2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

V2V-LLM: Vehicle-to-Vehicle Cooperative Autonomous Driving with Multimodal Large Language Models

Hsu-kuang Chiu[1,2] Ryo Hachiuma[1] Chien-Yi Wang[1] Stephen F. Smith[2] Yu-Chiang Frank Wang[1]
Min-Hung Chen[1]

***Abstract*— Current autonomous driving vehicles rely mainly on their individual sensors to understand surrounding scenes and plan for future trajectories, which can be unreliable when the sensors are malfunctioning or occluded. To address this problem, cooperative perception methods via vehicle-to-vehicle (V2V) communication have been proposed, but they have tended to focus on perception tasks like detection or tracking. How those approaches contribute to overall cooperative planning performance is still under-explored. Inspired by recent progress using Large Language Models (LLMs) to build autonomous driving systems, we propose a novel problem setting that integrates a Multimodal LLM into cooperative autonomous driving, with the proposed Vehicle-to-Vehicle Question-Answering (V2V-QA) dataset and benchmark. We also propose our baseline method Vehicle-to-Vehicle Multimodal Large Language Model (V2V-LLM), which uses an LLM to fuse perception information from multiple connected autonomous vehicles (CAVs) and answer various types of driving-related questions: grounding, notable object identification, and planning. Experimental results show that our proposed V2V-LLM can be a promising unified model architecture for performing various tasks in cooperative autonomous driving, and outperforms other baseline methods that use different fusion approaches. Our work also creates a new research direction that can improve the safety of future autonomous driving systems. Our code and dataset are released to facilitate open-source research at https://eddyhkchiu.github.io/v2vllm.github.io/.**

I. Introduction

Autonomous driving technology has advanced significantly due to the evolution of deep learning algorithms, and the release of large-scale real-world driving datasets and benchmarks [1]–[3]. However, the perception and planning systems of autonomous vehicles in daily operation rely mainly on their local LiDAR sensors and cameras to detect notable nearby objects and plan for future trajectories. This approach may encounter safety-critical problems when the sensors are occluded by nearby large objects.

To address this safety-critical issue, recent research proposes cooperative perception algorithms [4]–[9] via vehicle-to-vehicle (V2V) communication. In cooperative driving scenarios, multiple *Connected Autonomous Vehicles (CAVs)* driving nearby to each other share their perception information via V2V communication. The received perception data from multiple CAVs is then fused to generate better overall detection or tracking results. A number of cooperative autonomous driving datasets have been released to the public, including simulated ones [5], [6], [10], [11] and real ones [12]–[15]. These datasets also establish benchmarks to evaluate the performance of cooperative perception algorithms. However, to date, cooperative driving research and datasets have mostly focused on perception tasks. How these state-of-the-art cooperative perception models can be connected with the downstream planning models to generate good cooperative planning results is still under-explored.

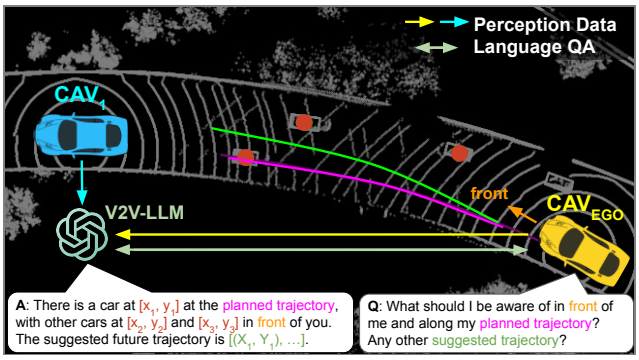


Fig. 1: Overview of our problem setting of LLM-based cooperative autonomous driving. All CAVs share their perception information with the LLM. Any CAV can ask the LLM a question to obtain useful information for driving safety.

Other recent research has attempted to use LLM-based methods to build end-to-end perception and planning algorithms for an individual autonomous vehicle [16]–[21] due to their common-sense reasoning and generalization ability from large-scale pre-trained data. These LLM-based models encode the raw sensor inputs and answer driving-related perception and planning questions. These approaches have shown some promise but have not yet explored the benefits of cooperative perception and planning.

In this paper, we propose and explore a novel problem setting wherein LLM-based methods are used to build end-to-end perception and planning algorithms for *Cooperative Autonomous Driving*, as illustrated in Figure 1. In this problem setting, we assume that there are multiple CAVs and a centralized LLM computing node. All CAVs share their individual perception information with the LLM. Any CAV can ask the LLM a question in natural language to obtain useful information for driving safety. To enable the study of this problem setting, we first create the **Vehicle-to-Vehicle Question-Answering (V2V-QA)** dataset, built upon the V2V4Real [12] and V2X-Real [13] cooperative perception datasets for autonomous driving. Our V2V-QA includes **grounding** , **notable object identification** , and **planning**

[1]NVIDIA, [2]Carnegie Mellon University
The authors thank Boyi Li, Zhiding Yu, Boris Ivanovic, and Marco Pavone from NVIDIA for valuable discussions and feedback.
This research was funded by NVIDIA, the CMU Safety21 University Transportation Center, and CMU Robotics Institute.

TABLE I: Comparison between our V2V-QA and recent related Autonomous Driving (AD) datasets.

Dataset	Publication	# CAVs	RSU	Sim/Real	# Frames	# QA	# QA/frame	Point Cloud	Planning
Cooperative perception in AD									
OPV2V [5]	ICRA 2022	2-7	-	Sim	11K	-	-	✓	
V2X-Sim [10]	RA-L 2022	2-5	✓	Sim	10K	-	-	✓	
V2XSet [6]	ECCV 2022	2-5	✓	Sim	11K	-	-	✓	
DAIR-V2X [14]	CVPR 2022	1	✓	Real	71K	-	-	✓	
V2V4Real [12]	CVPR 2023	2	-	Real	20K	-	-	✓	
TUMTrafV2X [15]	CVPR 2024	1	✓	Real	2K	-	-	✓	
V2X-Real [13]	ECCV 2024	2	✓	Real	33K	-	-	✓	
LLM-based AD									
NuScenes-QA [22]	AAAI 2024	-	-	Real	34K	460K	13.5	✓	
Lingo-QA [23]	ECCV 2024	-	-	Real	28K	420K	15.3		✓
DriveLM [16]	ECCV 2024	-	-	Sim+Real	69K	2M	29.1		✓
TOKEN [19]	CoRL 2024	-	-	Real	28K	434K	15.5		✓
OmniDrive [18]	CVPR 2025	-	-	Real	34K	450K	13.2		✓
V2V-QA (Ours)	-	2	✓	Real	48K	1.45M	**30.2**	✓	✓

question-answer pairs. There are several differences between our novel problem setting and other existing LLM-based driving research [16], [18], [19], [22], [23]. First, our LLM can fuse multiple perception information from different CAVs and provide answers to different questions from any CAV, rather than just serving a single self-driving car. Second, our grounding questions are specially designed to focus on the potential occluded regions of each individual CAV. More differences between our V2V-QA and other related datasets are summarized in Table I.

To establish a benchmark for the V2V-QA dataset, we first propose a strong baseline method: **Vehicle-to-Vehicle Multimodal Large Language Model (V2V-LLM)** for cooperative autonomous driving, as illustrated in Figure 3. Each CAV extracts its own perception features and shares them with V2V-LLM. The V2V-LLM fuses the scene-level feature maps and object-level feature vectors, and then performs vision and language understanding to provide the answer to the input driving-related questions in V2V-QA. We also compare V2V-LLM with other baselines corresponding to different feature fusion methods: *no fusion*, *early fusion*, and *intermediate fusion* [5]–[7], [12], [13]. The results show that V2V-LLM achieves better performance in the grounding, notable object identification, and planning tasks for the cooperative autonomous driving scenarios in general.

Our contribution can be summarized as follows:

- We create and introduce the V2V-QA dataset to support the development and evaluation of LLM-based approaches to end-to-end cooperative autonomous driving. V2V-QA includes grounding, notable object identification, and planning question-answering tasks.
- We propose a baseline method V2V-LLM for cooperative autonomous driving to provide an initial benchmark for V2V-QA. This method fuses scene-level and object-level features provided by multiple CAVs, and answers different CAV's driving-related questions.
- We create a benchmark for V2V-QA and show that V2V-LLM outperforms other baselines on the grounding, notable object identification, and planning tasks in general, indicating the potential of V2V-LLM to be a foundation model for cooperative autonomous driving.

II. Related Work

A. Cooperative Perception in Autonomous Driving

Cooperative perception [24] algorithms were proposed to address the potential occlusion problem in individual autonomous vehicles. Pioneering work F-Cooper [4] proposes the first intermediate fusion approach that merges feature maps to achieve good cooperative detection performance. More recent work, AttFuse [5], V2X-ViT [6], and CoBEVT [7] integrate attention-based models to aggregate features for cooperative detection and tracking.

From a dataset perspective [25], [26], simulation datasets OPV2V [5], V2X-Sim [10], and V2XSet [6] were first generated for cooperative perception research. More recently, real datasets have been collected. V2V4Real [12] is the first worldwide available real vehicle-to-vehicle cooperative perception dataset with perception benchmarks. V2X-Real [13], DAIR-V2X [14], and TUMTraf-V2X [15] further include sensor data from roadside infrastructures.

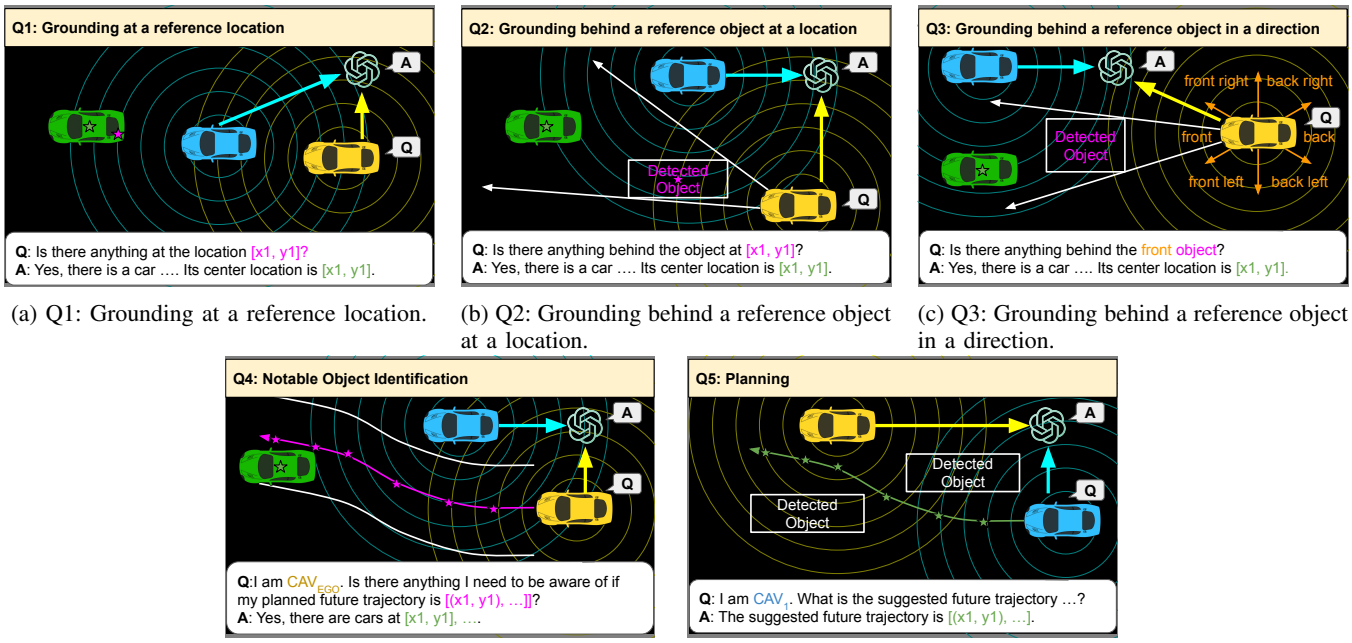
Different from this group of research, our problem setting and proposed dataset include both perception and planning question-answering tasks for multiple CAVs. Our proposed V2V-LLM also adopts a novel LLM-based fusion approach.

B. LLM-based Autonomous Driving

Language-based planning models [27]–[29] and more recent Multimodal Large Language Model (MLLM)-based approaches [16]–[19], [30] encode the driving scene and ego-vehicle's state into text and visual features and use them as input to the LLM. Then the LLM generates text output including the suggested action or future trajectory.

From a dataset perspective, several LLM-based autonomous driving datasets have been built on top of existing autonomous driving datasets. For example, NuPrompt [31], NuScenes-QA [22], and NuInstruct [32] create captioning, perception, prediction, and planning QA pairs based on the NuScenes [2] dataset. DriveLM [16] adopts real data from NuScenes [2] and simulated data from CARLA [33] to have larger-scale and more diverse driving QAs.

Different from those LLM-based driving research that only supports individual autonomous vehicles, our problem setting and proposed V2V-QA dataset are designed for cooperative driving with multiple CAVs.



(a) Q1: Grounding at a reference location.

(b) Q2: Grounding behind a reference object at a location.

(c) Q3: Grounding behind a reference object in a direction.

(d) Q4: Notable object identification.

(e) Q5: Planning.

Fig. 2: Illustration of V2V-QA's 5 types of QA pairs. The arrows pointing at LLM indicate the perception data from CAVs.

III. V2V-QA Dataset

To enable the research in our proposed novel problem setting: LLM-based cooperative autonomous driving, we create the **Vehicle-to-Vehicle Question-Answering (V2V-QA)** dataset to benchmark different models' performance on fusing perception information and answering safety-critical driving-related questions.

A. Problem Setting

Our proposed V2V cooperative autonomous driving with LLM problem is illustrated in Figure 1. In this setting, we assume there are multiple Connected Autonomous Vehicles (CAVs) and a centralized LLM computing node. All CAVs share their individual perception information, such as scene-level or object-level features, with the centralized LLM. Any CAV can ask the LLM a question in natural language to obtain information for driving safety. The LLM aggregates the received perception information from multiple CAVs and provides a natural language answer to the CAV's question. In this research, the questions and answers include **grounding (Q1-3)**, **notable object identification (Q4)**, and **planning (Q5)**, as illustrated in Figure 2.

B. Dataset Details

Our V2V-QA dataset contains two splits: **V2V-split** and **V2X-split**, which are built on top of V2V4Real [12] and V2X-Real [13] datasets, respectively. These base datasets are collected by driving two vehicles with LiDAR sensors simultaneously near to each other. These datasets also includes 3D bounding box annotations for other objects in the driving scenes. In V2V4Real [12], the training set has 32 driving sequences and a total of 7105 frames of data per CAV, and the testing set has 9 driving sequences and a total of 1993 frames of data per CAV. In V2X-Real [13], the training set has 43 driving sequences and a total of 5772 frames of data per CAV, and the testing set has 9 driving sequences and a total of 1253 frames of data per CAV. The frame rate is 10Hz. In V2X-Real [13], some driving scenes also provide LiDAR point clouds from roadside infrastructures. We also include them as perception inputs to the LLM with the same approach as using CAVs' LiDAR point clouds. We follow the same training and testing settings from V2V4Real [12] and V2X-Real [13] when building our V2V-split and V2X-split. Table II summarizes the numbers of QA pairs in our proposed V2V-QA's V2V-split and V2X-split. We have 1.45M QA pairs in total and 30.2 QA pairs per frame on average.

TABLE II: Dataset statistics of our V2V-QA's V2V-split and V2X-split. Q1: Grounding at a reference location. Q2: Grounding behind a reference object at a location. Q3: Grounding behind a reference object in a direction. Q4: Notable object identification. Q5: Planning.

QA type	V2V-split		V2X-split		Total
	Training	Testing	Training	Testing	
Q1	354820	121383	495290	128711	1100204
Q2	35700	13882	167694	35233	252509
Q3	14339	5097	28740	6465	54641
Q4	12290	3446	6274	1708	23718
Q5	12290	3446	6274	1708	23718
Total	429439	147254	704272	173825	1454790

C. Question and Answer Pairs Curation

For each frame of V2V4Real [12] and V2X-Real [13] datasets, we create 5 different types of QA pairs, including 3 types of grounding questions, 1 type of notable object identification question, and 1 type of planning question. These QAs are designed for cooperative driving scenarios. To generate instances of these QA pairs, we use V2V4Real [12]

and V2X-Real [13]'s ground-truth bounding box annotations, each CAV's ground-truth trajectories, and individual detection results as the source information. Then we use different manually designed rules based on the geometric relationship among the aforementioned entities and text templates to generate our QA pairs. The text template can be seen in Figures 4 and 5. The generation rule of each QA type is described as follows.

Q1. Grounding at a reference location (2a): In this type of question, we ask the LLM to identify whether an object that occupies a specific query 2D location exists. If so, the LLM is expected to provide the center location of the object. Otherwise, the LLM should indicate that there is nothing at the reference location. We use the center locations of ground-truth boxes and every CAV's individual detection result boxes as the query locations in the questions. By doing so, we can focus more on evaluating each model's cooperative grounding ability on the potential false positive and false negative detection results.

Q2. Grounding behind a reference object at a location (2b): When a CAV's field of view is occluded by a nearby large detected object, this CAV may want to ask the centralized LLM to determine whether there exists any object behind that occluding large object given the fused perception information from all CAVs. If so, the LLM is expected to return the object's location and the asking CAV may need to drive more defensively or adjust its planning. Otherwise, the LLM should indicate that there is nothing behind the reference object. We use the center location of each detection result box as the query locations in these questions. We draw a sector region based on the relative pose of the asking CAV and the reference object, and select the closest ground-truth object in the region as the answer.

Q3. Grounding behind a reference object in a direction (2c): We further challenge the LLM on language and spatial understanding ability by replacing Q2's reference 2D location with a reference directional keyword. We first get the closest detection result box in each of the 6 directions of a CAV as the reference object. Then we follow the same approach in Q2 to get the closest ground-truth box in the corresponding sector region as the answer.

Q4. Notable object identification (2d): The aforementioned grounding tasks are intermediate tasks in the autonomous driving pipeline. More critical abilities of autonomous vehicles involve both identifying notable objects near planned future trajectories and adjusting future planning to avoid potential collisions. We extract 6 waypoints from the ground-truth trajectory in the next 3 seconds as the reference future waypoints in the questions. Then we get, at most, the 3 closest ground-truth objects within 10 meters of the reference future trajectory as the answer.

Q5. Planning (2e): Planning is important because the ultimate goal of autonomous vehicles is to navigate through complex environments safely and avoid any potential collision in the future. To generate the planning QAs, we extract 6 future waypoints, evenly distributed in the next 3 seconds, from each CAV's ground-truth future trajectory as the answer. Our V2V-QA's planning task is more challenging than other NuScenes [2]-based LLM-driving related works for a couple reasons. First, we support multiple CAVs in cooperative driving scenarios. The LLM model needs to provide different answers depending on which CAV is asking, while prior works only need to generate planning results for a single autonomous vehicle. Second, our V2V-QA's ground-truth planning trajectories are more diverse. V2V-QA contains both urban and highway driving scenarios, while NuScenes [2] only includes urban driving scenarios.

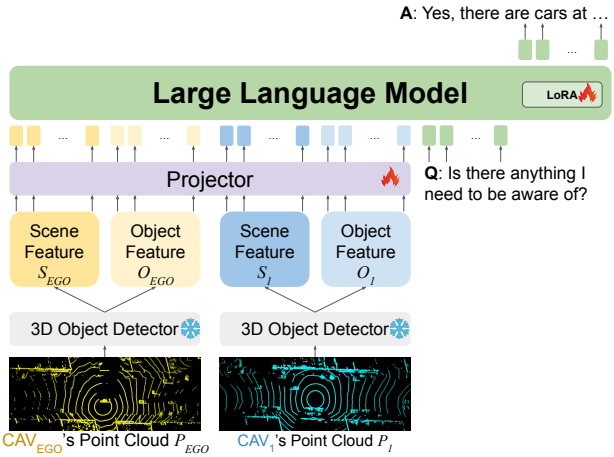


Fig. 3: Model diagram of our proposed V2V-LLM for cooperative autonomous driving.

D. Evaluation Metrics

We follow prior works [18], [19]'s approach to evaluate model performance. For the grounding questions (Q1, Q2, Q3) and the notable object identification question (Q4), the evaluation metric is F1 score. For the planning question (Q5), the evaluation metrics are L2 errors and collision rates.

IV. V2V-LLM

We propose a competitive baseline model, **V2V-LLM**, for this LLM-based collaborative driving problem, as shown in Figure 3. Our model is a Multimodal LLM (MLLM) that takes the individual perception features of every CAV as the vision input, a question as the language input, and generates an answer as the language output.

A. LiDAR-based Input Features

For extracting the perception input features, each CAV applies a 3D object detection model to its individual LiDAR point cloud: P_{EGO} and P_1. We extract the scene-level feature map S_{EGO} and S_1 from the 3D object detection model and transform the 3D object detection results as the object-level feature vectors O_{EGO} and O_1. Following prior works V2V4Real [12] and V2X-Real [13], we use PointPillars [34] as the 3D object detector for fair comparisons.

B. LiDAR-based Multimodal LLM

Model architecture: We utilize LLaVA [35] to develop our MLLM, given its superior performance on visual question-answering tasks. However, since the perception features of

our cooperative driving tasks are LiDAR-based instead of RGB images used by LLaVA [35], we use a LiDAR-based 3D object detector as the point cloud feature encoder, as described in the previous section, instead of LLaVA [35]'s CLIP [36] image feature encoders. We then feed the resulting features to a multi-layer perceptron-based projector network for feature alignment from the point cloud embedding space to the language embedding space. The aligned perception features are the input perception tokens digested by the LLM together with the input language tokens from the question. Finally, the LLM aggregates the perception information from all CAVs and returns an answer based on the question.

Training: We use 8 NVIDIA A100-80GB GPUs to train our model. Our V2V-LLM uses LLaVA-v1.5-7b [35]'s Vicuna [37] as the LLM backbone. To train our model, we initialize it by loading the pre-trained LLaVA-v1.5-7b [35]'s checkpoint. We freeze the LLM and the point cloud feature encoder, and finetune the projector and the LoRA [38] parts of the model. During training, we use batch size 32. Adam optimizer is adopted for training with a starting learning rate $2e-5$ and a cosine learning rate scheduler with a 3% warm-up ratio. For all other training settings and hyperparameters, we use the same ones from LLaVA-v1.5-7b [35].

V. Experiment

A. Baseline Methods

We follow V2V4Real [12] and V2X-Real [13] to establish a benchmark for our proposed V2V-QA dataset with experiments on baseline methods using different fusion approaches: **no fusion**, **early fusion**, **intermediate fusion**, and our proposed baseline, **LLM fusion** (3). All baseline methods also adopt the same projector and LLM architecture as in our V2V-LLM but with different point cloud feature encoders. In some driving sequences of V2X-split that have point clouds from roadside infrastructures, we include them as perception input as well in the same way as using CAVs' point clouds.

No fusion: Only a single CAV's LiDAR point cloud is fed to a single 3D object detector to extract the scene-level and object-level features as the LLM's visual input. The performance is expected to be worse than all other cooperative perception approaches.

Early fusion: The LiDAR point cloud from two CAVs is merged first. Then the merged point cloud is used as input to a 3D object detector to extract the visual features as the visual input to the LLM. This approach requires much higher communication cost and is less practical for deployment on real-world autonomous vehicles.

Intermediate fusion: Prior work CoBEVT [7], V2X-ViT [6], and AttFuse [5] propose cooperative detection models that merge feature maps from multiple CAVs via attention mechanisms. Such approaches require less communication cost and can still achieve good performance. In our benchmark, we extract both of the scene-level and object-level features from those cooperative detection models as the input tokens to the LLM.

LLM fusion: We categorize our proposed V2V-LLM as a new type of fusion method, *LLM fusion*, which lets each CAV perform its individual 3D object detection to extract the scene-level feature maps and object-level feature vectors, and uses the Multimodal LLM to fuse the features from multiple CAVs. This approach is related to the traditional *late fusion* method that performs individual 3D object detection and aggregates the results by non-maximum suppression (NMS). Instead of applying NMS, our method adopts LLM to perform more tasks than just detection.

B. Quantitative Results

1) Performance: **Grounding**: Our V2V-LLM and baseline methods' performance on V2V-QA's 3 types of grounding questions can be seen in Table III for V2V-split and V2X-split, respectively. CoBEVT [7] is not included in V2X-split's result because V2X-Real [13] does not release its CoBEVT [7] baseline model. In average, V2V-LLM achieves similar performance in V2V-split and outperforms all other baseline methods in V2X-split. Such results indicate that our V2V-LLM has a promising capability of fusing perception features from multiple CAVs to answer grounding questions.

Notable Object Identification: Table III show the performance on the notable object identification task (Q4). Our proposed V2V-LLM outperforms all other methods in both V2V-split and V2X-split. Compared with the aforementioned grounding tasks, this notable object identification task requires more spatial understanding ability to identify the objects close to the planned future waypoints. For such a task, our V2V-LLM, which lets the Multimodal LLM perform both perception feature fusion and question answering, achieves the best results.

Planning: Table III show the performance of the planning task (Q5) for V2V-split and V2X-split, respectively. Our proposed V2V-LLM outperforms other methods in this safety-critical task to generate a future trajectory that aims to avoid potential collisions.

2) Communication Cost and Scaling Analysis: In our centralized setting, each CAV sends one scene-level feature map ($\leq$ 0.2MB), one set of individual object detection result parameters ($\leq$ 0.003MB), one question ($\leq$ 0.0002MB) to the LLM computing node and receives one answer ($\leq$ 0.0002MB) at each timestep. If there are N_v CAVs and each asks N_q questions, the communication cost of each CAV is $(0.2+0.003+(0.0002+0.0002)N_q) = (0.203+0.0004N_q)$ MB, and the cost of the LLM is $(0.2+0.003+(0.0002+0.0002)N_q)N_v = (0.203N_v+0.0004N_qN_v)$ MB, as shown in Table IV. Note that each CAV only needs to send the same features to the LLM computing node once at each timestep because the LLM node can save and reuse them to answer multiple questions from the same or different CAVs at the same timestep.

3) Summary: In general, V2V-LLM achieves the best results in the notable object identification and planning tasks, which are critical in autonomous driving applications. V2V-LLM also achieves competitive results in the grounding tasks. In terms of communication costs, V2V-LLM only

TABLE III: V2V-LLM's testing performance in V2V-QA's V2V-split and V2X-split in comparison with baseline methods. Q1: Grounding at a reference location. Q2: Grounding behind a reference object at a location. Q3: Grounding behind a reference object in a direction. Q_{Gr}: Average of grounding (Q1, Q2, and Q3). Q4: Notable object identification. Q5: Planning. L2: L2 distance error. CR: Collision rate. Comm: Communication cost. In each column, the **best** results are in boldface, and the second-best results are in underline.

Method	V2V-split							V2X-split							Comm(MB) ↓
	Q1	Q2	Q3	Q_{Gr}	Q4	Q5		Q1	Q2	Q3	Q_{Gr}	Q4	Q5		
	F1 ↑	F1 ↑	F1 ↑	F1 ↑	F1 ↑	L2 (m) ↓	CR (%) ↓	F1 ↑	F1 ↑	F1 ↑	F1 ↑	F1 ↑	L2 (m) ↓	CR (%) ↓	
No Fusion	66.6	22.6	17.2	35.5	47.3	6.55	4.57	55.7	21.4	25.2	34.1	64.4	2.31	9.21	**0**
Early Fusion	**73.5**	23.3	20.8	39.2	53.9	6.20	3.55	59.7	23.3	26.1	36.4	67.6	2.12	8.61	1.9208
Intermediate Fusion															
AttFuse [5]	70.7	26.4	18.4	38.5	56.9	6.83	4.12	58.9	23.9	26.3	36.4	65.9	2.19	8.39	0.4008
V2X-ViT [6]	70.8	28.0	**22.6**	40.5	57.6	7.08	4.33	59.6	24.2	26.1	36.6	65.0	2.29	8.86	0.4008
CoBEVT [7]	72.2	29.3	21.3	**40.9**	57.6	6.72	3.88	-	-	-	-	-	-	-	0.4008
LLM Fusion															
V2V-LLM (Ours)	70.0	**30.8**	21.2	40.7	**59.7**	**4.99**	**3.00**	**60.5**	**25.3**	**26.7**	**37.5**	**69.3**	**1.71**	**6.89**	0.4068

TABLE IV: Communication cost (MB) and scaling analysis. N_v: number of CAVs. N_q: number of questions asked by each CAV at each timestep.

Setting	Each CAV	Centralized LLM
Centralized	$0.203 + 0.0004N_q$	$0.203N_v + 0.0004N_qN_v$

TABLE V: Perception and planning performance comparison with non-LLM baseline method.

Method	Q1	Q5	
	F1 ↑	$L2_{avg}$ (m) ↓	CR_{avg} (%) ↓
CoBEVT [7] + BEV-planner [39]	65.7	5.82	11.59
V2V-LLM (ours)	**70.0**	**4.99**	**3.00**

increases communication costs by 1.5% in comparison to other intermediate fusion baseline methods.

C. Comparison to Non-LLM baseline

To compare our V2V-LLM with non-LLM baseline in V2V-QA, for perception, we use the detection results from V2V4Real [12]'s best cooperative detection model CoBEVT [7] checkpoint (the same one as the feature extractor in our experiment) and evaluate them in our Q1 (grounding at a location). For planning, since there is no prior cooperative planning research on V2V4Real [12], we use the BEV features from the same CoBEVT [7] model checkpoint as input to initialize and train a BEV-planner [39], which is the best non-LLM planning baseline method in prior work OmniDrive [18]. Our V2V-LLM still outperforms the non-LLM cooperative perception and planning baseline, as shown in Table V.

D. Robustness Assessment

We follow V2X-ViT [6] to experiment on the impact of latency and sensor noise on positional errors. Our model is robust to these factors, as shown in Tables VI and VII.

E. Ablation Study

Input Features: We experiment with variants of our V2V-LLM model that use either only the scene-level feature maps or only the object-level feature vectors as the visual input.

TABLE VI: Experiments in V2V-split with communication latency.

Latency (s)	Q1	Q2	Q3	Q_{Gr}	Q4	Q5	
	F1 ↑	F1 ↑	F1 ↑	F1 ↑	F1 ↑	$L2_{avg}$ (m) ↓	CR_{avg} (%) ↓
1.0	69.3	29.7	18.9	39.3	55.0	5.26	4.09
0.4	69.7	30.3	20.1	40.0	56.0	5.09	3.49
0.3	69.8	30.7	20.6	40.4	57.2	5.07	3.31
0.2	69.8	**30.8**	20.8	40.5	57.7	5.05	3.21
0.1	69.8	30.7	21.1	40.5	59.4	5.02	3.05
0	**70.0**	**30.8**	**21.2**	**40.7**	**59.7**	**4.99**	**3.00**

TABLE VII: Experiments in V2V-split with positional errors. STD: standard deviation.

STD (m)	Q1	Q2	Q3	Q_{Gr}	Q4	Q5	
	F1 ↑	F1 ↑	F1 ↑	F1 ↑	F1 ↑	$L2_{avg}$ (m) ↓	CR_{avg} (%) ↓
1.0	69.8	29.9	**21.7**	40.5	57.2	5.21	3.86
0.4	69.8	**30.9**	21.5	**40.7**	59.2	5.03	3.27
0.3	69.8	30.7	21.0	40.5	59.1	5.00	3.20
0.2	69.8	**30.9**	21.0	40.6	**60.0**	4.99	3.10
0.1	69.8	30.8	21.3	40.6	59.8	**4.98**	3.05
0	**70.0**	30.8	21.2	**40.7**	59.7	4.99	**3.00**

TABLE VIII: Ablation study in V2V-split.

Method	Q1	Q2	Q3	Q_{Gr}	Q4	Q5	
	F1 ↑	F1 ↑	F1 ↑	F1 ↑	F1 ↑	L2 (m) ↓	CR (%) ↓
Scene only	69.9	15.4	17.9	34.4	43.2	7.21	15.55
Object only	69.0	26.9	17.6	37.8	52.6	5.24	7.78
Scratch	67.6	26.5	17.2	37.1	49.3	6.30	5.01
V2V-LLM	**70.0**	**30.8**	**21.2**	**40.7**	**59.7**	**4.99**	**3.00**

The ablation results can be seen in Table VIII. Both types of features contribute to final performance in all QA tasks. In general, the object-level-only model outperforms the scene-level-only model. This implies that the object-level features are easier for LLM to digest, which is consistent with the results observed in the previous work with the TOKEN model [19].

Training from Scratch: Table VIII also shows that training from scratch achieves worse performance, meaning that pre-training with LLaVA's VQA tasks improves our V2V-LLM's performance in V2V-QA.

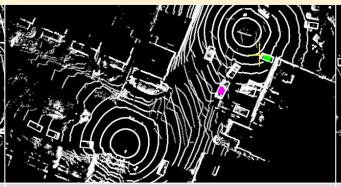
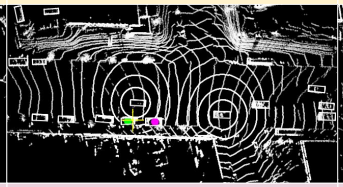
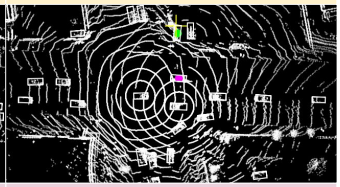
Q1: Grounding at a reference location	Q2: Grounding behind a reference object at a location	Q3: Grounding behind a reference object in a direction	Q3: Grounding behind a reference object in a direction
			
Q: Is there anything at the location [73.0, -0.4]?	**Q**: Is there anything behind the object at [-30.4, 17.1]?	**Q**: Is there anything behind the front object?	**Q**: Is there anything behind the front right object?
V2V-LLM: Yes, there is a car at the location. Its center location is [73.0, -0.4]. **GT**: Yes, there is a car at the location. Its center location is [73.8, -0.4].	**V2V-LLM**: Yes, there is a car behind the object. Its center location is [-42.8, 28.1]. **GT**: Yes, there is a car behind the object. Its center location is [-45.2, 27.3].	**V2V-LLM**: Yes, there is a car behind the front object. Its center location is [28.7, -1.8]. **GT**: Yes, there is a car behind the front object. Its center location is [30.3, -2.0].	**V2V-LLM**: Yes, there is a car behind the front right object. Its center location is [1.0, 25.6]. **GT**: Yes, there is a car behind the front right object. Its center location is [0.5, 23.3].

Fig. 4: V2V-LLM's *grounding* results on V2V-QA's testing set. Magenta ○: reference locations in questions. Yellow +: model output locations. Green ○: ground-truth answers.

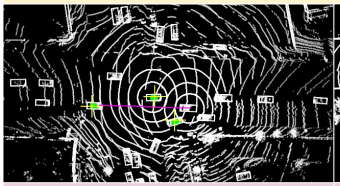
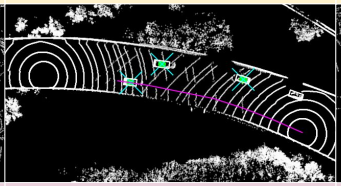
Q4: Notable Object Identification	Q4: Notable Object Identification	Q5: Planning	Q5: Planning
			
Q:I am CAV_EGO. Is there anything I need to be aware of if my planned future trajectory is [(4.2,0.1),(8.9,0.2),(13.5,0.4),(18.4,0.6),(23.5,0.7),(28.8,0.9)]?	**Q**: I am CAV_1. Is there anything I need to be aware of if my planned future trajectory is [(-73.0,-14.0),(-63.7,-10.4),(-54.2,-7.3),(-43.5,-5.1),(-33.6,-3.1),(-23.5,-1.6)]?	**Q**: I am CAV_EGO. What is the suggested future trajectory to avoid collision with nearby objects?	**Q**: I am CAV_1. What is the suggested future trajectory to avoid collision with nearby objects?
V2V-LLM: Yes, there are cars at [4.8, -3.8], [11.1, 3.6], [31.1, 0.5], which are close to your planned future trajectory. **GT**: Yes, there are cars at [4.7, -4.4], [11.5, 3.3], [30.6, 0.7], which are close to your planned future trajectory.	**V2V-LLM**: Yes, there are cars at [-27.6, -2.0], [-37.6, 3.7], [-63.3, -1.1], which are close to your planned future trajectory. **GT**: Yes, there are cars at [-27.5, -2.0], [-37.5, 3.7], [-63.3, -1.1], which are close to your planned future trajectory.	**V2V-LLM**: The suggested future trajectory is [(6.2,0.2),(12.6,0.3),(19.2,0.4),(25.9,0.5),(32.6,0.6),(39.4,0.7)]. **GT**: The suggested future trajectory is [(6.2,0.2),(12.1,0.4),(17.6,0.6),(22.9,0.8),(28.3,1.0),(33.5,1.1)].	**V2V-LLM**: The suggested future trajectory is [(5.3,0.1),(10.7,0.2),(16.4,0.3),(22.2,0.4),(28.2,0.5),(34.4,0.7)]. **GT**: The suggested future trajectory is [(5.7,0.1),(11.5,0.2),(17.4,0.4),(23.3,0.5),(29.3,0.6),(35.1,0.7)].

Fig. 5: V2V-LLM's *notable object identification* and *planning* results on V2V-QA's testing set. For notable object identification, Magenta curve: planned future trajectories in questions. Green ○: ground-truth notable object locations. Yellow + and Cyan ×: model identification outputs corresponding to CAV_EGO and CAV_1, respectively. For planning, Green line: future trajectories in ground-truth answers. Yellow curve and Cyan curve: model planning outputs corresponding to CAV_EGO and CAV_1, respectively.

F. Qualitative Results

Figure 4 shows our V2V-LLM's *grounding* results and the ground truth with visualization on V2V-QA's testing set. We can observe that our V2V-LLM is able to locate the objects given the provided reference information for each of the 3 types of grounding questions: grounding at a reference location, grounding behind a reference object at a location, and grounding behind a reference object in a direction. Figure 5's left part shows our V2V-LLM's *notable object identification* results. V2V-LLM demonstrate its capability of identifying multiple objects near the planned future trajectories specified in the questions for each CAV. Figure 5's right part shows V2V-LLM's *planning* results. Our model is able to suggest future trajectories that avoid potential collisions with nearby objects. Overall, the outputs of our model closely align with the ground-truth answers across all question types, indicating its robustness in cooperative autonomous driving tasks.

VI. Conclusion

In this work, we expand the research scope of cooperative autonomous driving by integrating the use of Multimodal LLM-based methods, aimed at improving the safety of future autonomous driving systems. We propose a new problem setting and create a novel V2V-QA dataset and benchmark that includes grounding, notable object identification, and planning question-answering tasks designed for varieties of cooperative driving scenarios. We propose a baseline model V2V-LLM that fuses each CAV's individual perception information and performs visual and language understanding to answer driving-related questions from any CAV. Our proposed V2V-LLM outperforms other baselines adopted from state-of-the-art cooperative perception algorithms in the grounding, notable object identification, and planning. Our method also outperforms non-LLM baseline and is robust to communication latency and noise. These experimental results indicate that V2V-LLM is promising as a unified multimodal foundation model that can effectively perform perception and planning tasks for cooperative autonomous driving. We have publicly released our V2V-QA dataset and V2V-LLM code to facilitate open-source research, and believe it will bring cooperative driving research to the next stage.

REFERENCES

[1] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[2] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[3] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, *et al.*, "Scalability in perception for autonomous driving: Waymo open dataset," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[4] Q. Chen, X. Ma, S. Tang, J. Guo, Q. Yang, and S. Fu, "F-cooper: Feature based cooperative perception for autonomous vehicle edge computing system using 3d point clouds," in *ACM/IEEE Symposium on Edge Computing (SEC)*, 2019.

[5] R. Xu, H. Xiang, X. Xia, X. Han, J. Li, and J. Ma, "Opv2v: An open benchmark dataset and fusion pipeline for perception with vehicle-to-vehicle communication," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2022.

[6] R. Xu, H. Xiang, Z. Tu, X. Xia, M.-H. Yang, and J. Ma, "V2x-vit: Vehicle-to-everything cooperative perception with vision transformer," in *European Conference on Computer Vision (ECCV)*, 2022.

[7] R. Xu, Z. Tu, H. Xiang, W. Shao, B. Zhou, and J. Ma, "Cobevt: Cooperative bird's eye view semantic segmentation with sparse transformers," in *Conference on Robot Learning (CoRL)*, 2022.

[8] H.-k. Chiu, C.-Y. Wang, M.-H. Chen, and S. F. Smith, "Probabilistic 3d multi-object cooperative tracking for autonomous driving via differentiable multi-sensor kalman filter," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[9] H.-k. Chiu and S. F. Smith, "Selective communication for cooperative perception in end-to-end autonomous driving," in *IEEE International Conference on Robotics and Automation (ICRA) Workshop*, 2023.

[10] Y. Li, D. Ma, Z. An, Z. Wang, Y. Zhong, S. Chen, and C. Feng, "V2x-sim: Multi-agent collaborative perception dataset and benchmark for autonomous driving," *IEEE Robotics and Automation Letters (RA-L)*, 2022.

[11] J. Cui, H. Qiu, D. Chen, P. Stone, and Y. Zhu, "Coopernaut: End-to-end driving with cooperative perception for networked vehicles," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[12] R. Xu, X. Xia, J. Li, H. Li, S. Zhang, Z. Tu, Z. Meng, H. Xiang, X. Dong, R. Song, H. Yu, B. Zhou, and J. Ma, "V2v4real: A real-world large-scale dataset for vehicle-to-vehicle cooperative perception," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[13] H. Xiang, Z. Zheng, X. Xia, R. Xu, L. Gao, Z. Zhou, X. Han, X. Ji, M. Li, Z. Meng, L. Jin, M. Lei, Z. Ma, Z. He, H. Ma, Y. Yuan, Y. Zhao, and J. Ma, "V2x-real: a largs-scale dataset for vehicle-to-everything cooperative perception," in *Europian Conference on Computer Vision (ECCV)*, 2024.

[14] H. Yu, Y. Luo, M. Shu, Y. Huo, Z. Yang, Y. Shi, Z. Guo, H. Li, X. Hu, J. Yuan, and Z. Nie, "Dair-v2x: A large-scale dataset for vehicle-infrastructure cooperative 3d object detection," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[15] W. Zimmer, G. A. Wardana, S. Sritharan, X. Zhou, R. Song, and A. C. Knoll, "Tumtraf v2x cooperative perception dataset," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[16] C. Sima, K. Renz, K. Chitta, L. Chen, H. Zhang, C. Xie, P. Luo, A. Geiger, and H. Li, "Drivelm: Driving with graph visual question answering," in *Europian Conference on Computer Vision (ECCV)*, 2024.

[17] X. Tian, J. Gu, B. Li, Y. Liu, Y. Wang, Z. Zhao, K. Zhan, P. Jia, X. Lang, and H. Zhao, "Drivevlm: The convergence of autonomous driving and large vision-language models," *arXiv preprint arXiv:2402.12289*, 2024.

[18] S. Wang, Z. Yu, X. Jiang, S. Lan, M. Shi, N. Chang, J. Kautz, Y. Li, and J. M. Alvarez, "OmniDrive: A holistic vision-language dataset for autonomous driving with counterfactual reasoning," in *CVPR*, 2025.

[19] R. Tian, B. Li, X. Weng, Y. Chen, E. Schmerling, Y. Wang, B. Ivanovic, and M. Pavone, "Tokenize the world into object-level knowledge to address long-tail events in autonomous driving," in *Conference on Robot Learning (CoRL)*, 2024.

[20] L. Chen, O. Sinavski, J. Hünermann, A. Karnsund, A. J. Willmott, D. Birch, D. Maund, and J. Shotton, "Driving with llms: Fusing object-level vector modality for explainable autonomous driving," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[21] W. Wang, J. Xie, C. Hu, H. Zou, J. Fan, W. Tong, Y. Wen, S. Wu, H. Deng, Z. Li, *et al.*, "Drivemlm: Aligning multi-modal large language models with behavioral planning states for autonomous driving," *arXiv preprint arXiv:2312.09245*, 2023.

[22] T. Qian, J. Chen, L. Zhuo, Y. Jiao, and Y.-G. Jiang, "Nuscenes-qa: A multi-modal visual question answering benchmark for autonomous driving scenario," in *AAAI Conference on Artificial Intelligence (AAAI)*, 2024.

[23] A.-M. Marcu, L. Chen, J. Hünermann, A. Karnsund, B. Hanotte, P. Chidananda, S. Nair, V. Badrinarayanan, A. Kendall, J. Shotton, *et al.*, "Lingoqa: Visual question answering for autonomous driving," in *European Conference on Computer Vision (ECCV)*, 2024.

[24] T. Huang, J. Liu, X. Zhou, D. C. Nguyen, M. R. Azghadi, Y. Xia, Q.-L. Han, and S. Sun, "V2x cooperative perception for autonomous driving: Recent advances and challenges," *arXiv preprint arXiv:2310.03525*, 2023.

[25] M. Liu, E. Yurtsever, J. Fossaert, X. Zhou, W. Zimmer, Y. Cui, B. L. Zagar, and A. C. Knoll, "A survey on autonomous driving datasets: Statistics, annotation quality, and a future outlook," *IEEE Transactions on Intelligent Vehicles (T-IV)*, 2024.

[26] M. Yazgan, M. V. Akkanapragada, and J. M. Zöllner, "Collaborative perception datasets in autonomous driving: A survey," in *IEEE Intelligent Vehicles Symposium (IV)*, 2024.

[27] J. Mao, Y. Qian, J. Ye, H. Zhao, and Y. Wang, "Gpt-driver: Learning to drive with gpt," in *Advances in Neural Information Processing Systems (NeurIPS) Workshop (Foundation Models for Decision Making)*, 2023.

[28] J. Mao, J. Ye, Y. Qian, M. Pavone, and Y. Wang, "A language agent for autonomous driving," in *Conference On Language Modeling (COLM)*, 2024.

[29] B. Li, Y. Wang, J. Mao, B. Ivanovic, S. Veer, K. Leung, and M. Pavone, "Driving everywhere with large language model policy adaptation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[30] T.-H. Wang, A. Maalouf, W. Xiao, Y. Ban, A. Amini, G. Rosman, S. Karaman, and D. Rus, "Drive anywhere: Generalizable end-to-end autonomous driving with multi-modal foundation models," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2023.

[31] D. Wu, W. Han, T. Wang, Y. Liu, X. Zhang, and J. Shen, "Language prompt for autonomous driving," *arXiv preprint*, 2023.

[32] D. Xinpeng, H. Jinahua, X. Hang, L. Xiaodan, H. Xu, Z. Wei, and L. Xiaomeng, "Holistic autonomous driving understanding by bird's-eye-view injected multi-modal large models," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[33] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Conference on Robot Learning (CoRL)*, 2017.

[34] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[35] H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual instruction tuning," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[36] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *International Conference on Machine Learning (ICML)*, 2021.

[37] W.-L. Chiang, Z. Li, Z. Lin, Y. Sheng, Z. Wu, H. Zhang, L. Zheng, S. Zhuang, Y. Zhuang, J. E. Gonzalez, I. Stoica, and E. P. Xing, "Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality," 2023. [Online]. Available: https://lmsys.org/blog/2023-03-30-vicuna/

[38] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "LoRA: Low-rank adaptation of large language models," in *International Conference on Learning Representations (ICLR)*, 2022.

[39] Z. Li, Z. Yu, S. Lan, J. Li, J. Kautz, T. Lu, and J. M. Alvarez, "Is ego status all you need for open-loop end-to-end autonomous driving?" in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.