MindDriver: Introducing Progressive Multimodal Reasoning for Autonomous Driving

Supplementary Material

1. Reasoning Annotation Pipeline

We develop a general pipeline for high-quality progressive multimodal reasoning data annotation. This pipeline includes three filtering processes to control the data quality across different aspects, along with a feedback-guided re-annotation strategy to iteratively refine filtered samples. Given inputs with the current camera, history video, driving command, and instruction, the powerful MLLM Qwen2.5-VL-72B [1] first generates a raw text CoT. The prompt for Qwen2.5-VL-72B is shown in Fig. 5. The text CoT then passes through three filters:

- **Format Filter:** This rule-based filter checks whether the text reasoning is composed of the four parts: (1) Scene Analysis, (2) Latent Risk Assessment, (3) Behavior Reasoning, and (4) Action Decision (including both direction and speed prediction).
- **Decision Filter:** It checks decision correctness by comparing the generated action to the GT decision derived from the GT trajectory. The process for generating ground truth (GT) labels is as follows: Leveraging statistical insights into dynamic vehicle parameters (e.g., velocity and acceleration) from the dataset, and informed by prior knowledge of real-world driving behaviors, we conducted clustering analysis on future vehicle trajectories. We experimented with different numbers of clusters (7, 10, and 49) to evaluate the effectiveness of trajectory pattern segmentation. The results revealed that smaller cluster counts led to highly imbalanced trajectory distributions, failing to capture the diversity of driving behaviors. To enhance model learning and generalization, we adopted a fine-grained trajectory categorization strategy. Specifically, for accelerating and decelerating vehicles, we used the 30th and 60th percentiles of their acceleration distributions as thresholds to differentiate behavior subcategories. A similar percentile-based thresholding approach was applied to left-turning and right-turning vehicles, based on their turning dynamics. This method enables more discriminative and behaviorally meaningful trajectory labeling, thereby supporting more accurate prediction modeling. The final selected meta actions are shown in the Tab. 1.
- **Logic Filter:** This filter evaluates the reasoning soundness of CoT. Instead of reusing Qwen2.5-VL-72B, we employ the more advanced text-LLM Qwen3-235B-A22B-Instruct [4] for robust logical validation and overcoming self-checking bias [5]. The prompt of Qwen3-235B-A22B-Instruct is illustrated in Fig. 1. To enable better understanding, we show an example of this logical quality check in Fig. 4. Based on Qwen3's robust logical analysis capabilities, a critical examination of the preliminary response from Qwen2.5VL-72B identified a broken causal chain in its reasoning process. The reasoning incorrectly conflates two distinct operational phases, firstly, valid recommendation for post-green-light behavior ("safe passage after the light turns green"), which is contextually appropriate; secondly, mandatory red-light behavior (complete stop and wait), which was not explicitly specified. This conflation erroneously applies the speed-adjustment guidance for post-green-light conditions to current red-light state, resulting in a conclusion that is fundamentally disconnected from the actual traffic scenario. Therefore, by applying similar logical validation, reasoning errors can be identified and corrected.

Table 1. Meta action type in desion filter.

Behavior	Meta Action Type
Direction Change	[Maintain Current Lane, Change Lane Left, Change Lane Right, Turn Left, Turn Right]
Speed Change	[Smooth Deceleration, Emergency Brake, Maintain Current Speed, Smooth Acceleration, Stop, Remain Stationary]

Feedback-guided Re-annotation. If any above filter fails, error feedback is returned as context to improve re-annotation. As shown in Tab. 2, the error feedback includes: (1) Format error: the detailed missing parts considering scene analysis, latent risk assessment, behavior reasoning, and action decision. (2) Decision error: Incorrect decisions vs. GT decisions, for both direction and speed decisions, and (3) Logic error: return the summarized logic errors generated by Qwen3-235B-A22B-Instruct. This feedback is combined with the raw CoT as input context for the next iteration. This process is shown in Fig. 2.

After that, the text CoT is concatenated with the ground truth future scene image and the trajectory with special tokens (<think>,<dream>,<answer>) to distinguish them, creating multimodal reasoning data. It is formatted as:

$$\text{<t>}\,\text{Tok}_{\text{Text CoT}}\text{</t><d>}\,\text{Tok}_{\text{Img}}\text{</d><a>}\,\text{Tok}_{\text{Traj}}\text{} \quad (1)$$

where <t>, <d>, <a> denotes <think>,<dream>,<answer> special tokens. $\text{Tok}_{\text{Text CoT}}$, Tok_{Img}, and Tok_{Traj} are the tokens of the text CoT, the dreamed image, and the predicted

1. Objective: You are an expert in detecting the quality of reasoning chains for autonomous driving models, tasked with determining whether the input reasoning process is correct or flawed.

2. Output Format: Reason + [Correct/Incorrect]

3. Judgment Criteria: Determine whether the input reasoning chain is logically correct and free from logical flaws. At the end of your response, specify the optimal direction and speed change for the ego vehicle as: <Direction Change>, <Speed Change>.
Direction Change (select one): [Maintain Current Lane, Change Lane Left, Change Lane Right, Turn Left, Turn Right].
Speed Change (select one): [Smooth Deceleration, Emergency Brake, Maintain Current Speed, Smooth Acceleration, Stop, Remain Stationary].

4. Input reasoning process:

Figure 1. Prompt for logical verification to Qwen3-235B-A22B-Instruct.

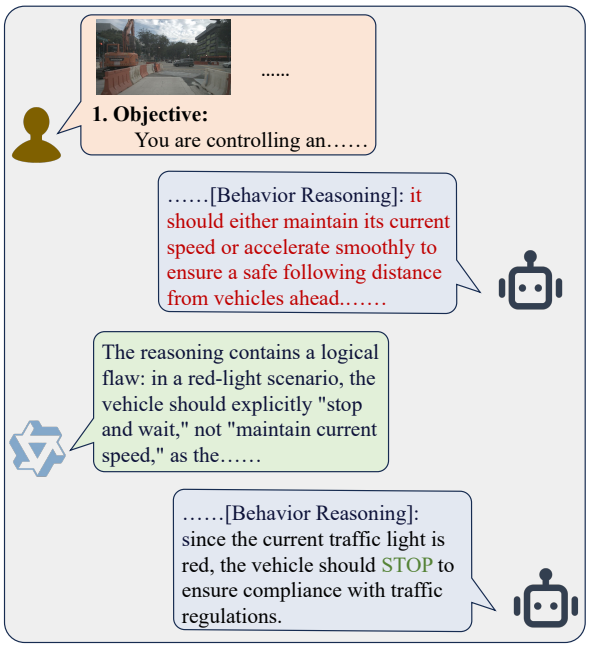

Figure 2. Combined COT with feedback for re-annotation.

trajectory.

2. Implementation Details

All experiments are conducted on 16 NVIDIA H20 GPUs (96 GB each). We employ Qwen2.5-VL-3B [1] as our base VLM. During SFT, we use 1×10^{-4} learning rate and batch size of 32, for 12 epochs (nuSences) and 6 epochs (Bench2Drive). For our feedback annotation pipline, We set the maximum number of iterative rounds to 3. The vision encoder of MindDriver is frozen, and the LLM is fully fine-tuned in the SFT stage. During progressive RFT, we use 3×10^{-6} learning rate and batch size 16, for 700 (stage 1) and 500 (stage 2) steps in nuSences, 1400 (stage 1) and 1000 (stage 2) steps in Bench2Drive. We set λ = 10 and α = 6 for L2 reward. In Eq. 7 of main paper, λ_1 and λ_1 are both set as 10 for strict format learning in RFT. The format reward r_{format} is to check whether the answer includes 6 parsed points (If yes, set 1; Otherwise set 0). The KL regularization weight β is set to 0.04. The generation parameter is with a sampling temperature as 1, top p as 1, and top k as 0 for diverse generation results in GRPO. In this stage, the vision encoder is frozen, and the LLM is fine-tuned using Low-Rank Adaptation (LoRA) [3] to reduce the training cost. The LoRA rank is set as 32. This RFT is implemented using VERL training framework.

Table 2. Specific feedback type in data auto-annotation.

Error Type	**Feedback Content Example**
Format Error	Missing Scene Analysis / ... Missing Action Decision part.
Decision Error	1.Direction decision error(GT: Change Lane Right; Prediction: Turn Right). 2.Speed decision error(GT: Smooth Deceleration; Prediction: Maintain Current Speed).
Logic Error	Reasoning is discontinuous ... consideration is incomplete

Model	Layers	Hidden size	Num Heads	Patch Size
Qwen2.5-VL	32	1280	16	14

Table 3. Model parameters of image encoder model (Vision Transformer).

Model	Layers	Hidden size	KV Heads	Head Size
Qwen2.5-VL	36	2048	2	128

Table 4. Model parameters of LLM.

3. Visualization of Dreamed Images

We selected two representative cases to illustrate the visualized outputs of MindDrivers predicted future scenes. As depicted in Fig. 3, the progressive reasoning process is clearly articulated in both examples.

In the first case, MindDriver first performs textual reasoning on the current scene. It identifies pedestrian motion trends, analyzes potential safety risks, and accordingly proposes future behavioral recommendations. Subsequently, based on the outcomes of this textual reasoning, the system generates a visual imagination of the future scene. In the resulting visualization, pedestrian positions can be observed

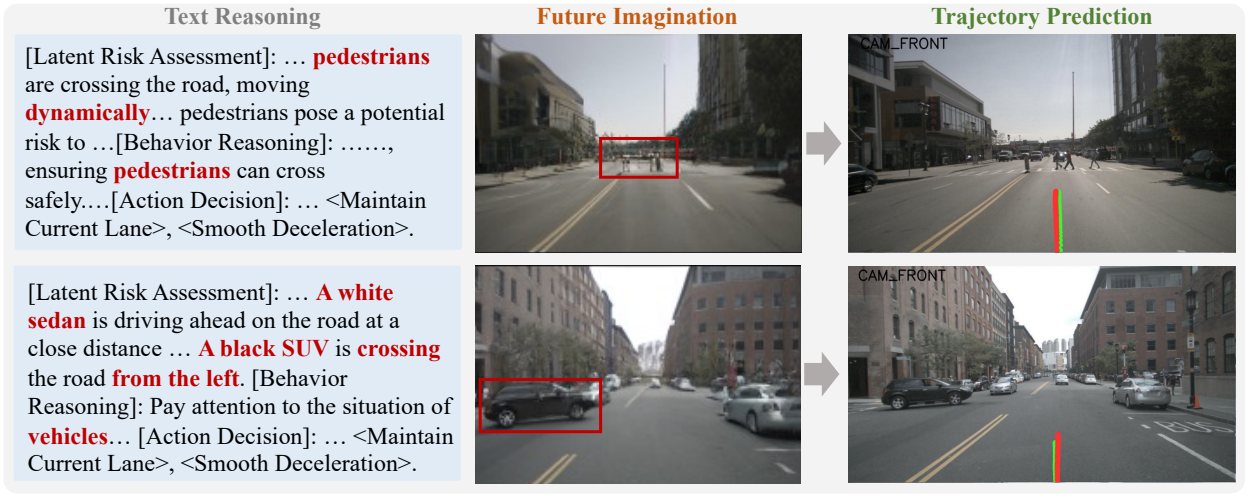

Figure 3. Qualitative Analysis of MindDriver. Red represents our predicted values, and green represents the ground truth (GT).

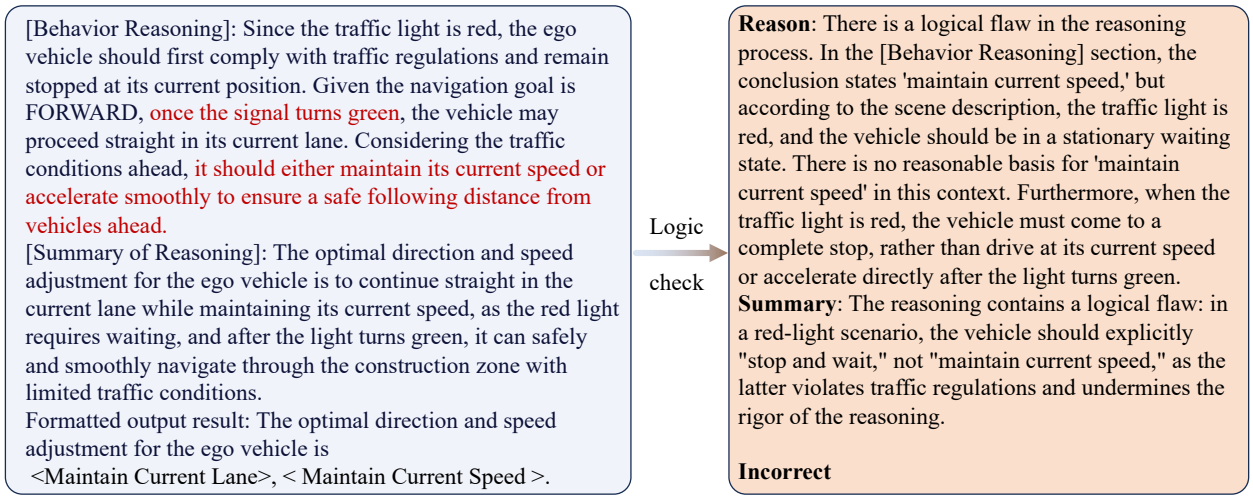

Figure 4. An example of a logical quality check.

to have changed, demonstrating that our method effectively establishes modeling capabilities for future spatiotemporal relationships. In the second case, a black SUV is crossing the road. If the ego vehicle maintains its current speed, a collision risk exists. Our method accurately imagines the motion state of the vehicle the black SUV is predicted to reach the center of the road after 0.5 seconds, thereby validating the effectiveness of our approach in both textual reasoning and future imagination.

4. More Visualization

We assess the model using closed-loop testing in the CARLA simulator. The model takes visual input in the form of four RGB images from the front-facing camera, encompassing a history of the past two seconds. MindDriver outputs a predicted two-second trajectory, which is then utilized by a PID controller to determine the control signals (throttle, brake, and steering) applied to the vehicle.

nuScenes results. In many challenging nuScenes [2] scenarios, such as nighttime, heavy rain, and high-curvature roads, MindDriver performs well and avoids collisions. As shown in Fig. 7. In the visualized comparisons, the green trajectory serves as the Ground Truth (GT), while the red trajectory illustrates the path planning executed by MindDriver. The results exemplify the model's resilience across a spectrum of real-world complexities. In the nighttime scenario (left), despite severe illumination changes and glare, the model maintains a steady path. Similarly, in the overcast and wet urban environment (center), MindDriver adeptly navigates through dynamic traffic, unaffected by the visual noise caused by rain. Most critically, the turning scenario

1. **Objective:**
   You are controlling an autonomous vehicle in a complex urban traffic environment with access to images fro m six camera perspectives and 2 seconds of historical footage from <CAM_FRONT>. Your task is to plan a s afe and reasonable driving trajectory for the next 3 seconds based on navigation targets. Navigation targets wi ll be provided in the following form: [FORWARD] / [RIGHT] / [LEFT] / [STOP], which should guide the pri oritization of actions.
2. **Process Overview:**
   Follow the steps outlined below for reasoning:
   a. **[Scene Description]**: Describe the weather, road conditions, visibility, and traffic signal states to determine drivable areas.
   b. **[Risk Object Identification]**: Identify 1-3 objects with the greatest impact on safety, analyze their position s and motion states, and update drivable areas.
   c. **[Reasoning Autonomous Driving Behavior]**: Propose three reasonable behavior combinations (direction + speed) and provide reasons.
   d. **[Summarizing Reasoning Results]**: Select the optimal behavior and output it in standard format.
3. **Scene Analysis:**
   Analyze the weather (sunny/rainy/foggy/night) and the impact of obstructions on visibility.
   Determine the traffic signal state most relevant to the current driving direction: [Red Light, Yellow Light, Gre en Light, Uncertain]. “Most relevant” refers to the traffic light controlling the right of way for the vehicle‘s la ne; if navigating a right turn, prioritize the right turn signal. Summarize the initial drivable area: [Large, Medi um, Small, Uncertain].**Large**: Multi-lane, unobstructed; **Medium**: Partially restricted; **Small**: Severely restric ted; **Uncertain**: Insufficient visibility.
4. **Latent Risk Assessment:**
   Use the vehicle's forward direction as the reference point, with <CAM_FRONT_LEFT> covering the left front area and <CAM_FRONT_RIGHT> covering the right front area.
   Combine multi-perspective and historical images to identify object categories (e.g., cars, buses, pedestrians, construction zones) and motion states (e.g., stationary, constant speed, accelerating toward, moving away).
   Select 1-3 highest-risk objects (priority: dynamic > static, lateral proximity > distant).
   Update drivable areas across perspectives, using the **smallest** value principle (if any direction is rated <Small>, the overall drivable area is <Small>).
5. **Behavior Reasoning:**
   Propose three safe and reasonable behavior combinations, each containing:
   Direction Change (select one): [Maintain Current Lane, Change Lane Left, Change Lane Right, Turn Left, Turn Right].
   Speed Change (select one): [Smooth Deceleration, Emergency Brake, Maintain Current Speed, Smooth Acceleration, Stop, Remain Stationary].
   Provide reasoning, considering traffic rules, obstacles, signal lights, and navigation targets.
6. **Summarizing Reasoning Results:**
   Select the optimal behavior from the proposed options and present it in the following format:
   Self-driving vehicle's optimal direction and speed change: <Direction>, <Speed>.
7. **Output Format:**
   [Scene Analysis]:
   <Description Results>, Drivable Area: <Large>
   [Latent Risk Assessment]:
   <CAM_FRONT>: <Object Category>, <Motion State>
   <CAM_FRONT_RIGHT>: <Object Category>, <Motion State>
   Combined Drivable Area: <Medium>
   [Behavior Reasoning]:
   <Maintain Current Lane>, <Smooth Deceleration>, <Reason>
   <Change Lane Left>, <Smooth Acceleration>, <Reason>
   <Turn Right>, <Smooth Deceleration>, <Reason>
   [Action Decision]:
   Self-driving vehicle's optimal direction and speed change: <Maintain Current Lane>, <Smooth Deceleration>.

Figure 5. Prompt for CoT annotation by Qwen2.5-VL-72B

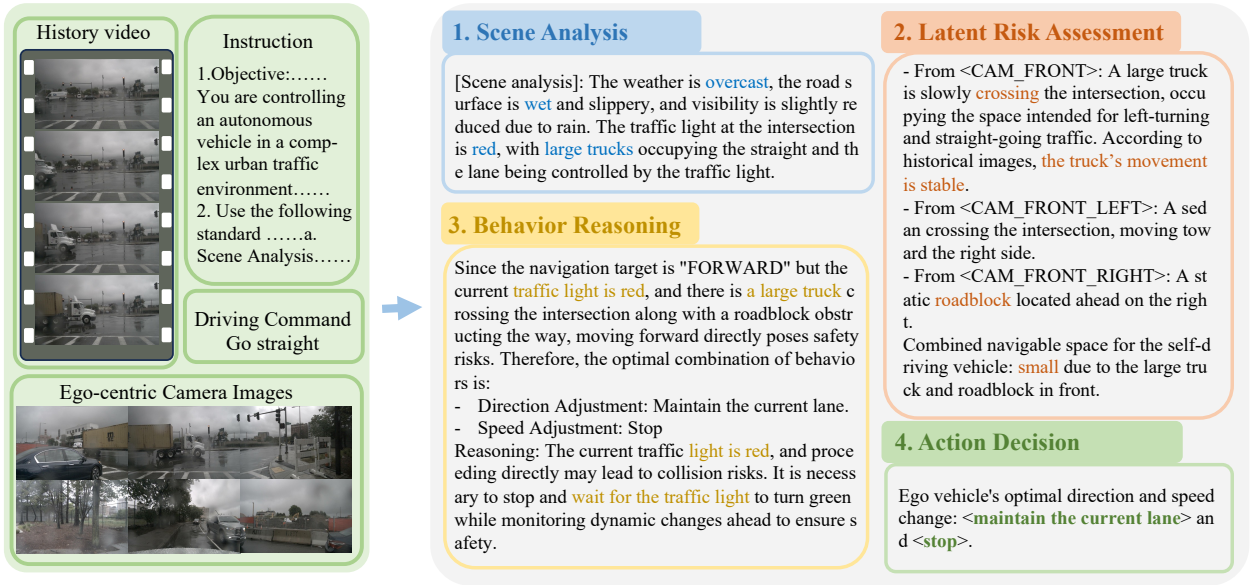

Figure 6. A complete sample of the annotation dataset.

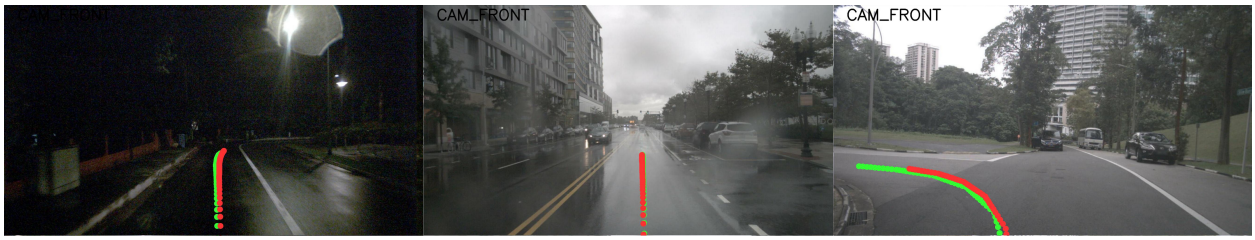

Figure 7. nuScenes results. The red trajectory is the prediction and the green one is the GT.

(right) demonstrates the advantage of our dynamics-driven approach. While traditional methods often struggle with the kinematic constraints of sharp turns, our model produces a smooth trajectory that nearly perfectly overlaps with the GT. This confirms that incorporating dynamics-related rewards significantly enhances the model's ability to handle complex geometric maneuvers with expert-level precision.

Bench2drive results. On the simulation dataset there are many scenarios that require risk reasoning. For example, pedestrians crossing the road, narrow roads, nighttime, and other extreme conditions, MindDriver successfully handles them all. As shown in Fig. 8. As illustrated in the visualization, our extensive closed-loop testing on the simulation dataset exposes the model to highly demanding scenarios. The dataset incorporates significant out-of-distribution (OOD) data, ranging from adverse weather conditions with wet road reflections (Top Row) to intense lighting variations. The model demonstrates remarkable robustness in safety-critical situations. For instance, it effectively anticipates and reacts to dynamic agents, such as vehicles cutting in and pedestrians jaywalking across the street (Middle Row). Notably, in complex intersection scenarios (Bottom Row), the model successfully obeys traffic rulesidentifying traffic lights and STOP signswhile making socially compliant decisions to stop and wait for multiple pedestrians, including children. This behavior highlights a significant improvement in planning logic and safety compared to previous SOTA methods like VAD.

References

[1] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 1, 2

[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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recog-*

Figure 8. Bench2drive results. From left to right indicates increasing timestamps.

nition, pages 11621–11631, 2020. 3

[3] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022. 2

[4] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. 1

[5] Yujian Yuan, Yanting Zheng, and Liangqiong Qu. Benchmarking radiology report generation from noisy free-texts. *IEEE Journal of Biomedical and Health Informatics*, 2025. 1