

LEAD: Minimizing Learner–Expert Asymmetry in End-to-End Driving

Long Nguyen^{1,3} Micha Fauth¹ Bernhard Jaeger^{1,3} Daniel Dauner^{1,3}
 Maximilian Igl² Andreas Geiger^{1,3} Kashyap Chitta²

¹University of Tübingen, Tübingen AI Center ²NVIDIA Research ³KE:SAI

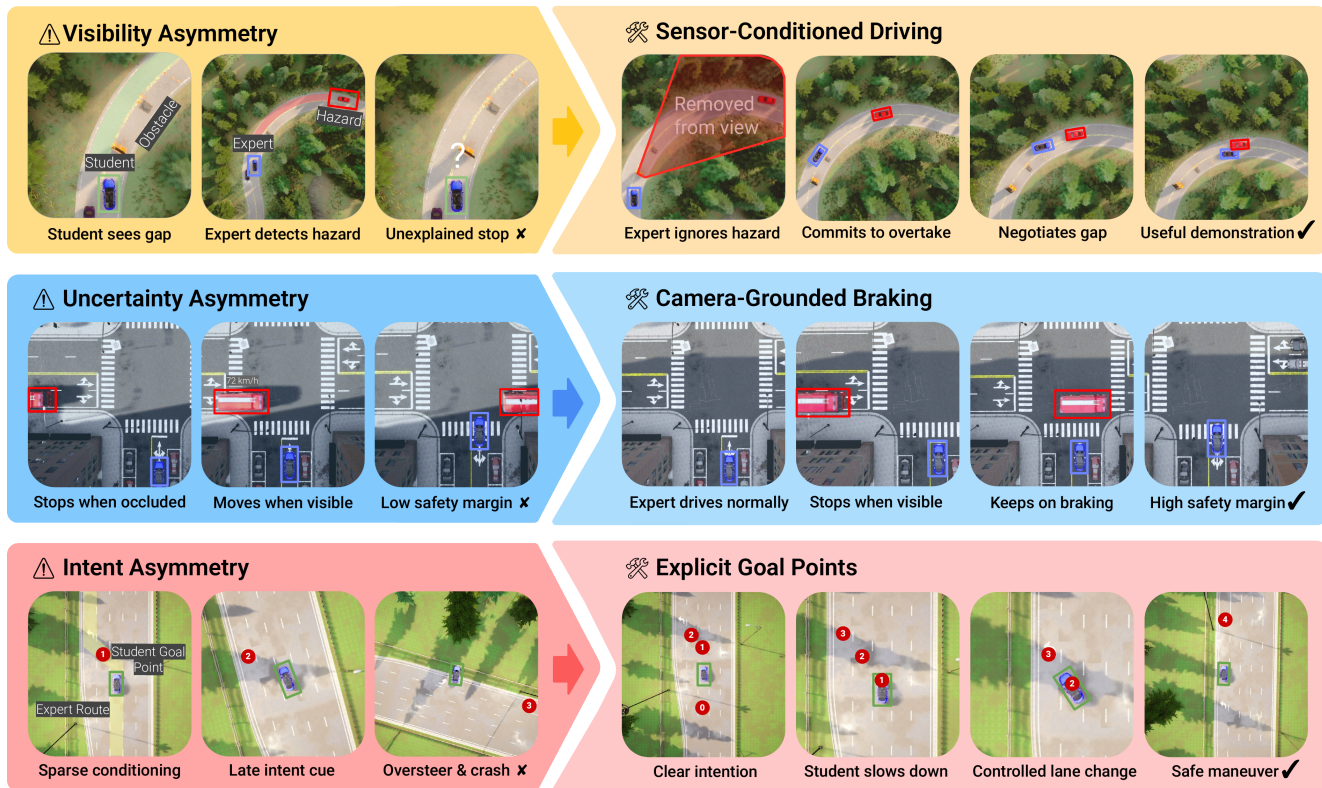


Figure 1. **Performing a task well and teaching it well are not the same.** An expert driver (blue bounding box) is most useful when its behavior can be transferred to a student policy (green bounding box) effectively. Current expert drivers for CARLA do not fulfill this requirement. We focus on three common asymmetries that hinder effective transfer. *Visibility asymmetry*: the expert reacts to occluded actors, leading to non-causal and less useful demonstrations. *Uncertainty asymmetry*: the expert’s noiseless state inputs (e.g., accelerations and velocities of other vehicles) lead to successful but dangerous demonstrations. *Intent asymmetry*: the student’s intent is under-specified (as a single target point) making it unaware of complex multi-lane maneuvers. Our approach reduces expert privileges, enforces sensor-aware demonstrations and redesigns the policy’s navigation conditioning, resulting in state-of-the-art closed-loop driving in CARLA.

Abstract

Simulators can generate virtually unlimited driving data, yet imitation learning policies in simulation still struggle to achieve robust closed-loop performance. Motivated by this gap, we empirically study how misalignment between privileged expert demonstrations and sensor-based student observations can limit the effectiveness of imitation learn-

ing. More precisely, experts have significantly higher visibility (e.g., ignoring occlusions) and far lower uncertainty (e.g., knowing other vehicles’ actions), making them difficult to imitate reliably. Furthermore, navigational intent (i.e., the route to follow) is under-specified in student models at test time via only a single target point. We demonstrate that these asymmetries can measurably limit driving performance in CARLA and offer practical interventions to

address them. After careful modifications to narrow the gaps between expert and student, our TransFuser v6 (TFv6) student policy achieves a new state of the art on all major publicly available CARLA closed-loop benchmarks, reaching 95 DS on Bench2Drive and more than doubling prior performances on Longest6 v2 and Town13. Additionally, by integrating perception supervision from our dataset into a shared sim-to-real pipeline, we show consistent gains on the NAVSIM and Waymo Vision-Based End-to-End driving benchmarks. Our code, data, and models are publicly available at <https://github.com/kesai-labs/lead>.

1. Introduction

Closed-loop evaluation is essential in robotics, in particular autonomous driving, because strong open-loop prediction does not necessarily translate into robust closed-loop control [13, 18]. While real-world closed-loop testing remains essential, its cost motivates the widespread use of simulators such as CARLA for benchmarking driving policies [19]. Within this setting, a two-stage imitation learning paradigm, often called Learning by Cheating (LBC) [6], has proven highly effective. First, a privileged expert policy is constructed using ground-truth state information (e.g., the precise map layout), allowing it to plan without perception errors. The second stage then trains a student policy to reproduce the expert’s actions from sensory observations (e.g., cameras). A rich body of recent literature supports this methodology [27, 45, 50, 51].

In initial work following this paradigm, constructing a reliable expert was itself a major challenge, and empirical progress was closely tied to expert quality: early improvements in expert behavior translated directly into better student performance [10, 23]. As a result, expert design focused primarily on maximizing performance, driven by the expectation that stronger experts would yield stronger students. Once expert policies achieved strong benchmark performance, most work treated them as fixed sources of supervision, prioritizing model improvements over further expert redesign [27, 34]. This approach was successful when models were the bottleneck, but the landscape has shifted. With student policies now plateauing well below expert performance [1], the expert itself deserves renewed attention.

Our work studies, in detail, the generalization gap between expert and student policies. We identify and analyze several sources of misalignment between expert and student that systematically hinder effective imitation in CARLA, as illustrated in Fig. 1. This motivates a new expert (and corresponding dataset), **LEAD**, explicitly designed to reduce Learner-Expert Asymmetry in Driving.

In parallel to expert misalignment, a second limitation arises from how student policies are conditioned on navigation intent. Prior work observes a pronounced *target point*

bias, where policies over-emphasize the immediate goal location and under-utilize driving context [25]. This behavior is commonly attributed to weak scene representations [25]. Our work shows that target point bias persists even when scene representations are made strong. Instead, we attribute it to two additional factors. First, driving intent is often insufficiently specified: a single target point does not provide information to disambiguate multi-step maneuvers such as lane changes, making direct goal pursuit a convenient shortcut. Second, the geometric target point coordinates are frequently injected late within the decoder of the driving policy, preventing meaningful interaction with perception features from the encoder and increasing the target point’s influence relative to them. Together, these two factors lead to the most dominant failure modes that negatively impact closed-loop performance of our baseline policy.

Building on these insights, we introduce TransFuser v6 (TFv6), which reduces these asymmetries to achieve substantially improved closed-loop driving. When trained on a large, diverse dataset generated by the LEAD expert, TFv6 achieves a new state of the art on Bench2Drive with 95 DS [27], an improvement of 8 DS compared to the previously established best performance. This is significant progress given the gap between the best-performing and fifth-best-performing published methods is approximately 2 DS [1]. Finally, we aggregate the LEAD dataset with real-world datasets into a unified training pipeline, where pre-training on LEAD yields consistent improvements on the NAVSIM and Waymo End-to-End Driving benchmarks [18, 59]. In summary, our contributions span:

1. **Reducing asymmetries.** We systematically analyze and reduce learner-expert asymmetries in CARLA by aligning expert supervision with the student’s observable state and strengthening the student’s navigation intent, improving the effectiveness of imitation-based driving.
2. **Mitigating target point bias.** We show that the target point bias persists with existing mitigation strategies and identify insufficient as well as poorly integrated intent conditioning as its key contributing factors.
3. **A complete training stack.** We release LEAD, a large-scale CARLA dataset and training pipeline enabling state-of-the-art closed-loop driving performance and measurable sim-to-real gains on NAVSIM and Waymo.

2. Related Work

End-to-End Driving via Imitation Learning: Imitation learning (IL) trains policies to map sensor observations to actions by imitating expert behavior [3, 42]. Following rapid progress on public benchmarks [5–7, 9, 10, 14, 15, 25, 43, 48, 49, 57], IL has become the standard approach for end-to-end driving [8, 28, 37, 38, 45, 52, 64]. While some work relies on human driving data and open-loop evalua-

tion on static datasets [4, 18, 22, 59], open-loop metrics are known to be unreliable for driving, which is inherently a closed-loop task [13, 36, 56, 61]. Closed-loop evaluation is commonly carried out in interactive simulators that allow policies to act, observe, and recover from outcomes of their own decisions [19, 33, 39, 47, 58, 60, 63]. We expand on this line of work in the CARLA simulator [19].

Learning by Cheating: Human collection of driving data is costly in simulation, so privileged experts that leverage ground-truth simulator states to generate supervision for an IL student are used ubiquitously—an approach also often called Learning by Cheating [6]. Such experts are either rule-based [2, 14, 17, 23, 51] or reinforcement learning-based [16, 26, 34, 62]. In this work, we build upon and propose several improvements to the rule-based open-source expert PDM-Lite [51], which enables controlled analysis of expert–student alignment. Rather than optimizing expert performance, we focus on making the expert’s demonstrations easier to follow for student models, providing downstream benefits.

State Asymmetry: Student–expert state asymmetry is a long-standing challenge in robotics [40, 53–55]. An expert policy that is optimal under full state observability can become suboptimal, even unsafe, when imitated by a student operating under partial observability. Adapting the expert and student jointly is a popular approach to mitigate this mismatch [40, 54]. While effective in low-dimensional settings, this typically relies on differentiable experts and expensive online interaction. In large-scale driving simulation, such assumptions are often impractical. Effective expert policies are often rule-based and non-differentiable.

In our work, we distinguish two practical sources of state asymmetry (Fig. 1). *Visibility asymmetry* arises when the expert conditions its actions on parts of the scene that fall outside the student’s sensor coverage. *Uncertainty asymmetry* stems from differences in state fidelity: experts operate on noise-free quantities (e.g., velocities and accelerations), enabling confident planning with minimal safety margins, while the student must act under substantially higher uncertainty. Prior work has shown that both visibility and uncertainty asymmetry can lead to expert demonstrations that are unreliable or unsafe when imitated from raw sensory inputs [64]. Our work further carefully constrains the expert’s inputs and decision logic to reduce reliance on information that is unobservable or unrealistically precise for the student, improving alignment between expert supervision and sensor-based imitation.

Intent Asymmetry: We refer to *intent asymmetry* as the mismatch between the information provided to the expert and the student for long-horizon decision making: the former acts on a dense route, while the latter must reason about direction from sparse target points, as depicted in Fig. 1. A

natural response is to increase the density of navigation conditioning. However, prior attempts that explicitly increased the density of navigation signals have shown limited success [64]. Our work shows that increasing intent density alone is insufficient; the manner and location of injecting navigation intent into the policy are equally important.

Target Point Bias: Beyond their original role as high-level *navigational guidance*, target points are often implicitly used by E2E driving models as *corrective signals* to recover from covariate shift, steering the vehicle back toward the route after deviations. This phenomenon is referred to as the target point bias [25]. We show that this behavior is closely tied to intent asymmetry: when the navigation intent is under-specified and the driving trajectory correlates highly with the target point, a policy naturally learns the shortcut of adjusting the trajectory steering towards the goal. By reducing intent asymmetry and restructuring the method by which navigation information is injected into the policy, our approach mitigates this effect, weakening the reliance on target points as implicit corrective signals while maintaining their role as navigational conditioning.

3. Minimizing Learner-Expert Asymmetry

In this section, we study the impact of various learner-expert asymmetries in a controlled manner. Using a fixed, strong baseline, we systematically reduce learner–expert mismatch by aligning the information used for expert driving and navigation with what is available to the student policy. Holding model architecture and dataset scale constant allows us to attribute performance differences to this alignment.

3.1. Preliminaries

We consider the task of navigating through urban scenarios along a predefined route. Each route is represented by a sparse sequence of GNSS coordinates, referred to as target points, which provide coarse navigational guidance.

Benchmark: We conduct evaluation in CARLA 0.9.15 using the Longest6 v2 and Bench2Drive (B2D) benchmarks [26, 27]. Longest6 v2 spans 36 routes of approximately 2 km each across six towns, while B2D spans 220 routes of approximately 150 m on all twelve CARLA towns.

Metrics: We use the official CARLA Leaderboard 2.0 evaluation protocol. The primary metric is the Driving Score (DS), which jointly reflects task completion and safety. DS is defined as the product of Route Completion (RC), the fraction of the route completed, and the Infraction Score (IS), a multiplicative penalty factor initialized at 1.0 and decayed with each infraction. All reported results are averaged over three independently trained models with different training seeds. Each model is evaluated once.

Baseline: Our approach builds on *TransFuser++* [25, 64], the current state of the art on Longest6 v2. To avoid

confusion due to the large number of TransFuser variants, we adopt the versioning nomenclature of [24] and refer to TransFuser++ as **TFv5**. TFv5 is an end-to-end (E2E) policy, trained via imitation learning, that produces Bird’s Eye View scene tokens from camera and LiDAR inputs through self-attention-based fusion [10, 43]. For lateral control, a set of learned route queries attend to these scene tokens. After a sequence of self- and cross-attention, the queries are further refined by a shallow and low-dimensional Gated Recurrent Unit (GRU) [12], whose hidden state is initialized with the GNSS target point. For longitudinal control, an additional learned query predicts target speed [25].

3.2. State Alignment

TFv5 is supervised by the privileged rule-based expert *PDM-Lite* [51]. PDM-Lite computes a dense route using A* search over the lane graph and refines this trajectory to avoid static obstacles. Potential collisions are forecast using ground-truth 3D bounding boxes and precise dynamic state information, which directly triggers braking behavior. While this design is simple and highly effective, it inherently relies on highly privileged information. As a result, expert actions may depend on non-observable actors (*visibility asymmetry*) or on motion estimates with unrealistically high precision (*uncertainty asymmetry*). While these issues are widely recognized, addressing them in practice remains challenging. To this end, we now provide an overview of how we adapt PDM-Lite to better reflect the information available to the student policy. Further implementation details are provided in supplementary material.

Reducing Visibility Asymmetry: To prevent expert decisions based on unobservable information, we constrain the expert’s planning inputs to signals accessible through the student’s sensors.

1. For dynamic actors, we exclude those outside the student’s camera view, such as pedestrians behind the vehicle, while accounting for actor extent and environmental conditions including weather and time of day. This avoids expert reactions to unobservable hazards.
2. We apply similar constraints to traffic infrastructure. The stopping logic for traffic lights only considers lights within the camera frustum. For traffic signs, explicit speed limit information is not provided to the model and signs are only intermittently visible. We therefore cap the expert’s target speed to the minimum of the posted limit and the typical flow of nearby vehicles, both of which are inferable from local traffic context.

Reducing Uncertainty Asymmetry: To account for uncertainty in student perception, we adjust the expert’s braking behavior.

1. Beyond reacting to predicted collision courses, the expert now brakes in the presence of nearby observable

hazards, reducing reliance on precise velocity or acceleration estimates that the student cannot reliably infer.

2. We further adapt expert behavior based on expected perception reliability. Under low-visibility conditions such as nighttime or heavy rain, we reduce the expert’s driving speed to reflect decreased perceptual confidence.
3. During unprotected turns at junctions, we enlarge the bounding boxes of oncoming actors for collision checks, encouraging safety decisions that rely on conservative spatial margins rather than precise motion prediction.

Using this state-aligned expert LEAD, we collect data with identical spatial coverage and scale to the original dataset. As shown in Table 1, training TFv5 on LEAD improves the Driving Score on Longest6 v2 by +11 points and on Bench2Drive by +1.37 points, while the expert’s own performance remains the same (Table 5).

TFv5 trained with...	Longest6 v2 DS ↑	B2D DS ↑
PDM-Lite dataset [51]	22.51 ± 4.42	83.56 ± 0.32
LEAD dataset (Ours)	34.05 ± 1.50	84.94 ± 0.50

Table 1. Effect of State Alignment.

3.3. Intent Alignment

After improving the quality of driving demonstrations, we observe two failure modes remain particularly prominent:

1. When the active target point is distant or placed in an unusual position relative to the ego vehicle, such as behind it in a roundabout, trajectory prediction breaks down entirely, producing disjoint and unusable planning outputs.
2. When the target lies on an adjacent lane, the policy becomes goal-fixated and steers aggressively toward it, ignoring static and dynamic hazards in the surrounding traffic context. The same behavior can be observed when the target point is inside static obstacles.

Notably, these failures occur even under clear weather, open road layouts, and low traffic density, where perception and expert supervision are unambiguous. This points us to limitations in the model rather than the expert, specifically, under-specified intent conditioning and the target point bias. We address these via architectural changes, and call the resulting model TFv6. Following [20, 45], we remove the Gated-Recurrent-Unit-based (GRU-based) refinement stage after the planning queries attend to BEV tokens and represent target point as an explicit token alongside the BEV tokens. Before embedding, the target point is normalized to $[-1, 1]$ using the training dataset statistics. Table 2 summarizes the effects of this change, showing improvements of +6 DS on Longest6 v2 and +2 DS on Bench2Drive. The improvement is driven by a marked reduction of the first failure mode, in which the planner previously failed to produce a coherent trajectory on several Longest6 v2 routes when target points were distant or unusually positioned.

Furthermore, we observe generally that the policy is less sensitive to the exact location of the target point.

Path planning...	Longest6 v2 DS \uparrow	B2D DS \uparrow
With GRU	34.05 \pm 1.50	84.94 \pm 0.50
Without GRU	40.70 \pm 2.86	87.26 \pm 0.47

Table 2. Effect of Removing GRU.

For the second failure mode, beyond poorly integrated target points, we hypothesize that the navigation signal itself is too sparse. To increase the density of navigation conditioning, we replace single-target conditioning with a compact three-point route representation consisting of the previous, current, and future targets. We further reduce the distance threshold at which the current target point is switched out, causing future targets to become relevant earlier and provide stronger supervision to the policy during training. For example, when the current target point is only 2-3 meters away, it provides insufficient information about the long-term path to be taken, for which the policy can now rely on the future target point. Together, these changes substantially improve closed-loop performance, as shown in Table 3. Implementation details and qualitative visualizations are provided in the supplementary material.

TFv6 trained with...	Longest6 v2 DS \uparrow	B2D DS \uparrow
With one TP	40.70 \pm 2.86	87.26 \pm 0.47
With three TPs	42.13 \pm 0.75	89.29 \pm 0.45

Table 3. Effect of Multiple Target Points (TPs).

3.4. Discussion

Fig. 2 summarizes how state and intent alignment contribute to infraction counts. While infractions in general decrease with each improvement, the weakened target point bias, achieved through intent alignment, leads to an increase in route deviation, since the model no longer aggressively snaps back toward the target points after getting off route.

Late Goal Conditioning as a Bottleneck: Although the GRU was originally introduced to model temporal dependencies, its role in modern policy architectures is limited. The planning queries already integrate spatial and temporal context through stacked self- and cross-attention, making the GRU largely redundant. Inserted after a substantially more expressive transformer decoder (six layers at 256 dimensions), the GRU forms a shallow bottleneck with a single recurrent layer and 64 hidden units. Rather than facilitating richer interactions between scene context and the target point, this bottleneck constrains the representation and defaults to reinforcing the influence of the stronger available signal, namely the target point.

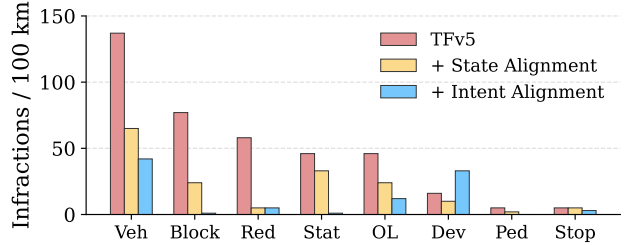


Figure 2. Effect of Aligned Supervision and Conditioning. Infractions counted per 100km, lower is better. *Stat*: Collisions with Layout; *Ped*: Collisions with Pedestrian; *Veh*: Collision Vehicle; *OL*: Outside Lane; *Red*: Red Light; *Dev*: Route Deviation; *SI*: Stop Infraction; *Block*: Vehicle Blocked.

Furthermore, the GRU conditions only the planned path and steering on the target point, while target speed prediction is handled independently of the target point. This design becomes problematic under distribution shift at deployment. During training, the policy is exposed exclusively to on-route data, where steering and speed remain implicitly consistent under nominal route-following behavior. Once the vehicle deviates from the route, the GRU-induced decoupling leads to errors: steering is strongly biased toward the target point, while speed prediction is no longer calibrated with respect to the resulting steering behavior.

4. Experiments

Having established in Section 3 that aligned supervision and conditioning significantly improve closed-loop performance, we now train TFv6 on an expanded version of LEAD that is not only larger, but also more diverse in terms of towns, lighting, weather conditions, sensor configurations, and scenario coverage. We evaluate across five benchmarks spanning both simulation and real-world driving. Our main experiments focus on long-horizon closed-loop evaluation in CARLA, where we also analyze sensor configurations and show that combining cameras with LiDAR and radar yields the best performance. Finally, we demonstrate that synthetic pre-training on LEAD consistently improves performance on multiple real-world open-loop benchmarks, indicating that synthetic data can provide transferable benefits beyond simulation.

4.1. Closed-Loop Benchmarks

Building on the controlled ablations in Section 3, we now evaluate TFv6 on long-horizon closed-loop benchmarks. In addition to Longest6 v2 and B2D, where the test towns are seen during training, we further consider the challenging Town13 validation benchmark, which consists of 20 routes on an unseen town that are on average 12.39 km long, featuring roughly 100 scenarios per route of 38 different types [64]. Unlike the prior two benchmarks, methods are not

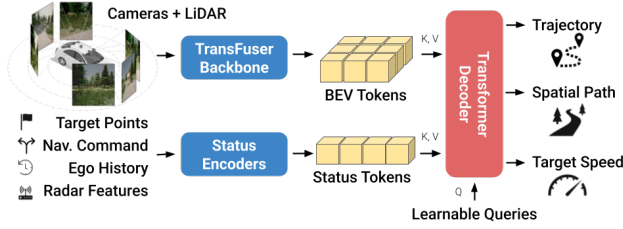


Figure 3. **TransFuser v6**. The model encodes dense sensor data and sparse status inputs (including multiple target points and per-object radar features) into a unified token space. A transformer decoder then produces a driving plan via learnable queries, removing the GRU bottleneck present in prior TransFuser versions.

allowed to train on data collected in that town, therefore testing the ability of methods to generalize to novel environments (akin to “level 5” autonomy). Town13 validation is among the most challenging driving benchmarks.

Metrics: We adopt the standard CARLA Leaderboard 2.0 metrics from Section 3.1. On long routes, however, the Driving Score (DS) can be counterintuitive: because the Infraction Score (IS) decays exponentially with each violation, agents are sometimes rewarded for stopping early rather than driving further [64]. We therefore report the Normalized Driving Score (NDS) on Town13, defined as $RC \times I$, where I is a distance-normalized version of IS [64]. For Bench2Drive, we additionally report Success Rate (SR), the fraction of infraction-free completions. Similar to Section 3, reported results are averaged over three independent random seeds.

Training: We train TFv6 using a dataset collected with the LEAD expert. The dataset is larger than the one used in Section 3, containing 73 hours of driving instead of 40 hours. We train TFv6 with 4 L40S GPUs for roughly one week in mixed-precision [41]. Further training and data curation details can be found in the supplementary material.

Radar Features: Radar is a common sensor modality in autonomous driving but rarely included in current CARLA benchmarks. To further reduce the learner-expert asymmetry, LEAD provides the student policy with four radar units (each with 75 detections per frame). As is common in industrial-grade radars, we pre-process the raw radar detections with a light-weight learned module to provide meaningful object-level features for the downstream policy. The pre-processing details are provided in the supplementary material. As shown in Fig. 3, these object-level radar features bypass the sensor fusion encoder and are input directly into the planning decoder as additional context tokens alongside dense BEV and status information.

Baselines: In addition to TFv5 (Section 3.1), we include several competitive baselines in our evaluation. *SimLingo* [45] is a recent vision–language–action model that

Method	RC \uparrow	IS \uparrow	DS \uparrow	I \uparrow	NDS \uparrow
‘Town13 Val’ – Town13 withheld during training					
TFv5 [64]	50.20	0.10	1.08	0.04	2.12
TFv6 (Ours)	39.70	0.30	3.52	0.22	4.04
<i>PDM-Lite</i> [51]	83.40	0.41	36.30	0.63	58.50
‘Town13 Train’ – Trained on all towns					
UniAD [22]	1.42	0.49	0.23	0.30	0.00
TFv5 [64]	68.53	0.04	0.96	0.07	4.94
TFv6 (Ours)	71.82	0.12	5.28	0.20	14.65

Table 4. **Benchmarking on CARLA Town13**. Results are averaged over three trained models with different training seeds. *Results included for completeness, though this setting is not the recommended default for this benchmark.

won the CARLA Leaderboard Challenge 2024, which also builds on PDM-Lite. *HiP-AD* [52] is a camera-only end-to-end method and the current published state of the art on B2D. Note that HiP-AD was trained on a different dataset, which is optimized for B2D. Finally, *UniAD* [22] is a widely used end-to-end approach that employs sequential auxiliary tasks for perception and planning.

Town13 Results: Table 4 shows the results on the CARLA Town13 benchmark. Note that we use the best version of our model here, including the radar and LiDAR input modalities, and a RegNet backbone [44] (with detailed ablations provided later in Table 5). To highlight the impact of generalization, we also evaluate TFv6 trained with data from Town13 (Town13 Train), but shade the results in gray to make it clear that these are only for analysis.

TFv6 outperforms TFv5 in terms of both DS and NDS, and in both the ‘Val’ and ‘Train’ settings, by large margins. Interestingly, there is still a significant generalization gap, where TFv6 achieves 14.65 NDS on Town13 Train which drops to 4.04 NDS on Town13 Validation. This emphasizes the importance of validation benchmarks where methods do not train on any data collected in the validation town. Qualitatively, TFv6 drives more safely than TFv5 as indicated by its higher IS and I metrics.

Bench2Drive Results: In Table 5, we see that scaling the data, backbone, and sensor modalities brings TFv6 within 2 DS of LEAD on B2D, but a 10-point SR gap remains. This discrepancy reflects how DS treats short routes: an infraction near the end has little impact when most progress is already secured. SR, being all-or-nothing, captures errors that DS discounts. This gap indicates the policy is less robust than DS alone suggests.

Longest6 v2 Results: On Longest6 v2, the improvement over TFv5 and SimLingo, both winners of CARLA Challenge 2024, is even more pronounced, with gains of

Method	Backbone	Input Modalities			Bench2Drive		Longest6 v2	
		Camera FOV	LiDAR	Radar	Driving Score \uparrow	Success Rate \uparrow	Driving Score \uparrow	Route Completion \uparrow
HiP-AD [52]	ResNet-50	360°	\times	\times	86.8	69.1	7	56
SimLingo [45]	InternViT-300M	110°	\times	\times	85.1	67.2	22	70
TFv5 [64]	RegNetY-032	110°	\checkmark	\times	83.5 \pm 0.3	67.3 \pm 1.0	23 \pm 4	70 \pm 8
TFv6 (Ours)	ResNet-34	360°	\times	\times	91.6 \pm 0.7	79.5 \pm 2.0	43 \pm 1	85 \pm 3
		360°	\checkmark	\times	94.7 \pm 0.6	85.6 \pm 0.0	52 \pm 7	88 \pm 5
		360°	\times	\checkmark	94.2 \pm 0.7	85.3 \pm 0.9	52 \pm 1	88 \pm 2
		360°	\checkmark	\checkmark	95.0 \pm 0.7	84.3 \pm 2.1	54 \pm 5	89 \pm 3
		140°	\checkmark	\checkmark	94.7 \pm 0.7	82.1 \pm 3.6	57 \pm 3	99 \pm 0
	RegNetY-032	140°	\checkmark	\checkmark	95.2 \pm 0.3	86.8 \pm 0.7	62 \pm 1	91 \pm 1
<i>PDM-Lite</i> [51]	-	-	-	97.0	92.3	73	100	
<i>LEAD</i> (Ours)	-	-	-	96.8	96.6	73	93	

Table 5. **CARLA Bench2Drive and Longest6 v2.** TFv6 outperforms all baselines. Our best model uses 140° FOV camera, LiDAR and Radar. Results are averaged over three independently trained models with different training seeds.

+39 DS and +21 RC. While HiP-AD performs competitively on short-route benchmarks, its substantially lower performance on Longest6 v2 illustrates the challenges of long-horizon evaluation. This highlights the importance of evaluating driving policies on extended routes, where the likelihood of encountering rare and challenging scenarios increases with route length, exposing weaknesses that short-route benchmarks may miss. CARLA is currently the only widely used simulator that supports long-form evaluation. Most recent benchmarks and simulation frameworks [4, 18, 21, 30, 31, 39, 59, 63] are limited to short-form driving due to log-replay or reconstruction-based designs that inherently prevent long-horizon testing. Generative approaches [11, 46] may offer a path forward in addressing this limitation.

Ablation: Table 5 also shows several ablations on different sensor setups with TFv6. We find that using a wide front camera-setup with 140° FOV with a combination of LiDAR and Radar sensors yields the best results. Additionally, we show that the RegNetY-032 backbone provides a notable performance improvement compared to the smaller ResNet-34 backbone, consistent with the findings of [10]. Finally, despite not being optimized for driving performance, LEAD matches the performance of PDM-Lite on Bench2Drive and Longest6 v2, indicating that improved learner-expert alignment does not require sacrificing expert competence.

4.2. Real-World Data Benchmarks

To complement our closed-loop CARLA evaluation, we further assess transfer to multiple open-loop real-world benchmarks for end-to-end driving.

NAVSIM v1 [18] evaluates 4-second trajectory rollout using multi-view camera inputs, historical states, and discrete commands over a 1.5-second history. Performance is measured using the Predictive Driver Model Score (PDMS), which aggregates collision avoidance, progress, time-to-

collision, driving-area compliance and comfort.

NAVSIM v2 [4] extends v1 with a two-stage pseudo-simulation evaluation pipeline designed to better approximate closed-loop behavior. In the first stage, trajectories are evaluated from real-world observations using an extended version of PDMS (EPDMS), incorporating additional rule-based and sub-metrics. The second stage evaluates the same planner on pre-rendered synthetic observations generated via 3D Gaussian Splatting [32, 35]. Stage 2 scores are aggregated using a proximity-based weighting scheme that prioritizes synthetic start states close to the planner’s Stage 1 predicted endpoint, yielding a final score that reflects robustness to small deviations and error recovery.

WOD-E2E [59] consists of driving scenes that over-sample rare, safety-critical events (< 0.03% of daily driving). It evaluates 5-second predicted trajectories using the Rater Feedback Score (RFS), a human-annotated metric designed to capture multi-modal, long-tail driving behavior by assigning full expert credit within predefined trust regions and exponentially decaying the score otherwise.

Training: We collect CARLA data with the camera parameters of target benchmarks, specifically sampled to emphasize dense agent interactions and matching lighting conditions. We only use the perception labels from the synthetic CARLA data to avoid the driving-style mismatch between human data collectors and LEAD. For the NAVSIM benchmarks, we use the full *navtrain* split and a subset of 100k frames from CARLA for training. We train on mixed data in the first 30 epochs, which smoothly excludes the synthetic data, followed by 90 epochs exclusively training on *navtrain*. For WOD-E2E, we subsample 300k frames uniformly for each epoch. Here, we pre-train for 30 epochs exclusively on CARLA data, followed by 30 epochs of fine-tuning on the WOD-E2E training split. We provide further information about hyper-parameters and data filtering in our supplementary material.

Method	NAVSIM v1 \uparrow	NAVSIM v2 \uparrow	WOD-E2E \uparrow
<i>Ego MLP</i> [18]	65.6	12.7	7.31
LTF [10]	83.8	23.1	-
LTFv6	85.4	28.3	7.51
+ LEAD	86.4	31.4	7.76
<i>Expert</i>	94.5	51.3	8.10

Table 6. **LTFv6 on Real-World Data.** We report PDMS on the `navtest` split of NAVSIM v1, EPDMS on the `navhard` split of NAVSIM v2, and RFS on the validation split of WOD-E2E. Across all benchmarks, our LTFv6 model and the LEAD training strategy achieve consistent performance gains. Results are averaged over three trained models with different training seeds.

Since LiDAR and radar data are not available in all the benchmarks, we drop these modalities and replace the LiDAR with a positional encoding, as done in Latent TransFuser (LTF) [10]. We call the resulting method **LTFv6** to indicate that this version matches the TFv6 architecture.

Results: As shown in Table 6, while we achieve modest improvements in absolute terms, they are consistent across benchmarks and training setups. On NAVSIM, LTFv6 improves in the ego progress, drivable area compliance, and traffic light compliance sub-metrics with minor trade-offs in comfort. Our proposed joint pre-training further increases the LTFv6 score across all benchmarks, demonstrating the value of synthetic data despite the presence of distribution shifts. Note that ego status can be interpreted as a lower bound while the expert driver upper bounds differ across benchmarks: NAVSIM v1 and WOD-E2E report human driving performance, whereas NAVSIM v2 uses a privileged planner as the upper bound [4]. Further comparisons are provided in the supplementary material.

5. Conclusion

We study how misalignment between privileged experts and student policies affects closed-loop driving. By reducing visibility, uncertainty, and intent asymmetries, we introduce LEAD, a student-centric expert and dataset designed to produce more transferable supervision. This alignment yields substantial closed-loop performance gains even without architectural changes, underscoring expert design as a practical lever in simulation-based imitation learning.

Beyond supervision, our experiments show that policy behavior is further constrained by how navigation intent is specified and integrated. Under-specified and late goal-point conditioning amplifies target point bias and promotes shortcut learning, limiting closed-loop robustness in CARLA. Restructuring intent conditioning to be denser and injected earlier enables a more balanced interaction between scene understanding and goal following, leading to more stable behavior over long horizons.

Combining aligned supervision with improved intent

conditioning, our new model TFv6 achieves state-of-the-art performance across both short- and long-horizon CARLA benchmarks, with particularly strong gains on challenging long-horizon evaluations where errors accumulate over time. We also provide preliminary evidence of the benefits of co-training on data from simulation and the real world for benchmarks such as NAVSIM and Waymo. Together, these results suggest that continued progress in simulation-based end-to-end driving will benefit from treating expert design and goal specification as integral, co-evolving components of the learning system. As end-to-end driving performance in simulation improves, progress increasingly depends on factors beyond model capacity and data scale.

5.1. Limitations

We conclude by discussing several limitations of our study and directions for future work.

Remaining Gap to Expert: Despite narrowing the learner-expert gap, a meaningful difference remains (Table 5). We attribute this largely to inherent limitations of behavior cloning, such as compounding errors and the inability to recover from off-route deviations. Closed-loop training via DAGger or reinforcement learning [29] is a promising direction to further close this gap.

Sim-to-Real Transfer: Our work investigates perception co-training but does not address planning co-training, where policy behaviors are trained by combining supervision from human driving trajectories and LEAD. Moreover, sim-to-real evaluation is currently limited to open-loop and pseudo-closed-loop benchmarks; demonstrating closed-loop real-world benefits remains future work.

Expert Design Scope: Our study is restricted to a rule-based expert in simulation, and the proposed alignment requires domain-specific tuning. Whether similar principles apply to learned experts, human demonstrations, or real-world driving remains open.

Acknowledgments

Bernhard Jaeger and Andreas Geiger were supported by the ERC Starting Grant LEGO-3D (850533) and the DFG EXC number 2064/1 - project number 390727645. Daniel Dauner was supported by the German Federal Ministry for Economic Affairs and Energy within the project NEXT GEN AI METHODS (19A23014S). We thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting Bernhard Jaeger and Daniel Dauner. This research used compute resources at the Tübingen Machine Learning Cloud, DFG FKZ INST 37/1057-1 FUGG as well as the Training Center for Machine Learning (TCML). We also thank Lara Pollehn and Simon Gerstenecker for helpful discussions.

References

- [1] autonomousvision. Bench2drive leaderboard. URL: <https://github.com/autonomousvision/Bench2Drive-Leaderboard>, 2025. 2
- [2] Jens Beißwenger. Pdm-lite: A rule-based planner for carla leaderboard 2.0. URL: https://github.com/OpenDriveLab/DriveLM/blob/DriveLM-CARLA/pdm_lite/docs/report.pdf, 2024. 3
- [3] Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D. Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, Xin Zhang, Jake Zhao, and Karol Zieba. End to end learning for self-driving cars. *arXiv.org*, 1604.07316, 2016. 2
- [4] Wei Cao, Marcel Hallgarten, Tianyu Li, Daniel Dauner, Xunjiang Gu, Caojun Wang, Yakov Miron, Marco Aiello, Hongyang Li, Igor Gilitschenski, et al. Pseudo-simulation for autonomous driving. *Proc. Conf. on Robot Learning (CoRL)*, 2025. 3, 7, 8
- [5] Dian Chen and Philipp Krähenbühl. Learning from all vehicles. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2
- [6] Dian Chen, Brady Zhou, Vladlen Koltun, and Philipp Krähenbühl. Learning by cheating. In *Proc. Conf. on Robot Learning (CoRL)*, 2019. 2, 3
- [7] Dian Chen, Vladlen Koltun, and Philipp Krähenbühl. Learning to drive from a world on rails. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2021. 2
- [8] Li Chen, Penghao Wu, Kashyap Chitta, Bernhard Jaeger, Andreas Geiger, and Hongyang Li. End-to-end autonomous driving: Challenges and frontiers. *Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, 2024. 2
- [9] Kashyap Chitta, Aditya Prakash, and Andreas Geiger. Neat: Neural attention fields for end-to-end autonomous driving. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2021. 2
- [10] Kashyap Chitta, Aditya Prakash, Bernhard Jaeger, Zehao Yu, Katrin Renz, and Andreas Geiger. Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, 2023. 2, 4, 7, 8
- [11] Kashyap Chitta, Daniel Dauner, and Andreas Geiger. Sledge: Synthesizing driving environments with generative models and rule-based traffic. In *Proc. of the European Conf. on Computer Vision (ECCV)*, 2024. 7
- [12] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling, 2014. 4
- [13] Felipe Codevilla, Antonio M. Lopez, Vladlen Koltun, and Alexey Dosovitskiy. On offline evaluation of vision-based driving models. In *Proc. of the European Conf. on Computer Vision (ECCV)*, 2018. 2, 3
- [14] Felipe Codevilla, Matthias Müller, Antonio López, Vladlen Koltun, and Alexey Dosovitskiy. End-to-end driving via conditional imitation learning. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA)*, 2018. 2, 3
- [15] Felipe Codevilla, Eder Santana, Antonio M. López, and Adrien Gaidon. Exploring the limitations of behavior cloning for autonomous driving. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2019. 2
- [16] Marco Cusumano-Towner, David Hafner, Alex Hertzberg, Brody Huval, Aleksei Petrenko, Eugene Vinitsky, Erik Wilmans, Taylor Killian, Stuart Bowers, Ozan Sener, Philipp Krähenbühl, and Vladlen Koltun. Robust autonomy emerges from self-play. In *Proc. of the International Conf. on Machine Learning (ICML)*, 2025. 3
- [17] Daniel Dauner, Marcel Hallgarten, Andreas Geiger, and Kashyap Chitta. Parting with misconceptions about learning-based vehicle motion planning. In *Proc. Conf. on Robot Learning (CoRL)*, 2023. 3
- [18] Daniel Dauner, Marcel Hallgarten, Tianyu Li, Xinshuo Weng, Zhiyu Huang, Zetong Yang, Hongyang Li, Igor Gilitschenski, Boris Ivanovic, Marco Pavone, et al. Navsim: Data-driven non-reactive autonomous vehicle simulation and benchmarking. *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. 2, 3, 7, 8
- [19] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. CARLA: An open urban driving simulator. In *Proc. Conf. on Robot Learning (CoRL)*, 2017. 2, 3
- [20] Simon Gerstenecker, Andreas Geiger, and Katrin Renz. Plant 2.0: Exposing biases and structural flaws in closed-loop driving. *arXiv.org*, 2025. 4
- [21] Cole Gulino, Justin Fu, Wenjie Luo, George Tucker, Eli Bronstein, Yiren Lu, Jean Harb, Xinlei Pan, Yan Wang, Xiangyu Chen, John D. Co-Reyes, Rishabh Agarwal, Rebecca Roelofs, Yao Lu, Nico Montali, Paul Mouglin, Zoey Yang, Brandyn White, Aleksandra Faust, Rowan McAllister, Dragomir Anguelov, and Benjamin Sapp. Waymax: An accelerated, data-driven simulator for large-scale autonomous driving research. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 7
- [22] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, et al. Planning-oriented autonomous driving. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2023. 3, 6
- [23] Bernhard Jaeger. Expert drivers for autonomous driving. Master's thesis, University of Tübingen, 2021. 2, 3
- [24] Bernhard Jaeger. Transfuser versions. URL: https://github.com/autonomousvision/carla_garage/blob/leaderboard.2/docs/history.md, 2024. 4
- [25] Bernhard Jaeger, Kashyap Chitta, and Andreas Geiger. Hidden biases of end-to-end driving models. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2023. 2, 3, 4
- [26] Bernhard Jaeger, Daniel Dauner, Jens Beißwenger, Simon Gerstenecker, Kashyap Chitta, and Andreas Geiger. Carl: Learning scalable planning policies with simple rewards. *Proc. Conf. on Robot Learning (CoRL)*, 2025. 3
- [27] Xiaosong Jia, Zhenjie Yang, Qifeng Li, Zhiyuan Zhang, and Junchi Yan. Bench2drive: Towards multi-ability benchmarking of closed-loop end-to-end autonomous driving. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. 2, 3

- [28] Xiaosong Jia, Junqi You, Zhiyuan Zhang, and Junchi Yan. Drivetransformer: Unified transformer for scalable end-to-end autonomous driving. In *Proc. of the International Conf. on Learning Representations (ICLR)*. OpenReview.net, 2025. 2
- [29] Peter Karkus, Maximilian Igl, Yuxiao Chen, Kashyap Chitta, Jef Packer, Bertrand Douillard, Ran Tian, Alexander Naumann, Guillermo Garcia-Cobo, Shuhan Tan, Alperen Değirmenci, Alexander Popov, Nikolai Smolyanskiy, Urs Muller, B. Ivanovic, and Marco Pavone. Beyond behavior cloning in autonomous driving: a survey of closed-loop training techniques, 2025. 8
- [30] Napat Karnchanachari, Dimitris Geromichalos, Kok Seang Tan, Nanxiang Li, Christopher Eriksen, Shakiba Yaghoubi, Noushin Mehdipour, Gianmarco Bernasconi, Whye Kit Fong, Yiluan Guo, and Holger Caesar. Towards learning-based planning: The nuplan benchmark for real-world autonomous driving. In *Proc. IEEE International Conf. on Robotics and Automation (ICRA)*, 2024. 7
- [31] Saman Kazemkhani, Aarav Pandya, Daphne Cornelisse, Brennan Shacklett, and Eugene Vinitsky. Gpudrive: Data-driven, multi-agent driving simulation at 1 million FPS. In *Proc. of the International Conf. on Learning Representations (ICLR)*, 2025. 7
- [32] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 2023. 7
- [33] Quanyi Li, Zhenghao Peng, Lan Feng, Qihang Zhang, Zhenghai Xue, and Bolei Zhou. Metadrive: Composing diverse driving scenarios for generalizable reinforcement learning. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 2022. 3
- [34] Qifeng Li, Xiaosong Jia, Shaobo Wang, and Junchi Yan. Think2drive: Efficient reinforcement learning by thinking with latent world model for autonomous driving (in CARLA-V2). In *Proc. of the European Conf. on Computer Vision (ECCV)*, 2024. 2, 3
- [35] Tianyu Li, Yihang Qiu, Zhenhua Wu, Carl Lindström, Peng Su, Matthias Nießner, and Hongyang Li. MTGS: Multi-traversal gaussian splatting. *arXiv.org*, 2503.12552, 2025. 7
- [36] Zhiqi Li, Zhiding Yu, Shiyi Lan, Jiahua Li, Jan Kautz, Tong Lu, and José M. Álvarez. Is ego status all you need for open-loop end-to-end autonomous driving? In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2024. 3
- [37] Bencheng Liao, Shaoyu Chen, Haoran Yin, Bo Jiang, Cheng Wang, Sixu Yan, Xinbang Zhang, Xiangyu Li, Ying Zhang, Qian Zhang, and Xinggang Wang. Diffusiondrive: Truncated diffusion model for end-to-end autonomous driving. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2025. 2
- [38] Shuai Liu, Quanmin Liang, Zefeng Li, Boyang Li, and Kai Huang. Gaussianfusion: Gaussian-based multi-sensor fusion for end-to-end autonomous driving. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. 2
- [39] William Ljungbergh, Adam Tonderski, Joakim Johnander, Holger Caesar, Kalle Åström, Michael Felsberg, and Christoffer Petersson. Neuroncap: Photorealistic closed-loop safety testing for autonomous driving. In *Proc. of the European Conf. on Computer Vision (ECCV)*, 2024. 3, 7
- [40] Nico Messikommer, Jiaxu Xing, Elie Aljalbout, and Davide Scaramuzza. Student-informed teacher training, 2025. 3
- [41] Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory F. Diamos, Erich Elsen, David García, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, and Hao Wu. Mixed precision training. In *Proc. of the International Conf. on Learning Representations (ICLR)*, 2018. 6
- [42] Dean Pomerleau. ALVINN: an autonomous land vehicle in a neural network. In *Advances in Neural Information Processing Systems (NIPS)*, 1988. 2
- [43] Aditya Prakash, Kashyap Chitta, and Andreas Geiger. Multi-modal fusion transformer for end-to-end autonomous driving. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2, 4
- [44] Ilija Radosavovic, Raj Prateek Kosaraju, Ross Girshick, Kaiming He, and Piotr Dollár. Designing network design spaces, 2020. 6
- [45] Katrin Renz, Long Chen, Elahe Arani, and Oleg Sinavski. Simlingo: Vision-only closed-loop autonomous driving with language-action alignment. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2025. 2, 4, 6, 7
- [46] Lloyd Russell, Anthony Hu, Lorenzo Bertoni, George Fedoseev, Jamie Shotton, Elahe Arani, and Gianluca Corrado. GAIA-2: A controllable multi-view generative world model for autonomous driving. *arXiv.org*, 2503.20523, 2025. 7
- [47] Shital Shah, Debadeepta Dey, Chris Lovett, and Ashish Kapoor. Airsim: High-fidelity visual and physical simulation for autonomous vehicles. In *Field and service robotics: Results of the 11th international conference*, pages 621–635. Springer, 2017. 3
- [48] Hao Shao, Letian Wang, RuoBing Chen, Hongsheng Li, and Yu Liu. Safety-enhanced autonomous driving using interpretable sensor fusion transformer. In *Proc. Conf. on Robot Learning (CoRL)*, 2022. 2
- [49] Hao Shao, Letian Wang, Ruobing Chen, Steven L. Waslander, Hongsheng Li, and Yu Liu. Reasonnet: End-to-end driving with temporal and global reasoning. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2
- [50] Chonghao Sima, Katrin Renz, Kashyap Chitta, Li Chen, Hanxue Zhang, Chengen Xie, Jens Beißwenger, Ping Luo, Andreas Geiger, and Hongyang Li. Pdm-lite dataset for carla leaderboard 2.0. URL: <https://huggingface.co/datasets/autonomousvision/PDM.Lite.Carla.LB2>, 2024. 2
- [51] Chonghao Sima, Katrin Renz, Kashyap Chitta, Li Chen, Hanxue Zhang, Chengen Xie, Jens Beißwenger, Ping Luo, Andreas Geiger, and Hongyang Li. Drivelm: Driving with graph visual question answering. In *Proc. of the European Conf. on Computer Vision (ECCV)*, 2024. 2, 3, 4, 6, 7
- [52] Yingqi Tang, Zhuoran Xu, Zhaotie Meng, and Erkang Cheng. Hip-ad: Hierarchical and multi-granularity planning

- with deformable attention for autonomous driving in a single decoder. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2025. 2, 6, 7
- [53] Aaron Walsman, Muru Zhang, Sanjiban Choudhury, Dieter Fox, and Ali Farhadi. Impossibly good experts and how to follow them. In *Proc. of the International Conf. on Learning Representations (ICLR)*, 2023. 3
- [54] Andrew Warrington, J. Wilder Lavington, Adam Ścibior, Mark Schmidt, and Frank Wood. Robust asymmetric learning in pomdps, 2021. 3
- [55] Luca Weihs, Unnat Jain, Iou-Jen Liu, Jordi Salvador, Svetlana Lazebnik, Aniruddha Kembhavi, and Alexander Schwing. Bridging the imitation gap by adaptive insubordination, 2021. 3
- [56] Xinshuo Weng, Boris Ivanovic, Yan Wang, Yue Wang, and Marco Pavone. Para-drive: Parallelized architecture for real-time autonomous driving. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2024. 3
- [57] Penghao Wu, Xiaosong Jia, Li Chen, Junchi Yan, Hongyang Li, and Yu Qiao. Trajectory-guided control prediction for end-to-end autonomous driving: A simple yet strong baseline. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 2
- [58] Bernhard Wymann, Christos Dimitrakakis, Andrew Sumnery, Eric Espié, and Christophe Guionneauz. Torcs: The open racing car simulator, 2015. 3
- [59] Runsheng Xu, Hubert Lin, Wonseok Jeon, Hao Feng, Yuliang Zou, Liting Sun, John Gorman, Kate Tolstaya, Sarah Tang, Brandyn White, et al. Wod-e2e: Waymo open dataset for end-to-end driving in challenging long-tail scenarios. *arXiv.org*, 2025. 2, 3, 7
- [60] Ze Yang, Yun Chen, Jingkang Wang, Sivabalan Manivasagam, Wei-Chiu Ma, Anqi Joyce Yang, and Raquel Urtasun. Unisim: A neural closed-loop sensor simulator. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2023. 3
- [61] Jiang-Tian Zhai, Ze Feng, Jihao Du, Yongqiang Mao, Jiang-Jiang Liu, Zichang Tan, Yifu Zhang, Xiaoqing Ye, and Jingdong Wang. Rethinking the open-loop evaluation of end-to-end autonomous driving in nuscenes. *arXiv.org*, 2023. 3
- [62] Zhejun Zhang, Alexander Liniger, Dengxin Dai, Fisher Yu, and Luc Van Gool. End-to-end urban driving by imitating a reinforcement learning coach. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2021. 3
- [63] Hongyu Zhou, Longzhong Lin, Jiabao Wang, Yichong Lu, Dongfeng Bai, Bingbing Liu, Yue Wang, Andreas Geiger, and Yiyi Liao. Hugsim: A real-time, photo-realistic and closed-loop simulator for autonomous driving. *arXiv.org*, 2024. 3, 7
- [64] Julian Zimmerlin, Jens Beißwenger, Bernhard Jaeger, Andreas Geiger, and Kashyap Chitta. Hidden biases of end-to-end driving datasets. *arXiv.org*, 2412.09602, 2024. 2, 3, 5, 6, 7