

TB-Bench: Training and Testing Multi-Modal AI for Understanding Spatio-Temporal Traffic Behaviors from Dashcam Images/Videos

Supplementary Material

This material includes the following sections:

- **Discussions:** The broader impact, limitations, and future directions of our work.
- **Access Information:** A URL for accessing the benchmark, datasets, and future update.
- **Task Definitions and Dataset Statistics:** A detailed overview of the task definitions and relevant dataset statistics.
- **Data Generation Pipeline:** Insights into the Data Generation Pipeline used in our study.
- **Evaluation Details:** Information on metrics, models, and evaluation methods.
- **Experiments and Results:** Implementation details, quantitative analyses, qualitative results, and ablation studies.

8. Discussions

8.1. Broader Impact

This study represents progress in enhancing the capabilities of Multi-Modal Large Language Models (MLLMs) by focusing on a limited set of AD perception tasks. Specifically, we introduce a new benchmark to evaluate MLLMs on understanding diverse traffic behaviors and provide high-quality VLIT datasets that enhance MLLMs' generalizability. We hope this will advance MLLMs' applications in AD, contributing to the development of more robust autonomous driving systems.

8.2. Limitations

Firstly, our study utilizes the moderate large language models (Qwen 0.5B series) due to limited computational resources, which can be scaled up as needed.

Secondly, we acknowledge the dataset imbalance arising from the natural occurrence of specific autonomous driving behaviors; please refer to Section Dataset Statistics for more details.

Lastly, the free-form text output templates in TB-100k and TB-250k are limited for certain tasks. However, we believe that the diversity of images is also important for the model to understand visual concepts. That being said, when combined with other (vision-)language instruction tuning datasets, our datasets still enhance the performance of MLLMs, enabling them to generalize better in traffic domains, particularly in understanding traffic behaviors.

8.3. Future Work

Future research could expand this work by incorporating a wider range of perception tasks or by exploring subsequent stages, such as prediction and planning.

Additionally, an important direction for future investigation is the optimal application of upstream perception tuning sets, including the TB-100k and TB-250k datasets, to relevant downstream traffic tasks. This approach may enhance model performance in real-world applications.

Furthermore, integrating real-time traffic data, such as video feeds and sensory inputs, could improve the MLLMs' understanding of dynamic traffic situations. Finally, enhancing the explainability of MLLMs in traffic behavior scenarios will help users understand the rationale behind model predictions.

9. Access to the Benchmark and Datasets

9.1. Availability

The Traffic Behavior Benchmark (TB-Bench) and the training datasets (TB-100k, TB-250k) will be publicly available at the following Github repository:

- <https://github.com/TB-AD/TB-Bench-110k-250k>

The source code for conducting and analyzing the experiments will also be publicly available in the repository upon publication, permitting free use for research purposes.

9.2. Future Update

We also plan to establish an evaluation server and leaderboard on HuggingFace in the future. Any updates will be communicated through the above Github repository to ensure users have access to the latest information.

10. Benchmark and Datasets

10.1. Task Definition

10.1.1. Relative Distance (RD).

The task is to predict the Euclidean distance in meters between two entities in an image; see Figure 14 for two examples.

10.1.2. Spatial Reasoning (SR).

The task is to predict the spatial position of one entity relative to another from the perspective of a reference entity; see Figure 15 for examples. Specifically, the relationship between two objects is defined by the angle θ , as follows:

$$\text{Relation} = \begin{cases} \text{front} & \text{if } -30^\circ < \theta \leq 30^\circ, \\ \text{front left} & \text{if } 30^\circ < \theta \leq 90^\circ, \\ \text{front right} & \text{if } -90^\circ < \theta \leq -30^\circ, \\ \text{back left} & \text{if } 90^\circ < \theta \leq 150^\circ, \\ \text{back right} & \text{if } -150^\circ < \theta \leq -90^\circ, \\ \text{back} & \text{otherwise.} \end{cases} \quad (1)$$

This angular relationship is similar to that defined in [35].

10.1.3. Orientation Reasoning (OR).

This task is to predict the facing relationship between two entities from the perspective of a reference entity, categorized as: ‘similar’, ‘opposite’, or ‘perpendicular’. Please refer to Figure 16 for examples. The relationship is defined based on the absolute difference in facing angles $|\theta|$, as follows:

$$\text{Relation} = \begin{cases} \text{similar} & \text{if } 0^\circ \leq |\theta| \leq 45^\circ, \\ \text{opposite} & \text{if } 135^\circ \leq |\theta| \leq 180^\circ, \\ \text{perpendicular} & \text{otherwise.} \end{cases} \quad (2)$$

It is noted that this angle is measured from the facing direction of a reference entity to the position of the target entity in Euclidean space, irrespective of the target entity’s facing direction.

10.1.4. Other Lane to Ego-Vehicle (EGO-LANE).

This task is to predict the lane of a target vehicle relative to the ego-vehicle’s perspective; see Figure 17 for examples. The categories include: ‘front lane’, ‘front left lane’, ‘front right lane’, and ‘oncoming traffic lane’ (the lane on the opposite side of the road).

It is noted that when the ego-vehicle is on a road with multiple lanes, the ‘front lane’ is further classified into three fine-grained categories: ‘front lane’, ‘front left lane’, and ‘front right lane’.

10.1.5. Other Lane Changing (OBJ-LANE).

This task is to predict whether the target vehicle is changing lanes, categorized as ‘left lane change’, ‘right lane change’, or ‘no change’; see Figure 18 for examples. Lane changes are evaluated based on the target vehicle’s viewpoint. For instance, if the target vehicle in the oncoming traffic lane executes a right lane change, the ego vehicle perceives it as moving to the left.

10.1.6. Other Turning (OBJ-TURN).

This task is to predict whether the target vehicle is making a turn, categorized as ‘turning left’, ‘turning right’, or ‘go straight’. The target vehicle is considered to be turning, if it changes direction by more than 25 degrees within a period of 1.6 seconds. Please refer to Figure 19 for examples.

10.1.7. Ego Turning (EGO-TURN).

This task is to predict whether the ego-vehicle is making a turn, categorized as turning left, turning right, or going straight. The turning maneuver of the ego-vehicle is also defined by a change in direction of more than 25 degrees within a period of 1.6 seconds. Please refer to Figure 20 for examples.

10.1.8. Ego Traverse Distance (EGO-TRA).

This task is to predict the traverse distance of the ego vehicle in meters over a period of 1.6 seconds. Please see Figure 21 for examples.

10.2. Dataset Statistics

Table 7, 8, and 9 show the distribution of categories for the TB-Bench, TB-100k, and TB-250k datasets, respectively, detailing the count and percentage of samples for various task types.

To create the TB-Bench, we manually screened the frames thoroughly to select samples with clearly visible target entities. Each task in TB-Bench has an equal count of 250 samples. We ensure that the distribution of categories in each task closely resembles that of the instruction tuning datasets.

It is seen from Table 8, and 9 that TB-250k represents a normal scene occurrence distribution in real-world scenarios, while TB-100k is a more label-balanced version.

11. Data Generation Pipeline Details

11.1. Information Extraction

Figure 4 shows the extraction process. It begins with obtaining raw sensory data from input samples, which may include static images with entity attributes from datasets like KITTI or ONCE, or sequential data from Argoverse2. This sensory data is processed to filter out insignificant scene information.

For Argoverse2, lane geometry information is processed concurrently. Lane coordinates are used to create polygons with attributes, such as neighboring, successor, and predecessor lanes. This information helps determine lane direction and angle, which are then projected onto vehicle attributes to obtain the vehicle’s lane ID and relevant lane information. This data is subsequently passed to the next processing step to extract all scene attributes.

11.2. Rule-based Q&A Generation

The process begins with obtaining attribute data from either the nodes or edges of the relationship graph. This data is then processed through rule-based functions to extract behavioral or spatial information. Next, we generate behavioral attributes in a Q&A format using templates provided in Table 10.

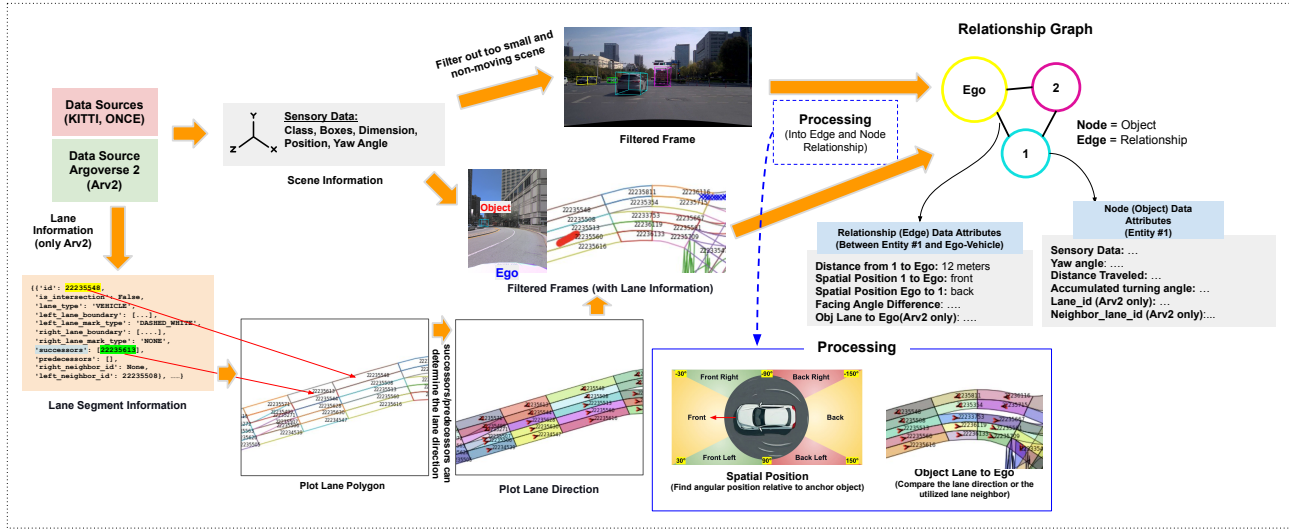


Figure 4. Data Extraction Process.

Table 7. TB-Bench Statistics

Task Type	Category	Count	Percentage (%)
Relative Distance	numerical value	250	12.5
Spatial Reasoning	back	61	3.0
	back left	30	1.5
	back right	9	0.4
	front	87	4.3
	front left	45	2.2
	front right	18	0.9
Orientation Reasoning	numerical value	122	6.1
	opposite	51	2.5
	perpendicular	16	0.8
Other Lane to Ego-Vehicle	similar	61	3.0
	front lane	71	3.5
	front left lane	40	2.0
Other Lane Changing	front right lane	31	1.6
	oncoming traffic lane	108	5.4
Other Turning	left lane change	62	3.1
	no change	142	7.1
	right lane change	46	2.3
Ego Turning	go straight	126	6.3
	left turn	67	3.4
	right turn	57	2.9
Ego Traverse Distance	go straight	122	6.1
	left turn	38	1.9
	right turn	90	4.5
Ego Traverse Distance	numerical value	250	12.5

Table 8. TB-100k Statistics

Task Type	Category	Count	Percentage (%)
Relative Distance	numerical value	10000	9.1
Spatial Reasoning	back	3580	3.3
	back left	3183	2.9
	back right	3115	2.8
	front	7873	7.2
	front left	7321	6.7
	front right	4928	4.5
Orientation Reasoning	numerical value	10000	9.1
	opposite	10013	9.1
	perpendicular	2387	2.2
Other Lane to Ego-Vehicle	similar	7600	6.9
	front lane	3889	3.5
	front left lane	3231	2.9
Other Lane Changing	front right lane	4182	3.8
	oncoming traffic lane	8698	7.9
Other Turning	left lane change	414	0.4
	no change	807	0.7
	right lane change	279	0.3
Ego Turning	go straight	744	0.7
	left turn	435	0.4
	right turn	321	0.3
Ego Traverse Distance	go straight	753	0.7
	left turn	331	0.3
	right turn	416	0.4
Ego Traverse Distance	numerical value	15500	14.1

Generation depends on the task type. Tasks 1-4 and task 8 ('Relative Distance,' 'Spatial Reasoning,' 'Orientation Reasoning,' 'Other Lane to Ego,' and 'Ego Traverse Distance') can be created in any frame, as their attributes

are available in all frames.

In contrast, tasks 5-7 ('Other Lane Changing,' 'Other Turning,' and 'Ego Turning') require a triggering event, specifically a change in attributes. The following details

Table 9. TB-250k Statistics

Task Type	Category	Count	Percentage (%)
Relative Distance	numerical value	34721	13.7
Spatial Reasoning	back	17023	6.7
	back left	6247	2.5
	back right	3966	1.6
	front	26917	10.6
	front left	10793	4.3
Orientation Reasoning	front right	4804	1.9
	numerical value	34872	13.7
	opposite	19242	7.6
Other Lane to Ego-Vehicle	perpendicular	3355	1.3
	similar	12283	4.8
	front lane	14312	5.6
Other Lane Changing	front left lane	4454	1.8
	front right lane	6401	2.5
	oncoming traffic lane	24833	9.8
	left lane change	414	0.2
Other Turning	no change	807	0.3
	right lane change	279	0.1
	go straight	744	0.3
Ego Turning	left turn	435	0.2
	right turn	321	0.1
	go straight	753	0.3
Ego Traverse Distance	left turn	331	0.1
	right turn	416	0.2
	numerical value	25000	9.9

explain how to trigger an event:

Event Triggering: Other Lane Changing

- Check if the current `lane_id` is in the `future_right_neighbor_id`.
If yes, then assign: **Right Lane Change**.
- Check if the current `lane_id` is in the `future_left_neighbor_id`.
If yes, then assign: **Left Lane Change**.
- If neither condition is met, assign: **No Change**.

Note: `future_right_neighbor_id` refers to the `right_neighbor_id` of the next time step; the same applies to the left side.

Event Triggering: Other Turning

- Check if the accumulated object yaw angle is greater than 25 degrees in 1.6 seconds.
If yes, then assign: **Turn Left**.
- Check if the accumulated object yaw angle is less than -25 degrees in 1.6 seconds.
If yes, then assign: **Turn Right**.
- If neither condition is met, assign: **Go straight**.

Event Triggering: Ego Turning

- Check if the accumulated ego-vehicle yaw angle is greater than 25 degrees in 1.6 seconds.
If yes, then assign: **Turn Left**.
- Check if the accumulated ego-vehicle yaw angle is less than -25 degrees in 1.6 seconds.
If yes, then assign: **Turn Right**.
- If neither condition is met, assign: **Go straight**.

11.3. Q&A Augmentation

The augmentation process converts short question-answer (Q&A) pairs into natural language sentences. Each short QA pair was expanded into a full sentence using a pre-defined structure. We employ the Microsoft-Phi3-medium model to generate these sentences, using the following prompt:

Complete Prompt

```
system_text = "You are a language expert assistant. In this task, we want to expand the following answer to longer wording but no additional information."
full_prompt = f"{system_text}. The question is: {question} and the short answer is {answer}. Give the complex answer in a short sentence no more than 15 words."
```

The parameters for `{question}`, and `{answer}` are dynamically inserted for each instance. This approach ensures that the augmented data remains concise (up to 15 words) while incorporating the original short answer in a more elaborated context, maintaining the correctness and relevance of the response.

11.4. Pre-crash Scenarios

Figure 5 presents the full list of 65 pre-crash scenarios as described in Section Task Design, based on National Automotive Sampling System. Each scenario is categorized into a specific accident type, such as ‘Animal’, ‘Off-road’, etc.

12. Evaluation Details

12.1. Evaluation Metrics

As mentioned in the main paper, we employ the rule-based methods for evaluation. Figure 6 shows the keyword list and regular expression used in the evaluation pipeline.

12.2. Additional Details on Evaluated Models

In this study, we evaluate open-source state-of-the-art models and proprietary models on our TB-Bench in a zero-shot manner. We provide additional information for the evaluated models in Table 11.

Table 10. **Q&A Templates.** The placeholder `<entity_n>` refers to any entity, such as ‘Entity #1’, ‘Entity #2’, or ‘Ego-vehicle’, ensuring that no sentence contains duplicate entities. ‘Short Answer Template’ denotes a basic class of concise responses that can be expanded into more complex sentences.

Task Type	Question Template	Short Answer Template
Relative Distance	Can you measure straight-line distance in meters between <code><entity_n></code> and <code><entity_n></code> ? How far is <code><entity_n></code> from <code><entity_n></code> in meters? How many meters apart are <code><entity_n></code> and <code><entity_n></code> ? What is distance from <code><entity_n></code> to <code><entity_n></code> along road’s surface in meters?	xx.xx meters
Spatial Reasoning	How are <code><entity_n></code> and <code><entity_n></code> spatially related, from <code><entity_n></code> perspective? What is spatial position of <code><entity_n></code> relative to <code><entity_n></code> ? What is spatial relation of <code><entity_n></code> to <code><entity_n></code> ?	back, back left, back right, front, front left, front right
Orientation Reasoning	How do you describe orientation of <code><entity_n></code> relative to <code><entity_n></code> , similar, opposite or perpendicular? How is <code><entity_n></code> oriented relative to <code><entity_n></code> , similar, opposite or perpendicular? What is angle between <code><entity_n></code> and <code><entity_n></code> , in degrees? What is facing angle of <code><entity_n></code> relative to <code><entity_n></code> , in degrees? What is orientation of <code><entity_n></code> relative to <code><entity_n></code> , similar, opposite or perpendicular? What is yaw angle different between <code><entity_n></code> and <code><entity_n></code> , in degrees?	opposite, perpendicular, similar, xx.xx degrees
Other Lane to Ego-Vehicle	How would you describe lane position of Entity#1? Options: front lane, front left lane, front right lane, or oncoming traffic lane.	front_lane, front_left_lane, front_right_lane, oncoming_traffic_lane
Other Lane Changing	How would you describe driving scene involving Entity#1? Please explain, focusing on vehicle’s lane change maneuver.	left_lane_change, no_change, right_lane_change
Other Turning	How would you describe driving scene involving Entity#1? Please explain, focusing on vehicle’s turning maneuver.	go_straight, left_turn, right_turn
Ego Turning	How would you describe driving scene involving our car? Please explain, focusing on our car’s turning maneuver.	go_straight, left_turn, right_turn
Ego Traverse Distance	How far has our car driven and what kind of steering maneuver did it perform in current scene?	xx.xx meters

The first category consists of open-source models (LLaVA, Bunny, and InternVL), which are accessible via the Hugging Face API. These models are fully fine-tuned with specific settings for each version available in their Huggingface repositories.

The second category consists of proprietary models (GPT-4o and Gemini), which require specific API calls and image formatting. It is noted that we evaluate the latest version of these models on our TB-Bench at the time of submission.

12.3. Prompt for Zero-Shot Evaluation

For zero-shot evaluation of existing models, we use an Option Template that presents multiple-choice options to define possible answer classes. This approach accommodates the varied terminology that pre-trained models may employ

to describe situations.

The details of the Option Template, which varies based on the task type, are as follows:

Option Template

Distance-Related Tasks:

- Answer in `xx.x` meters format.

Angle-Related Tasks:

- Answer in `xx.x` degrees format.

Tasks with Predefined Answer Choices:

- Retrieve the answer choices.
- Assign a letter to each choice (e.g., A, B, C).
- Present options as follows:

```
Options:
A. choice1,
B. choice2,
C. choice3, ...
```

Table 11. Additional information of the models evaluated on TB-Bench.

Model Name	Full Repository/API Name	Vision Part	Language Part
Open-source models			
LLaVA-1.5-7B	llava-hf/llava-1.5-7b-hf	CLIP-L/14	Vicuna-7b-v1.5
LLaVA-v1.6-Mistral-7B	llava-hf/llava-v1.6-mistral-7b-hf	CLIP-L/14	Mistral-7B-Instruct-v0.2
LLaVA-NeXT-Video-7B	llava-hf/LLaVA-NeXT-Video-7B-hf	CLIP-L/14	Vicuna-7B-v1.5
LLaVA-Interleave-Qwen-7B	llava-hf/llava-interleave-qwen-7b-hf	SigLIP-L/14	Qwen1.5-7B-Chat
Bunny-v1.1-4B	BAAI/Bunny-v1.1-4B	SigLIP-L/14	Phi-3-mini-4k-instruct
Bunny-v1.1-Llama-3-8B-V	BAAI/Bunny-v1.1-Llama-3-8B-V	SigLIP-L/14	Llama-3-8B-Instruct
InternVL2-8B	OpenGVLab/InternVL2-8B	InternViT-300M-448px	Qwen2-8B-Instruct
Magma-8B	microsoft/Magma-8B	ConvNext-XXlarge	LLaMA-3-8B
Mini-InternVL2-1B-DriveLM	OpenGVLab/Mini-InternVL2-1B-DA-DriveLM	InternViT-300M-448px	Qwen2-0.5B
DriveLM-mantis-8b	francepfl/DriveLM-mantis-8b-idefics2.8192	SigLIP	Mistral-7B-v0.1
Proprietary models			
Gemini-1.5-flash	Gemini-1.5-flash	Unknown	Unknown
GPT-4o-2024-08-06	GPT-4o-2024-08-06	Unknown	Unknown

Pre-trained models often use specific vocabularies based on their training data. For instance, a model might say ‘opposite side of the road’ instead of ‘oncoming traffic lane’ if it lacks specific instruction training. By offering explicit choices, the model can select the appropriate terminology despite variations.

For numerical answers, we specify the expected format within the prompt to ensure clarity and consistency, such as instructing the model to Answer in xx.x meters format.

This structured approach allows the model to account for variations in wording and select the most appropriate option, demonstrating its understanding.

13. Experiments and Results

13.1. Implementation Details

Table 12. Hyper-parameter settings for finetuning our models on TB-100k or TB-250k.

Hyper-parameter	Value
Epochs	10
Warmup steps	2,000
Learning rate	1e-5
LoRA learning rate	1e-4
Effective Batch size	64
AdamW β	(0.9, 0.999)
Weight decay	0.05
Drop path	0
Attention dropout	0
Torch data type	bf16
Inference temperature	0

All models are finetuned on an Ubuntu 20.04 server equipped with four A6000 GPUs, each with 48GB of memory. The source code is built on the Transformers library

[47] and utilizes the PyTorch 2.4 framework [34].

Additional information on hyper-parameter settings for finetuning our baseline models on TB-100k and TB-250k is presented in Table 12.

13.2. Quantitative Analyses

We provide quantitative analyses and the qualitative results of the model’s predictions on TB-Bench. The baseline model ((SigLIP-L/14 and Qwen1.5-0.5b) finetuned on TB-100k. For numerical output tasks, we visualize error distributions using box plots. On the other hand, we use confusion matrices for classification tasks.

13.2.1. Relative Distance and Ego Traverse Distance Tasks.

Figure 7 shows the box plot for distance errors of our model predictions on the two tasks. For RD, distance errors are generally centered around zero, with a narrow interquartile range, indicating consistent performance, though a few outliers suggest overestimation. Predictions on EGO-TRA show a similar error distribution, with the median slightly above zero and more positive outliers, indicating a tendency to overestimate distance.

13.2.2. Orientation Reasoning Task.

Figure 8 shows the box plot for angular errors of our model predictions on the Orientation Reasoning (OR) task. The median and interquartile range are close to zero, indicating precise and consistent predictions. Short whiskers further highlight this accuracy. Outliers are grouped near 0, 90, and 180 degrees, suggesting small angle misestimations. Overall, the model demonstrates minimal errors in this task.

13.2.3. Spatial Reasoning Task.

Figure 9 shows the confusion matrix of our model predictions on the Spatial Reasoning (SR) task. The ‘front’ position is classified most accurately at 85.1%, while ‘back’ and ‘back left’ positions have lower accuracies of 63.3% and

No.	Scenario Definition
1	Animal: other
2	Animal: vehicle going straight and animal in road
3	Animal: vehicle negotiating a curve and animal in road
4	Off-road: single vehicle performing avoidance maneuver
5	Off-road: single vehicle going straight and departing road edge
6	Off-road: single vehicle going straight and losing control
7	Off-road: single vehicle initiating a maneuver and departing road edge
8	Off-road: single vehicle initiating a maneuver and losing control
9	Off-road: single vehicle negotiating a curve and departing road edge
10	Off-road: single vehicle negotiating a curve and losing control
11	Off-road: single vehicle and other loss of control
12	Off-road: single vehicle due to vehicle failure
13	Off-road: single vehicle and other road edge departure
14	Off-road: single vehicle with other/unknown
15	Off-road: backing
16	Off-road: no impact
17	Pedalcyclist: other/unknown
18	Pedalcyclist: vehicle going straight on crossing paths
19	Pedalcyclist: vehicle going straight on parallel paths
20	Pedalcyclist: vehicle starting in traffic lane on crossing paths
21	Pedalcyclist: vehicle turning left on crossing paths
22	Pedalcyclist: vehicle turning left on parallel paths
23	Pedalcyclist: vehicle turning right on crossing paths
24	Pedalcyclist: vehicle turning right on parallel paths
25	Pedestrian: other
26	Pedestrian: vehicle backing
27	Pedestrian: vehicle going straight and pedestrian crossing road
28	Pedestrian: vehicle going straight and pedestrian darting onto road
No.	Scenario Definition
29	Pedestrian: vehicle going straight and pedestrian playing/working on Road
30	Pedestrian: vehicle going straight and pedestrian walking along road
31	Pedestrian: vehicle turning left and pedestrian crossing road
32	Pedestrian: vehicle turning right and pedestrian crossing road
33	Backing: at driveways
34	Backing: at intersections
35	Backing: other
36	Lane change: 2 vehicles going straight and 1 vehicle encroaching in same lane
37	Lane change: 2 vehicles going straight and 1 vehicle encroaching into another lane
38	Lane change: 1 vehicle going straight and another changing lanes
39	Lane change: 1 vehicle going straight and another entering or leaving parking position
40	Lane change: 1 vehicle going straight and another passing
41	Lane change: 1 vehicle going straight and another turning
42	Lane change: 2 vehicles in other combinations
43	Lane change: 1 vehicle passing and another turning
44	Opposite direction: control loss
45	Opposite direction: 2 vehicles going straight and 1 vehicle encroaching
46	Opposite direction: 2 vehicles going straight both in same lane
47	Opposite direction: 2 vehicles negotiating a curve and 1 vehicle encroaching
48	Opposite direction: 2 vehicles negotiating a curve both in same lane
49	Opposite direction: other/unknown
50	Opposite direction: involves 1 vehicle passing
51	Opposite direction: involves vehicle failure
52	Rear-end: following vehicle changing lanes
53	Rear-end: lead vehicle accelerating
54	Rear-end: lead vehicle changing lanes
55	Rear-end: lead vehicle decelerating
56	Rear-end: lead vehicle moving at constant, slower speed
57	Rear-end: lead vehicle stopped
58	Rear-end: other/unknown
59	Crossing paths: left turn across path from lateral direction (LTAP/LD)
60	Crossing paths: left turn across path from opposite direction (LTAP/OD)
61	Crossing paths: left turn into path (LTIP)
62	Crossing paths: other/unknown
63	Crossing paths: right turn across path from lateral direction (RTAP/LD)
64	Crossing paths: right turn into path (RTIP)
65	Crossing paths: straight crossing paths (SCP)

Figure 5. List of pre-crash scenarios based on National Automotive Sampling System (NASS) variables.

66.7%. The matrix also shows moderate confusion between similar positions, such as back left' being misclassified as front right' (23.33%) and 'back' as 'front' (19.67%).

13.2.4. Other Lane to Ego-Vehicle Task.

Figure 10 shows the confusion matrix of our model predictions on the Other Lane to Ego-Vehicle (EGO-LANE) task. Overall, the model shows high accuracy on most categories (over 96%), except for the 'front lane,' which has an accuracy of only 81.7%. The primary misclassification pattern involves confusion between the 'front lane' and its adjacent lanes, with 9.9% of 'front lane' samples being misclassified as 'front right lane.'

13.2.5. Other Lane Changing Task.

Figure 11 shows the confusion matrix on the Other Lane Changing (OBJ-LANE) task, where samples are categorized into 'no change,' 'left lane change,' and 'right lane change.' In this case, the model shows decent performance with an accuracy of around 78.87% in the 'no change' category. However, it struggles significantly with lane change predictions. For both 'left lane change' and 'right lane change' classifications, the most misclassified predictions are in the 'no change' category, with 32.3% and 30.4% misclassified, respectively. This indicates the model's difficulty in distinguishing between lane changes and no change, underscoring the task's challenges.

13.2.6. Other Lane Changing Task.

Figure 12 shows the confusion matrix on the Other Turning (OBJ-TURN) task, where samples are categorized as 'left turn,' 'go straight,' and 'right turn.' The model excels in identifying the go straight' category, achieving an accuracy of 80.16%. However, it shows over 30% misclassification rates for both 'left turn' and 'right turn.' Notably, misclassifications of 'left turn' are nearly evenly divided between 'right turn' and go 'straight,' despite 'right turn' errors being more theoretically opposed. The model's performance indicates that it struggles to accurately interpret turns from the perspective of other vehicles, influenced by road orientation and vehicle positioning.

13.2.7. Ego Turning Task.

Figure 13 shows the confusion matrix on the task, where the actions are categorized as 'left turn,' 'go straight,' and 'right turn.' The model demonstrates strong performance in identifying turns, with high accuracy rates of 86.8% for 'left turn' and 86.67% for 'right turn.' Interestingly, the turning maneuvers have stronger performance than the 'go straight' action, with a notable 20.49% of 'go straight' samples being misclassified as 'right turn.'

13.3. Qualitative Results

For brevity, we present two samples per task, each with input frame(s), the task question, and the ground truth answer. Each sample also includes predictions from our fine-tuned baseline model (SigLIP-L/14 and Qwen1.5-0.5b) and the

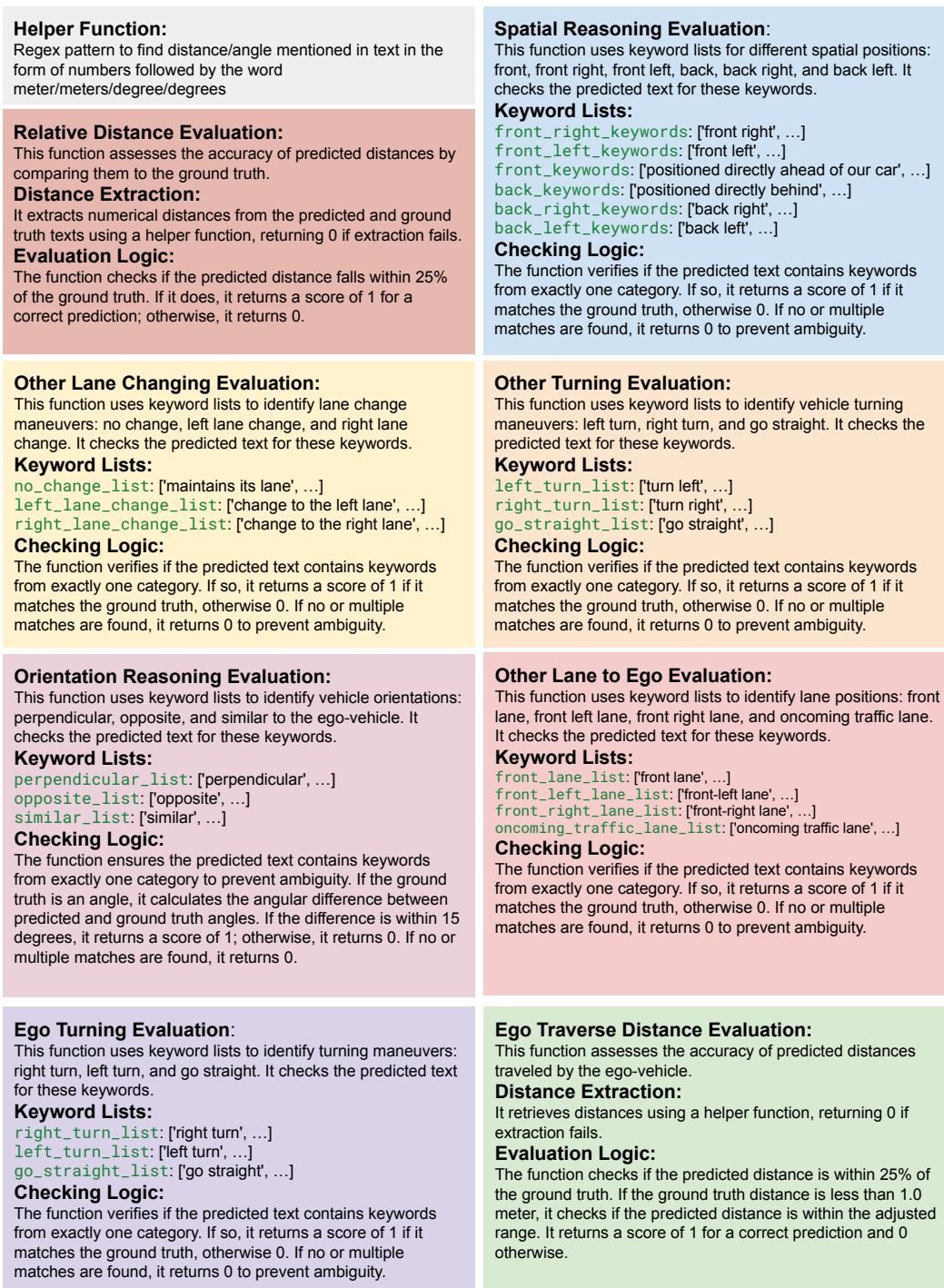


Figure 6. **Evaluation Metric Methodology for Each Task:** The method uses rule-based and regular expressions techniques to assess accuracy.

best performing zero-shot model, GPT-4o (GPT-4o-2024-08-06 version).

Figures for each task are as follows:

- Figure 14: Relative Distance (RD)

- Figure 15: Spatial Reasoning (SR)
- Figure 16: Orientation Reasoning (OR)
- Figure 17: Other Lane to Ego-Vehicle (EGO-LANE)
- Figure 18: Other Lane Changing (OBJ-LANE)

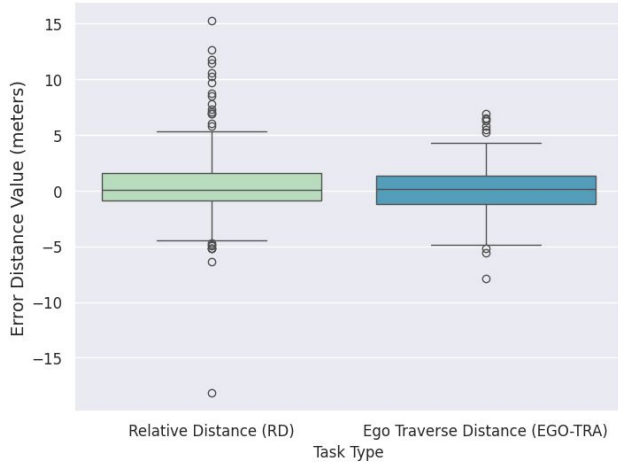


Figure 7. Distance error on Relative Distance (RD) and Ego Traverse Distance (EGO-TRA) tasks.

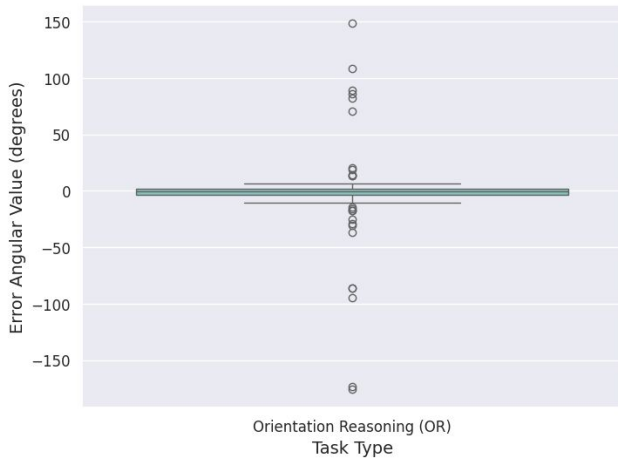


Figure 8. Angular error on Orientation Reasoning (OR) task.

- Figure 19: Other Turning (OBJ-TURN) task
- Figure 20: Ego Turning (EGO-TURN)
- Figure 21: Ego Traverse Distance (EGO-TRA)

13.4. Ablation Study Details

We provide detailed ablation results across eight tasks in Table 13.

Results indicate that stronger visual encoders significantly improve performance. For instance, comparing CLIP-L/14 to SigLIP-L/14 shows improvements of over 15.2% in Relative Distance (RD), 4.0% in Orientation Reasoning (OR), 5.6% in Other Turning (OBJ-TURN), and 10.4% in Ego Turning (EGO-TURN).

The optimal number of visual tokens is 16. Increasing this to 36 tokens improves Ego Traverse Distance (EGO-TRA) by only 2.8%, while performance in other tasks de-

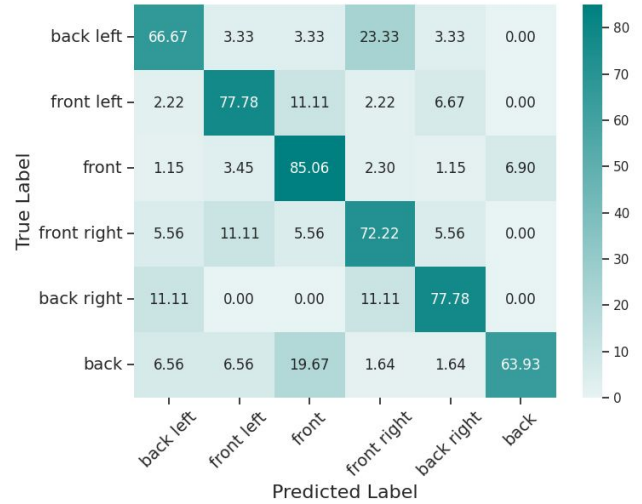


Figure 9. Confusion matrix on Spatial Reasoning (SR) task.

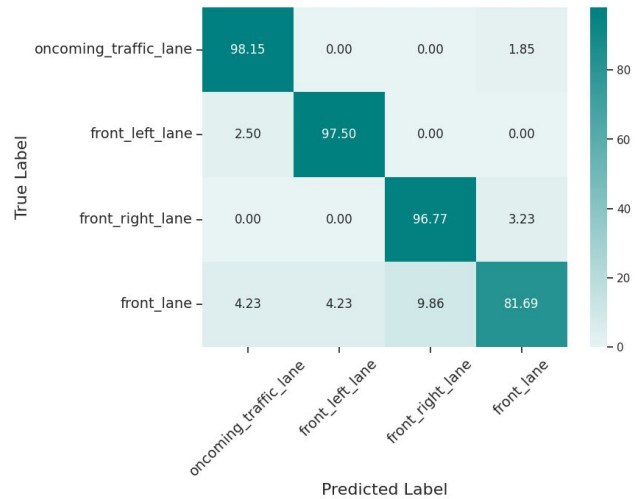


Figure 10. Confusion Matrix on Other Lane to Ego-Vehicle (EGO-LANE).

clines compared to the 16-token variant.

Utilizing more sequential frames generally enhances performance, especially in the tasks requiring temporal information (tasks 3-8). Single-frame tasks like Spatial Reasoning also benefit from training on multi-frame tasks, showing notable improvements. For ego-focused tasks, using 8 frames instead of 2 results in significant gains of over 14% in EGO-TURN and 12.8% in EGO-TRA, indicating that the number of frames is more critical for ego-focused tasks than for object-focused ones.

Table 13. **Ablation results per task.** All the models are finetuned on the TB-100k dataset, with their performance evaluated on TB-Bench and reported in accuracy (percentage).

Model	TrafficBehaviorBenchmark (TB-Bench)								
	RD ↑	SR ↑	OR ↑	EGO-LANE ↑	OBJ-LANE ↑	OBJ-TURN ↑	EGO-TURN ↑	EGO-TRA ↑	Avg ↑
Visual encoder									
CLIP-L/14	61.2	72.8	82.8	91.6	61.2	69.2	70.8	66.0	72.0
SigLIP-B/16	65.2	70.4	86.8	90.4	70.0	69.6	75.2	65.6	74.3
SigLIP-L/14	76.4	74.4	86.8	94.0	68.8	74.8	81.2	63.2	77.5
Visual tokens per frame									
4	68.8	70.0	86.4	94.0	67.6	74.0	71.6	49.2	72.7
16	76.4	74.4	86.8	94.0	68.8	74.8	81.2	63.2	77.5
36	75.5	70.8	84.4	91.2	64.8	71.2	77.6	66.0	76.2
Number of frames									
2	72.4	70.8	86.0	92.8	67.2	70.0	67.2	50.4	72.1
4	74.4	72.0	87.2	92.4	66.8	66.0	72.8	58.4	73.8
8	76.4	74.4	86.8	94.0	68.8	74.8	81.2	63.2	77.5

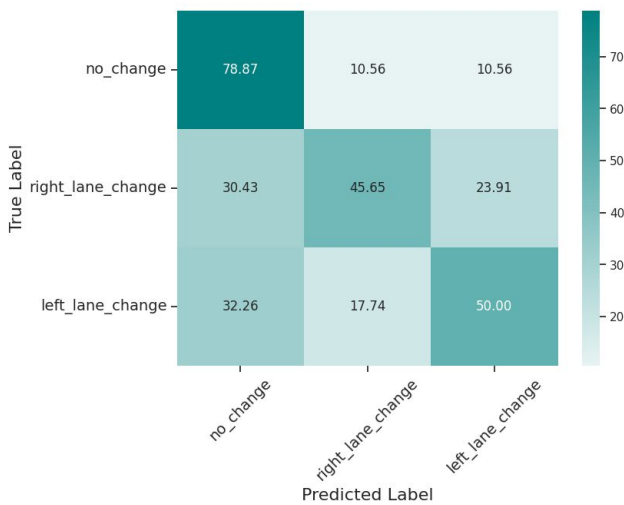


Figure 11. Confusion Matrix on Other Lane Changing (OBJ-LANE).

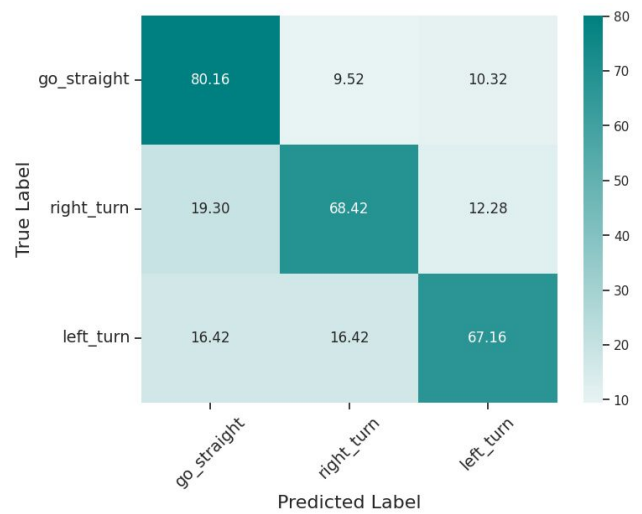


Figure 12. Confusion matrix on Other Turning (OBJ-TURN).

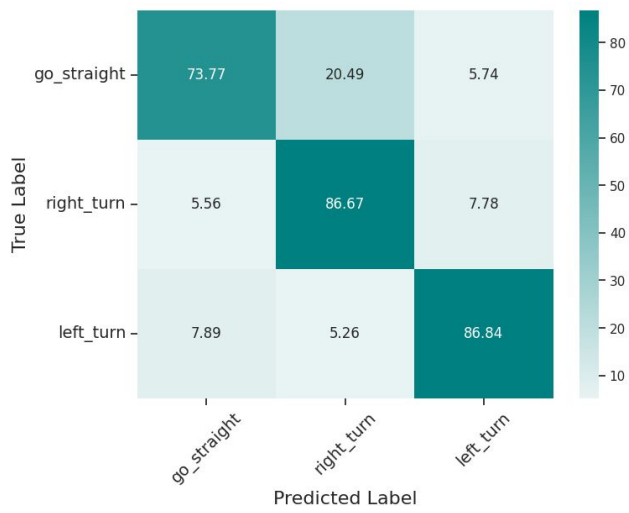


Figure 13. Confusion matrix on Ego Turning (EGO-TURN)

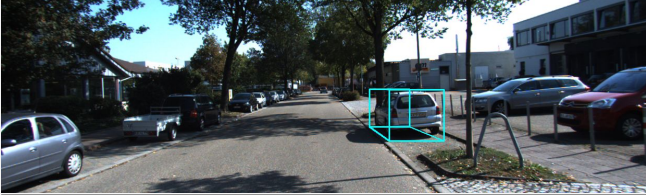
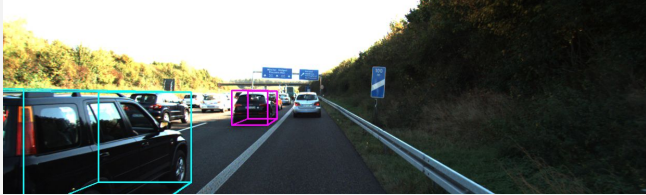
<p>Question: How far is Entity #1 from the self-car in meters?</p>	<p>Question: Can you measure the straight-line distance in meters between Entity #1 and Entity #2?</p>
	
<p>Annotation: Entity #1 is positioned at a distance of 12.23 meters from the self-car.</p>	<p>Annotation: Yes, the straight-line distance between Entity #1 and Entity #2 is precisely 13.41 meters.</p>
<p>Ours: Entity #1 is situated at a distance of 13.72 meters from the self-car. ✓</p>	<p>Ours: Yes, the straight-line distance between Entity #1 and Entity #2 is precisely 14.80 meters. ✓</p>
<p>GPT-4o: The distance from Entity #1 to the self-car is approximately 23.5 meters. ✗</p>	<p>GPT-4o: i'm unable to measure distances in images. ✗</p>

Figure 14. Examples and predictions from our baseline method and GPT-4o for the Relative Distance (RD) task.

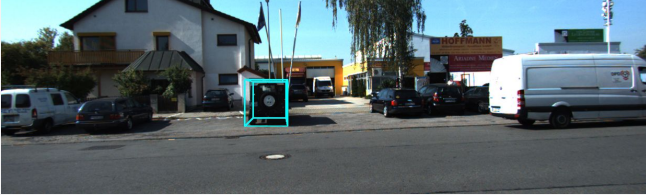
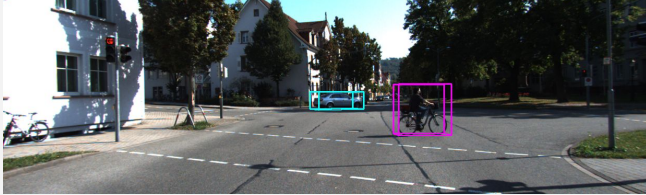
<p>Question: How are Entity #1 and the self-car spatially related, from the self-car perspective?</p>	<p>Question: How are Entity #1 and Entity #2 spatially related, from the Entity #2 perspective?</p>
	
<p>Annotation: Entity #1 is positioned directly ahead of the self-car, establishing a forward spatial relationship from the self-car's perspective.</p>	<p>Annotation: From the perspective of Entity #2, Entity #1 is positioned at the back left relative to its own orientation.</p>
<p>Ours: Entity #1 is positioned directly ahead of the self-car, establishing a forward spatial relationship from the self-car's perspective. ✓</p>	<p>Ours: From the perspective of Entity #2, Entity #1 is positioned at the back right side. ✗</p>
<p>GPT-4o: from the self-car perspective, Entity #1 is located at the front. the correct option is a. Front. ✓</p>	<p>GPT-4o: From the perspective of Entity #2 (the bicycle), Entity #1 (the car) is spatially related as option b: front right. ✗</p>

Figure 15. Examples and predictions from our baseline method and GPT-4o for the Spatial Reasoning (SR) task.

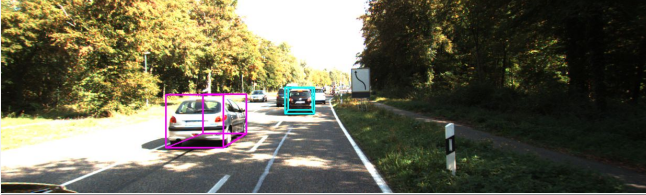
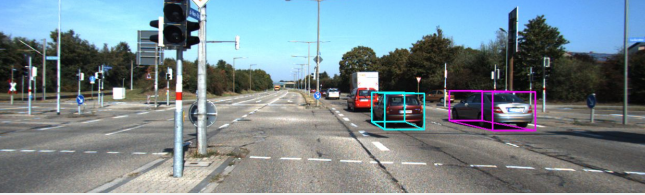
<p>Question: What is the angle between Entity #2 and Entity #1, in degrees?</p>	<p>Question: What is the orientation of Entity #2 relative to Entity #1, similar, opposite or perpendicular?</p>
	
<p>Annotation: The angle between Entity #2 and Entity #1 is approximately 1.72 degrees.</p>	<p>Annotation: Entity #2 is oriented in a similar direction or alignment to Entity #1.</p>
<p>Ours: The angle between Entity #2 and Entity #1 is precisely 0.00 degrees, indicating they are perfectly aligned. ✓</p>	<p>Ours: Entity #2 is oriented in a similar direction or alignment to Entity #1. ✓</p>
<p>GPT-4o: The angle between Entity #2 and Entity #1 is 115.5 degrees. ✗</p>	<p>GPT-4o: The orientation of Entity #2 relative to Entity #1 is a. similar. both vehicles are facing in the same direction along the road. ✓</p>

Figure 16. Examples and predictions from our baseline method and GPT-4o for the Orientation Reasoning (OR) task.

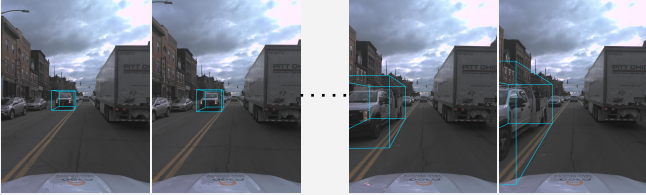

<p>Question: How would you describe the lane position of Entity #1?</p>	<p>Question: How would you describe the lane position of Entity #1?</p>
	
<p>Annotation: Entity #1 is positioned in the lane designated for oncoming traffic.</p>	<p>Annotation: Entity #1 is positioned in the front lane.</p>
<p>Ours: Entity #1 is positioned in the lane designated for oncoming traffic. ✓</p>	<p>Ours: Entity #1 is positioned in the front lane. ✓</p>
<p>GPT-4o: Entity #1 is in the oncoming traffic lane (option d). ✓</p>	<p>GPT-4o: The lane position of Entity #1 is in the **d. oncoming traffic lane**. ✗</p>

Figure 17. Examples and predictions from our baseline method and GPT-4o for the Other Lane to Ego-Vehicle (EGO-LANE) task.

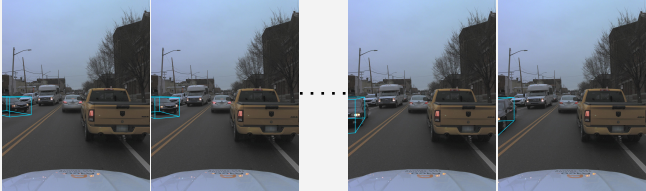

<p>Question: How would you describe the driving scene involving Entity #1? Please explain, focusing on the vehicle's lane change maneuver.</p>	<p>Question: How would you describe the driving scene involving Entity #1? Please explain, focusing on the vehicle's lane change maneuver.</p>
	
<p>Annotation: Entity #1 smoothly executed a left lane change.</p>	<p>Annotation: Entity #1 smoothly executed a right lane change.</p>
<p>Ours: Entity #1 executed a left lane change maneuver, smoothly transitioning to the adjacent lane. ✓</p>	<p>Ours: Entity #1 smoothly merges into the ego-vehicle's lane from the left to the right. ✓</p>
<p>GPT-4o: the vehicle identified as Entity #1 is performing a "left lane change". you can observe this by noting its movement from the right side of the lane towards the left, indicated by the vehicle's gradual shift in position across the series of images. ✓</p>	<p>GPT-4o: the driving scene involving Entity #1 shows no evidence of a lane change maneuver. the vehicle appears to maintain its position within the same lane across the series of images. therefore, the correct option is: a. no change ✗</p>

Figure 18. Examples and predictions from our baseline method and GPT-4o for the Other Lane Changing (OBJ-LANE) task.



<p>Question: How would you describe the driving scene involving Entity #1? Please explain, focusing on the vehicle's turning maneuver.</p>	<p>Question: How would you describe the driving scene involving Entity #1? Please explain, focusing on the vehicle's turning maneuver.</p>
	
<p>Annotation: Entity #1 executes a right-turn maneuver, smoothly navigating the vehicle in a clockwise direction.</p>	<p>Annotation: Entity #1 proceeds directly ahead without executing any turning maneuvers.</p>
<p>Ours: Entity #1 proceeds directly ahead without executing any turns. ✗</p>	<p>Ours: Entity #1 proceeds directly ahead without executing any turns. ✓</p>
<p>GPT-4o: the driving scene involving Entity #1 depicts the vehicle executing a maneuver characterized as b. turning left. ✗</p>	<p>GPT-4o: The driving scene involving Entity #1 is best described as option "b: turning left". the vehicle is positioned in a left-turn lane and appears to be angled towards a left turn at the intersection. ✗</p>

Figure 19. Examples and predictions from our baseline method and GPT-4o for the Other Turning (OBJ-TURN) task.



<p>Question: How would you describe the driving scene involving our car? Please explain, focusing on our car's turning maneuver.</p>	<p>Question: How would you describe the driving scene involving our car? Please explain, focusing on our car's turning maneuver.</p>
	
<p>Annotation: Our car executed a precise right-turn maneuver, smoothly navigating the corner with adept control.</p>	<p>Annotation: Our car executed a precise left-turn maneuver, showcasing excellent handling and driver control.</p>
<p>Ours: Our car executed a precise right turn, smoothly navigating the corner. ✓</p>	<p>Ours: Our car executed a precise right turn, smoothly navigating the corner. ✗</p>
<p>GPT-4o: Based on the sequence of images, the car is making a **right turn** (option c). the car approaches an intersection and moves towards the right-hand lane, making it suitable for turning right. ✓</p>	<p>GPT-4o: The driving scene involving your car shows it traveling straight along the road. there is no indication of a turning maneuver. therefore, the correct option is: a. go straight ✗</p>

Figure 20. Examples and predictions from our baseline method and GPT-4o for the Ego Turning (EGO-TURN) task.

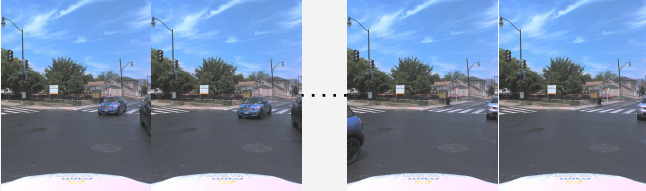

<p>Question: How far has our car driven and what kind of steering maneuver did it perform in the current scene?</p>	<p>Question: How far has our car driven and what kind of steering maneuver did it perform in the current scene?</p>
	
<p>Annotation: Our car has not driven any distance and is not performing any steering maneuver.</p>	<p>Annotation: Our car has driven 10.83 meters with a straight steering maneuver.</p>
<p>Ours: Our car has driven 1.01 meters with a straight steering maneuver. ✗</p>	<p>Ours: Our car has driven 12.42 meters with a straight steering maneuver. ✓</p>
<p>GPT-4o: The car has driven 6.3 meters, performing a left steering maneuver. ✗</p>	<p>GPT-4o: the car has driven 39.3 meters, and it performed a left lane change maneuver. ✗</p>

Figure 21. Examples and predictions from our baseline method and GPT-4o for the Ego Traverse Distance (EGO-TRA) task.