GROUNDING WHAT SHAPES THE PLAN: RETHINKING GROUNDEDNESS FOR PHYSICAL INTELLIGENCE IN AUTONOMOUS DRIVING

Anonymous authors
Paper under double-blind review

ABSTRACT

Driving models increasingly ground reasoning in causal relations, spatial structure, perceptual evidence, and predicted futures. These advances make reasoning more faithful to the driving scene, but leave a fundamental question unresolved: *what should groundedness mean when the model ultimately outputs an action?* Correctly grounded reasoning does not, by itself, ensure desirable driving outcomes. We introduce **GroundAct**, which starts from a simple premise: driving unfolds through physical entities and their interactions. Entities therefore become the unit of grounding; a lightweight reference token keeps each selected entity's continuous state addressable through symbolic reasoning; and only the referenced entities' interactions with the evolving proposal correct the plan. The result is an explicit path from what reasoning grounds to what the plan does, which we call *grounded planning*. To assess its practical value, we evaluate GroundAct in both open- and closed-loop settings. GroundAct shows strong open-loop planning across normal, out-of-distribution, and safety-critical scenarios, with closed-loop results extending this evidence to driving in simulation.

1 INTRODUCTION

Autonomous-driving models increasingly incorporate reasoning for its potential to support generalization across diverse and long-tail scenarios, grounding it in causal relationships, spatial structure, perceptual evidence, and predicted future states (Zeng et al., 2025; Shang et al., 2026; Peng et al., 2026; Tan et al., 2026). Structured reasoning traces further connect grounded observations and predictions to high-level intentions for how the vehicle should behave (NVIDIA et al., 2025c). Collectively, these advances improve what a driving model can explain about the scene, why a decision may be appropriate, and what may happen next.

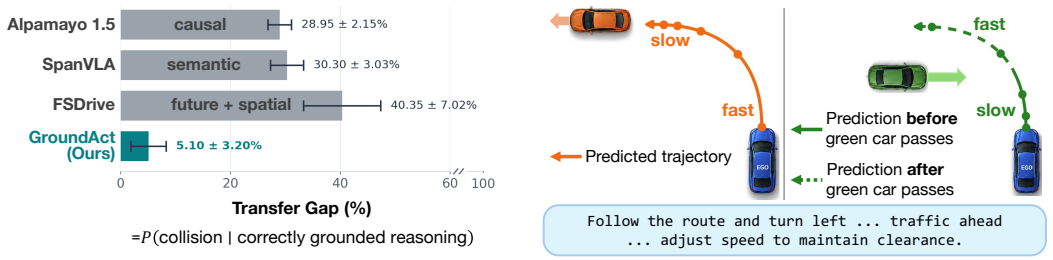
Yet reasoning is not the endpoint of driving: the output is a plan. Correctly grounded reasoning does not by itself ensure desirable driving outcomes, such as safety and comfort. Our motivating audit measures this *transfer gap*: the collision rate in the audited interactions, conditioned on correctly grounded reasoning (Figure 1(a)). This gap appears across causal, semantic, and spatiotemporal grounding. Figure 1(b) further illustrates how the same reasoning trace can arise in different driving contexts, even when the appropriate speed profiles differ. Swapping the speed profiles between the two contexts can lead to safety-critical outcomes, such as collisions, despite correctly grounded reasoning in both cases.

This leads to our central question:

What should groundedness mean when the output is an action?

Our answer begins from a physical-AI view of driving: the ego vehicle acts in a world of physical entities and their interactions. Realizing a maneuver requires accounting for where relevant entities are, how they move, and how the ego plan interacts with them. We therefore choose the *entity* as our grounding unit: it provides a common physical referent linking what reasoning identifies as relevant to what planning must account for. Our principle is to carry grounded reasoning into action generation by preserving this physical connection, so that the selected entities' interactions shape the evolving plan and improve the resulting driving behavior. We call this *grounded planning*.

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.



(a) Grounding-to-Plan Transfer Gap. Measured on PAI-AV* safety-critical samples (Appendix A).

(b) Same Reasoning Trace, Different Actions Required. Blue: ego; orange/green: surrounding vehicles.

Figure 1: Grounded reasoning does not by itself ensure desirable driving outcomes.

In this work, we introduce **GroundAct**, whose *Token-as-Reference* interface interleaves generated reasoning content with references to continuous entity states (Figure 2). Resolving these references gives subsequent reasoning and planning access to the same selected states. The action expert retains the full scene context, while an explicit refinement pathway evaluates the referenced states against the evolving ego proposal and uses their predicted interactions to guide selective correction.

We evaluate GroundAct on three PAI-AV-based open-loop benchmarks (Normal, OOD, and Safety-Critical) and the AlpaSim closed-loop driving benchmark (NVIDIA, 2025; NVIDIA et al., 2025b). With 3.9B parameters, GroundAct demonstrates strong planning and safety performance across these benchmarks, remaining competitive with the 34B state-of-the-art driving foundation model (NVIDIA, 2026b) and outperforming it on several metrics. Controlled ablations test whether reasoning-selected physical states remain consequential for planning by removing or replacing their references and by comparing reasoning-based selection against heuristic alternatives. Matched correction analyses further examine whether GroundAct corrects unsafe plans while preserving those that are already safe.

Our contributions are: (i) **Grounded planning**, which establishes an explicit and traceable path from the entity grounded in reasoning to its consequence for action; (ii) **Token-as-Reference**, which keeps continuous entity state in the model while reasoning selects planning-critical entities through lightweight symbolic references; and (iii) **Interaction-driven selective correction**, which uses the referenced entities' predicted interactions with the evolving ego plan to guide selective refinement.

2 Background and Related Work

Grounded Reasoning as Token-as-Content. Driving VLAs ground their reasoning in three main forms, as illustrated on the left of Figure 2. *Causally grounded* reasoning states the factors behind a decision in language, as in Alpamayo-R1's causal chains (NVIDIA et al., 2025c) and in adaptive or self-reflective reasoning before action (Zhou et al., 2025; Peng et al., 2026; Zhou et al., 2026). *Spatially grounded* reasoning carries positions: object coordinates written into the reasoning text (Tian et al., 2025) or object boxes drawn into generated frames (Zeng et al., 2025). *Future-state grounded* reasoning incorporates predicted futures, represented as image tokens (Zeng et al., 2025), latent tokens (Tan et al., 2026; Luo et al., 2026; Jia et al., 2026), or compact dynamics tokens (Shang et al., 2026). Despite these differences, the three forms share one interface, which we call *Token-as-Content*: whatever is grounded is expressed in the generated tokens and is consumed as such by subsequent reasoning or action generation. Grounding is thereby a property of the reasoning process; how that grounding should remain consequential for action generation is left unspecified. Token-as-Reference draws inspiration from recent visual-grounding work: Chung et al. (2025) points to image patches and copies their embeddings into the reasoning stream, while Zhang et al. (2026) serializes explicit points or boxes. Whereas these methods focus on grounding intermediate reasoning, GroundAct preserves the reference link from reasoning through downstream planning.

Related Directions. GroundAct draws on two further lines of work, discussed in Appendix B. Prediction- and interaction-aware planners show that structured prediction and physical interaction can shape planning directly (Hu et al., 2023; Huang et al., 2023; Sadat et al., 2020; Li et al., 2026; Jia et al., 2026); GroundAct asks which entities should receive that computation and how the choice stays connected to reasoning. Guided and residual generative planners modify a pretrained generator

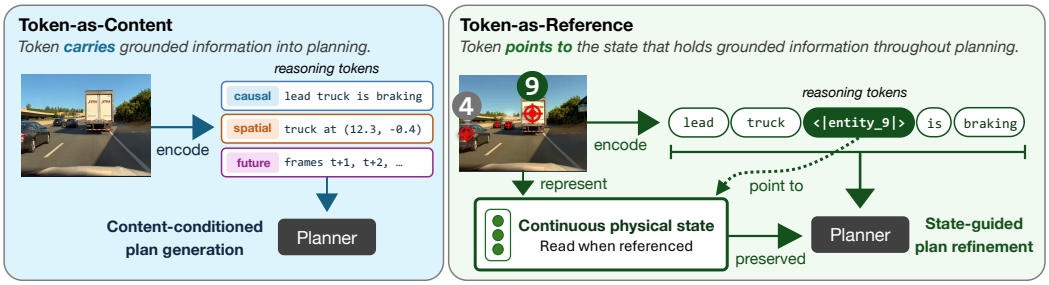


Figure 2: **Token-as-Content *vs.* Token-as-Reference. [Left]** Grounded information is encoded in generated content. **[Right]** GroundAct retains this content path and adds entity-reference tokens <|entity_i|>, each indexing a continuous entity state. These references keep the planning-critical entities selected during reasoning addressable throughout reasoning and planning. Only the referenced states enter the explicit refinement pathway.

without replacing it (Lipman et al., 2023; Song et al., 2021; Zhang et al., 2023a; Jing et al., 2026); GroundAct uses individual referenced entities, rather than a global intent, as the mediator. Both lines make predictions or guidance signals consequential for planning; GroundAct makes the *reasoning's* grounding consequential, by keeping the selected entities addressable until they act on the plan.

3 GroundAct

3.1 Overview

GroundAct instantiates our three design choices: entity as the grounding unit, *Token-as-Reference* as the grounding mechanism, and interaction-driven correction for planning. These choices span continuous multimodal encoding, discrete autoregressive reasoning, and iterative continuous action generation (Figure 3).

Figure 3 summarizes the computation. A physical entity is first represented by a continuous, addressable state (§3.2). During reasoning, an entity mention is associated with its corresponding state in the current entity pool and expressed through a symbolic reference, without serializing the underlying geometry into text (§3.3). For planning, the referenced states are retrieved and explicitly evaluated against the evolving ego proposal; their predicted futures and physical interactions then produce selective corrections to the generated action (§3.4). The representation changes across these computations, from continuous state to discrete reference to interaction feature, while the physical referent remains fixed.

3.2 Prefill-Time Grounding

For reasoning and planning to refer to the same physical entity, the model first needs a continuous state that is aligned with that entity and can be addressed later. Given temporal observations and driving context O, we append K entity tokens $q_1, \ldots, q_K$ (K=64) to the multimodal prefill. Each acts as a learnable query that attends to and aggregates temporal visual features together with the surrounding context; its final hidden state

$$M_i = H_\theta(O, q_1, \ldots, q_K)[q_i],$$

the Transformer output H_θ read at the position of q_i, is the entity state at position i. The entity states $\mathcal{M} = \{M_i\}_{i=1}^{K}$ stay inside the model, each addressable by its index i.

Entity-Level Supervision. Attention gathers scene evidence; entity-level 3D supervision determines what each state comes to represent. Training-only heads decode an object category, including a null class, and an oriented BEV box from each M_i; Hungarian matching assigns predictions one-to-one to annotated entities, and unmatched queries receive the null target. The index i is therefore scene-local—the same position represents different entities in different scenes—so reasoning and planning must bind to the state an index currently addresses, not to the index itself. The entity tokens and heads are trained with this supervision jointly with the language-model objective, warming up the new modules before fine-tuning the pretrained backbones (Appendix F).

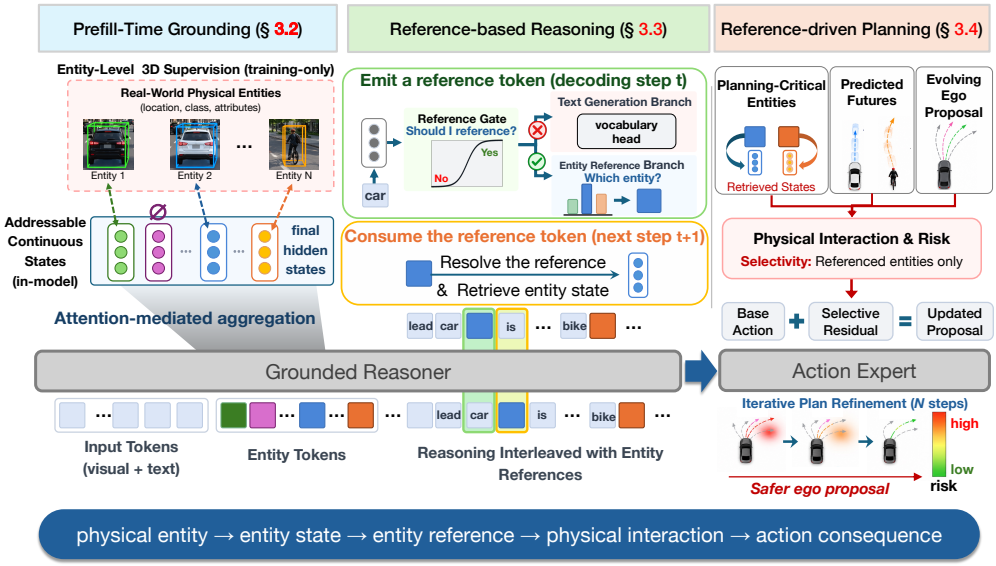


Figure 3: **GroundAct Architecture.** The three columns correspond to §3.2 through §3.4. The row below follows the computation from input tokens through the Physical AI reasoner to the action expert's refined plan, while the bottom bar summarizes the design invariant preserved throughout.

3.3 Reference-based Reasoning

Reasoning interprets the driving situation and the intended maneuver, and in doing so singles out the planning-critical entities. To carry this choice into subsequent reasoning and planning, *Token-as-Reference* binds a discrete reference $e_i = \langle \text{entity } i \rangle$ to the continuous prefill state M_i. The token names the entity; its physical information remains in the addressable state.

Reference Generation. At decoding step t, let h_t be the reasoning hidden state and y_{t+1} the next token. A gate assigns probability mass π_t to references, and an entity pointer distributes it according to the correspondence between h_t and the current states:

$$p(y_{t+1} = e_i \mid h_t, \mathcal{M}) = \pi_t \, \text{softmax}_i[\tau \cos(W_q h_t, W_k M_i)], \qquad \pi_t = \sigma(g(h_t)). \tag{1}$$

Here W_q, W_k are learned projections, $\tau > 0$ controls score sharpness, and g is a scalar gate. The pointer softmax excludes detector-null slots whenever non-null candidates are available. The vocabulary head distributes the remaining mass $1 - \pi_t$ over non-reference tokens. Here, the division of probability mass between vocabulary and reference outputs is motivated by the gated pointer-generator models (See et al., 2017). This separates *when* to reference from *which* entity to name within one next-token distribution.

Reference Resolution and Consumption. When $y_{t+1} = e_i$, resolving the address directly reads the original prefill state M_i from the pool. The next decoding step consumes this state through the input embedding u_{t+1}:

$$\text{Resolve}(e_i, \mathcal{M}) = M_i, \qquad u_{t+1} = (1 - \alpha) b_{\text{ref}} + \alpha \big(b_i + P(M_i)\big). \tag{2}$$

Here b_i combines the entity-query base embedding and slot identity, P projects and scales the retrieved state into the decoder input space, and $\alpha \in [0, 1]$ interpolates between this input and a shared reference embedding b_{ref}. The decoder processes u_{t+1} with its existing context to produce a new reasoning hidden state; the retrieved M_i remains the prefill representation. Planning reads the same M_i (§3.4), preserving the selected physical referent across both computations.

Association Supervision. The reference targets are physical objects, not indices. Offline annotation links each accepted referring expression in the reasoning text to an annotated object. At each training pass, the valid entities are identified by BEV-center distance before assigning a reference label. Text prediction, gating, and association are jointly trained with a regularizer that anchors the entity states to those learned in §3.2. See Appendix F for further details.

3.4 REFERENCE-DRIVEN PLANNING

A referenced entity does not prescribe an action: what it implies depends on where the ego proposal (i.e., the planner's current candidate trajectory) is heading. Let $\mathcal{R}(r) \subseteq \{1, \ldots, K\}$ denote the entities referenced in reasoning sequence r. Only their states $\mathcal{M}_{\mathcal{R}} = \{M_i : i \in \mathcal{R}(r)\}$ enter explicit future prediction and interaction modeling, while the action expert—a flow-matching trajectory generator conditioned on the complete multimodal and reasoning context—keeps its full view of the scene. Reasoning thus decides where additional physical computation is spent.

Interaction with the Evolving Proposal. In the continuous action mode, the action expert produces a trajectory by integrating a learned vector field over a fixed number of solver steps. At step k, it evaluates the base velocity $v_{\text{base}}^{(k)}$ at the current flow state $x^{(k)}$; from the two we decode the complete trajectory the current field implies, the ego proposal $\hat{P}^{(k)}$. For each referenced entity, a future predictor forecasts its motion $\hat{S}_i^{(k)}$ over the planning horizon (in the joint variant, conditioned on the current proposal), and the proposal-relative interaction

$$C_i^{(k)} = \text{Interact}\left(\hat{P}^{(k)}, \hat{S}_i^{(k)}\right), \qquad i \in \mathcal{R}(r),$$

encodes quantities such as clearance, relative motion, and validity over that horizon. Because the proposal changes at every step, so do these interactions; k indexes refinement of the whole proposal, not time within the horizon.

Selective Correction. A permutation-invariant set encoder aggregates the referenced states, their forecasts, and their interactions, and a residual adapter R_ϕ maps the result, together with the current state and base velocity, to a correction of the base field:

$$z_{\mathcal{R}}^{(k)} = \text{SetEnc}\left(\left\{M_i, \hat{S}_i^{(k)}, C_i^{(k)}\right\}_{i \in \mathcal{R}(r)}\right), \qquad v_{\text{GroundAct}}^{(k)} = v_{\text{base}}^{(k)} + \gamma^{(k)}\, R_\phi\left(x^{(k)}, v_{\text{base}}^{(k)}, z_{\mathcal{R}}^{(k)}\right),$$

where $\gamma^{(k)}$ is a scalar gate driven by the predicted interactions, so the correction scales with the physical risk they indicate. The solver integrates the corrected field, and the new proposal is evaluated again. The correction is defined jointly over the referenced set, and the pathway is bypassed (i.e., recovering the base field) when no valid entity is referenced or the gate is inactive.

Correction Supervision. With grounding, reasoning, and the action expert frozen, we train the future predictor and the residual pathway. Ground-truth entity futures (e.g., 3D positions) supervise the forecasts. The correction is supervised by safer alternative actions computed offline using ground-truth entity futures and accepted only if they reduce physical risk while remaining valid scene-wide; each defines a residual target relative to the frozen base field at the flow state from which it was generated. For already-acceptable proposals, preservation terms encourage the corrected action to remain close to the frozen base action, discouraging unnecessary intervention. We evaluate the resulting preservation behavior in §4.3.

4 EVALUATION

4.1 EXPERIMENTAL SETUP

Models and Comparisons. We instantiate GroundAct with a Qwen3-VL-2B vision–language backbone (Bai et al., 2025) and a flow-matching action expert (2.1B and 1.8B parameters; Table 1). Actions are generated either as discrete action tokens from the VLM backbone or as a continuous trajectory from the action expert; the selective correction of §3.4 modifies the expert's vector field, so it is active only in the continuous mode. We nonetheless report discrete generation in the open-loop evaluation, which isolates the effect of entity grounding on downstream reasoning from that of the correction. We compare against Alpamayo-R1 (NVIDIA et al., 2025c), Alpamayo 1.5 (NVIDIA, 2026a), Alpamayo 2 Super (NVIDIA, 2026b), FSDrive† (Zeng et al., 2025), LCDrive (Tan et al., 2026), AutoVLA† (Zhou et al., 2025), and SpanVLA† (Zhou et al., 2026). The Alpamayo models and LCDrive are evaluated from their released checkpoints without fine-tuning. FSDrive†, AutoVLA† and SpanVLA† are the matched controls: each method's defining component is re-implemented on our backbone and fine-tuned, without subsequent reinforcement learning, on the same 140.5K PAI-AV training samples as GroundAct. See Appendix H for further details.

Table 1: **Open-Loop Driving Performance by Grounding Paradigm.** Generation mode: (D) discrete; (C) continuous; best, second, and third distinct values per column.

Model	Token represents	Size	Mode	PAI-AV Normal (n=4K) ADE 2.5 s	ADE 5.0 s	ADE 6.4 s	Collision 2.5 s	Collision 5.0 s	Corner Dist. 6.4 s	PAI-AV OOD (n=290) ADE 2.5 s	ADE 5.0 s	ADE 6.4 s	Collision 2.5 s	Collision 5.0 s	Corner Dist. 6.4 s
Token-as-Content															
FSDrive†	spatio-temporal states	2.1B	D	0.417	1.809	3.112	2.943	10.721	3.231	0.546	2.145	3.404	3.549	20.469	3.480
Alpamayo-R1	causes	10B	D	0.551	2.612	4.417	4.269	13.443	4.599	0.596	2.691	4.402	5.805	24.844	4.568
Alpamayo-R1	causes	10B	C	0.535	2.263	3.802	3.714	12.472	3.945	0.541	2.295	3.751	4.182	21.036	3.872
AutoVLA†	semantic knowledge	2.1B	D	0.555	1.780	2.848	2.557	9.014	2.910	0.563	1.973	3.096	2.809	14.170	3.148
LCDrive	action-conditioned futures	0.6B	D	0.332	1.503	2.615	2.292	9.100	2.676	0.475	1.994	3.268	2.809	17.353	3.330
Alpamayo 1.5	causes	10B	D	0.425	1.881	3.163	3.311	10.951	3.279	0.522	2.148	3.412	3.995	21.223	3.513
Alpamayo 1.5	causes	10B	C	0.575	2.247	3.647	3.952	12.806	3.770	0.585	2.240	3.528	4.370	21.848	3.627
SpanVLA†	semantic knowledge	3.8B	C	0.496	1.818	2.844	2.531	8.339	2.931	0.482	2.059	3.167	2.310	14.232	3.532
Alpamayo 2 Super	causes	34B	D	0.345	1.338	2.245	2.591	6.028	2.304	0.470	1.696	2.683	2.871	9.613	2.756
Alpamayo 2 Super	causes	34B	C	0.299	1.304	2.214	2.097	5.473	2.260	0.391	1.606	2.601	2.060	8.864	2.645
Token-as-Reference (Ours)															
GroundAct	entity-state addresses	3.9B	D	0.266	1.204	2.091	2.149	5.967	2.138	0.371	1.545	2.540	2.185	11.923	2.592
GroundAct	entity-state addresses	3.9B	C	0.325	1.332	2.210	2.474	6.873	2.278	0.459	1.692	2.680	2.310	10.861	2.747

Model	Token represents	Size	Mode	PAI-AV* Safety-Critical (n=512) ADE 2.5 s	ADE 5.0 s	ADE 6.4 s	Collision 2.5 s	Collision 5.0 s	Off-road 2.5 s	Off-road 5.0 s	Corner Dist. 6.4 s
Token-as-Content											
FSDrive†	spatio-temporal states	2.1B	D	0.533	2.249	3.823	3.797	20.161	5.712	28.394	3.992
Alpamayo-R1	causes	10B	D	0.977	4.084	6.584	6.770	29.490	12.695	37.565	6.939
Alpamayo-R1	causes	10B	C	1.014	4.054	6.521	7.230	28.840	14.388	38.411	6.872
AutoVLA†	semantic knowledge	2.1B	D	0.570	1.907	3.027	1.465	14.290	4.297	17.383	3.069
LCDrive	action-conditioned futures	0.6B	D	0.384	1.585	2.633	1.074	12.370	1.530	9.668	2.709
Alpamayo 1.5	causes	10B	D	0.782	3.278	5.281	6.413	29.330	10.286	36.198	5.548
Alpamayo 1.5	causes	10B	C	0.865	3.384	5.396	6.152	29.460	10.221	37.402	5.665
SpanVLA†	semantic knowledge	3.8B	C	0.561	2.213	3.182	1.302	15.625	2.539	17.643	5.195
Alpamayo 2 Super	causes	34B	D	0.528	1.768	2.797	1.335	7.585	3.092	7.878	2.887
Alpamayo 2 Super	causes	34B	C	0.384	1.515	2.491	0.390	7.420	1.497	7.585	2.558
Token-as-Reference (Ours)											
GroundAct	entity-state addresses	3.9B	D	0.382	1.528	2.506	0.553	7.096	1.107	6.868	2.580
GroundAct	entity-state addresses	3.9B	C	0.431	1.565	2.501	0.488	9.668	1.042	8.496	2.571

Driving Evaluation Protocols. Open-loop evaluation uses the PAI-AV Normal (4K samples) and OOD (290 samples) splits, along with 512 high-risk samples from PAI-AV* collected under the same data configuration (Appendix D). All models receive four temporally ordered front-wide camera frames (t−0.3 s, t−0.2 s, t−0.1 s, t) and generate six candidate trajectories over a 6.4 s horizon. Following LCDrive (Tan et al., 2026), we report ADE, corner distance, collision rate, and off-road rate averaged over the six candidates. Closed-loop evaluation uses 913 scenes from the AlpaSim 'public_2601' test suite (NVIDIA et al., 2025b), with the same four-frame input. We report progress, three safety metrics, and two diagnostic distances (Table 2; Appendix G).

4.2 Driving Performance

Open-Loop Planning. Table 1 evaluates GroundAct under both discrete and continuous generation. Across all three datasets, GroundAct achieves the best or comparable performance on nearly all metrics, despite being substantially smaller than Alpamayo 2 Super. The discrete variant operates solely with the 2.1B VLM, rather than the full 3.4B model. Here, entity references retrieve the corresponding physical states into the reasoning process, yielding strong trajectory accuracy across all three datasets and long-horizon safety on the safety-critical set. The continuous variant additionally incorporates state-based plan correction through the action expert. Its performance remains competitive but slightly trails Alpamayo 2 Super on several metrics. Table 4 attributes this gap to the flow-matching head (i.e., action expert trained from scratch on substantially less data and without Alpamayo's RL post-training) rather than to the correction itself. Together, these results show the benefit of entity grounding in both reasoning and continuous plan refinement.

Table 2: **Closed-Loop Driving Performance on AlpaSim.** Each model uses its representative generation mode: discrete generation for FSDrive, AutoVLA, and LCDrive, and continuous generation otherwise; best, second, and third distinct values for Progress and Safety Metrics. Metric definitions are provided in Appendix G.

			Safety Metrics ↓			Additional Analysis (m)	
Model	**# Params**	**Progress ↑**	At-fault collision	Off-road	Wrong lane	Distance to GT Traj.	Min. obstacle distance
FSDrive† (Zeng et al., 2025)	2.1B	0.604	0.193	0.271	0.210	2.577	4.447
Alpamayo-R1 (NVIDIA et al., 2025c)	10B	0.682	0.156	0.189	0.295	2.781	4.649
AutoVLA† (Zhou et al., 2025)	2.1B	0.706	0.118	0.185	0.211	2.544	3.989
LCDrive (Tan et al., 2026)	0.6B	0.639	0.259	0.263	0.189	2.825	3.871
Alpamayo 1.5 (NVIDIA, 2026a)	10B	0.682	0.166	0.155	0.297	2.843	4.407
SpanVLA† (Zhou et al., 2026)	3.8B	0.696	0.156	0.184	0.207	2.896	3.794
Alpamayo 2 Super (NVIDIA, 2026b)	34B	0.713	0.084	0.147	0.220	2.799	3.721
GroundAct	3.9B	0.752	0.105	0.153	0.207	2.511	3.641

Table 3: **Reference Dependence and Selective Correction.** (a) Collision-rate increase relative to Ours (pp). *Matched distractor*: another same-class entity at a similar distance, with at least 1 m clearance from the base plan. *Highest predicted risk*: the entity with the highest predicted risk to the ego plan. (b) *Preserved (X m)*: percentage of initially safe plans whose maximum time-aligned distance from the base plan is at most X m over 6.4 s.

(a) Reference-to-Action Dependence

Reference Selection	PAI-AV* Safety	PAI-AV OOD
No reference	+4.54%	+7.03%
Matched distractor	+3.15%	+7.67%
Nearest entity	+1.43%	+2.61%
Highest predicted risk	+1.26%	+1.51%

(b) Selective Correction

Outcome	Rate
Safe → Unsafe (Collision or Off-road)	2.2%
Preserved (0m; exactly the same)	53.9%
Preserved (3m)	70.4%

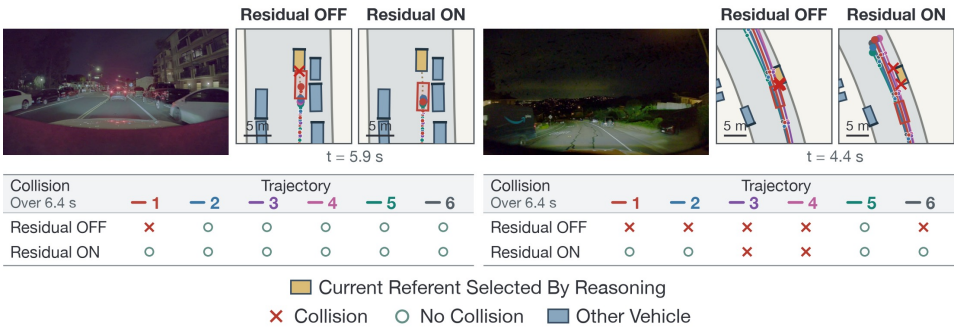


Figure 4: **Selective Correction.** BEV details show six matched trajectories.

Closed-Loop Driving. In Table 2, GroundAct, with the correction of §3.4 active, attains the highest progress, ahead of the much larger, RL post-trained Alpamayo 2 Super, and is the only model in the top two on progress and all three safety metrics. It is better than or tied with every matched control, which shares its backbone and training data, on each of these metrics, including SpanVLA† at comparable size. It is also comparable to Alpamayo 2 Super, with only marginal gaps in at-fault collision and off-road performance. The diagnostic columns indicate that this safety does not come from conservative margins: GroundAct deviates least from the GT path despite driving furthest and has the smallest clearance to other actors, consistent with selective correction (Table 3(b)), whereas the two earlier Alpamayo models keep wider clearance yet show the highest wrong-lane rates.

4.3 Ablation and Analysis

We examine three aspects of GroundAct's grounding pathway: whether the grounding connection should be preserved through action refinement, whether the model learns current entity–reference associations, and whether reasoning-selected references remain consequential for planning and support selective correction.

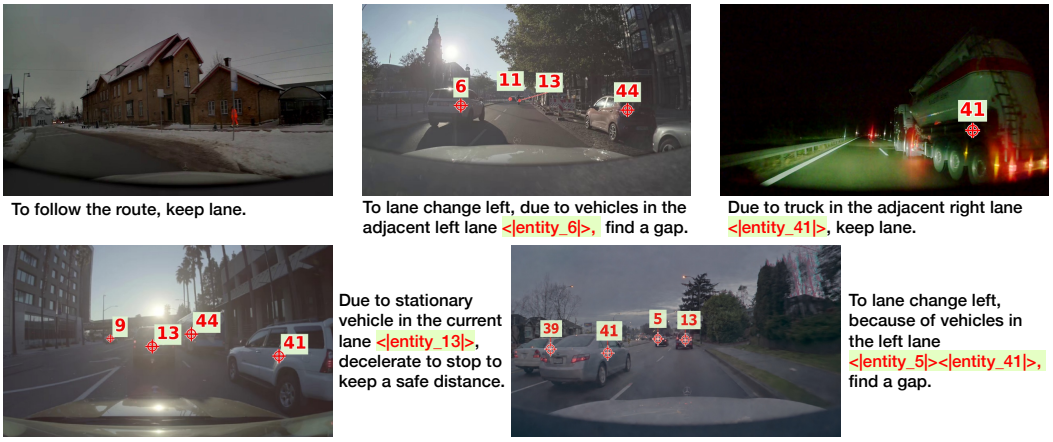


Figure 5: Reference-Based Reasoning with Varying Entity References.

Table 4: **Make Grounding Consequential.** Continuous generation; **Best**; second-best.

	VLM backbone	Spatial grounding	Grounded reasoning	**GroundAct (Ours)**
Ground	–	✓	✓	✓
Reference	–	–	✓	✓
Act	–	–	–	✓
Trajectory (m)↓				
ADE 2.5 s	0.551	0.479	0.463	**0.431**
ADE 5.0 s	2.308	1.893	1.651	**1.565**
ADE 6.4 s	3.498	2.985	2.581	**2.501**
Safety (%)↓				
Coll. 2.5 s	1.465	1.237	0.716	**0.488**
Coll. 5.0 s	17.741	13.639	11.979	**9.668**
Off-road 2.5 s	2.637	1.855	1.302	**1.042**
Off-road 5.0 s	19.857	17.415	12.988	**8.496**

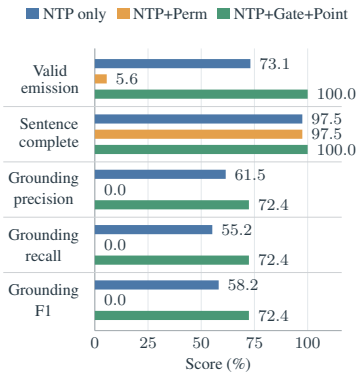


Figure 6: **Learn to Reference.**

Grounding Connection. Table 4 examines the value of carrying grounded information through reasoning into action refinement. With a shared backbone and continuous-generation protocol, grounded reasoning improves on spatial grounding alone, and interaction-driven correction further reduces trajectory error, collision rates, and off-road rates. These results support the combined design, without isolating each component's standalone contribution.

Learning Current Entity Associations. Under the reported evaluation, successful language generation can coexist with failed entity referencing (Figure 6). The comparison supports explicitly supervising both when to refer and which current entity state to select, accommodating changing entity–token associations.

Reference Dependence and Selection. The intervention audit shows that references are consequential for planning, with evidence of safety utility on the Safety cohort (Table 3(a)). It does not establish that reasoning-based selection is safer than the tested heuristics, or equivalent to them.

Selective Correction and Preservation. The final policy can repair unsafe plans while preserving some already-safe outputs, but may also introduce new safety issues (Table 3(b)) or leave existing ones insufficiently corrected (Figure 4). Preservation reflects both correction and final arbitration and should therefore be attributed to the complete policy. We discuss these failures further in §4.4.

Grounding Reasoning through Entity References. Figure 5 illustrates how GroundAct references specific perceived entities when reasoning about following, lane keeping, lane changing, and stopping. These references enable retrieval of the corresponding continuous physical states, grounding subsequent reasoning and planning in the entities selected during reasoning. Their physical states and interactions then guide downstream decisions and shape the evolving plan.

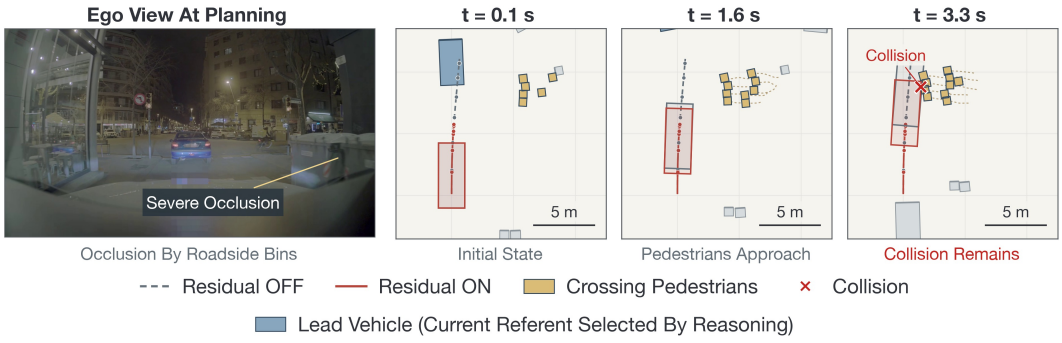


Figure 7: **Failure Case.** Pedestrians occluded at planning time emerge during execution.

4.4 Additional Analysis

Reasoning Alignment with Predicted and GT Behaviors. We further investigate how grounding affects reasoning–action alignment and alignment with ground-truth behavior. Specifically, we measure the consistency between meta-actions (e.g., stop, decelerate) inferred from the reasoning and those derived from either the predicted or ground-truth trajectories (Table 5). Our method yields substantial improvements in both settings. We attribute these gains to two key factors: (1) improved spatial awareness of surrounding objects and (2) an explicit grounding chain that links reasoning to planning.

Table 5: **Reasoning-Action Alignment.** (D) discrete; (C) continuous. Appendix C.

Model	F1	Recall	Precision
vs. meta-action from predicted trajectory			
Alpamayo-1.5 (D)	55.98	47.24	**68.70**
Alpamayo-1.5 (C)	51.61	44.74	60.96
GroundAct (Ours) (C)	**68.44**	**69.38**	67.52
vs. meta-action from GT trajectory			
Alpamayo-1.5 (D)	58.48	48.77	73.03
Alpamayo-1.5 (C)	58.52	49.06	72.51
GroundAct (Ours) (C)	**73.79**	**71.81**	**75.89**

Object Detection and Referencing. We follow Rex-Omni (Jiang et al., 2026), which evaluates object detection and pointing in 2D images, and adapt its evaluation protocol and metrics to BEV space. See Appendix C for evaluation settings and baseline details. As shown in Table 6, GroundAct performs strongly on both detection and geometry-aware referencing of planning-critical entities. These results indicate that our entity representation captures both semantic identity and spatial geometry, providing the physical information needed to ground prediction and planning in entity states and compute their physical risks.

Table 6: **Detection and Referencing.** Appendix C for baseline details.

	Detection (%)		Referencing (%)		
Method	mAP	NDS (BEV)	F1@ Point	F1@ IoU	Critical mIoU
WildDet3D^z	11.71	17.07	28.00	19.43	52.99
FCOS3D	27.07	44.40	49.30	36.25	69.22
MonoDETR	21.98	39.43	43.26	30.72	74.52
PETR	33.52	46.75	41.11	26.50	69.02
PETRv2	31.27	47.95	**55.02**	40.09	70.29
GroundAct	**35.96**	**48.65**	54.52	**42.44**	**75.20**

* WildDet3D^z: zero-shot; others: fine-tuned.

Failure Case: Stale References. The correction acts only on entities referenced during reasoning, using their states at planning time. After the plan is generated, references can become outdated when occluded entities emerge (Figure 7) or surrounding entities move rapidly. In either case, the correction may lack up-to-date referents to act on. Despite this limitation, GroundAct maintains strong planning performance in the safety-critical and OOD scenarios in Table 1. A natural next step is anticipatory reasoning: predicting future states to identify which entities will become planning-critical, and when, during execution.

5 Conclusion

We revisit what groundedness should mean when the output is an action. GroundAct preserves access to the continuous states of reasoning-selected entities through Token-as-Reference, allowing their predicted interactions to selectively shape the ego vehicle's evolving plan. Open- and closed-loop evaluations show strong planning performance, while a controlled audit shows that disabling the referencing pathway from reasoning to planning can compromise driving safety. These results support *grounded planning*: preserving the physical connection between what reasoning identifies as relevant and how the vehicle acts.

AI USE DISCLOSURE

In this work, we used generative AI tools to identify relevant literature, implement methods, generate synthetic training data for our reference-based reasoning method, assist in analyzing and interpreting experimental results, provide feedback on research methodology and experiments, and suggest experimental parameters. Additionally, we used generative AI tools to search for information, revise manuscript text, prepare figures and tables, and edit the manuscript for readability. The authors reviewed and verified all AI-assisted content, code, and analyses: AI-suggested references were checked against the original sources, and AI-assisted analyses were discussed among the co-authors. We take full responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work uses the publicly released PAI-AV dataset and AlpaSim scenes as distributed; no human-subject data were collected. GroundAct is a research model for studying how grounded reasoning should shape planning, not a deployable driving system; its safety metrics are measured on recorded and simulated scenes and do not certify on-road safety.

REPRODUCIBILITY STATEMENT

Appendix H specifies the datasets, splits, sample counts, candidate generation, and metric definitions for all open-loop results; Appendix G the closed-loop protocol and metrics; Appendix F the training stages and supervision; and Appendix C the metrics of the additional analysis, including baseline configurations.

REFERENCES

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

Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving, 2020. URL https://arxiv.org/abs/1903.11027.

Minkyoung Cho, Ruben Ohana, Christian Jacobsen, Adityan Jothi, Min-Hung Chen, Z. Morley Mao, and Ethem Can. Tc-lora: Temporally modulated conditional lora for adaptive diffusion control, 2025. URL https://arxiv.org/abs/2510.09561.

Jiwan Chung, Junhyeok Kim, Siyeol Kim, Jaeyoung Lee, Min Soo Kim, and Youngjae Yu. v1: Learning to point visual tokens for multimodal grounded reasoning, 2025. URL https://arxiv.org/abs/2505.18842.

Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, Lewei Lu, Xiaosong Jia, Qiang Liu, Jifeng Dai, Yu Qiao, and Hongyang Li. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17853–17862, June 2023.

Weikai Huang, Jieyu Zhang, Sijun Li, Taoyang Jia, Jiafei Duan, Yunqian Cheng, Jaemin Cho, Matthew Wallingford, Rustin Soraki, Chris Dongjoo Kim, Shuo Liu, Donovan Clay, Taira Anderson, Winson Han, Ali Farhadi, Bharath Hariharan, Zhongzheng Ren, and Ranjay Krishna.

WildDet3D: Scaling promptable 3d detection in the wild, 2026. URL https://arxiv.org/abs/2604.08626.

Zhiyu Huang, Haochen Liu, and Chen Lv. Gameformer: Game-theoretic modeling and learning of transformer-based interactive prediction and planning for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3903–3913, October 2023.

Feiyang Jia, Lin Liu, Ziying Song, Caiyan Jia, Hangjun Ye, Xiaoshuai Hao, and Long Chen. Driveworld-VLA: Unified latent-space world modeling with vision–language–action for autonomous driving. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=sseBmS2MGa.

Qing Jiang, Junan Huo, Xingyu Chen, Yuda Xiong, Zhaoyang Zeng, Yihao Chen, Tianhe Ren, Junzhi Yu, and Lei Zhang. Detect anything via next point prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 25472–25483, June 2026.

Pengfei Jing, Victor Shea-Jay Huang, Hengtong Lu, Jifeng Dai, Yan Xie, and Benjin Zhu. Action emergence from streaming intent, 2026. URL https://arxiv.org/abs/2605.12622.

Yingyan Li, Shuyao Shang, Weisong Liu, Bing Zhan, Haochen Wang, Yuqi Wang, Yuntao Chen, Xiaoman Wang, Yasong An, Chufeng Tang, Lu Hou, Lue Fan, and Zhaoxiang Zhang. DriveVLA-w0: World models amplify data scaling law in autonomous driving. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=plrGn3RdzN.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. PETR: Position embedding transformation for multi-view 3d object detection. In *European Conference on Computer Vision (ECCV)*, pp. 531–548. Springer, 2022. doi: 10.1007/978-3-031-19812-0_31.

Yingfei Liu, Junjie Yan, Fan Jia, Shuailin Li, Aqi Gao, Tiancai Wang, and Xiangyu Zhang. Petrv2: A unified framework for 3d perception from multi-camera images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3262–3272, October 2023.

Yuechen Luo, Fang Li, Shaoqing Xu, Yang Ji, Zehan Zhang, Bing Wang, Yuannan Shen, Jianwei Cui, Long Chen, Guang Chen, Hangjun Ye, Zhi-Xin Yang, and Fuxi Wen. Last-vla: Thinking in latent spatio-temporal space for vision-language-action in autonomous driving, 2026. URL https://arxiv.org/abs/2603.01928.

NVIDIA. Physical AI Autonomous Vehicles. Hugging Face dataset card, 2025. URL https://huggingface.co/datasets/nvidia/PhysicalAI-Autonomous-Vehicles. Accessed September 2026.

NVIDIA. Alpamayo 1.5 10B. Hugging Face model card, 2026a. URL https://huggingface.co/nvidia/Alpamayo-1.5-10B. Accessed September 2026.

NVIDIA. Alpamayo 2 Super. Hugging Face model card, 2026b. URL https://huggingface.co/nvidia/Alpamayo2-Super. Accessed September 2026.

NVIDIA, Hassan Abu Alhaija, Jose Alvarez, Maciej Bala, Tiffany Cai, Tianshi Cao, Liz Cha, Joshua Chen, Mike Chen, Francesco Ferroni, Sanja Fidler, Dieter Fox, Yunhao Ge, Jinwei Gu, Ali Hassani, Michael Isaev, Pooya Jannaty, Shiyi Lan, Tobias Lasser, Huan Ling, Ming-Yu Liu, Xian Liu, Yifan Lu, Alice Luo, Qianli Ma, Hanzi Mao, Fabio Ramos, Xuanchi Ren, Tianchang Shen, Xinglong Sun, Shitao Tang, Ting-Chun Wang, Jay Wu, Jiashu Xu, Stella Xu, Kevin Xie, Yuchong Ye, Xiaodong Yang, Xiaohui Zeng, and Yu Zeng. Cosmos-Transfer1: Conditional world generation with adaptive multimodal control, 2025a. URL https://arxiv.org/abs/2503.14492.

NVIDIA, Yulong Cao, Riccardo de Lutio, Sanja Fidler, Guillermo Garcia Cobo, Zan Gojcic, Maximilian Igl, Boris Ivanovic, Peter Karkus, Janick Martinez Esturo, Marco Pavone, Aaron Smith, Ellie Tanimura, Michal Tyszkiewicz, Michael Watson, Qi Wu, and Le Zhang. Alpasim: A modular, lightweight, and data-driven research simulator for autonomous driving, October 2025b. URL https://github.com/NVlabs/alpasim.

NVIDIA, Yan Wang, Wenjie Luo, Junjie Bai, Yulong Cao, Tong Che, Ke Chen, Yuxiao Chen, Jenna Diamond, Yifan Ding, Wenhao Ding, Liang Feng, Greg Heinrich, Jack Huang, Peter Karkus, Boyi Li, Pinyi Li, Tsung-Yi Lin, Dongran Liu, Ming-Yu Liu, Langechuan Liu, Zhijian Liu, Jason Lu, Yunxiang Mao, Pavlo Molchanov, Lindsey Pavao, Zhenghao Peng, Mike Ranzinger, Ed Schmerling, Shida Shen, Yunfei Shi, Sarah Tariq, Ran Tian, Tilman Wekel, Xinshuo Weng, Tianjun Xiao, Eric Yang, Xiaodong Yang, Yurong You, Xiaohui Zeng, Wenyuan Zhang, Boris Ivanovic, and Marco Pavone. Alpamayo-R1: Bridging reasoning and action prediction for generalizable autonomous driving in the long tail. *arXiv preprint arXiv:2511.00088*, 2025c.

Zhenghao Peng, Wenhao Ding, Yurong You, Yuxiao Chen, Wenjie Luo, Thomas Tian, Yulong Cao, Apoorva Sharma, Danfei Xu, Boris Ivanovic, Boyi Li, Yan Wang, and Marco Pavone. Counterfactual VLA: Self-reflective vision-language-action model with adaptive reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4022–4031, June 2026.

Abbas Sadat, Sergio Casas, Mengye Ren, Xinyu Wu, Pranaab Dhawan, and Raquel Urtasun. Perceive, predict, and plan: Safe motion planning through interpretable semantic representations, 2020. URL https://arxiv.org/abs/2008.05930.

Abigail See, Peter J. Liu, and Christopher D. Manning. Get to the point: Summarization with pointer-generator networks, 2017. URL https://arxiv.org/abs/1704.04368.

Shuyao Shang, Bing Zhan, Yunfei Yan, Yuqi Wang, Yingyan Li, Yasong An, Xiaoman Wang, Jierui Liu, Lu Hou, Lue Fan, Zhaoxiang Zhang, and Tieniu Tan. DynVLA: Learning world dynamics for action reasoning in autonomous driving. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=UB6M4BSvwJ.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=PxTIG12RRHS.

Shuhan Tan, Kashyap Chitta, Yuxiao Chen, Ran Tian, Yurong You, Yan Wang, Wenjie Luo, Yulong Cao, Philipp Krähenbühl, Marco Pavone, and Boris Ivanovic. Latent chain-of-thought world modeling for end-to-end autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 39724–39733, June 2026.

Thomas Tian, Boyi Li, Xinshuo Weng, Yuxiao Chen, Edward Schmerling, Yue Wang, Boris Ivanovic, and Marco Pavone. Tokenize the world into object-level knowledge to address long-tail events in autonomous driving. In *Proceedings of The 8th Conference on Robot Learning (CoRL)*, volume 270 of *Proceedings of Machine Learning Research*, pp. 3656–3673. PMLR, 2025. URL https://proceedings.mlr.press/v270/tian25b.html.

Tai Wang, Xinge Zhu, Jiangmiao Pang, and Dahua Lin. Fcos3d: Fully convolutional one-stage monocular 3d object detection, 2021. URL https://arxiv.org/abs/2104.10956.

Shuang Zeng, Xinyuan Chang, Mengwei Xie, Xinran Liu, Yifan Bai, Zheng Pan, Mu Xu, and Xing Wei. FutureSightDrive: Thinking visually with spatio-temporal cot for autonomous driving. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=fCirUh6FRb.

Junkai Zhang, Yihe Deng, Kai-Wei Chang, and Wei Wang. Thinking with visual grounding, 2026. URL https://arxiv.org/abs/2606.16122.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3836–3847, October 2023a.

Renrui Zhang, Han Qiu, Tai Wang, Ziyu Guo, Ziteng Cui, Yu Qiao, Hongsheng Li, and Peng Gao. Monodetr: Depth-guided transformer for monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9155–9166, October 2023b.

Zewei Zhou, Tianhui Cai, Seth Z. Zhao, Yun Zhang, Zhiyu Huang, Bolei Zhou, and Jiaqi Ma. AutoVLA: A vision-language-action model for end-to-end autonomous driving with adaptive reasoning and reinforcement fine-tuning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=28qUA2bSe5.

Zewei Zhou, Ruining Yang, Xuewei, Qi, Yiluan Guo, Sherry X. Chen, Tao Feng, Kateryna Pistunova, Yishan Shen, Lili Su, and Jiaqi Ma. Spanvla: Efficient action bridging and learning from negative-recovery samples for vision-language-action model, 2026. URL https://arxiv.org/abs/2604.19710.

A Grounding Criteria And Transfer Gap

An *actor* is an annotated non-ego traffic participant with a GT track ID (e.g., a vehicle, pedestrian, or cyclist. We manually designate each scene's *target actor* and constraint-relevant physical fact based on driving situation. Using each model's native representation (Table 7), we assign $G = 1$ for correct grounding and $G = 0$ for an established failure; missing or ambiguous results.

Table 7: **Model-Specific Grounding Criteria.**

Model	Criterion For Correct Grounding
Alpamayo-1.5 & SpanVLA	Reasoning correctly identifies the target actor and relevant physical fact at the specified time, without contradiction.
FSDrive	The decoded future image at approximately $+0.5$ s preserves actor class, coarse location relative to ego and road, and local road geometry.
GroundAct	Entity-reference validation requires a valid generated reference to the target actor in the same-run entity representation, a correct relevant fact, and BEV IoU ≥ 0.5 with GT after temporal alignment. The above text criterion is used to validate textual reasoning.

For candidate i, $C_i = 1$ denotes ego–actor footprint overlap or contact within 0.1–5.0 s, evaluated at 10 Hz without at-fault attribution. Let $\mathcal{E}$ denote candidates with both grounding and collision labels. TransferGap is then computed as:

$$\text{TransferGap} = \frac{\sum_{i \in \mathcal{E}} G_i C_i}{\sum_{i \in \mathcal{E}} G_i}.$$

B Extended Related Work

Prediction and Interaction-Aware Planning. Future-aware driving methods use visual futures, latent dynamics, or action-conditioned world models to inform planning (Li et al., 2026; Jia et al., 2026). UniAD and GameFormer couple perception, multi-agent prediction, interaction, and planning through query-based or game-theoretic structures (Hu et al., 2023; Huang et al., 2023), while P3 exposes semantic occupancy as a differentiable planning cost (Sadat et al., 2020). These works establish that structured prediction and physical interaction can directly shape planning. GroundAct asks a complementary question: *which* physical entities should receive explicit interaction-mediated computation, and how should that choice remain connected to reasoning? We use reasoning-referenced entities as the interface to downstream future and interaction modeling, making the selected grounding explicit in the planning computation.

Reasoning-Guided Generative Planning. Flow matching and diffusion-based policies generate trajectories through iterative refinement (Lipman et al., 2023; Song et al., 2021), while guidance and residual pathways can modify a pretrained generator or controller without replacing its base behavior (Zhang et al., 2023a; NVIDIA et al., 2025a; Cho et al., 2025). Streaming Intent further connects reasoning to generative planning by deriving a discrete driving intent from chain-of-thought reasoning and using classifier-free guidance to steer a rectified-flow trajectory generator (Jing et al., 2026). GroundAct shares the goal of making reasoning consequential for generative action, but differs in the mediator. Rather than steering generation with a global semantic intent, GroundAct preserves references to individual physical entities, evaluates their interactions with the evolving ego proposal, and uses the resulting physical signal for selective residual correction.

C Evaluation Details for Additional Analysis

Perception Baselines. We fine-tune FCOS3D (Wang et al., 2021), MonoDETR (Zhang et al., 2023b), PETR (Liu et al., 2022), and PETRv2 (Liu et al., 2023) from released checkpoints on 140,554 training frames, using one current front-wide image and camera calibration. PETRv2 uses no temporal inputs. WildDet3D (Huang et al., 2026) is evaluated zero-shot. Table 8 summarizes training settings; model names link to official implementations.

Table 8: **Perception Baseline Training Settings.** Batch is global; LR and WD denote base learning rate and default weight decay. WildDet3D is evaluated without fine-tuning.

Model / Backbone	Optimizer	Batch	LR	WD
FCOS3D / R101-DCN	SGD	16	2×10^{-3}	10^{-4}
MonoDETR / R50	AdamW	16	2×10^{-4}	10^{-4}
PETR / R50-DCN	AdamW	8	2×10^{-4}	10^{-2}
PETRv2 / V-99-eSE	AdamW	8	2×10^{-4}	10^{-2}
WildDet3D	–	–	–	–

Detection And Referencing Metrics. We adapt nuScenes detection metrics (Caesar et al., 2020) and Rex-Omni pointing and overlap criteria (Jiang et al., 2026) to BEV space (Table 9). Tables 6 and 5 report percentages (higher is better), with the best value per metric and evaluation target in bold.

Table 9: **Detection And Referencing Metrics.** All scores are reported as percentages. For the two F1 metrics, $F(H) = 2H/(N_P + N_G)$, where N_P and N_G count retained predictions and GT objects, and H counts GT objects hit at least once.

Metric	Definition
mAP	Average precision using BEV center-distance matching, averaged over evaluated categories and thresholds $\{0.5, 1, 2, 4\}$ m.
NDS (BEV)	Combines mAP with BEV translation, scale, and orientation errors; velocity and attribute terms are omitted: $[5\,\mathrm{mAP} + \sum_{e\in\{\mathrm{mATE,mASE,mAOE}\}} \max(0, 1-e)]/8$, using unscaled mAP.
F1@Point	$F(H)$ with a GT hit when any predicted center falls inside its BEV footprint, irrespective of category.
F1@IoU	$F(H)$ with a GT hit when any prediction has rotated BEV IoU ≥ 0.5 with it, irrespective of category.
Critical mIoU	Mean BEV IoU for matched GT-CoC-referenced objects, excluding unmatched objects. Pairs are selected after class-agnostic, one-to-one center matching to all GT objects within 2 m.

D Safety-Critical Evaluation Set

The PAI-AV* open-loop evaluation set contains 512 PAI-AV samples selected as high-risk scenes; none overlaps the 140.5K training samples used by GroundAct, FSDrive†, AutoVLA†, and SpanVLA†. Figure 8 characterizes the set through the open-loop plans of the released Alpamayo-R1 10B checkpoint: at least one of collision, off-road, or wrong-lane occurs in 390 samples (76.17%), including 264 (51.56%) with two or three factors, while 122 (23.83%) show none. These factors describe a baseline's behavior on the set rather than a property of the scenes.

E Reference-Objective Evaluation Metrics

The reference-objective comparison separates the ability to emit usable references and complete reasoning text from the ability to identify the intended physical objects. A reference must be interpreted through the current entity pool: agreement between integer reference names alone does not establish agreement between physical referents.

Physical Correspondence. The historical annotation framework links noun phrases in a reasoning trace to reference object IDs for each clip and event time using semantic matching. The framework evaluates emitted references against the intended physical objects and counts missing objects as missed targets. The current-pool association matters because an object's scene-local entity index may change across checkpoints.

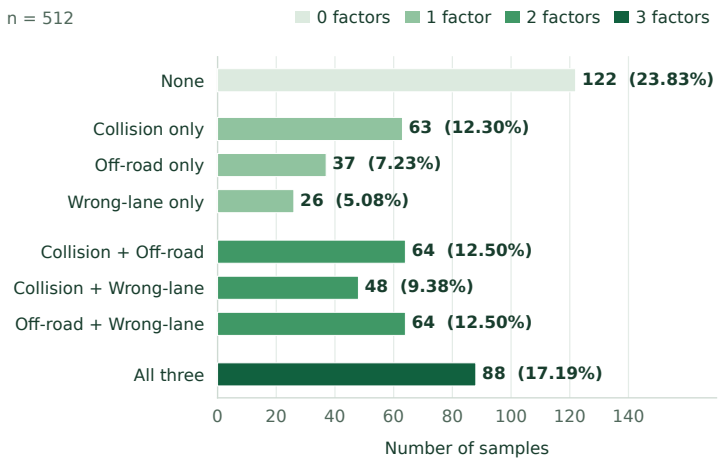


Figure 8: **Distribution of safety-critical factors in PAI-AV* ($n = 512$).** Factors are identified from the open-loop plans generated by the released Alpamayo-R1 10B model. Bars represent mutually exclusive factor combinations, with labels indicating counts and percentages and colors indicating the number of factors. "None" denotes samples with none of the listed factors.

Table 10: **Reference-Objective Evaluation Metrics.** All five reported scores are percentages. TP, FP, and FN denote matched predictions, unmatched predictions, and missed targets in physical-reference matching.

Metric	Definition
Valid Emission	Validity of the references emitted in generated reasoning.
Sentence Complete	Completion of generated reasoning, assessed separately from the correctness of its physical references.
Grounding Precision	Count-based precision: $100\,TP/(TP + FP)$, where TP and FP denote matched and unmatched predicted physical referents.
Grounding Recall	Count-based recall: $100\,TP/(TP+FN)$, where FN denotes unmatched target referents.
Grounding F1	Count-based harmonic mean: $100\,2TP/(2TP + FP + FN)$.

Calculation. For a given set of matching counts, precision measures correctness among predicted referents, recall measures recovery of target referents, and F1 combines the two. These formulas define the metrics for a given matching-count set; the aggregation across evaluation samples must use the same convention for all compared configurations.

F Training Supervision for Selective Action Correction

Act training preserves the grounded representation established by Ground–Reference and learns how the referenced physical states should shape the plan. Table 11 summarizes which modules are optimized.

Future-State Supervision. Ground-truth entity futures supervise the future provider, connecting referenced entity states to their subsequent physical motion. This prediction supplies the physical context for evaluating interactions with an ego proposal. At inference, Act uses predicted futures of the entities referenced in reasoning and computes their interactions with the evolving proposal at each solver step (Section 3.4). The temporal prediction horizon and the solver's refinement steps therefore play distinct roles.

Residual-Field Supervision. Offline alternative actions supply targets for correcting the base proposal. An alternative is accepted only when it reduces physical risk while satisfying progress, comfort, action-bound, and all-entity checks. At a stored flow state, the accepted action defines a residual-field target relative to the frozen base field evaluated at that state. Training thus ties a proposed correction to the state and base field that it is intended to modify.

Preservation and Validity. Preservation terms discourage unnecessary changes when the base proposal is already safe. The all-entity acceptance checks assess a candidate beyond the reasoning-selected subset, so a local risk reduction must also satisfy the scene-wide constraints. These checks filter correction targets and complement preservation supervision. At inference, the residual pathway uses the referenced entities; the base action expert retains the complete multimodal and reasoning context. This distinction supports selective correction within a scene-wide planning context.

Table 11: **Optimization Scope for Act.**

Component	Status	Role during Act training
Ground–Reference	Frozen	Retain the reference-to-state connection.
Base action expert	Frozen	Provide the field relative to which corrections are learned.
Future provider and residual pathway	Trainable	Learn motion prediction and selective correction.

G Closed-Loop Evaluation Metrics

For Table 2, we evaluate each model on the same $N = 913$ `public_2601` scenes in AlpaSim (NVIDIA et al., 2025b), with one rollout per scene. The valid evaluation interval extends from the start of policy control through the first collision of any type, off-road event, or deviation of at least $4\,\mathrm{m}$ from the recorded ground-truth path. All metrics are computed using AlpaSim's default protocol.

Table 12: **Closed-loop evaluation metrics.** The two distance metrics are diagnostic and have no preferred direction.

Metric	**Definition**
Progress ↑	Distance along the ground-truth path reached by projecting the final ego position onto the path, normalized by its total length.
At-fault collision ↓	Fraction of scenes with at least one front or lateral ego collision.
Off-road ↓	Fraction of scenes where the ego footprint contacts a road edge while not fully contained within the lane area.
Wrong lane ↓	Fraction of scenes where the ego heading differs by more than 120° from the direction of a lane containing the ego center.
Maximum Distance to GT Trajectory (m)	Maximum distance from the ego center to the nearest point on the GT path.
Min. obstacle distance (m)	Minimum distance between the ego footprint and any other actor's footprint.

H Experimental Setup Details

Baseline Configurations. Alpamayo-R1, Alpamayo 1.5, Alpamayo 2 Super, and LCDrive are evaluated from their authors' released checkpoints without fine-tuning; all were trained on PAI-AV, with training sets that differ in size from ours. The evaluated LCDrive uses Qwen2.5-0.5B and DINOv2-Small, with approximately 0.6B parameters. FSDrive†, AutoVLA† and SpanVLA† share GroundAct's Qwen3-VL-2B backbone: we re-implement each method's defining component on this

backbone (e.g., action tokenization) and fine-tune the resulting model on the same 140.5K PAI-AV training samples as GroundAct, without subsequent reinforcement learning.

Open-Loop Sample Selection. Normal evaluation uses the first 4,000 samples of the PAI-AV default split, and OOD evaluation uses its 290-sample OOD split. The PAI-AV* set contains 512 high-risk samples (Appendix D).

Open-Loop Metrics. Each sample is a clip and a prediction timestamp. Every model generates six candidate trajectories per sample, each with 64 poses at 10 Hz (6.4 s). Table 13 defines the four metrics. Each metric is averaged over the six candidates of a sample and then over samples. Collision and off-road are per-candidate binary events, counted once per candidate, evaluated at the discrete poses without at-fault filtering, and reported as percentages. ADE and corner distance use all samples; collision uses the annotation-available subsets.

Table 13: **Open-loop evaluation metrics.** Lower is better for all four.

Metric	Horizons	Definition	Samples
ADE (m) ↓	2.5, 5.0, 6.4 s	Mean ground-plane distance between predicted and ground-truth ego positions over the horizon.	All
Corner distance (m) ↓	6.4 s	The same distance averaged over the four corners of the oriented ego box.	All
Collision (%) ↓	2.5, 5.0 s	Fraction of candidates whose predicted ego footprint intersects any annotated actor footprint within the horizon.	3,846 / 267 / 512[a]
Off-road (%) ↓	2.5, 5.0 s	Fraction of candidates whose predicted ego footprint contains a road-boundary point within the horizon.	PAI-AV* only[b]

[a]Annotation-available subsets of Normal / OOD / PAI-AV*. [b]Normal and OOD lack map data.

Computational Setup. Training uses 32 NVIDIA A100-SXM4-80GB GPUs throughout all stages. For each evaluated sample, open-loop evaluation runs on a single NVIDIA A100-SXM4-80GB GPU and closed-loop evaluation on a single NVIDIA H100 80GB node.

I Inference Efficiency.

As shown in Table 14, GroundAct adds 0.22 s to the backbone VLM's mean inference latency (0.88 s), bringing the total to 1.10 s on a single NVIDIA A100-SXM4-80GB GPU without model- or inference-level optimization. Potential avenues for reducing latency include compiled inference and TensorRT deployment. Beyond latency reduction, another deployment direction is to decouple high-level planning from low-level control. GroundAct's strong open-loop planning performance over a 6.4 s prediction horizon motivates exploring lower-frequency grounded planning (e.g., approximately 1 Hz) paired with a faster low-level controller.

Table 14: **E2E Latency.**

Config.	Ground	Ref.	Act	Time (s)
Driving VLM backbone	–	–	–	0.88 ± 0.14
Spatial grounding	✓	–	–	0.94 ± 0.12
Grounded reasoning	✓	✓	–	0.99 ± 0.14
GroundAct	✓	✓	✓	$\mathbf{1.10 \pm 0.19}$