

Learning to Drive by Imitating Surrounding Vehicles

*Yasin Sonmez
Hanna Krasowski
Murat Arca*



Electrical Engineering and Computer Sciences
University of California, Berkeley

Technical Report No. UCB/EECS-2025-14

<http://www2.eecs.berkeley.edu/Pubs/TechRpts/2025/EECS-2025-14.html>

April 29, 2025

Copyright © 2025, by the author(s).
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

Learning to Drive by Imitating Surrounding Vehicles

by Yasin Sonmez

Research Project

Submitted to the Department of Electrical Engineering and Computer Sciences,
University of California at Berkeley, in partial satisfaction of the requirements for
the degree of **Master of Science, Plan II**.

Approval for the Report and Comprehensive Examination:

Committee:

Murat Arcak

Professor Murat Arcak
Research Advisor

4/29/2025

(Date)

* * * * *

Kameshwar Poolla

Professor Kameshwar Poolla
Second Reader

4/29/2025

(Date)

Learning to Drive by Imitating Surrounding Vehicles

Yasin Sonmez, Hanna Krasowski, and Murat Arcaak
Department of EECS, UC Berkeley
{yasin.sonmez, krasowski, arcak}@berkeley.edu

Abstract: Imitation learning is a promising approach for training autonomous vehicles (AV) to navigate complex traffic environments by mimicking expert driver behaviors. While existing imitation learning frameworks focus on leveraging expert demonstrations, they often overlook the potential of additional complex driving data from surrounding traffic participants. In this paper, we propose a data augmentation strategy that enhances imitation learning by leveraging the observed trajectories of nearby vehicles, captured by the AV’s sensors, as additional expert demonstrations. We introduce a vehicle selection sampling strategy that prioritizes informative and diverse driving behaviors, contributing to a richer and more diverse dataset for training. We evaluate our approach using the state-of-the-art learning-based planning method PLUTO on the nuPlan dataset and demonstrate that our augmentation method leads to improved performance in complex driving scenarios. Specifically, our method reduces collision rates and improves safety metrics compared to the baseline. Notably, even when using only 10% of the original dataset, our method matches or exceeds the performance of the full dataset, with improved collision rates. Our findings highlight the importance of leveraging diverse real-world trajectory data in imitation learning and provide insights into data augmentation strategies for autonomous driving.

Keywords: Autonomous driving, imitation learning, planning

1 Introduction

By learning from expert demonstrations, imitation learning enables autonomous vehicles (AVs) to develop policies that mimic human-like driving behavior. Recently, imitation learning models [1, 2] have started to outperform traditional rule-based methods [3] on benchmarks with large-scale real-world data such as nuPlan [4], indicating the increasing viability of imitation learning for real-world deployment. However, imitation learning also suffers from three major challenges. First, recent studies demonstrate that imitation learning models can learn shortcuts from data [5], leading to undesired behaviors. For example, it has been demonstrated that models with historical AV motion data excel in open-loop evaluation but underperform in closed-loop metrics, likely due to learning shortcuts [6]. Second, imitation learning suffers from the distribution shift problem, where the training and test sets have different distributions due to the nature of the problem, such as learning from data collected in one location and deploying elsewhere. To address this challenge, several studies suggest that imitation learning benefits from reinforcement learning refinements [7]. Lastly, imitation learning suffers from causal confusion [8] when a model learns spurious correlations instead of true causal relationships between actions and outcomes. Since imitation learning relies on mimicking expert demonstrations, the model may pick up on irrelevant features or unintended cues that correlate with successful behavior but do not actually cause it.

Addressing these challenges through effective data augmentation, model architecture, and loss choices is crucial for improving real-world performance. As such, it is crucial to maximize the utility of the data collected through various means. Despite the availability of large datasets, simulators, and benchmarks (e.g., [4, 9, 10]), effectively utilizing this data for imitation learning remains a challenge. Different datasets capture driving information at varying levels of abstraction, ranging

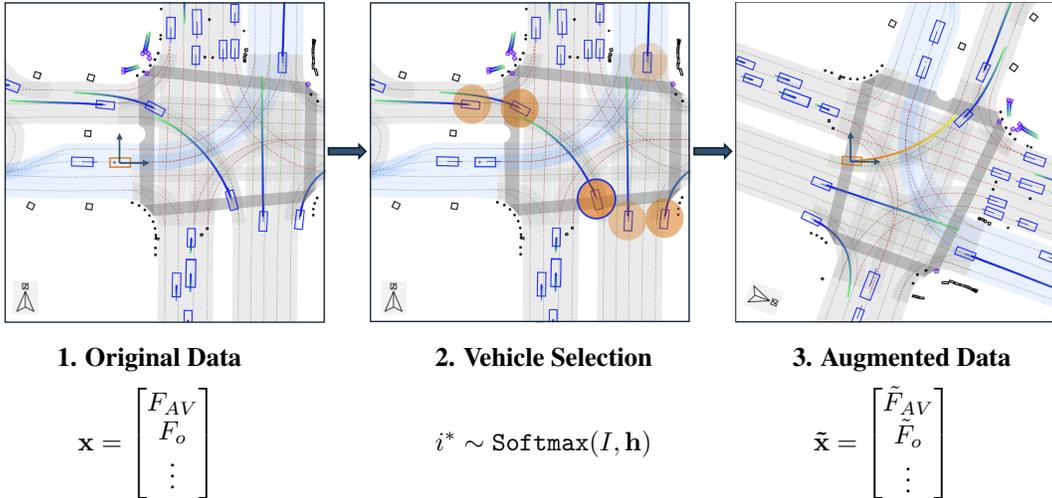


Figure 1: Data augmentation framework illustrated for a traffic scenario: (1) The expert driver remains stationary at a red light while surrounding vehicles follow diverse trajectories. (2) A suitable surrounding vehicle i^* (shown with a blue circle) is sampled from the weighted categorical distribution defined in Eq. 3, where darker circles represent higher selection probabilities and \mathbf{h} is the vector of sum of absolute heading deviations for all vehicles I . (3) A reference frame transformation generates features from the perspective of the new ego vehicle.

from object-level annotations to raw sensor images. Furthermore, recent studies have indicated that simply increasing the volume of training data does not necessarily result in improved model performance. For example, [11] highlights that the addition of more data may not always translate to better outcomes, suggesting that other factors, such as data quality and relevance, play a more significant role in model effectiveness.

In this paper, we propose a data augmentation technique that enhances imitation learning by incorporating trajectories beyond those of expert drivers in driving datasets. Specifically, we build on the success of PLUTO [1] and introduce a new training methodology that harnesses trajectory data from nearby vehicles as expert demonstrations. To this end, we introduce a traffic participant selection criterion that prioritizes informative and diverse driving trajectories from observed vehicles. We validate our method through ablation studies using state-of-the-art imitation learning models, including PLUTO [1] and the nuPlan benchmark [4], demonstrating the effectiveness of our augmentation strategy in improving autonomous driving. The main contributions of this work are:

- We introduce a vehicle selection mechanism for data augmentation based on heading deviation using a softmax-weighted sampling strategy to focus on dynamic and contextually rich scenarios.
- We propose a dynamic behavior-driven data augmentation technique for object-based expert driving datasets, prioritizing informative and diverse driving trajectories to demonstrate that augmenting the dataset with trajectories from surrounding agents improves performance, particularly in low-data regimes, achieving competitive performance with only 10% of the original dataset.
- We perform experiments using the state-of-the-art PLUTO planner and the nuPlan dataset, including ablation studies on dataset size, the number of augmented vehicles, and the impact of different sampling temperatures.

2 Related Work

Datasets for autonomous driving: Although traffic data is tedious and costly to collect, there is an increasing amount of open-source datasets for autonomous driving research. These datasets can be

broadly categorized into perception (i.e., sensor-based [12, 13, 14]) and motion planning (i.e., object-based [15, 16, 4, 17]) datasets. Since imitation learning is commonly used for motion planning tasks, we focus on object-based datasets, which have the additional advantage of allowing for interpretable and complete data augmentation. The object-based dataset NGSIM [18] is one of the earlier large datasets initially published in 2006. Since then many new datasets with increasing size and traffic complexity have been published. In particular, the nuPlan benchmark [19, 4] consists of real-world autonomous driving datasets and evaluation frameworks. nuPlan offers a comprehensive dataset for both prediction and planning, with 1282 hours of driving data from four cities, and introduces a taxonomy of driving scenarios. Due to these features, nuPlan has been used to compare various planning approaches in the literature such as [20, 3, 21].

Data augmentation for autonomous driving: Although above datasets are of increasing size, data augmentation can significantly enhance their value. For example, Guo et al. [22] develops context-aware data augmentation for imitation learning that is based on a variational autoencoder. Another interesting approach for sensor-based datasets is proposed by Chen and Krähenbühl [23]. By learning from the trajectories of all vehicles observed by the ego-vehicle, the system effectively increases sample efficiency and exposes the model to a wider variety of safety-critical and complex driving scenarios. A key challenge in this approach is the partial observability of surrounding vehicles. This is solved by a perception module that generates a viewpoint-invariant 2D top-down representation of the scene, helping the motion planner generalize across different vehicles. In contrast, we propose a data augmentation for object-based datasets that uses the more complex trajectories of other drivers by introducing a biased sampling criteria and data augmentation. PLUTO [1] employs contrastive imitation learning (CIL) to address distribution shift by applying both positive and negative data augmentations, where positive augmentations agree with the ground truth and negative augmentations intentionally disagree.

Imitation learning for autonomous driving: The two main learning approaches for autonomous driving are reinforcement learning and imitation learning. Reinforcement learning usually relies on a realistic simulation environment and significant reward-shaping to achieve performant driving policies [24]. Imitation learning is usually easier to tune but requires a diverse and large dataset of driving trajectories to achieve expert-like driving behavior [25, 26]. Leading companies in autonomous driving, such as Tesla and Waymo [27, 7], as well as open-source projects like OpenPilot [28], leverage imitation learning to train models by mimicking expert driving behavior. A recent example is the work by Zheng et al. [2], who propose a transformer-based Diffusion Planner for closed-loop planning, capable of modeling multi-modal driving behavior. Another notable framework in imitation learning is PLUTO [1], which introduces key innovations for more efficient driving behavior generation: a longitudinal-lateral aware transformer architecture, contrastive learning to mitigate causal confusion and distribution shift, and ego-related data augmentation. In this study, we use PLUTO as a baseline and perform ablation studies on the nuPlan dataset.

3 Methodology

3.1 Problem Formulation

Our method is a general data augmentation approach applicable to object-based planning frameworks. In this study, we integrate our augmentation into PLUTO [1], a state-of-the-art learning-based planning method, to demonstrate its effectiveness. Below, we briefly formulate the planning problem from the perspective of PLUTO.

As in PLUTO [1], we consider an AV, N_A dynamic agents, N_O static obstacles, a high-definition map M , and traffic context information C (e.g., traffic light status). The feature set for dynamic agents is denoted as $\mathcal{A} = A_{0:N_A}$, where A_0 represents the AV, and the static obstacle set is $\mathcal{O} = O_{1:N_O}$. The future state of agent a at time t is denoted as \mathbf{y}_a^t , with historical and future horizons of T_H and T_F , respectively. PLUTO generates N_T multi-modal planning trajectories for the AV along with predictions for each dynamic agent. The final trajectory τ^* is selected via a scoring module S ,

which integrates these outputs with the scene context. The overall formulation is given as:

$$(\mathbf{T}_0, \pi_0), \mathbf{P}_{1:N_A} = f(A, \mathcal{O}, M, C \mid \phi) \quad (1)$$

$$(\tau^*, \pi^*) = \arg \max_{(\tau, \pi) \in (\mathbf{T}_0, \pi_0)} S(\tau, \pi, \mathbf{P}_{1:N_A}, \mathcal{O}, M, C), \quad (2)$$

where f represents PLUTO’s neural network, ϕ are the model parameters, $(\mathbf{T}_0, \pi_0) = \{(\mathbf{y}_{0,i}^{1:T_F}, \pi_i) \mid i = 1, \dots, N_T\}$ are the generated planning trajectories with confidence scores, and $\mathbf{P}_{1:N_A} = \{\mathbf{y}_a^{1:T_F} \mid a = 1, \dots, N_A\}$ are the predicted future states of dynamic agents.

Agent History Encoding: PLUTO represents each agent’s state at time t as $\mathbf{s}_i^t = (\mathbf{p}_i^t, \theta_i^t, \mathbf{v}_i^t, \mathbf{b}_i^t, I_i^t)$, where $\mathbf{p}_i^t \in \mathbb{R}^2$ and $\theta_i^t \in \mathbb{R}$ denote position and heading, $\mathbf{v}_i^t \in \mathbb{R}^2$ represents velocity, and $\mathbf{b}_i^t \in \mathbb{R}^2$ and $I_i^t \in \{0, 1\}$ correspond to the bounding box dimensions and observation status, respectively. The temporal evolution of agent states is captured by computing differences between consecutive timesteps, resulting in a feature matrix $\mathbf{F}_A \in \mathbb{R}^{N_A \times (T_H - 1) \times 8}$.

Static Obstacles Encoding: Static obstacles in the drivable area are encoded as $\mathbf{o}_i = (\mathbf{p}_i, \theta_i, \mathbf{b}_i)$, producing a feature matrix $\mathbf{F}_O \in \mathbb{R}^{N_O \times 5}$.

AV’s State Encoding: Imitation learning models often develop shortcuts based on historical states, which can degrade performance [6, 29]. To mitigate this, only the current state of the AV is used as input features without using the history. These include the AV’s position, heading angle, velocity, acceleration, and steering angle, represented as $\mathbf{F}_{AV} \in \mathbb{R}^{1 \times 8}$.

Vectorized Map Encoding: The map consists of N_p polylines, each undergoing an initial sub-sampling step to standardize the number of points. Feature vectors are then computed for each polyline point. Specifically, for the i -th point of a polyline, the feature vector consists of $(\mathbf{p}_i - \mathbf{p}_0, \mathbf{p}_i - \mathbf{p}_{i-1}, \mathbf{p}_i - \mathbf{p}_i^{\text{left}}, \mathbf{p}_i - \mathbf{p}_i^{\text{right}})$ where \mathbf{p}_0 is the initial point of the polyline, and $\mathbf{p}_i^{\text{left}}$ and $\mathbf{p}_i^{\text{right}}$ represent the left and right lane boundary points, respectively. The final representation of the polyline features is $\mathbf{F}_P \in \mathbb{R}^{N_P \times n_p \times 8}$, where n_p is the number of points per polyline.

Scene Encoding: To capture interactions between dynamic agents, static obstacles, polylines, and the autonomous vehicle, these inputs are concatenated and processed through transformer encoders, with Fourier-based positional embeddings and learnable semantic attributes compensating for the loss of global positional information.

Trajectory Planning and Post-processing: The model generates multimodal trajectories with confidence scores and employs an additional rule-based post-processing module to ensure safe and robust selection. Forward simulation utilizing a linear quadratic regulator for trajectory tracking and a kinematic bicycle model for state updates assesses rollouts based on different metrics such as driving comfort, traffic rule adherence, and time-to-collision. The final trajectory is selected by combining learning-based confidence scores with rule-based evaluations, balancing data-driven learning with human-like decision-making. This approach enhances interpretability and safety without modifying the originally planned trajectory.

3.2 Learning From Surrounding Traffic

Many real-world driving scenarios are inherently dynamic, with multiple interacting agents. While some situations may involve routine behaviors such as lane-keeping or waiting at a red light, even in these types of scenarios usually there is at least one vehicle exhibiting complex or “interesting” behaviors in the surrounding traffic. Examples include lane changes, turning at intersections, yielding to pedestrians, or reacting to bicycles. These nuanced interactions provide a rich source of data for understanding diverse driving behaviors and the decision-making processes of road users. Our approach capitalizes on this by augmenting the imitation learning dataset with estimated trajectories of selected agents from the surrounding traffic.

The key advantages of our approach are:

1- Diverse Behavioral Representation: Incorporating trajectories from surrounding vehicles enhances dataset diversity by introducing a broader range of driving behaviors. This diversity helps

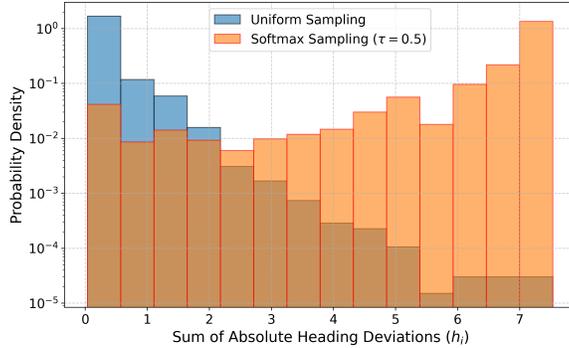


Figure 2: Histogram of the sum of absolute heading deviations h_i for observed vehicles in the dataset. The blue plot represents the original data, while the orange plot corresponds to vehicles sampled using the softmax distribution defined in Equation 3. The histogram is presented on a logarithmic scale to account for the large differences in probabilities.

mitigate model bias toward a limited subset of driving styles, improving the model’s ability to generalize across varied real-world scenarios. Prior research in imitation learning and reinforcement learning has shown that exposure to diverse state-action distributions reduces overfitting and improves robustness in novel environments [30, 31].

2- Contextual Interaction Learning: Learning from surrounding agents enables the model to capture multi-agent interactions, improving its ability to anticipate and respond to dynamic traffic scenarios. For example, Zhang et al. [32] shows that incorporating contextual interactions enhances decision-making and overall driving performance.

3- Focus on Dynamic Scenarios: Rather than relying only on the AV’s ground truth, which may often reflect relatively static or routine behaviors, our approach prioritizes learning from more dynamic and contextually rich scenarios, which are shown to increase the performance [11]. The distribution of the sum of absolute heading angle deviations, illustrated in Figure 2, further supports this point. The blue histogram indicates that the majority of observed vehicle data consists of minimal deviation movements, such as lane-keeping, simple acceleration, and deceleration, highlighting the need for incorporating more diverse motion patterns.

3.3 Vehicle Selection Criteria

Most simulation scenarios have many vehicles present, but it is infeasible to augment the dataset with all of them. Many of these vehicles may exhibit little to no interesting behavior, and including them would unnecessarily increase the computation and contribute little to the learning process. Prior research has shown that treating all training data equally can lead to suboptimal performance, especially in safety-critical situations, and that prioritizing more informative samples can improve robustness while reducing data requirements. For instance, [11] demonstrated that an imitation-learning-based planner trained on only 10% of a dataset—carefully curated using a trained scenario difficulty predictor—performed as well as one trained on the full dataset while significantly reducing collisions and improving route adherence. Therefore, vehicle selection criteria are essential to ensure the dataset contains the most beneficial scenarios.

To ensure the quality and consistency of the augmented data, we first apply a series of filtering steps to discard unsuitable candidate vehicles. Specifically, any vehicle that does not appear in all timesteps of a given scenario is removed, as its absence could be due to partial observability or sensor noise, introducing inconsistencies in the training data. Next, we discard vehicles that do not remain within a fixed radius of the AV, set at $r = 50 m$, across all timesteps. This constraint ensures that only vehicles that have reliable sensory measurements are considered, as distant vehicles are usually more noisy. Additionally, vehicles identified as being outside the drivable area—such as parked cars in driveways—are excluded, as their trajectories do not contribute to meaningful driving interactions. For the remaining pool of vehicles, we assign a weight h_i based on their sum of

absolute heading angle deviation over time. This deviation serves as a proxy for dynamic behaviors such as turns, lane changes, and parking maneuvers, which are crucial for learning diverse driving interactions. However, such behaviors are relatively rare, as illustrated in Figure 2. To sample from this distribution more effectively, we apply a softmax function:

$$h_i = \sum_t |\theta_i^t - \theta_i^{t-1}|, \quad \text{Softmax}(i) = \frac{\exp(h_i/\tau)}{\sum_j \exp(h_j/\tau)} \quad (3)$$

where h_i represents the weight for vehicle i , $\text{Softmax}(i)$ is its assigned probability, and τ is the temperature parameter that controls the sharpness of the weight distribution. A lower τ makes the selection more focused on vehicles with higher deviations, while a higher τ results in a more uniform weighting across all vehicles. Using these weights, we sample N_s distinct scenarios for each ego vehicle scenario, augmenting the dataset by incorporating the selected vehicles. This ensures that the dataset is both realistic and diverse, improving the robustness of the learned model.

Since certain vehicle-specific parameters, such as wheelbase and center of gravity, are typically unobserved in object-based datasets, we estimate these values using empirically derived approximations available in the implementation. Once a vehicle is selected, all scene features—including the positions of other objects, map polylines, and lane boundaries—are transformed into the reference frame of the selected vehicle, generating a new augmented feature representation \tilde{x} as in Figure 1.

4 Experiments

We train the baseline PLUTO planner using varying numbers of scenarios extracted from the nuPlan dataset. For each scenario, we generated N_S additional scenarios by leveraging trajectories from other vehicles. However, in some cases, no suitable vehicles were consistently observable across all time steps due to partial observability, and augmentation was not applied in those instances. Both the baseline and augmented models were trained to convergence, monitored using validation error. We used the hyperparameters reported in the original PLUTO implementation.

For evaluation, we evaluated both methods on the test14-hard benchmark, which is collected by executing 100 scenarios for each of 14 scenario types then 20 lowest-performing instances per type selected for evaluation using the state-of-the-art rule-based planner PDM-Closed [3]. We did not repeat experiments on the val14 benchmark, which consists of uniformly sampled nuPlan scenarios, as prior studies indicate a strong correlation between learning-based methods’ performance on test14-hard and val14 as seen in Figure 5 in the Appendix B and computation time for val14 is much higher than test14-hard. The test14-hard dataset captures the most challenging and informative scenarios.

The nuPlan framework provides a comprehensive evaluation score for each simulation, incorporating key metrics such as (1) **No Ego At-Fault Collisions**, where only AV-initiated collisions are considered; (2) **TTC (Time-to-Collision) Compliance**, ensuring time-to-collision remains above a threshold; (3) **Drivable Area Compliance**, requiring the AV to stay within road boundaries; (4) **Comfort**, assessed via acceleration, jerk, and yaw dynamics within empirical thresholds; (5) **Progress**, measured as the AV’s traveled distance relative to the expert driver. We use the non-reactive closed-loop score as our performance evaluation metric.

We conducted experiments across datasets of varying sizes, specifically with 1K, 10K, and 100K scenarios. As shown in Table 1, our data augmentation method consistently outperforms the baseline across all dataset sizes, achieving performance improvements of 7.13%, 10.22%, and 2.57%, respectively. A deeper analysis of individual metrics reveals the most significant enhancements in collision and time-to-collision (TTC), indicating that our method substantially reduces the likelihood of collisions and enhances safety. Notably, even when using only 10% of the original dataset, our augmented approach achieves performance comparable to using the full dataset, while outperforming the baseline in terms of both collision rate and TTC. This highlights the efficiency and effectiveness of our data augmentation strategy in improving both safety and model performance.

This improvement is exemplified in a challenging scenario depicted in Figure 3, where a vehicle attempts to merge onto the road from the right. The baseline model exhibits hesitation, waiting for

Scenarios	Planner	Score	Collisions	TTC	Drivable	Comfort	Progress
1K	PLUTO (Baseline)	58.60	83.02	76.23	92.45	70.94	71.20
	PLUTO (Ours, $\tau = 0.5, N_s = 2$)	65.73	89.84	82.52	93.90	71.54	75.20
10K	PLUTO (Baseline)	61.95	83.90	74.91	93.26	76.03	78.72
	PLUTO (Ours, $\tau = 0.5, N_s = 1$)	72.17	91.29	82.58	95.08	80.68	80.61
100K	PLUTO (Baseline)	74.81	91.23	83.96	97.01	86.57	78.64
	PLUTO (Ours, $\tau = 0.5, N_s = 1$)	77.38	93.75	84.09	96.97	87.88	80.47

Table 1: Performance comparison of PLUTO planner under base and augmented datasets with varying number of scenarios (1K, 10K, 100K) and different parameters. The Test14-hard dataset is used for evaluation. All scores are between 0 and 100, where the higher the score, the better. More results are available in Appendix C.

the merging vehicle to proceed. However, this indecisiveness results in a collision, as the baseline fails to anticipate the interaction effectively. In contrast, our augmented model demonstrates a more confident decision-making process, correctly interpreting the traffic dynamics to take the lead while adhering to traffic regulations. This underscores the effectiveness of our approach in improving driving performance, particularly in high-stakes situations that demand adaptive behavior.

4.1 Ablation Studies

Effect of Dataset Size: We conducted ablation studies on dataset sizes of 1K, 10K, and 100K scenarios, as shown in Figure 4. The results indicate that increasing the dataset size improves performance for both the baseline and our proposed method. Notably, the data augmentation method consistently enhances performance across all dataset sizes, with the most significant gains observed at 10K scenarios. This trend aligns with the expectation that larger datasets generally lead to better generalization and improved model robustness.

Impact of the Number of Selected Vehicles N_s : To analyze the effect of augmenting with multiple vehicles, we experimented with selecting 1 or 2 additional vehicles per AV scenario and compared the results against the baseline (no augmentation). In the 1K scenario setting, performance improved when augmenting with 2 vehicles, whereas in the 10K scenario dataset, the best performance was achieved with $N_s = 1$. This suggests that while data augmentation is beneficial, excessive augmentation beyond a certain point does not provide further improvements. The results indicate that

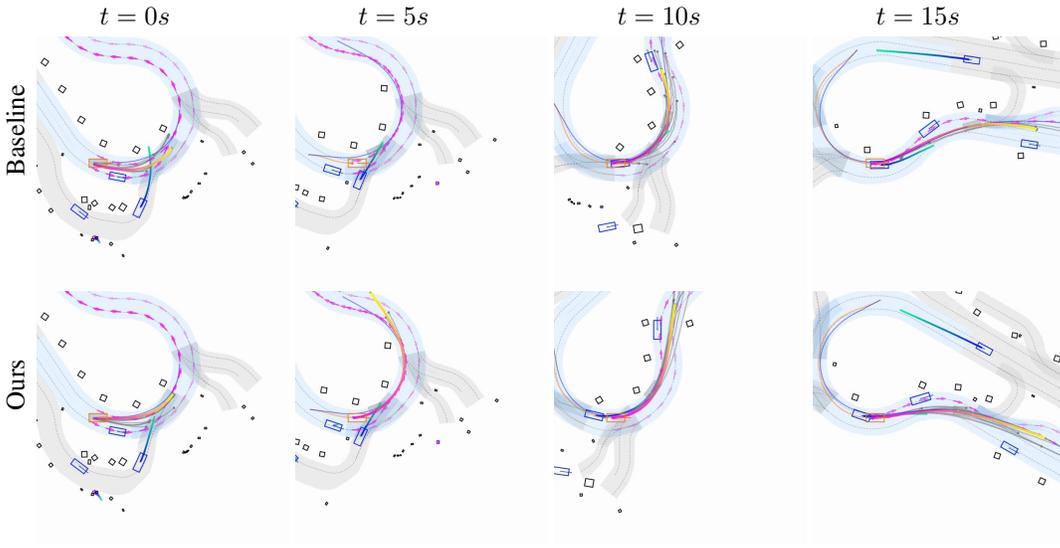


Figure 3: Comparison of the baseline and our method in a merging scenario. The baseline hesitates and collides, while our method confidently avoids the collision.

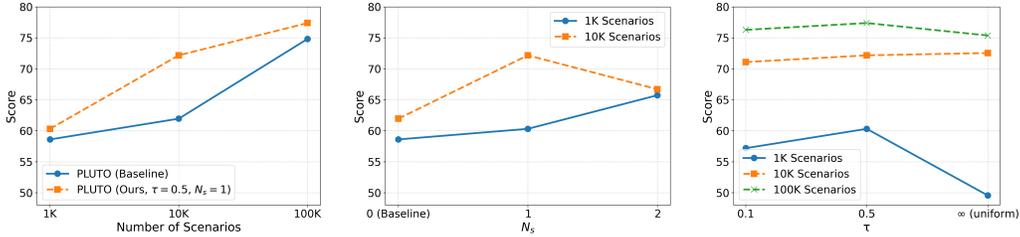


Figure 4: Ablation experiments. (1) Dataset size. (2) Number of selected vehicles N_s . (3) Temperature parameter τ .

in low-data regimes, increasing the amount of augmented data is advantageous, but as dataset size grows, additional augmentation yields diminishing improvements.

Influence of the Temperature Parameter τ : The temperature parameter τ in the softmax distribution controls sampling bias toward vehicles with higher heading deviations. Lower τ focuses on dynamic vehicles, while higher τ leads to more uniform sampling. We tested $\tau = 0.1$, $\tau = 0.5$, and uniform sampling ($\tau \rightarrow \infty$). In the 1K dataset, $\tau = 0.5$ outperformed others and even exceeded the 10K baseline only using 10% of it, highlighting the benefit of selective augmentation with limited data. However, uniform sampling degraded performance. For 10K and 100K datasets, all τ values performed similarly, though higher τ was slightly better for 10K and $\tau = 0.5$ for 100K. This suggests τ selection is crucial in low-data regimes but less impactful with larger datasets.

5 Conclusion

We proposed a novel data augmentation strategy for imitation learning in autonomous driving by leveraging the observed trajectories of surrounding traffic participants as additional expert demonstrations. Our approach introduces vehicle selection criteria based on heading deviation, prioritizing dynamic and contextually rich driving behaviors to enrich the training dataset.

We evaluated our method using the PLUTO planner and the nuPlan dataset, demonstrating that the augmentation strategy consistently outperforms the baseline across varying dataset sizes. In particular, we observed substantial improvements in safety-critical metrics such as collision rates and time-to-collision (TTC), especially in low-data regimes, where our method achieved competitive or superior performance using only 10% of the original dataset. We further conducted ablation experiments to investigate the effects of different parameters.

Beyond autonomous driving, the underlying principle of leveraging observed agent interactions for improved decision-making could extend to other robotics tasks and less structured multi-agent environments, such as aerial traffic or maritime navigation, where large-scale expert demonstrations are difficult to obtain. Our results reinforce the importance of exploiting diverse real-world trajectory data in imitation learning and offer a promising direction for enhancing data efficiency and safety in autonomous decision-making.

6 Limitations

While our approach shows strong improvements, several limitations remain. First, our method assumes that all selected vehicles provide valuable learning signals. However, some observed agents may exhibit suboptimal or unsafe driving behaviors, potentially introducing noise into the training data. Future work could explore sophisticated driver classification techniques, such as temporal logic-based filtering [33], to prioritize learning from high-quality expert-like demonstrations while filtering out unsafe behaviors.

Second, although we demonstrated the effectiveness of our approach on the highly diverse nu-Plan dataset, further investigation is needed across other datasets and planning algorithms to assess the generalizability of the observed performance gains. Applying our method to perception-based datasets would also require an additional preprocessing step to generate object-level representations [34, 35], which may introduce new challenges related to sensor noise and partial observability.

Lastly, the benefits of our method might vary depending on the baseline model architecture and the specific augmentation parameters, such as the vehicle sampling strategy and softmax temperature. Exploring adaptive augmentation policies and dataset curation strategies could further improve robustness and scalability in future work.

References

- [1] J. Cheng, Y. Chen, and Q. Chen. Pluto: Pushing the limit of imitation learning-based planning for autonomous driving, 2024. URL <https://arxiv.org/abs/2404.14327>.
- [2] Y. Zheng, R. Liang, K. Zheng, J. Zheng, L. Mao, J. Li, W. Gu, R. Ai, S. E. Li, X. Zhan, et al. Diffusion-based planning for autonomous driving with flexible guidance. *arXiv preprint arXiv:2501.15564*, 2025.
- [3] D. Dauner, M. Hallgarten, A. Geiger, and K. Chitta. Parting with misconceptions about learning-based vehicle motion planning. In *Conference on Robot Learning*, pages 1268–1281. PMLR, 2023.
- [4] H. Caesar, J. Kabzan, K. S. Tan, W. K. Fong, E. Wolff, A. Lang, L. Fletcher, O. Beijbom, and S. Omari. nuplan: A closed-loop ml-based planning benchmark for autonomous vehicles. *arXiv preprint arXiv:2106.11810*, 2021.
- [5] B. Jaeger, K. Chitta, and A. Geiger. Hidden biases of end-to-end driving models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8240–8249, 2023.
- [6] J. Cheng, Y. Chen, X. Mei, B. Yang, B. Li, and M. Liu. Rethinking imitation-based planners for autonomous driving. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 14123–14130. IEEE, 2024.
- [7] Y. Lu, J. Fu, G. Tucker, X. Pan, E. Bronstein, R. Roelofs, B. Sapp, B. White, A. Faust, S. Whiteson, et al. Imitation is not enough: Robustifying imitation with reinforcement learning for challenging driving scenarios. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7553–7560. IEEE, 2023.
- [8] P. De Haan, D. Jayaraman, and S. Levine. Causal confusion in imitation learning. *Advances in neural information processing systems*, 32, 2019.
- [9] C. Gulino, J. Fu, W. Luo, G. Tucker, E. Bronstein, Y. Lu, J. Harb, X. Pan, Y. Wang, X. Chen, et al. Waymax: An accelerated, data-driven simulator for large-scale autonomous driving research. *Advances in Neural Information Processing Systems*, 36:7730–7742, 2023.
- [10] D. Dauner, M. Hallgarten, T. Li, X. Weng, Z. Huang, Z. Yang, H. Li, I. Gilitschenski, B. Ivanovic, M. Pavone, et al. Navsim: Data-driven non-reactive autonomous vehicle simulation and benchmarking. *Advances in Neural Information Processing Systems*, 37:28706–28719, 2025.
- [11] E. Bronstein, S. Srinivasan, S. Paul, A. Sinha, M. O’Kelly, P. Nikdel, and S. Whiteson. Embedding synthetic off-policy experience for autonomous driving via zero-shot curricula. In K. Liu, D. Kulic, and J. Ichnowski, editors, *Proceedings of The 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pages 188–198. PMLR, 14–18 Dec 2023. URL <https://proceedings.mlr.press/v205/bronstein23a.html>.
- [12] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The KITTI dataset. *International Journal of Robotics Research (IJRR)*, 2013.
- [13] M.-F. Chang, J. W. Lambert, P. Sangkloy, J. Singh, S. Bak, A. Hartnett, D. Wang, P. Carr, S. Lucey, D. Ramanan, and J. Hays. Argoverse: 3D tracking and forecasting with rich maps. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
- [14] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom. nuScenes: A multimodal dataset for autonomous driving. *arXiv:1903.11027*, 2019.

- [15] R. Krajewski, J. Bock, L. Kloeker, and L. Eckstein. The highD Dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems. In *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, pages 2118–2125, 2018. doi:10.1109/ITSC.2018.8569552.
- [16] W. Zhan, L. Sun, D. Wang, H. Shi, A. Clause, M. Naumann, J. Kümmerle, H. Königshof, C. Stiller, A. de La Fortelle, and M. Tomizuka. INTERACTION Dataset: An INTERNATIONAL, Adversarial and Cooperative moTION Dataset in Interactive Driving Scenarios with Semantic Maps. *arXiv:1910.03088 [cs, eess]*, 2019.
- [17] E. Barmounakis and N. Geroliminis. On the new era of urban traffic monitoring with massive drone data: The pNEUMA large-scale field experiment. *Transportation Research Part C: Emerging Technologies*, 111:50–71, 2020. ISSN 0968-090X. doi:<https://doi.org/10.1016/j.trc.2019.11.023>. URL <https://www.sciencedirect.com/science/article/pii/S0968090X19310320>.
- [18] U. D. of Transportation Federal Highway Administration. Next generation simulation (ngsim) vehicle trajectories and supporting data. Technical report, Provided by ITS DataHub through Data.transportation.gov, 2016. URL <http://doi.org/10.21949/1504477>.
- [19] N. Karnchanachari, D. Geromichalos, K. S. Tan, N. Li, C. Eriksen, S. Yaghoubi, N. Mehdipour, G. Bernasconi, W. K. Fong, Y. Guo, et al. Towards learning-based planning: The nuplan benchmark for real-world autonomous driving. *arXiv preprint arXiv:2403.04133*, 2024.
- [20] Z. Huang, H. Liu, and C. Lv. Gameformer: Game-theoretic modeling and learning of transformer-based interactive prediction and planning for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3903–3913, 2023.
- [21] S. Sharan, F. Pittaluga, M. Chandraker, et al. Llm-assist: Enhancing closed-loop planning with language-based reasoning. *arXiv preprint arXiv:2401.00125*, 2023.
- [22] K. Guo, Z. Miao, W. Jing, W. Liu, W. Li, D. Hao, and J. Pan. Lasil: Learner-aware supervised imitation learning for long-term microscopic traffic simulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15386–15395, June 2024.
- [23] D. Chen and P. Krähenbühl. Learning from all vehicles. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17201–17210, 2022. URL <https://api.semanticscholar.org/CorpusID:247596695>.
- [24] B. R. Kiran, I. Sobh, V. Talpaert, P. Mannion, A. A. A. Sallab, S. Yogamani, and P. Pérez. Deep reinforcement learning for autonomous driving: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 23(6):4909–4926, 2022. doi:10.1109/TITS.2021.3054625.
- [25] 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*, 23(9):14128–14147, 2022. doi:10.1109/TITS.2022.3144867.
- [26] A. O. Ly and M. Akhloufi. Learning to drive by imitation: An overview of deep behavior cloning methods. *IEEE Transactions on Intelligent Vehicles*, 6(2):195–209, 2021. doi:10.1109/TIV.2020.3002505.
- [27] M. Bansal, A. Krizhevsky, and A. Ogale. Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst. *arXiv preprint arXiv:1812.03079*, 2018.
- [28] Comma.ai. Openpilot: An open-source self-driving car system, 2025. URL <https://github.com/commaai/openpilot>. Accessed: 2025-02-26.

- [29] C. Wen, J. Lin, T. Darrell, D. Jayaraman, and Y. Gao. Fighting copycat agents in behavioral cloning from observation histories. *Advances in Neural Information Processing Systems*, 33: 2564–2575, 2020.
- [30] K. Wang, B. Kang, J. Shao, and J. Feng. Improving generalization in reinforcement learning with mixture regularization. *Advances in Neural Information Processing Systems*, 33:7968–7978, 2020.
- [31] Y. Yu, S. Khadivi, and J. Xu. Can data diversity enhance learning generalization? In N. Calzolari, C.-R. Huang, H. Kim, J. Pustejovsky, L. Wanner, K.-S. Choi, P.-M. Ryu, H.-H. Chen, L. Donatelli, H. Ji, S. Kurohashi, P. Paggio, N. Xue, S. Kim, Y. Hahm, Z. He, T. K. Lee, E. Santus, F. Bond, and S.-H. Na, editors, *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4933–4945, Gyeongju, Republic of Korea, Oct. 2022. International Committee on Computational Linguistics. URL <https://aclanthology.org/2022.coling-1.437/>.
- [32] R. Zhang, J. Hou, F. Walter, S. Gu, J. Guan, F. Röhrbein, Y. Du, P. Cai, G. Chen, and A. Knoll. Multi-agent reinforcement learning for autonomous driving: A survey. *arXiv preprint arXiv:2408.09675*, 2024.
- [33] R. Karagulle, N. Aréchiga, J. DeCastro, and N. Ozay. Classification of driving behaviors using stl formulas: A comparative study. In *International Conference on Formal Modeling and Analysis of Timed Systems*, pages 153–162, 2022.
- [34] J. Phillion and S. Fidler. Lift, Splat, Shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3D. In *European Conference on Computer Vision (ECCV)*, pages 194—210, 2020.
- [35] Z. Li, W. Wang, H. Li, E. Xie, C. Sima, T. Lu, Q. Yu, and J. Dai. Bevformer: learning bird’s-eye-view representation from lidar-camera via spatiotemporal transformers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(3):2020–2036, 2025.

A Implementation Details

- The cost map (auxiliary loss) is not used, as it requires the vehicle’s dimensions during loss computation and provides minimal benefit [1].
- Training is conducted on four Nvidia RTX A5000 GPUs with a batch size of 32. For 100K scenarios, training takes approximately three days.
- Some scenarios fail during evaluation; failed tests are discarded to ensure a fair calculation.
- Due to augmentation constraints, some features cannot be computed. On average, for $N_s = 1$, we obtain 0.75 augmented samples per scenario.
- The same hyperparameters as for PLUTO are used for training, as shown in Table 2
- Heading deviations exceeding $\frac{\pi}{4}$ that corresponds to noisy observations are discarded.
- During the feature computation for augmented vehicles, certain vehicle parameters are approximated. Further details are available in the code.

Notation	Parameters	Values
T_H	Historical timesteps	20
T_F	Future timesteps	80
D	Hidden dimension	128
L_{enc}	Num. encoder layers	4
L_{dec}	Num. decoder layers	4
N_L	Num. lon. queries	12
α	Score weight	0.3
σ	Temperature parameter	0.1

Table 2: Parameters used in PLUTO

B Test14-hard vs. Val14 Sets

We conducted our evaluation using only the test14-hard set due to the strong correlation observed between val14 and test14-hard results for learning-based planners (see Figure 5). The computational demands of PLUTO evaluation are significant—test14-hard requires 5 hours using 4 Nvidia RTX A5000 GPUs, while val14 would require approximately 24 hours on the same hardware.

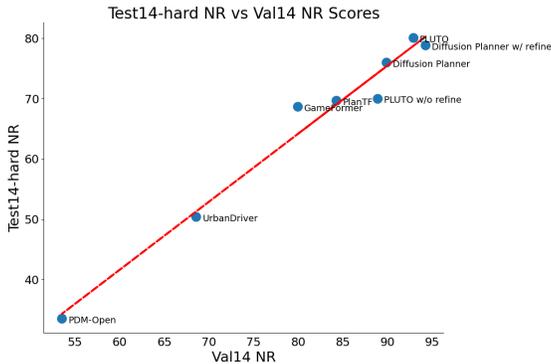


Figure 5: Correlation between test14-hard and val14 results for learning based planners

C Experiment Results

Here we present more detailed experiment results in Table 3.

Planner	Score	Collisions	TTC	Drivable	Comfort	Progress
1K Scenarios						
PLUTO (Baseline)	58.60	83.02	76.23	92.45	70.94	71.20
PLUTO (Ours, $\tau = 0.5, N_s = 1$)	60.31	82.76	75.48	92.34	68.20	74.66
PLUTO (Ours, $\tau = 0.1, N_s = 1$)	57.19	85.85	75.85	83.01	67.92	73.41
PLUTO (Ours, $\tau = 0.5, N_s = 2$)	65.73	89.84	82.52	93.90	71.54	75.20
PLUTO (Ours, $\tau = \infty, N_s = 1$)	49.57	76.28	66.01	91.70	66.40	71.73
10K Scenarios						
PLUTO (Baseline)	61.95	83.90	74.91	93.26	76.03	78.72
PLUTO (Ours, $\tau = 0.5, N_s = 1$)	72.17	91.29	82.58	95.08	80.68	80.61
PLUTO (Ours, $\tau = 0.1, N_s = 1$)	71.08	90.94	83.02	95.09	79.24	76.62
PLUTO (Ours, $\tau = 0.5, N_s = 2$)	66.70	85.23	78.03	93.56	77.27	82.35
PLUTO (Ours, $\tau = \infty, N_s = 1$)	72.54	92.45	83.77	94.72	78.49	78.61
100K Scenarios						
PLUTO (Baseline)	74.81	91.23	83.96	97.01	86.57	78.64
PLUTO (Ours, $\tau = 0.5, N_s = 1$)	77.38	93.75	84.09	96.97	87.88	80.47
PLUTO (Ours, $\tau = 0.1, N_s = 1$)	76.28	93.04	84.21	96.62	83.83	80.78
PLUTO (Ours, $\tau = \infty, N_s = 1$)	75.37	92.48	84.21	95.86	81.58	83.75

Table 3: Performance comparison of PLUTO planner under base and augmented datasets with varying number of scenarios (1K, 10K, 100K) and different parameters. Test14-hard dataset is used for evaluation. All scores are between 0 and 100 where the higher is the better.

D Further simulation results

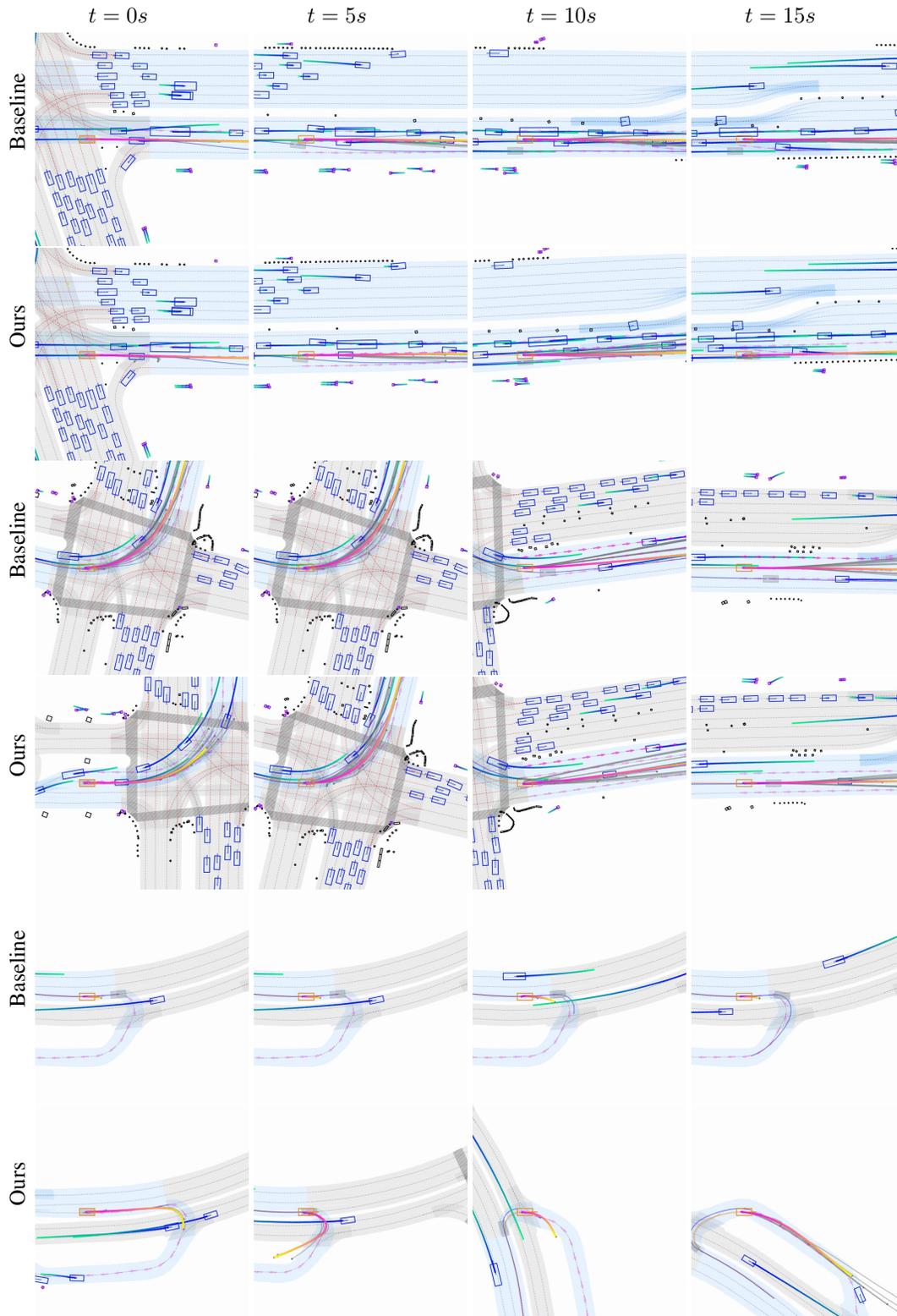


Figure 6: Comparison between baseline and augmented policy outputs on a variety of scenarios. Note that the videos will be published on the project webpage.