DASR: Distributed Adaptive Scene Recognition - A Multi-Agent Cloud-Edge Framework for Language-Guided Scene Detection

Can Cui[1,2,*], Yongkang Liu[2], Seyhan Ucar[2], Juntong Peng[1], Ahmadreza Moradipari[2], Maryam Khabazi[2], Ziran Wang[1]
[1]Purdue University, [2]Toyota InfoTech Labs
Correspondence: cancui@purdue.edu

Abstract

The increasing complexity of modern driving systems demands efficient collection and analysis of specific driving scenarios that are crucial for system development and validation. Current approaches either rely on massive data collection followed by manual filtering, or rigid threshold-based recording systems that often miss important edge cases. In this paper, we present Distributed Adaptive Scene Recognition (DASR), a novel multi-agent cloud-edge framework for language-guided scene detection in connected vehicles. Our system leverages the complementary strengths of cloud-based large language models and edge-deployed vision language models to intelligently identify and preserve relevant driving scenarios while optimizing limited on-vehicle buffer storage. The cloud-based LLM serves as an intelligent coordinator that analyzes developer prompts to determine which specialized tools and sensor data streams should be incorporated, while the edge-deployed VLM efficiently processes video streams in real time to make relevant decisions. Extensive experiments across multiple driving datasets demonstrate that our framework achieves superior performance compared to larger baseline models, with exceptional performance on complex driving tasks requiring sophisticated reasoning. DASR also shows strong generalization capabilities on out-of-distribution datasets and significantly reduces storage requirements (28.73 %) compared to baseline methods.

1 Introduction

The field of computer vision has witnessed impressive advancements with the emergence of Vision-Language Models (VLMs) (Tian et al., 2024; Ma et al., 2024), which provide new opportunities for intelligent scene understanding and visual comprehension. These models have evolved from traditional computer vision algorithms, which relied heavily on pre-defined feature extraction and classification, to more sophisticated systems capable of understanding complex visual scenarios through natural language interactions. Recent developments in VLMs, such as GPT-4o (OpenAI et al., 2024) and QwenVL (Bai et al., 2023), have demonstrated unprecedented capabilities in bridging the gap between visual perception and language understanding, enabling more intuitive and flexible scene analysis (Cui et al., 2024a,b).

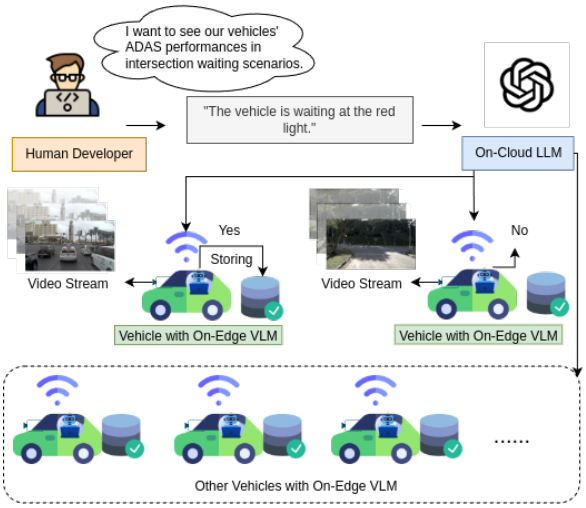

Figure 1: Overview of our multi-agent collaboration framework. Our vehicle space-limited buffers will only sotre the relevent scenes determined by our framework.

Despite these technological advances, the demand for intelligent scene understanding has grown significantly across various domains (Qi et al., 2025; Park et al., 2024). In the development and validation of intelligent driving systems, engineers and developers have a particular interest in collecting and analyzing specific types of scenarios that are crucial for understanding system performance. For instance, vehicle company engineers often need to examine how their systems behave in near-miss cases, such as when a vehicle suddenly brakes ahead, or when a pedestrian appears

[*]Work conducted as an intern at Toyota InfoTech Labs

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track, pages 850–858
November 4-9, 2025

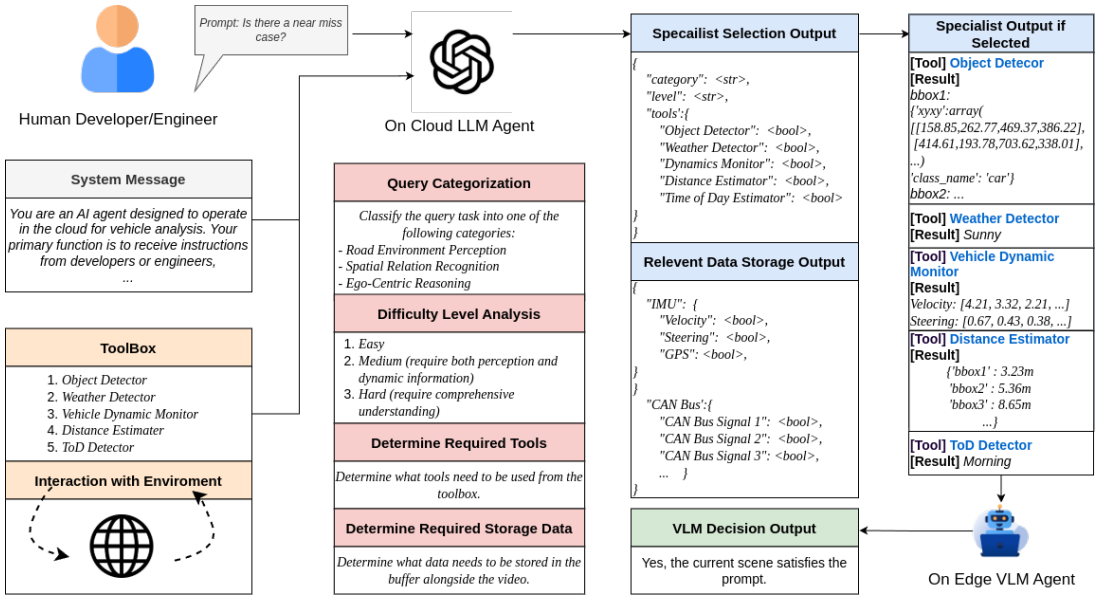

Figure 2: An overview of the proposed framework for smart scene detection.

unexpectedly at the edge of visibility. Other scenarios of interest might include merge interactions on highways, complex intersection negotiations, or instances where multiple road users interact in close proximity. These cases are vital for understanding system behavior, validating safety mechanisms, and identifying areas for improvement (Montanari et al., 2020a; Elspas et al., 2021).

However, efficiently collecting and analyzing such specific scenarios presents significant challenges (Elspas et al., 2021; Elrofai et al., 2016). Current approaches primarily fall into two categories: mass data collection and threshold-based recording. The first approach involves gathering massive amounts of driving data, followed by labor-intensive processes to identify and extract the relevant cases. The second approach relies on predefined CAN bus thresholds - for instance, only recording when specific vehicle parameters exceed certain values, such as brake pedal pressure above 90% or steering angle beyond certain degrees. While these threshold-based methods can reduce data volume, they are often too rigid and may miss important scenarios that do not trigger these predefined thresholds. Additionally, while state-of-the-art VLMs offer powerful scene understanding capabilities, they often require relatively large computational resources, making real-time scenario identification and analysis challenging (Cui et al., 2024d,c). These limitations create significant bottlenecks in the development process, where engineers must either spend considerable time on massive amounts of data or risk missing valuable cases that do not trigger conventional thresholds.

Our **D**istributed **A**daptive **S**cene **R**ecognition (DASR) solution, a multi-agent collaboration between cloud and edge models, presents an opportunity to address these challenges. In this paradigm, the cloud-based LLM serves as an intelligent coordinator that determines which additional inputs from the toolbox should be incorporated to enhance the scene understanding while the core VLM is deployed at the edge and consistently handles the fundamental scene detection tasks. Additionally, the on-cloud LLM also specifies which sensor data streams should be preserved alongside the video footage in the vehicle buffer for comprehensive post-analysis by developers and engineers. This paradigm allows the system to flexibly enhance its analysis capabilities by incorporating the most relevant supplementary augmented data for each specific scenario type. We highlight the contributions of our paper as follows.

- We propose DASR, a novel multi-agent cloud-edge collaboration decision-making framework. Our system uses a cloud-based LLM as an intelligent coordinator to adaptively select specialists and sensor data streams, while an efficient edge-deployed VLM makes real-time scene relevance decisions, addressing the limitations of traditional threshold-based and mass data collection approaches.

- We demonstrate DASR's performance through extensive experiments, achieving 91.35% precision across various autonomous

driving tasks with baselines, outperforming larger baselines while showing exceptional generalization to out-of-distribution datasets (91.93% on DRAMA, 77.68% on HAD).

- We validate DASR's practical utility by reducing storage requirements by 28.73% compared to conventional approaches, enabling more efficient use of limited on-vehicle buffer capacity while maintaining high detection quality for critical driving scenarios.

- We productize DASR as an enterprise solution for automotive manufacturers, demonstrating significant business impact through reduced development time, faster validation cycles, and improved data quality.

2 Problem Definition

Given the complexity of modern vehicle systems, it is crucial for automotive companies to efficiently collect and analyze specific scenarios that are relevant for system development and validation. Due to hardware constraints in commercial vehicles, where the video buffer typically can only store approximately one minute of recording, there is a critical need to intelligently identify and preserve the most relevant scenes. This limitation formulates our task from simple data collection to precise, real-time decision making about which moments are truly valuable for system development and validation. Formally, given an instructive prompt *P* from the developer describing scenarios of interest (e.g., "emergency braking scenarios"), $S = \{v, \theta\}_t^{t+T}$ representing the scene state, where v_t represents vehicle speed information at time t while θ_t represents the steering information, and a continuous video stream $V = \{f_t, f_{t+1}, ..., f_{t+T}\}_t^{t+T}$ from vehicle cameras, our pipeline f aims to determine whether the current frame sequence should be preserved. This can be expressed as:

$$\begin{aligned} \text{Smart Data Collection}: &\quad f(P, V, S) \rightarrow V'; \\ \text{Store Relevent Data}: &\quad [V'] \xrightarrow{Store} B \end{aligned} \tag{1}$$

where $V' \in V$ represents the identified frames of interest that should be stored in the vehicle's limited buffer*B*. Given the buffer constraint $|B| \approx$ 60s, the function f must efficiently identify and preserve only the most relevant segments while operating in real time.

3 Distributed Adaptive Scene Recognition

3.1 Multi-Agent Cloud-Edge Collaboration Framework

We propose a multi-agent cloud-edge collaboration framework that efficiently and intelligently collects relevant data while minimizing vehicle buffer usage and maintaining real-time operation capabilities. Our framework consists of three main components: a cloud-based LLM that serves as an intelligent coordinator, an edge-deployed efficient VLM that processes the video stream in real-time, and an edge toolbox containing various supplementary tools that can be selected. Every time a developer provides a prompt describing their scene of interest, our framework processes it through this sequential pipeline. First, the prompt is sent to the cloud-based LLM, which analyzes it to determine what supplementary tools and vehicle sensor data (CAN Bus, IMU) would be most relevant for detecting such scenes. The LLM's analysis is then transmitted to the edge device, where the selected tools from the toolbox provide additional information to enhance the edge VLM's decision-making capabilities. Using this enriched input, the edge VLM makes more accurate binary decisions about whether the current scene fulfills the prompt requirements. When a scene is identified as relevant, both the video frames and the LLM-specified sensor data are preserved in the vehicle's buffer for later analysis.

3.2 Powerful Cloud-based LLM Coordinator

We utilize OpenAI GPT-4o (OpenAI et al., 2024) as our LLM coordinator. Specifically, our cloud-based LLM implements a hierarchical classification process $g(\cdot)$, categorizing each task into three predefined classes *(Road Environment Perception, Spatial Relation Recognition, Ego Vehicle Centric Reasoning)* and three complexity levels *(Easy, Medium, Hard)*. This classification enables a chain-of-thought reasoning (Wei et al., 2023; Nie et al., 2024) process, following a fundamental principle of information efficiency: we avoid providing superfluous supplementary data for simple tasks such as traffic light state detection, as excessive information could potentially degrade the VLM's decision-making process. Based on this analysis, the LLM outputs both the selected tools O_{tool} and required sensor data specifications O_{store} (e.g., acceleration vectors from IMU, brake pressure signals from CAN Bus) to the edge VLM. For instance, in per-

Table 1: Specialist source and description.

Specialist	Source	Description
Object Detector	YOLOv10-base (Wang et al., 2024a)	Accurately identifies and localizes various objects with bounding box information.
Weather Detector	OpenWeather API (OpenWeather API., 2023)	Identify weather conditions and their impact on visibility and road conditions.
Dynamics Monitor	Vehicle CAN Bus & IMU	Process CAN bus signals and IMU data to interpret complex vehicle dynamic data.
Distance Estimator	MiDas-v3.1-Hybrid (Birkl et al., 2023)	Estimate precise spatial distances for detected objects in the scene.
Time-of-Day Detector	GPS & TomTom API (TomTom, 2023)	Identify lighting conditions for scene understanding across different times of day.

ception tasks, the LLM might select an object detector to provide bounding box information to the edge VLM. In more complex scenarios, such as near-collision detection prompts, the LLM would additionally activate a depth estimator to provide crucial spatial information to the VLM agent. The example output from LLM can be seen in Sec. **??** and the process from LLM is as follows:

$$g(P) \rightarrow [O_{tool}, O_{store}] \tag{2}$$

3.3 Useful Toolbox on Edge

Given LLM's demonstrated capabilities in tool utilization and reasoning, providing domain-specific specialists can significantly enhance the VLM agent's detection accuracy. The toolbox transmits (this process is defined as $l(\cdot)$) specialist output S to the on-edge VLM after receiving the LLM tool decision output O_{tool}:

$$l(O_{tool}) \rightarrow S \tag{3}$$

Our toolbox comprises five categories of specialists, each excelling in their respective domains: Object Detection Specialist for precise entity localization, Vehicle Dynamics Specialist for accurate motion estimation, Distance Estimation Specialist for detailed spatial computation, Temporal Context Specialist for time-specific features, and Weather Recognition Specialist for environmental condition assessment. This multi-specialist design leverages each specialist's domain expertise to complement the VLM's general scene understanding capabilities: While VLMs perform well in general scene comprehension, each specialist provides precise quantitative measurements within their specific domain of expertise. This combination of generalist VLM capabilities with specialized domain expertise enables more robust and accurate scene detection across diverse automotive scenarios. The detailed information of the specialist is in Tab. 1

3.4 Efficient Edge-based VLM Classifier

As mentioned in the previous parts, the edge-deployed VLM serves as the core decision-making component $h(\cdot)$, integrating both the current video stream V and supplementary information S from LLM-selected tools to determine whether the observed scene matches the developer's prompt requirements P. This integration process combines real-time video analysis with tool-generated outputs to make binary decisions about scene relevance. The process is as follows:

$$\begin{aligned} &h(P, T, V) \rightarrow \{1, 0\}, \\ &V' = \begin{cases} V, & \text{if } h(P, S, V) = 1, \\ \text{None}, & \text{if } h(P, S, V) = 0, \end{cases} \\ &[V'] \xrightarrow{\text{Store}} B \end{aligned} \tag{4}$$

One of our first priorities is that we want the on-edge VLM to be efficient and can be deployed on the vehicle side. Therefore, we utilize Qwen2-VL-2B (Wang et al., 2024b) checkpoint as our pretrained foundation checkpoint due to its small size while maintaining acceptable reasoning capabilities. Followed by the visual instruction tuning approach (Liu et al., 2023), we performed comprehensive fine-tuning across the model's visual encoder, LLM components, and projection layers. This end-to-end fine-tuning approach is crucial as our automotive scene understanding task involves domain-specific images that may differ from Qwen2VL's pretraining data, enabling the visual encoder to learn task-specific feature representations. Cross-entropy loss is used during training to optimize the VLM's outputs.

$$Loss = -\sum_{c=1}^{M} y_{o,c} \log(p_{o,c}) \tag{5}$$

Additionally, to ensure optimal deployment efficiency on the vehicle's edge device, we apply Activation-Aware Weight Quantization (AWQ) (Lin et al., 2024) to our fine-tuned model. AWQ enables compression of our VLM from FP16 to 4-bit precision, significantly reducing memory usage while maintaining model performance. This quantization approach is important for automotive applications where computational resources

Models	#Para	Details	Road Env. Perception			Spatial Relations Recog.			Ego-centric Reasoning			Total	mAP
			Trfc. Light	*Wea -ther*	*Road Type*	*Sur. Obj.*	*Trfc. Flow*	*Key Obj.*	*Ego Ctrl.*	*Situ. Asse.*	*Act. Rec.*		
MiniCPM-2.6o	8B	-	56.14	90.67	75.05	68.18	69.06	55.09	69.35	75.31	65.64	69.08	69.39
Qwen2VL-7B	9B	-	51.30	88.24	87.56	57.14	65.46	54.22	72.02	71.82	69.26	70.00	68.56
Qwen2VL-2B	2B	-	43.85	85.81	78.89	52.34	50.87	45.56	43.01	58.19	46.33	54.38	56.09
Qwen2VL-2B	2B	FT	81.56	79.84	**99.73**	**99.38**	82.67	**88.35**	86.06	89.58	96.53	89.79	89.30
Qwen2VL-2B	2B	FT,4-bit	86.30	96.34	99.18	94.47	**87.43**	77.57	85.49	**90.76**	96.51	91.12	90.45
DASR (Ours)	2B	FT,4-bit	**86.33**	**97.57**	99.13	94.16	83.15	81.12	**86.52**	89.59	**97.99**	**91.35**	**90.62**

Table 2: Scene recognition performance on Nuplan-QA-Eval Dataset (Park et al., 2025). The metric used is precision (%). The best-performing model in each task is **bolded**, while the second-best is underlined.

and power consumption are highly constrained, yet high accuracy must be maintained for safety-critical scene detection tasks.

4 System Evaluation

4.1 Implementation Details

We adopted Qwen2-VL-2B (Wang et al., 2024b) as our pretrained foundation model. For fine-tuning, we utilized the Nuplan-QA-Eval (Park et al., 2025) dataset, which we restructured from its original multiple-choice format into a binary question-answering dataset. The processed dataset contains approximately 18,000 training samples and 8,004 test samples. To efficiently fine-tune the vision-language model while maintaining performance comparable to full fine-tuning, we employed the LoRA (Low-Rank Adaptation) technique (Hu et al., 2021). Based on empirical observations, we set the LoRA rank to 8. The model was trained for five epochs using a learning rate of 1e-4, with a batch size of one.

4.2 Experiment Setup

For our experiments, we utilize three test sets: Nuplan-QA-Eval (Park et al., 2025), DRAMA (Malla et al., 2023) and HAD (Kim et al., 2019). Our lightweight VLM was trained on Nuplan-QA-Eval data collected from Boston, Pittsburgh, Las Vegas, and Singapore, making this our primary test set. To verify our framework's scalability across different driving environments, we evaluated on the DRAMA dataset from Tokyo, Japan and the HAD dataset from San Francisco, USA as our Out-Of-Distribution (OOD) test sets. Tokyo and the Bay Area's traffic patterns differ significantly from those in Boston, Pittsburgh, Las Vegas, and Singapore, providing a robust test of generalization. Additionally, the input lengths vary considerably between datasets - DRAMA uses 2-second videos while HAD uses 20-second videos - allowing us to assess the generalizability and scalability of our models across different temporal spans. Since the DRAMA and HAD datasets use a caption format, we developed rule-based converters to transform these into balanced yes/no questions, ensuring an equal distribution of positive and negative answers in our evaluation. Given the limited buffer capacity of on-vehicle storage systems, DASR uses precision as the ideal evaluation metric because it directly measures how accurately the system identifies truly relevant driving scenes, ensuring optimal use of the constrained storage space.

4.3 Intelligent Scene Collection Performance

To validate the scene recognition performance of our framework, we conducted a comprehensive comparison of our 4-bit 2B fine-tuned VLM framework against several baseline models: a standard finetuned-2B-4bit Qwen2VL without our framework, a Finetuned Qwen2VL-2B, Qwen2VL-2B, Qwen2VL-9B, and MiniCPM-2.6o.

As shown in Tab. 2, our framework achieved the highest total precision and mean average precision across all baselines. We showed even substantial improvements over larger models like Qwen2VL-7B and MiniCPM-2.6o. Notably, our framework demonstrated exceptional performance in action recommendation precision (0.9799), weather/condition precision (0.9757), and road type/condition precision (0.9913).

The results conclusively demonstrate that our framework delivers substantial value in complex decision-making tasks such as action recommendation and ego-vehicle maneuver precision, where deeper contextual understanding is critical. This performance pattern validates our core hypothesis that complex driving tasks benefit significantly from decomposition into specialists, each designed to handle specific aspects of scene understanding. Rather than relying on a single model to solve all driving-related challenges, our approach of providing the right specialized tools for each sub-task enables the AI system to achieve higher overall precision, particularly in scenarios requiring so-

Table 3: Performance on OOD datasets. The metric used is precision (%). The best-performance model is **bolded**, while the second-best is underlined.

Method	# Params	FT	Quantilized	DRAMA	HAD
MiniCPM-2.6o	9B	✗	16 float	78.03	69.93
Qwen2VL	9B	✗	16 float	85.09	65.12
Qwen2VL	2B	✗	16 float	83.96	61.69
Qwen2VL	**2B**	✓	16 float	**97.77**	73.17
Qwen2VL	**2B**	✓	4 bit	81.96	70.01
DASR (Ours)	**2B**	✓	4 bit	91.93	**77.68**

phisticated reasoning and decision-making.

4.4 Performance on OOD Dataset

To evaluate generalization capabilities, we tested our framework on two out-of-distribution (OOD) datasets: DRAMA (Tokyo, Japan, 3s-video) and HAD (San Francisco, USA, 20-s video). As shown in Table 3, DASR achieved 91.93% precision on DRAMA and 77.68% on HAD. Results demonstrate that our approach significantly improves precision compared to baseline models without the framework, indicating robust performance even when faced with previously unseen data distributions.

The strong performance across dramatically different geographical and temporal contexts not only validates the framework's transferability but also indicates its potential for deployment in diverse global settings without requiring extensive region-specific retraining. This generalization capability represents a significant advancement toward developing VLMs that can reliably support connected driving systems across varied environments.

4.5 Data Storage Efficiency Performance

We evaluated data storage efficiency using the HAD dataset. Each testing scenario in the HAD dataset contains a 20-second video, with not all frames necessarily relevant to the scene's caption. We aimed to assess the data storage efficiency of our method compared to two baselines: (1) storing all frames indiscriminately (Qian et al., 2024), and (2) using thresholds (Montanari et al., 2020b; Kreutz et al., 2022) (filtering the stopped scenarios using velocity and steering) on sensor or CANBus data to determine which frames to store. Our framework exam a 3-second window, and the window will slide second by second, if the scene satisfies the prompt, the data stream will be stored.

Our DASR framework significantly outperformed both approaches, requiring only 14.34 seconds of storage on average—a 28.73% reduction compared to Baseline One. This improvement

Table 4: Effectiveness in data storage. The best-performance model is **bolded**.

Method	Average Storage (s)	Improvement (%)
Method 1 (Caesar et al., 2022)	20.00	-
Method 2 (Montanari et al., 2020b)	17.30	13.50%
DASR (Ours)	**14.34**	**28.73%**

demonstrates DASR's ability to intelligently identify and preserve only the most relevant portions of each driving scenario, making more efficient use of limited on-vehicle buffer capacity while reducing subsequent data processing requirements.

5 Application Impact and Payoff

DASR aims to deliver substantial practical benefits for driving data collection and analysis with four key projected benefits:

First, we target a 25-30% reduction in storage usage by intelligently preserving only relevant scenes and essential sensor data. Second, we anticipate reducing engineers' data review time by approximately 60% through automated scenario identification. Third, the intelligent selection of complementary CAN bus signals and sensor data enhances analysis quality without additional collection efforts. Finally, these improvements will translate to roughly 40% faster validation cycles for new ADAS features, accelerating the time-to-market.

6 Conclusion

We presented DASR, a multi-agent cloud-edge framework for language-guided scene detection in autonomous vehicles. Our approach distributes tasks between cloud-based LLMs that analyze developer prompts and select appropriate tools, and lightweight edge VLMs that perform real-time scene recognition. Experiments demonstrate superior detection performance (91.35% precision), strong generalization to out-of-distribution datasets (91.93% on DRAMA, 77.68% on HAD), and 28.73% reduction in storage requirements.

In conclusion, DASR accelerates automated driving development by eliminating manual data filtering, enabling faster iteration and validation of ADAS features and it represents a significant advancement toward efficient, intelligent data collection for automated driving, effectively balancing edge deployment constraints with the reasoning capabilities of language models.

Limitations

While our cloud-edge architecture demonstrates promising results for intelligent ADAS data collection, several limitations should be acknowledged:

Cyber Security Risk Our cloud-edge architecture for smart data collection introduces several potential cybersecurity vulnerabilities. The distributed nature of the system creates multiple attack surfaces, including the communication channels between the vehicle and cloud infrastructure, where adversaries could potentially hack or attack transmitted tokens.

Network Dependency The proposed system relies on consistent connectivity between vehicles and cloud infrastructure. In areas with limited network coverage or during connectivity interruptions, the system may temporarily lose its ability to identify valuable data collection opportunities, potentially missing important corner cases.

Limited Multi-Modal Integration The current system primarily focuses on visual data and does not fully leverage other sensor modalities available in modern vehicles, such as LiDAR, radar, or ultrasonic sensors, which could provide complementary information for more robust scene understanding.

References

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *Preprint*, arXiv:2308.12966.

Reiner Birkl, Diana Wofk, and Matthias Müller. 2023. Midas v3.1 – a model zoo for robust monocular relative depth estimation. *arXiv preprint arXiv:2307.14460*.

Holger Caesar, Juraj Kabzan, Kok Seang Tan, Whye Kit Fong, Eric Wolff, Alex Lang, Luke Fletcher, Oscar Beijbom, and Sammy Omari. 2022. Nuplan: A closed-loop ml-based planning benchmark for autonomous vehicles. *Preprint*, arXiv:2106.11810.

Can Cui, Yunsheng Ma, Xu Cao, Wenqian Ye, and Ziran Wang. 2024a. Drive as You Speak: Enabling Human-Like Interaction with Large Language Models in Autonomous Vehicles. In *2024 IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW)*, pages 902–909, Waikoloa, HI, USA. IEEE.

Can Cui, Yunsheng Ma, Xu Cao, Wenqian Ye, Yang Zhou, Kaizhao Liang, Jintai Chen, Juanwu Lu, Zichong Yang, Kuei-Da Liao, Tianren Gao, Erlong Li, Kun Tang, Zhipeng Cao, Tong Zhou, Ao Liu, Xinrui Yan, Shuqi Mei, Jianguo Cao, Ziran Wang, and Chao Zheng. 2024b. A survey on multimodal large language models for autonomous driving. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops*, pages 958–979.

Can Cui, Yunsheng Ma, Zichong Yang, Yupeng Zhou, Peiran Liu, Juanwu Lu, Lingxi Li, Yaobin Chen, Jitesh H. Panchal, Amr Abdelraouf, Rohit Gupta, Kyungtae Han, and Ziran Wang. 2024c. Large language models for autonomous driving (llm4ad): Concept, benchmark, simulation, and real-vehicle experiment. *Preprint*, arXiv:2410.15281.

Can Cui, Zichong Yang, Yupeng Zhou, Juntong Peng, Sung-Yeon Park, Cong Zhang, Yunsheng Ma, Xu Cao, Wenqian Ye, Yiheng Feng, Jitesh Panchal, Lingxi Li, Yaobin Chen, and Ziran Wang. 2024d. On-board vision-language models for personalized autonomous vehicle motion control: System design and real-world validation. *Preprint*, arXiv:2411.11913.

Hala Elrofai, Daniël Worm, and Olaf Op den Camp. 2016. Scenario identification for validation of automated driving functions. In *Advanced Microsystems for Automotive Applications 2016: Smart Systems for the Automobile of the Future*, pages 153–163. Springer.

Philip Elspas, Yannick Klose, Simon T Isele, Johannes Bach, and Eric Sax. 2021. Time series segmentation for driving scenario detection with fully convolutional networks. In *VEHITS*, pages 56–64.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Jinkyu Kim, Teruhisa Misu, Yi-Ting Chen, Ashish Tawari, and John Canny. 2019. Grounding human-to-vehicle advice for self-driving vehicles. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Thomas Kreutz, Ousama Esbel, Max Muhlhauser, and Alejandro Sanchez Guinea. 2022. Unsupervised driving event discovery based on vehicle can-data. In *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, page 4169–4174. IEEE.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. 2024. Awq: Activation-aware weight quantization for llm compression and acceleration. *Preprint*, arXiv:2306.00978.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *Preprint*, arXiv:2304.08485.

Yunsheng Ma, Burhaneddin Yaman, Xin Ye, Feng Tao, Abhirup Mallik, Ziran Wang, and Liu Ren. 2024.

Mta: Multimodal task alignment for bev perception and captioning. *Preprint*, arXiv:2411.10639.

Srikanth Malla, Chiho Choi, Isht Dwivedi, Joon Hee Choi, and Jiachen Li. 2023. Drama: Joint risk localization and captioning in driving. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1043–1052.

Francesco Montanari, Reinhard German, and Anatoli Djanatliev. 2020a. Pattern recognition for driving scenario detection in real driving data. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, pages 590–597. IEEE.

Francesco Montanari, Reinhard German, and Anatoli Djanatliev. 2020b. Pattern recognition for driving scenario detection in real driving data. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, pages 590–597.

Ming Nie, Renyuan Peng, Chunwei Wang, Xinyue Cai, Jianhua Han, Hang Xu, and Li Zhang. 2024. Reason2drive: Towards interpretable and chain-based reasoning for autonomous driving. *Preprint*, arXiv:2312.03661.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

OpenWeather API. 2023. Openweather mobile application.

Sung-Yeon Park, Can Cui, Yunsheng Ma, Ahmadreza Moradipari, Rohit Gupta, Kyungtae Han, and Ziran Wang. 2025. Nuplanqa: A large-scale dataset and benchmark for multi-view driving scene understanding in multi-modal large language models. *Preprint*, arXiv:2503.12772.

SungYeon Park, MinJae Lee, JiHyuk Kang, Hahyeon Choi, Yoonah Park, Juhwan Cho, Adam Lee, and DongKyu Kim. 2024. Vlaad: Vision and language assistant for autonomous driving. In *Proceedings of*

the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops, pages 980–987.

Zhangyang Qi, Zhixiong Zhang, Ye Fang, Jiaqi Wang, and Hengshuang Zhao. 2025. GPT4Scene: Understand 3D Scenes from Videos with Vision-Language Models. *arXiv preprint*. ArXiv:2501.01428.

Tianwen Qian, Jingjing Chen, Linhai Zhuo, Yang Jiao, and Yu-Gang Jiang. 2024. Nuscenes-qa: A multi-modal visual question answering benchmark for autonomous driving scenario. *Preprint*, arXiv:2305.14836.

Xiaoyu Tian, Junru Gu, Bailin Li, Yicheng Liu, Chenxu Hu, Yang Wang, Kun Zhan, Peng Jia, Xianpeng Lang, and Hang Zhao. 2024. DriveVLM: The Convergence of Autonomous Driving and Large Vision-Language Models. *arXiv*.

TomTom. 2023. "real-time traffic data".

Ao Wang, Hui Chen, Lihao Liu, Kai Chen, Zijia Lin, Jungong Han, and Guiguang Ding. 2024a. Yolov10: Real-time end-to-end object detection. *Preprint*, arXiv:2405.14458.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. 2024b. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *Preprint*, arXiv:2409.12191.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models. *Preprint*, arXiv:2201.11903.