Think-Driver: From Driving-Scene Understanding to Decision-Making with Vision Language Models

Qiming Zhang[1], Meixin Zhu[1]*, and Hao (Frank) Yang[2]

[1] Hong Kong University of Science and Technology(Guangzhou), Guangzhou, China
qzhang255@connect.hkust-gz.edu.cn, meixin@ust.hk
[2] Johns Hopkins University, Baltimore, USA
haofrankyang@jhu.edu

Abstract. Autonomous driving has recently made impressive strides in both simulation and real-world performance, especially with end-to-end methods. However, these models often function as black boxes and lack explainability. The emergence of large language models (LLMs) offers a potential solution by combining modular autonomous driving with language explanations. Most recent LLM solutions convert driving input information into languages, which often require manually designed prompts and perhaps lead to suboptimal information efficiency. Vision language models(VLMs) can directly extract information from images but sometimes struggle with tasks involving continuous driving scene understanding and context reasoning. In this paper, we propose Think-Driver, a vision-language model that uses multi-view images to generate rational driving decisions and reasoning processes. Our model assesses perceived traffic conditions and evaluates the risks of current driving maneuvers, contributing to rational decisions. Through closed-loop experiments, Think-Driver outperforms other vision-language model baselines, producing interpretable driving decisions, which demonstrates its effectiveness and potential in future applications.

Keywords: Vision Language Model · Driving Risk Assessment · Decision-Making · Autonomous Driving

1 Introduction

End-to-end autonomous driving methods have become popular and advanced across various driving scenarios. Despite reducing information loss, these methods often lack interpretability for their final output. Also, human drivers can quickly comprehend their surroundings using visual information around the vehicle, assess the risks of current driving behavior, and make reasonable driving decisions. Such cognitive processes cannot be fully replicated by deep-learning-based models and end-to-end methods. Recently, large language models (LLMs) have been explored for tasks in autonomous driving, such as perception, prediction, and decision-making [3] [7] [16], due to their advantages in reasoning ability and explainability.

* Corresponding author: Meixin Zhu.

LLMs excel in reasoning and capturing latent representations from textual input, enabling them to process scene descriptions, generate reasoning, and make decisions. Recent works have proposed a similar framework incorporating LLMs with autonomous driving, using text descriptions of environments and vehicles as input. The LLM-based autonomous driving scheme usually converts sensor inputs as text modality and combines with the instruction part to guide LLMs to finish autonomous driving tasks like perception, prediction, and decision. However, current works with excellent performance largely depend on the great power of the GPT series models [11] [20], which are time-consuming to request. Also, these methods require transforming the nearby traffic and vehicle information into languages that may lose information to some extent and are not as intuitive as visuals information. Some vision language model(VLMs) methods [6] [12] only consider the front information of the ego vehicle and ignore other perspectives, leading to potential risks of final decisions.

To address the mentioned challenges, we propose Think-Driver, a VLM framework designed to understand driving scenes from the vehicle's six-view cameras, think at the behavioral level, and ultimately make informed decisions. To achieve this function, we consider historical visual and decision-reasoning information as the supplementary input. We fine-tuned vision-language models to enhance the model's understanding of visual information from CARLA's simulation environment and assess driving behavior risks. With multi-perspective images as input, our model can complete the entire reasoning process, from scene understanding and collision risk assessment to decision-making. In a closed-loop simulation environment-LimSim++ [6], our framework also demonstrated competitive driving performance compared to baselines. Generally, the contributions of our work are summarized as follows:

-We proposed a VLM-based framework, Think-Driver that takes multi-view images as input, makes explainable decisions, and learns from the driving experience.

-Our model systematically evaluates different driving environments, assessing driving behaviors and forming a thought chain from perception to decision-making, resulting in well-founded decisions.

-Through closed-loop simulation experiments, Think-Driver outperforms mainstream VLM baselines in both perception tasks and overall driving performance.

2 Related Works

Most recently, LLM-based autonomous driving has emerged with its satisfying performance in driving tasks and interpretable solutions. GPT-driver [11] design prompts covering vehicle and history information, then fine-tune ChatGPT to get planning results and noticeable objects in the environment, even performing better than UniAD [9]. Usual formulations of subtasks in autonomous driving are converted into language-based question answering. DriveGPT4 [22], with vision transformers(ViT) as vision encoder takes images and videos as input to finetune Llama2 to generate answers for perception and decision questions.

DiLU [20] proposed a knowledge-driven framework that makes GPT summarize and learn from past successful driving experiences in a highway environment. Nevertheless, it only considers text modality. To fully leverage the great power of LLMs, more tries on end-to-end autonomous driving frameworks. DriveMLM [18] and LMDrive [15] consider more modalities including images and point clouds, with language as instructions, to generate control signals directly, but lack interpretability.

Typical autonomous driving tasks are transformed into vision-based question-and-answer formats. It also allows for the integration of human driving experience as prior knowledge. To enhance knowledge and shorten the understanding gaps, nuSenseQA [13] and nuPrompt [21] provide human perception and understanding as labels to construct driving-language datasets, which can guide models to get better knowledge about the real world. DriveLM [16] considers perception, prediction, and decision problems together and transforms them into graph question answering to make models implement multi-tasks. DriveCoT [17] provides and organizes the chain of thought processes for autonomous driving to think in an organized manner. This not only helps to increase user trust in the system but also facilitates accountability in the event of a decision.

3 Methodology

In this section, we will illustrate our framework overall and introduce the vision language model in detail, at the third part how to instruct the model to think in a chain will be covered.

3.1 VLM-based Framework

We design a knowledge-driven autonomous driving framework with VLMs that can understand driving scenes and assess driving behaviors. As shown in Figure 1, our framework extracts images around the ego vehicle from the CARLA simulation environment, specifically from six perspectives: Front-left, Front, Front-right, Back-left, Back, and Back-right. Additionally, input information includes a language description of the task and driving instructions, which guide the model to output appropriate decisions and reasoning processes. Then, Think-driver outputs one driving action from a predefined available set of meta-actions, including Proceed, Acceleration, Brake, Left lane-change, and Right lane-change. After the model outputs a decision through reasoning, the ego vehicle in the simulation environment evolves by executing the trajectory corresponding to the meta-action. Since driving behaviors like lane-changing are temporally continuous, incorporating historical temporal information helps the model understand the dynamic surrounding environment. The visual and decision information from the previous frame is retrieved from memory for better behavior understanding. During each inference loop, the current frame along with the model's decisions and reasoning processes, will be stored and utilized as input for the next model inference.

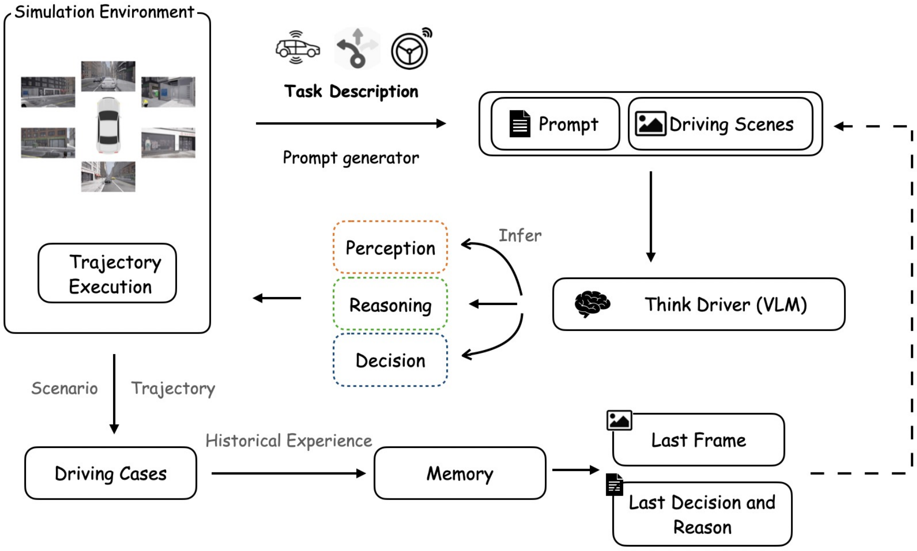


Fig. 1: VLM-based Decision Framework: Our model uses visual information from various perspectives to generate safe and reasonable driving decisions. The simulation environment updates the next state based on the trajectory corresponding to these decisions. To enhance the model's understanding of the temporal aspects of driving behavior, the VLM input in each iteration includes both the current visual information and the decision reasoning from the previous frame extracted from the stored memory.

3.2 Think-Driver Model

To construct our vision-language model, we follow the vision instruction tuning format [10] to fine-tune large language models. Mainstream VLMs use Vision Transformers (ViT) as vision encoders, multi-layer perceptrons (MLP) as projectors, and large language models as backbones. The key factor for LLMs to process image tokens is aligning the image data into the language distribution space.

For image input, we use the pre-trained InternViT [2] as our vision encoder. This model, pre-trained with Clip techniques [14] on a large amount of image-text datasets, can map vision features into an image-language distribution space. Considering the multi-view information input, we introduced the multi-view visual-text Alignment (MV-alignment) approach. To help the model better understand the positional information represented by different image perspectives, we performed encoding and mapping for each view with specific MLPs, the hidden feature F_i of each view patch P_i can be computed as defined.

$$F_i = f_{MLP_i}(f_{ViT}(P_i)) \tag{1}$$

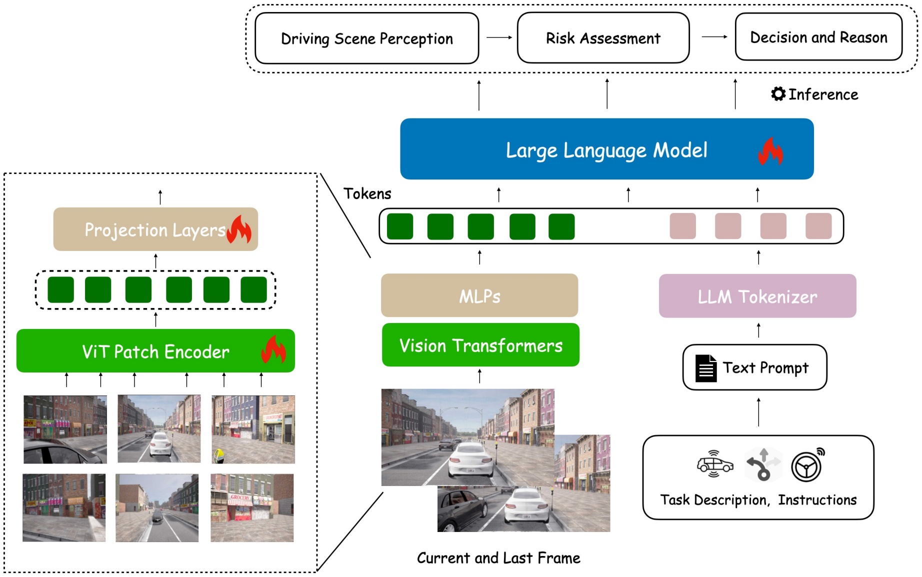


Fig. 2: Model architecture: Our model takes the surrounding view images and instruction text as input. The ViT encoders and projection layers will align image patches of multi-views and the last frame into vision tokens. The instruction text includes the task definition and the last decision with the reasoning process. With the combination of image and text tokens, all information is fed into large language models to generate scene descriptions, thinking processes from nearby traffic and driving behaviors, available decisions, and reasoning. During the fine-tuning process, the ViT encoder, MLP connector, and LLM are all tuned with QLoRA techniques.

All MLP parameters and q,k,v of ViTs are trained in this module during the tuning process. Also, we expanded the model's vocabulary with specific placeholders to represent the hidden feature F_i of different perspectives. For instance, $< front\_left >$ stands for the encoded and mapped features of the visual information from the front-left view of the vehicle. Hence, this approach helps the VLM model more efficiently utilize visual information and generate accurate spatial position descriptions. Similarly, we retained the overall image information by using $< frame >$ to refer to the features of the stitched images, reducing potential information loss. This approach ensures that the model captures the spatial relationships between different viewpoints while maintaining the global context of the scene. To enable the LLM to differentiate between two frames of images, we use specially defined language tags to represent historical information and the current image: $<$ *current frame* $>$ and $<$ *last frame* $>$ to represent the temporal information respectively.

For the textual component, the model tokenizes the constructed prompts into language tokens with the LLM tokenizer. The image and text tokens are then combined and fed into large language models, as shown in Figure 2. We utilize the advanced open-source LLM, Interlm2-chat, as our backbone language model, fine-tuning it with Q-LoRA [4], a parameter-efficient method that modifies the self-attention layers with low-rank matrices to reduce the number of trainable parameters. During supervised fine-tuning, VQA driving tasks are formatted as next-token generation tasks. The model generates answer texts autoregressively by optimizing the cross-entropy function for the answer label. Considering visual token feature T_{F_i} and text instruction token T_t as input, the loss function can be defined as:

$$\mathbb{L}(Y, \hat{Y}) = -\sum_{j=1}^{n} P(Y_j) \log P(\hat{Y}_j | \hat{Y}_{1:j-1}; T_{F_i}, T_t) \tag{2}$$

Y_j and $\hat{Y}_j$ are the j-th label and predicted tokens, with $\hat{Y}_{1:j-1}$ as previously generated tokens.

3.3 CoT Instruction-tuning

Human-like driving involves a sequential and gradual thinking process. A driver typically observes the surrounding environment—traffic lights, facilities, nearby vehicles, and pedestrians —before deciding on actions based on traffic rules and potential collision risks. For future actions, such as changing lanes or following another vehicle, the driver also considers possible future changes. Finally, a decision is made after considering all factors comprehensively. To facilitate LLM reasoning in such a manner, we incorporate the Chain of Thought (CoT) mechanism [19], as illustrated in Figure 3, during model fine-tuning.

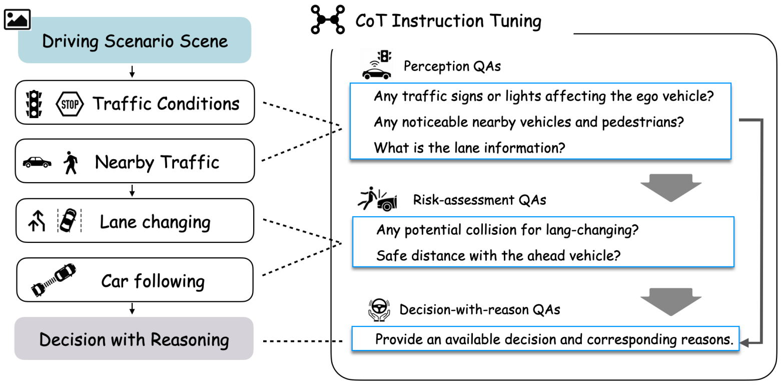
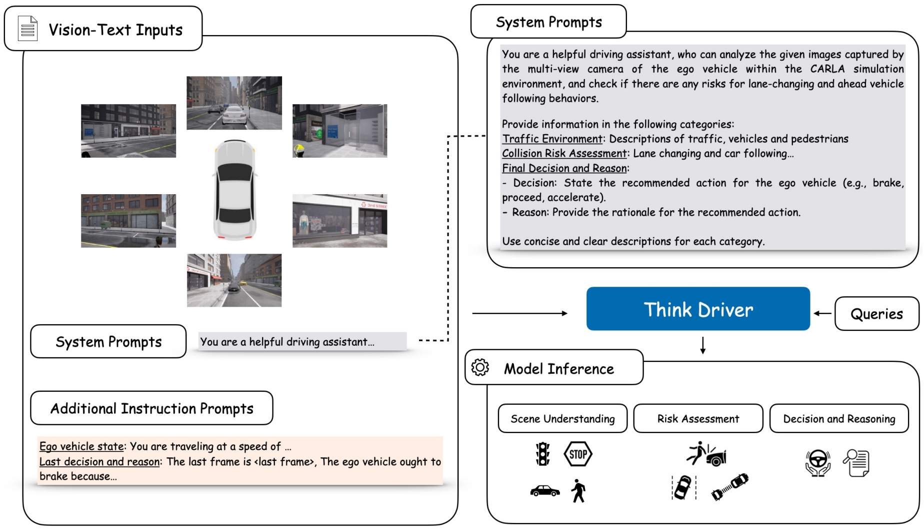

Driving Scenario Scene
Traffic Conditions
Nearby Traffic
Lane changing
Car following
Decision with Reasoning
CoT Instruction Tuning
Perception QAs
Any traffic signs or lights affecting the ego vehicle?
Any noticeable nearby vehicles and pedestrians?
What is the lane information?
Risk-assessment QAs
Any potential collision for lang-changing?
Safe distance with the ahead vehicle?
Decision-with-reason QAs
Provide an available decision and corresponding reasons.

Fig. 3: CoT Process in Think-Driver: To enhance the thinking abilities based on the perception information for decision tasks, we fine-tune our model through constructed CoT-instruction QA datasets under multi-turn dialogue settings. For risk prediction and decision questions, the model will summarize and reason through generated information, and then make a reasonable analysis.



You are a helpful driving assistant...

Ego vehicle state: You are traveling at a speed of ...
Last decision and reason: The last frame is <last frame>, The ego vehicle ought to brake because...

You are a helpful driving assistant, who can analyze the given images captured by the multi-view camera of the ego vehicle within the CARLA simulation environment, and check if there are any risks for lane-changing and ahead vehicle following behaviors.

Provide information in the following categories:
Traffic Environment: Descriptions of traffic, vehicles and pedestrians
Collision Risk Assessment: Lane changing and car following...
Final Decision and Reason:
- Decision: State the recommended action for the ego vehicle (e.g., brake, proceed, accelerate).
- Reason: Provide the rationale for the recommended action.

Use concise and clear descriptions for each category.

Fig. 4: Vision-Text Prompt Design: Multi-view images and the historical vision information are organized with additional text instructions, which include the current state of the ego vehicle and the last driving experience. System prompts are tailored to instruct models to leverage their knowledge and generate related descriptions.

The CoT mechanism guides the LLM through step-by-step reasoning: identifying signs, vehicles, and pedestrians in the traffic environment that could impact vehicle status; assessing collision risks associated with available driving behaviors; and determining the most appropriate driving decision. Empirically, this structured CoT prompt format is tailored to convey diverse driving information and reasoning processes, boost the model's comprehension, and refine the analysis accuracy. Each item of prompts is illustrated below, and the complete content is shown in Figure 4.

- **System Prompts**. To enhance the integration of LLMs into driving decision tasks, system prompts are structured with clear task descriptions. Task settings explicitly outline the role of LLMs and leverage their pre-existing understanding of the autonomous driving domain.
- **Ego vehicle state**. The ego vehicle state describes the speed, acceleration, and location of the ego vehicle at the current frame, which provides direct information for understanding its driving state.
- **Last decision and reasoning**. The historical decision and reasoning information are mentioned for a better understanding of its driving behavior. Also, the last frame is included and depicted as sign $< last frame >$.

We employed a mixed data fine-tuning approach to enable our model to output all relevant information in a single inference, while maintaining the ability to handle single-driving task queries. CoT-instruction QA pairs are constructed with multi-turn dialogue samples, especially for the reasoning risks- driving risk assessment and decision with reasoning. Then, we combined single-driving task QA samples with CoT-instruction QAs to form the fine-tuning dataset. This approach allows our model to output the reasoning process from perception to decision-making in a single inference, while also retaining the ability to handle single driving task queries.

4 Experiments and Analysis

In this section, the dataset and experiment settings are covered in detail. Also, our model's performance in the close-loop evaluation and driving case studies are included for the analysis of the ability of our models.

4.1 Dataset and Training Details

We reconstructed the multi-modal DriveCoT dataset [17], collected from the Town12 Map in the Carla environment, which comprises various driving scenarios in different environments: Traffic Negotiation, Ahead Vehicle Break, Pedestrian Crossing, and Lane Change. Also, it consists of 1058 scenarios with 36k labeled samples. We adjust the decision labels as four meta-actions: proceed, acceleration, deceleration, left lane-change, and right lane-change. Considering that our model only utilizes visual modality inputs, we retained information about vehicles visible within the RGB cameras, excluding those detectable only by lidars. The coordinates of nearby vehicles and pedestrians are converted into state and relative position descriptions to enhance the model's spatial awareness. By fine-tuning our model with a reconstructed dataset, Think-Driver gains the ability to understand driving environment information, reason effectively, and make decisions. For the model training, the dataset is split into training, validation, and testing sets at a ratio of 70%, 20%, and 10%, respectively. To fine-tune our model efficiently, we utilize Q-LoRA to fine-tune the weights of ViT, MLP projectors, and LLM weights based on 4 RTX-4090s. The model is trained with 10 epochs using AdamW optimizer with the start learning rate 2e-4 and a cosine annealing scheduler. For model inference in all experiments, we take the zero-shot settings, leverage the same query prompt, and set the same parameters for the output of LLMs, with temperature=0.7, max tokens = 1024, and seed=50, in order to reproduce the results.

4.2 Close-loop Experiments

In this section, we assess our model's overall driving performance in the simulation environment-CARLA [5], focusing on instruction following, driving efficiency, driving comfort, and safety.

Experiment settings We take Limsim++ as our closed-loop simulation environment, which is specifically designed for vision large language models. It is co-simulated with CARLA and SUMO with abundant city driving scenarios. Here we selected Town06 as our Map and No.50 as the controlled vehicle, other vehicles are generated in simulation settings. In the close-loop evaluations, Think-Driver takes the role of the decision-maker of the vehicle. For the limited speed, 0.1m/s is the lowest value and the max speed is referred to as the road limit. To compare the performance, we implement 20-time tests for each VLM to take the average results to evaluate that driving performance. For the driving actions, we separated driving decisions into several categories, which is more direct and effective for LLMs to make decisions: "Acceleration" - accelerate the vehicle; "Deceleration" - decelerate the vehicle; "Proceed" - remain in the current lane with current speed; "Turn-left" - change lane to the left of the current lane; "Turn-right" - change lane to the right of the current lane. Different from the Limsim++ settings, we considered multi-viewpoints around the vehicle as visual inputs and adjusted the prompt for each inference accordingly. For other VLM baselines, we combined the multi-view images into a single image as input to retain the full range of visual information. This approach allows us to maintain comprehensive spatial awareness while tailoring the input to better suit the model's reasoning process, potentially improving the model's performance on complex driving tasks.

Evaluation Metrics We take the same metric settings as LimSim++, which cover the route completion, driving scores, and successful rate. The driving performance comprehensively considers route completion R and driving score S.

- **Route Completion** The route completion R indicates the ratio of completed route length to total length, shown as:

$$R = \frac{L_{completed}}{L_{total}} \tag{3}$$

  where $L_{completed}$ represent the successful route distance and L_{total} is the sum of set route distance.
- **Driving Score** Driving score S consists of assessments of driving efficiency, ride comfort, and driving safety, which is:

$$S = \alpha^{\lambda_1}\beta^{\lambda_2}\gamma^{\lambda_3}(k_1 r_c + k_2 r_e + k_3 r_s) \tag{4}$$

  The α, β, γ are penalty terms that are denoted for collision, signal, and speed violations. Here we choose 0.6,0.7,0.9 respectively.
  - **Ride Comfort** Lateral and longitudinal accelerations, as well as jerking, are considered for evaluating ride comfort. It can be computed as:

$$r_c = \frac{(S_x(a) + S_x(j) + S_y(a) + S_y(j))}{4} \tag{5}$$

    In this equation, $S_x(a,j), S_y(a,j)$ represent the lateral accelerations and jerks, longitudinal accelerations and jerks.

- **Driving Effciency** The controlled vehicle should reach the speed limit of the road to reach the goal as soon as possible.

$$r_e = \begin{cases} 1.0, & \text{if } v_e \geq v^* \\ v_e/v^*, & \text{else} \end{cases} \tag{6}$$

  v_e is the speed of the vehicle, and v^* is a value between the average speed and limited speed.

- **Driving Safety** For the drive safety part, it is measured by Time to Conflict(TTC). If the TTC drops below a certain threshold, it indicates potential risks that require penalties. The method for calculating driving safety is detailed below:

$$r_s = \begin{cases} 1.0, & \text{if } \tau_e \geq \tau_{threhold} \\ \tau_e/\tau_{threhold}, & \text{else} \end{cases} \tag{7}$$

  where $\tau_{threhold}$, τ_e represents the TTC of the threshold and the ego vehicle.

- **Success Rate** Success rate S_r represents the number of samples that successfully complete the driving task, where the total number of samples is N, and the number of successful samples is N_s. It is computed as follows:

$$S_r = \frac{N_s}{N} \times 100\% \tag{8}$$

Comparison Results In this part, we compare our model with baselines, Llava series [10] and MiniCPM-V [8], as well as close-source vision-language models GLM-4V [23], GPT-4v [1], under the same framework. The LimSim++ environment executes the corresponding defined trajectories according to the meta-action generated by VLMs. The comparison results are shown in Table 1.

Models	Model Size	RC(% ↑)	DS ↑	SR(% ↑).
MiniCPM-V2.5	8b	56.7	46.5	55.0
Llava 1.6-Llama3	8b	51.2	48.9	40.0
GLM-4V	-	46.4	52.9	25.0
GPT-4V	-	57.3	51.6	45.0
Think-Driver	8b	**71.3**	**65.0**	**90.0**

Table 1: Close-loop experiments compared with VLM baselines. RC: Route Completion, DS: Driving Score, SR: Success Rate. In each evaluation turn, all VLM models take zero-shot reasoning and output available actions.

From Table 1, we can find that the Think-Driver model demonstrated the highest performance across all metrics, with a Route Completion Percentage of

71.3%, a Driving Score of 65.0, and a Successful Rate of 90%. This indicates that Think-Driver not only excels in following instructions but also maintains a high driving performance, making it the most robust model among those evaluated. Trained on CoT datasets, Think-Driver is capable of step-by-step reasoning from perception to decision, producing well-supported and interpretable behavior decisions. This reasoning process reduces potential collision risks and contributes to higher driving scores.

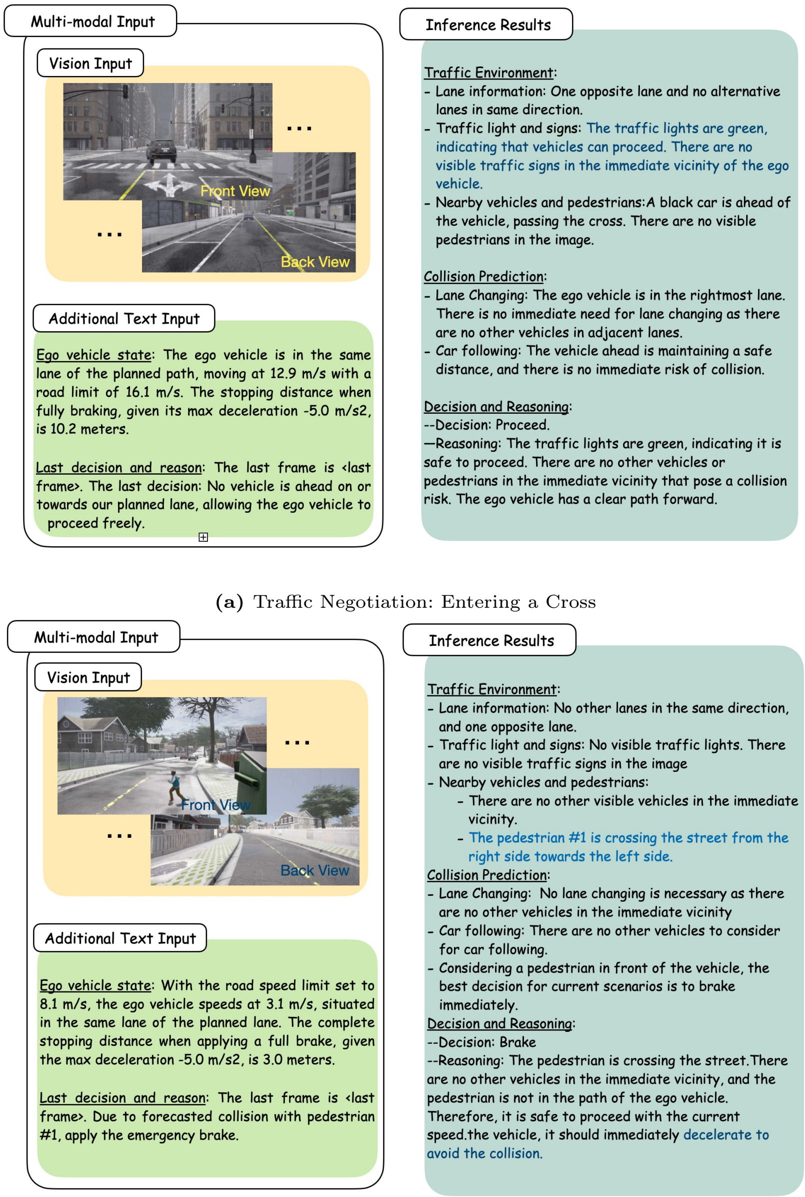
4.3 Case Studies

For case studies, we selected two typical driving scenarios to demonstrate the perception and reasoning abilities of the Think-Driver, as shown in Figure 5. The organized inference results are listed at the right part of each case. From two cases, our model can extract surrounding traffic environment information, as well as the positions of vehicles and pedestrians, from images. Based on this information, it assesses the potential risks of lane changes and the following behavior, outputting reasonable driving actions and reasoning processes.

For the traffic negotiation scenario in Figure 5(a), the model accurately determined the driving behavior of the black car ahead as "exiting the intersection", through input images and historical temporal information. It also detected the green traffic light ahead, allowing the vehicle to proceed straight into the intersection. Considering that there are no traffic conditions ahead affecting the vehicle, the model suggests that maintaining the current speed and passing through the intersection is a reasonable choice. For the pedestrian crossing scenario in Figure 5(b), our model also accurately identified the potential collision risk of a pedestrian crossing the road ahead, outputting a decision to brake and decelerate.

5 Conclusion and Limitations

In this work, we introduced a novel VLM-based framework, Think-Driver, which enhances autonomous driving by integrating perception, prediction, and decision-making into a coherent thought process. This systematic approach improves adaptability and accuracy in real-world scenarios. Our experiments show that Think-Driver performs competitively with other VLM baselines, demonstrating the value of combining VLM with explainable end-to-end methods in autonomous driving. However, there are areas for improvement. Future work could explore extreme scenarios and long-tail samples to further test the model. Additionally, integrating spatio-temporal methods could improve the model's understanding of motion and spatial relationships. Exploring outputs like speed, acceleration, or predicted trajectories could also lead to more precise control.



Multi-modal Input

Vision Input

Additional Text Input

Ego vehicle state: The ego vehicle is in the same lane of the planned path, moving at 12.9 m/s with a road limit of 16.1 m/s. The stopping distance when fully braking, given its max deceleration -5.0 m/s2, is 10.2 meters.

Last decision and reason: The last frame is <last frame>. The last decision: No vehicle is ahead on or towards our planned lane, allowing the ego vehicle to proceed freely.

Inference Results

Traffic Environment:
- Lane information: One opposite lane and no alternative lanes in same direction.
- Traffic light and signs: The traffic lights are green, indicating that vehicles can proceed. There are no visible traffic signs in the immediate vicinity of the ego vehicle.
- Nearby vehicles and pedestrians:A black car is ahead of the vehicle, passing the cross. There are no visible pedestrians in the image.

Collision Prediction:
- Lane Changing: The ego vehicle is in the rightmost lane. There is no immediate need for lane changing as there are no other vehicles in adjacent lanes.
- Car following: The vehicle ahead is maintaining a safe distance, and there is no immediate risk of collision.

Decision and Reasoning:
--Decision: Proceed.
—Reasoning: The traffic lights are green, indicating it is safe to proceed. There are no other vehicles or pedestrians in the immediate vicinity that pose a collision risk. The ego vehicle has a clear path forward.

(a) Traffic Negotiation: Entering a Cross

Multi-modal Input

Vision Input

Additional Text Input

Ego vehicle state: With the road speed limit set to 8.1 m/s, the ego vehicle speeds at 3.1 m/s, situated in the same lane of the planned lane. The complete stopping distance when applying a full brake, given the max deceleration -5.0 m/s2, is 3.0 meters.

Last decision and reason: The last frame is <last frame>. Due to forecasted collision with pedestrian #1, apply the emergency brake.

Inference Results

Traffic Environment:
- Lane information: No other lanes in the same direction, and one opposite lane.
- Traffic light and signs: No visible traffic lights. There are no visible traffic signs in the image
- Nearby vehicles and pedestrians:
  - There are no other visible vehicles in the immediate vicinity.
  - The pedestrian #1 is crossing the street from the right side towards the left side.

Collision Prediction:
- Lane Changing: No lane changing is necessary as there are no other vehicles in the immediate vicinity
- Car following: There are no other vehicles to consider for car following.
- Considering a pedestrian in front of the vehicle, the best decision for current scenarios is to brake immediately.

Decision and Reasoning:
--Decision: Brake
--Reasoning: The pedestrian is crossing the street.There are no other vehicles in the immediate vicinity, and the pedestrian is not in the path of the ego vehicle. Therefore, it is safe to proceed with the current speed.the vehicle, it should immediately decelerate to avoid the collision.

(b) Pedestrian Crossing

Fig. 5: Cases Studies for Traffic Negotiation and Pedestrian Crossing Scenarios

Acknowledgements

This study is supported by the National Natural Science Foundation of China under Grant 52302379, Guangdong Provincial Natural Science Foundation-General Project with Grant 2024A1515011790, Guangzhou Basic and Applied Basic Research Projects under Grants 2023A03J0106 and 2024A04J4290, Guangdong Province General Universities Youth Innovative Talents Project under Grant 2023KQNCX100, Guangzhou Municipal Science and Technology Project 2023A-03J0011, Nansha District Key R&D Project 2023ZD006.

References

1. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)
2. Chen, Z., Wang, W., Tian, H., Ye, S., Gao, Z., Cui, E., Tong, W., Hu, K., Luo, J., Ma, Z., et al.: How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites. arXiv preprint arXiv:2404.16821 (2024)
3. Cui, C., Ma, Y., Cao, X., Ye, W., Zhou, Y., Liang, K., Chen, J., Lu, J., Yang, Z., Liao, K.D., et al.: A survey on multimodal large language models for autonomous driving. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 958–979 (2024)
4. Dettmers, T., Pagnoni, A., Holtzman, A., Zettlemoyer, L.: Qlora: Efficient finetuning of quantized llms. Advances in Neural Information Processing Systems **36** (2024)
5. Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V.: Carla: An open urban driving simulator. In: Conference on robot learning. pp. 1–16. PMLR (2017)
6. Fu, D., Lei, W., Wen, L., Cai, P., Mao, S., Dou, M., Shi, B., Qiao, Y.: Limsim++: A closed-loop platform for deploying multimodal llms in autonomous driving. arXiv preprint arXiv:2402.01246 (2024)
7. Guo, X., Zhang, Q., Peng, M., Zhua, M., et al.: Explainable traffic flow prediction with large language models. arXiv preprint arXiv:2404.02937 (2024)
8. Hu, S., Tu, Y., Han, X., He, C., Cui, G., Long, X., Zheng, Z., Fang, Y., Huang, Y., Zhao, W., et al.: Minicpm: Unveiling the potential of small language models with scalable training strategies. arXiv preprint arXiv:2404.06395 (2024)
9. Hu, Y., Yang, J., Chen, L., Li, K., Sima, C., Zhu, X., Chai, S., Du, S., Lin, T., Wang, W., et al.: Planning-oriented autonomous driving. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 17853–17862 (2023)
10. Liu, S., Cheng, H., Liu, H., Zhang, H., Li, F., Ren, T., Zou, X., Yang, J., Su, H., Zhu, J., et al.: Llava-plus: Learning to use tools for creating multimodal agents. arXiv preprint arXiv:2311.05437 (2023)
11. Mao, J., Qian, Y., Zhao, H., Wang, Y.: Gpt-driver: Learning to drive with gpt. arXiv preprint arXiv:2310.01415 (2023)
12. Park, S., Lee, M., Kang, J., Choi, H., Park, Y., Cho, J., Lee, A., Kim, D.: Vlaad: Vision and language assistant for autonomous driving. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 980–987 (2024)

13. Qian, T., Chen, J., Zhuo, L., Jiao, Y., Jiang, Y.G.: Nuscenes-qa: A multi-modal visual question answering benchmark for autonomous driving scenario. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 4542–4550 (2024)
14. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021)
15. Shao, H., Hu, Y., Wang, L., Waslander, S.L., Liu, Y., Li, H.: Lmdrive: Closed-loop end-to-end driving with large language models. arXiv preprint arXiv:2312.07488 (2023)
16. Sima, C., Renz, K., Chitta, K., Chen, L., Zhang, H., Xie, C., Luo, P., Geiger, A., Li, H.: Drivelm: Driving with graph visual question answering. arXiv preprint arXiv:2312.14150 (2023)
17. Wang, T., Xie, E., Chu, R., Li, Z., Luo, P.: Drivecot: Integrating chain-of-thought reasoning with end-to-end driving. arXiv preprint arXiv:2403.16996 (2024)
18. Wang, W., Xie, J., Hu, C., Zou, H., Fan, J., Tong, W., Wen, Y., Wu, S., Deng, H., Li, Z., et al.: Drivemlm: Aligning multi-modal large language models with behavioral planning states for autonomous driving. arXiv preprint arXiv:2312.09245 (2023)
19. Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q.V., Zhou, D., et al.: Chain-of-thought prompting elicits reasoning in large language models. Advances in neural information processing systems **35**, 24824–24837 (2022)
20. Wen, L., Fu, D., Li, X., Cai, X., Ma, T., Cai, P., Dou, M., Shi, B., He, L., Qiao, Y.: Dilu: A knowledge-driven approach to autonomous driving with large language models. arXiv preprint arXiv:2309.16292 (2023)
21. Wu, D., Han, W., Wang, T., Liu, Y., Zhang, X., Shen, J.: Language prompt for autonomous driving. arXiv preprint arXiv:2309.04379 (2023)
22. Xu, Z., Zhang, Y., Xie, E., Zhao, Z., Guo, Y., Wong, K.K., Li, Z., Zhao, H.: Drivegpt4: Interpretable end-to-end autonomous driving via large language model. arXiv preprint arXiv:2310.01412 (2023)
23. ZhipuAi: General language model 3, 4-vision (2024), `https://open.bigmodel.cn/dev/howuse/introduction`, date2024-01-26