A Lightweight Vision-Language Model Pipeline for Corner-Case Scene Understanding in Autonomous Driving

Ying Cheng[1], Min-Hung Chen[2], Shang-Hong Lai[1]
[1]National Tsing Hua University, Taiwan, [2]NVIDIA

Abstract

*This paper describes our method for the ECCV 2024 Workshop W-CODA Track 1: Corner Case Scene Understanding. We propose **LiteViLA**: a **Li**ght**w**eight **Vi**sion-**L**anguage model pipeline for corner-case scene understanding in **A**utonomous driving, leveraging the TinyLLaVA [7] backbone for efficiently processing large-scale multimodal data. Our approach extracts visual features through a Vision Encoder and Q-Former [2], with the integration of visual and language modalities handled by the Language Model (LM) through a Mixture-of-Adapters (MoA) mechanism. The MoA dynamically selects task-specific adapters for General Perception, Region Perception, and Driving Suggestions, optimizing performance across these critical tasks. Finally, a Reviewer component refines the generated answers, ensuring their accuracy and relevance. The project page can be found at* https://imyingcheng.github.io/LiteViLA/.

1. Introduction

Autonomous driving demands advanced processing of multimodal data, particularly in complex and challenging scenarios. To address this, we propose a lightweight Vision-Language Model (VLM) pipeline, named **LiteViLA**, that integrates visual and language modalities for effective scene understanding and decision-making.

At the core of our approach is a lightweight backbone, TinyLLaVA [7], selected for its efficiency in handling large-scale data without compromising performance. This backbone includes SigLIP [6] as vision encoder that extracts visual features, refined by a Q-Former [2] into fine-grained visual tokens. These tokens are then processed by a small-scale Language Model (LM) Phi-2 [3] equipped with a Mixture-of-Adapters (MoA) mechanism, allowing task-specific adapters to optimize performance in General Perception, Region Perception, and Driving Suggestions tasks.

To enhance scene comprehension, we employ a multi-turn QA method [5] that sequentially queries different road objects, leading to more detailed and context-aware responses compared to traditional single-turn QA. Additionally, our Progressive Instruction Tuning Strategy fine-tunes the model for each task individually before implementing joint training, ensuring deep, task-specific understanding while maintaining the model's lightweight nature. Finally, a Reviewer component refines the model's generated answers, ensuring accuracy and relevance.

Our ablation studies confirm the effectiveness of these components, demonstrating significant performance improvements with our lightweight backbone. The final results affirm the success of our architecture in managing complex driving environments, advancing autonomous driving technology with both accuracy and efficiency.

2. Method

2.1. Overview

The proposed LiteViLA showcases a lightweight and efficient approach to integrating visual and language modalities for corner-case scene understanding in autonomous driving. The system begins with a Vision Encoder that transforms visual inputs into relevant features. These features are then processed by a Q-Former proposed by InstructBLIP [2], which further extracts essential query tokens, streamlining the information before it reaches the small-scale Language Model (LM). The LM is equipped with a Mixture-of-Adapters mechanism, which includes specialized adapters for General Perception, Region Perception, and Driving Suggestions. Each adapter functions as an expert for its respective task, ensuring that the most relevant knowledge is applied depending on the specific task. This design optimizes performance while maintaining a compact structure, allowing the model to generate detailed and context-aware answers with reduced resource requirements. Finally, the process concludes with a Reviewer component, refining the final output to ensure accuracy and relevance. The overall framework is shown in Fig. 1.

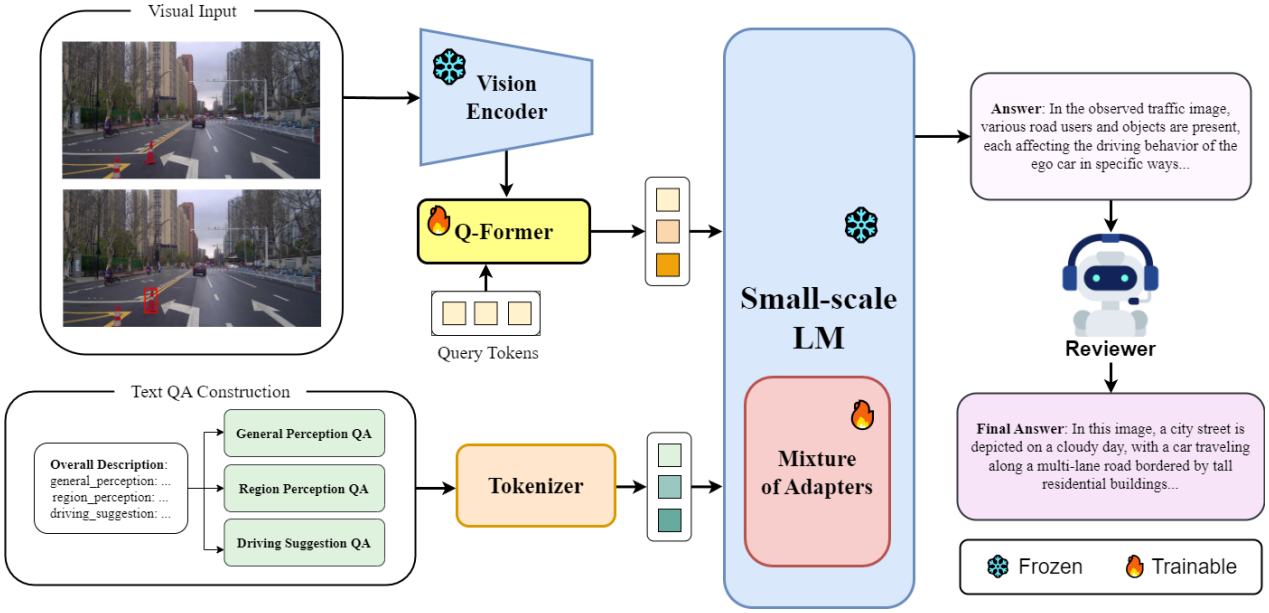


Figure 1. The overall framework of our LiteViLA. The vision encoder obtains the holistic visual features and Q-Former further extracts and transforms them into fine-grained visual tokens. The Mixture of Adapters in the frozen small-scale Language Model dynamically fuses visual knowledge learned from different adapters based on the task types.

2.2. The Progressive Training Strategy

Directly instruction-tuning VLMs across all three tasks — General Perception, Region Perception, and Driving Suggestions — in a single training step is sub-optimal because it forces the model to simultaneously balance conflicting task requirements, which dilutes its ability to specialize in the distinct features of each task, and risks overfitting to general features rather than developing the deep, task-specific understanding necessary for accurate and context-aware performance across all areas.

To address the challenges of single-step instruction-tuning, we first perform instruction-tuning individually for each task, allowing the model to develop specialized expertise without the interference of competing task requirements. In each tuning process, images are first processed through an Image Encoder to extract relevant features, which are then refined into query tokens by the Q-Former. These tokens are passed to a task-specific adapter, which is fine-tuned to excel at the specific task at hand — whether it understands the overall scene, analyzes specific regions, or generates driving suggestions. This targeted approach ensures that the model deeply understands the distinct features and demands of each task.

Once the individual task adapters are finely tuned, we implement the Mixture-of-Adapters mechanism for joint training. This final stage combines the specialized knowledge from the individual adapters, enabling the model to seamlessly integrate general scene understanding, precise regional analysis, and effective driving decision-making into a unified, context-aware output. The joint training phase ensures the model performs optimally across all tasks, leveraging its specialized components in a coordinated and efficient manner.

2.3. Mixture of Adapters with Router

In the final stage of our Progressive Instruction Tuning Strategy, we propose a unified model that integrates the specialized adapters trained in earlier stages. Inspired by the Mixture of Experts (MoE) framework, we treat each adapter [7] as an expert in its respective task. To efficiently manage the interaction between these experts, we follow LION [1] to employ a router module that dynamically directs the input features to the appropriate adapter based on the task type, as shown in Fig. 2.

At each Feed-Forward Network (FFN) layer, the input features are processed through a series of task-specific adapters, with the router determining which adapter's output will be utilized. The routing function allows for the selective activation of adapters, ensuring that the model applies the most relevant task-specific knowledge to the current input.

Let X represent the hidden representations generated by the self-attention layer. The output representation after the FFN layer, with the inclusion of an adapter (denoted by H), is given by:

$$O = F(X) + H(X)$$

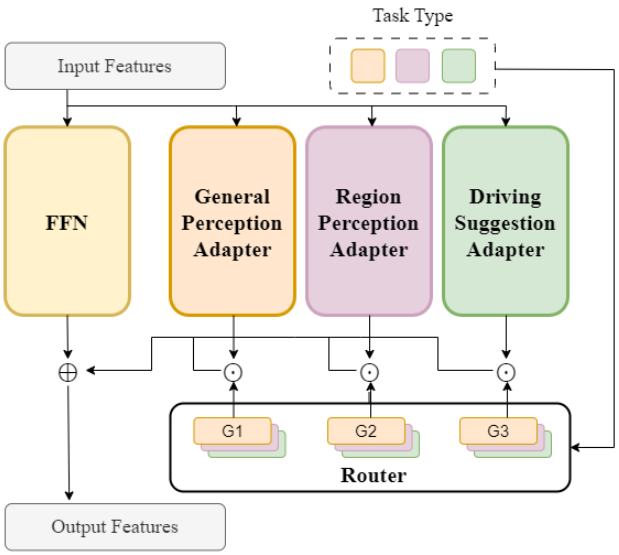


Figure 2. The architecture of the Mixture of Adapters with a Router mechanism, where input features are processed by a Feed-Forward Network (FFN) and routed to the appropriate task-specific adapter based on the task type.

where $H(X) = W_u(\sigma(W_d X))$, with σ representing a non-linear ReLU function. The router aggregates hidden features from the main branch and the adapters (denoted by $H_1, H_2, ..., H_k$), modulating them based on the task-specific vector G_k^t as:

$$O^t = F(X) + \sum_{k=1}^{K} G_k^t \odot H_k(X)$$

This routing mechanism ensures that the model leverages task-specific knowledge effectively, leading to optimized performance across all tasks.

2.4. Text QA Construction

In the Text QA Construction module, we design a multi-turn QA approach [5] for tasks such as General Perception and Driving Suggestion that require a comprehensive understanding of the entire scene. This method allows the model to sequentially query different road objects, leading to a deeper and more nuanced understanding of the scene. Compared to a single-turn QA, which might only provide a superficial overview, the multi-turn QA enables the model to explore specific details and relationships between objects, resulting in more accurate and context-aware responses.

2.5. Implementation Details

Dataset. We only use the CODA-LM dataset [4] for training, which contains 4,884 images for training and 4,384 images for validation. Each image is paired with detailed textual descriptions covering General Perceptions, Region-specific details, and Driving Suggestions, with a focus on challenging corner cases.

Backbone Network. We use TinyLLaVA-3.1B [7] as our backbone Vision-Language Model (VLM) due to its lightweight design and efficiency. While TinyLLaVA [7] originally uses a simple MLP as the connector, we have replaced it with Q-Former to enhance connectivity between the SigLIP [6] image encoder and the Phi-2 [3] language model.

Experimental Details. Our training process is structured into four stages, with each stage fine-tuning the model for one epoch. All experiments are conducted on a single Nvidia RTX 4090 GPU, ensuring consistent performance and efficient utilization of computational resources.

2.6. Ablation Study

We perform some ablation experiments to validate the effectiveness of the modules used in the final method. Note that in the tables, G score represents GPT score of General Perception task, R score represents GPT score of Region Perception task, and D score represents GPT score of Driving Suggestion task.

Effects of Using Q-Former as Connector. As shown in Tab. 1. Using Q-Former as a connector instead of MLP enhances the integration of visual and textual features in Vision-Language Models (VLM) by leveraging attention mechanisms, which better capture complex multimodal interactions. This leads to improved performance across tasks.

Comparison of Default QA and Multi-turn QA Approaches. As shown in Tab. 2. The Multi-turn QA method [5], designed to handle tasks requiring global scene understanding such as General Perception and Driving Suggestions, outperforms the Default QA method across all metrics. This demonstrates the effectiveness of the Multi-turn QA in providing more detailed and context-aware responses by sequentially querying different road objects.

Effects of Adding Mixture of Adapters (MoA). As shown in Tab. 3. The Mixture-of-Adapters (MoA) mechanism and the progressive instruction tuning strategy further improve performance across all three tasks compared to fine-tuning VLM with LoRA. By dynamically selecting task-specific adapters, the model efficiently shares knowledge across tasks while optimizing for task-specific complexities, especially in multi-task fusion scenarios like Driving Suggestions. This results in better adaptability and overall performance than individual fine-tuning.

2.7. Final Results

The final results, with G score of 55.16, an R score of 82.88, and a D score of 65.50, demonstrate the effectiveness of our LiteViLA, with an average score 4 points higher than

Method	G score	R score	D score
TinyLLaVA	47.32	72.58	51.60
TinyLLaVA + Q-Former	47.62	82.27	53.22

Table 1. Ablation Study on the impact of using Q-Former as a connector

Method	G score	R score	D score
Default QA	47.62	82.27	53.22
Multi-turn QA	51.58	82.76	56.42

Table 2. Ablation study on comparing default single-turn QA and multi-turn QA approaches

Method	G score	R score	D score
TinyLLaVA + LoRA	51.58	82.76	53.22
TinyLLaVA + MoA	55.16	82.88	65.50

Table 3. Ablation study on comparing the performance of fine-tuning each task individually with LoRA and using the MoA method

the baseline CODA-VLM. Examples of the results generated by LiteViLA are shown in Fig. 3 and Fig. 4. By integrating Q-Former [2] as a connector, utilizing a multi-turn QA approach [5], and employing a Mixture-of-Adapters module for specialized task handling, we show LiteViLA significantly enhances its ability to manage complex tasks like General Perception and Driving Suggestions. These components work together to capture and process the intricate relationships within the driving environment, leading to more accurate, context-aware outputs, demonstrating the overall success of the architecture.

However, despite these improvements, the model's ranking in the competition was not as high as anticipated, indicating there is still room for enhancement. Future research could focus on refining the multi-turn QA approach for better contextual understanding, enhancing visual-language integration with advanced fusion techniques, and optimizing the Progressive Instruction Tuning Strategy in order to further elevate the model's capabilities in handling complex driving scenarios.

3. Conclusion

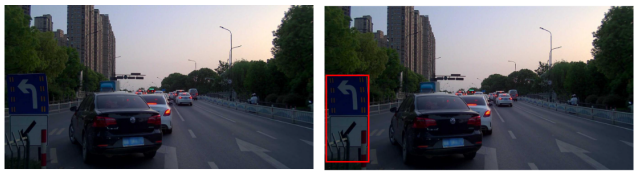
The proposed LiteViLA demonstrates significant improvement in handling complex urban scenes. Through integrating Q-Former as a connector, utilizing a multi-turn QA approach, and employing a Mixture of Adapters for task-specific processing, LiteViLA effectively captures and processes intricate relationships within driving environments. The experimental results show its ability to deliver more accurate, context-aware outputs, confirming the system's effectiveness and efficiency.

General Perception: In the image, a blue tricycle is parked on the right side of the road and is partially obstructing the lane, requiring the ego car to cautiously navigate around it. A single traffic cone is placed near the truck, indicating a potential hazard or road work ahead. Given the dark road environment, the ego car should rely on its headlights and remain alert, proceeding with caution and possibly reducing speed to safely maneuver around these obstacles.

Driving Suggestion: Given the dark and low-visibility conditions, the ego car should reduce speed to ensure better reaction time, keep a safe distance from the parked vehicle on the left to avoid potential collisions, stay alert for any additional cones or signs indicating obstructions, move slightly to the right to navigate around the visible traffic cone while ensuring not to veer off the road, use its headlights and sensors to detect any other potential hazards that may be obscured by the darkness, and possibly consider using hazard lights if the situation involves a sudden stop or significant hazard to alert other road users of the vehicle's presence and intentions.

Region Perception: This object is a tricycle, which appears to be a three-wheeled vehicle commonly used for transportation in urban areas. The tricycle is equipped with a rear-mounted engine and a cargo area, suggesting it may be used for carrying goods or passengers. In the context of driving behavior, the presence of a tricycle on the road requires the ego car to maintain a safe following distance and be prepared for potential stops or slow movement, especially in areas where tricycles are more common.

Figure 3. In this corner-case scenario, our LiteViLA accurately identifies key objects in the image, including a traffic cone, and a tricycle. Despite the challenging conditions, the model successfully provides precise driving suggestions, such as reducing speed and using headlights, ensuring safe navigation around these potential hazards.



General Perception: In the image, the black Volkswagen ahead requires the ego car to maintain a safe distance and adjust speed. Vehicles in adjacent lanes, like a white sedan and a truck, require monitoring for lane changes. Pedestrians on the right sidewalk should be watched for sudden movements. The blue traffic sign on the left signals a left turn, guiding the ego car to prepare for it. Metal barriers on the right define the lane, and the green traffic light ahead requires the ego car to stay alert for changes.

Driving Suggestion: Maintain a safe following distance from the black Volkswagen ahead, monitor adjacent vehicles for sudden lane changes, stay aware of pedestrians on the sidewalk, follow the left-turn sign to ensure you're in the correct lane, observe the green traffic light for potential changes, and adjust your speed as needed to match the traffic flow.

Region Perception: The object is a blue directional traffic sign with a white arrow indicating a left turn, instructing the ego car to turn left at the upcoming intersection. It requires the vehicle to adjust speed, position, and lane to safely follow traffic regulations and avoid violations.

Figure 4. In this urban traffic scenario, our model accurately identifies key elements such as a black Volkswagen, a left-turn traffic sign, and adjacent vehicles. Despite the complex setting, it provides precise driving suggestions, like maintaining safe distances and observing traffic signals, ensuring smooth navigation through potential hazards.

References

[1] Gongwei Chen, Leyang Shen, Rui Shao, Xiang Deng, and Liqiang Nie. Lion: Empowering multimodal large language model with dual-level visual knowledge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26540–26550, 2024. 2

[2] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *arXiv preprint arXiv:2305.06500*, 2024. 1, 4

[3] Mojan Javaheripi, Sébastien Bubeck, Marah Abdin, Jyoti Aneja, Sebastien Bubeck, Caio César Teodoro Mendes, Weizhu Chen, Allie Del Giorno, Ronen Eldan, Sivakanth Gopi, et al. Phi-2: The surprising power of small language models. *Microsoft Research Blog*, 2023. 1, 3

[4] Yanze Li, Wenhua Zhang, Kai Chen, Yanxin Liu, Pengxiang Li, Ruiyuan Gao, Lanqing Hong, Meng Tian, Xinhai Zhao, Zhenguo Li, et al. Automated evaluation of large vision-language models on self-driving corner cases. *arXiv preprint arXiv:2404.10595*, 2024. 3

[5] Na Liu, Liangyu Chen, Xiaoyu Tian, Wei Zou, Kaijiang Chen, and Ming Cui. From llm to conversational agent: A memory enhanced architecture with fine-tuning of large language models. *arXiv preprint arXiv:2401.02777*, 2024. 1, 3, 4

[6] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11975–11986, 2023. 1, 3

[7] Baichuan Zhou, Ying Hu, Xi Weng, Junlong Jia, Jie Luo, Xien Liu, Ji Wu, and Lei Huang. Tinyllava: A framework of small-scale large multimodal models. *arXiv preprint arXiv:2402.14289*, 2024. 1, 2, 3