer Vision (ECCV)*, pp. 445–461, 2016.

Matthias Muller, Adel Bibi, Silvio Giancola, Salman Alsubaihi, and Bernard Ghanem. Trackingnet: A large-scale dataset and benchmark for object tracking in the wild. In *European Conference on Computer Vision (ECCV)*, pp. 300–317, 2018.

Hy Nguyen, Srikanth Thudumu, Hung Du, Kon Mouzakis, and Rajesh Vasa. UAV dynamic object tracking with lightweight deep vision reinforcement learning. *Algorithms*, 16(5):227, 2023.

Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A platform for embodied AI research. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9339–9347, 2019.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Shital Shah, Debadeepta Dey, Chris Lovett, and Ashish Kapoor. Airsim: High-fidelity visual and physical simulation for autonomous vehicles. In *Field and Service Robotics*, pp. 621–635, 2018a.

Shital Shah, Debadeepta Dey, Chris Lovett, and Ashish Kapoor. Airsim: High-fidelity visual and physical simulation for autonomous vehicles. In *Field and Service Robotics*, pp. 621–635, 2018b.

Yong Wang, Zirong Huang, Robert Laganière, Huanlong Zhang, and Lu Ding. A UAV to UAV tracking benchmark. *Knowledge-Based Systems*, 261:110197, 2023.

Xin Wu, Wei Li, Danfeng Hong, Ran Tao, and Qian Du. Deep learning for UAV-based object detection and tracking: A survey. *IEEE Geoscience and Remote Sensing Magazine*, 10(4):91–124, 2022.

Fei Xia, Amir R. Zamir, Zhi-Yang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. Gibson Env: Real-world perception for embodied agents. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Bin Yan, Houwen Peng, Jianlong Fu, Dong Wang, and Huchuan Lu. STARK: Learning spatio-temporal transformer for visual tracking. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

Botao Ye, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. Joint feature learning and relation modeling for tracking: A one-stream framework. In *European Conference on Computer Vision (ECCV)*, 2022.

Yue Zhao, Dong Wang, Huchuan Lu, Ying Wang, Xue Zhang, and Xiang Li. Vision-based anti-UAV detection and tracking. *IEEE Transactions on Intelligent Transportation Systems*, 23(12): 23639–23652, 2022.

Pengfei Zhu, Longyin Wen, Xiao Bian, Haibin Ling, and Qinghua Hu. Vision meets drones: A challenge. *arXiv preprint arXiv:1804.07437*, 2018.

Pengfei Zhu, Longyin Wen, Dawei Du, Xiao Bian, Haibin Ling, Qinghua Hu, et al. VisDrone-VDT2018: The vision meets drone video detection and tracking challenge results. In *Computer Vision – ECCV 2018 Workshops*, pp. 496–518, 2019.

Pengfei Zhu, Longyin Wen, Xiao Bian, Haibin Ling, and Qinghua Hu. Detection and tracking meet drones challenge. *arXiv preprint arXiv:2001.06303*, 2020.

A Additional Benchmark Details

Following the benchmark overview above, this subsection provides detailed statistical analysis from two aspects: diverse interactive environments and detailed target trajectory statistics. The interactive environments provide diverse 3D scenes, semantic regions, rendering conditions, and moving distractors for closed-loop drone-embodied tracking. The trajectory statistics summarize the data characteristics of target motions, including trajectory scale, waypoint density, and spatial distribution.

(1) *Diverse interactive environments and rendering conditions.* Our benchmark environments are constructed in an interactive Unreal Engine 4.27 simulator with the AirSim plugin, covering four representative scene styles: Town, Rural, Park, and City. These scene styles differ in global layout and visual statistics, such as building density, vegetation coverage, road topology, and background clutter, providing diverse geometric structures and natural occlusion sources across environments. To further increase visual diversity, each scene is rendered under multiple domain conditions, including Day, Night, Fog, Sun, Rain, and Snow. As shown in Fig. 7, the same scene style can exhibit substantially different appearances under day, fog, night, and snow conditions, which introduces additional challenges such as low illumination, reduced visibility, and background variation.



Figure 7: Illustration of scene-style and rendering-condition diversity in the DeTrack benchmark. The figure shows four representative scene styles, including Town, Rural, Park, and City, under different rendering conditions, including day, fog, night, and snow, demonstrating the visual diversity introduced by scene layouts, illumination changes, and weather variations.

In addition to scene-level and rendering-level diversity, each scene contains multiple semantic regions, such as bridges and underpasses, open plazas, dense vegetation, waterfronts, sports fields, intersections, and narrow corridors. These regions introduce substantial variation in local geometry, occlusion patterns, and background appearance. Fig. 8 provides representative examples of within-scene semantic region diversity. Meanwhile, the environments contain primary tracking targets together with additional moving distractors, such as pedestrians, vehicles, and bicycles. This setting makes the benchmark closer to interactive drone operation, where the tracking agent needs to maintain the target while facing visual distractors and dynamic occlusions.

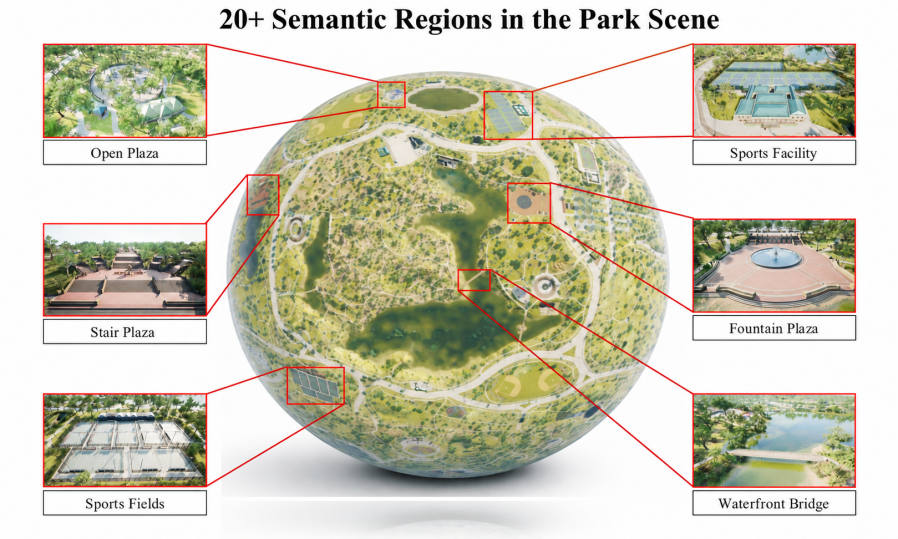


Figure 8: Examples of semantic region diversity within the Park scene. The central map and surrounding thumbnails show representative semantic regions, including open plaza, stair plaza, sports field, sports facility, fountain plaza, and waterfront bridge. These regions introduce diverse local layouts, occlusion patterns, and background appearances for drone-embodied tracking.

(2) *Detailed target trajectory statistics and analysis*. The DeTrack trajectory library contains 11,368 target trajectories. Each trajectory consists of 2 to 99 waypoints, with a mean of 13.84 and a median of 8. In terms of spatial extent, trajectory lengths range from 8.27 m to 482.74 m, with a mean length of 80.09 m and a median length of 69.33 m. We summarize the basic numerical statistics of the target trajectory library in Fig. 9. These statistics show that the trajectory library covers diverse waypoint densities and motion scales, making it suitable for closed-loop drone-embodied tracking evaluation under different target-following horizons.

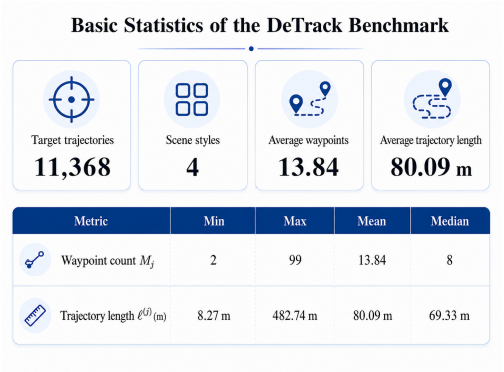


Figure 9: Basic statistics of the target trajectory library in DeTrack.

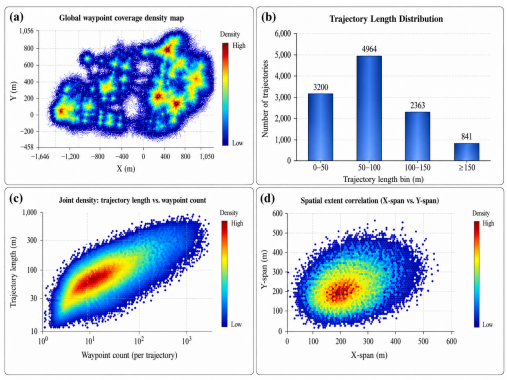


Figure 10: Geometry-level statistics of DeTrack trajectories: (a) waypoint density, (b) trajectory-length distribution, (c) length-waypoint joint density, and (d) planar spatial extent.

Beyond the basic numerical statistics in Fig. 9, we further analyze the trajectory library from four geometry-level aspects, as shown in Fig. 10, including global waypoint coverage, trajectory length distribution, the joint distribution between trajectory length and waypoint count, and planar spatial extent correlation. The global waypoint coverage indicates that the target trajectories span a broad simulator space. The trajectory length distribution further reflects the variation of motion scales in the library. The joint distribution between trajectory length and waypoint count shows the correlation between motion range and waypoint density. For planar spatial extent correlation, we compute the X-span and Y-span of each target trajectory by subtracting the minimum waypoint coordinate from the maximum waypoint coordinate along the corresponding axis. This statistic further reveals diverse movement ranges on the simulator ground plane. Taken together, these statistics show that the trajectory library covers diverse motion scales, waypoint densities, and planar movement ranges, which supports embodied tracking evaluation under both short-range maneuvers and longer-horizon target following.

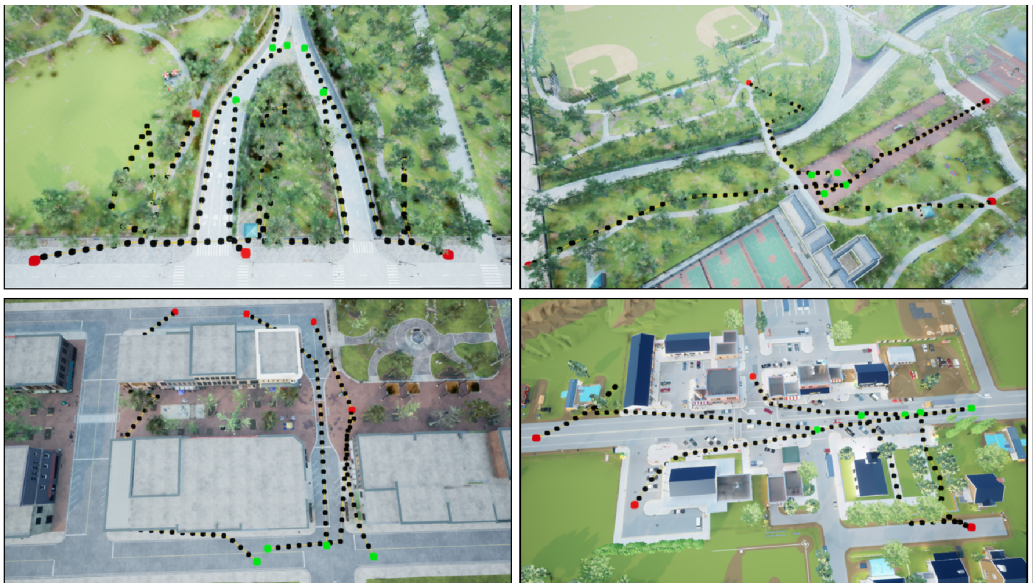


Figure 11: Sample illustration of target trajectories in the DeTrack benchmark.

In addition to statistical analysis, Fig. 11 provides qualitative trajectory examples to illustrate the typical geometry and waypoint density of target trajectories in the simulator. The start waypoint is marked in green, the end waypoint is marked in red, intermediate waypoints are shown as black dots, and consecutive waypoints are connected by a yellow polyline. These examples provide an intuitive visualization of the trajectory geometry and waypoint density in the benchmark.

B Additional Experimental Details

B.1 Implementation Details

Regarding the main settings of our AaDWorlds framework, the backbone of the OSTrack tracking model is a Transformer-based visual encoder initialized with publicly available pretrained weights. In addition, a pretrained ResNet-18 backbone is adopted for the policy network to encode the online egocentric observations. For model optimization, we use the AdamW optimizer in the first two training stages. The AaP module is pretrained for 300 epochs with a streaming batch size of 8, a base learning rate of 3×10^{-4}, and a weight decay of 0.05. The DWM is subsequently optimized for 300 epochs on trajectory data collected under the DeTrack benchmark protocol, using a base learning rate of 1×10^{-3} and a weight decay of 0.01. During the final ReDeT policy learning stage, the drone agent is trained for 320,000 simulator steps using PPO, with a learning rate of 3×10^{-4}, a clipping parameter of 0.2, and a generalized advantage estimation factor of 0.95. For the closed-loop drone-embodied tracking protocol, the embodied agent is equipped with a rigidly mounted RGB camera that renders images at a resolution of 640×360, with a 90° horizontal field

of view (FOV) and a fixed 0° pitch angle. For all learning-based modules, input frames are resized to 256×256. In the altitude-aware setting, the high-altitude and low-altitude prophet observations are rendered at fixed flight heights of 20 m and 3 m, respectively. The simulator advances with a discrete time step of $\Delta t = 0.5$ s. During flight, the drone is subject to a maximum linear speed of 8.0 m/s and a yaw-rate limit of $\pm 45^\circ$/s.

B.2 Evaluation Metric Details

We provide additional details of the four evaluation metrics summarized in Table 1. For a trajectory of length T, B_t^{gt} and B_t^{trk} denote the ground-truth and tracker-predicted bounding boxes at step t, respectively. The variable p_t^{vis} denotes the number of visible target pixels in the segmentation mask, while τ_{vis} denotes the visibility threshold and τ_{trk} denotes the IoU threshold for effective tracking.

Visible rate (VR) evaluates the proportion of tracking steps in which the target remains sufficiently visible. Mean IoU (mIoU) measures the average overlap between the tracker-predicted and ground-truth bounding boxes over the trajectory. Tracking rate (TR) measures the proportion of steps in which the target is both sufficiently visible and effectively tracked.

For trajectory-level evaluation, N_{traj} denotes the number of evaluated trajectories, T_j denotes the length of the j-th trajectory, $d_{xy,T_j}^{(j)}$ denotes the final horizontal distance between the UAV and the target, and τ_d denotes the trajectory success threshold. Trajectory success rate (SR) measures the proportion of evaluated trajectories whose final UAV-target horizontal distance satisfies the predefined success criterion.

Table 4: Influence of altitude combinations on the DeTrack benchmark (%).

Low Alt.	High Alt.	VR↑	mIoU↑	TR↑	SR↑
1 m	10 m	25.62	14.91	19.34	13.22
	20 m	27.52	16.28	21.12	16.01
	40 m	24.96	14.58	18.83	12.57
3 m	10 m	27.71	16.17	21.08	15.92
	20 m	**29.00**	**17.06**	**22.57**	**17.66**
	40 m	26.93	15.82	20.61	15.34
6 m	10 m	26.21	15.25	19.76	13.68
	20 m	27.86	16.39	21.34	16.24
	40 m	25.34	14.79	19.07	12.95

B.3 Influence analysis of the altitude combinations

To further analyze the effect of flight altitude on altitude-aware prophet observations, we conduct an altitude-pair sensitivity analysis by varying the low- and high-altitude settings used in AaP and DWM. In the altitude-aware setting, the low-altitude prophet observation usually provides larger target scale and richer local details, but overly low viewpoints may introduce stronger near-view occlusion and unstable perspective changes. In contrast, the high-altitude prophet observation provides a wider field of view and more global motion context, while overly high viewpoints may make the target too small and visually ambiguous. As shown in Table 4, the 3m/20m pair achieves the best performance across all evaluation metrics. Compared with 1m/20m, it avoids excessive near-view occlusion and perspective instability; compared with 6m/20m, it better preserves the target-detail advantage of the low-altitude branch. For the high-altitude branch, 10m provides relatively limited global context, whereas 40m weakens fine-grained tracking cues due to the smaller target scale. The lower performance of the corner combinations further indicates that the two altitude branches should provide complementary local and global information rather than overly close or overly distant observations. These results suggest that the 3m/20m setting forms a more suitable trade-off for altitude-aware perception and future imagination in drone-embodied tracking.