Applications of Robotics and Autonomous Systems Report: Fine-Tuning a Vision Language Model for Visual Question Answering in Autonomous Driving

Can Aydin, Fatih Mercan, Justin Eduard Hulha, Jasmin Michelle Hulha
Jan Henri Christian Evard, Jiaao Li

March 2025

1 Introduction

With progress in autonomous driving and deep learning technologies, a large portion of the solutions that have emerged focus on utilizing LiDAR/Radar data, computer vision, and geometry information from maps to enhance the performance of autonomous driving systems (Li Chen et al. 2024). However, the community still faces a series of challenges, such as poor generalization ability and a lack of transparency and explainability (PAVE 2020; Tang et al. 2024). Meanwhile, ever-growing Large Language Models (LLMs) and Vision Language Models (VLMs) are showing excellent generalization performance and possess the natural advantage of interacting with humans effortlessly (Tang et al. 2024). Therefore, more and more researchers have explored the potential of using LLMs and VLMs to improve Autonomous Driving (AD) systems (Long Chen et al. 2023; Jin et al. 2023; Tang et al. 2024). LLMs have demonstrated great reasoning, reference, and generalization abilities (Long Chen et al. 2023), while VLMs have also shown remarkable performance in Visual Question Answering (VQA) in driving scenarios (Marcu et al. 2024).

In this project, we investigate whether fine-tuning a VLM on the LingoQA dataset improves performance on the LingoQA benchmark for VQA in autonomous driving. Our results show that this approach not only outperforms the base model but also exhibits a stronger grasp of understanding relevant contextual information and critical fine-grained details. Furthermore, we surpass the fine-tuned reference model presented in the LingoQA paper using the Lingo-Judge evaluation method on the LingoQA evaluation dataset. Finally, to interactively demonstrate our fine-tuned model, we developed a Gradio-based user interface that allows flexible, interactive VQA in driving scenarios.

2 Literature Review

2.1 End-to-End Autonomous Driving

The past decade has witnessed significant progress in end-to-end autonomous driving systems that use computer vision and LiDAR- or radar-based methods (Li Chen et al. 2024). UniAD incorporates full-stack driving tasks into its framework, and communication between different tasks is based on query-unified interfaces (Y. Hu et al. 2023). InterFuser proposes a framework

that adapts the trending transformer architecture to fully fuse and process signals from multi-modal and multi-view sensors for better perception of the driving scene (Shao et al. 2022). Recent work such as MILE (A. Hu et al. 2022) came up with a method that leverages 3D geometry as an inductive bias and videos of expert demonstrations, such that it enables the model to learn a highly compact latent space directly. Beyond these solutions, reinforcement learning strategies have also been trending. Zhang et al. (2021) investigated a reinforcement learning agent that maps a bird's-eye view to access environmental information and distills a model as the final end-to-end agent, which performs well (78% success rate) on a generalized testing dataset.

However, challenges are still present, especially when it comes to interpretability in the decision-making process for autonomous driving (Barredo Arrieta et al. 2020). Partners for Automated Vehicle Education (PAVE) noted that 60% of the participants expressed the expectation of understanding more about the underlying reference process of the autonomous driving system (PAVE 2020). In addition to that, traditional deep learning-based methods are inadequate at dealing with corner cases, leading to a potential collapse of safety controls in a variety of real-world scenarios.

2.2 Multi-modal Large Language Models in Autonomous Driving

Handling long-tail corner cases and explainability have been major challenges faced by autonomous driving systems. Because of their greater ability in addressing these issues, the idea of integrating LLMs and VLMs as support systems in autonomous driving has gained ever-growing research interest. The community has been calling for explainable solutions to help build trust between the autonomous driving system and the driver to enable an effective and safe human-AI collaboration (Long Chen et al. 2023). Tang et al. (2024) found that LLMs have great potential in connected self-driving cars. They presented their model with a test dataset containing questions similar to the UK Driving Theory Test, showing that the GPT-family from OpenAI, the Ali-Qwen model from AliBaBa and the Tsinghua's open model MiniCPM could all handle driving theory knowledge and test questions with good generalization (Tang et al. 2024).

Since onboard cameras are heavily used in modern automotive driver support systems, it is reasonable to consider solutions that use images as another primary input source. VLMs have shown remarkable performance in tasks such as VQA. Therefore, it is rational to bridge the gap between data-driven decision-making and user trust by integrating VLMs into the field of autonomous driving systems (Marcu et al. 2024). The excellent driving scene understanding and reasoning ability of LLMs/VLMs has been shown in works like ADAPT (Jin et al. 2023) and LLM-Driver (Long Chen et al. 2023), where they proposed multi-task learning frameworks jointly predicting language and control outputs and have demonstrated that a pre-trained LLM can be specifically improved for understanding driving situations by using a unique object-level multimodal LLM architecture. Pioneering models such as GPT4-V (Wen et al. 2023) have explored the potential of VLMs on autonomous driving tasks. DriveGPT leverages a multi-modal vision-language-action model that tokenizes videos, as well as text and control actions, and developed an interpretable end-to-end autonomous driving solution (Xu et al. 2024). Hwang et al. (2024) have investigated a promising solution that suggests mapping all non-sensor inputs to various driving-specific descriptive texts, and fine-tuning a pre-trained LLM for autonomous driving (Hwang et al. 2024). Research findings from LingoQA (Marcu et al. 2024) also suggest that approaches that involve partially fine-tuning the attention layers of VLMs with their domain-specific datasets seem effective and promising. Their findings show that this method allows the model to jointly process complex driving scenarios in a unified language space, and generate outputs by using task-specific prompts (Hwang et al. 2024).

2.3 Datasets and Evaluation Metrics for Language-grounded Autonomous Driving Tasks

For driving scenarios, multimodal datasets are particularly important as autonomous driving in modern vehicles involves multiple types of sensors. A popular configuration is using a camera for capturing road impacts with radar providing complementary fine-grained and comprehensive 3D road information as well as information on the speed of a wide range of moving objects (Caesar et al. 2020). The recent breakthroughs in generative AI, especially in LLMs/VLMs have shown that large and comprehensive training datasets that provide sufficient and domain-specific information can improve the performance of pre-trained base models significantly (Alayrac et al. 2022).

To address the shortage of current autonomous vehicle datasets, NuScenes data provides a large and comprehensive multimodal dataset with 360-degree coverage across all visions and range sensors collected from various situations alongside map information (Caesar et al. 2020). Their work has encouraged strong interest from the autonomous driving community and also motivated a score of further research. Talk2Car (Deruyttere et al. 2019) subsequently complemented the NuScenes (Caesar et al. 2020) dataset with free-form captions to guide the future trajectory of the autonomous car using referred objects in the scene. As an open-source simulator for autonomous driving, CARLA (Dosovitskiy et al. 2017) provides a diverse and complex environment and the simulator has also been used to generate driving scenario data for related research work, such as DriveLM (Sima et al. 2025) which selects different towns in the CARLA simulator to serve as training and evaluation sandboxes. Focusing on vision-only end-to-end autonomous driving, LingoQA (Marcu et al. 2024) introduced a comprehensive dataset containing 419k QA pairs tailored to autonomous driving VQA. The novelty can be seen in its free-form question-and-answer approach that expands the scope of driving behavior quality assurance to include reasoning and reasons for action. Based on the work done in NuScences and CARLA, DriveLM (Sima et al. 2025) presented a novel framework leveraging VLMs through a Graph VQA structure, integrating a series of driving behavior question-answer pairs within a directed acyclic graph (DAG) to effectively mimic human reasoning for driving tasks. Furthermore, a multi-component metric system, DriveLM-Metric (Sima et al. 2025), was introduced which comprises standard trajectory prediction metrics (ADE, FDE, and collision rate) for motion, classification accuracy for behavior, and a combination of SPICE and GPT-Score to assess structured and semantic alignment in language-based responses. Progress in VLMs for autonomous driving has been moving towards exploring automated, reproducible evaluation metrics that are highly correlated with human ratings. Human feedback faces the challenges of incurring substantial costs while providing poor reproducibility and being highly subjective (Marcu et al. 2024). Addressing these challenges, another core contribution of LingoQA is the novel evaluation metric, Lingo-Judge, which is a text classifier based on a DeBERTa-V3 backbone and fine-tuned with LoRA. Lingo-Judge takes in the model's output to a VQA task and a ground truth answer labeled by humans and outputs a probability score that the model answer matches the human answer. It achieves a 0.95 Spearman and 0.993 Pearson correlation with human evaluation and thereby outperforms traditional metrics like METEOR, BLEU and GPT-4 while taking significantly less processing time (10.5 seconds using Lingo-Judge vs. 812.4 seconds using GPT-4 to evaluate the entire LingoQA evaluation dataset).

In this project, we address the concerns raised above and explore the connection between AD and VLMs. In so doing, we aim to improve the reasoning, interpretability, and decision-making of AD systems. Our project specifically utilizes the LingoQA dataset (Marcu et al. 2024) to fine-tune a pre-trained Ali-Qwen VLM (Qwen2-VL-7B-Instruct) for the specific task of VQA in driving scenarios. To evaluate our results, we decided to use the Lingo-Judge classifier due to its high correlation with human judgment and rapid processing capabilities.

3 Methodology

3.1 Dataset: LingoQA

The LingoQA dataset was used for fine-tuning the Qwen2-VL-7B-Instruct model. This dataset consists of an evaluation dataset and a training dataset, which is divided into two complementary datasets: the Action and Scenery dataset (Marcu et al. 2024). A sample consists of a 4-second clip along with corresponding questions and answers. The entire dataset contains approximately 419,000 question and answer pairs, with the answers being free-form and averaging 17 words in length (Marcu et al. 2024). The questions in the two training datasets have different foci: The questions in the Action dataset pertain to driving behavior. The Scenery dataset extends the Action dataset and focuses on perception capabilities. The evaluation dataset covers the same categories as the training datasets but contains more complex driving situations (Marcu et al. 2024). The Action dataset contains driving recordings where the behavior of the car changes,

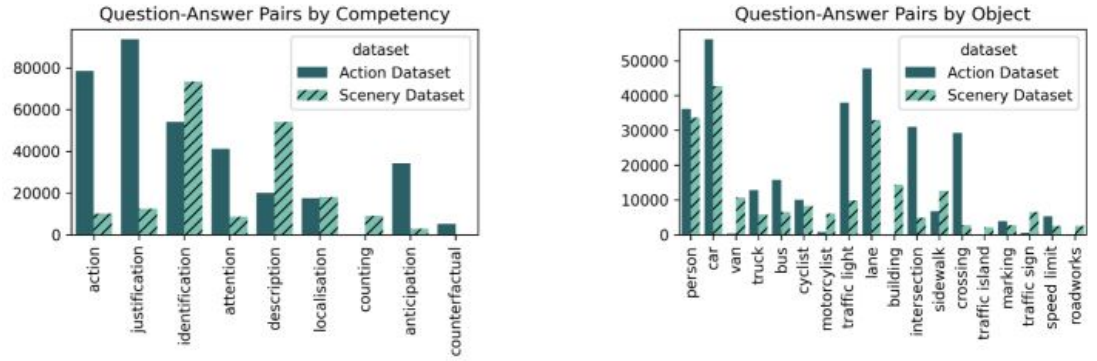

Figure 1: In the left chart, the number of question-answer pairs related to the different skills on the vertical axis is shown. In the right chart, the number of question-answer pairs related to the different objects on the vertical axis is shown (Marcu et al. 2024)

.

such as the acceleration of the car or lane changes. In addition to the descriptions of these recordings by the driver, metadata from perception systems, such as weather conditions, the presence of traffic lights, vehicles, or pedestrians, and other data were considered. Based on the driver descriptions and metadata, a prompt template was created that describes both the current situation and its reasons, as well as formulating questions and hints for possible answers. These prompts were then expanded and reformulated with GPT-3.5, and answers were generated. To obtain a balanced dataset, the driving events were categorized based on the actions, and up to 500 events per category were selected (Marcu et al. 2024). The Scenery dataset was created from three 30-minute driving recordings. Using the video annotation software ELAN, the driving scenes were annotated in detail for each frame on a second-by-second basis. From this, questions regarding perception and their answers to the driving scenario were generated using GPT-4 (Marcu et al. 2024). The training datasets consist of 9 different competencies, and the questions cover a large number of objects. In Figure 1, the number of questions and answers for each of the 9 competencies and for the different objects in the respective datasets are shown (Marcu et al. 2024). The evaluation dataset was fully annotated by humans – both the questions and the answers – and allows for the assessment of the competencies shown in Figure 1.

3.2 Model: Qwen2-VL-7B-Instruct

As mentioned earlier, we chose to use Qwen2-VL-7B-Instruct for our project. Qwen2-VL builds upon Qwen-VL, adding native resolution support and understanding of videos with more than 20 minutes of content. Similarly to Qwen-VL, Qwen2-VL consists of a Vision Encoder that takes in images and/or videos as input. The Vision Encoder returns the tokens of the images and/or videos, which are then given to the LLM, Qwen2. With that, Qwen2 is capable of accurately identifying and understanding the context within images.
The Vision Encoder itself consists of a ViT (Vision Tranformer) and an MLP compression layer. The ViT has approximately 675 million parameters and is adept at handling both image and video inputs. The output of the ViT is then forwarded to the MLP compression layer, which compresses *2x2* tokens into a single token, ensuring lower amounts of tokens while maintaining accuracy. To support native resolution support, the authors introduced *M-RoPE* positional encoding instead of absolute positional encoding, which effectively models the information of multi-modal inputs, such as temporal, height, and width dimensions (Wang et al. 2024).

3.3 Fine-Tuning

For the fine-tuning process, we used the open-source framework LLaMA-Factory (Hiyouga 2023), which is designed to streamline the training process of transformer-based language models.
To manage the memory constraints of a single NVIDIA A100 GPU with 40GB of VRAM, we applied parameter-efficient fine-tuning techniques, specifically LoRA (Low-Rank Adaptation) (E. J. Hu et al. 2021) and QLoRA (Quantized LoRA) (Dettmers et al. 2023). With QLoRA, we leveraged 4-bit quantization to further reduce GPU memory consumption without significantly impacting model accuracy. We explored various hyperparameter settings, including different dataset sizes, image resolutions, and sequence lengths, to identify the best-performing configuration (see Table 2).

3.4 Gradio UI

To demonstrate our fine-tuned model, we developed a Gradio-based (Abid et al. 2019) user interface that allows flexible, interactive VQA on driving scenarios. Using this demo, users can input a question and upload one or multiple images alongside a text prompt. The images are fed to the model in the order given.
The demo also supports video clips by pre-processing them as a series of images, extracted at one frame per second of footage. While this simple frame sampling loses some motion continuity, it provides a manageable summary of the video for the model to reason about and is consistent with LingoQA's approach of sampling.
To allow for quick comparisons between the base Qwen2-VL model and our fine-tuned models, the UI includes a feature to switch between different model weights on the fly. Loading a 7B VLM with vision components is memory-intensive, utilizing nearly the entire 40GB GPU memory during inference with 5 images and a text prompt. To avoid memory issues, we implemented a mechanism to unload the currently loaded model from GPU memory before loading another model, preventing out-of-memory (OOM) errors when switching. This allows evaluators to seamlessly test the same question on different models and observe differences.
For convenience and qualitative benchmarking, the interface supports loading example questions and images drawn from the LingoQA evaluation set using a dropdown. By selecting such examples, users can quickly see how the model performs on known evaluation questions from the LingoQA benchmark.
We also integrated speech-to-text (STT) and text-to-speech (TTS) capabilities into the Gradio

UI. Users can press a microphone button to ask a question by voice and the demo will convert it to text using an automated speech recognition system based on the Whisper-base STT model (Radford et al. 2022). After the model generates an answer, the answer can be spoken aloud through TTS (hexgrad 2025) so the user hears the response. This is supposed to mimic an in-car conversational assistant scenario.

4 Results and Discussion

4.1 Qualitative Results

After fine-tuning on the LingoQA Scenery data, the model shows notable improvements in its answers to driving-related questions compared to the base Qwen2-VL model. We present a qualitative comparison highlighting improvements in context understanding as well as adherence to details.

The fine-tuned model exhibits a better grasp of the driver's perspective and the relevant context, whereas the base model was sometimes lacking in this regard. For example, when shown an entry from the LingoQA evaluation set, which depicts an intersection with a green light and a red pedestrian light, the base Qwen2-VL interpreted the light as red, while the fine-tuned model correctly identified it as green. When asked about which elements are capturing its attention, the base model included generic out-of-context descriptions about the artistic style of the image and described the scene from the perspective of a pedestrian or bystander. In comparison, the fine-tuned model directed its attention to traffic-related elements from the perspective of the ego-vehicle.

In other examples, we observed that the fine-tuned model is far more attentive to critical fine-grained details in the images. The base model would often overlook or misidentify subtle yet important details such as the color of a traffic light. After fine-tuning, the model incorporated such details into its answers much more reliably.

4.2 Quantitative Results

Table 1 compares runtime performance and token usage for three different inference methods (API, Chat, and vLLM) under four different querying scenarios. The inference methods vary in the use of the inference backend and the ability to parallelize sample processing. Inherently, the API supports threading, making it feasible to process multiple queries simultaneously given enough hardware resources. Both the API and chat methods use the Hugging Face backend, whereas the vLLM version uses the vLLM inference backend. The vLLM library is specialized in the inference of transformer-based models and supports batch processing of multiple queries. The chat version does not support batch processing or threading but includes a chat history.

In Table 1, the four querying scenarios can be divided into two sub-scenarios, single data point and batch processing. For the batch processing scenario, the model receives a single request containing 500 samples, whereas in the single data point scenario, each of the 500 samples is processed consecutively. In the single data point scenario, we then compute the average response time, prompt tokens, and completion tokens across all 500 samples. Both the batch and single data point scenarios are further divided into single image and five images per query, as the training dataset consists of five images per query. The single data point scenario is especially relevant for real-time situation analysis in a driving car, whereas the batch scenario is more relevant for fast dataset processing.

Scenario	API (with threading)			Chat			vLLM		
	Time	**PTs**	**CTs**	**Time**	**PTs**	**CTs**	**Time**	**PTs**	**CTs**
multiple_queries_multiple_images	3.61	3891.74	15.12	3.94	3891.74	15.07	1.15	3891.74	44.34
single_query_single_image	0.80	803.74	15.45	1.01	803.74	15.10	35.54	803.74	6.40
single_query_multiple_images	3.74	3891.74	15.09	3.96	3891.74	15.32	37.09	3891.74	7.40
multiple_queries_single_image	0.74	803.74	15.07	1.00	803.74	15.13	0.33	803.74	45.35

Table 1: Comparison of scenarios across API (4 threads), Chat, and vLLM for three key metrics on the first 500 samples of the LigoQA Evaluation dataset. The metrics are time in seconds, prompt tokens (PTs), and completion tokens (CTs). Single queries are averaged over 500 single-sample batches, while multiple queries uses a single batch of 500 samples. Multiple images refers to five images per query, single image refers to one image.

The key finding from Table 1 is that using the vLLM backend greatly accelerates batch processing, leading to substantially reduced total inference times (1.15s and 0.33s) compared to the chat interface with 3.94s and 1.00s. However, for single sample queries, the vLLM backend slows down considerably because it must reload the parameters into the GPU each time a new inference request is made. Additionally, the API and Chat runtimes are nearly identical, reflecting VRAM constraints (40 GB) that limit parallelization and hence do not allow multiple threads to run simultaneously in our setup, as the fine-tuned model already exhausts about 33GB of the VRAM.

Table 3 highlights the performance gains from fine-tuning the Qwen2-VL-7B-Instruct Model. It shows the fixed image resolution and adapted image resolution Lingo-Judge scores for each fine-tuning run. Fixed image resolution refers to the maximum image resolution during inference, which we set to 589,824 pixels, representing a 768x768 image. However, during fine-tuning, the image resolution is mapped according to the image resolution hyperparameter as listed in Table 2. For the adapted image resolution Lingo-Judge score, we use the same image resolution for inference and fine-tuning as listed in Table 2.

Model	Dataset	Size	Image Res.	Cutoff Length	Batch Size	Quantization
Qwen2-VL-7B-Instruct #1	Action	1,000	262,144	2,048	2	-
Qwen2-VL-7B-Instruct #2	Scenery	1,000	262,144	2,048	2	-
Qwen2-VL-7B-Instruct #3	Action	10,000	262,144	2,048	2	-
Qwen2-VL-7B-Instruct #4	Scenery	10,000	262,144	2,048	2	-
Qwen2-VL-7B-Instruct #5	Action	10,000	700,000	8,192	1	4
Qwen2-VL-7B-Instruct #6	Scenery	10,000	700,000	8,192	1	4
Qwen2-VL-7B-Instruct #7	Action	25,000	262,144	2,048	2	-
Qwen2-VL-7B-Instruct #8	Scenery	25,000	262,144	2,048	2	-
Qwen2-VL-7B-Instruct #9	Action, Scenery	20,000	700,000	8,192	1	4

Table 2: Overview of multiple fine-tuning runs on the Qwen2-VL-7B-Instruct Model. The table lists the key hyperparameters for each run alongside the resulting score.

Across both action and scenery datasets, most of the fine-tuned versions outperform the base model and the OpenAI reference (Model: GPT-4o), fulfilling our main research objective. We also see that the dataset size plays a mixed role: while increasing the number of training samples boosts the Lingo-Judge score on the scenery dataset, it has no significant effect on the action dataset. When comparing the fine-tuning runs 2, 4, and 8 where only the dataset size of the scenery dataset is changed, we see an increase in the Lingo-Judge score from 1,000 samples and 52.90%, 10,000 samples and 58,60% up to 25,000 samples and 63,50% with a total increase of about 10.60%. Additionally, increasing the image resolution during fine-tuning consistently raises Lingo-Judge scores. Comparing the fine-tuning run 3 to 5 and 4 to 6, we can see an increase in the Lingo-Judge score of 4.89% and 3.20% respectively. Yet, at inference time,

using that same high resolution does not always yield higher scores compared to a standardized resolution of 589,824 pixels, suggesting a trade-off between the models' capability to process a large context size and a higher image resolution. However, when the image resolution during inference is reduced to the image resolution used in fine-tuning, as was done in fine-tuning runs 1, 2 and when we ran the base model, we see a drop in the Lingo-Judge score by 2.10%, 2.30%, and 8.30% respectively between the fixed and adapted resolution scores. This indicates that a too-low resolution has a strong negative effect on the model's ability to derive information from the images. Notably, when comparing the fine-tuning run 7 and 8, the scenery dataset achieves the overall highest score with 63.50%, outperforming the action dataset with 55.30% in fixed-resolution inference, despite the fact that action annotations were fully human-labeled while scenery labels were partly human, partly LLM-generated.

Model	**Lingo Score Adapted Res.**	**Lingo Score Fixed Res.**
OpenaiAI GPT-4o	53.90	53.90
Qwen2-VL-7B-Instruct Base	45.90	54.20
Qwen2-VL-7B-Instruct #1	53.20	55.30
Qwen2-VL-7B-Instruct #2	50.60	52.90
Qwen2-VL-7B-Instruct #3	55.50	55.20
Qwen2-VL-7B-Instruct #4	58.50	58.60
Qwen2-VL-7B-Instruct #5	58.80	60.09
Qwen2-VL-7B-Instruct #6	**63.20**	61.80
Qwen2-VL-7B-Instruct #7	55.80	55.30
Qwen2-VL-7B-Instruct #8	61.90	**63.50**
Qwen2-VL-7B-Instruct #9	61.60	62.40

Table 3: Overview of multiple fine-tuning runs on the Qwen2-VL-7B-Instruct Model. For each run, 1,000 samples from the LingoQA Evaluation Dataset were used to generate predictions, which were then scored by the Lingo-Judge benchmark to yield a single overall score. The predictions for the fixed resolution Lingo-Judge score were generated with the Hugging Face backend and an image resolution of 589,824 pixels. The predictions for the adapted resolution Lingo-Judge score were generated with the Hugging Face backend and the same image resolution they were fine-tuned on, as listed in Table 2. The image resolution using the OpenAI API is unknown.

5 Conclusion

Overall, the fine-tuning on LingoQA data significantly improved the model's performance on autonomous driving VQA. The base Qwen2-VL model often produced answers that were technically plausible but not context-appropriate for driving, or misinterpreted small yet critical details. After domain adaptation, the model's responses show improved contextual correctness as well as more precision in key details such as traffic signals and signs. Our best fine-tuned model thus achieves a state-of-the-art score of 63,50% on the LingoQA evaluation dataset using Lingo-Judge, highlighting the potential of fine-tuning vision-language models on high-quality domain-specific training data.

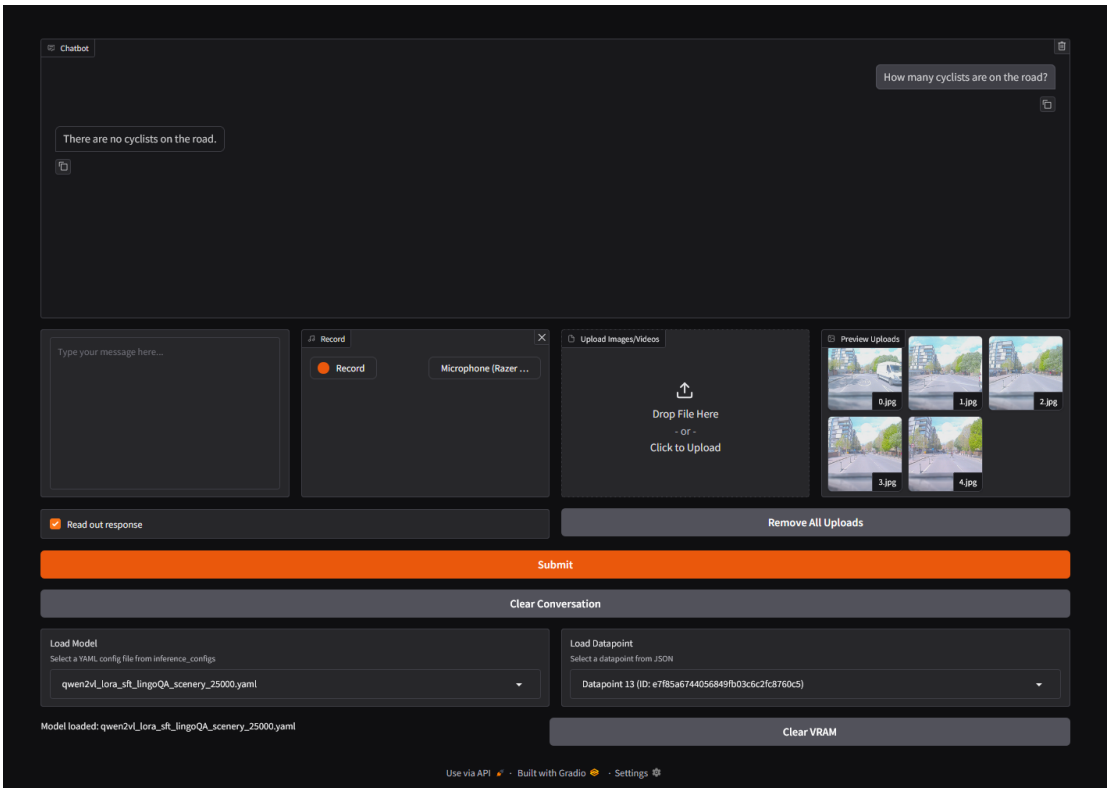

Figure 2: Our Gradio-based user interface demonstrating interactive VQA on driving scenarios.

A Appendix

References

Abid, Abubakar et al. (2019). "Gradio: Hassle-Free Sharing and Testing of ML Models in the Wild". In: *arXiv preprint arXiv:1906.02569*. URL: https://arxiv.org/abs/1906.02569.

Alayrac, Jean-Baptiste et al. (2022). *Flamingo: a Visual Language Model for Few-Shot Learning*. arXiv: 2204.14198 [cs.CV]. URL: https://arxiv.org/abs/2204.14198.

Barredo Arrieta, Alejandro et al. (2020). "Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI". In: *Information Fusion* 58, pp. 82–115. ISSN: 1566-2535. DOI: https://doi.org/10.1016/j.inffus.2019.12.012. URL: https://www.sciencedirect.com/science/article/pii/S1566253519308103.

Caesar, Holger et al. (2020). *nuScenes: A multimodal dataset for autonomous driving*. arXiv: 1903.11027 [cs.LG]. URL: https://arxiv.org/abs/1903.11027.

Chen, Li et al. (2024). *End-to-end Autonomous Driving: Challenges and Frontiers*. arXiv: 2306.16927 [cs.RO]. URL: https://arxiv.org/abs/2306.16927.

Chen, Long et al. (2023). *Driving with LLMs: Fusing Object-Level Vector Modality for Explainable Autonomous Driving*. arXiv: 2310.01957 [cs.RO]. URL: https://arxiv.org/abs/2310.01957.

Deruyttere, Thierry et al. (2019). "Talk2Car: Taking Control of Your Self-Driving Car". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Process-*

ing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Association for Computational Linguistics. DOI: 10.18653/v1/d19-1215. URL: http://dx.doi.org/10.18653/v1/D19-1215.

Dettmers, Tim et al. (2023). "QLoRA: Efficient Finetuning of Quantized LLMs". In: *Proceedings of the 40th International Conference on Machine Learning*. URL: https://arxiv.org/abs/2305.14314.

Dosovitskiy, Alexey et al. (2017). *CARLA: An Open Urban Driving Simulator*. arXiv: 1711.03938 [cs.LG]. URL: https://arxiv.org/abs/1711.03938.

hexgrad (2025). *Kokoro-82M: An Efficient Open-Weight Text-to-Speech Model*. Accessed: 2025-03-09. URL: https://huggingface.co/hexgrad/Kokoro-82M.

Hiyouga (2023). *LLaMA-Factory: Efficient and Friendly Framework for LLM Fine-Tuning*. Accessed: 2024-03-11. URL: https://github.com/hiyouga/LLaMA-Factory.

Hu, Anthony et al. (2022). *Model-Based Imitation Learning for Urban Driving*. arXiv: 2210.07729 [cs.CV]. URL: https://arxiv.org/abs/2210.07729.

Hu, Edward J. et al. (2021). *LoRA: Low-Rank Adaptation of Large Language Models*. arXiv: 2106.09685 [cs.CL]. URL: https://arxiv.org/abs/2106.09685.

Hu, Yihan et al. (2023). *Planning-oriented Autonomous Driving*. arXiv: 2212.10156 [cs.CV]. URL: https://arxiv.org/abs/2212.10156.

Hwang, Jyh-Jing et al. (2024). *EMMA: End-to-End Multimodal Model for Autonomous Driving*. arXiv: 2410.23262 [cs.CV]. URL: https://arxiv.org/abs/2410.23262.

Jin, Bu et al. (2023). *ADAPT: Action-aware Driving Caption Transformer*. arXiv: 2302.00673 [cs.CV]. URL: https://arxiv.org/abs/2302.00673.

Marcu, Ana-Maria et al. (2024). *LingoQA: Visual Question Answering for Autonomous Driving*. arXiv: 2312.14115 [cs.RO]. URL: https://arxiv.org/abs/2312.14115.

PAVE (2020). *PAVE Poll: Americans Wary of AVs but Say Education and Experience with Technology Can Build Trust*. Accessed: 2024-03-11. URL: https://pavecampaign.org/pave-poll-americans-wary-of-avs-but-say-education-and-experience-with-technology-can-build-trust/.

Radford, Alec et al. (2022). *Robust Speech Recognition via Large-Scale Weak Supervision*. arXiv: 2212.04356 [eess.AS]. URL: https://arxiv.org/abs/2212.04356.

Shao, Hao et al. (2022). *Safety-Enhanced Autonomous Driving Using Interpretable Sensor Fusion Transformer*. arXiv: 2207.14024 [cs.CV]. URL: https://arxiv.org/abs/2207.14024.

Sima, Chonghao et al. (2025). *DriveLM: Driving with Graph Visual Question Answering*. arXiv: 2312.14150 [cs.CV]. URL: https://arxiv.org/abs/2312.14150.

Tang, Zuoyin et al. (2024). *Testing Large Language Models on Driving Theory Knowledge and Skills for Connected Autonomous Vehicles*. arXiv: 2407.17211 [cs.AI]. URL: https://arxiv.org/abs/2407.17211.

Wang, Peng et al. (2024). *Qwen2-VL: Enhancing Vision-Language Model's Perception of the World at Any Resolution*. arXiv: 2409.12191 [cs.CV]. URL: https://arxiv.org/abs/2409.12191.

Wen, Licheng et al. (2023). *On the Road with GPT-4V(ision): Early Explorations of Visual-Language Model on Autonomous Driving*. arXiv: 2311.05332 [cs.CV]. URL: https://arxiv.org/abs/2311.05332.

Xu, Zhenhua et al. (2024). *DriveGPT4: Interpretable End-to-end Autonomous Driving via Large Language Model*. arXiv: 2310.01412 [cs.CV]. URL: https://arxiv.org/abs/2310.01412.

Zhang, Zhejun et al. (2021). *End-to-End Urban Driving by Imitating a Reinforcement Learning Coach*. arXiv: 2108.08265 [cs.CV]. URL: https://arxiv.org/abs/2108.08265.