TRACE: TARGET-STATE RISK-ALIGNED CONDITIONAL EXECUTION FOR EMBODIED TRACKING

Anonymous authors
Paper under double-blind review

ABSTRACT

Embodied visual tracking requires a robot to follow a language-specified person under occlusion, look-alike distractors, and ambiguous instructions. We show that improving tracking persistence can simultaneously increase safety violations. Using a compact 0.6B vision-language-action (VLA) tracker with a frozen backbone, we update only the planner head with proximal policy optimization (PPO). PPO improves success and following rates across all three EVT-Bench settings, but increases the collision rate on distracted tracking because episodes that previously ended in target loss can instead end in contact. We introduce TRACE, an execution-time target-state risk gate that preserves the learned policy and intervenes selectively. A lightweight risk head processes 84 cues, including candidate boxes, identity margins, a learned range-and-bearing probe, and the policy's commands. When the calibrated risk exceeds a threshold, TRACE rescales the translational component of the next waypoint while leaving yaw unchanged. The executed action is identical to PPO's on more than 98% of steps. In paired evaluations on identical episodes, TRACE reduces the distracted-tracking collision rate by 3.96 percentage points, while further improving success and following rates. Matched slowdown controls without target-state timing do not reproduce this improvement. Concurrent evaluation reproduces a collision reduction of 3.77 points, compared with 0.29 points for identical-arm null pairs, indicating that the effect exceeds evaluation variability. These results show that target-state-aware execution can recover the safety cost of policy post-training with minimal intervention.

1 INTRODUCTION

Embodied visual tracking (EVT) requires an agent to continuously follow a language-specified person from egocentric observations over long horizons (Luo et al., 2018; Zhong et al., 2024; Wang et al., 2025). The task requires the joint solution of target identification, temporal association, and motion planning under occlusion, look-alike distractors, and ambiguous instructions. Recent vision-language-action (VLA) models integrate visual perception, language understanding, and robot control (Brohan et al., 2023a;b; Kim et al., 2024; Black et al., 2024). The strongest reported results on EVT-Bench come from a 7B-scale model trained on large-scale data (Wang et al., 2025), whereas deployment favors compact models with limited adaptation cost and low execution overhead.

We construct a compact three-stage tracking system based on a frozen Qwen3-0.6B backbone. The supervised base model keeps the vision encoder, language backbone, and recognition adapter fixed, while a planner head predicts eight future waypoints. We then apply PPO (Schulman et al., 2017; 2015) only to the planner head, which improves success and following rates across all three EVT-Bench settings: the planner head is an effective control point for tracking persistence, and the representation backbone need not be updated. The improvement carries a safety cost in distracted tracking. Relative to the supervised model, PPO reduces early target loss but increases collisions, changing the terminal outcome of some episodes from target loss to physical contact (Figure 1A).

This observation exposes a limitation of optimizing persistence with a scalar task reward. Existing safe-control and recovery methods (Achiam et al., 2017; Ray et al., 2019; Johannink et al., 2019; Thananjeyan et al., 2021; Ames et al., 2019; Dalal et al., 2018) typically impose geometric

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.

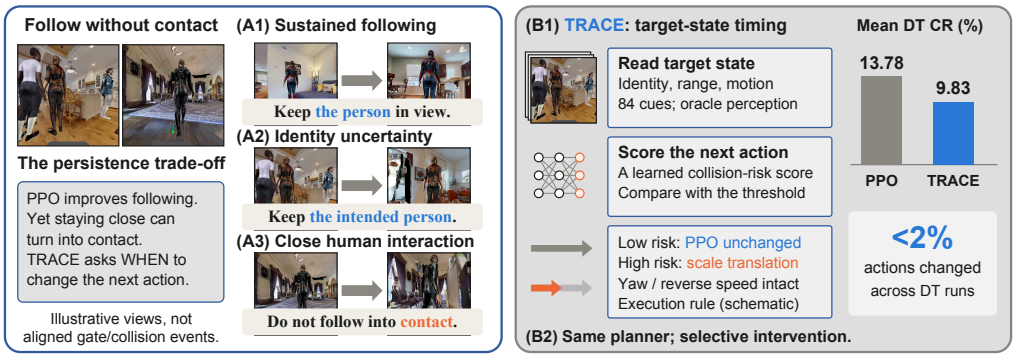

Figure 1: **The safety cost of persistent following, and the gate that removes it.** **(A)** The robot must keep a language-specified person in view under distractors and ambiguity; PPO on the planner head makes it persist longer, and on distracted tracking part of that persistence ends in contact. **(B)** TRACE scores the next action from target-state cues and rescales its translation only when the score crosses a threshold; the action is unchanged on over 98% of steps.

constraints or respond to local collision conditions. They do not explicitly model the target-state uncertainty that precedes collisions during distracted tracking. In our experiments, a distance-only execution filter does not recover the safety degradation introduced by PPO (§4.4). These results motivate an execution-time mechanism that uses the tracker's target-state estimate to intervene selectively while preserving the persistence learned by PPO.

We therefore introduce TRACE, a target-state risk gate applied after policy prediction (Figure 1B). A lightweight head receives 84 target-related cues, including candidate bounding boxes, identity margins, a learned range-and-bearing probe, and the policy's commands and predicted waypoints. The head produces a collision-risk score. Once the score exceeds a calibrated threshold, TRACE rescales the translational component of the next waypoint while preserving its yaw component. At all other steps, the executed action is identical to the PPO action. The gate reduces the distracted-tracking collision rate (CR) from 13.78 to 9.83, below the supervised baseline, while further improving success and following rates (SR, FR).

Our contributions are as follows.

- **A compact VLA tracking stack with an empirical analysis of post-training safety costs.** We develop a 0.6B frozen-backbone VLA tracker and restrict PPO post-training to its planner head. This design improves success and following rates on all three EVT-Bench settings but raises the distracted-tracking collision rate from 11.01 to 13.78, establishing a measurable trade-off between persistence and collision safety under planner-only post-training.
- **TRACE: selective target-state risk gating for safe execution.** We introduce an execution-time gate that predicts collision risk from target-state cues and modifies only the translational component of the next waypoint when the risk exceeds a calibrated threshold. On distracted tracking, TRACE reduces the collision rate by 3.96 percentage points while increasing the success rate by 2.03 points and the following rate by 2.14 points, at approximately 50 μs per step and fewer than 2% of executed actions changed. Matched slowdown controls without target-state timing remain at the PPO reference, and the effect persists across two scene corpora, three collision geometries, held-out episodes, and a later code version.
- **Concurrent evaluation for separating intervention effects from benchmark variability.** Re-evaluating an unchanged policy shifts collision rate by up to 1.4 points and success rate by up to 2.7, so we evaluate the gated and ungated policies concurrently together with identical-arm null pairs. Under this protocol, TRACE produces a collision-rate change of -3.77 points, compared with -0.29 points for identical-arm pairs, providing a paired estimate of the intervention effect relative to evaluator variability.

2 Related Work

Embodied visual tracking. Early work casts the task as reinforcement learning over egocentric observations (Luo et al., 2018; 2019), later hardened against evasive targets and occlusion by adversarial and structured variants (Zhong et al., 2019; 2021b; 2023). A second line decouples the problem into a visual tracker and a controller, pairing detection or re-identification with image-based servoing (Gupta et al., 2017; Bhat et al., 2019; Maalouf et al., 2024). TrackVLA (Wang et al., 2025) instead unifies recognition and planning in a single 7B model and releases EVT-Bench, and TrackVLA++ (Liu et al., 2025) extends it with explicit reasoning and memory. Both train large backbones end-to-end for the task. In contrast, we hold a compact frozen-backbone tracker fixed and ask what post-training its planner head alone does to collision behavior. Appendix A covers the adjacent literature on robot following, social navigation, and pursuit–evasion.

Post-training and runtime safety. Post-training a supervised policy with on-policy RL is standard practice for language, vision-language, and embodied controllers (Brohan et al., 2023b; Kim et al., 2024; Black et al., 2024), with PPO and generalized advantage estimation the usual choice (Schulman et al., 2017; 2015). Safety is typically added as a separate channel that edits or constrains a proposed action: control-barrier functions (Ames et al., 2019), learned safety layers (Dalal et al., 2018), recovery policies (Thananjeyan et al., 2021), residual corrections (Johannink et al., 2019), and constrained MDPs that fold the constraint into the objective (Achiam et al., 2017; Ray et al., 2019). Each of these reads geometry, known dynamics, or a constraint signal defined independently of the followed person. In contrast, we condition the intervention on the tracker's own estimate of the target's state, which is the quantity that changes when persistent following turns into contact.

Pretrained components and simulation benchmarks. The quantities our risk head reads are produced by components we do not retrain: re-identification confidence from tracking-by-detection pipelines (Wojke et al., 2017; Zhang et al., 2022), frozen vision-language backbones (Radford et al., 2021; Oquab et al., 2023; Bai et al., 2025), and waypoint heads of the kind used in diffusion and transformer policies (Ho et al., 2020; Chi et al., 2024; Zhao et al., 2023). EVT-Bench itself is built on Habitat (Savva et al., 2019; Puig et al., 2023) with HM3D and Matterport3D scenes (Ramakrishnan et al., 2021; Chang et al., 2017), and inherits its episode and instruction conventions from vision-language navigation (Anderson et al., 2018; Krantz et al., 2020). We take all of these as given; Appendix A lists the remaining references.

3 Method

Problem and overview. At each step t, the agent receives a language instruction I that specifies the target and an egocentric RGB history O_t. A compact VLA tracker maps (I, O_t) to an 8×3 waypoint sequence $W_t = \{w_{1,t}, \ldots, w_{8,t}\}$, where each waypoint is $w_{i,t} = (v_x, v_y, \omega)$. Only the first waypoint is executed through a PD controller, after which the agent observes again and replans. An episode is successful when the agent keeps the described target within the evaluator's following band without terminating in collision. Our goal is to increase persistence without converting lost-target failures into collisions. The construction has three components: a compact supervised tracker whose recognition components are then frozen (§3.1), PPO on its planner head alone (§3.2), and a gate that rescales the next waypoint only when the risk score is high (§3.3; Figure 2). Each component leaves the previous one unchanged, so we can measure the effect of each separately.

3.1 A Compact VLA Tracker

The base tracker couples a vision encoder, a Qwen3-0.6B language backbone, and a recognition adapter that projects visual tokens into the language space, followed by a planner head that emits the 8×3 waypoint sequence. It is trained by supervised fine-tuning (SFT; Appendix B) and then serves as the fixed reference: the encoder, backbone, and adapter are never updated again, so every later change is attributable to the planner head or the gate. The limitation is persistence rather than recognition: most failures are episodes in which the target is dropped early and never recovered, rather than episodes in which the wrong person is followed.

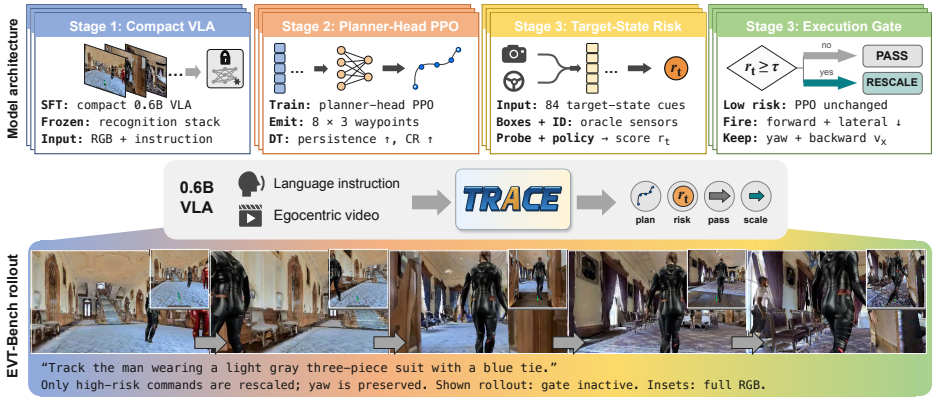

Figure 2: **The tracking stack.** Supervised tracker, then PPO on the planner head alone, then a risk score r_t that gates execution: below τ the PPO command passes through; above it, forward and lateral motion are rescaled and yaw is preserved. The 84 cues use EVT-Bench's own detection and identity (§4.4). The rollout below is one episode on which the gate never fires.

3.2 Planner-Head PPO

To improve persistence without disturbing recognition, we freeze the vision encoder, language backbone, and recognition adapter; only the deterministic waypoint head μ_θ, an exploration term, and a value head are updated. Training samples actions around the waypoint mean, $\hat{\mathbf{w}}_t \sim \mathcal{N}(\mu_\theta(h_t), \mathrm{diag}(\sigma^2))$, and evaluation uses the mean itself. All RL-style variants share the same reward:

$$r_t = w_f \mathbb{1}[\text{follow}_t] - w_c \mathbb{1}[\text{collide}_t] + w_d\,(d_{t-1} - d_t) - w_{\text{far}} \mathbb{1}[d_t > 4\,\text{m}], \tag{1}$$

where $(w_f, w_c, w_d, w_{\text{far}}) = (1.0, 5.0, 0.5, 0.3)$. The follow and collision indicators come from the EVT-Bench following band and simulator contact flag, and d_t is the geodesic distance to the target. We optimize this objective with PPO and GAE (Schulman et al., 2017; 2015) over two rounds of 200 on-policy episodes. Because the perception path is frozen, this update cannot change which person the tracker believes it is following; it can only change how long the policy commits to that belief, so the collision term in Eq. 1 does not prevent the failure it is meant to penalize. The update improves following but does not by itself indicate when the next action is unsafe.

Single-target tracking. When the target is unambiguous, the risk score rarely crosses a threshold that changes the trajectory without lowering SR, so we treat this setting separately. Starting from the collision-anchored planner, we run one further low-learning-rate PPO round at c=16 and c=18, with no gate or recovery residual. For each training run a selection subset picks the penalty level with lower CR, and the resulting mix is evaluated on an independent subset and on the full five-run setting; the selection is stable under a wider grid (Appendix Q). The single-target tracking (STT) result is therefore planner-side; distracted tracking (DT) and ambiguity tracking (AT) test conditional execution at runtime.

3.3 Target-State Risk Gate

The same waypoint can be safe when the target is clearly visible and dangerous when identity, range, or recent motion indicate a close and ambiguous situation. We therefore assemble a target-state vector x_t of 84 quantities, all computed online before the step executes (Appendix K): detector-side box geometry and confidence together with the top-2 identity margin, distance and bearing bins from a learned grounding probe, memory age and status flags, the previous command and the planner's predicted waypoints, image-plane flags, a visible-human count, and the step index. None of these reads simulator pose, geodesic distance, or the benchmark's following bands; those privileged quantities are excluded by construction, so the score is computable from what the deployed tracker already holds. In this benchmark, however, the candidate boxes and the target association are provided by the benchmark's own humanoid sensors rather than by a detector we train (§4.4).

A risk head f_ϕ maps x_t to a collision-risk score $r_t = f_\phi(x_t) \in [0, 1]$: a small feed-forward network of 28k parameters with auxiliary lost-target and failure outputs (Appendix C). It is fitted offline on distracted-tracking rollouts of earlier policies, with a positive label on the steps inside a fixed window preceding a collision termination, so the head is trained to anticipate contact rather than to report the current distance. The score ranks risk well but is not a calibrated probability, so τ is set from the trigger rate and the collision reduction rather than from a nominal probability level (Appendix J). At inference time the proposed waypoint is executed as

$$w_{1,t}^{\text{exec}} = \begin{cases} \big(\min(v_x, 0) + s_f \max(v_x, 0),\ s_l v_y,\ s_\omega \omega\big), & r_t \geq \tau, \\ w_{1,t}^{\text{base}}, & \text{otherwise.} \end{cases} \tag{2}$$

Whenever $r_t < \tau$, the deployed policy is identical to planner-head PPO; when the gate fires, only the current action is rescaled: yaw is kept ($s_\omega = 1$), lateral motion is attenuated, and forward motion is attenuated only when the command is forward, because slowing a retreat does not reduce contact. The gate therefore cannot introduce a motion the planner did not propose. The DT operating point is $(\tau, s_f, s_l, s_\omega) = (0.868, 0.25, 0.60, 1.00)$, calibrated on diagnostic trajectories; the score is learned rather than certified, so a high-risk step the head misses is executed unchanged. Because only the executed waypoint is edited and no policy parameter is touched, this operating point can be re-calibrated for a new setting without repeating the post-training. For AT we fit a separate risk head on AT rollouts and calibrate its own threshold and decision rule (Appendix L); the two operating points are denoted jointly as the gate. This design makes the mechanism testable, because the amount of attenuation can be held fixed while its timing is varied. The controls we report do exactly that: the same attenuation applied at every step, a random trigger matched to the gate's measured firing rate, and the gate's own score delayed in time so that it fires as often but at the wrong steps. If timing aligned with the target state does not matter, any of them should reproduce the gate's safety and success numbers.

4 Experiments

4.1 Setup

Benchmark. We evaluate on EVT-Bench (Wang et al., 2025). Single-target tracking (STT) has one language-specified target in a clean scene; distracted tracking (DT) adds agents with similar appearance and motion; ambiguity tracking (AT) relaxes the instruction so the policy must disambiguate the intended target. Each setting contributes 1,380 evaluated episodes per run (Appendix D), and episodes terminate on sustained following, collision, or step-budget exhaustion.

Baselines. Planner-head PPO (PPO below) is initialized from the supervised model and the gate is deployed on the resulting policy, so internal comparisons isolate post-training from the execution-time intervention. Panel (a) of Table 1 lists four prior trackers with the numbers published by TrackVLA, whose trained models are not available: IBVS (Gupta et al., 2017; Liu et al., 2023b), EVT (Zhong et al., 2024; OpenAI, 2024; Yang et al., 2023), and the 7B Uni-NaVid (Zhang et al., 2024a) and TrackVLA (Wang et al., 2025).

Metrics. We report success rate (SR), following rate (FR), and collision rate (CR) in percent, following the EVT-Bench protocol (Wang et al., 2025). FR is the pooled fraction of episode steps in the following band (the official aggregator's ratio), and CR is the fraction of episodes that end in collision. All compact-VLA numbers are obtained with the corrected full-episode evaluator (Appendix D); five-run results use five planner-head training runs evaluated under the same configuration.

Evaluation protocol. EVT-Bench's released evaluator leaks episode state across resets; correcting it shifts DT and AT metrics by several points, and all our rows use the corrected evaluator on the same 1,380 episodes (Appendix D). Even so, repeated evaluations of one unchanged policy differ by up to 1.4 CR and 2.7 SR points. Every TRACE-versus-PPO delta is therefore paired on the same episodes and the same five training runs, with episode-level bootstrap intervals and a sign-flip test on collisions (Table 2); the primary DT comparison is additionally evaluated with both arms concurrent and with null pairs whose arms are identical.

4.2 MAIN RESULTS

Planner-head post-training and the gate contribute separately (Table 1; panel (a) gives published numbers as context under different protocols, panel (b) is ours under one evaluator, the same episodes, and five training runs). Planner-head PPO raises SR and FR over the SFT base on every setting ($+8.19/+5.18$ on STT, $+2.61/+0.98$ on DT, $+4.54/+2.66$ on AT). On DT it also raises CR from 11.01 to 13.78. Adding the gate removes this collision cost without giving back the persistence: DT CR falls to 9.83, below the SFT base, SR and FR rise further, and the run-to-run spread of CR narrows from ± 1.31 to ± 0.49. On STT, where the risk score never crosses τ, per-run model selection adds $+1.06$ SR and $+0.92$ FR at unchanged CR (§3.2). On AT the gated row is the AT-calibrated gate.

Method	**STT** SR↑ / FR↑ / CR↓	**DT** SR↑ / FR↑ / CR↓	**AT** SR↑ / FR↑ / CR↓
(a) Reported by the original authors under their own protocols; not compared with (b).			
IBVS (Gupta et al., 2017)	42.9 / 56.2 / 3.75	10.6 / 28.4 / 6.14	15.2 / 39.5 / 4.90
EVT (Zhong et al., 2024)	32.5 / 49.9 / 40.5	15.7 / 35.7 / 53.3	18.3 / 21.0 / 44.9
Uni-NaVid (Vicuna-7B) (Zhang et al., 2024a)	25.7 / 39.5 / 41.9	11.3 / 27.4 / 43.5	8.26 / 28.6 / 43.7
TrackVLA (Vicuna-7B) (Wang et al., 2025)	85.1 / 78.6 / 1.65	57.6 / 63.2 / 5.80	50.2 / 63.7 / 17.1
(b) This work: one corrected evaluator, identical 1,380 *episodes; the PPO and* TRACE *rows share the same five training runs.*			
Qwen3-0.6B SFT base	78.26 / 75.70 / 4.13	41.38 / 54.46 / 11.01	59.13 / 69.82 / 7.39
+ planner-head PPO	86.45±1.30 / 80.88±0.79 / 4.09±0.40	43.99±0.50 / 55.44±0.66 / 13.78±1.31	63.67±1.04 / 72.48±0.30 / 8.74±0.65
+ TRACE	**87.51**±0.69 / **81.80**±0.30 / **3.91**±0.24†	**46.02**±0.70 / **57.58**±0.47 / **9.83**±0.49	**64.41**±1.35 / **72.57**±0.58 / **8.14**±0.46⋆
Paired effect of TRACE *over planner-head PPO (intervals in Table 2)*			
Δ (TRACE − PPO)	+1.06 / +0.92 / −0.17	**+2.03 / +2.14 / −3.96**	+0.74 / +0.09 / −0.59

Table 1: **EVT-Bench results** (SR/FR/CR, percent). Panel (a) reproduces numbers published by the original authors under their own protocols; panel (b) is measured by us under one corrected evaluator (Appendix D). Our claims are the paired deltas of the last row, computed from unrounded means (they can differ from differences of the rounded entries by 0.01). Bold: best within panel (b); ±: std over five training runs; the SFT base is a single supervised model. †STT uses per-run model selection, not the gate, which fires on 0% of STT steps (§3.2); the penalty level was chosen on half of the episodes, so the unbiased STT inference is the independent-subset row of Table 2. ⋆AT uses the AT-calibrated gate; its paired deltas did not reproduce under concurrent evaluation (Table 2).

The collision reduction holds on each of the five runs (9.3–10.6 against 12.75–15.22 for PPO; Figure 4b), and it is unchanged on the 281 episodes held out from fitting the risk head (ΔCR -4.13; Table 2). **Distracted tracking** carries the causal claim: SR $+2.03$, FR $+2.14$, and CR -3.96 are all significant (Table 2), and the collision reduction holds on every run (Appendix R). On ambiguity tracking the AT-calibrated gate stays within the benchmark's drift (Table 2). Figure 3 shows one DT episode step by step.

Paired statistics. All five DT runs improve on all three metrics, and each of the three gate intervals excludes zero (Table 2; the episode is the sampling unit, because the five runs share episodes). With both arms evaluated concurrently, ΔSR is $+2.71$, ΔFR $+2.36$, and ΔCR -3.77, against -0.29 for null pairs whose two arms are identical, whose interval covers zero (Appendix G).

4.3 THE COLLISION COST OF PPO

Both the persistence gain and the collision cost originate in the planner head. On distracted tracking PPO raises CR from 11.01 to 13.78, and every one of the five runs (12.75–15.22) ends above the base. On one matched run the endings shift without converting one-to-one (Table 3; Appendix R, Figure 7a–b). Because recognition is frozen, the shift is attributable to the planner persisting longer on the same identity estimates. Retuning the reward does not remove this cost (§4.4), so the gate intervenes at execution time, where the failure arises.

4.4 TARGET-STATE TIMING VERSUS GENERIC ATTENUATION

Matched attenuation controls. We compare TRACE with three controls that impose a comparable attenuation budget without using contemporaneous target-state risk (Ames et al., 2019; Dalal et al., 2018; Thananjeyan et al., 2021). The *average-attenuation* control applies a constant attenuation at every step, with its magnitude selected to match the mean attenuation induced by TRACE

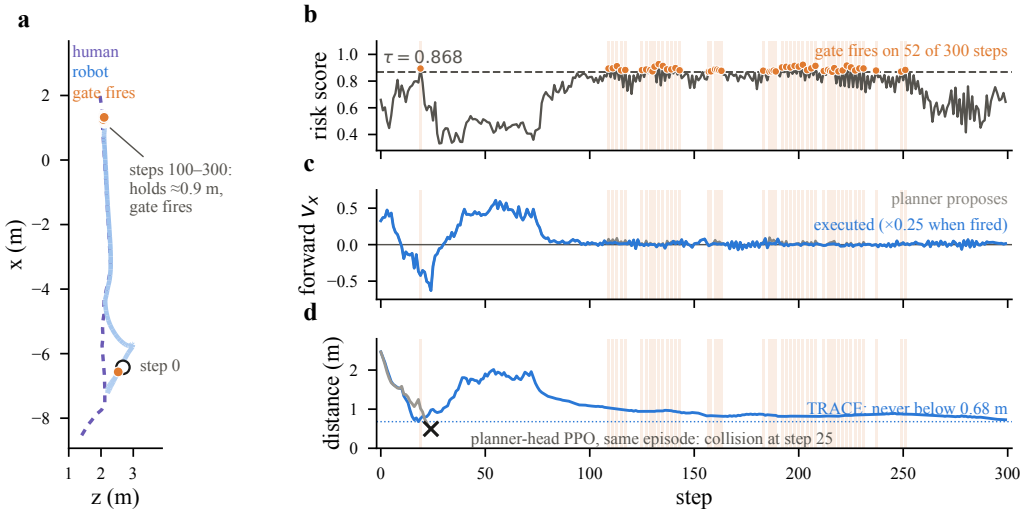

Figure 3: **One DT episode under TRACE, on a single time axis. a**, Bird's-eye paths of the human and the robot (robot path shaded light to dark with time); after step 100 the robot holds 0.7–1.1 m behind the human, and the fired steps cluster there. **b**, The risk score crosses τ=0.868 on 52 of 300 steps (shaded; a close-quarters episode chosen for illustration, the run-wide rate is below 2%). **c**, Forward component of the proposed and the executed waypoint: on the 47 fired steps moving forward the executed value is 0.25× the proposed one, and on the 5 reversing fired steps and all 248 unfired steps the two coincide. **d**, Distance to the human stays above 0.68 m for all 300 steps, whereas planner-head PPO on the same episode collides at step 25.

Setting	Intervention	ΔSR [95% CI]	ΔFR [95% CI]	ΔCR [95% CI]	p (coll.)
DT	risk gate	$+\mathbf{2.03}$ $[+0.90, +3.17]$	$+\mathbf{2.14}$ $[+1.22, +3.05]$	$-\mathbf{3.96}$ $[-4.78, -3.14]$	$< 10^{-4}$
AT	AT-calibrated gate, orig. eval.	$+0.74$ $[-0.22, +1.70]$	$+0.09$ $[-0.50, +0.67]$	-0.59 $[-1.16, -0.04]$	0.043
AT	concurrent re-eval.	-0.20 $[-1.13, +0.72]$	-0.47 $[-1.03, +0.09]$	-0.06 $[-0.65, +0.51]$	0.90
STT	model selection, full set	$+\mathbf{1.06}$ $[+0.49, +1.61]$	$+\mathbf{0.92}$ $[+0.57, +1.29]$	-0.17 $[-0.52, +0.17]$	0.36
STT	independent subset only	$+0.78$ $[0.00, +1.58]$	$+\mathbf{0.93}$ $[+0.43, +1.43]$	-0.20 $[-0.70, +0.29]$	0.49

Table 2: **Paired statistics against planner-head PPO, with the episode as the sampling unit.** The five runs share the same 1,380 episodes, so the 6,900 pairs are not independent: intervals are percentile bootstraps over episodes (10^4 resamples, each episode carrying its five paired outcomes) and p is a sign-flip permutation test (2×10^4 flips) on the per-episode mean collision difference; the estimand is conditional on these five runs. Per-run exact McNemar p values are in Table 26's caption. Bold: interval excludes zero; intervals are uncorrected for multiple comparisons. The AT collision result of the original evaluation is left unbolded because it did not reproduce when both arms were evaluated concurrently (second AT row, five pairs; Appendix L). On the 281 DT episodes whose rows were excluded from risk-head fitting (Appendix J) the collision reduction is unchanged (ΔCR -4.13 $[-6.26, -2.21]$, $p < 10^{-4}$) while SR and FR do not fall ($+0.07$ $[-2.35, +2.56]$, $+0.99$ $[-0.72, +2.79]$); the SR/FR gains are not resolved at that sample size. The second STT row restricts the test to the episodes outside each run's selection subset (715 distinct, 690 per run): the following gain remains significant and the success gain is borderline.

Base → PPO ending	normal	lost	collision	base total
normal	642	74	69	785
lost	109	307	30	446
collision	33	15	101	149
PPO total	784	396	200	1380

Table 3: **Episode-level ending transitions, compact base to planner-head PPO** (one matched run). 330 episodes change ending in both directions, for a net -50 lost and $+51$ collision; the added collisions come mostly from episodes the base ended normally (69) or lost (30).

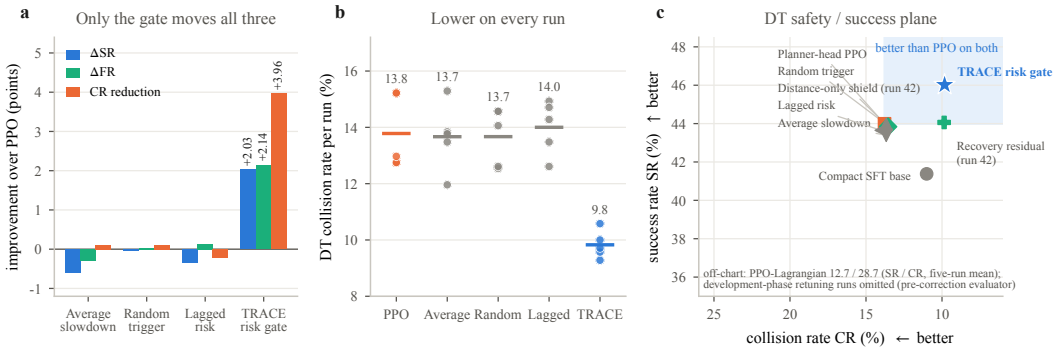

Figure 4: **TRACE against the DT alternatives, all under the corrected evaluator.** **a**, Only TRACE moves all three metrics ($+2.03$ SR, $+2.14$ FR, -3.96 CR); the matched attenuation controls stay near zero. **b**, Per-run CR: PPO and the three controls average 13.7–14.0, while TRACE is lower on every run (9.3–10.6, mean 9.83). **c**, The DT success/collision plane: the controls and the diagnostic filters cluster around PPO, and only TRACE reaches the high-SR, low-CR region. Single-run points are marked; all others are five-run means. PPO-Lagrangian lies off the chart and is quoted in the corner.

over each run. The *random-trigger* control applies the attenuation factors used by TRACE at uniformly sampled time steps, and the *lagged-risk* control applies them using a risk score delayed by 37 steps; both match the gate's intervention rate to within a few tenths of a percent, and the attenuation budgets are matched in expectation rather than on every trajectory. If generic motion attenuation or sparse intervention were sufficient, at least one control would reduce the collision rate while preserving the tracking metrics. Three findings follow from Table 4 and Figure 4. First, all three controls remain at the PPO reference on all three metrics, whereas TRACE moves all three. The same controls remain at the PPO reference on ambiguity tracking over 15 additional full evaluations (Appendix F; per-run results in Table 21). Second, a substantially stronger uniform attenuation reduces the collision rate to 6.01, but also decreases the success rate to 6.54 and the following rate to 24.25. The collision reduction therefore reflects degraded tracking rather than selective intervention. Third, a proximity-based rule operating at the gate's intervention rate changes the collision rate by only -0.68, with the reported interval including zero (Appendix H). These results indicate that the observed improvement depends on the temporal alignment between the intervention and the target-state risk, rather than on attenuation magnitude or intervention sparsity alone.

DT, five runs	**SR ↑**	**FR ↑**	**CR ↓**	**Intervention rate**
Planner-head PPO	43.99 ± 0.50	55.44 ± 0.66	13.78 ± 1.31	–
Average attenuation	43.39 ± 0.87	55.14 ± 0.62	13.67 ± 1.19	100%
Random-trigger attenuation	43.96 ± 0.76	55.47 ± 0.43	13.67 ± 1.02	1.6%
Lagged-risk attenuation (37 steps)	43.65 ± 0.99	55.57 ± 0.58	14.00 ± 0.96	0.8%
TRACE risk gate	$\mathbf{46.02 \pm 0.70}$	$\mathbf{57.58 \pm 0.47}$	$\mathbf{9.83 \pm 0.49}$	1.6%

Table 4: **Matched attenuation controls on DT.** SR, FR, and CR are reported in percent as mean±std over five runs on the same 1,380 episodes. The three controls are defined in the text; their intervention budgets are matched in expectation rather than on every trajectory.

Two further groups of alternatives appear in Figure 4c, with their numbers in Table 20. Modifying the scalar objective or the optimizer does not resolve the trade-off: doubling the PPO budget (Schulman et al., 2017; 2015) raises SR by 4.2 points and CR by 2.2 on a matched run; doubling the collision penalty (Achiam et al., 2017; Ray et al., 2019) raises CR by 7 points at unchanged SR; and neither critic-free REINFORCE nor PPO-Lagrangian reaches a usable policy. The two execution-time interventions fall short in different ways: a distance-only learned filter (Ames et al., 2019; Dalal et al., 2018; Thananjeyan et al., 2021) does not improve on PPO, and a supervised recovery residual (Johannink et al., 2019; Silver et al., 2018) reaches a collision rate comparable to TRACE at lower SR and with greater variation across runs. TRACE is the only evaluated method that improves tracking success and collision safety together.

Sparse firing concentrates on high-risk steps. The gate fires on 1.0–2.0% of steps per run (1.6% pooled; Appendix E, Figure 5), yet touches 31% of episodes, and the mean risk score on fired steps is 0.91 against 0.36–0.39 over all steps. On 68% of fired steps the command is forward and is scaled to $s_f{=}0.25$; yaw is preserved, so the robot keeps facing the target during the pause and resumes at full speed once the score drops below τ, which accounts for the rise in FR. Every feature group of the risk head contributes to collision prediction, with action context and identity margin contributing most (Appendix E), and the gate reduces collision-ending episodes on every one of the five runs (Appendix R). Across thresholds $\tau \in \{0.83, \ldots, 0.90\}$ CR stays below PPO's, whereas the joint SR/FR/CR gain is specific to the calibrated $\tau{=}0.868$ (Appendix N).

The gate slows near the human, not away from the scene. The gate slows the robot where the human is close (0.86 m median on fired steps against 1.54 m on unfired steps; Table 16), while per-step wall clearance matches PPO at every percentile (deltas ≤ 0.015 m; Appendix O).

What the target-state cues contribute. The head ranks risk well across label windows (ROC-AUC 0.78–0.84), with the identity group among its strongest inputs (Appendix E). Because the boxes and the association come from the benchmark's own sensors, the evidence concerns *when* to intervene rather than a perception stack: with a learned COCO person detector and the probe's own association in their place, the effect is about one CR point (Appendix G).

4.5 Robustness within EVT-Bench

Within the released environment we examine four axes (Appendix I). The paired DT effect holds on both scene corpora, on success as well as on collisions (HM3D (Ramakrishnan et al., 2021) ΔSR $+2.11$ / ΔCR -4.02 over 3,360 pairs, MP3D (Chang et al., 2017) $+1.95$ / -3.90 over 3,540). Under three alternative collision geometries (0.10/0.15/0.20 m lateral offsets against the default 0.25 m) the CR reduction keeps its direction (-1.03/-2.15/-0.72 against -3.96). Transferring the DT operating point to AT *unchanged* lowers CR on all five runs (8.01 against 8.74) at a one-point SR/FR cost that task-specific calibration recovers, and the head stack trains at three backbone scales with no backbone-dependent gap.

5 Conclusion

PPO on the planner head of a frozen-backbone 0.6B VLA tracker raises success and following rates on every EVT-Bench setting, but on distracted tracking part of the added persistence turns into collisions. A target-state risk gate at execution time (TRACE) removes this cost without removing the persistence: collisions fall by 3.96 points while success and following rates keep rising, and matched attenuation controls without target-state timing do not reproduce this. The effect is about three times the benchmark's re-evaluation drift and reproduces under concurrent evaluation. Because the gate reads only quantities the tracker already computes and leaves the policy untouched, its operating point can be recalibrated for a new setting without retraining the planner. Target-state-aware execution therefore recovers the safety cost of policy post-training with minimal intervention.

AI USE STATEMENT

In this work, we used generative AI tools only to edit author-written text for grammar, style, and readability. The authors reviewed all AI-assisted revisions and take full responsibility for the final manuscript, including its claims, citations, and experimental reporting.

ETHICS STATEMENT

This work studies robots that follow people, and the deployment framing carries risks that grow rather than shrink with the method's success. A tracker that is better at persisting is also better at following someone who does not wish to be followed, and the risk head depends on identity estimates that can fail asymmetrically across appearance groups; we have not audited that asymmetry and cannot claim it is absent. All experiments here run in simulation with synthetic humanoid avatars, so no human subjects or personal data are involved, but the collision rates we report are simulator contact events and must not be read as a physical-safety guarantee for a real platform. The gate offers no formal guarantee and fails silently when it does not fire.

REPRODUCIBILITY STATEMENT

Section 4.1 describes the benchmark and evaluation settings. Appendices B, C, and D document model architectures, hyperparameters, training procedures, and evaluation implementation details, including the evaluator correction. Risk-head data preparation and partitions are detailed in Appendix J. Risk-feature definitions and gate thresholds appear in Section 3.3 and Appendices K and L, with sensitivity analyses and additional diagnostics in Appendices N and E. Results measured on the diagnostic set used to calibrate the gate and on the STT selection subsets are explicitly identified. Training and evaluation code is included in the supplementary material and will be publicly released.

REFERENCES

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *ICML*, 2017.

Aaron D. Ames, Samuel Coogan, Magnus Egerstedt, Gennaro Notomista, Koushil Sreenath, and Paulo Tabuada. Control barrier functions: Theory and applications. In *ECC*, 2019.

Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton van den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *CVPR*, 2018.

David Arthur and Sergei Vassilvitskii. k-means++: The advantages of careful seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pp. 1027–1035, 2007.

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

Andrea Bajcsy, Antonio Loquercio, Ashish Kumar, and Jitendra Malik. Learning vision-based pursuit-evasion robot policies. In *ICRA*, 2024.

Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *ICIP*, 2016.

Goutam Bhat, Martin Danelljan, Luc Van Gool, and Radu Timofte. Learning discriminative model prediction for tracking. In *ICCV*, 2019.

Kevin Black, Noah Brown, Danny Driess, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

Kevin Black, Noah Brown, James Darpinian, et al. $\pi_{0.5}$: A vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*, 2025.

Anthony Brohan, Noah Brown, Justice Carbajal, et al. Rt-1: Robotics transformer for real-world control at scale. In *RSS*, 2023a.

Anthony Brohan, Noah Brown, Justice Carbajal, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *CoRL*, 2023b.

Yihan Cao, Jiazhao Zhang, Zhinan Yu, Shuzhen Liu, Zheng Qin, Qin Zou, Bo Du, and Kai Xu. Cognav: Cognitive process modeling for object goal navigation with llms. *arXiv preprint arXiv:2412.10439*, 2024.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021.

Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niebner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. In *3DV*, 2017.

Devendra Singh Chaplot, Dhiraj P. Gandhi, Abhinav Gupta, and Ruslan Salakhutdinov. Object goal navigation using goal-oriented semantic exploration. In *NeurIPS*, 2020.

An-Chieh Cheng, Yandong Ji, Zhaojing Yang, Zaitian Gongye, Xueyan Zou, Jan Kautz, Erdem Bıyık, Hongxu Yin, Sifei Liu, and Xiaolong Wang. Navila: Legged robot vision-language-action model for navigation. *arXiv preprint arXiv:2412.04453*, 2024.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 2024.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, et al. Vicuna: An open-source chatbot impressing gpt-4 with 90% chatgpt quality, 2023. https://lmsys.org/blog/2023-03-30-vicuna/.

Gal Dalal, Krishnamurthy Dvijotham, Matej Vecerik, Todd Hester, Cosmin Paduraru, and Yuval Tassa. Safe exploration in continuous action spaces. *arXiv preprint arXiv:1801.08757*, 2018.

Alessandro Devo, Alberto Dionigi, and Gabriele Costante. Enhancing continuous control of mobile robots for end-to-end visual active tracking. *Robotics and Autonomous Systems*, 142:103799, 2021.

Yan Ding, Haoran Geng, Chuanruo Xu, Xiaomeng Fang, Jiazhao Zhang, Songlin Wei, Qiyu Dai, Zhizheng Zhang, and He Wang. Open6dor: Benchmarking open-instruction 6-dof object rearrangement and a vlm-based approach. In *IROS*, 2024.

Anthony Francis, Claudia Pérez-d'Arpino, Chengshu Li, Fei Xia, Alexandre Alahi, Rachid Alami, Aniket Bera, Abhijat Biswas, Joydeep Biswas, Rohan Chandra, et al. Principles and guidelines for evaluating social robot navigation algorithms. *ACM Transactions on Human-Robot Interaction*, 14(2):1–65, 2025.

Gemini Team. Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Meenakshi Gupta, Swagat Kumar, Laxmidhar Behera, and Venkatesh K. Subramanian. A novel vision-based tracking algorithm for a human-following mobile robot. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(7):1415–1427, 2017.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *ICML*, 2018.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

Md Mofijul Islam, Alexi Gladstone, Riashat Islam, and Tariq Iqbal. Eqa-mx: Embodied question answering using multimodal expression. In *ICLR*, 2024.

Tobias Johannink, Shikhar Bahl, Ashvin Nair, Jianlan Luo, Avinash Kumar, Matthias Loskyll, Juan Aparicio Ojea, Eugen Solowjow, and Sergey Levine. Residual reinforcement learning for robot control. In *ICRA*, 2019.

Tsung-Wei Ke, Nikolaos Gkanatsios, and Katerina Fragkiadaki. 3d diffuser actor: Policy diffusion with 3d scene representations. *arXiv preprint arXiv:2402.10885*, 2024.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, et al. Segment anything. In *ICCV*, 2023.

Richard L. Knoblauch, Martin T. Pietrucha, and Marsha Nitzburg. Field studies of pedestrian walking speed and start-up time. *Transportation Research Record*, 1538(1):27–38, 1996.

Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *ECCV*, 2020.

Alexander Ku, Peter Anderson, Roma Patel, Eugene Ie, and Jason Baldridge. Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. In *EMNLP*, 2020.

Jing Li, Jing Xu, Fangwei Zhong, Xiangyu Kong, Yu Qiao, and Yizhou Wang. Pose-assisted multi-camera collaboration for active object tracking. In *AAAI*, 2020.

Qixiu Li, Yaobo Liang, Zhuohao Wang, Lin Luo, Xinpeng Chen, Mingkun Liao, Fangyun Wei, Yu Deng, Sijie Xu, Yu Zhang, et al. Cogact: A foundational vision-language-action model for synergizing cognition and action in robotic manipulation. *arXiv preprint arXiv:2411.19650*, 2024.

Yanwei Li, Chengyao Wang, and Jiaya Jia. Llama-vid: An image is worth 2 tokens in large language models. *arXiv preprint arXiv:2311.17043*, 2023.

Bencheng Liao, Shaoyu Chen, Haoran Yin, Bo Jiang, Cheng Wang, Shenfeng Yan, Xinbang Zhang, Xiangyu Li, Ying Zhang, Qian Zhang, et al. Diffusiondrive: Truncated diffusion model for end-to-end autonomous driving. *arXiv preprint arXiv:2411.15139*, 2024.

Haoran Liu, Shaoan Wang, Minghan Li, Jiazhao Zhang, Songlin Wei, Anyang Li, Zichao Wang, Botao Yang, and He Wang. TrackVLA++: Unleashing reasoning and memory capabilities in VLA models for embodied visual tracking. *arXiv preprint arXiv:2510.07134*, 2025.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023a.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023b.

Yuxing Long, Wenzhe Cai, Hongcheng Wang, Guanqi Zhan, and Hao Dong. Instructnav: Zero-shot system for generic instruction navigation in unexplored environment. *arXiv preprint arXiv:2406.04882*, 2024.

Wenhan Luo, Peng Sun, Fangwei Zhong, Wei Liu, Tong Zhang, and Yizhou Wang. End-to-end active object tracking via reinforcement learning. In *ICML*, 2018.

Wenhan Luo, Peng Sun, Fangwei Zhong, Wei Liu, Tong Zhang, and Yizhou Wang. End-to-end active object tracking and its real-world deployment via reinforcement learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(6):1317–1332, 2019.

Alaa Maalouf, Ninad Jadhav, Krishna Murthy Jatavallabhula, Makram Chahine, Daniel M. Vogt, Robert J. Wood, Antonio Torralba, and Daniela Rus. Follow anything: Open-set detection, tracking, and following in real-time. *IEEE Robotics and Automation Letters*, 9(4):3283–3290, 2024.

Arjun Majumdar, Anurag Ajay, Xiaohan Zhang, Pranav Putta, Sriram Yenamandra, Mikael Henaff, Sneha Silwal, Paul Mcvay, Oleksandr Maksymets, Sergio Arnaud, et al. Openeqa: Embodied question answering in the era of foundation models. In *CVPR*, 2024.

Christoforos Mavrogiannis, Francesca Baldini, Allan Wang, Dapeng Zhao, Pete Trautman, Aaron Steinfeld, and Jean Oh. Core challenges of social robot navigation: A survey. *ACM Transactions on Human-Robot Interaction*, 12(3):1–39, 2023.

OpenAI. Hello gpt-4o, 2024. `https://openai.com/index/hello-gpt-4o/`.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, 2023.

Jan Peters and Stefan Schaal. Reinforcement learning by reward-weighted regression for operational space control. In *ICML*, 2007.

Xavier Puig, Eric Undersander, Andrew Szot, Mikael Dallaire Cote, Tsung-Yen Yang, Ruslan Partsey, Ruta Desai, Alexander William Clegg, Michal Hlavac, So Yeon Min, et al. Habitat 3.0: A co-habitat for humans, avatars and robots. *arXiv preprint arXiv:2310.13724*, 2023.

Delin Qu, Haoming Song, Qiqi Chen, Yuxuan Yao, Xueting Ye, Yiding Ding, Zhibing Wang, Jingbo Gu, Bin Zhao, Donglin Wang, et al. Spatialvla: Exploring spatial representations for visual-language-action model. *arXiv preprint arXiv:2501.15830*, 2025.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*, 2023.

Santhosh K. Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alexander Clegg, John M. Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X. Chang, et al. Habitat-matterport 3d dataset (hm3d): 1000 large-scale 3d environments for embodied ai. In *NeurIPS Datasets and Benchmarks*, 2021.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.

Alex Ray, Joshua Achiam, and Dario Amodei. Benchmarking safe exploration in deep reinforcement learning. *arXiv preprint arXiv:1910.01708*, 2019.

Pascal Roth, Julian Nubert, Fan Yang, Mayank Mittal, and Marco Hutter. Viplanner: Visual semantic imperative learning for local navigation. In *ICRA*, 2024.

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 *ICCV*, 2019.

John Schulman, Philipp Moritz, Sergey Levine, Michael I. Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015.

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

Luca Scofano, Alessio Sampieri, Tommaso Campari, Valentino Sacco, Indro Spinelli, Lamberto Ballan, and Fabio Galasso. Following the human thread in social navigation. *arXiv preprint arXiv:2404.11327*, 2024.

Dhruv Shah, Arjun Bhorkar, Hardo Leen, Ilya Kostrikov, Nicholas Rhinehart, and Sergey Levine. Offline reinforcement learning for visual navigation. *arXiv preprint arXiv:2212.08244*, 2022.

Dhruv Shah, Ajay Sridhar, Arjun Bhorkar, Noriaki Hirose, and Sergey Levine. Gnm: A general navigation model to drive any robot. In *ICRA*, 2023.

Zhihong Shao, Peiyi Wang, Qihao Zhu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Tom Silver, Kelsey R. Allen, Joshua B. Tenenbaum, and Leslie Pack Kaelbling. Residual policy learning. *arXiv preprint arXiv:1812.06298*, 2018.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021.

Ajay Sridhar, Dhruv Shah, Catherine Glossop, and Sergey Levine. Nomad: Goal masked diffusion policies for navigation and exploration. In *ICRA*, 2024.

Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023.

Andrew Szot, Alexander Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Singh Chaplot, Oleksandr Maksymets, et al. Habitat 2.0: Training home assistants to rearrange their habitat. In *NeurIPS*, 2021.

Brijen Thananjeyan, Ashwin Balakrishna, Suraj Nair, Michael Luo, Krishnan Srinivasan, Minho Hwang, Joseph E. Gonzalez, Julian Ibarz, Chelsea Finn, and Ken Goldberg. Recovery RL: Safe reinforcement learning with learned recovery zones. *IEEE Robotics and Automation Letters*, 6 (3):4915–4922, 2021.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Jur van den Berg, Stephen J. Guy, Ming Lin, and Dinesh Manocha. Reciprocal n-body collision avoidance. In *Robotics Research*, 2011.

Shaoan Wang, Jiazhao Zhang, Minghan Li, Jiahang Liu, Anqi Li, Kui Wu, Fangwei Zhong, Junzhi Yu, Zhizheng Zhang, and He Wang. Trackvla: Embodied visual tracking in the wild. *arXiv preprint arXiv:2505.23189*, 2025.

Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *ICIP*, 2017.

Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv preprint arXiv:2310.11441*, 2023.

Hanjing Ye, Jieting Zhao, Yu Zhan, Wenhao Chen, Lin He, and Hong Zhang. Person re-identification for robot person following with online continual learning. *IEEE Robotics and Automation Letters*, 2024.

Hanjing Ye, Kuanqi Cai, Yu Zhan, Bingyi Xia, Arash Ajoudani, and Hong Zhang. Rpf-search: Field-based search for robot person following in unknown dynamic environments. *arXiv preprint arXiv:2503.02188*, 2025.

Mang Ye, Jianbing Shen, Gaojie Lin, Tao Xiang, Ling Shao, and Steven C. H. Hoi. Deep learning for person re-identification: A survey and outlook. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(6):2872–2893, 2021.

Kuo-Hao Zeng, Zichen Zhang, Kiana Ehsani, Rose Hendrix, Jordi Salvador, Andrei Herrasti, Ross Girshick, Aniruddha Kembhavi, and Luca Weihs. Poliformer: Scaling on-policy rl with transformers results in masterful navigators. *arXiv preprint arXiv:2406.20083*, 2024.

Jiazhao Zhang, Liu Dai, Fanpeng Meng, Qingnan Fan, Xun Chen, Kai Xu, and He Wang. 3d-aware object goal navigation via simultaneous exploration and identification. In *CVPR*, 2023.

Jiazhao Zhang, Kunyu Wang, Shaoan Wang, Minghan Li, Haoran Liu, Songlin Wei, Zichao Wang, Zhizheng Zhang, and He Wang. Uni-navid: A video-based vision-language-action model for unifying embodied navigation tasks. *arXiv preprint arXiv:2412.06224*, 2024a.

Jiazhao Zhang, Kunyu Wang, Rongtao Xu, Gengze Zhou, Yicong Hong, Xiaomeng Fang, Qi Wu, Zhizheng Zhang, and He Wang. Navid: Video-based vlm plans the next step for vision-and-language navigation. In *RSS*, 2024b.

Wei Zhang, Kechen Song, Xuanyu Rong, and Yunhui Li. Coarse-to-fine uav target tracking with deep reinforcement learning. *IEEE Transactions on Automation Science and Engineering*, 16(4): 1522–1530, 2018.

Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. In *ECCV*, 2022.

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

Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *RSS*, 2023.

Fangwei Zhong, Peng Sun, Wenhan Luo, Tingyun Yan, and Yizhou Wang. Ad-vat: An asymmetric dueling mechanism for learning visual active tracking. In *ICLR*, 2019.

Fangwei Zhong, Peng Sun, Wenhan Luo, Tingyun Yan, and Yizhou Wang. Ad-vat+: An asymmetric dueling mechanism for learning and understanding visual active tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(5):1467–1482, 2021a.

Fangwei Zhong, Peng Sun, Wenhan Luo, Tingyun Yan, and Yizhou Wang. Towards distraction-robust active visual tracking. In *ICML*, 2021b.

Fangwei Zhong, Xiao Bi, Yixin Zhang, Wenjia Zhang, and Yizhou Wang. Rspt: Reconstruct surroundings and predict trajectory for generalizable active object tracking. In *AAAI*, 2023.

Fangwei Zhong, Kui Wu, Hai Ci, Churan Wang, and Hao Chen. Empowering embodied visual tracking with visual foundation models and offline rl. In *European Conference on Computer Vision (ECCV)*, 2024.

Yifan Zhong, Xueyi Huang, Ruihai Li, Chenyu Zhang, Yuxin Liang, Yaodong Yang, and Yuanpei Chen. Dexgraspvla: A vision-language-action framework towards general dexterous grasping. *arXiv preprint arXiv:2502.20900*, 2025.

Gengze Zhou, Yicong Hong, Zun Wang, Xin Eric Wang, and Qi Wu. Navgpt-2: Unleashing navigational reasoning capability for large vision-language models. In *ECCV*, 2024.

A Extended Related Work

Embodied tracking, following, and social navigation. Beyond the lineages cited in §2, adversarial training with a learned target (Zhong et al., 2021a) and the environment released with EVT (Zhong et al., 2024) shaped the benchmark we use. Decoupled designs have also been built on association-based trackers (Zhang et al., 2022), open-vocabulary detection (Liu et al., 2023b), coarse-to-fine visual tracking (Zhang et al., 2018), transformer navigation policies (Zeng et al., 2024), and pose-assisted control (Li et al., 2020). Adjacent problems share the same difficulty of keeping a specific person: person-following with re-identification (Ye et al., 2024; 2025), social navigation (Mavrogiannis et al., 2023; Francis et al., 2025; Scofano et al., 2024), and pursuit–evasion (Bajcsy et al., 2024).

Post-training objectives and safe control. Off-policy actor–critic methods (Haarnoja et al., 2018) and preference- or imitation-style objectives (Peters & Schaal, 2007; Rafailov et al., 2023; Shao et al., 2024) are the other common post-training routes; we report attempts with several of them in Appendix P. The embodied controllers post-trained this way include navigation and manipulation policies (Shah et al., 2022; Devo et al., 2021; Brohan et al., 2023a; Black et al., 2025; Li et al., 2024; Qu et al., 2025). Residual policies for safety also have a supervised variant (Silver et al., 2018), which we evaluate as a diagnostic in Appendix P.

Pretrained components and simulation infrastructure. The re-identification literature includes the association baselines (Bewley et al., 2016) and surveys (Ye et al., 2021) behind the identity quantities we consume. Frozen backbones of this kind include (Sun et al., 2023; Caron et al., 2021; Kirillov et al., 2023; Ravi et al., 2024; Yang et al., 2023; OpenAI, 2024; Touvron et al., 2023; Chiang et al., 2023; Gemini Team, 2023; Liu et al., 2023a; Li et al., 2023), and the action-head literature includes (Song et al., 2021; Peebles & Xie, 2023; Sridhar et al., 2024; Ke et al., 2024; Liao et al., 2024). The simulator and benchmark conventions we inherit come from Habitat (Szot et al., 2021), vision-language navigation (Ku et al., 2020; Zhang et al., 2024c; Zhou et al., 2024; Long et al., 2024), object navigation and embodied question answering (Chaplot et al., 2020; Zhang et al., 2023; Cao et al., 2024; Islam et al., 2024; Majumdar et al., 2024), navigation foundation models (Shah et al., 2023; Zhang et al., 2024b; Cheng et al., 2024), and related embodied settings (Ding et al., 2024; Roth et al., 2024; Zhong et al., 2025). Pedestrian-speed conventions (Knoblauch et al., 1996), reciprocal collision avoidance (van den Berg et al., 2011), and k-means (Arthur & Vassilvitskii, 2007) are used in the benchmark's motion model and in our diagnostics.

B Training Details

Supervised tracker. The supervised tracker follows the TrackVLA recipe: per-frame vision tokens from a frozen SigLIP-so400m-patch14-384 encoder concatenated with DINO-family tokens ($1{,}536$ dimensions in total) are projected by a recognition adapter into a frozen Qwen3-0.6B language backbone, and a planner head emits 8×3 waypoints. The head stack is trained by supervised fine-tuning for 2 epochs at batch size 8 and learning rate 2×10^{-5} on the original tracking corpus (about 1.4M samples; the 183k-sample expert-rollout corpus of Appendix I is about 13% of it), with the encoder and backbone frozen throughout.

Backbone and freeze policy. The TrackVLA-style model couples a vision encoder, a language backbone, and a planner head producing 8×3 waypoints. For post-training we freeze the vision encoder, the language backbone, and the recognition adapter projecting visual tokens into the language space. Only the planner head, the stochastic wrapper around it, and a learned value head receive gradients.

Optimizer, rollout, and reward. We use AdamW (learning rate 10^{-4}, weight decay 0.01, grad-norm clip 1.0) with a constant schedule; PPO uses clip $\epsilon = 0.2$, value-loss coefficient 0.5, entropy coefficient 0.01, GAE $\gamma = 0.99$, $\lambda = 0.95$, four PPO epochs per round, and mini-batch size 32. The stochastic wrapper is initialized at $\log \sigma = -2.0$ on the full 8×3 waypoint head. Each round collects 100 on-policy episodes; the two-round main run uses 200 episodes total. The reward of Equation 1 is a per-step sum: follow / collision indicators read EVT-Bench's follow-distance and

contact flags, d_t is the simulator-reported geodesic distance, and the far penalty fires when $d_t > 4\,\text{m}$. Table 5 consolidates these settings with the evaluation budgets.

Hardware and wall-clock. Training and the full evaluations were run on a server with eight `NVIDIA A100` (40 GB) GPUs. Planner-head PPO (200 episodes, 2 rounds) takes $\sim$2 h on one GPU. A full 1,380-episode DT evaluation is split into 30 episode partitions; each partition worker uses one GPU, and with the eight GPUs sharing the partitions a full evaluation completes in $\sim$3.5 h, with or without the gate. The gate runs the risk head on every step — the score determines whether the step fires — and on fired steps additionally rescales the executed waypoint in place. The deployed head is a 28k-parameter multilayer perceptron (MLP) ($84 \to 128 \to 128 \to 3$ with layer normalization); a batch-1 forward pass takes $\sim$52 μs in FP16 on an `RTX PRO 6000` datacenter GPU and $\sim$145 μs on one CPU core, measured over 3,000 timed calls of the deployed architecture. Both are far below the 100 ms period of the 10 Hz controller, but neither is an end-to-end latency measurement on a robot platform, which we have not performed.

C Risk Head and Recovery Residual Architecture

Inputs and heads. The risk head and the analysis-only recovery residual head share a deployable target-state input (Appendix K): detector-side box geometry and confidence, top-2 identity margin, probe outputs including distance and bearing bins, memory age and status flags, the previous command, and the planner's predicted waypoints. The risk head is a two-layer MLP with hidden width 128, layer normalization, GELU activations and dropout 0.10 ($84 \to 128 \to 128 \to 3$, 28k parameters); it emits a collision-risk score through a sigmoid, with auxiliary lost and failure outputs under the same trunk. The residual head shares the design and emits a three-dimensional residual on (v_x, v_y, ω). Training uses AdamW (weight decay 1×10^{-4}): the risk head with a balanced positive sampler for up to 12 epochs, the residual head with MSE against ground-truth deltas (Appendix S) for up to 80 epochs, both with early stopping on held-out validation. The residual is not part of TRACE and is reported only as analysis in Appendix T (Figure 8a).

Online integration. At deployment, the risk head consumes the same per-step target-state quantities that the diagnostic evaluation records. The gate reads the risk score r_t, applies threshold τ, and scales the executed waypoint by (s_f, s_l, s_ω) when triggered; the residual head, when run for analysis, instead adds its clipped residual to the executed waypoint.

D Evaluation Protocol Details

Episode count and metrics. Each EVT-Bench task setting contains 1,380 episodes. SR is the fraction of episodes that satisfy the EVT-Bench success condition; FR is the fraction of timesteps in which the agent satisfies the follow-distance condition; CR is the fraction of episodes ending in collision.

Leakage-free episode resets. The corrected evaluator clears all episode-local state at episode boundaries, including coarse history tokens, the frame buffer, local controller state, the last executed action, and the last predicted trajectory. Planner failure (e.g., a tokenizer exception) is mapped to a stop action so that the controller does not run on stale velocity commands. We adopted this convention after observing that residual state across episode boundaries shifts metrics by several points on DT and AT in particular, where the policy is supposed to re-acquire the correct target identity at the start of each new episode. All internal deltas in the headline tables compare same-protocol numbers under this convention, and numbers from the released (leaking) evaluator are never mixed in.

Reproducibility of the evaluator, and the resolution it allows. The evaluator is not bit-reproducible: the tracker's GPU forward pass is not deterministic, so repeated evaluations of one policy execute slightly different commands and diverge. Three independent evaluations of planner-head PPO run 42 under one code version give $41.88/54.50/14.06$, $42.54/54.19/13.48$ and $42.83/54.50/12.83$; pairwise, the same episode ends the same way in 75.3–77.8% of cases (90–91% agreement on the collision flag alone), and the paired CR differences between these identical

arms are -0.58, -1.23 and -0.65 points, each with an episode-bootstrap interval of about ± 1.6 points. Outcomes also drift across code versions: the September re-evaluation differs from the April $44.57/56.41/12.75$ of Table 1, agreeing on 74.6% of episodes.

Two consequences. First, every paired comparison must be made between evaluations under one code version, which is how the supplementary experiments of Appendices G and H are built. Second, with one evaluation per arm per run, differences of about one CR point or two SR points are not resolvable. Three lines of evidence: a mis-configured gate that fires on 0.18% of steps measures -1.07 CR points over four runs, the same as the learned-detector gate's -1.09; a null pair whose two arms are the identical configuration, run concurrently, moves CR by -1.38; and two identical gate-off evaluations of one run differ in SR by -2.68 with an episode-bootstrap interval of $[-4.93, -0.43]$ that excludes zero. The episode bootstrap therefore states the uncertainty of one pair of evaluations, not the uncertainty of the quantity those evaluations estimate. Effects of that size in this paper are reported as unresolved. The DT risk gate's effect is three times larger, holds on every run in both April (-3.96) and September (-3.28, Appendix G), and concentrates where the gate fires, so it is not explained by this noise.

Seed convention. Five-seed full evaluations use seeds $\{42, 43, 44, 45, 46\}$, one per independent planner-head training run; "seed" and "run" therefore name the same unit throughout the paper. Each seed runs the same $1{,}380$ episodes; metrics across seeds are aggregated as mean and standard deviation. All paired-seed comparisons (e.g., PPO vs. gated PPO) use the same episode order and the same evaluator seed on both sides. Single-seed full evaluations are tagged with the specific seed and compared only against the same seed of the matched baseline.

Logging. The evaluator writes per-episode JSON records with episode ID, success flag, collision flag, follow timesteps, target-state trace, and per-step diagnostic fields. These records are reused without modification by the ending-taxonomy, risk, and residual analyses, so that the tables below draw from the same evaluation.

Module	Hyperparameter	Value
PPO planner-head	Learning rate (planner / value)	1×10^{-4}
	Weight decay	0.01
	Clip ϵ	0.2
	Discount γ	0.99
	GAE λ	0.95
	Value-loss coefficient	0.5
	Entropy coefficient	0.01
	PPO epochs / mini-batch	4 / 32
Stochastic wrapper	Initial $\log \sigma$	-2.0
	Action shape	8×3 waypoints
	Action parameterization	Planar waypoints
	Grad-norm clip	1.0
Reward	w_f (follow)	1.0
	w_c (collide)	5.0
	w_d (distance shaping)	0.5
	w_{far}, d_{far}	0.3, 4.0 m
Rollout budget	Episodes per round	100
	Rounds	2
	Total episodes	200
Risk head	Hidden / depth / dropout	128 / 2 / 0.10
	Weight decay	1×10^{-4}
	Epochs (max)	12
	DT $(\tau, s_f, s_l, s_\omega)$	$(0.868, 0.25, 0.60, 1.00)$
Evaluation	Episodes per evaluation setting	1,380
	Seeds (full eval)	$\{42, 43, 44, 45, 46\}$

Table 5: **Centralized hyperparameters.** All values are those used for the headline numbers. Recovery-residual hyperparameters used in the bottleneck analysis (Appendix T) are listed separately in Appendix S.

E Gate Activation Profile and Risk-Feature Ablation

This appendix collects the three diagnostic figures behind §4.4. Figure 4 places every DT alternative evaluated under the corrected evaluator on the safety/success plane; Figure 5 gives the gate's activation profile from the deployed step logs of the five runs; Figure 6 reports the offline feature ablation of the risk head.

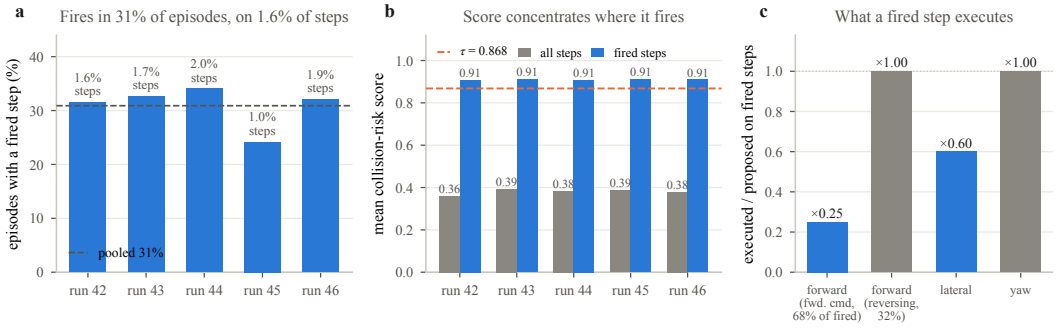

Figure 5: **Gate activation profile on DT, from the deployed step logs of the five runs. a**, Share of episodes on which the gate fires at least once (bars: 24–34%, 31% pooled) and share of steps on which it fires (printed: 1.0–2.0%, 1.6% pooled over 741,278 steps). **b**, Mean collision-risk score on all steps against fired steps (0.36–0.39 against 0.91; dashed line τ=0.868). **c**, What a fired step executes: a forward command (68% of fired steps) is scaled to 0.25, a reversing command is left at 1.0, lateral motion is scaled to 0.60, and yaw is unchanged.

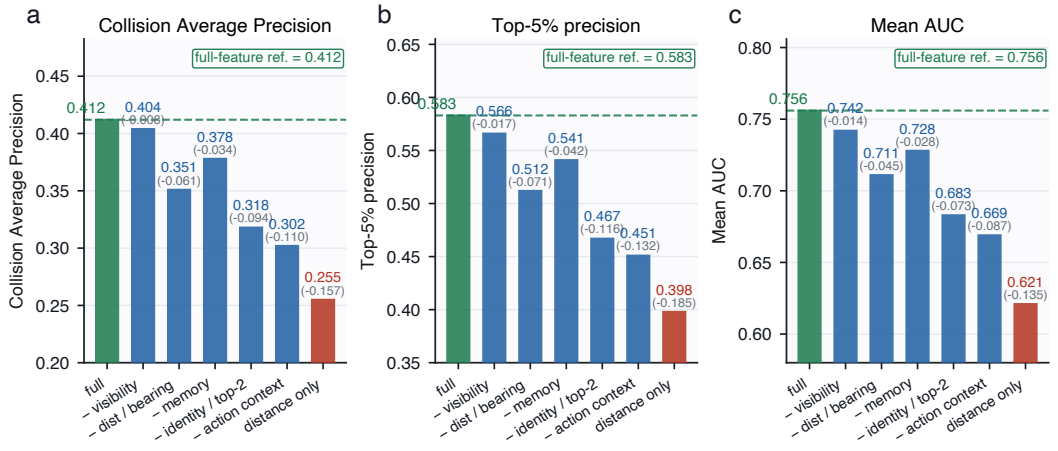

Figure 6: **Offline risk-feature ablation.** Collision-ranking quality—average precision, top-5% precision, and mean AUC—as inputs to the risk head are removed. Dropping *action context* or *identity/top-2* hurts most, and a *distance-only* head is weakest; the ordering is consistent across all three metrics. Dashed lines mark the full-feature reference.

F Matched Controls on Ambiguity Tracking

Distracted tracking carries the matched controls of §4.4; to give ambiguity tracking the same treatment we ran the same three controls on AT with the calibrated gate's risk head, threshold, and scales — average slowdown at every step, random-trigger slowdown at approximately each run's measured gate rate (realized 0.83–1.23% of steps against the gate's 0.76–1.15%), and lagged-risk slowdown reading the gate's score delayed by 37 steps, which on AT crosses the threshold on 3.4–6.7% of steps, three to six times the gate's own rate — for 15 further full 1,380-episode evaluations (Table 6). All three stay at the PPO reference, with CR within −0.13/+0.07 of 8.74 and SR at or

below PPO. In the original evaluation the gate lowered CR with non-decreasing mean SR and FR, but this margin did not reproduce when both arms were evaluated concurrently (Appendix L), so on AT neither the controls nor the gate shows an effect that the benchmark can resolve. Per-run values (Table 7) show that the original CR reduction came from runs 42 and 43 and the SR gain from runs 43 and 45, consistent with the marginal ambiguity-tracking intervals in Table 2.

AT, five runs	SR ↑	FR ↑	CR ↓	Intervention rate
Planner-head PPO	63.67 ± 1.04	72.48 ± 0.30	8.74 ± 0.65	–
Average attenuation	63.46 ± 0.88	71.99 ± 0.70	8.81 ± 0.50	100%
Random-trigger attenuation	63.62 ± 1.84	72.27 ± 1.04	8.61 ± 0.95	0.83–1.23%
Lagged-risk attenuation (37 steps)	62.91 ± 1.77	71.78 ± 1.10	8.62 ± 1.05	3.4–6.7%
TRACE calibrated gate	$\mathbf{64.41 \pm 1.35}$	$\mathbf{72.57 \pm 0.58}$	$\mathbf{8.14 \pm 0.46}$	0.76–1.15%

Table 6: **Matched slowdown controls on AT.** SR/FR/CR in percent, mean±std over the five training runs on the identical 1,380 episodes; controls reuse the calibrated gate's risk head, threshold, and scales and differ in the trigger source. Budgets are comparable, not identical (intervention-rate column).

Run	PPO	Average	Random	Lagged	Calibrated gate	Gate rate
42	62.54/72.52/9.06	62.90/71.04/8.55	62.03/71.35/8.26	61.30/70.83/9.20	63.33/71.90/8.04	0.98%
43	63.04/72.12/9.49	64.71/72.55/8.84	64.93/73.23/9.13	63.62/72.38/8.70	66.30/73.22/7.83	0.92%
44	65.14/72.66/8.99	63.55/71.96/9.64	63.19/72.33/9.86	63.33/71.62/10.00	63.91/72.63/8.91	1.15%
45	63.33/72.83/7.90	63.77/72.77/8.33	66.09/73.35/7.32	65.29/73.35/7.39	65.29/73.04/7.75	0.76%
46	64.28/72.24/8.26	62.39/71.65/8.70	61.88/71.08/8.48	61.01/70.73/7.83	63.19/72.05/8.19	1.08%

Table 7: **AT controls per run** (SR/FR/CR in percent, 1,380 episodes each). The five control evaluations per row share one training run; the gate's gain is not uniform across runs. The last column is the gate's measured step-trigger rate on that run; the random control is matched to it approximately (realized $1.00/1.23/0.90/0.83/1.08\%$), and the lagged control fires on $5.4/5.1/6.7/3.4/5.6\%$ of steps.

G Perception: Replacing the Benchmark's Detection Stream

The headline gate's candidate boxes and target association come from EVT-Bench's panoptic humanoid sensors (Appendix K). To measure what the gate does with realistic perception we replaced that stream end-to-end and re-ran the DT protocol.

Perception. Candidates are the *person* detections of an RT-DETR (ResNet-50, COCO) detector run on the egocentric RGB frame (score ≥ 0.30, at most 20 boxes ordered by area); the target is the candidate selected by the SigLIP grounding probe from the instruction text, exactly as in the headline system, and the sixteen detector features are recomputed from that selection: the selected box, its probe confidence in place of the benchmark-box overlap, the probe's top-2 margin, and the statistics of the probe's candidate confidences. The visible-human count and the image-plane flags follow the same boxes. The detector finds a single person more often than the panoptic enumeration, so the probe, which needs two candidates, is unavailable on 44.6% of steps against 32.2% before; we kept that requirement unchanged.

Head and threshold. Features were logged from the five planner-head PPO runs with the gate disabled (143,277 rows, stride 5, at most 80 rows per episode, the same $80/20$ episode-hash split as Appendix J) and the same 84-input head was retrained; its validation ranking is close to the original head's on the same kind of rows (collision ROC-AUC 0.690 against 0.713, AP 0.112 against 0.146). The threshold was set before any gated evaluation by matching the original head's validation trigger rate (2.0% of validation rows at τ=0.868), giving τ=0.798; two neighbours (0.785 at 2.5%, 0.810 at 1.5%) were evaluated on run 42 and the a priori rule of Appendix L (minimum CR on the held-out-hash episodes subject to an SR drop of at most 0.5 points and a CR drop of at least 1.5) selected τ=0.798: on those 281 episodes the three thresholds give CR 10.32 / 13.52 / 11.74 against 12.10 for PPO, with SR -0.36 / -0.36 / 0.00 (full-set triplets $43.12/54.98/11.67$, $42.97/54.07/13.99$, $43.84/55.70/12.46$); the scales are the DT scales.

Same-code pairing and replication of the headline effect. Outcomes drift across code versions (Appendix D), so every comparison here is paired against the same policy evaluated under the same code with the gate disabled (the feature-logging runs), never against Table 1. Under that pairing the *original* gate, with the benchmark's panoptic perception and the deployed head, reproduces the headline effect on all five runs: ΔSR $+2.19\,[+1.10, +3.26]$, ΔFR $+1.88\,[+1.03, +2.74]$, ΔCR $-3.28\,[-4.13, -2.42]$ ($p < 10^{-4}$; per run -3.91, -3.26, -3.48, -2.83, -2.90), firing on 1.5% of steps and 30% of episodes. One caveat of provenance: a May change to a feature helper altered how the probe's normalized box is expanded into columns, which shifts the deployed head's input distribution; the replication above restores the April computation, and without that restoration the same head fires on only 0.18% of steps. That mis-configured variant serves as a null control below.

Run	PPO, same code	Learned-detector gate	ΔSR / ΔFR / ΔCR	Trigger (steps / ep.)
42	41.88/54.55/14.06	43.12/54.98/11.67	$+1.23$ / $+0.43$ / -2.39	2.06% / 44.4%
43	44.20/54.96/13.99	44.49/55.63/13.77	$+0.29$ / $+0.67$ / -0.22	2.48% / 44.1%
44	44.06/56.55/13.70	45.00/57.56/13.12	$+0.94$ / $+1.01$ / -0.58	2.06% / 44.2%
45	43.99/55.25/12.68	45.07/56.40/11.38	$+1.09$ / $+1.16$ / -1.30	1.88% / 44.5%
46	43.26/54.87/13.55	43.04/56.59/12.61	-0.22 / $+1.72$ / -0.94	2.21% / 45.9%
pooled	43.48/55.24/13.59	44.14/56.23/12.51	$+0.67$ / $+1.00$ / -1.09	2.1% / 44.6%

Table 8: **Gate with a learned detector, per run** (SR/FR/CR in percent, full 1,380-episode DT set, corrected evaluator, current code). The reference is planner-head PPO evaluated under the same code with the gate disabled.

First measurement, and why it does not survive. Table 8 gives the five runs as first measured, each treatment evaluated after its reference. Pooled and paired at the episode level the gate changes SR by $+0.67\,[-0.35, +1.78]$, FR by $+1.00\,[+0.23, +1.77]$ and CR by $-1.09\,[-1.80, -0.36]$ (sign-flip p=0.004), firing on 2.1% of steps and 45% of episodes against 1.5% and 30% for the original gate. Read alone this is a reduced but surviving effect. Two null measurements say otherwise. Repeated evaluations of one *unchanged* policy differ by -0.58, -1.23 and -0.65 CR points (Appendix D), and the mis-configured gate above — 0.18% of steps, 9% of episodes — measures ΔCR -1.07 over four runs, indistinguishable from the learned-detector gate's -1.09, with its apparent gain sitting on the episodes it never touched. Because every treatment was evaluated after its reference, an evaluator drift of this size is not separable from a treatment effect.

Time-matched re-test. We therefore re-ran the comparison with both arms *concurrent*: the 30 splits of the reference and the 30 splits of the gated arm are interleaved over the same fourteen worker slots, so both experience the same machine state and the same wall-clock window. We added null pairs in which the two arms are the identical gate-off configuration, which calibrates the protocol on the same axis. Table 9 gives all seven pairs. Pooled over the five runs the gated arm moves CR by $-0.22\,[-0.96, +0.52]$ (p=0.59), SR by $-0.54\,[-1.62, +0.51]$ and FR by $-0.27\,[-1.07, +0.52]$; the two null pairs, whose arms differ in nothing, move CR by $-0.29\,[-1.52, +0.98]$ and SR by $+0.11\,[-1.63, +1.81]$. Treatment and null are statistically indistinguishable, and the first measurement's -1.09 is accounted for by the ordering of the two arms rather than by the gate. The floor is visible directly in the per-pair numbers: identical configurations move CR by -1.38 and $+0.80$ on the two null pairs, and three evaluations of one unchanged run-42 configuration give SR 45.29, 42.61, 42.54 and CR 12.68, 14.06, 12.68.

The same protocol on the original gate. Applied to the gate of §3.3 — the benchmark's own detection, the deployed head, τ=0.868 — the concurrent protocol reproduces the headline effect on all five runs (Table 10): ΔSR $+2.71\,[+1.51, +3.88]$, ΔFR $+2.36\,[+1.40, +3.29]$, ΔCR $-3.77\,[-4.67, -2.90]$, sign-flip $p < 10^{-5}$, against $-0.29\,[-1.52, +0.98]$ for the null pairs. Five of five runs improve on all three metrics, and the effect is 2.7 times the null spread, so it is reported as a result whereas the learned-detector variant is not.

What this establishes is a boundary on the benchmark rather than a property of the method: with EVT-Bench's own detection replaced, the question moves to a scale the benchmark does not measure, since one evaluation per arm resolves neither one CR point nor two SR points. A perception-level claim therefore needs either an evaluator with repeated measurements per arm or an effect several times larger than the one at issue. The gate's own effect is that large — three to four times the floor, concentrated on the episodes it fires in (-6.2 CR points there against -2.0 elsewhere), and

Pair (concurrent arms)	Run	Arm A	Arm B	ΔSR / ΔFR / ΔCR
gate-off vs learned gate	42	45.29/56.40/12.68	42.54/54.82/13.41	−2.75 / −1.58 / +0.72
gate-off vs learned gate	43	43.91/54.84/15.43	42.32/54.57/14.93	−1.59 / −0.27 / −0.51
gate-off vs learned gate	44	44.49/55.22/14.28	43.12/55.16/13.62	−1.38 / −0.06 / −0.65
gate-off vs learned gate	45	42.46/55.32/13.12	44.28/55.47/12.97	+1.81 / +0.14 / −0.14
gate-off vs learned gate	46	44.06/55.59/13.62	45.29/55.99/13.12	+1.23 / +0.40 / −0.51
pooled, five runs				**−0.54 / −0.27 / −0.22**
gate-off vs gate-off (null)	42	42.61/55.04/14.06	42.54/54.63/12.68	−0.07 / −0.41 / −1.38
gate-off vs gate-off (null)	43	43.26/54.09/14.49	43.55/55.40/15.29	+0.29 / +1.31 / +0.80
pooled, two null pairs				**+0.11 / +0.45 / −0.29**

Table 9: **Time-matched paired re-test** (SR/FR/CR, percent, full 1,380-episode DT set, seven paired evaluations of 2,760 episodes each). Both arms of a row run concurrently over the same worker slots. The null rows' two arms are the same configuration, so their spread is the protocol's resolution: the gated arm's pooled deltas sit inside it.

Run	Gate off	Risk gate (τ=0.868)	ΔSR / ΔFR / ΔCR
42	43.41/55.33/13.33	44.42/56.02/9.28	+1.01 / +0.69 / −4.06
43	44.06/54.67/14.49	45.72/56.61/11.30	+1.67 / +1.94 / −3.19
44	43.33/55.02/14.93	47.75/58.24/10.29	+4.42 / +3.22 / −4.64
45	43.84/55.34/12.83	46.59/57.77/8.70	+2.75 / +2.43 / −4.13
46	43.04/54.61/13.04	46.74/58.14/10.22	+3.70 / +3.53 / −2.83
pooled	43.54/55.00/13.72	46.25/57.36/9.96	**+2.71 / +2.36 / −3.77**

Table 10: **The DT risk gate under the concurrent protocol** (SR/FR/CR, percent, full 1,380-episode set, five paired evaluations of 2,760 episodes each). Both arms of a row are evaluated at the same time on the same worker slots, the design that removes the drift visible in Table 9's null pairs.

replicated across code versions, so it is reported as a result and the learned-detector variant as the next experiment.

H Strong Slowdown and a Proximity Rule

The matched controls of §4.4 test whether the gate's small attenuation budget is sufficient without its timing. Two further controls, run under the current code and paired against the same-code planner-head PPO reference of Appendix G, test the converse.

Strong uniform slowdown. Applying the gate's own scales $(s_f, s_l, s_\omega) = (0.25, 0.60, 1.00)$ at every step, on all five runs, lowers CR from 13.59 to 6.01 but drives SR from 43.48 to 6.54 and FR from 55.24 to 24.25 (paired ΔSR $-36.9\,[-39.1, -34.7]$, ΔFR $-31.0\,[-33.0, -29.0]$, ΔCR $-7.6\,[-8.9, -6.4]$; per run 6.52/24.09/5.87, 6.67/24.87/5.80, 6.30/23.72/5.80, 6.59/24.06/6.45, 6.59/24.53/6.16). Uniform slowing therefore lowers the collision rate without tracking, whereas the timing of the gate's intervention lowers it while the following rates are preserved.

Rule-based proximity trigger. The converse control replaces the learned head by a hand-written rule with the same scales: fire when the learned probe places the target in its nearest distance bin with distance confidence at least 0.588, a threshold chosen on the run-42 logs to match the gate's 1.6% step rate (realized 1.5–1.7% of steps and 35–36% of episodes). Over the five runs the rule changes SR by $+0.17\,[-0.86, +1.22]$, FR by $+0.27\,[-0.48, +1.04]$, and CR by $-0.68\,[-1.46, +0.12]$ (sign-flip p=0.10), with CR falling on four runs and rising on one (per run −1.38, +0.36, −0.29, −1.09, −1.01). Proximity alone, at the gate's budget, does not produce a resolved safety effect. It is also at the noise floor of Appendix D, so this control bounds the effect of such a rule rather than showing it to be zero; the comparison it supports is with the original gate's −3.28 under the same pairing, not with the learned-detector configuration, which is itself unresolved.

I Generalization Evidence in Detail

We probe four axes on which the effect could plausibly fail inside the released environment. None substitutes for a second benchmark or a real robot.

Scene distribution. EVT-Bench episodes span two independent scene corpora. Recomputing the paired DT comparison separately on each shows the safety–success pair on both: HM3D ΔSR $+2.11$ / ΔCR -4.02 (3,360 pairs), MP3D $+1.95$ / -3.90 (3,540 pairs). The effect is not carried by one scene distribution.

Collision definition. Because CR depends on the simulator's contact geometry, we re-evaluated PPO against the gate under three alternative navmesh footprints (0.10/0.15/0.20 m lateral offsets, against the default 0.25 m; 16 full evaluations over four, three, and one training runs respectively). The CR reduction keeps its direction (ΔCR -1.03/-2.15/-0.72 against -3.96; ΔSR $+0.60$/$+0.48$/-0.14 against $+2.03$); its magnitude does not vary monotonically with the footprint and the 0.20 m point is a single run, so the direction holds under every tested collision definition.

Operating-point transfer. Applying the DT-calibrated gate to AT *unchanged* — same risk head, same τ, same scales — still reduces CR on all five runs (8.01 against PPO's 8.74), at a one-point SR/FR cost that task-specific calibration recovers. What transfers is the interface; calibration is a knob that trades following performance back.

Backbone. The head-stack trains and evaluates end-to-end on three backbone scales (Qwen2.5-0.5B, Qwen3-0.6B, Qwen2.5-1.5B) with no code changes (Table 11). To separate backbone from data we trained all three on one shared self-distilled corpus (446k samples, one epoch): they land within 1.3 DT-SR points of each other (7.54 / 7.03 / 6.30) and far below the main model trained on the original corpus (41.38), which is consistent with the corpus, not the backbone, explaining the gap, although it does not exclude backbone effects at full corpus scale. Planner-head PPO runs end-to-end on the new backbone but does not rescue a weak base (3.70 DT SR). To probe corpus scale directly we re-collected an expert-rollout corpus of 3,121 episodes (183k samples, about 13% of the original corpus) and retrained Qwen2.5-0.5B under the original recipe (2 epochs, batch 8, learning rate 2×10^{-5}): STT reaches 55.36/64.35/4.28, a collision rate already at the main model's 4.13, and DT 15.58/38.50/13.41, twice the distilled-corpus result. These are intermediate numbers on a partial corpus: they support the reading that corpus scale is the remaining factor, and they are evidence of portability, not a cross-backbone performance claim, which requires the original full-scale corpus.

Backbone	SFT corpus	DT SR / FR / CR	STT SR / FR / CR
Qwen3-0.6B (main model)	original	41.38/54.46/11.01	78.26/75.70/4.13
Qwen2.5-0.5B	distilled, 446k	7.54/26.14/8.19	–
Qwen3-0.6B (control)	distilled, 446k	7.03/25.58/7.25	–
Qwen2.5-1.5B	distilled, 446k	6.30/23.77/8.91	–
Qwen2.5-0.5B + PPO	distilled, 446k	3.70/18.86/5.14	6.38/21.75/4.86
Qwen2.5-0.5B	expert rollouts, 183k	15.58/38.50/13.41	55.36/64.35/4.28

Table 11: **Backbone portability.** All rows are single training runs (one SFT model each, plus one PPO run) evaluated on the full 1,380-episode settings under the corrected evaluator; they are not five-run means. The three distilled-corpus rows share one corpus and recipe.

J Risk-Head Reliability and Recalibration

Fitting an isotonic regression on one run and applying it to the four held-out runs gives ECE 0.0016/0.0009/0.0013/0.0005 and Brier $\approx$0.015: a roughly two-hundred-fold reduction in ECE and a thirteen-fold reduction in Brier. The deployed system keeps the raw score and the raw threshold $\tau = 0.868$ (which the map sends to a calibrated risk of 0.112), so no trigger decision reported in this paper depends on the map; we give it for interpretability only.

Training data, partitions, and the scope of the deployed evaluation. The deployed head is trained on 142,976 step rows logged at stride 5 (at most 80 rows per episode) from five full DT eval-

Label window K	Positive rate (count)	ROC-AUC	AP	ECE (raw)	Brier (raw)
10	0.89% (6,583)	0.842	0.080	0.371	0.201
20	1.60% (11,897)	0.813	0.103	0.364	0.200
40	2.50% (18,543)	0.781	0.120	0.355	0.198

Table 12: **Ranking quality and probability calibration of the deployed risk head**, over 741,278 scored steps from the five deployed runs. A positive label is a step inside the K-step window preceding a collision termination. Ranking is strong at every window; the raw sigmoid is badly calibrated as a probability at every window. We therefore report the output as a risk score.

uation runs of earlier policies — the compact base, planner-head PPO (run 42), two distance-only filters, and an image-plane filter — all on the same 1,380 DT episodes used for every evaluation in this paper. Training labels are episode-terminal: every logged step of an episode that ends in collision (lost, failure) is a positive for the collision (lost, failure) output; the K-step windows of Table 12 are used only to measure the deployed score afterwards. Rows are split by a stable hash of the episode identifier into 80% training and 20% validation (30,207 validation rows, collision base rate 4.9%); the threshold and scales are chosen on the validation rows' predictions by an intervention-rate sweep (Appendix L) and frozen before any full five-run evaluation. Table 13 summarizes the sets.

Set	Source	Episodes	Rows / steps	Used for
head training	5 DT runs, 80% of episode ids by hash	≈1,104	112,769 rows	fitting f_ϕ
head validation	same runs, remaining 20%	≈276	30,207 rows	τ, (s_f, s_l, s_ω)
deployed evaluation	5 PPO runs × all episodes	1,380	741,278 steps	Tables 1–2

Table 13: **Risk-head data partitions.** The deployed-evaluation episode set contains both partitions above, so the five-run results measure the gate on episodes whose scenes and identities the head saw during training under other policies' rollouts (and, for run 42, under a diagnostic pass of the same PPO policy).

Consequently the head's generalization to unseen episodes is supported only by the validation partition, and the five-run deployed evaluation is a within-benchmark measurement: it establishes what the gate does on this benchmark, not how the head would transfer to scenes or identities it has never scored. The scene-corpus split of Appendix I does not remove this overlap, because both corpora contribute training episodes. We state this as a limit of the evidence rather than as a detected leak; the paired comparison against PPO on the same episodes remains valid as a measurement of behavior on the benchmark, and on the 281 episodes of the validation partition, whose rows never entered fitting, the collision reduction (ΔCR $-4.13\,[-6.26, -2.21]$) matches the full-set effect while the success and following gains ($+0.07$, $+0.99$) are not resolved at that sample size.

K Deployed Risk-Feature Definitions

The deployed head consumes 84 scalars, all available before the step's action is executed. They fall into eight groups. *Nine action components* (the previous command, the planner's proposed command, and the previously executed command) and *three base-velocity components*, from the policy. *Sixteen detector quantities*: candidate boxes are the image-plane boxes of humanoid instances in EVT-Bench's panoptic segmentation, the selected box is the candidate with the highest overlap with the benchmark's target box, its score is that overlap, the top-2 identity margin is the gap to the runner-up, and four statistics summarize the candidate scores (count, maximum, mean, top-2 margin); detection and target association are therefore simulator-perfect in this study. *Six image-plane quantities* (box size, centering error, and two image-plane gate flags) computed on the selected box. *Fifteen planner outputs* (the twelve-dimensional trajectory-head summary and the first predicted waypoint). *Thirty-three probe outputs* from a learned SigLIP grounding probe run on RGB crops of the candidates (angle and distance bins with their confidences, selection confidence and top-2 margin, memory age, recovery and status flags, and selected-box geometry in pixel and normalized coordinates). A *visible-human count* from the same panoptic enumeration, and the *step index*. The

logged feature set also contains quantities that become available only after the step is executed; these are not inputs of the deployed head.

No feature reads simulator pose, geodesic distance, or the benchmark following bands. These privileged quantities are excluded from the head's inputs by construction. The detector group is a different case: it is deployable in form (boxes and scores), but its quality is that of the simulator, because the benchmark supplies detection and identity from its own sensors rather than from a learned detector, and Figure 6 shows the identity group is among the head's strongest inputs. Replacing it with a learned detector and re-identifier is the step from this study to a deployable system.

L Risk Gate Calibration Protocol

Diagnostic set. Calibration proceeds in two steps on data logged before any five-run evaluation. First, an offline intervention-rate sweep on the 30,207 validation rows of the risk-head dataset (Appendix J) ranks rows by score and reads collision precision, lift, and episode capture at the top 1–20% of rows, which shortlists thresholds whose intervention rate is a few percent of steps. Second, the shortlisted $(\tau, s_f, s_l, s_\omega)$ settings are checked closed-loop on a diagnostic set built from a single full DT run under the corrected protocol, with 20% of its episodes as a tuning subset and the remaining 80% as a verification sample for gate decisions before any full five-run evaluation. The diagnostic episodes belong to the same 1,380-episode benchmark as the five-run evaluations; what is held out is the evaluated policies' rollouts, not the scenes (the overlap is stated in Appendix J).

Threshold sweep. The collision-risk threshold τ is swept over a coarse grid from 0.50 to 0.90 and then refined near the knee of the trigger-rate curve, where 0.868 was selected; the scale triple (s_f, s_l, s_ω) is swept on a small Cartesian grid that prefers slow-down on the forward axis and preserves angular freedom. The selected DT setting $(\tau, s_f, s_l, s_\omega) = (0.868, 0.25, 0.60, 1.00)$ is the one that minimizes CR on the tuning subset subject to a non-decrease constraint on SR.

AT calibration. The DT setting is too aggressive for AT: it lowers FR more than it lowers CR. For AT we therefore use a separate risk head and freeze two AT-specific operating points before the full five-run evaluation. The AT head takes an 80-dimensional input built from the robot-frame probe features (Appendix T) and is trained on rollouts over AT episode partitions 8–29, with a collision label at horizon $H{=}20$ and lost-target and failure labels at $H{=}40$. Its decision variable is a benefit margin, the collision score minus 0.35 times the lost-target score, and the gate fires only when, in addition, the selection confidence is at least 0.70, the top-2 identity margin is at least 0.12, and the commanded forward speed is at least 0.01. The training partitions and the evaluation set are drawn from the same released AT episodes, so the AT head is not episode-disjoint from its evaluation. The calibrated gate uses $(\tau, s_f, s_l, s_\omega) = (0.586, 0.65, 0.85, 1.00)$, selected on the AT tuning subset for all-metric improvement under a non-worsening CR criterion; the margin gate uses $(0.904, 0.70, 0.90, 1.00)$, selected for higher FR under a matching CR constraint. Both attenuate more gently than the DT gate, and the calibrated gate fires at a much lower score, so it triggers on 0.76–1.15% of AT steps. Table 14 reports both full five-run operating points.

Concurrent re-evaluation of the AT gate. A single evaluation per arm cannot resolve differences near one CR point (Appendix D). We therefore re-evaluated the calibrated gate against its PPO reference with both arms running concurrently on the same worker slots, five pairs with the full 1,380 AT episodes each. Pooled over episodes, ΔSR is $-0.20\,[-1.13, +0.72]$, ΔFR is $-0.47\,[-1.03, +0.09]$, and ΔCR is $-0.06\,[-0.65, +0.51]$ (sign-flip $p{=}0.90$). Per run, CR moves from 8.41 to 8.33, 8.91 to 8.55, 8.33 to 8.84, 7.10 to 7.17, and 8.33 to 7.90 (reference to gate, runs 42–46). A null pair with two identical arms gives ΔCR $+0.58\,[-0.65, +1.88]$. The -0.59 reported for AT in Table 2 from the original evaluation therefore lies within the benchmark's measurement floor.

Promotion criteria. A configuration is carried to a full five-run evaluation only if (i) on the tuning subset, CR drops by at least 1.5 absolute points relative to the PPO baseline, (ii) SR does not drop by more than 0.5 absolute points, and (iii) the average per-step gate-fire rate stays below 15% so that the policy remains close to the base PPO behavior. The reported DT gate met all three criteria. The AT rows of Table 14 are calibrated extensions rather than the primary causal claim. Both settings carry matched slowdown, random-trigger, and lagged-risk controls (Table 4, Appendix F), but only

AT operating point	SR	FR	CR	Role
Planner-head PPO	$63.67_{\pm 1.04}$	$72.48_{\pm 0.30}$	$8.74_{\pm 0.65}$	Ref.
TRACE calibrated gate	**64.41**	72.57	8.14	Main
TRACE margin gate	63.67	**72.61**	**8.07**	FR-oriented

Table 14: **AT calibrated operating points** (drill-down of the AT column of Table 1). Both gates were frozen before the five-run evaluation; the calibrated gate is the one reported in Table 1 because its selection rule (all-metric improvement) is the paper's default, and the margin gate reaches a similar CR (8.07) with the SR gain traded for FR. Per run the calibrated gate is not uniformly better than PPO (Table 7), and its paired deltas did not reproduce under concurrent evaluation (see below).

on DT do all three metrics move significantly, and only the DT effect reproduces under concurrent evaluation.

Operating-point selection at deployment. For a deployment that must choose between the two AT gates, the choice reduces to whether the user requires non-decreasing SR or non-increasing CR as the strict constraint. The default operating point reported in Table 1 is the calibrated gate because, in the original evaluation, it improved SR, FR, and CR together. Under the concurrent re-evaluation above, neither gate changes SR, FR, or CR by a resolvable amount, so these measurements do not select between them.

Post-hoc τ trigger-rate curve. The recorded per-step risk scores on the DT gated run let us characterize the calibration shape *without* re-running eval. On all 149,149 scored steps of the full 1,380-episode DT run 42 run, the fraction of steps that would have triggered as a function of τ is:

τ	0.80	0.84	0.85	0.86	**0.868**	0.88	0.89	0.90	0.92
step-trigger rate (%)	4.37	2.53	2.18	1.81	**1.59**	1.27	1.02	0.81	0.48

Table 15: **Post-hoc τ trigger-rate curve on DT run 42.** The risk-score distribution is gradual rather than bimodal at the calibrated threshold: τ perturbations of ± 0.02 around 0.868 shift trigger rate by less than 0.6 absolute points (from 2.18% at τ=0.85 to 1.27% at τ=0.88), and even τ=0.80, the smallest value in the table, keeps the trigger rate under 5%. The full τ-vs-SR/FR/CR sweep at re-evaluated trajectories is reported in Appendix N.

M Auxiliary Safety Diagnostics: Time-to-Human and Min-Distance

Condition	Robot–human proximity				Wall clearance d_{obs} (five runs)				
	d_t med. (m)	d_t p10 (m)	TTH med. (s)	TTH p10 (s)	p10 (m)	p25 (m)	p50 (m)	p75 (m)	p90 (m)
Planner-head PPO	1.520	1.014	8.558	3.268	0.027	0.093	0.221	0.450	0.715
TRACE (non-triggered)	1.537	1.038	8.639	3.320	0.024	0.086	0.212	0.437	0.701
TRACE (triggered)	**0.861**	**0.622**	11.716	**1.278**	**0.014**	**0.043**	**0.142**	**0.254**	**0.355**

Table 16: **Safety audit on DT: robot–human proximity and wall clearance for PPO and the TRACE gate, split into triggered and non-triggered steps.** d_t: robot–human distance (m); TTH: time-to-human, $d_t/|v_x|$ (s); d_{obs}: robot–obstacle clearance (m); "med."/p10–p90: median and percentiles (smaller = closer or shorter). Human columns are a 600-episode sample of the seed-42 run; wall clearance is over 230 episodes from each of the five runs.

To complement CR (a binary simulator contact event), we report two post-hoc safety diagnostics (Table 17) on a 600-episode sample of the TRACE DT gated run and of the matched PPO baseline (seed 42; 59,343 and 56,292 steps). Every step records the robot-to-human distance d_t (m) and

the executed forward velocity v_x^{exec} (m/s); we define **time-to-human** $\text{TTH}_t = d_t/|v_x^{\text{exec}}|$ (seconds; undefined when stationary). TTH is to a human-following robot what time-to-collision (TTC) is to autonomous driving: a per-step time margin under current velocity.

Metric (DT seed 42, 600-episode sample)	PPO	Gate (non-trig.)	Gate (trig.)
n steps	56,292	58,472	871 (1.47%)
d_t median (m)	1.520	1.537	0.861
d_t p10 (m)	1.014	1.038	0.622
TTH median (s)	8.558	8.639	11.716
TTH p10 (s)	3.268	3.320	1.278

Table 17: **TTH and d_t on a 600-episode sample of the matched DT seed-42 runs.** Triggered steps concentrate where the human is closer (median d_t on triggered steps is $0.86\,\text{m}$ vs. $1.54\,\text{m}$ on non-triggered steps) and where the time margin tail is dangerous (TTH p10 of $1.28\,\text{s}$ on triggered vs. $3.32\,\text{s}$ on non-triggered). Outside the triggered window the gated run's d_t medians differ from PPO's by $0.017\,\text{m}$, consistent with an intervention that is sparse and targeted rather than a blanket slowdown.

Min-distance per episode, grouped by outcome. For each episode we record the minimum d_t along its trajectory. Conditioned on *collision-ending* episodes, the median min-d_t is $0.455\,\text{m}$ for PPO and $0.464\,\text{m}$ for the gate, i.e. when contact occurs, both policies were equally close to the human, so the gate's CR reduction is not coming from a different distribution of close-approach behavior on the collided tail. Conditioned on *non-collision* episodes, the median min-d_t is $1.085\,\text{m}$ (PPO) vs. $1.055\,\text{m}$ (gate), i.e. the gate does not push the follower further away from the human in nominal following. The CR delta therefore reflects a true reduction in the number of contact-event episodes, not a metric artifact.

N τ Sensitivity Sweep (SR / FR / CR)

The post-hoc trigger-rate curve (Table 15) fixes the trajectory under τ=0.868; the true SR/FR/CR sensitivity to τ requires re-evaluation per setting. We sweep $\tau \in \{0.83, 0.85, 0.868, 0.88, 0.90\}$ over five DT runs (Table 18), keeping all other gate parameters $(s_f, s_l, s_\omega) = (0.25, 0.60, 1.00)$ fixed. Trigger rates are reported from the recorded seed-42 score distribution, while SR/FR/CR are five-run means.

τ	step trigger rate (%)	SR	FR	CR
0.83	2.95	44.44	55.84	12.52
0.85	2.18	44.12	56.24	12.97
0.868 (calibrated)	**1.59**	**46.02**	**57.58**	**9.83**
0.88	1.27	44.41	55.57	12.09
0.90	0.81	44.16	55.87	12.45

Table 18: **τ sensitivity on full DT five-run evaluation.** The calibrated row is the main DT operating point. All four nearby thresholds keep CR below the PPO reference $(43.99/55.44/13.78)$, so the safety direction does not depend on the exact threshold; the joint SR/FR/CR gain, however, is specific to the calibrated threshold, which was selected on the tuning data of Appendix L, and the response is not monotone in τ. We report this as a limit of the operating point.

O Wall Clearance in a Separate Re-evaluation

To test whether the gate's CR reduction is a contact-event accounting artifact (i.e. that the gate turns "contact" into "close-but-no-contact"), we recorded, at every step of a separate re-evaluation (Table 19), the navmesh distance from the robot to the nearest obstacle, d_{obs} (values clipped at 2 m). We re-evaluated DT on a matched subset of every run (episode partitions 0–4, 230 episodes per run, 1,150 per policy) for both the PPO baseline and the calibrated TRACE gate.

d_{obs} **percentile (m)**	**PPO**	**Gate (all)**	**Gate (trig.)**	**Gate (non-trig.)**
n steps	136,012	141,798	443	141,355
p10	0.027	0.024	0.014	0.024
p25	0.093	0.086	0.043	0.086
p50	0.221	0.212	0.142	0.212
p75	0.450	0.436	0.254	0.437
p90	0.715	0.700	0.355	0.701

Table 19: **Wall-clearance distribution under PPO vs. TRACE gate (DT, 230 episodes from each of the five runs, $1{,}150$ per policy).** PPO and the gated run have nearly identical per-step d_{obs} distributions at every reported percentile (deltas ≤ 0.015 m). The gate's triggered steps are concentrated at lower d_{obs} (median 0.142 m vs. 0.212 m non-triggered), confirming that the gate fires in close obstacle states. *Caveat*: this re-evaluation does not reproduce the headline gate's activation profile — the gate fired on 443 of $141{,}798$ steps (0.31%, against 1.0–2.0% in the headline runs) and collision-ending episodes moved only from 201 to 184 (17.5% to 16.0% on this subset, against 13.78% to 9.83% on the full runs) — so we use it only for the clearance distribution, which is the quantity the headline runs do not log, and not as evidence of the CR effect.

P Reward Retuning and Geometric Filtering

Cheaper interventions were tried first, and this appendix reports the outcome of each. Table 20 summarizes the methodological alternatives that the main paper points to from §4.4, and Figure 4 plots the same controls on the DT Pareto plane next to the validated TRACE gate. All experiments below start from the same base policy and reward. Rows marked *development phase* were evaluated before the evaluator correction of Appendix D, or are training-rollout statistics, and are therefore reported qualitatively rather than beside Table 1; the distance-only filter, the geometry sweeps, the matched controls, and PPO-Lagrangian were evaluated under the corrected evaluator.

REINFORCE. A critic-free policy gradient using the same reward as PPO, run in the development phase. Its training rollouts collapse to a near-deterministic high-collision policy within the 200-episode budget (second round: SR 1%, CR 52% over 100 on-policy episodes); the collapsed policy was not evaluated with the corrected evaluator. The collapse confirms that the learned value head is necessary under the long horizon and noisy returns of EVT, not just a performance optimization.

PPO budget $\times 2$. A doubled rollout budget under the same hyperparameters (two further rounds from the run-42 model), re-evaluated on the full DT set under the corrected evaluator and paired with planner-head PPO under the same code (Appendix G explains the same-code pairing): SR rises from 41.88 to 46.09 while CR rises from 14.06 to 16.30, indicating that the DT problem is not a sample-efficiency problem within this budget regime.

Collision penalty $\times 2$. Doubling w_c to 10, in a configuration that also used a lower learning rate and a 600-episode budget, re-evaluated on the full DT set under the corrected evaluator (run 42): $41.88/55.04/21.01$ against $41.88/54.55/14.06$ for PPO under the same code. The policy becomes more aggressive in following without becoming safer, and DT CR worsens by 7 points. This behavior is consistent with the marginal shift of Figure 7a–b: stronger following pressure moves more endings from lost-after-following toward collision-while-following.

Task-specific PPO. Two development-phase variants train on DT-only or transfer from STT to DT (pre-correction evaluator). Neither beats the STT-trained planner-head PPO model on DT, ruling out a simple training-task mismatch.

Imitation / preference objectives. RWR variants, self-SFT/FTSFT, RAFT, DPO, and GRPO, run in the development phase under the pre-correction evaluator, all show no stable gain or regress under matched conditions. We attribute this to the fact that imitation-style objectives distill rollout errors when the supervised reference is the deployed policy itself.

Alternative	Result and takeaway
REINFORCE (critic-free policy gradient)	*Development phase (training-rollout statistics).* Collapses within the 200-episode budget: the second round's 100 on-policy episodes end at SR 1% and CR 52%; not evaluated with the corrected evaluator. *Takeaway:* a learned critic is necessary in this long-horizon setting.
PPO budget $\times 2$	*Corrected evaluator, full DT, run 42 (re-evaluated).* $46.09/56.99/16.30$ against $41.88/54.55/14.06$ for planner-head PPO under the same code ($+4.2$ SR, $+2.2$ CR): more persistence, and more contact. *Takeaway:* additional RL raises success and collisions together.
Collision penalty $\times 2$	*Corrected evaluator, full DT, run 42 (re-evaluated).* $w_c{=}10$ with a lower learning rate and a 600-episode budget: $41.88/55.04/21.01$ against $41.88/54.55/14.06$ for PPO under the same code, i.e. CR $+7$ points at equal SR. *Takeaway:* scalar collision retuning produces aggressive following, not safe control.
Imitation / preference (RWR, RAFT, DPO, GRPO, FTSFT)	*Development phase (pre-correction evaluator).* No stable gain, or regression, under matched conditions. *Takeaway:* imitation-style objectives distill rollout errors when the reference is the deployed policy.
Distance-only learned filter	*Corrected evaluator, full DT, run 42.* $43.84/54.70/13.55$ against PPO $44.57/56.41/12.75$. *Takeaway:* distance is necessary but not sufficient as a risk feature.
Heuristic / geometry filtering	*Corrected evaluator, 368-episode subset.* Navmesh-radius sweeps fail the CR-drop / SR-retention criterion. *Takeaway:* safety needs richer target-state conditioning.
PPO-Lagrangian (five runs, two rates)	*Corrected evaluator, full DT, five runs.* $12.70{\pm}4.25/32.20{\pm}5.15/28.67{\pm}15.33$ and $12.28{\pm}5.30/31.37{\pm}8.10/26.61{\pm}16.10$ (Table 22). *Takeaway:* dual-ascent retuning of the scalar objective destabilizes on every run.

Table 20: **Cheaper post-training and filtering alternatives do not reproduce the safety–success pair.** All start from the same base policy and reward. Rows marked *development phase* predate the evaluator correction (or are training-rollout statistics) and are qualitative; the remaining rows use the corrected evaluator. None matches TRACE's DT safety–success pair. Triplets read SR/FR/CR in percent. The paragraphs below expand each row.

Recognition refresh SFT. A small-data SFT on the recognition pathway, evaluated in the development phase on a 368-episode subset with the pre-correction evaluator. DT refresh SFT and recognition-only refresh fail the promotion criteria. PPO-initialized clean refresh is less destructive but still below planner-head PPO.

Geometry / distance-only filtering. Navmesh radius sweeps (368-episode subset) and a learned distance-only filter (full DT, run 42), both under the corrected evaluator. None pass the CR-drop / SR-retention criterion. The learned distance-only filter underperforms the PPO seed-42 reference, showing that distance is necessary but not sufficient as a risk feature.

Matched action-scaling controls. The three controls—average slowdown, random-trigger slowdown, and lagged-risk slowdown—are discussed in §4.4 and Table 4, with per-run numbers in Table 21. All fifteen full-DT evaluations (3 controls $\times$ 5 runs, 30 episode partitions and 1,380 episodes per run/control) reuse the same risk head, threshold, scales, target-state features, and probe as the headline TRACE run, so the trigger source is the intended difference; trigger rates are matched approximately rather than exactly: the gate fires on $1.59/1.69/1.95/1.02/1.88\%$ of steps on runs 42–46 (1.6% pooled; $31.5/32.6/34.1/24.2/32.1\%$ of episodes), random on 1.1–2.0% and lagged-risk on 0.5–0.9% (Table 21). Average slowdown applies the per-run trigger-rate-implied mean attenuation at every step; random slowdown samples timesteps approximately at each run's measured gate rate; lagged-risk uses the gate's risk head and threshold but reads a 37-step lagged score. None reproduces the gate's safety–success pair. These controls test whether the gate's small attenuation budget is sufficient on its own; they do not test a strong uniform slowdown or a hand-crafted proximity trigger at a matched rate, which would probe whether the learned head is necessary.

Control	Run	SR	FR	CR	ΔSR vs PPO	ΔFR vs PPO	ΔCR vs PPO	Step rate
average	42	42.75	55.01	13.84	-1.82	-1.40	+1.09	100.00%
average	43	43.26	55.10	15.29	+0.00	+0.49	+0.07	100.00%
average	44	43.55	55.83	13.77	-0.73	+0.19	-1.45	100.00%
average	45	44.78	55.53	11.96	+0.72	+0.28	-0.79	100.00%
average	46	42.61	54.21	13.48	-1.16	-1.07	+0.51	100.00%
random	42	42.83	55.04	14.06	-1.74	-1.37	+1.31	1.36%
random	43	44.28	55.43	14.57	+1.02	+0.82	-0.65	1.81%
random	44	43.55	55.11	14.57	-0.73	-0.53	-0.65	1.99%
random	45	44.49	56.10	12.54	+0.43	+0.85	-0.21	1.13%
random	46	44.64	55.66	12.61	+0.87	+0.38	-0.36	1.66%
lagged	42	42.54	54.72	13.48	-2.03	-1.69	+0.73	0.81%
lagged	43	42.90	55.84	14.93	-0.36	+1.23	-0.29	0.66%
lagged	44	43.48	55.24	14.71	-0.80	-0.40	-0.51	0.90%
lagged	45	44.71	55.89	12.61	+0.65	+0.64	-0.14	0.45%
lagged	46	44.64	56.18	14.28	+0.87	+0.90	+1.31	0.94%

Table 21: **Full matched controls, per run.** All rows use full DT evaluation with 1,380 episodes. Positive ΔCR is worse. The controls stay close to PPO and remain far from TRACE's 46.02 / 57.58 / 9.83 five-run mean, supporting the conclusion that the gate's gain is not explained by slowdown, sparse trigger budget, or risk-score density alone.

PPO-Lagrangian (five runs, two dual-ascent rates). A principled constrained-RL alternative keeps the scalar objective and adds a Lagrangian collision constraint: the collision indicator of Equation 1 is weighted by an additional multiplier λ, updated once per round by dual ascent, $\lambda \leftarrow \max(0, \lambda + \eta\,(\mathrm{CR}_{\mathrm{round}} - 0.10))$, with $\lambda_0 = 5$ and $\eta \in \{20, 5\}$, over four rounds of 50 on-policy episodes (the same 200-episode budget as planner-head PPO's two rounds of 100) from the same base. On run 42 with η=20 the multiplier moves $5 \to 13.8 \to 11.8 \to 9.8 \to 7.8$ while the rollouts collapse to zero following after the first round, so the failure is a collapse under the constraint rather than a slow drift; we did not tune the dual-ascent schedule beyond the two rates. It destabilizes on every run. With a dual-ascent rate of 20 the five runs reach 12.70±4.25 / 32.20±5.15 / 28.67±15.33 (SR / FR / CR), and a softer rate of 5 collapses the same way (12.28±5.30 / 31.37±8.10 / 26.61±16.10), so the collapse is not specific to one dual-ascent rate at this budget (Table 22). The five planner-head PPO runs it is compared with reach 43.99 / 55.44 / 13.78.

Run	λ-rate 20: SR / FR / CR	λ-rate 5: SR / FR / CR
42	12.83 / 33.37 / 26.45	12.97 / 34.77 / 10.43
43	11.88 / 34.74 / 14.28	14.71 / 37.33 / 14.78
44	19.57 / 37.84 / 14.71	18.99 / 38.13 / 37.25
45	7.97 / 24.16 / 49.71	9.86 / 27.77 / 21.67
46	11.23 / 30.91 / 38.19	4.86 / 18.85 / 48.91
mean±std	12.70±4.25 / 32.20±5.15 / 28.67±15.33	12.28±5.30 / 31.37±8.10 / 26.61±16.10

Table 22: **PPO-Lagrangian on DT, per run.** Same base, reward, rollout budget, and corrected evaluator as planner-head PPO; only the constraint machinery is added. Both dual-ascent rates destabilize training on every run.

Q Single-Target Tracking: Alternatives Considered

Planner-head PPO on single-target tracking raises SR by 8.2 points and FR by 5.2 points over the supervised baseline while leaving CR essentially unchanged ($4.13 \to 4.09$). Any further extension must meet a strict promotion criterion—non-degrading CR with at least $+0.25$ SR and $+0.25$ FR on an independent validation subset—rather than merely improve tracking. Table 23 reports the completed alternatives; micro-gate and gated planner apply the DT gate to the STT planner at two operating points, collision-anchored continuation is one further PPO round anchored on the collision term, the two gate-repair rows re-tune the gate on top of it (on all runs, and on the two weakest runs), and the last three rows are the c=16 and c=18 continuations and their per-run mix. Plain planner continuation can raise SR and FR but does not by itself meet the CR criterion: the best collision-anchored continuation remains $+0.05$ above PPO on CR. Every gate-repair configuration we tried

Attempt	Scope	SR / FR / CR	Decision
Planner-head PPO	5 runs	$86.45/80.88/4.09$	Main STT reference.
Micro-gate on planner	5 runs	$86.39/80.73/4.06$	Essentially neutral; no tracking gain.
Gated planner	5 runs	$86.84/81.27/4.30$	SR and FR improve, but CR worsens.
Collision-anchored continuation	5 runs	$87.35/81.69/4.14$	Strongest tracking among single continuations; CR remains 0.05 above PPO.
Gate-repair sweep	selection subset	$88.84/84.35/2.61$	Best selection-subset row among four gate settings, all of which trigger on 0% of episodes and steps, so the shift comes from the planner.
Weak-run gate repair	runs 44/45	not aggregated	CR rises on both diagnostic runs while the gate still triggers on 0% of episodes and steps.
CR-constrained selection sweep	selection subset	$90.29/85.41/2.32$	Three candidates meet the selection criterion after one extra low-learning-rate PPO round; two are carried to full five-run evaluation.
CR-preserving continuation (c=16)	5 runs	$87.26/81.68/4.13$	Improves SR/FR over PPO but leaves CR 0.04 above PPO; strict-Pareto only on the independent subset.
Balanced continuation (c=18)	5 runs	$87.36/81.77/4.14$	Repairs runs 44 and 45 but inflates run 46; strict-Pareto only on the selection subset.
Per-run CR-anchored mix	5 runs + independent	$\mathbf{87.51/81.80/3.91}$	**Reported in Table 1.** Uses selection-subset CR to select c=16 or c=18 per run; strict-Pareto on all three subsets.

Table 23: **Completed STT attempts.** Values are SR/FR/CR. The two single-c continuations are each strict-Pareto on exactly one evaluation subset; the per-run CR-anchored mix selects the lower-CR variant per run using only the selection subset and remains strict-Pareto on selection, independent, and full five-run settings. Gate-repair diagnostics are retained because they show that the runtime gate is not the active lever on STT.

produced a trigger rate of exactly 0% on both episodes and steps, so the runtime-gate interface cannot supply the missing safety lever on this setting. The two single-c planner continuations (c=16 and c=18) each land just above the PPO CR in full five-run mean, but inspection of the selection and independent subsets shows they have *complementary* per-run CR profiles: c=16 is CR-fail on the selection subset and strict-Pareto on the independent subset; c=18 is the reverse. The per-run CR-anchored selection chooses, for each run, the collision-penalty level with lower selection-subset CR (selection rule: $\{42{:}c18, 43{:}c16, 44{:}c18, 45{:}c18, 46{:}c16\}$); the resulting mix is strict-Pareto on all three subsets (selection, independent, full five-run) and is the row reported in Table 1.

Selection-grid stability. To test whether the two-level mix depends on its grid, we trained three further penalty levels per run ($c \in \{15, 17, 20\}$; 15 trainings and 15 full evaluations) and re-ran the identical a priori rule — minimum selection-subset CR, ties to the lower penalty — on the five-level grid. The selection subset is the first 690 episodes of each run in evaluation order (partitions 0–14) and the independent subset is the remaining 690; the two sets coincide on four runs and differ by 25 episodes on run 43. On the selection subset adjacent levels differ by at most five episodes (0.72 points), and the two picks that change do so at an exact tie (run 42, four levels at 2.46) or a three-episode gap (run 43, 2.32 against 2.75), so the rule is choosing among near-equivalent models (Table 24). The re-selected mix reaches $86.00/78.14/4.67$ on the independent subset against $85.77/78.05/4.81$ for the two-level mix of Table 1 and $84.99/77.11/5.01$ for PPO on the same subset (its full-set values are in Table 1): both mixes are strict-Pareto over PPO on this subset, as are three of the five fixed levels (c=15: $86.09/78.10/4.99$; c=16: $85.74/77.91/4.84$; c=20: $85.74/77.88/4.99$; c=17 at $85.62/77.82/5.01$ and c=18 at $85.65/77.90/5.07$ are not). The difference is that a fixed level can only be chosen by looking at held-out data, whereas the rule's pick is made on the selection subset alone: the two-level mix remains strict-Pareto on the selection, independent, and full sets (Table 23), and the five-level mix on the independent and full sets ($87.58/81.89/3.80$ against PPO's $86.45/80.88/4.09$ on the full set). The result therefore does not depend on which grid the rule is offered.

Run	c=15	c=16	c=17	c=18	c=20	Two-level pick	Five-level pick
42	**2.46**	3.04	**2.46**	**2.46**	**2.46**	c=18	c=15
43	**2.32**	2.75	2.75	3.04	2.46	c=16	c=15
44	2.90	3.48	3.04	**2.75**	3.04	c=18	c=18
45	3.91	4.49	3.91	**3.77**	3.91	c=18	c=18
46	4.06	**3.33**	**3.33**	4.06	3.48	c=16	c=16

Table 24: **Selection-subset CR (%) per run and penalty level.** Bold marks the minimum on the five-level grid. The two picks that change (runs 42 and 43) do so at a tie or a three-episode gap; the two-level rule is the one used for Table 1.

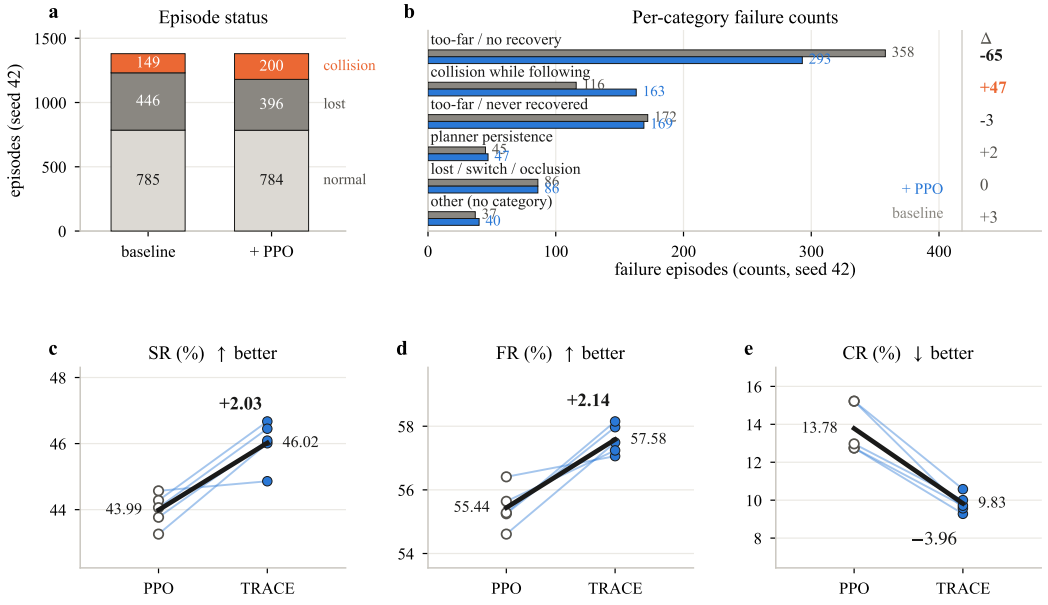

Figure 7: **DT ending-taxonomy evidence and five-run gate effect. a–b**, On one matched diagnostic run (seed 42) planner-head PPO leaves the number of normal endings essentially unchanged (785 to 784) while shifting the ending distribution from lost toward collision: too-far/no-recovery failures fall by 65 and collision-while-following failures rise by 47 (Table 25); the episode-level transitions (Table 3) show a net shift, not a one-to-one conversion. **c–e**, PPO→TRACE on all five independent training runs, aggregated by the official evaluator — the same convention as Table 1, so the panel means reproduce its paired row exactly (+2.03/+2.14/−3.96). Hollow = PPO, filled = TRACE, thin lines pair the runs, the black line joins the means. All three metrics improve on all five runs; CR falls by 2.97 to 5.65 points.

R Episode-Ending Taxonomy and Transition Analysis

Definition of the proxy. The diagnostic evaluator records, per step, a detector-to-panoptic overlap proxy for the selected target. The proxy is the IoU between the detector box of the policy's selected target and the panoptic instance mask of the ground-truth target. A successful identity is one where the proxy stays above a small threshold (≥ 0.20 in the released analysis) across most of the followed timesteps.

Failure taxonomy. We classify each non-success episode into one of five categories: too-far or no-recovery after following; too-far or never recovered; collision while following; lost after following with switch or occlusion; planner persistence or late failure. The categories are mutually exclusive but not exhaustive: 37 (4.5%) baseline and 40 (5.0%) PPO non-success episodes match none of them and are reported as *other*; see Figure 7b and Table 25.

DT ending diagnostic	Compact baseline	+ Planner-head PPO (run 42)	Change
SR / FR / CR	41.01 / 53.37 / 10.80	42.17 / 54.45 / 14.49	+1.16 / +1.08 / +3.69
Episode status: normal / lost / collision	785 / 446 / 149	784 / 396 / 200	-1 / -50 / +51
too-far / no-recovery after following	44.0% (358)	36.7% (293)	-65 failures
too-far / never recovered	21.1% (172)	21.2% (169)	-3 failures
collision while following	14.3% (116)	20.4% (163)	$+47$ failures
lost after following / switch / occlusion proxy	10.6% (86)	10.8% (86)	unchanged
planner persistence or late failure proxy	5.5% (45)	5.9% (47)	$+2$ failures
other (matches no category)	4.5% (37)	5.0% (40)	$+3$ failures

Table 25: **Full DT identity / failure-case diagnostic (single matched seed).** Numeric backing for Figure 7a–b. The SR/FR/CR triplets differ from the headline run-42 evaluation ($44.57/56.41/12.75$ for PPO) because the diagnostic evaluator is a separate pass whose per-step logging changes outcomes by up to 2.4 SR and 1.7 CR points; we use this run only for the failure taxonomy, never as a performance estimate. The per-category counts are from one matched seed; the aggregate five-run CR rise of $+2.77$ in Table 1 is consistent with the same shift.

Caveats. The proxy is detector-conditioned, so detector failure modes (missed detections, identity switches between detector tracklets) propagate into the proxy. We use the proxy only to compare distributions of failure categories under matched evaluators, not to claim absolute target-switch rates.

Multi-run status mix: the gate reverses the shift on every run. Table 25 reports the baseline→PPO direction on a single matched seed (run 42). Here we extend the diagnosis to the PPO→gate direction on *all five* matched DT runs, using the same full 1,380-episode protocol (Table 26):

Run	ΔSR	ΔFR	ΔCR	ΔNormal	ΔLost	ΔCollision
42	$+0.29$	$+0.65$	$\mathbf{-3.48}$	$+22$	$+26$	$\mathbf{-48}$
43	$+2.75$	$+2.88$	$\mathbf{-4.64}$	$+57$	$+7$	$\mathbf{-64}$
44	$+2.39$	$+1.60$	$\mathbf{-5.65}$	$+45$	$+33$	$\mathbf{-78}$
45	$+2.39$	$+2.72$	$\mathbf{-3.04}$	$+48$	-6	$\mathbf{-42}$
46	$+2.32$	$+2.87$	$\mathbf{-2.97}$	$+34$	$+7$	$\mathbf{-41}$
Mean	$\mathbf{+2.03}$	$\mathbf{+2.14}$	$\mathbf{-3.96}$	$\mathbf{+41.2}$	$\mathbf{+13.4}$	$\mathbf{-54.6}$

Table 26: **PPO→gate deltas per matched DT run (1,380 episodes each).** SR, FR and CR are aggregated by the official EVT-Bench aggregator — the same convention as Table 1, so the run means here reproduce its paired row exactly ($+2.03$/$+2.14$/-3.96). All three metrics improve on all five runs; endings shift from collision mostly to normal ($+41.2$ per run) and less to lost ($+13.4$), so the collision reduction is not purchased mainly by abandoning the target. Episode-level intervals for these deltas are in Table 2; we prefer them to the five-run one-sided sign test (p=0.031), which this table would also pass. Per-run exact McNemar p on collisions: DT below 2×10^{-4} on every run; AT $0.15/0.022/1.0/0.91/1.0$; STT $0.33/0.23/0.68/1.0/0.15$.

S Recovery Residual Dataset Construction

Source. Source episodes come from a full DT evaluation run with the TRACE gate enabled and identity debugging on. The run reports 1,380 episodes at 45.29 / 56.55 / 9.64 (SR / FR / CR) and provides target-state, simulator-pose, and recovery-event annotations per step.

Label windows. For each lost or too-far event, we mark a positive window of width K steps preceding the event. Three variants are produced: (i) the *default* variant with the longest positive window, yielding 11,214 rows at 56.14% positives; (ii) a *short-20* variant narrows the window and yields 23,831 rows at 20.07% positives; (iii) a *short-25* variant is intermediate at 19,232 rows and 28.22% positives. Negative rows are sampled uniformly from non-event timesteps in the same episodes.

Targets. Each row carries a ground-truth delta $(\Delta v_x, \Delta v_y, \Delta\omega)$ toward the target in the corrected robot frame ($+X$ forward). Rows without a valid robot-frame pose are dropped at construction time; the released datasets contain zero missing-pose rows.

Partitions. Each variant is partitioned episode-disjointly into train / val / test partitions (0.8 / 0.1 / 0.1) so that supervised checks are evaluated on episodes unseen during head training.

T Robot-Frame Grounding Pipeline

Ground-plane convention. The initial target-grounding data collectors used local $-Z$ as the ground-plane forward axis. This axis is close to vertical for the embodied agent we study, while the executed base velocity uses local $+X$. Bearings computed against $-Z$ are therefore systematically rotated by approximately ninety degrees relative to the controller's actual forward direction.

Corrected convention. The corrected collector projects target position into the local ground-plane frame using local $+X$ as forward and local $+Y$ as left. Bearing is the signed angle between the projected target vector and the forward axis. The same convention is used to label left / front / right buckets in the failure taxonomy.

Effect on the deployable probe (offline) and verification. We retrain the deployable target-direction probe on the corrected labels. Under the $-Z$ convention the probe was below chance (13.4% angle L/F/R, 6.5% high-confidence), making it unusable as a closed-loop signal. Under the corrected $+X$ convention it reaches 59.9% / 53.0% on the primary model and 56.9% / 52.8% on a second model (Figure 8b), comfortably above chance on the high-confidence subset—the regime relevant for recovery control. Closed-loop verification follows a two-gate protocol (a probe criterion for the direction signal alone, a recovery criterion for the controller on top); a configuration is carried forward only after meeting both on a second model, so any future recovery claim is attributable to the probe correction *or* the controller, never to a mixture.

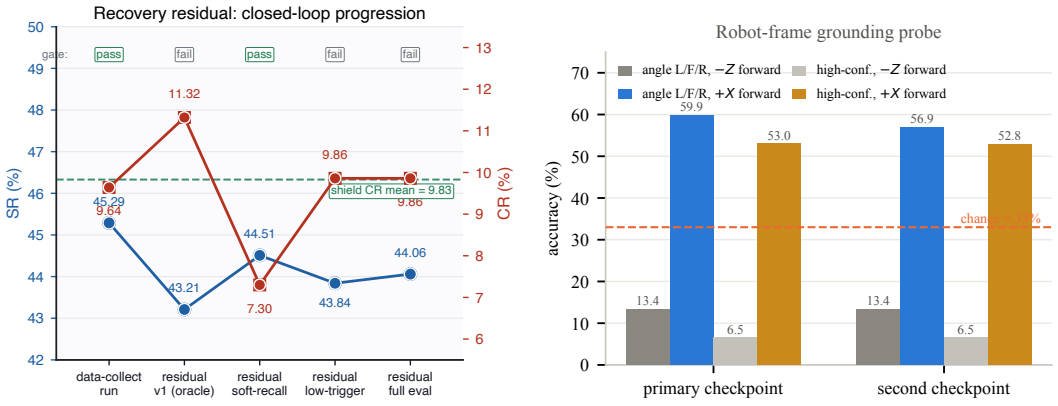

Figure 8: **Recovery-residual progression and robot-frame grounding probe.** **(a)** Supervised recovery residuals, closed-loop progression: SR (blue, left axis) and CR (red, right axis) for the data-collection run and four residual configurations. Dashed green line marks the TRACE gate CR (five-run mean 9.83); soft-recall meets the validation promotion criterion (CR 7.3) but its full DT seed-42 evaluation rises back above the gate, motivating a diagnosis based on the target's state. **(b)** The robot-frame grounding correction recovers the deployable direction probe: validation angle L/F/R and high-confidence-subset accuracy before (grey, $-Z$ forward) and after (blue / ochre, $+X$ forward) the convention correction, on two trained models; dashed line is 33% chance.