Stability and Failure Modes of End-to-End Autonomous Driving under Limited Vision and Sparse Range Sensing

Shreyas Talole* Antje Muntzinger*

Abstract: End-to-end learning is widely used in autonomous driving, yet its stability under limited perception is not well understood. We analyze end-to-end driving policies operating with limited visual resolution and sparse two-dimensional range sensing. Using a controlled simulation setup, we evaluate lightweight imitation-learned architectures and assess their behavior in closed-loop driving. Our results reveal characteristic failure modes such as accumulated drift and oscillatory control, which are influenced by control dimensionality, state feedback, and sensor fusion. These insights clarify the conditions under which end-to-end approaches remain stable and highlight directions for enhancing robustness through temporal modeling, structural priors, and integration of additional constraints.

Keywords: Autonomous driving, end-to-end learning, imitation learning, behavior cloning, sensor fusion, steering control

1 Introduction and Related Work

End-to-end learning has become a prominent paradigm in autonomous driving, with early work demonstrating direct mapping from camera images to steering commands [1]. More recent approaches extend end-to-end policies to richer control outputs, temporal modeling, and multi-modal perception, achieving impressive performance in large-scale real-world deployments when trained with extensive data and sensor configurations [2, 3, 4].

However, most existing studies assume rich, high-resolution sensor setups, while the stability of end-to-end driving under strongly constrained perception remains poorly understood [5, 6]. In this work, we investigate the stability and failure modes of end-to-end learned driving policies operating under limited sensor input.

We consider a perception setup with restricted visual resolution and sparse two-dimensional range sensing, representing a bottlenecked sensory regime with partial observability. Using a controlled simulation environment, we evaluate representative lightweight end-to-end architectures inspired by established camera- and LiDAR-based models, both individually and in naive fusion configurations. All models are trained via imitation learning (IL), and their behavior is assessed over extended driving sequences rather than short-term prediction accuracy. Imitation learning focuses of understanding the states available and policies as defined by the expert. The IL models thus tries to copy the expert behavior [7]. Behavior Cloning works on the assumption that to make a decision

*Both authors are with Stuttgart University of Applied Sciences (e-mail: {firstname}.{lastname}@hft-stuttgart.de).

or prediction, all information can be inferred from the perception (mostly via all possible sensors) [7].

The dataset of expert demonstrations is defined as:

$$\mathcal{D} = \{\langle o_i, a_i \rangle\}_{i=1}^{N} \tag{1}$$

where o_i is the observation and a_i is the expert's corresponding action at step i.

The parameters θ of a function approximator $F(o;\, \theta)$ are optimized by minimizing a surrogate loss over this dataset [8, 9]:

$$\underset{\theta}{\text{minimize}} \sum_{i} \ell\big(F(o_i;\, \theta),\ a_i\big) \tag{2}$$

This formulation implicitly assumes a deterministic mapping from observations to actions — that is, there exists an expert function E such that $a_i = E(o_i)$ [8]. If this holds, a sufficiently expressive approximator F can learn E given enough data.

2 End-to-End Architectures

We evaluate a set of representative lightweight end-to-end architectures: While PilotNet [1] serves as a known baseline, the remaining architectures are purpose-built variants used as experimental probes. The goal is not architectural novelty, but a controlled analysis of stability and failure behavior under constrained sensory input.

PilotNet. PilotNet receives monocular camera images with limited resolution as input and predicts steering commands, while throttle is kept constant. This setup represents the simplest end-to-end configuration and is used to establish baseline stability behavior under constrained visual perception. It acts as a reference backbone for further exploration and modifications within the architecture.

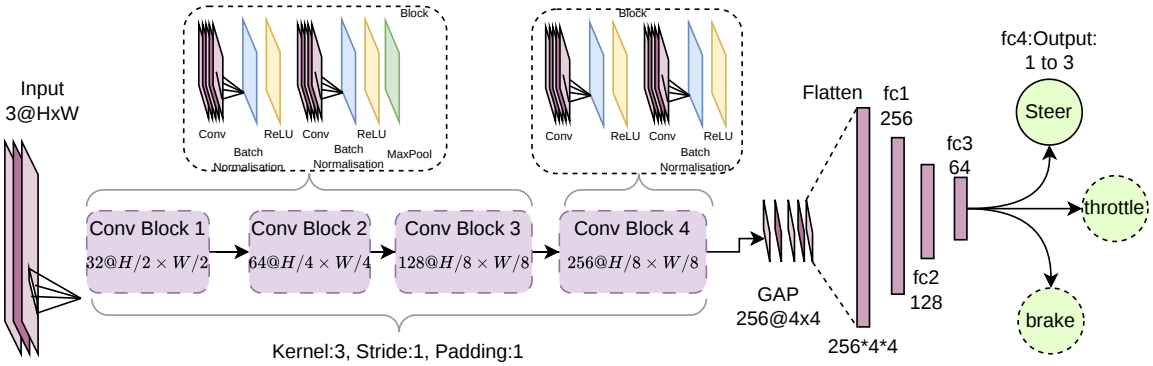

Figure 1: VisionDriveNet Architecture

VisionDriveNet. This vision-based variant predicts steering and potentially throttle and brake commands. It accepts any resolution image; in this work, we used resolutions of 132×400 and 400×400 pixels. The final evaluations are made with the later. The architecture comprises four convolution blocks followed by a global average pooling (GAP) layer and four fully connected layers (including output layer). Each block contains two sets of a convolutional layer followed by batch normalization and ReLU. Except for the last convolutional block, all other convolutional blocks have a MaxPooling layer to reduce the spatial dimensions. Similar to Simonyan et al.'s [10] deep ConvNet (often referred to

as VGG), our architecture relies on a stack of convolutional layers with small receptive fields for feature extraction. An overview of the VisionDriveNet architecture is shown in Fig. 1. This architecture serves as a more complex end-to-end policy that jointly predicts multiple control outputs, allowing us to study the effect of control dimensionality on stability.

VisionLidarNet. VisionLidarNet augments the monocular camera input with sparse two-dimensional range measurements. Camera and range data captured via a 2D-LiDAR sensor are combined with a late-fusion in an end-to-end manner without explicit intermediate representations. They follow a base structure similar to VisionDriveNet but has higher GAP channels and one extra fully connected layer. Figure 2 shows the architecture of VisionLidarNet. This architecture allows us to study the effect of naive sensor fusion of sparse range sensing on stability under constrained perception.

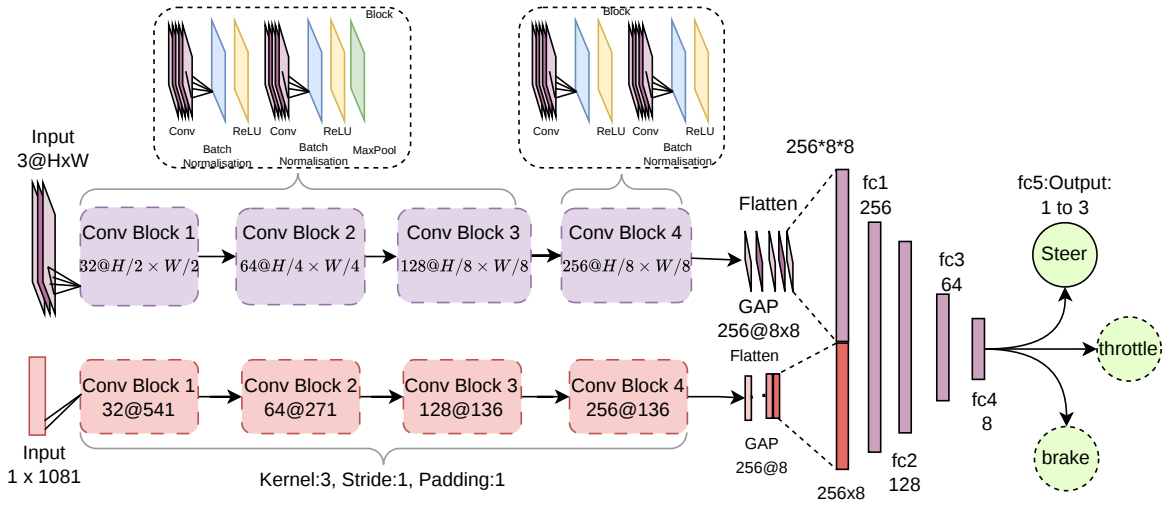

Figure 2: VisionLidarNet Architecture

3 Data Collection and Experimental Setup

All experiments are conducted in the CARLA simulator [11]. The simulated route includes straight road segments, left and right curves, T-intersections, cross-intersections, and roundabouts. To introduce perceptual variation while preserving identical road geometry, data is collected under dynamic weather conditions utilizing a CARLA script for a complete weather cycle from night to noon with dynamically changing wind, cloud density, and rain. Fig. 3 shows the view for each weather condition upon which we later execute our evaluations. Expert demonstrations are generated using the CARLA autopilot, while the presence of other traffic participants is enabled or disabled via the traffic manager to study the effect of dynamic obstacles on stability. Data is collected in multiple driving sessions across multiple town setups. Sensor data consists of monocular camera images, sparse two-dimensional range measurements, and associated vehicle state information. Camera images are recorded at a resolution of 600×800 and downsampled to 132×400 or 400×400 for training and evaluation and a field of view of $90°$. The resolution is therefore increased compared to the original PilotNet setup of 66×200 to ensure that traffic signals and signs are still discernible after downsampling for throttle and brake prediction. Sparse range measurements are obtained from a 2D LiDAR with a

(a) Clear Noon

(b) Noon + Hard Rain

(c) Clear Night

(d) Night + Hard Rain

Figure 3: View for different Weather Settings

maximum range of 50 m. Sensor streams are temporally synchronized during data recording. In total, 419,621 frames are collected and split into 335,746 training frames and 83,875 validation frames. All architectures are trained with imitation learning using the same dataset, loss function, optimizer, and training procedure.

During the evaluation, all models are deployed in closed-loop control for open driving under multiple specific and dynamic weather conditions. Failures are detected based on collisions, lane invasions, traffic signal violations, and excessive lateral deviation. An ego vehicle is randomly spawned within the CARLA map and it continues to openly drive for a duration of 120 seconds and three additional cool-down seconds. During this time, the number of collisions, lane invasions, wrong lane driving and traffic signal violations are recorded and logged. In addition, the time to the first occurrence of each above incident for each lap is measured. Each evaluation is repeated three times for each weather setting and a total of fifteen times for each Strategy to account for variability in the closed-loop behavior, and the reported metrics are averaged across runs. The ego vehicle was reset to the nearest spawn point in the event of reaching an irrecoverable state.

4 Results

The following tables summarize the stability metrics for the different end-to-end architectures under constrained perception. We tabulate per model metrics for different weather types and discuss the observations accordingly. Each metric represented as a column contains the data as **median value/mean value/standard deviation**.

Weather	**Total Collisions**	**Lane Violations**	**Total Red Light Violations**	**Total Wrong Lane Incidents**
Dynamic	1/2.33/1.89	13/12.67/3.68	3/2.67/1.25	1/1.67/0.94
Noon	3/3.33/0.47	17/17/4.08	4/4/0.82	3/2.67/0.47
Noon + Hard Rain	2/1.67/0.47	18/19.67/3.09	3/2/1.41	0/0.67/0.94
Night	2/2.33/1.25	9/11/2.83	1/1.33/1.25	1/1/0.82
Night + Hard Rain	2/1.67/1.25	7/6/1.41	1/1/0.82	1/1/0.82
Total	2/2.27/1.34	13/13.27/5.71	3/2.2/1.56	1/1.4/1.08

Table 1: Stability metrics for VisionDriveNet (VDN) under constrained perception represented as median/mean/standard deviation.

We evaluate the proposed driving strategies under five weather settings: *Dynamic* weather, in which conditions continuously rotate between day and night with varying rain, wind, and fog, *Noon*, *Noon with Hard Rain*, *Night*, and *Night with Hard Rain*. Tables 1 and 2 summarize collisions, lane-violations, red-light violations and wrong-lane incidents statistics for VisionDriveNet (VDN) and VisionLidarNet (VLN), respectively.

For VDN, the results show a clear dependence on weather and illumination. In the collision-related metrics, *Night with Hard Rain* yields the lowest collision count followed by *Night*, indicating the safest driving, although the variability remains high across runs. By contrast, *Noon* is the weakest setting, with higher collision and violation counts. This may seem counterintuitive at first, but it suggests that the model may be confused by the high-contrast and dynamic visual features present in the noon setting, such as strong shadows and reflections, which can lead to misinterpretation of the scene. In contrast, the more uniform lighting conditions at night may provide a more stable visual input for the model, even though it is generally more challenging for human drivers. The presence of hard rain seems to have a complex effect: while it degrades visibility, it may also reduce the amount of distracting visual information, leading to fewer collisions in some cases.

The lane-violation results follow a similar pattern: *Night* and *Night with Hard Rain* generally support safer driving, whereas *Noon* produces earlier and more frequent deviations. *Dynamic* weather produces intermediate performance, closely reflecting the average behavior across the evaluated iterations.

For VLN, the collision profile is generally stronger than that of VDN. The model typically achieves lower mean collision counts at night and in rainy conditions, suggesting improved robustness to perception degradation. For lane violations, *Night* and *Night with Hard Rain* produce the lowest mean violation counts, and VLN avoids wrong-lane incidents entirely in both settings. This indicates substantially stronger lane-keeping behavior in low-light conditions. The red-light and wrong-lane analyses further reinforce these findings. For VDN, *Noon* is the most unstable setting, with the highest red-light violation rate. In contrast, *Night* and *Night with Hard Rain* are more favorable, showing

Weather	Total Collisions	Lane Violations	Total Red Light Violations	Total Wrong Lane Incidents
Dynamic	1/1.33/1.25	7/5.67/4.19	0/1/1.41	0/1/1.41
Noon	1/0.67/0.47	7/7/1.63	6/6/0.82	3/2.67/1.25
Noon + Hard Rain	0/1/1.41	5/7.33/4.03	5/5/0.82	1/1.33/0.47
Night	1/0.67/0.47	1/1.33/1.25	2/2/0.82	0/0/0
Night + Hard Rain	1/1.33/1.25	4/3.33/2.49	0/0.33/0.47	0/0/0
Total	1/1/1.1	5/4.93/3.75	3/2.87/2.42	0/1/1.32

Table 2: Stability metrics for VisionLidarNet (VLN) under constrained perception represented as median/mean/std-deviation.

lower incidents of red-light violation and more stable behavior overall. A similar trend appears for wrong-lane incidents, where *Noon* again performs worst, while *Night* and *Night with Hard Rain* provide the safer handling with lower mean wrong-lane events. VDN encounters the lowest mean wrong lane driving incidents during *Noon with Hard Rain*. VLN follows the same general trend but with better robustness: red-light violations are lowest at *Night with Hard Rain* followed by *Night*, and wrong-lane incidents are completely eliminated in *Night* and *Night with Hard Rain*.

In general, both driving strategies are sensitive to weather variation, but VLN is generally more robust than VDN in low-visibility and rainy conditions, especially in terms of collision avoidance and lane-keeping stability.

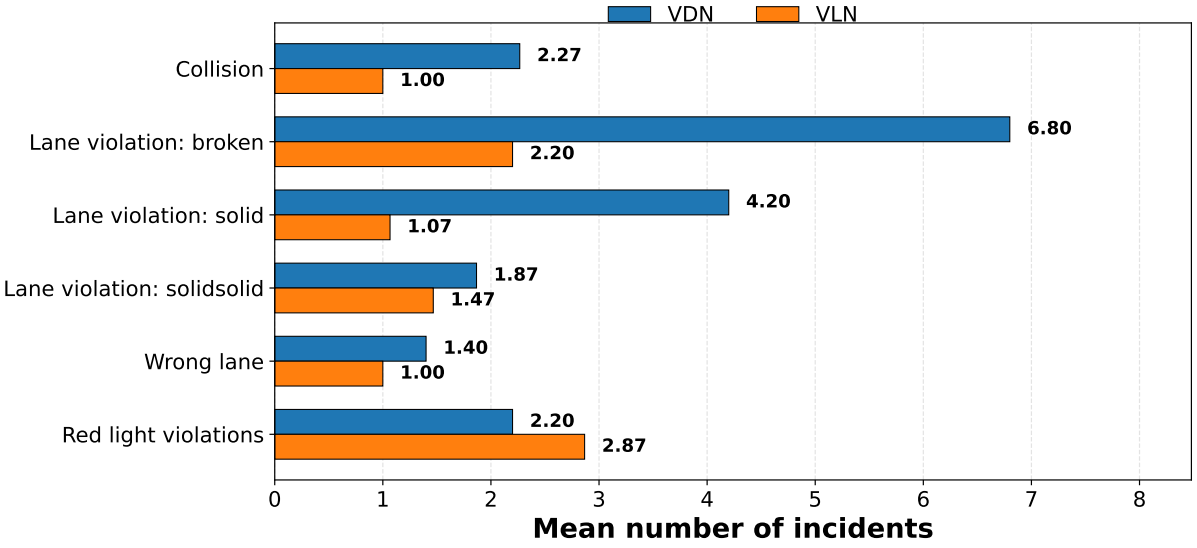

Figure 4: VisionDriveNet vs VisionLidarNet - Mean Incident Count

Figure 4 compares the mean incident counts for VDN and VLN across weather settings. The plot also distinguishes between different lane types: dashed (broken) lane markings, solid lane markings, and double lane markings ("solidsolid"). Taken together, these results indicate that both models remain vulnerable to weather-induced perception shifts, but VLN is consistently more resilient than VDN, particularly in adverse night and rainy

conditions. This suggests that the addition of sparse range sensing provides valuable information that enhances stability under challenging perception conditions.

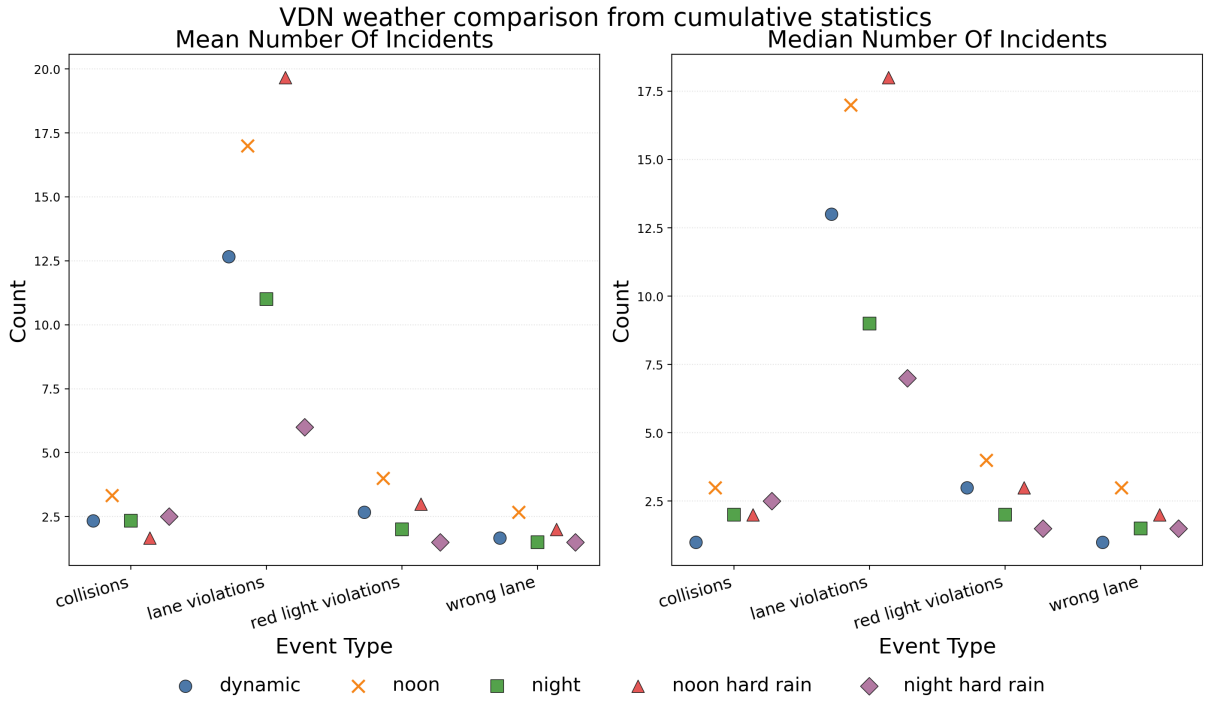

Figure 5: VisionDriveNet Incident Distribution

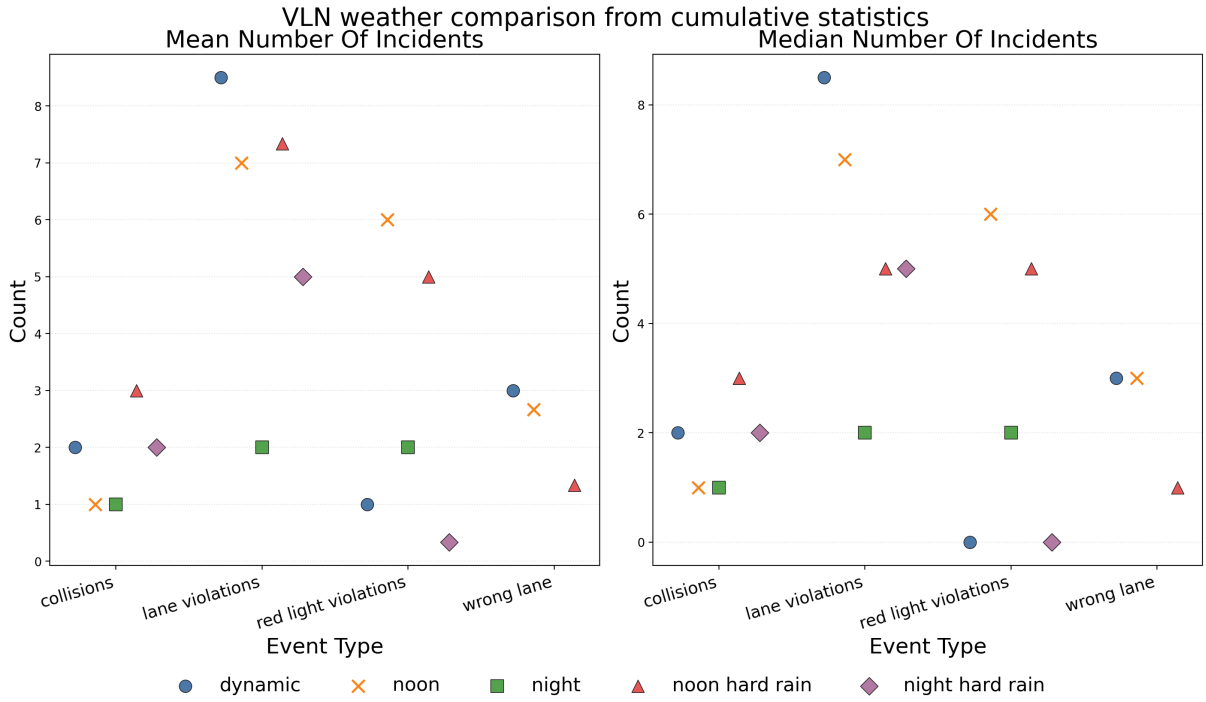

Figure 6: VisionLidarNet Incident Distribution

Figs. 5 and 6 project the weather wise distribution of incidents and the gravity of the margin by which the model has trouble extracting relevant information for the particular

aspect of basic driving ability such as lane maintenance and collision avoidance. The graphs also visualize if there exist any extreme outliers that dominate and disturb the mean values.

Besides incident statistics, we evaluate the lateral deviation from the lane center for each strategy. Table 3 and Table 4 show that VisionLidarNet has significantly reduced lateral deviation as compared VisionDriveNet, thus resulting in a much higher driving stability.

Weather	**Mean Deviation[m]**	**Median Deviation[m]**	**Std. Deviation[m]**	**Max Deviation[m]**
Dynamic	1.49	0.95	1.53	13.182
Noon	1.267	0.841	1.369	10.623
Noon + Hard Rain	1.067	0.799	1.050	11.925
Night	0.683	0.419	0.726	5.982
Night + Hard Rain	0.590	0.321	0.685	4.787
Total	1.0548	0.650	1.181	13.182

Table 3: Cumulative lateral deviation evaluation for **VisionDriveNet** with different weather settings

Weather	**Mean Deviation[m]**	**Median Deviation[m]**	**Std. Deviation[m]**	**Max Deviation[m]**
Dynamic	0.565	0.211	0.97	8.863
Noon	0.995	0.365	1.382	11.574
Noon + Hard Rain	0.752	0.324	1.051	6.935
Night	0.594	0.305	0.811	7.245
Night + Hard Rain	0.890	0.339	1.366	10.298
Total	0.776	0.309	1.162	11.574

Table 4: Cumulative lateral deviation evaluation for **VisionLidarNet** with different weather settings

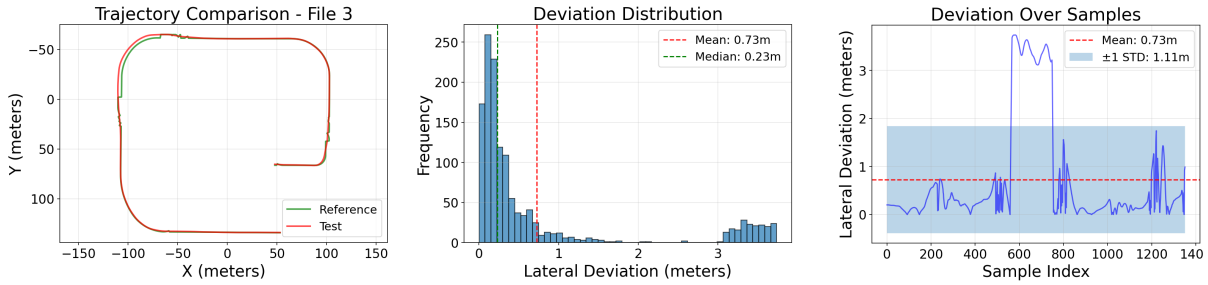

Figure 7: Qualitative example illustrating accumulated lateral drift: a) top-down view of the trajectory, b) lateral deviation distribution, c) lateral deviation over travel distance.

To illustrate the mechanisms behind the observed metrics, Fig. 7 shows a qualitative trajectory example for a VisionLidarNet-based end-to-end policy. The example highlights a characteristic failure mode observed across multiple architectures: gradual lateral drift.

The vehicle maintains initial stability and appears well-controlled over some time, but small systematic errors accumulate until the lane boundary is finally crossed. Interestingly, the vehicle continues driving in the wrong lane without further lane violations, then recovers back to the correct lane, and eventually completes the lap. This oscillatory behavior suggests that the policy is not fundamentally unstable, but rather operates near a stability boundary where small perturbations can lead to deviations over time. The observed failure mode is consistent with the quantitative metrics, which show a high number of lane violations despite relatively low collision rates.

5 Discussion and Future Work

End-to-end learning has demonstrated impressive performance in real-world autonomous driving systems when trained with large-scale datasets and rich sensor configurations. Our findings do not contradict these successes, but rather provide insight into the conditions under which end-to-end policies remain stable. By restricting perception and control structure, we expose failure modes that are masked in more data-rich and highly engineered systems.

The observed instability under constrained perception highlights the implicit reliance of end-to-end driving on perceptual richness, data diversity, and unmodeled structure. When these factors are reduced, purely data-driven policies struggle to maintain long-term stability, particularly under partial observability. This suggests that robust end-to-end driving in practice benefits from additional structure, whether through richer sensing, explicit temporal modeling, or physical constraints that stabilize long-horizon behavior.

Several limitations of this study should be noted. The reported metrics are based on a limited number of evaluation runs due to the computational and manual effort required for closed-loop simulation and resets. Future work will extend the analysis to multiple maps and larger-scale evaluations, and explore learning paradigms such as reinforcement or hybrid learning to improve robustness under changing environments.

Some experiments with a simple feedback loop, where the predicted control commands are fed back into the model as additional input, were conducted but did not yield significant improvements in stability. Also, the evaluation on a completely unknown map did not perform well, which is expected given the limited generalization capabilities of the trained models. We expect improvements from more sophisticated feedback mechanisms that explicitly model temporal dependencies and uncertainty, such as recurrent architectures or attention-based temporal fusion, as well as using the imitation learning models as a warm start for reinforcement learning.

Interestingly, even though the models were trained with other traffic participants enabled, the models did not learn to react to them appropriately. This indicates that passive exposure alone is insufficient; explicit supervision, targeted scenario design, or reward-based learning may be required for robust reactions to dynamic agents. Future work will explore reinforcement learning approaches that optimize for long-term stability rather than short-term imitation.

6 Conclusion

This paper presented an analysis of stability and failure modes in end-to-end autonomous driving under constrained perception with light-weight neural network architectures. Our results show that variations in control dimensionality, state feedback, and sensor fusion

influence stability. These insights help clarify the conditions under which end-to-end approaches remain stable and highlight promising directions for enhancing robustness through temporal modeling, structural priors, and integration of additional constraints.

References

[1] M. Bojarski, D. D. Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, X. Zhang, J. Zhao, and K. Zieba, "End to End Learning for Self-Driving Cars," 2016. [Online]. Available: https://arxiv.org/abs/1604.07316

[2] J.-J. Hwang, R. Xu, H. Lin, W.-C. Hung, J. Ji, K. Choi, D. Huang, T. He, P. Covington, B. Sapp, J. Guo, D. Anguelov, and M. Tan, "EMMA: End-to-End Multimodal Model for Autonomous Driving," 2024. [Online]. Available: https://arxiv.org/abs/2410.23262

[3] K. Chitta, A. Prakash, B. Jaeger, Z. Yu, K. Renz, and A. Geiger, "TransFuser: Imitation With Transformer-Based Sensor Fusion for Autonomous Driving," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 45, no. 11, pp. 12 878–12 895, 2023.

[4] Y. Xiao, F. Codevilla, A. Gurram, O. Urfalioglu, and A. M. López, "Multimodal End-to-End Autonomous Driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 1, pp. 537–547, 2022.

[5] P. S. Chib and P. Singh, "Recent Advancements in End-to-End Autonomous Driving Using Deep Learning: A Survey," *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 1, pp. 103–118, 2024.

[6] L. Chen, P. Wu, K. Chitta, B. Jaeger, A. Geiger, and H. Li, "End-to-End Autonomous Driving: Challenges and Frontiers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 12, pp. 10 164–10 183, 2024.

[7] L. Le Mero, D. Yi, M. Dianati, and A. Mouzakitis, "A survey on imitation learning techniques for end-to-end autonomous vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 14 128–14 147, 2022.

[8] F. Codevilla, M. Müller, A. López, V. Koltun, and A. Dosovitskiy, "End-to-end driving via conditional imitation learning," 2018. [Online]. Available: https://arxiv.org/abs/1710.02410

[9] T. Osa, J. Pajarinen, G. Neumann, J. A. Bagnell, P. Abbeel, and J. Peters, "An algorithmic perspective on imitation learning," *Foundations and Trends® in Robotics*, vol. 7, no. 1–2, p. 1–179, Mar. 2018. [Online]. Available: http://dx.doi.org/10.1561/2300000053

[10] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2015. [Online]. Available: https://arxiv.org/abs/1409.1556

[11] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An Open Urban Driving Simulator," in *Proc. Conf. Robot Learning (CoRL)*, 2017.