Is Ego Status All You Need for Open-Loop End-to-End Autonomous Driving?

Zhiqi Li[1,2*], Zhiding Yu[2†], Shiyi Lan[2], Jiahan Li[1], Jan Kautz[2], Tong Lu[1], Jose M. Alvarez[2]
[1]National Key Lab for Novel Software Technology, Nanjing University [2]NVIDIA

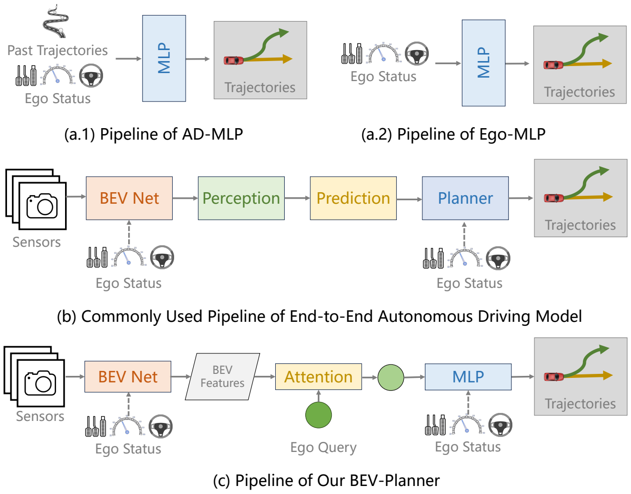
Abstract

End-to-end autonomous driving recently emerged as a promising research direction to target autonomy from a full-stack perspective. Along this line, many of the latest works follow an open-loop evaluation setting on nuScenes to study the planning behavior. In this paper, we delve deeper into the problem by conducting thorough analyses and demystifying more devils in the details. We initially observed that the nuScenes dataset, characterized by relatively simple driving scenarios, leads to an under-utilization of perception information in end-to-end models incorporating ego status, such as the ego vehicle's velocity. These models tend to rely predominantly on the ego vehicle's status for future path planning. Beyond the limitations of the dataset, we also note that current metrics do not comprehensively assess the planning quality, leading to potentially biased conclusions drawn from existing benchmarks. To address this issue, we introduce a new metric to evaluate whether the predicted trajectories adhere to the road. We further propose a simple baseline able to achieve competitive results without relying on perception annotations. Given the current limitations on the benchmark and metrics, we suggest the community reassess relevant prevailing research and be cautious about whether the continued pursuit of state-of-the-art would yield convincing and universal conclusions. Code and models are available at https://github.com/NVlabs/BEV-Planner.

1. Introduction

End-to-end autonomous driving aims to jointly consider perception and planning in a full-stack manner [1, 5, 32, 35]. An underlying motivation is to evaluate autonomous vehicle (AV) perception as a means to an end (planning), instead of overfitting to certain perception metrics.

Unlike perception, the planning is generally much more open-ended and hard to quantify [6, 7]. This open-ended nature of planning would ideally favor a closed-loop evaluation setting where other agents could react to the behavior of the ego vehicle, and the raw sensor data could also change accordingly. However, both agent behavior modeling and real-world data simulation within closed-loop simulators [8, 19] remain challenging open problems to date. As such, closed-loop evaluation inevitably introduces considerable domain gaps to the real world.



(a.1) Pipeline of AD-MLP (a.2) Pipeline of Ego-MLP

(b) Commonly Used Pipeline of End-to-End Autonomous Driving Model

(c) Pipeline of Our BEV-Planner

Figure 1. (a) AD-MLP uses both ego status and past trajectory GTs as input. Our reproduced version (Ego-MLP) drops the past trajectories. (b) The existing end-to-end autonomous driving pipeline consists of perception, prediction, and planning modules. Ego status can be integrated into the bird's-eye view (BEV) generation module or within the planning module. (c) We design a simple baseline for comparison with existing methods. The simple baseline does not leverage the perception or prediction module and directly predicts the final trajectories based on BEV features.

Open-loop evaluation, on the other hand, aims to treat human driving as the ground truth and formulate planning as imitation learning [13]. Such formulation allows the readily usage of real-world datasets via simple log-replay, avoiding the domain gaps from simulation. It also offers other advantages, such as the capacity to train and validate models in complex and diverse traffic scenarios, which are often difficult to generate with high fidelity in simulations [5]. For these benefits, a well-established body of re-

[*] Work done during an internship at NVIDIA.
[†] Corresponding author: zhidingy@nvidia.com.

search focuses on open-loop end-to-end autonomous driving with real-world dataset [2, 12, 13, 16, 43].

Current prevailing end-to-end autonomous driving methods [12, 13, 16, 43] commonly use nuScenes [2] for open-loop evaluation of their planning behavior. For instance, UniAD [13] studies the influence of different perception task modules to the final planning behavior. However, AD-MLP [45] recently points out that a simple MLP network can also achieve state-of-the-art planning results, relying solely on the ego status information. This motivates us to ask an important question:

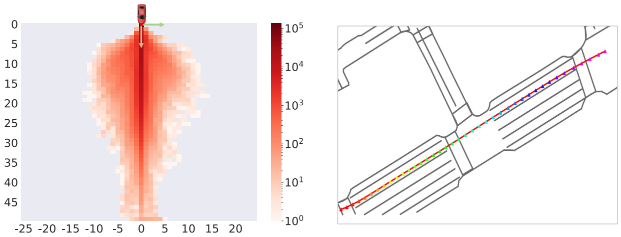
> *Is ego status all you need for open-loop end-to-end autonomous driving?*

Our answer is **yes and no**, considering both the pros and cons of using ego status in current benchmarks:

Yes. Information such as velocity, acceleration and yaw angle in the ego status should apparently benefit the planning task. To verify this, we fix an open issue[1] of AD-MLP and remove the use of history trajectory ground truths (GTs) to prevent potential label leakage. Our reproduced model, Ego-MLP (Fig. 1 a.2), relies solely on the ego status and is on par with state-of-the-art methods in terms of existing L2 distance and collision rate metrics. Another observation is that only existing methods [13, 16, 43], which incorporate ego status information within the planner module, can obtain results on par with Ego-MLP. Although these methods employ additional perception information (tracking, HD map, *etc.*), they don't demonstrate superiority compared to Ego-MLP. These observations verify the dominating role of ego status in the open-loop evaluation of end-to-end autonomous driving.

And No. It is also evident that autonomous driving as a safety-critical application should not depend solely on ego status for decision-making. So why does this phenomenon occur where using only ego status can achieve state-of-the-art planning results? To address the question, we present a comprehensive set of analyses covering existing open-loop end-to-end autonomous driving methods. We identify major shortcomings within existing research, including aspects related to datasets, evaluation metrics, and specific model implementations. We itemize and detail these shortcomings in the rest of the section:

Imblanced dataset. NuScenes is a commonly used benchmark for open-loop evaluation tasks[11–13, 16, 17, 43]. However, our analysis shows that 73.9% of the nuScenes data involve scenarios of driving straightforwardly, as reflected by the distribution of the trajectory in Fig. 2. For these straight-driving scenarios, maintaining the current velocity, direction, or turning rate can be sufficient most of the time. Hence, ego status information can be easily leveraged as a shortcut to fit the planning task, leading to the strong performance of Ego-MLP on nuScenes.



(a) Trajectory Heatmap

(b) Typical Scene of nuScenes

Figure 2. (a) The ego car trajectory heatmap on nuScenes dataset. (b) The majority of the scenes within the nuScenes dataset consist of straightforward driving situations.

Existing metrics are not comprehensive. The remaining 26.1% of nuScenes data involve more challenging driving scenarios for potentially better benchmarks of planning behaviors. However, we argue that the widely used current metrics, such as the L2 distance between prediction and planning GT and the collision rates between ego vehicle and surrounding obstacles, fail to accurately measure the model's planning behavior quality. Through visualizing numerous predicted trajectories generated from various methods, we note that some highly risky trajectories, such as running off the road may not get severely penalized in existing metrics. In response to this issue, we introduce a new metric to calculate the interaction rate between the predicted trajectories and the road boundaries. While focusing on intersection rates with road boundaries, the benchmark will experience a substantial transformation. In terms of this new metric, Ego-MLP tends to predict trajectories that deviate from the road more frequently than UniAD.

Ego status bias against driving logic. With ego status being a potential source causing overfitting, we further observe an interesting phenomenon. Our experiment results suggest that, in some cases, completely removing visual input from an existing end-to-end autonomous driving framework does not significantly degrade the planning behavior. This contradicts the basic driving logic in the sense that perception is expected to provide useful information for planning. For instance, blanking all the camera input in VAD [16] leads to complete failure of the perception module but minor degrade in planning, when ego status is present. However, altering the input ego velocity can significantly influence the final predicted trajectory.

In conclusion, we conjecture that recent efforts in end-to-end autonomous driving and their state-of-the-art scores on nuScenes are likely to be caused by the over-reliance on ego status, coupled with the dominance of simple driving scenarios. Furthermore, current metrics fall short in comprehensively assessing the quality of model-predicted trajecto-

[1] https://github.com/E2E-AD/AD-MLP/issues/4.

ries. These open issues and shortcomings may have underpresented the potential complexity of the planning task and created a **misleading impression** that ego status is all you need for open-loop end-to-end autonomous driving.

The potential interference of ego status in current open-loop end-to-end autonomous driving research raises another question: Is it possible to negate this influence by removing ego status from the whole model? However, it's important to note that even excluding the impact of ego status, *the reliability of open-loop autonomous driving research based on the nuScenes dataset remains in question.*

2. Related Work

2.1. BEV perception

In recent years, BEV-based autonomous driving perception methods have made great progress. Lift-Splat-Shoot [31] firstly propose to use latent depth distribution to perform view transformation. BEVFormer [22] introduces temporal clues into BEV perception and greatly boosts the 3D detection performance. A series of subsequent works [14, 15, 21, 23, 24, 26–28, 30, 41, 42] obtain more accurate 3D perception results by obtaining more accurate depth information or making better use of temporal information. The incorporation of temporal information typically necessitates the alignment of features across different timesteps [14, 18, 22, 39]. In the alignment process, the ego status is either implicitly encoded within the input feature [39] or is explicitly used to translate BEV features [14]. Methods [4, 20, 25, 29, 38, 44] explored map perception based on BEV features.

2.2. End-to-end autonomous driving

Modern autonomous driving systems are usually divided into three main tasks: perception, prediction, and planning. End-to-end autonomous driving that directs learning from raw sensor data to planning trajectories or driving commands eliminates the need for manual feature extraction, leading to efficient data utilization and adaptability to diverse driving scenarios. There exists a body of research [34, 37, 40] focused on closed-loop end-to-end driving within simulators [8, 19]. However, a domain gap persists between the simulator environment and the real world, particularly concerning sensor data and the motion status of agents. Recently, open-loop end-to-end autonomous driving has attracted more attention. End-to-end autonomous driving methods [3, 9, 13, 16, 33, 43] that involve learning intermediate tasks claim their effectiveness in improving final planning performance. AD-MLP [45] pointed out the issue of imbalanced data distribution in nuScenes and attempted to use only ego status as the model input to achieve arts performance. However, AD-MLP benefits from utilizing the historical trajectory of the ego car as input. Given that none of the existing methods use the historical trajectory information of the ego car, we argue that using the historical trajectory in open-loop autonomous driving is a subject of debate, as the model itself does not generate this historical trajectory but rather by an actual human driver.

3. Proposed BEV-Planner

In fact, ST-P3 [12], a previous method that often serves as a baseline, uses partially incorrect GT data during training and evaluation[2]. Consequently, when conducting comparisons between other methods and ST-P3, the validity of the conclusions drawn must be carefully evaluated. Therefore, in this paper, it is necessary for us to redesign a baseline method to compare with existing methods. At the same time, to better explore the impact of ego status, we also need a relatively clear baseline method. Based on these considerations, we have designed a very simple baseline in this paper, named **BEV-Planner**, as shown in the Fig. 1(c). For our pipeline, we first generate the BEV feature and concatenate it with history BEV features, mainly following the previous method [12, 14, 21]. Please note while concatenating BEV features from different timesteps, we didn't perform feature alignment. After obtaining the BEV features, we directly perform a cross-attention [36] between the BEV features and the ego query, which is a learnable embedding. The final trajectories are predicted based on the refined ego query through MLPs. The process can be formulated as follows:

$$\tau = \mathbf{MLP}(\mathbf{attn}(q=Q, k=B, v=B)), \tag{1}$$

where Q is the ego query, B is the BEV features after temporal fusion. τ is the final predicted trajectories.

To align with existing methods, we also designed baseline approaches that incorporate ego status into the BEV or planner modules. The strategy of incorporating ego status into the BEV aligns with previous approaches [13, 16, 22]. The strategy of incorporating ego status in the planner is directly concatenating the ego query with a vector containing ego status.

Compared to existing methods, this simple method didn't require any human-labeled data, including bounding boxes, tracking IDs, HD maps, *etc.* For this proposed baselines, we only use one L1 loss for trajectory supervision. We wish to underscore that our proposed baseline method is not intended for real-world deployment, owing to its deficiencies in providing adequate constraints and interoperability.

4. Experiments

4.1. Implementation Details

Our baseline uses an R50 backbone [10]. The input resolution is 256 × 704, smaller than existing methods [13, 16].

[2] https://github.com/OpenDriveLab/ST-P3/issues/24

The BEV resolution is 128×128 with a perception range of around 50 meters. For the baseline that uses history BEV features, we directly concatenate the BEV features from the past 4 timesteps to current BEV features along the channel dimension without alignment. A BEV encoder from method [14] is further used to squeeze the channel dimension to 256. We train our model for 12 epochs on 8 V100 GPUs, with a batch size of 32 and a learning rate of 1e-4.

4.2. Metrics

In the Appendix, we will introduce the shortcomings of the currently commonly used collision rate and another metric used to evaluate the smoothness of predicted trajectory.

Curb Collision Rate (CCR). In this study, to more comprehensively assess the quality of predicted trajectories, we employed a new metric that calculates the collision rate between the predicted trajectories and curbs (road boundaries). Staying on the road is vital for the safety of autonomous driving systems, yet existing evaluation metrics overlook the integration of map priors. Intuitively, safe trajectories should avoid collision with curbs. Our Curb Collision Rate (CCR) typically indicate the possibility of leaving the drivable area, which can pose safety hazards. We recognize that certain annotated road boundaries on nuScenes are indeed traversable, and ground truth trajectories may intersect with these boundaries under specific conditions. However, from a statistical viewpoint, this metric can effectively represent the overall rationality of the model's predicted trajectories. The implementation of the CCR metric is informed by the collision rate. To facilitate this, we rasterize the road boundary using a resolution of 0.1 meters. More details are in the Appendix.

Union Implementation. Given the lack of a standardized approach to assessment metrics, methodologies might differ in the nuances of their metric executions. In this study, we utilized the official open-source repositories from various methodologies to produce predicted trajectories. Subsequently, we employed a consistent metric implementation across all methods for evaluation, guaranteeing equity. Addressing concerns raised by AD-MLP [45] regarding the potential for false collisions due to a coarse-grained grid size (0.5m), we adopt a finer default grid size of 0.1m in our work to mitigate this issue.

4.3. Discussion

Ego status plays a key role. While focusing solely on previous metrics **L2 distance** and **collision rate**, it is observed that the simple strategy (ID-7), which simply continues straight at the current velocity, achieves surprisingly good results. The Ego-MLP model, which didn't leverage perception clues, is actually on par with UniAD and VAD, which use more complex pipelines. From another perspective, it is observed that existing methods can only match the performance of Ego-MLP when the ego vehicle's status is incorporated into the planner. In contrast, reliance solely on camera inputs leads to results significantly inferior to those achieved by Ego-MLP. Considering these observations, we may tentatively infer an intriguing conclusion: utilizing a combination of sensory information and ego status appears to yield results comparable to those achieved by employing ego status alone. Therefore, in models that integrate both ego vehicle status and perception information, a pertinent question arises: *What specific role does the perception information, acquired from camera inputs, play within the final planning module?*

Ego Status *vs.* Perceptual Information Undoubtedly, perceptual information constitutes the indispensable foundation of all autonomous driving systems, with ego status additionally offering crucial data such as the vehicle's velocity and acceleration to aid the system's decision-making process. Incorporating both perceptual information and ego status for the ultimate planning should indeed be a judicious strategy within an end-to-end autonomous driving system. However, as shown in Tab. 1, relying solely on ego status can yield planning results that are on par with or even superior to those methods that utilize both ego status and perception modules on previous L2 or collation rate metrics. To ascertain the roles that perceptual information and ego status play in the final planning process, we introduced varying degrees of perturbation to the images and the ego status, as shown in the Tab. 2. We use the official VAD model (which leverages ego status in the planner module) as the base model. It is observable that when disturbances are added to the images, the results of planning marginally decrease and may even exhibit improvement, while perceptual performance significantly deteriorates. Surprisingly, even when blank images are used as input, leading to the complete breakdown of the perception module, the model's planning capabilities remain largely unaffected. The corresponding visualization results are as illustrated in the Fig. 3. In contrast to the model's remarkable robustness to variations in image inputs, it exhibits considerable sensitivity to ego status. Upon altering the velocity of the ego car, we can observe that the planning results of the model are significantly impacted, as shown in Fig. 4. Setting the ego car's speed to 100 m/s results in the model generating wildly impractical planning trajectories. We posit that an autonomous driving system displaying such heightened sensitivity to ego status information harbors considerable safety risks. Furthermore, with planning results being predominantly dictated by ego status, the functions of other modules in the model cannot be reflected. For example, while comparing VAD(ID-6) and BEV-Planner++ (ID-12), we can observe that they

ID	Method	Ego Status		L2 (m) ↓				Collision (%) ↓				CCR (%) ↓				ckpt. source
		in BEV	in Planer	1s	2s	3s	Avg.	1s	2s	3s	Avg.	1s	2s	3s	Avg.	
0	ST-P3	✗	✗	1.59†	2.64†	3.73†	2.65†	0.69†	3.62†	8.39†	4.23†	2.53†	8.17†	14.4†	8.37†	Official
1	UniAD	✗	✗	0.59	1.01	1.48	1.03	0.16	0.51	1.64	0.77	0.35	1.46	3.99	1.93	Reproduce
2	UniAD	✓	✗	0.35	0.63	0.99	0.66	0.16	0.43	1.27	0.62	0.21	**1.32**	3.63	1.72	Official
3	UniAD	✓	✓	0.20	0.42	0.75	0.46	0.02	**0.25**	0.84	0.37	**0.20**	1.33	**3.24**	**1.59**	Reproduce
4	VAD-Base	✗	✗	0.69	1.22	1.83	1.25	0.06	0.68	2.52	1.09	1.02	3.44	7.00	3.82	Reproduce
5	VAD-Base	✓	✗	0.41	0.70	1.06	0.72	0.04	0.43	1.15	0.54	0.60	2.38	5.18	2.72	Official
6	VAD-Base	✓	✓	0.17	0.34	0.60	0.37	0.04	0.27	**0.67**	**0.33**	0.21	2.13	5.06	2.47	Official
7	GoStright	-	✓	0.38	0.79	1.33	0.83	0.15	0.60	2.50	1.08	2.07	8.09	15.7	8.62	-
8	Ego-MLP	-	✓	**0.15**	**0.32**	0.59	**0.35**	**0.00**	0.27	0.85	0.37	0.27	2.52	6.60	2.93	
9	BEV-Planner*	✗	✗	0.27	0.54	0.90	0.57	0.04	0.35	1.80	0.73	0.63	3.38	7.93	3.98	-
10	BEV-Planner	✗	✗	0.30	0.52	0.83	0.55	0.10	0.37	1.30	0.59	0.78	3.79	8.22	4.26	-
11	BEV-Planner+	✓	✗	0.28	0.42	0.68	0.46	0.04	0.37	1.07	0.49	0.70	3.77	8.15	4.21	-
12	BEV-Planner++	✓	✓	0.16	0.32	**0.57**	**0.35**	**0.00**	0.29	0.73	0.34	0.35	2.62	6.51	3.16	-

Table 1. **Open-loop planning performance.** †: The official implementation of ST-P3 (ID-0) utilized partial erroneous ground truth trajectories, with details provided in the appendix. The official UniAD (ID-2) utilized ego status in its BEV module. It is of particular note that the performance of the officially open-sourced model exceeds the results reported in the original paper [13]. We implemented minor modifications to the official codebases of UniAD and VAD to investigate the variations in results arising from different applications of ego status (ID-1, 3 & 4). A naive strategy (ID-7) of proceeding at the current speed also yields satisfactory results. Without the perception module, Ego-MLP (ID-8), utilizing solely ego velocity, acceleration, yaw angle, and driving command, achieves performance on par with current state-of-the-art models on previous L2 distance and collision rate metrics. *: Our simple baseline (ID-9) didn't utilize historical temporal information. The baseline (ID-10) utilizes the temporal clues from the past 4 frames. To ensure the comprehensiveness of our experiment, we also conduct investigations into the influence of ego status on our baseline model (ID-11& 12).

Method	Img Corruption	Ego Status Noise	L2 (m) ↓				Collision (%) ↓				CCR (%) ↓				Det. (NDS)	Map (mAP)
			1s	2s	3s	Avg.	1s	2s	3s	Avg.	1s	2s	3s	Avg.		
VAD-Base*	-	-	0.41	0.70	1.06	0.72	0.04	0.43	1.15	0.54	0.60	2.38	5.18	2.72	**46.0**	**47.5**
VAD-Base	-	-	**0.17**	**0.34**	**0.60**	**0.37**	0.04	0.27	0.67	0.33	0.21	2.13	5.06	2.47	45.5	47.0
VAD-Base	Snow	-	0.19	0.41	0.76	0.45	**0.00**	0.20	0.76	0.32	0.21	2.27	5.98	2.82	36.1	29.4
VAD-Base	Fog	-	0.19	0.40	0.75	0.45	0.02	0.20	0.68	0.30	0.23	2.21	5.90	2.78	34.3	29.4
VAD-Base	Glare	-	0.19	0.40	0.74	0.44	0.02	**0.18**	**0.59**	**0.26**	0.21	2.11	5.57	2.63	41.7	38.3
VAD-Base	Rain	-	0.19	0.41	0.75	0.45	0.02	**0.18**	0.66	0.29	0.31	2.38	5.98	2.89	29.1	13.0
VAD-Base	Blank	-	0.19	0.41	0.77	0.46	**0.00**	0.40	1.21	0.54	0.35	3.05	7.73	3.71	0.0	0.0
VAD-Base	-	$v\times0.0$	3.81	6.19	8.48	6.16	1.00	6.76	16.18	7.98	**0.19**	**0.41**	**3.10**	**1.23**	45.5	47.0
VAD-Base	-	$v\times0.5$	1.95	3.20	4.41	3.19	0.02	1.00	4.10	1.71	0.37	2.56	5.57	2,83	45.5	47.0
VAD-Base	-	$v\times1.5$	1.94	3.20	4.47	3.20	0.14	2.89	6.21	3.08	1.54	6.29	13.2	7.01	45.5	47.0
VAD-Base	-	$v=100m/s$	113	206	306	208	8.00†	9.66†	10.49†	9.38†	24.8†	27.5†	28.8†	27.0†	45.5	47.0

Table 2. **The VAD-base model's robustness to images and ego status.** To ascertain the impact of perceptual information and ego status on the ultimate planning performance, we systematically introduced noise into each component separately. We utilize the official VAD-Base checkpoint that uses ego status in its planner module. *: the results of VAD-Base without ego status in its planner. We can observe that introducing corruption to images markedly affects the perception results, especially in the case of using blank images; nonetheless, this does not markedly disrupt the ultimate planning results. In contrast to the minor impact of image corruption on planning, modifications to the ego vehicle's velocity have a significant effect on the planning results. Experimental results reveal that in an end-to-end model incorporating both ego status and perceptual information, decision-making is disproportionately influenced by ego status, thereby substantially increasing the model's safety risks. †: The collision rate is not precise as the ego car may have departed from the local BEV area. When the input velocity is zero, the model produces almost stationary trajectories, resulting in excellent performance in the CCR. This can be seen as a limitation of the CCR metric.

Method	Ego Status		L2 (m) ↓				Collision (%) ↓				CCR (%) ↓			
	in BEV	in Planer	1s	2s	3s	Avg.	1s	2s	3s	Avg.	1s	2s	3s	Avg.
BEV-Planner	✗	✗	0.30	0.52	0.83	0.55	0.10	0.37	1.30	0.59	0.78	3.79	8.22	4.26
BEV-Planner (init*)	✗	✗	0.26	0.49	0.81	0.52	0.03	0.20	1.00	0.42	0.59	3.18	7.36	3.71
BEV-Planner+Map	✗	✗	0.53	0.94	1.40	0.96	0.12	0.37	2.19	0.89	0.68	2.38	4.73	2.60

Table 3. While adding map perception task into the BEV-Planner method, we can observe that the model obtains worse L2 distance and collision rate performance, but achieves better CCR. We use a pretrained map perception checkpoint as the initialization of the BEV-Planner (init*) and BEV-Planner+Map model.

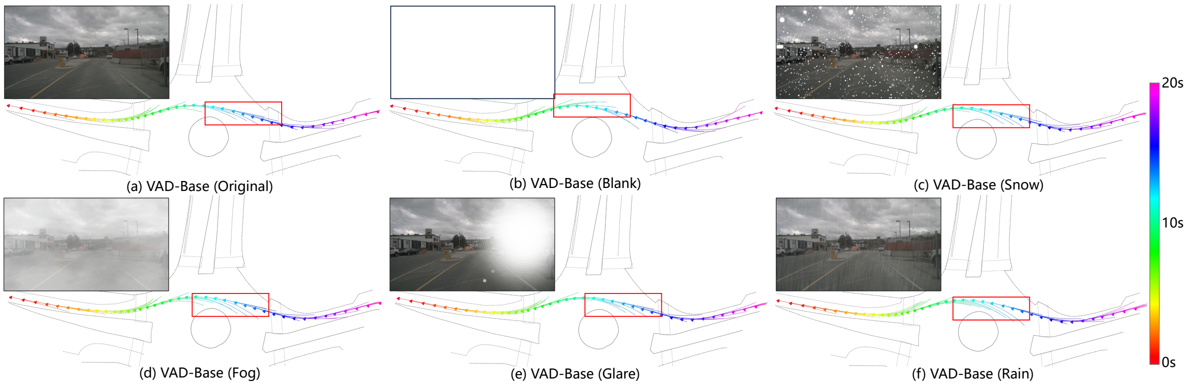


Figure 3. We exhibit the predicted trajectories of the VAD model (incorporating ego status in its planner) under various image corruptions. All trajectories within a given scene (spanning 20 seconds) are presented in the global coordinate system. Each triangular marker signifies a ground truth trajectory point of the ego vehicle, with different colors representing distinct timesteps. Notably, the model's predicted trajectory maintains plausibility, even when blank images serve as input. The trajectories within the red boxes, however, are suboptimal, as further elucidated in Appendix. While corruptions were applied to all surround-view images, for the sake of visualization, only the corresponding front-view images at the initial timestep are displayed.

Method	Ego Status		L2 (m) ↓				L2-ST (m) ↓				L2-LR (m) ↓			
	in BEV	in Planer	1s	2s	3s	Avg.	1s	2s	3s	Avg.	1s	2s	3s	Avg.
BEV-Planner	✗	✗	0.30	0.52	0.83	0.55	0.27	0.47	0.78	0.48	0.43	0.78	1.23	0.81
BEV-Planner+Map	✗	✗	0.53	0.94	1.40	0.96	0.54	0.95	1.42	0.97	0.52	0.87	1.28	0.89

Table 4. L2-ST is the L2 distance with going straight driving commands. L2-LR is the L2 distance with turning left/right commands.

Method	Ego Status		Collision (m) ↓				Collision-ST (m) ↓				Collision-LR (m) ↓			
	in BEV	in Planer	1s	2s	3s	Avg.	1s	2s	3s	Avg.	1s	2s	3s	Avg.
BEV-Planner	✗	✗	0.10	0.37	1.30	0.59	0.07	0.20	0.92	0.40	0.15	1.76	4.84	2.25
BEV-Planner+Map	✗	✗	0.12	0.37	2.19	0.89	0.14	0.38	2.20	0.91	0.00	0.29	2.05	0.78

Table 5. Collision-ST is the collision rate with going straight driving commands. Collision-LR is the collision rate with turning left/right commands.

obtain basically similar results in terms of L2 and collision rate. Is it justifiable to assert that our BEV-Planner++ design, characterized by its simplicity and effectiveness, can attain comparable outcomes to other more intricate methodologies, even in the absence of utilizing perception data? In fact, as the performance of the final planning module is predominantly influenced by the ego vehicle status, the design of other components does not significantly affect the planning results. Consequently, we argue that methods utilizing ego status are not directly comparable and conclusions should not be drawn from such comparisons.

How about not using ego status? Given that the ego vehicle status exerts a dominant influence on the planning results, it prompts an important inquiry: Is it feasible and beneficial to exclude ego status in open-loop end-to-end research?

Neglected Ego Status in Perception Stage. In fact, existing methods [16, 43] ignore the impact of using ego status on planning in BEV Encoder. More details are in the Appendix.

Without Ego Status, the Simpler, the Better? People might wonder why our BEV-Planner, without using additional perception tasks (including Depth, HD map, Tracking, *etc.*) and ego status, achieves better results in L2 distance and collision rate than other methods (ID-1 and 4). Since our BEV-Planner performs poorly in terms of CCR, what would happen if we added map perception tasks to our baseline? To address these questions, we designed a "BEV-Planner+Map" model by introducing a map perception task into our pipeline, mainly following the designs of UniAD. As shown in Tab. 3, when map perception is introduced, the model exhibits poorer results in terms of L2 distance and collision rate metrics. The only aspect that aligns with our expectations is that the introduction of map perception significantly reduces the CCR. Through a comparison of BEV-Planner with BEV-Planner (init*), we observe that the use of map-pretrained weights can enhance performance. This finding implies that the decrease in L2

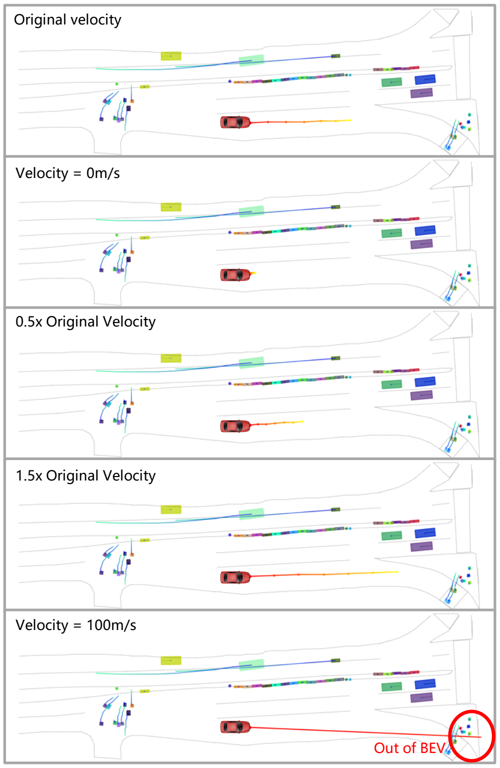


Figure 4. For the VAD-based model that incorporates ego status in its planner, we introduce the noise to the ego velocity with the visual inputs remaining constant. Notably, when the velocity data of the ego vehicle is perturbed, the resulting trajectories exhibit substantial alterations. Setting the vehicle's speed to zero results in a stationary prediction, while a speed of 100 m/s leads to the projection of an implausible trajectory. This indicates a disproportionate dependence of the model's decision-making process on the ego status, even though the perception module continues to provide accurate surrounding information.

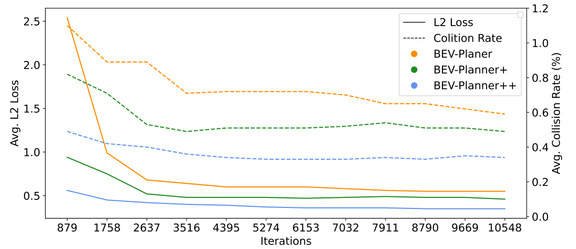


Figure 5. Introducing ego status in the BEV-Planner++ enables the model to converge very rapidly.

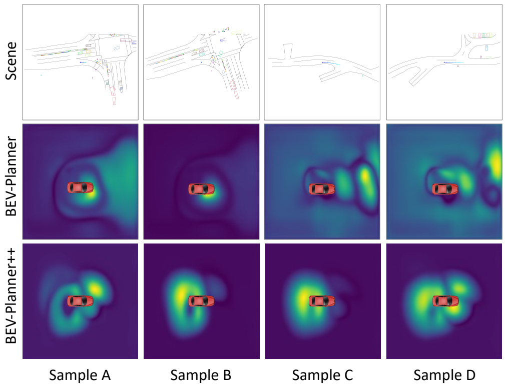


Figure 6. Comparing the BEV features of our baselines with the corresponding scenes.

Method	P.P.	Avg. L2(m)	Avg. Colli.(%)	Avg. CCR(%)
UniAD	✓	0.77	**0.51**	7.83
UniAD	✗	**0.66**	0.62	**1.72**

Table 6. P.P. indicates the post-processing optimization module of UniAD. We use the officially released weights of UniAD for this ablation study. By default, UniAD incorporates a post-processing step to refine the predicted trajectory from the end-to-end model, aiming to mitigate collisions with other agents. Nonetheless, this approach is limited by this singular optimization objective, which overlooks additional safety-critical factors in autonomous driving, such as lane adherence. Our new metrics, the CCR reveal that UniAD's post-processing substantially increases the risk of the ego vehicle running off the road.

and Collision rate observed with the integration of MapFormer in "BEV-Planner+Map" is not due to the pretrained weights. We posit that in most straight-driving scenarios, the addition of lane information may not yield markedly effective information and could indeed introduce some degree of interference. To verify our hypothesis, we evaluated the performance of these methods under varying driving commands. As shown in Tab. 4 and Tab. 5, adding map information significantly increases the L2 distance error and collision rate with going straight commands. In contrast, for turning scenarios, the incorporation of map information effectively reduces the collision rate. Based on the above observations, we can tentatively draw the following conclusions:

- In simple straightforward driving scenarios, the addition of perceptual information does not appear to enhance the model's performance with respect to L2 distance and collision rate. Conversely, the implementation of more intricate multi-task learning paradigms may, in fact, lead to a decrease in the model's overall efficacy.
- In more complex scenarios, such as turns, incorporating perceptual information can be beneficial for planning purposes. However, given the relatively small proportion (13%) of turning scenes in the existing evaluation datasets, the introduction of perceptual information tends to adversely affect the average performance metrics (L2 distance and collision rate) in the final analysis.
- It is imperative to develop a more robust and representative evaluation dataset. The metrics derived from the current evaluation dataset are not entirely persuasive and fail to accurately reflect the true capabilities of the model.

New metrics will bring new conclusions. The preceding methodology primarily centered around the L2 distance and collision rate metrics. Our discussion thus far has been largely concentrated on these two metrics. What we want to emphasize is that these two metrics, L2 distance and collision rate, only reflect a partial aspect of a model's planning capabilities. It is not advisable to assess the quality of a model based exclusively on these two metrics. In this paper, we introduce a new metric to evaluate the model's comprehension and adherence to the map: Curb Collision Rate (CCR). As shown in Tab. 1, we can observe that the GoStright strategy frequently intersects with road boundaries, which is in line with our expectations. In terms of this new metric, Ego-MLP performs worse than UniAD and VAD. Our method BEV-Planner, performs the worst on this metric because it does not use any map information. This suggests that relying on past metrics to judge the superiority of different open-loop methods is biased.

Based on our proposed new metrics, we also found that the existing collision rate metric can be manipulated with post-processing. More specifically, within UniAD [13], a non-linear optimization module is employed to refine the trajectory predicted by the end-to-end model, ensuring that the anticipated path steers clear of the occupancy grid, thereby aiming to prevent collisions. However, this optimization, while significantly reducing the collision rates with other agents, inadvertently introduces additional safety risks. The absence of adequate constraints in its optimization process, such as the integration of map priors, markedly increases the risk of the optimized trajectory encroaching upon road boundaries, as shown in Tab. 6. In this paper, we report the results of UniAD without its post-processing by default.

What we wish to underscore is that the primary intention behind proposing CCR is to illuminate the inadequacies within the existing evaluation systems. However, even with the incorporation of CCR, open-loop evaluation systems still encounter numerous challenges. We assert that the evaluation of open-loop autonomous driving systems necessitates a more diverse and stringent evaluation framework. This would enable a more accurate reflection of these systems' capabilities and limitations.

What the baseline learned in its BEV? As shown in Fig. 5, with the influence of ego status, the models converge rapidly. Considering the challenge of generating valuable BEV features from visual images and comparing the convergence curve of BEV-Planer that does not use ego status, this further demonstrates that ego status information dominates the learning process. Since our baselines are solely supervised by ego trajectory, we are wondering what the model is learning from the images. As shown in Fig. 6, we observed a distinct phenomenon: in BEV-Planner++, the activation range of the feature map predominantly encompasses the immediate vicinity around the ego vehicle, frequently manifesting behind the vehicle itself. This pattern marks a significant deviation from the BEV-Planner's BEV features, which typically concentrate on the area ahead of the vehicle. We speculate that this is due to the introduction of ego status information, which negates the model's need to extract information from BEV features. Hence, the BEV-Planner++ method has almost not learned any effective information.

5. Conclusion

In this paper, we present an in-depth analysis of the shortcomings inherent in current open-loop, end-to-end autonomous driving methods. Our objective is to contribute findings that will foster the progressive development of end-to-end autonomous driving.

Our conclusions are summarized as follows:

- The planning performance of existing open-loop autonomous driving models based on nuScenes is highly affected by ego status (velocity, acceleration, yaw angle). With ego status involved, the model's final predicted trajectories are basically dominated by it, resulting in a diminished use of sensory information.
- Existing planning metrics fall short of fully capturing the true performance of models. The evaluation results of the model may vary significantly across different metrics. We advocate for the adoption of more diverse and comprehensive metrics to prevent models from achieving local optimality on specific metrics, which may lead to the neglect of other safety hazards.
- Compared to pushing the state-of-the-art performance on the existing nuScenes dataset, we assert that the development of more appropriate datasets and metrics represents a more critical and urgent challenge to tackle.

Limitation There are trade-offs between different planning metrics. Designing an integrated evaluation system for open-loop evaluation presents a significant challenge. Although our baseline method excels in terms of L2 distance and collision rate, its performance is not exceptional in the CCR, primarily because our approach does not utilize any perception annotations, such as HD maps.

Acknowledgments

Thanks to Bencheng Liao and Shaoyu Chen for providing helpful information and the model weights of VAD. Tong Lu and Zhiqi Li are supported by the National Natural Science Foundation of China (Grant No. 62372223) and China Mobile Zijin Innovation Insititute (No. NR2310J7M). Zhiqi Li is also supported by the NVIDIA Graduate Fellowship Program.

References

[1] Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, et al. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*, 2016.

[2] 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. In *CVPR*, 2020.

[3] Sergio Casas, Abbas Sadat, and Raquel Urtasun. Mp3: A unified model to map, perceive, predict and plan. In *CVPR*, 2021.

[4] Li Chen, Chonghao Sima, Yang Li, Zehan Zheng, Jiajie Xu, Xiangwei Geng, Hongyang Li, Conghui He, Jianping Shi, Yu Qiao, et al. Persformer: 3d lane detection via perspective transformer and the openlane benchmark. In *European Conference on Computer Vision*, pages 550–567. Springer, 2022.

[5] Li Chen, Penghao Wu, Kashyap Chitta, Bernhard Jaeger, Andreas Geiger, and Hongyang Li. End-to-end autonomous driving: Challenges and frontiers. *arXiv preprint arXiv:2306.16927*, 2023.

[6] Laurene Claussmann, Marc Revilloud, Dominique Gruyer, and Sébastien Glaser. A review of motion planning for highway autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 21(5):1826–1848, 2019.

[7] Dmitri Dolgov, Sebastian Thrun, Michael Montemerlo, and James Diebel. Practical search techniques in path planning for autonomous driving. *Ann Arbor*, 1001(48105):18–80, 2008.

[8] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. Carla: An open urban driving simulator. 2017.

[9] Junru Gu, Chenxu Hu, Tianyuan Zhang, Xuanyao Chen, Yilun Wang, Yue Wang, and Hang Zhao. Vip3d: End-to-end visual trajectory prediction via 3d agent queries. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5496–5506, 2023.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[11] Peiyun Hu, Aaron Huang, John Dolan, David Held, and Deva Ramanan. Safe local motion planning with self-supervised freespace forecasting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12732–12741, 2021.

[12] Shengchao Hu, Li Chen, Penghao Wu, Hongyang Li, Junchi Yan, and Dacheng Tao. St-p3: End-to-end vision-based autonomous driving via spatial-temporal feature learning. In *European Conference on Computer Vision*, pages 533–549. Springer, 2022.

[13] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17853–17862, 2023.

[14] Junjie Huang and Guan Huang. BEVDet4D: Exploit temporal cues in multi-camera 3d object detection. *arXiv preprint arXiv:2203.17054*, 2022.

[15] Linyan Huang, Zhiqi Li, Chonghao Sima, Wenhai Wang, Jingdong Wang, Yu Qiao, and Hongyang Li. Leveraging vision-centric multi-modal expertise for 3d object detection. *arXiv preprint arXiv:2310.15670*, 2023.

[16] Bo Jiang, Shaoyu Chen, Qing Xu, Bencheng Liao, Jiajie Chen, Helong Zhou, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. Vad: Vectorized scene representation for efficient autonomous driving. *arXiv preprint arXiv:2303.12077*, 2023.

[17] Tarasha Khurana, Peiyun Hu, Achal Dave, Jason Ziglar, David Held, and Deva Ramanan. Differentiable raycasting for self-supervised occupancy forecasting. In *European Conference on Computer Vision*, pages 353–369. Springer, 2022.

[18] Hongyang Li, Chonghao Sima, Jifeng Dai, Wenhai Wang, Lewei Lu, Huijie Wang, Enze Xie, Zhiqi Li, Hanming Deng, Hao Tian, et al. Delving into the devils of bird's-eye-view perception: A review, evaluation and recipe. *arXiv preprint arXiv:2209.05324*, 2022.

[19] Quanyi Li, Zhenghao Peng, Lan Feng, Qihang Zhang, Zhenghai Xue, and Bolei Zhou. Metadrive: Composing diverse driving scenarios for generalizable reinforcement learning. *IEEE transactions on pattern analysis and machine intelligence*, 45(3):3461–3475, 2022.

[20] Qi Li, Yue Wang, Yilun Wang, and Hang Zhao. Hdmapnet: An online hd map construction and evaluation framework. 2022.

[21] Yinhao Li, Zheng Ge, Guanyi Yu, Jinrong Yang, Zengran Wang, Yukang Shi, Jianjian Sun, and Zeming Li. BEVDepth: Acquisition of reliable depth for multi-view 3d object detection. *arXiv preprint arXiv:2206.10092*, 2022.

[22] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Qiao Yu, and Jifeng Dai. BEVFormer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. *arXiv preprint arXiv:2203.17270*, 2022.

[23] Zhiqi Li, Zhiding Yu, David Austin, Mingsheng Fang, Shiyi Lan, Jan Kautz, and Jose M Alvarez. Fb-occ: 3d occupancy prediction based on forward-backward view transformation. *arXiv preprint arXiv:2307.01492*, 2023.

[24] Zhiqi Li, Zhiding Yu, Wenhai Wang, Anima Anandkumar, Tong Lu, and Jose M Alvarez. Fb-bev: Bev representation from forward-backward view transformations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6919–6928, 2023.

[25] Bencheng Liao, Shaoyu Chen, Xinggang Wang, Tianheng Cheng, Qian Zhang, Wenyu Liu, and Chang Huang. Maptr: Structured modeling and learning for online vectorized hd map construction. *arXiv preprint arXiv:2208.14437*, 2022.

[26] Xuewu Lin, Tianwei Lin, Zixiang Pei, Lichao Huang, and Zhizhong Su. Sparse4d v2: Recurrent temporal fusion with sparse model. *arXiv preprint arXiv:2305.14018*, 2023.

[27] Haisong Liu, Yao Teng, Tao Lu, Haiguang Wang, and Limin Wang. Sparsebev: High-performance sparse 3d object detection from multi-camera videos. In *Proceedings of the*

IEEE/CVF International Conference on Computer Vision, pages 18580–18590, 2023.

[28] Yingfei Liu, Junjie Yan, Fan Jia, Shuailin Li, Qi Gao, Tiancai Wang, Xiangyu Zhang, and Jian Sun. PETRv2: A unified framework for 3d perception from multi-camera images. *arXiv preprint arXiv:2206.01256*, 2022.

[29] Yicheng Liu, Tianyuan Yuan, Yue Wang, Yilun Wang, and Hang Zhao. Vectormapnet: End-to-end vectorized hd map learning. In *International Conference on Machine Learning*, pages 22352–22369. PMLR, 2023.

[30] Jinhyung Park, Chenfeng Xu, Shijia Yang, Kurt Keutzer, Kris Kitani, Masayoshi Tomizuka, and Wei Zhan. Time will tell: New outlooks and a baseline for temporal multi-view 3d object detection. *arXiv preprint arXiv:2210.02443*, 2022.

[31] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *ECCV*, 2020.

[32] Aditya Prakash, Kashyap Chitta, and Andreas Geiger. Multi-modal fusion transformer for end-to-end autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7077–7087, 2021.

[33] Abbas Sadat, Sergio Casas, Mengye Ren, Xinyu Wu, Pranaab Dhawan, and Raquel Urtasun. Perceive, predict, and plan: Safe motion planning through interpretable semantic representations. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIII 16*, pages 414–430. Springer, 2020.

[34] Hao Shao, Letian Wang, Ruobing Chen, Hongsheng Li, and Yu Liu. Safety-enhanced autonomous driving using interpretable sensor fusion transformer. In *Conference on Robot Learning*, pages 726–737. PMLR, 2023.

[35] Ardi Tampuu, Tambet Matiisen, Maksym Semikin, Dmytro Fishman, and Naveed Muhammad. A survey of end-to-end driving: Architectures and training methods. *IEEE Transactions on Neural Networks and Learning Systems*, 33(4): 1364–1384, 2020.

[36] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

[37] Li Wang, Xinyu Zhang, Baowei Xv, Jinzhao Zhang, Rong Fu, Xiaoyu Wang, Lei Zhu, Haibing Ren, Pingping Lu, Jun Li, et al. Interfusion: Interaction-based 4d radar and lidar fusion for 3d object detection. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 12247–12253. IEEE, 2022.

[38] Ruihao Wang, Jian Qin, Kaiying Li, Yaochen Li, Dong Cao, and Jintao Xu. Bev-lanedet: An efficient 3d lane detection based on virtual camera via key-points. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1002–1011, 2023.

[39] Shihao Wang, Yingfei Liu, Tiancai Wang, Ying Li, and Xiangyu Zhang. Exploring object-centric temporal modeling for efficient multi-view 3d object detection. *arXiv preprint arXiv:2303.11926*, 2023.

[40] 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. *Advances in Neural Information Processing Systems*, 35:6119–6132, 2022.

[41] Enze Xie, Zhiding Yu, Daquan Zhou, Jonah Philion, Anima Anandkumar, Sanja Fidler, Ping Luo, and Jose M Alvarez. M^{2}BEV: Multi-camera joint 3d detection and segmentation with unified birds-eye view representation. *arXiv preprint arXiv:2204.05088*, 2022.

[42] Chenyu Yang, Yuntao Chen, Hao Tian, Chenxin Tao, Xizhou Zhu, Zhaoxiang Zhang, Gao Huang, Hongyang Li, Yu Qiao, Lewei Lu, et al. Bevformer v2: Adapting modern image backbones to bird's-eye-view recognition via perspective supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17830–17839, 2023.

[43] Tengju Ye, Wei Jing, Chunyong Hu, Shikun Huang, Lingping Gao, Fangzhen Li, Jingke Wang, Ke Guo, Wencong Xiao, Weibo Mao, et al. Fusionad: Multi-modality fusion for prediction and planning tasks of autonomous driving. *arXiv preprint arXiv:2308.01006*, 2023.

[44] Tianyuan Yuan, Yicheng Liu, Yue Wang, Yilun Wang, and Hang Zhao. Streammapnet: Streaming mapping network for vectorized online hd map construction. *arXiv preprint arXiv:2308.12570*, 2023.

[45] Jiang-Tian Zhai, Ze Feng, Jinhao 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 preprint arXiv:2305.10430*, 2023.

Is Ego Status All You Need for Open-Loop End-to-End Autonomous Driving?

Supplementary Material

A. Implementation Details

ST-P3 uses partially incorrect training and evaluation data. For the common practice, the future GT planning trajectory is generated from the ego locations of the samples in the subsequent 3 seconds. However, since one nuScenes clip is usually a 20s video, which means that the samples at the tail of the video (within 17s-20s) cannot produce a complete future trajectory, normal methods [13, 16] will perform special processing on these special samples by using masks, but ST-P3 [12] did not do this. ST-P3 mistakenly used samples from other scenes while generating GT of these tail samples, so errors occurred during training and testing. Related issue: https://github.com/OpenDriveLab/ST-P3/issues/24.

Ego Status Usage Details For UniAD (ID-1) and VAD-Base (ID-4), in order to exclude ego status from the Bird's Eye View (BEV) generation phase, we set the use_can_bus flag to False. Conversely, for UniAD (ID-3), to incorporate ego status into its planner, we adhered to the methodology used in VAD, which involves concatenating the ego status vector with the query features.

B. Metrics Details.

Collision Rate. While current methods tend to evaluate the collision rates of planned trajectories [11–13, 16, 17, 43], there are issues in both the definition and implementation of this metric in existing approaches. First of all, in open-loop end-to-end autonomous driving, other agents do not provoke a response from the ego car. Instead, they strictly adhere to their predetermined trajectories. Consequently, this leads to a bias in the calculation of collision rates. The second issue arises from the fact that the planning predictions generated by current methods consist solely of a series of trajectory points. As a consequence, in the final collision calculation, the yaw angle of the ego car is not taken into account. Instead, it is assumed to remain unchanged. This assumption leads to erroneous results, particularly in turning scenarios, as shown in Fig. 1.

There are also problems in the current implementation. The current definition of the collision rate of each single sample is:

$$CR(t) = \frac{\sum_{i=0}^{N} \mathbb{I}_i}{N}, N = t/0.5, \tag{2}$$

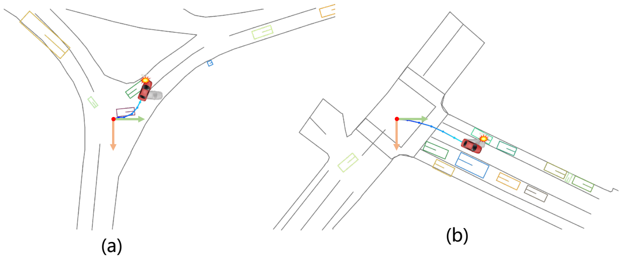


Figure 1. Current methods [12, 13, 16] neglect to consider yaw angle variations of the ego vehicle, consistently preserving a 0 yaw angle (depicted by the gray vehicle), thereby resulting in an increased incidence of false negatives (a) and false positives in (b) collision detection. In this paper, we improve collision detection accuracy by estimating the vehicle's yaw angle from variations in its trajectory (depicted by the red vehicle).

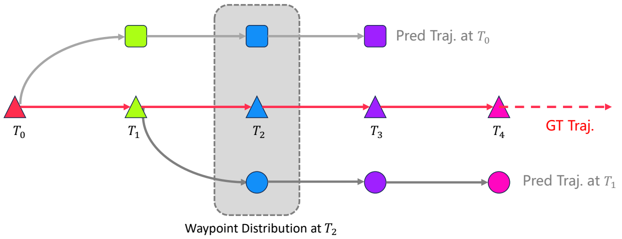


Figure 2. In open-loop autonomous driving approaches, the future trajectory is forecasted from the starting location of the ego vehicle. Within the imitation learning paradigm, the predicted trajectory ideally should closely align with the actual ground truth trajectory. Furthermore, trajectories forecasted at successive time steps should maintain consistency, thereby guaranteeing the continuity and smoothness of the driving strategy. Consequently, the predicted trajectories depicted in red boxes of Fig. 3 not only deviate from the ground truth trajectory but also demonstrate significant divergence at various timestamps.

N represents the number of steps at intervals of t seconds, and $\mathbb{I}_i$ denotes whether the ego car at step i will intersect with other agents. In this paper, we modify the definition of collision to

$$CR(t) = (\sum_{i=0}^{N} \mathbb{I}_i) > 0, N = t/0.5. \tag{3}$$

For previous implementation, they assumed that collisions at each moment were mutually independent, which does not align with real-world scenarios. Our modified version yields values that more precisely indicate the collision rate occurring along the predicted trajectory.

Trajectory Smoothness We also assessed the stability of the model's predicted trajectories. Given that the model predicts the trajectory for the next three seconds at each moment, it means that for every absolute moment in time t,

Method	L2 (m) ↓				(σ_{wd}) ↓			
	1s	2s	3s	Avg.	1s	2s	3s	Avg.
Baseline	0.30	0.52	0.85	0.56	0.03	0.19	0.70	0.31

Table 1. The smoothness σ_{wd} of predicted trajectories.

the model predicted multiple waypoints at time t from various preceding times. We see these different waypoints as a distribution. In non-extreme conditions, this distribution should be as concentrated as possible to ensure smoothness in the driving process, as shown in Fig. 2. To quantitatively analyze this distribution, we calculated the squared deviation distance of these distribution points, as shown in Tab. 1. We found that this smoothness metric does not convey more information than the L2 metric, and we believe that this requires more exploration to verify the rationality of an metric.

Valid Samples We discussed above that for the tail samples without complete GT trajectories. The normal method will use the mask for special identification. During evaluation, the previous methods have different processing methods. One is that if a sample does not have a complete GT future trajectory, it will not be considered during evaluation. The second strategy only considers the valid part of the GT future trajectory if the length of the GT trajectory is less than 3s. In this paper, we follow the first strategy. For 6019 samples of nuScenes val split, the number of final valid samples is 5119 (85% of all samples). The reason why we didn't reproduce the correct version of ST-P3 is that the definition of valid samples of ST-P3 is different from others. Valid samples of ST-P3 must use sufficient historical data, so ST-P3 does not predict trajectories for the first few samples of each clip. Even if we reproduce the correct ST-P3, we cannot compare it with other methods for a fair comparison.

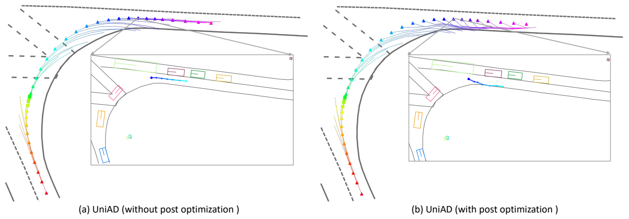

(a) UniAD (without post optimization)
(b) UniAD (with post optimization)

Figure 3. In order to avoid collisions as much as possible, post-optimization is introduced in UniAD [13] to keep the predicted trajectory away from other vehicles. However, during actual traffic driving, other factors need to be considered, such as road conditions. As shown in figure (b), UniAD rushed to the road boundary in order to avoid the possible danger caused by the opposite lane and actually caused another accident.

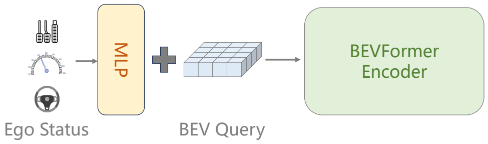


Figure 4. BEVFormer incorporates ego status information during the initialization of BEV queries, a nuance not addressed by current end-to-end autonomous driving approaches [13, 16, 43].

C. Neglected Ego Status in Perception Stage.

In fact, a crucial question is whether the method really completely eliminates the influence of ego status. The pipeline of the existing open-loop end-to-end autonomous driving methods [12, 13, 16, 43] basically follows the Fig. 1 (b). Given that ego status exerts a substantial influence on the planning results, these methods actually have clear explanations on whether to introduce ego status in the planner. However, methods [13, 16] ignored the impact of introducing ego status in the early perception stage on the planning results. In detail, both UniAD [13] and VAD [16] utilize BEVFormer [22] as their BEV generation module. For BEVFormer, it involves projecting the ego status onto the hidden features and incorporating it into the BEV query, as shown in Fig. 4. This trick exerts a marginal effect on perception performance, as shown in Tab. 2. However, when BEVFormer is integrated into an end-to-end pipeline, the introduction of ego status at this initial stage can wield a substantial influence on the ultimate planning performance. As shown in Tab. 1, upon the removal of the ego status input during the BEV stage, the planning performance of both VAD and UniAD exhibits a marked decline. It is important to clarify that our position is not opposed to the use of ego status; rather, we argue that within the context of current datasets and evaluation metrics, the integration of ego status can significantly impact, and even determine, the planning results. Unfortunately, the incorporation of ego status within the perception module is often overlooked in the existing end-to-end autonomous driving methods. Therefore, it is essential in comparative analyses of different methodologies to carefully examine the role and impact of ego status to ensure fairness and consistency in the evaluations.

Methods	Ego Status	mAP↑	NDS↑
BEVFormer	✓	41.6	51.7
BEVFormer	✗	41.3	51.5

Table 2. The integration of ego status within BEVFormer exerts only a marginal effect on the perception performance.

D. Post Optimization of UniAD.

As demonstrated in Fig. 3, we observe that while UniAD utilizes collision optimization, the resulting optimized trajectory tends to intersect with the road boundary at a higher rate. This occurs because the collision optimizer overlooks map priors. In its effort to avoid collisions, the optimizer disregards other factors that could pose safety risks. However, if the optimizer were to consider all relevant factors, it would more closely resemble traditional Planning and Navigation Control (PNC) systems, contradicting the fundamental motivation of end-to-end autonomous driving.

E. Dropping Cameras

Referencing Table 2 in our main paper, it is observed that when VAD incorporates ego status as an input, the removal of camera input does not markedly impair its performance. A parallel experiment was conducted with VAD [16] devoid of ego status. We also provide visualization results in Fig. 5. As delineated in Tab. 3, excluding camera inputs in VAD without ego status leads to a significant decline in performance, particularly regarding L2 distance and collision rate metrics. Intriguingly, this decrease was not mirrored in the Intersection rate with road boundary metric.

In fact, when the model operates without using ego status and with the camera input removed, it relies solely on driving commands to guide its future direction. In this scenario, the fact that the intersection rate with road boundaries does not increase is counter-intuitive. This counter-intuitive phenomenon has driven us to delve deeper into the evaluation process. As shown in Tab. 4, we compiled statistics on the intersection metric under different driving commands. The Intersection-LR metrics show that the model, when operating without camera input, significantly increases the probability of interacting with boundaries in turning scenarios. This is also consistent with our observations from visualization. The real reason lies in the fact that in straight-driving scenarios (87% of all evaluation samples), removing the camera input leads the model to adopt a relatively conservative straight-driving strategy, making it less likely to intersect with road boundaries (indicated by Intersection-ST). Since straight-driving scenarios constitute a large proportion of the *val* split, this results in the model achieving better overall average results when operating without camera input.

Failure Cases Although the majority of scenarios in the nuScenes dataset are relatively straightforward, it does include certain challenging scenes, notably those involving continuous cornering. As shown in Fig. 6, we can observe that methods with various settings all yielded suboptimal predicted trajectories when navigating high-curvature bends. For challenging scenarios like cornering, where the system must continuously make evolving decisions, evaluating open-loop autonomous driving systems poses a significant challenge. One limitation of open-loop methods is that they do not suffer from cumulative errors. In detail, in the case of an extremely erroneous trajectory predicted at a given timestep, the trajectory starting point for the next timestep is still based on the GT trajectory. The metric we utilize, CCR, is adept at identifying low-quality trajectories. However, an appropriate metric that can effectively highlight high-quality trajectories remains an intriguing direction for further exploration.

Method	Img Corruption	Ego Status	L2 (m) ↓				Collision (%) ↓				Intersetion (%) ↓				Det. (NDS)	Map (mAP)
			1s	2s	3s	Avg.	1s	2s	3s	Avg.	1s	2s	3s	Avg.		
VAD-Base	-	✓	**0.17**	**0.34**	**0.60**	**0.37**	0.04	**0.27**	**0.67**	**0.33**	**0.21**	**2.13**	**5.06**	**2.47**	**45.5**	47.0
VAD-Base	Blank	✓	0.19	0.41	0.77	0.46	**0.00**	0.40	1.21	0.54	0.35	3.05	7.73	3.71	0.0	0.0
VAD-Base	-	✗	0.69	1.22	1.83	0.06	0.68	2.52	0.84	0.37	1.02	3.44	7.00	3.82	45.1	**53.7**
VAD-Base	Blank	✗	2.59	4.32	6.09	4.33	2.29	7.89	12.7	7.63	1.07	3.73	6.64	3.81	0.0	0.0

Table 3. Omitting camera inputs in the VAD model, when it does not utilize ego status, results in a marked reduction in performance, as evidenced by the metrics for L2 distance and collision rate.

Method	Img Corruption	Ego Status	CCR ↓				CCR-ST (%) ↓				CCR-LR (%) ↓				Det. (NDS)	Map (mAP)
			1s	2s	3s	Avg.	1s	2s	3s	Avg.	1s	2s	3s	Avg.		
VAD-Base	-	✗	1.02	3.44	7.00	3.82	2.49	8.50	16.4	9.13	0.95	2.70	5.50	3.05	45.1	53.7
VAD-Base	Blank	✗	1.07	3.73	6.64	3.81	2.63	18.2	32.1	17.6	0.83	1.51	2.72	1.69	0.0	0.0

Table 4. CCR-ST is the CCR rate with going straight driving commands. CCR-LR is the CCR rate with turning left/right commands.

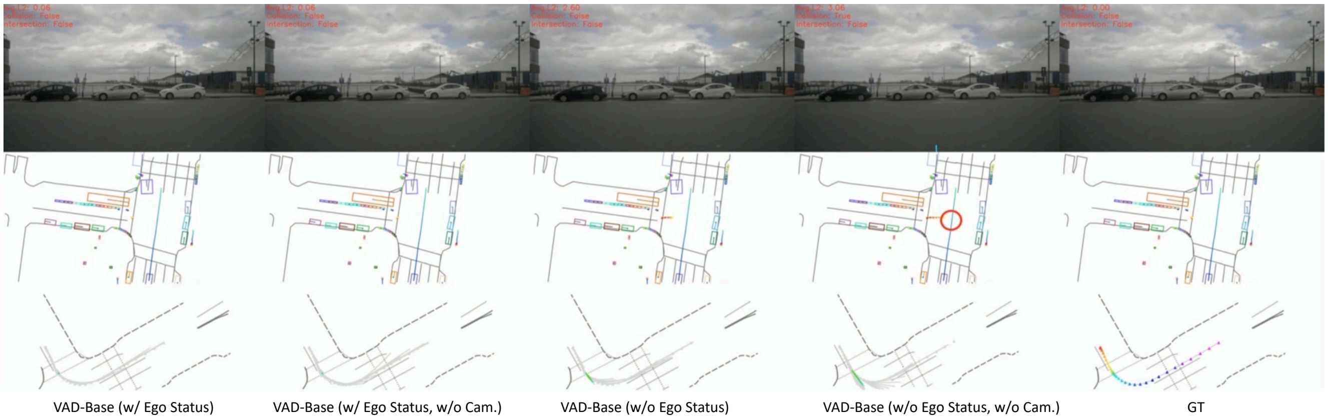
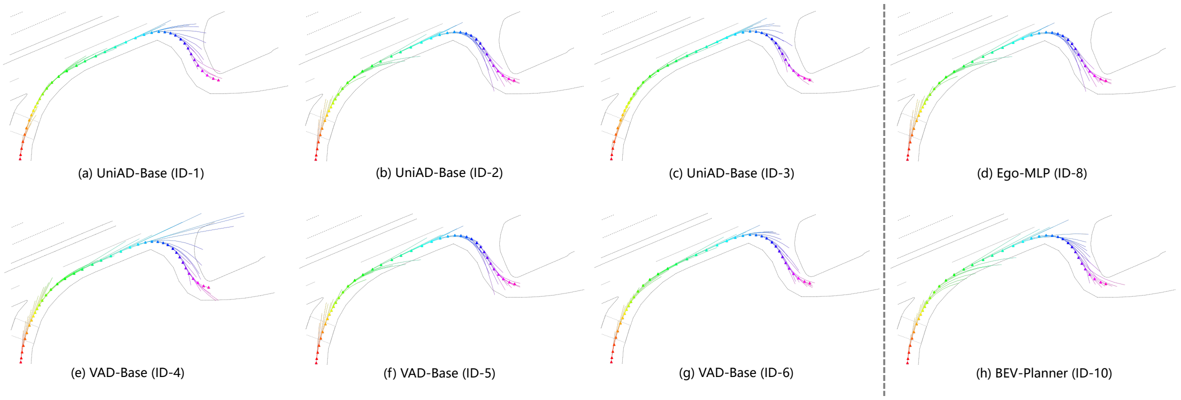

VAD-Base (w/ Ego Status)
VAD-Base (w/ Ego Status, w/o Cam.)
VAD-Base (w/o Ego Status)
VAD-Base (w/o Ego Status, w/o Cam.)
GT

Figure 5. When the model uses ego status as an input, removing the camera does not significantly impact its performance. However, without ego status, omitting camera inputs makes the model more prone to erroneous planning. Red circles indicate potential collisions.


(a) UniAD-Base (ID-1)
(b) UniAD-Base (ID-2)
(c) UniAD-Base (ID-3)
(d) Ego-MLP (ID-8)
(e) VAD-Base (ID-4)
(f) VAD-Base (ID-5)
(g) VAD-Base (ID-6)
(h) BEV-Planner (ID-10)

t

Figure 6. In scenarios that necessitate continuous turning, all methods predict suboptimal trajectories.