

WOD-E2E: Waymo Open Dataset for End-to-End Driving in Challenging Long-tail Scenarios

Runsheng Xu^{*†}, Hubert Lin^{*}, Wonseok Jeon , Hao Feng , Yuliang Zou , Liting Sun
John Gorman , Ekaterina Tolstaya , Sarah Tang , Brandyn White , Ben Sapp
Mingxing Tan , Jyh-Jing Hwang[†], Dragomir Anguelov

Waymo LLC

Abstract

Vision-based end-to-end (E2E) driving has garnered significant interest in the research community due to its scalability and synergy with multimodal large language models (MLLMs). However, current E2E driving benchmarks primarily feature nominal scenarios, failing to adequately test the true potential of these systems. Furthermore, existing open-loop evaluation metrics often fall short in capturing the multi-modal nature of driving or effectively evaluating performance in long-tail scenarios. To address these gaps, we introduce the Waymo Open Dataset for End-to-End Driving (WOD-E2E). WOD-E2E contains 4,021 driving segments (approximately 12 hours), specifically curated for challenging long-tail scenarios that are rare in daily life with an occurring frequency of less than 0.03%. Concretely, each segment in WOD-E2E includes the high-level routing information, ego states, and 360-degree camera views from 8 surrounding cameras.

To evaluate the E2E driving performance on these long-tail situations, we propose a novel open-loop evaluation metric: Rater Feedback Score (RFS). Unlike conventional metrics that measure the distance between predicted way points and the logs, RFS measures how closely the predicted trajectory matches rater-annotated trajectory preference labels. We have released rater preference labels for all WOD-E2E validation set segments, while the held out test set labels have been used for the 2025 WOD-E2E Challenge. Through our work, we aim to foster state of the art research into generalizable, robust, and safe end-to-end autonomous driving agents capable of handling complex real-world situations.

1. Introduction

Autonomous driving systems have traditionally followed a modular design approach that decomposes the driving task into distinct sub-tasks such as perception, prediction, and planning [13, 15, 22, 27, 36, 38]. While this modular design offers benefits in terms of interpretability and debugging, the research community has recently shifted its attention to exploring vision-based end-to-end (E2E) architectures [7, 21, 30, 34, 37]. This shift is primarily driven by the inherent scalability of E2E systems, which directly map raw sensor data to driving actions, reducing the underlying system complexity and the need for rater annotations of intermediate concepts [33]. Furthermore, as previous works [14, 28] indicate, there is a promise of leveraging multi-modal large language models (MLLMs) and their world knowledge for E2E driving.

Despite this promise, current real-world E2E driving datasets, such as NAVSIM [8], WODM [10] and CoVLA [1], predominantly feature *nominal* driving scenarios that do not fully expose systems to the long tail of possible real-world situations. This scarcity of long-tail examples hinders the accurate evaluation of the true potential, robustness, and generalization ability of E2E driving systems.

In this paper, we introduce the newly released Waymo Open Dataset for End-to-End Driving (WOD-E2E), which explicitly focuses on long-tail situations. As shown in Figure 1, WOD-E2E features rare real-world scenarios, which occur with a frequency of less than 0.03%. We provide 4,021 challenging driving segments comprising approximately 12 hours in total, where each segment contains 8 surrounding cameras covering a 360-degree field of view, high-level routing information, ego vehicle position history, and 5s of its future trajectory. These driving segments are collected from a mixture of autonomous and manual driving.

^{*}Equal contributions.

[†]Contact emails: Runsheng Xu <runshengxu@waymo.com>, Jyh-Jing Hwang <jyhh@waymo.com>.

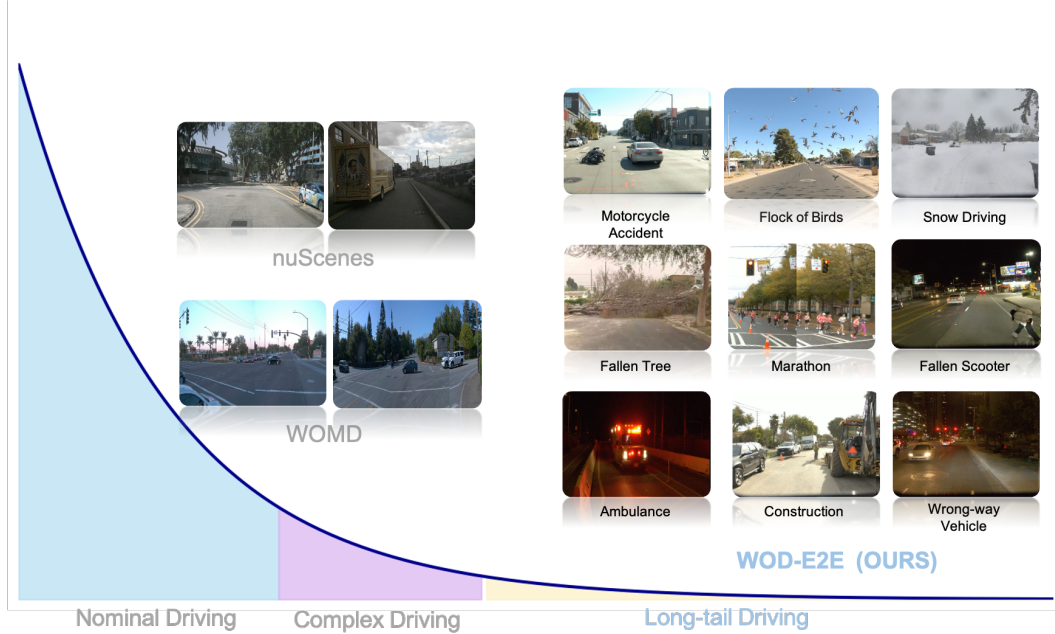


Figure 1. **Long-tail scenario examples from the Waymo Open Dataset for End-to-End Driving (WOD-E2E).** Unlike existing datasets that are commonly used for E2E Driving benchmarking, WOD-E2E dataset has more explicit focus on long-tail scenarios. Our analysis in Section 3.3 shows that WOD-E2E captures the long-tail scenarios with a frequency of less than **0.03%** in daily driving.

Moreover, we observe that previous open-loop metrics often fail to adequately evaluate the driving performance in these long-tail scenarios. The popular Average Distance Error (ADE) or L2 error metric captures only the error between a prediction and a single future ground truth trajectory, despite the driving behavior being inherently multi-modal, where multiple reasonable future trajectories are possible. Predictive metrics, such as PDMS scores [8], require annotated positions and future trajectories of road agents to calculate collision rates, and thus become impractical in many long-tail scenarios involving novel or hard-to-detect objects (e.g., the flock of birds shown in Figure 1). Furthermore, off-road behaviors typically incur high penalties in PDMS, yet in numerous safety-critical long-tail scenarios, an autonomous vehicle might reasonably deviate partially off-road to avoid an emergency. To address these limitations, WOD-E2E dataset also includes a subset of human driving preference labels, providing expert ratings on multiple potential trajectories in each example. Leveraging these labels, we propose a novel open-loop evaluation metric, the Rater Feedback Score (RFS), to better evaluate the E2E driving performance in an open-loop setting.

We conduct rigorous studies with robust baseline models to verify the dataset and RFS. Since the dataset release, we have garnered significant interest from the

research community, with numerous methods already submitted and evaluated on our public leaderboard. The diversity of these top-performing methods, employing approaches such as MLLMs [25, 29], diffusion models [17], and CNN/ViT with GRU/MLP architectures [23], further underscores the utility of the WOD-E2E dataset and its promise to drive further advances in end-to-end autonomous driving research. Our contribution can be summarized as:

- We introduce **WOD-E2E**, a new open dataset focusing on long-tail scenarios for benchmarking end-to-end autonomous driving systems. It contains 4,021 challenging driving segments, totaling approximately 12 hours of data and representing real-world long-tail scenarios occurring with a frequency of less than 0.03% in daily driving.
- We propose Rater Feedback Score (RFS), a novel and human-aligned open-loop metric. RFS is designed to better assess E2E driving performance in long-tail scenarios, addressing the limitations of traditional open-loop metrics like ADE and PDMS.
- We provide detailed comparison and analysis for our baseline E2E model and multiple methods submitted to our public leaderboard, based on this new dataset. The widespread participation validates the dataset’s usefulness for facilitating the E2E driving research.

In the remainder of this paper, we first discuss related works in Section 2. In Section 3, we describe in detail the proposed WOD-E2E dataset, including overview, quantitative analysis, mining strategy, labeling, and the rater feedback score metric. Finally, we summarize all the experimental results and detailed analysis in Section 4 and conclude the paper in Section 5.

2. Related Works

2.1. End-to-end autonomous driving research

The paradigm of E2E autonomous driving, directly mapping raw sensor inputs to control outputs, continues to be a vibrant area of research, seeking to overcome the complexities in traditional modular pipelines [14, 33]. Recent works have significantly advanced the capabilities of E2E systems, particularly through the use of foundation models. Overall, the current methods can be divided into three categories:

Bird’s-Eye-View (BEV) Based E2E Planner: This type of method aims to fuse information from multiple sensors into a single, comprehensive BEV representation, from which both perception and planning tasks can be directly performed. UniAD [12] exemplifies this by propagating BEV queries from its perception module to downstream tasks such as tracking, motion forecasting, and occupancy prediction, ultimately enabling end-to-end planning. Similarly, BEV-Planner [16] focuses on learning an explicit planning policy directly from BEV features, demonstrating how dense BEV representations can facilitate robust end-to-end control. These approaches move beyond explicit intermediate perception outputs for planning. Overall, these unified BEV-centric methods offer advantages in terms of computational efficiency and coherence by providing a consistent spatial understanding across various driving sub-tasks.

Multi-modal Large Language Model Based E2E Planner: A prominent trend involves leveraging Multimodal Large Language Models (MLLMs) to imbue E2E driving systems with enhanced reasoning capabilities and world knowledge. DriveGPT4 [37] utilizes LLMs to both explain vehicle actions and predict control signals in an iterative question-and-answer format. DriveVLM [28] applies chain-of-thought for end-to-end driving, while VLP [21] applies the reasoning of MLLMs directly on the Bird’s-Eye-View (BEV) space. EMMA [14] leverages Gemini to process multiple driving tasks, including planning, 3D detection, and road understanding, within a unified language space. OpenEMMA [34] and LightEMMA [24] follow a similar paradigm to build an open-source and lightweight version, respectively. Additionally, S4-Driver [33] proposes to lift the vision

tokens from MLLMs to a 3D space.

Diffusion Based E2E Planner: Diffusion models excel at capturing the multi-modal nature of driving actions and generating diverse, plausible trajectories. Notably, DiffusionDrive [17] introduces a truncated diffusion policy and efficient cascade decoder for real-time E2E driving. EnDfuser [32] further explores using diffusion ensembles to estimate uncertainty in trajectory planning, leveraging fused camera and LiDAR features to produce distributions of candidate trajectories.

2.2. End-to-end autonomous driving open dataset

A multitude of autonomous driving datasets are available today, supporting a diverse range of driving tasks. Notable examples include Kitti [11], Argoverse [5], Argoverse 2 [31], WOD-Perception [26], and V2V4Real [35]. While these datasets serve various purposes, a specific subset focuses on end-to-end driving. Among the most prominent open datasets in this category are nuScenes [2], NAVSIM [8], WOMD [10], and CoVLA [1].

2.2.1. nuScenes

nuScenes [2] is initially developed for perception tasks and features multiple sensor modalities. While recent research [14, 21, 28] has explored end-to-end driving on this dataset, often using ADE as a primary performance indicator, the core focus of nuScenes remains perception rather than planning. Some studies [16, 39] have observed that even simple extrapolation of historical behavior can yield strong performance without relying on camera images, suggesting that nuScenes may not be ideally suited for complex planning tasks.

2.2.2. NAVSIM

NAVSIM [8] is a compact simulation and benchmarking framework built upon a filtered version of nuPlan [3]. Its core contribution lies in enabling large-scale real-world evaluation through a non-reactive simulator, which effectively bridges the gap between open-loop and closed-loop testing via simulation-based metrics. While NAVSIM has helped to significantly advance end-to-end driving research, its approach presents two major limitations that motivated our work. First, as a simulation framework, it relies on filtering existing datasets rather than providing a **raw data collection effort** specifically for long-tail events, which may preclude it from capturing the full, nuanced diversity of real-world long-tail scenarios. Second, the PDMS proposed in NAVSIM—which heavily prioritizes ego progress and comfort, along with time-to-collision (TTC)—may prove insufficient for true safety-critical situations. For instance, TTC is challenging to measure with amorphous obstacles like a flock of birds, as depicted in Figure 1, and the metric of comfort should be secondary

to safety when the vehicle must perform an emergency maneuver, such as avoiding a falling scooter (Figure 1). Our dataset and evaluation methodology are explicitly designed to overcome these two limitations by providing targeted, diverse, long-tail data and a more safety-focused scoring mechanism.

2.2.3. WOMD

The Waymo Open Motion Dataset (WOMD) [10] is a component of the broader Waymo Open Dataset, with a specific emphasis on motion prediction and behavior research. Although recent works such as MoST [19] and S4-Driver [33] conduct end-to-end driving research on it, WOMD is primarily designed for motion prediction and for modeling complex agent interactions, rather than planning. Furthermore, the lack of full camera images (only embeddings are provided) makes it difficult for external researchers to conduct comprehensive E2E research.

2.2.4. CoVLA

CoVLA [1] provides a large-scale, richly annotated collection of real-world driving scenarios, integrating vision, language, and action modalities. It is designed to enable the training of Vision-Language-Action models that can generate descriptive scene captions and predict vehicle trajectories. While CoVLA’s automated captioning aims for diversity and covers a wide range of common driving conditions, the available information does not detail specific mechanisms for over-sampling or synthesizing rare, safety-critical long-tail events beyond general diversity.

3. WOD-E2E Dataset

3.1. Dataset Overview

This dataset contains 4,021 driving segments mined from real driving logs. Each segment is 20-second long and focused on long-tail scenarios. The dataset is partitioned as: 2,037 segments for training, 479 segments for validation, and the rest 1,505 segments for testing.

3.1.1. Coordinate System

This dataset employs two primary coordinate systems: vehicle coordinates and sensor frame coordinates.

Vehicle Coordinates: The vehicle coordinate system is located at the ego vehicle’s center. The x-axis points forward, the y-axis points left, and the z-axis points upward. All trajectory data is referenced to this vehicle coordinate system.

Sensor Frames: Each sensor frame is related to the vehicle frame by an extrinsic transformation. For cameras, the frame is centered at the lens. The x-axis points out from the lens, the z-axis points upward, and the

y/z plane is parallel to the camera’s image plane. This is a right-handed coordinate system.

3.1.2. Camera Data

This dataset includes images from eight cameras, providing 360-degree coverage around the vehicle: front, front left, front right, side left, side right, rear, rear left, and rear right. The sensor layout configuration is similar to that described in [26]. For each direction, a single JPEG image is provided. Alongside the image data, we supply camera intrinsics and extrinsics, which define the camera’s internal parameters and its position relative to the vehicle’s center, respectively. These parameters enable the projection of 3D trajectories onto the camera images. Each driving segment includes 10Hz camera video sequences. Training data spans 20 seconds, while testing data covers 12 seconds, with the subsequent 8 seconds of future data hidden for evaluation purposes.

3.1.3. Routing Information

We provide a routing input for the model in the form of a high-level command, following conventional academic benchmarks [4, 12].

The high-level command is encoded as an enum {GO_STRAIGHT, GO_LEFT, GO_RIGHT}. These commands specify expected driving direction at decision points, such as intersections or highway on/off ramps. GO_STRAIGHT means the vehicle should continue along the current path, while GO_{LEFT,RIGHT} means the vehicle should take a branching path instead. Note that commands do not refer to micro maneuvers, such as lane changes or nudges around objects on the road, and do not provide any speed profile information.

We construct high-level commands by comparing the vehicle’s 10s future driven route against its current position along the route. An illustration is shown in Fig. 2.

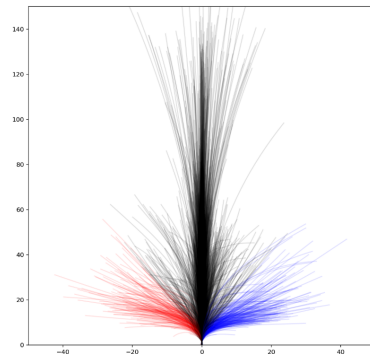


Figure 2. High-level routing input. Ground-truth vehicle trajectories over future 5s are shown. Each trajectory is colored red/black/blue corresponding to left/straight/right routing input, derived from 10s futures. Units are in meters.

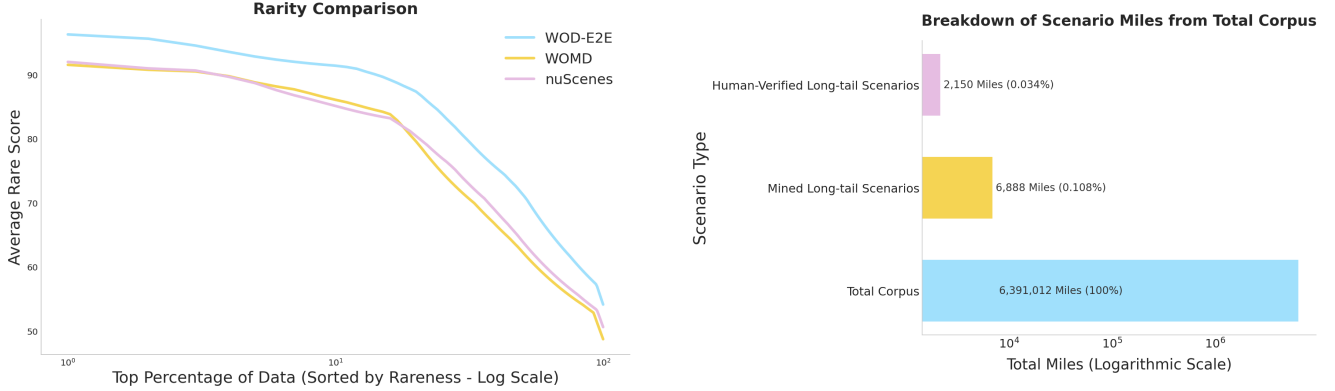


Figure 3. **Left:** Rarity comparison of driving Datasets. This figure shows the average rarity score for the top percentage of data in each dataset, highlighting the distribution of rare events in WOD-E2E. **Right:** Proportion of mined long-tail scenarios (**0.03%**) from the total driving corpus (**6.4 million miles**).

3.1.4. Ego Status

Each driving segment includes ego vehicle status information, comprising:

Past Trajectory: The ego vehicle’s past 4-second trajectory, aligned with the current camera timestamp, is provided as waypoints [(x1, y1), (x2, y2),...] at 4Hz frequency. All waypoints are in vehicle coordinates.

Velocity and Acceleration: The ego vehicle’s velocity and acceleration, aligned with its past trajectory, are also provided.

Future Trajectory: The ego vehicle’s future 5-second trajectory from the driving log is provided in the same format and frequency as the past trajectory. This information is available only for the training and validation sets.

3.1.5. Labels

Scenario Cluster: Each segment is tagged with one of 11 scenario types, which will be explained in detail in the following sections.

Rater Feedback Labels: To capture the diversity of acceptable driving decisions during critical events, this dataset includes rater feedback labels. At specific moments within each driving segment, expert labelers rate three distinct 5-second future trajectories on a scale of 0 to 10, where 0 indicates the worst driving and 10 the best. Importantly, we ensure that at least one of the rater-specified trajectories receives a score higher than 6. This label is provided only for the validation set. Details on the creation of these labels will be provided in a subsequent section.

3.2. Quantitative Rareness Comparison

In this section, we quantitatively compare the **rarity** of WOD-E2E against other popular E2E driving datasets. To achieve a standardized, impartial rarity assessment,

we utilized a large language model, Gemini 2.5 Pro [6], to score the test set of each dataset. The model was provided with the front camera sequences and a detailed scoring prompt outlining four tiers of rarity based on complexity, risk, and long-tail factors. The prompt required the output to be a JSON object containing the **rarity_score** that is ranged from 0-100, identified **rare_factors**, and a **reasoning** trace for maximum transparency.

After scoring each scene, we plot the comparative rarity distribution in Figure 3 (left). This curve is generated by ranking all scenes by their rarity score (high to low) and plotting the average rarity score for all scenes up to that percentage of the dataset.

The figure clearly demonstrates the long-tailed focus of our dataset. We can see that the WOD-E2E curve is significantly higher than all other datasets across all percentage tiles, confirming a higher concentration of long-tail events. Specifically, WOD-E2E maintains a higher average score (around 93) for the most extreme 10% of the data, and crucially, its score remains elevated even when considering the full dataset, which indicates the high density of rare scenarios relative to other datasets.

3.3. Long-tail Data Mining

3.3.1. Mining Strategy

We have access to a very large database containing diverse, real-world driving logs that span millions of miles. The vast majority of this data, however, consists of nominal scenarios. To effectively extract only the long-tail scenarios, we developed an efficient mining strategy that combines rule-based heuristics and MLLMs. Firstly, we categorized all driving logs into 11 different categories:

- **Construction:** Scenarios involving construction zones.
- **Intersection:** Scenarios with complex interactions at intersections.
- **Pedestrians:** Scenarios involving interactions with pedestrians.
- **Cyclists:** Scenarios involving interactions with cyclists.
- **Multi-Lane Maneuvers:** Scenarios where the ego vehicle is required to change lanes on multi-lane roads.
- **Single-Lane Maneuvers:** Scenarios where the ego vehicle is required to take actions on single-lane roads.
- **Cut-ins:** Scenarios where other on-road agents cut into the ego vehicle’s lane.
- **Foreign Object Debris:** Scenarios with rare objects such as animals or furniture.
- **Special Vehicles:** Scenarios involving special vehicles.
- **Spotlight:** Manually selected challenging scenarios.
- **Others:** Scenarios that do not belong to any of the above clusters.

The detailed mining criteria for each category are shown in Table 1. These criteria are made possible by the rich auto-labels available in our dataset, including 3D detection, mapping, tracking, and prediction, which provide the necessary heuristics for our mining process.

3.3.2. Case Study

To validate the effectiveness of our mining strategy, we conducted a case study on a recent set of driving logs that includes a total of 6,391,012 miles. After applying our automated mining strategy, we found that only 6,888 miles (0.1%) of the data fit our criteria for long-tail scenarios. This initial result shows that our strategy is highly effective at isolating rare, challenging events from a massive volume of nominal driving data, as demonstrated in the right figure of Figure 3.

Moreover, to ensure the highest quality of our dataset, we perform a subsequent round of human filtering. This manual review process, which has a conversion rate of 30%, further refined the mined data by removing non-long-tail scenarios. This final filtering step reduced the overall portion of long-tail scenarios to an even rarer 0.03%, highlighting the significant infrequency of these critical events in real-world driving.

3.3.3. Data Analysis

As shown in Figure 4, we analyze the dataset’s distributions across three key dimensions: city locations, scenario clusters, and driving behaviors.

City Distribution. The top-left subfigure shows the geographical distribution of the dataset across different

cities. For confidential reasons, all city names have been anonymized. The data is predominantly sourced from cities L, K, and J, and the remaining cities, which are only present in the test set, contribute a smaller but more diverse set of scenarios, which is crucial for evaluating model generalization.

Scenario Clusters. The bottom-left subfigure provides a clear overview of our dataset’s composition by problem cluster and road type.

- We first analyze the distribution of long-tail scenarios by their problem clusters. The clusters for Intersections, Foreign Object Debris (FOD), and Pedestrians account for the largest share of the dataset. This highlights our focus on a variety of complex and safety-critical events, including intricate interactions at intersections, challenging scenes for the perception module, and high-risk encounters with pedestrians.
- Our dataset contains three major road types: Local Road, Arterial Minor, and Freeway. The Freeway road type is most prominent in the Cut-ins cluster, which is a particularly safety-critical event at high speeds. It is also notably present in the Intersections cluster. This is because these scenarios specifically capture interactions at freeway entrances and exits, such as making a right turn to enter an on-ramp.

Driving Behavior Distribution. The right subfigure shows the distribution of driving behaviors. We have a variety of diverse behaviors, including moving straight, lane changes, left turns, right turns, and on-ramp maneuvers. The majority of behavior is moving straight, which includes typical lane-following, but also hard braking and swerving for emergency situations. Turning behaviors at intersections, including left and right turns, make up approximately 30% of the data, with roughly equal proportions. Additionally, lane changes account for 10.3% of the scenarios, which usually involve collision or obstacle avoidance. Finally, a small portion of the data (1.7%) is dedicated to on-ramp behaviors, which are often challenging to tackle due to the interaction of merging vehicles at high speeds.

3.4. Data Labeling

The mined data is sent to our data labeling pipeline, which consists of three major steps: critical moment selection, trajectory sampling, and trajectory scoring.

3.4.1. Critical Moment Selection

The critical moment is defined as the specific frame where a critical event emerges, requiring the vehicle to make an important driving decision. These decisions can include actions like slowing down, nudging, or giving way to other vehicles in the scene. An example can be found in Figure 5.

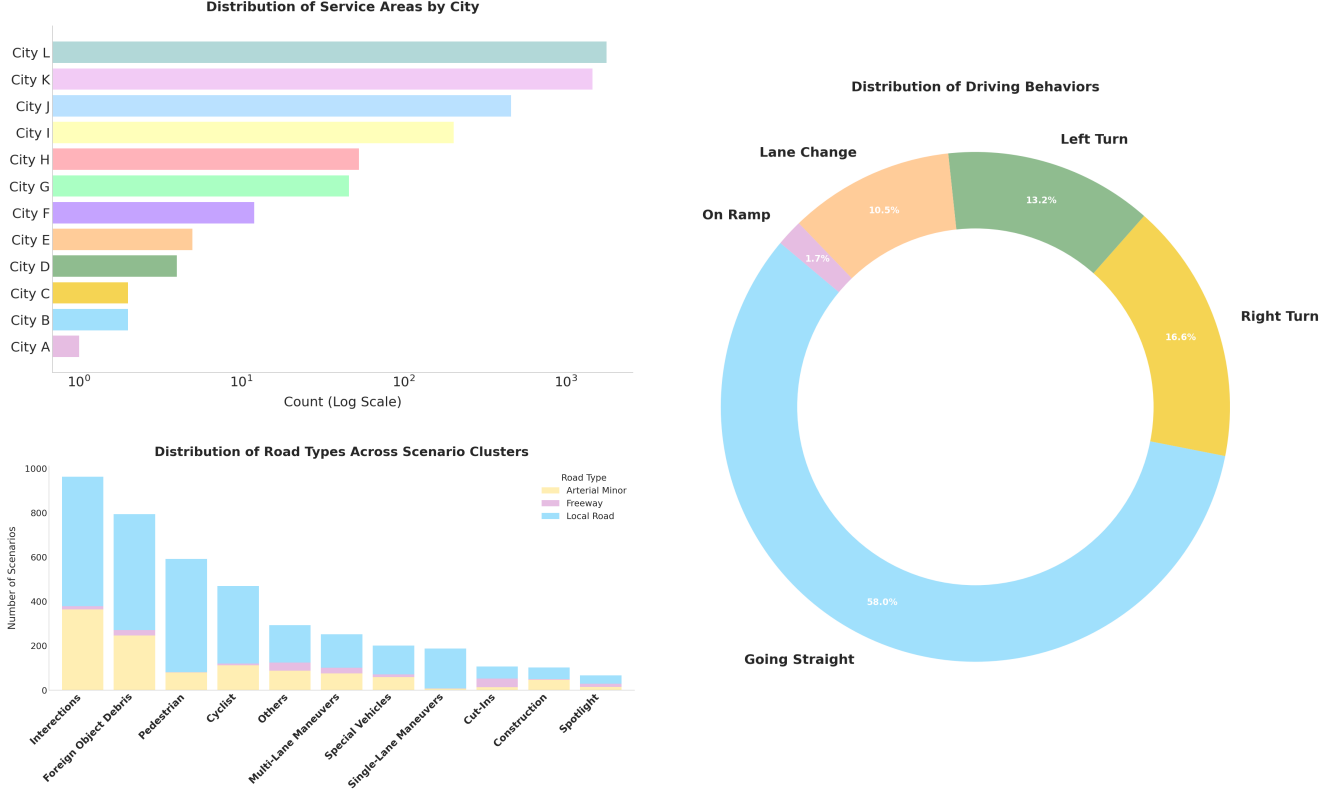


Figure 4. Comprehensive data distribution analysis. This figure illustrates the key characteristics of the WOD-E2E dataset across three critical dimensions. **Top Left:** Distribution of service areas by city. **Bottom Left:** Distribution of scenario clusters and their breakdowns by road type. **Right:** Distribution of driving behaviors.

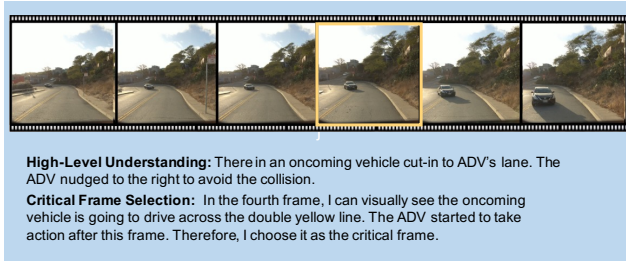


Figure 5. An illustration of how a critical frame is selected. The human raters first scan through the video for high-level understanding, and then select the critical frame, which is the earliest moment when a critical event is visually apparent in the camera images. Finally, the raters also document the rationales for the critical frame selection.

We instruct our labelers to follow a three-step process for selecting the precise moment:

1. **High-level Understanding:** Labelers must first scan the entire video to understand the critical event within the segment and identify the correct driving decision to be made.
2. **Moment Selection Based on Visual Cues:** La-

belers must then find the earliest moment where the critical event is visually apparent in the camera feed. They are instructed to select the frame where the autonomous vehicle has already started taking action to avoid reaction bias introduced by the history motion information. This is typically the frame where the target behavior is most clearly exhibited, such as the initial moment of a lane change or the point of start braking.

3. **Reasoning Documentation:** The final step involves briefly documenting the rationale for selecting the specific frame. This documentation ensures consistency and provides valuable feedback for model training and analysis.

3.4.2. Trajectory Sampling

Trajectory sampling is the process of generating a diverse set of possible motion plans for later human review and selection in a specific driving scenario. Our approach utilizes an existing machine learning model, such as Wayformer [20], to produce an initial set of up to 64 diverse trajectories for a given critical moment. These trajectories are generated using various inputs,

Table 1. Mining criteria for each long-tail scenario category.

Construction	Intersection
• Driving route changes due to road closures from a construction zone. • Uniformed pedestrians directing traffic. • Abnormal road surface conditions due to construction.	• Unprotected maneuvers with limited visibility or heavy traffic interactions. • Complex interactions at stop sign intersections. • Interactions with other traffic-violating agents at traffic light intersections. • Interactions with rails and cable cars at intersections.
Pedestrians	Cyclists
• Pedestrians crossing with low visibility due to occlusion or weather. • Emergent behavior required to avoid collisions with pedestrians exhibiting unexpected behaviors. • Pedestrians performing unsafe maneuvers specific to the autonomous vehicle.	• Cyclists losing control nearby. • Interactions with a group of cyclists.
Cut-ins	Foreign Object Debris
• Oncoming agent cuts across the ego vehicle’s trajectory. • An agent in a neighboring lane cuts across the ego vehicle’s lane aggressively.	• Interactions with animals on road • Debris that can causes damage on the ADV’s path, such as large box, glass debris, and metal debris • Abnormal road condition, such as flooded road, fire on the roadside, severely and degraded road.
Multi-lane Maneuvers	Single Lane Maneuvers
• Nudge maneuvers to overtake blocked agents in the current lane • Lane merging maneuvers on freeway • Other agents in the other lane get too close to ADV that could cause hazards	• Overtake maneuvers in narrow single lane roads • Interactions with open-door vehicle in a narrow single lane road
Special Vehicles	Spotlight
• Emergency vehicles blocking road due to accidents or construction • Pull-over required due to the emergency vehicles	• Leveraging Gemini to search over the database to find scenarios containing certain long-tail objects

including perception detections, mapping elements, and predicted behaviors of other road agents.

Our trajectory selection process employs a two-step approach that leverages both automated filtering and human-guided selection to identify the most representative motion plans for rating. Initially, the generated trajectories are automatically sorted into different “buckets” based on driving decisions, such as velocity, acceleration, and lane changes. From these buckets, we sample a set of diverse candidates (usually fewer than 12). This sampling typically involves selecting the leftmost, middle, and rightmost trajectories to capture a spectrum of lateral movements. This small set of diverse trajectories is then passed to human labelers. The labelers’ task is to select three trajectories from these candidates for final ranking and reasoning, ensuring the labeled data includes the optimal path alongside plausible alternative and suboptimal behaviors.

3.4.3. Trajectory Scoring

The sampled trajectory candidates, along with the selected critical scenario, are sent to trained human raters under a rigorous manual grading process.

1. **Scenario Representation:** The selected long-tail scenarios are represented within a visualization tool to ensure effective and precise labeling. Each scenario is **20** seconds long and includes comprehensive data, such as mapping elements, camera images, and annotations for all on-road agents. Candidate trajectories are also plotted directly in this environment. Labelers can easily navigate different timestamps to precisely visualize how each candidate trajectory interacts with the logged future behavior of other road agents or static map elements. This capability is crucial for informed decision-making.
2. **Trajectory Selection and Grading Criteria:** Within a selected scenario, raters first select three diverse trajectories from the available candidates. This

selection must include at least one trajectory that is considered optimal or appropriate behavior, while the other two should represent different behavioral modes that may be sub-optimal. The labelers then rate these three trajectories based on five distinct dimensions:

Safety: Whether the trajectory results in collisions, near-misses, or other unsafe conditions.

Legality: Whether the trajectory complies with all traffic laws and regulations, including proper behavior around emergency vehicles.

Reaction Time: Whether the autonomous vehicle’s actions within the trajectory are timely in response to unfolding events.

Braking Necessity: Whether the trajectory includes unnecessary, sudden, or overly conservative braking.

Efficiency: Whether the trajectory demonstrates efficient progress, avoiding unnecessary lane changes, hesitations, or over-reactions to distant or irrelevant agents.

3. **Scoring Mechanism:** Trajectories are scored on a scale from 0 (worst) to 10 (perfect). Each trajectory is initialized with a base score of **10 points**. Points are then deducted based on violations of the grading criteria:

- Major infractions: A deduction of **2 points** is applied for violations related to safety, reaction time, or legal violations.
- Minor infractions: A deduction of **1 point** is applied for violations related to braking necessity or efficiency.

These penalties are cumulative. In cases where a trajectory exhibits multiple concurrent violations, raters may apply additional discretionary deductions to reflect the severity of the combined faults, ensuring the final score accurately reflects the trajectory’s overall quality.

The distribution of final human ratings for the top three trajectories is visualized in Figure 6. This plot clearly demonstrates a deliberate separation of trajectory quality: The Rank 1 trajectory shows a strong bias towards optimal behavior, with its lowest observed score being 6, which is the minimum score required to regard a trajectory as safe and feasible. In contrast, the Rank 2 and Rank 3 trajectories span a much wider range, with a significant amount of data, particularly for Rank 3, falling below a score of 6. This diverse scoring range successfully captures the desired multi-modality in driving behavior. By including plausible sub-optimal and unsafe alternatives alongside the optimal path, the label distribution provides essential boundaries for estimating robust end-to-end driving models.

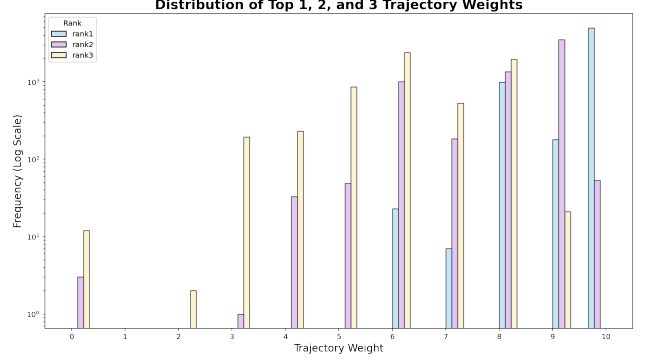


Figure 6. Human rating distribution among 3 candidate trajectories in the WOD-E2E dataset. This plot reveals a deliberate separation of trajectory quality. By including plausible sub-optimal and unsafe alternatives (Rank 3) alongside the optimal path (Rank 1), the label distribution provides essential boundaries for estimating robust end-to-end driving models.

3.5. Rater Feedback Score

The Rater Feedback Score (RFS) is a metric designed to evaluate the quality of a model’s predicted trajectory with the reference of multiple human-annotated trajectories. The WOD-E2E dataset includes 3 reference trajectories generated by human raters, each assigned a score s_{rater} in $[0, 10]$.

The RFS is designed to see how much the model’s prediction is aligned with three rated trajectories by considering trust regions, as illustrated in Figure 7. A *trust region* is defined around each rater trajectory at evaluation times t in $\{3, 5\}$ seconds. This region represents the rectangular space within specified longitudinal and lateral distance thresholds from the rater trajectory at a given time t .

The base thresholds follows WOMD [10], and they are set as $\bar{\tau}_{\text{lat}} = 1.0, \bar{\tau}_{\text{lng}} = 4.0$ at $t = 3$ and $\bar{\tau}_{\text{lat}} = 1.8, \bar{\tau}_{\text{lng}} = 7.2$ at $t = 5$, where the longitudinal threshold $\bar{\tau}_{\text{lng}}$ is always set to be 4 times larger than the lateral threshold $\bar{\tau}_{\text{lat}}$. These base thresholds are scaled based on the initial speed v (m/s) of the rater trajectory. The scaling function is a piece-wise linear function of v :

$$\text{scale}(v) = \begin{cases} 0.5, & v < 1.4, \\ 0.5 + 0.5 \times \frac{v-1.4}{11-1.4}, & 1.4 \leq v < 11, \\ 1, & v \geq 11. \end{cases}$$

The final thresholds at $t = 3, 5$ are determined by

$$\tau_{\text{lng}} = \text{scale}(v) \times \bar{\tau}_{\text{lng}}, \tau_{\text{lat}} = \text{scale}(v) \times \bar{\tau}_{\text{lat}}.$$

For distance errors Δ_{lng} (longitudinal) and Δ_{lat} (lateral) and the final thresholds, the score from each rater

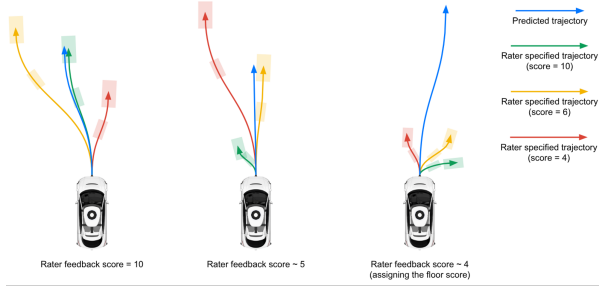


Figure 7. **Rater Feedback Score Mechanism.** This figure illustrates how the RFS evaluates a model’s Predicted Trajectory (Blue) against three human-rated reference trajectories. The predicted score is based on the highest-rated reference trajectory it aligns with within the defined trust region.

feedback trajectory is defined by

$$s_{\text{rater}} \times 0.1^{\max\left\{\max\left\{\frac{\Delta_{\text{lng}}}{\tau_{\text{lng}}}, \frac{\Delta_{\text{lat}}}{\tau_{\text{lat}}}\right\} - 1, 0\right\}}.$$

Intuitively, we assign either the flat score s_{rater} , if a predicted trajectory is within the trust region, or the score exponentially decayed from s_{rater} . Then, the final score is determined by choosing the maximum score over all rater specified trajectories, followed by averaging over $t = 3, 5$ and flooring with 4.

4. Experimental Results

4.1. Baseline Model Setup

We use a highly simplified version of EMMA [14], which we call NaiveEMMA, as our baseline model. The architecture of NaiveEMMA is illustrated in Figure 8. NaiveEMMA is finetuned directly from Gemini Flash [6] and has not been trained on any internal driving datasets: it is finetuned exclusively on the released WOD-E2E training split. The model consumes a combined image from all eight cameras at the current timestep, concatenated into a single 768×768 resolution image. It also takes in 3 seconds of past ego-status history and the high-level routing input. Crucially, it does not use past camera frames. Note that NaiveEMMA omits several advanced components of the original EMMA model, specifically generalist task training mixtures, Chain-of-Thought reasoning, and any test-time scaling methods.

4.2. RFS Metric Validation

4.2.1. Quantitative Validation

We train several models based on NaiveEMMA and evaluate RFS on an internal test split. This test split

contains long-tailed scenarios similar to the WOD-E2E test split. This experiment controls for several factors that are expected to improve model quality in long-tail settings: exposure to long-tailed scenarios via the WOD-E2E training split, multi-camera inputs to reason about surroundings, and test-time scaling to handle scenario ambiguities. RFS aligns with these intuitions, assigning higher scores to models that utilize more of these features (Table 3).

Model	RFS
Baseline	7.14
+ WOD E2E finetuning	7.22
+ multi-camera inputs	7.30
+ test-time scaling (multi sampling)	7.39

Table 3. RFS assigns higher scores to models that are better-equipped to handle long-tailed scenarios. Evaluation is performed on an internal test split.

4.2.2. Qualitative Validation

In this section, we validate the RFS metric through several qualitative examples, as Figure 9 shows.

Scores within the Trust Region (Figure 9a). (*Left*) This scene shows a slow-moving construction vehicle, where the optimal trajectory is to follow carefully. The model’s prediction aligns closely with the best-rated trajectory (Score 10.0), resulting in a perfect RFS of 10.0. (*Center*) In this complex urban intersection, a cable car is moving while another vehicle is executing a right turn. The most preferred trajectory (Score 8.0) is to proceed carefully through the intersection, whereas the two lower-rated trajectories involve suboptimal actions like hard braking or deviating from the route. Since the model’s prediction is well-aligned with the preferred path, it receives an RFS of 8.0. (*Right*) The best behavior here is to safely nudge right to proceed past the bus without collision. The model’s prediction accurately follows this optimal behavior, yielding an RFS of 10.0.

Decayed Scores outside the Trust Region (Figure 9b). (*Left*) In snowy conditions, labeled trajectories include proceeding straight and turning left. The prediction follows the left-turn maneuver but at a slightly higher velocity than the labeled trajectory, causing the score to decay. (*Center*) An oncoming motorcycle necessitates an avoidance maneuver. The prediction executes a similar lateral swerve at a comparable velocity but maintains a smaller lateral distance to the lane edge, resulting in a decayed score. (*Right*) The objective is to proceed straight at a moderate velocity to avoid a cyclist approaching from the left. The prediction is significantly slower than the optimal (Score 10.0) trajectories, leading to a decayed score.

Methods	RFS↑	ADE↓	# of Dataset	Training Schema	Model Arch.	Model Param.
MLP-based						
Swin-Trajectory	7.543	2.814	1	SFT	Swin Transformer	36M
Diffusion-based						
DiffusionLTF	7.717	2.977	4	SFT	DiffusionDrive	60M
UniPlan	7.779	2.986	2	SFT	DiffusionDrive	60M
MLLM-based						
Baseline	7.528	3.018	1	SFT	Gemini1 Nano	3B
AutoVLA	7.556	2.958	3	SFT+RL	Qwen2.5	3B
HMVLM	7.736	3.071	1	SFT	Qwen2.5	3B
Poutine	7.986	2.741	2	SFT+RL	Qwen 2.5	3B

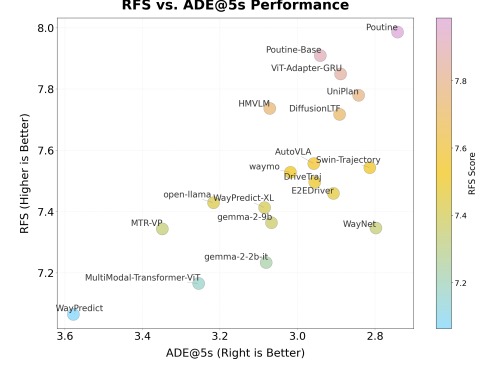


Table 2. WOD-E2E leaderboard submission results. **Left:** We summarize the results and configurations of selected representative methods among 3 categorical methodology (MLP-based, Diffusion-based, and MLLM-based). **Right:** We plot RFS vs ADE using 19 submissions. We only observe a mild positive correlation between RFS and ADE.

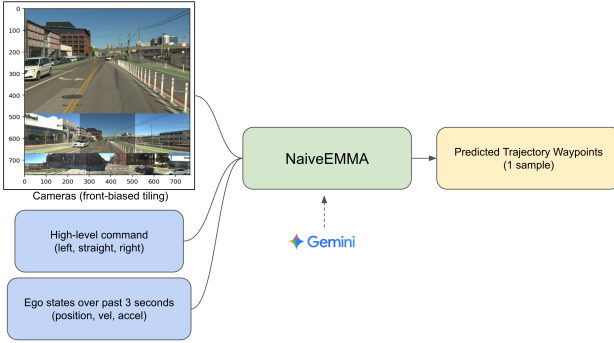


Figure 8. Architecture of NaiveEMMA, which serves as the challenge leaderboard baseline. NaiveEMMA is a highly simplified version of EMMA [14], fine-tuned from Gemini Flash [6]. The model takes as input all 8 camera images, 3 seconds of past ego-status history, and the high-level routing input. It then predicts the future trajectory in 5 seconds

Floor Scores for Predictions Far from Rater-Specified Trajectories (Figure 9c). (Left) Labeled trajectories demonstrate both lane-following and a lane-change. The prediction, however, proceeds at a high velocity in the unrated region between the two maneuvers, thereby receiving the floor score. (Center) While all labeled trajectories indicate a left turn, the prediction erroneously turns right. This significant deviation from the valid region results in the floor score. (Right) The labeled trajectories execute a right turn. The prediction proceeds straight, diverging completely from the specified maneuvers and receiving the floor score.

4.3. Benchmark Models

Since the release of WOD-E2E, we have received a significant number of submissions utilizing various models. These can be broadly divided into three categories: MLLM-based, Diffusion-based, and MLP-based models.

The following section details the methods that have released a detailed report, as shown in Table 2.

Swin-Trajectory [23] is a lightweight, MLP-based model. It uses a Swin Transformer [18] to extract image features from three front cameras and a simple MLP to directly predict waypoints. The model is lightweight and achieves a slightly better RFS (7.543) than the baseline.

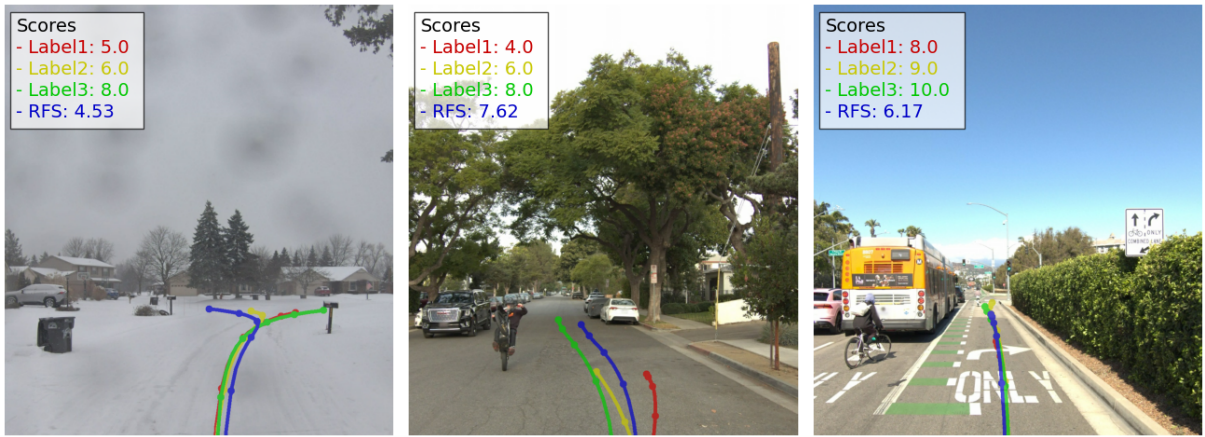
DiffusionLTF and **UniPlan** are both Diffusion-based models built on the DiffusionDrive [17] architecture. Their primary difference lies in the training datasets used: DiffusionLTF utilizes WOD-E2E, CARLA [9], NAVSIM [8], and WOD-Perception [26], whereas UniPlan is trained on WOD-E2E and nuPlan [3]. They achieve comparable performance, with RFS scores of 7.717 and 7.779, respectively.

Poutine [25], **HMVLM** [29], and **AutoVLA** [40] are all MLLM-based models that use Qwen2.5 as their backbone. They share a similar problem formulation, taking camera images and ego states as input modalities and outputting future waypoints as text. Additionally, all three models utilize Chain-of-Thought (CoT) reasoning before generating a trajectory. Despite these similarities, their results show a significant performance gap, with AutoVLA achieving an RFS of 7.556, HMVLM 7.736, and Poutine 7.986. The primary differences among these methods stem from:

- **Training data sources:** AutoVLA uses a combination of WOD-E2E, nuPlan, and nuScenes. In contrast, HMVLM is trained exclusively on WOD-E2E, whereas Poutine uses a blend of WOD-E2E and the CoVLA dataset.
- **CoT captioning style** These three models employ different methods for generating reasoning captions and use distinct prompt templates.
- **RL training:** HMVLM does not include any post-training reinforcement learning. AutoVLA incor-



(a) The model predicted future trajectory (blue) aligns well with one of the rater specified trajectories. The corresponding flat scores are assigned as the predictions fall within the trust region.



(b) The model predicted future trajectory (blue) deviates from rater specified trajectories. Since the predictions fall outside the trust regions, final scores are exponentially decayed.



(c) Floored scores (RFS=4) are assigned because predictions are far from any of the rater-specified trajectories.

Figure 9. Visualization for the RFS metric in 3 different conditions. **Top:** The model predictions fall within the trust region. **Middle:** The model predictions fall slightly outside the trust region. **Bottom:** The model predictions are far from any of the rater-specified trajectories.

porates GPRO with ADE as the reward, whereas Poutine uses GPRO with RFS as the reward.

4.4. Discussion of the Results

From the results of these benchmark models, below we discuss important research questions in E2E Driving.

Q1: Is extra data source with large data distribution gap helpful for E2E Driving?

It depends. For MLLM-based models, *e.g.* Poutine and AutoVLA, adding extra data source is helpful, resulting in an obvious performance gain. However, for Diffusion-based models, *e.g.* UniPlan and DiffusionLTF, only minor improvements are observed. A possible explanation for this divergence lies in the architectural capabilities of the MLLMs. We hypothesize that the CoT reasoning utilized by the MLLM-based models allows them to effectively leverage the diverse world knowledge and logical structures inherent in multiple datasets. This explicit reasoning mechanism helps the MLLMs internalize abstract driving knowledge that remains helpful regardless of the visual or geometric distribution shift between datasets. In contrast, diffusion-based models, which rely more directly on dense, pixel-level prediction, are more susceptible to performance degradation when combining visually disparate data sources.

Q2: Does a better ADE always lead to a better RFS?

No, a better ADE does not guarantee a better RFS. We plotted a few data points from different model submissions, showing both their ADE and RFS scores in the right figure of Table 2. While the two metrics exhibit a rough positive correlation, we observe numerous models where better ADE performance does not translate to a higher RFS score. For instance, WayNet achieves a highly competitive ADE of 2.8, ranking among the best submissions, yet its RFS is significantly lower than most other models. Conversely, HMMVLM demonstrates the opposite trend: its ADE is worse than many submissions, but its RFS ranks near the top. This clear divergence confirms the need for the RFS metric, as ADE alone is insufficient to evaluate a model’s true effectiveness in handling safety-critical, multi-modal long-tail scenarios.

Q3: Is RL effective in E2E Driving?

Yes, particularly when the reward is aligned with the target evaluation metric. Both Poutine and AutoVLA demonstrate performance improvements by incorporating RL into their post-training phase. However, the gain observed in Poutine is significantly more pronounced. The major reason for this difference lies in the reward signal used: Poutine utilizes RFS as its reward, which is directly aligned with our long-tail evaluation metric, whereas AutoVLA uses ADE. As the preceding research question demonstrated, ADE does not always maintain a strong positive correlation with RFS, making it a sub-optimal choice for optimizing performance on safety-critical scenarios.

5. Conclusion

In this paper, we introduced the Waymo Open Dataset for End-to-End Driving (WOD-E2E), a new benchmark specifically curated to evaluate end-to-end driving systems on challenging, long-tail scenarios. Existing datasets primarily feature nominal driving, failing to test true robustness. Our dataset provides 4,021 driving segments totaling approximately 12 hours, focusing on rare events that occur with a frequency of less than 0.03%.

To overcome the limitations of traditional metrics like ADE in these complex, multi-modal situations, we also introduced a new metric: Rater Feedback Score (RFS). RFS is a novel, human-aligned metric that evaluates a model’s trajectory against expert-annotated preference labels. Our benchmark analysis validates the dataset’s utility, demonstrating a clear divergence between ADE and RFS scores. This confirms that RFS is essential for capturing true performance in safety-critical scenarios. The benchmark results also highlight the promise of MLLM-based models and the effectiveness of reinforcement learning when its reward is directly aligned with the RFS metric.

We adopted an open-loop setup for WOD-E2E due to the prohibitive computational cost of realistic sensor simulation. While this presents a limitation, WOD-E2E advances the state-of-the-art for open-loop E2E driving benchmarks. Moreover, the long tail real world driving scenarios in our dataset could be applicable for testing the generalizability of high-fidelity simulators.

We hope that our WOD-E2E dataset and RFS metric will continue contributing to the development of more generalizable, robust, and safe autonomous driving agents.

References

- [1] Hidehisa Arai, Keita Miwa, Kento Sasaki, Kohei Watanabe, Yu Yamaguchi, Shunsuke Aoki, and Issei Yamamoto. Covla: Comprehensive vision-language-action dataset for autonomous driving. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1933–1943. IEEE, 2025. 1, 3, 4
- [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
- [3] Holger Caesar, Juraj Kabzan, Kok Seang Tan, Whye Kit Fong, Eric Wolff, Alex Lang, Luke Fletcher, Oscar Beijbom, and Sammy Omari. nuplan: A closed-loop ml-based planning benchmark for autonomous vehicles. *arXiv preprint arXiv:2106.11810*, 2021. 3, 11
- [4] Sergio Casas, Abbas Sadat, and Raquel Urtasun. Mp3: A unified model to map, perceive, predict and plan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 4
- [5] Ming-Fang Chang, John Lambert, Patsorn Sangkloy, Jagjeet Singh, Slawomir Bak, Andrew Hartnett, De Wang, Peter Carr, Simon Lucey, Deva Ramanan, et al. Argoverse: 3d tracking and forecasting with rich maps. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8748–8757, 2019. 3
- [6] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025. 5, 10, 11
- [7] Can Cui, Yupeng Zhou, Juntong Peng, Sung-Yeon Park, Zichong Yang, Prashanth Sankaranarayanan, Jiaru Zhang, Ruqi Zhang, and Ziran Wang. Vilad: A large vision language diffusion framework for end-to-end autonomous driving. *arXiv preprint arXiv:2508.12603*, 2025. 1
- [8] Daniel Dauner, Marcel Hallgarten, Tianyu Li, Xinshuo Weng, Zhiyu Huang, Zetong Yang, Hongyang Li, Igor Gilitschenski, Boris Ivanovic, Marco Pavone, Andreas Geiger, and Kashyap Chitta. Navsim: Data-driven non-reactive autonomous vehicle simulation and benchmarking. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. 1, 2, 3, 11
- [9] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. Carla: An open urban driving simulator. In *CoRL*, 2017. 11
- [10] Scott Ettinger, Shuyang Cheng, Benjamin Caine, Chenxi Liu, Hang Zhao, Sabeek Pradhan, Yuning Chai, Ben Sapp, Charles R Qi, Yin Zhou, et al. Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset. In *ICCV*, 2021. 1, 3, 4, 9
- [11] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The international journal of robotics research*, 2013. 3
- [12] 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 *CVPR*, 2023. 3, 4
- [13] Jyh-Jing Hwang, Henrik Kretzschmar, Joshua Manela, Sean Rafferty, Nicholas Armstrong-Crews, Tiffany Chen, and Dragomir Anguelov. Cramnet: Camera-radar fusion with ray-constrained cross-attention for robust 3d object detection. In *ECCV*, 2022. 1
- [14] Jyh-Jing Hwang, Runsheng Xu, Hubert Lin, Wei-Chih Hung, Jingwei Ji, Kristy Choi, Di Huang, Tong He, Paul Covington, Benjamin Sapp, Yin Zhou, James Guo, Dragomir Anguelov, and Mingxing Tan. EMMA: End-to-end multimodal model for autonomous driving. *Transactions on Machine Learning Research*, 2025. 1, 3, 10, 11
- [15] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Yu Qiao, and Jifeng Dai. Bevformer: Learning bird’s-eye-view representation from multi-camera images via spatiotemporal transformers. In *ECCV*, 2022. 1
- [16] Zhiqi Li, Zhiding Yu, Shiyi Lan, Jiahao Li, Jan Kautz, Tong Lu, and Jose M Alvarez. Is ego status all you need for open-loop end-to-end autonomous driving? In *CVPR*, 2024. 3
- [17] Bencheng Liao, Shaoyu Chen, Haoran Yin, Bo Jiang, Cheng Wang, Sixu Yan, Xinbang Zhang, Xiangyu Li, Ying Zhang, Qian Zhang, et al. Diffusiondrive: Truncated diffusion model for end-to-end autonomous driving. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025. 2, 3, 11
- [18] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2021. 11
- [19] Norman Mu, Jingwei Ji, Zhenpei Yang, Nate Harada, Haotian Tang, Kan Chen, Charles R Qi, Runzhou Ge, Kratarth Goel, Zoey Yang, et al. Most: Multi-modality scene tokenization for motion prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14988–14999, 2024. 4
- [20] Nigamaa Nayakanti, Rami Al-Rfou, Aurick Zhou, Kratarth Goel, Khaled S Refaat, and Benjamin Sapp. Wayformer: Motion forecasting via simple & efficient attention networks. In *ICRA*, 2023. 7
- [21] Chenbin Pan, Burhaneddin Yaman, Tommaso Nesti, Abhirup Mallik, Alessandro G Allievi, Senem Velipasalar, and Liu Ren. Vlp: Vision language planning for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14760–14769, 2024. 1, 3
- [22] Chenbin Pan, Burhaneddin Yaman, Senem Velipasalar, and Liu Ren. Clip-bevformer: Enhancing multi-view

- image-based bev detector with ground truth flow. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15216–15225, 2024. 1
- [23] Sungjin Park, Gwangik Shin, Jaeha Song, Sumin Lee, Hyukju Shon, Byounggun Park, Jinhee Na, Hawook Jeong, and Soonmin Hwang. Swin-trajectory: Technical report for 2025 waymo vision-based end-to-end driving challenge. 2, 11
- [24] Zhijie Qiao, Haowei Li, Zhong Cao, and Henry X Liu. Lightemma: Lightweight end-to-end multimodal model for autonomous driving. *arXiv preprint arXiv:2505.00284*, 2025. 3
- [25] Luke Rowe, Rodrigue de Schaetzen, Roger Girgis, Christopher Pal, and Liam Paull. Poutine: Vision-language-trajectory pre-training and reinforcement learning post-training enable robust end-to-end autonomous driving. *arXiv preprint arXiv:2506.11234*, 2025. 2, 11
- [26] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *CVPR*, 2020. 3, 4, 11
- [27] Pei Sun, Mingxing Tan, Weiyue Wang, Chenxi Liu, Fei Xia, Zhaoqi Leng, and Dragomir Anguelov. Swformer: Sparse window transformer for 3d object detection in point clouds. In *ECCV*, 2022. 1
- [28] Xiaoyu Tian, Junru Gu, Bailin Li, Yicheng Liu, Chenxu Hu, Yang Wang, Kun Zhan, Peng Jia, Xianpeng Lang, and Hang Zhao. Drivevlm: The convergence of autonomous driving and large vision-language models. In *CoRL*, 2024. 1, 3
- [29] Daming Wang, Yuhao Song, Zijian He, Kangliang Chen, Xing Pan, Lu Deng, and Weihao Gu. Hmvlm: Multistage reasoning-enhanced vision-language model for long-tailed driving scenarios. *arXiv preprint arXiv:2506.05883*, 2025. 2, 11
- [30] Hang Wang, Xin Ye, Feng Tao, Chenbin Pan, Abhirup Mallik, Burhaneddin Yaman, Liu Ren, and Junshan Zhang. AdaWM: Adaptive world model based planning for autonomous driving. In *The Thirteenth International Conference on Learning Representations*, 2025. 1
- [31] Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaese-model Pontes, et al. Argoverse 2: Next generation datasets for self-driving perception and forecasting. *arXiv preprint arXiv:2301.00493*, 2023. 3
- [32] Florian Wintel, Sigmund H Høeg, Gabriel Kiss, and Frank Lindseth. Using diffusion ensembles to estimate uncertainty for end-to-end autonomous driving. *arXiv preprint arXiv:2506.00560*, 2025. 3
- [33] Yichen Xie, Runsheng Xu, Tong He, Jyh-Jing Hwang, Katie Luo, Jingwei Ji, Hubert Lin, Letian Chen, Yiren Lu, Zhaoqi Leng, et al. S4-driver: Scalable self-supervised driving multimodal large language model with spatio-temporal visual representation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025. 1, 3, 4
- [34] Shuo Xing, Chengyuan Qian, Yuping Wang, Hongyuan Hua, Kexin Tian, Yang Zhou, and Zhengzhong Tu. Openemma: Open-source multimodal model for end-to-end autonomous driving. In *Proceedings of the Winter Conference on Applications of Computer Vision*, 2025. 1, 3
- [35] Runsheng Xu, Xin Xia, Jinlong Li, Hanzhao Li, Shuo Zhang, Zhengzhong Tu, Zonglin Meng, Hao Xiang, Xiaoyu Dong, Rui Song, et al. V2v4real: A real-world large-scale dataset for vehicle-to-vehicle cooperative perception. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023. 3
- [36] Runsheng Xu, Chia-Ju Chen, Zhengzhong Tu, and Ming-Hsuan Yang. V2x-vitv2: Improved vision transformers for vehicle-to-everything cooperative perception. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025. 1
- [37] Zhenhua Xu, Yujia Zhang, Enze Xie, Zhen Zhao, Yong Guo, Kwan-Yee K Wong, Zhenguo Li, and Hengshuang Zhao. Drivegpt4: Interpretable end-to-end autonomous driving via large language model. *RA-L*, 2024. 1, 3
- [38] Ekim Yurtsever, Jacob Lambert, Alexander Carballo, and Kazuya Takeda. A survey of autonomous driving: Common practices and emerging technologies. *IEEE access*, 2020. 1
- [39] 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. 3
- [40] Zewei Zhou, Tianhui Cai, Seth Z Zhao, Yun Zhang, Zhiyu Huang, Bolei Zhou, and Jiaqi Ma. Autovla: A vision-language-action model for end-to-end autonomous driving with adaptive reasoning and reinforcement fine-tuning. *arXiv preprint arXiv:2506.13757*, 2025. 11