SURDS: Benchmarking Spatial Understanding and Reasoning in Driving Scenarios with Vision Language Models

Xianda Guo[1,*] , **Ruijun Zhang**[2,3,*] , **Yiqun Duan**[4,*] , **Yuhang He**[5], **Dujun Nie**[2,3],
Wenke Huang[1], **Chenming Zhang**[6,3], **Shuai Liu**[7], **Hao Zhao**[8], **Long Chen**[2,3,6,†]
[1] School of Computer Science, Wuhan University
[2] Institute of Automation, Chinese Academy of Sciences [3] Waytous
[4] University of Technology Sydney [5] Microsoft Research
[6] IAIR, Xi'an Jiaotong University [7] ByteDance [8] AIR, Tsinghua University
xianda_guo@163.com; {zhangruijun2023, long.chen}@ia.ac.cn; duanyiquncc@gmail.com

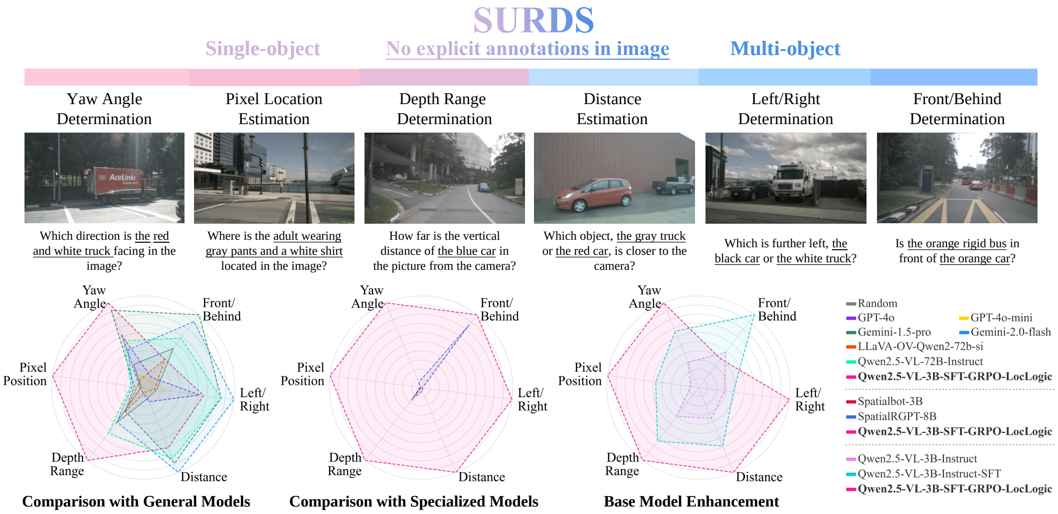


Figure 1: **Overview of the SURDS benchmark and the proposed method's performance.** The upper part illustrates the SURDS benchmark, which comprises six challenging tasks within driving scenarios, divided into single-object and multi-object categories. The lower part presents three radar charts evaluating the proposed method: the bottom left compares its performance against large-scale open-source and proprietary models; the bottom center highlights comparisons with existing spatial understanding methods; and the bottom right shows successive enhancements from the base model.

Abstract

Accurate spatial reasoning in outdoor environments—covering geometry, object pose, and inter-object relationships—is fundamental to downstream tasks such as mapping, motion forecasting, and high-level planning in autonomous driving. We introduce SURDS, a large-scale benchmark designed to systematically evaluate the spatial reasoning capabilities of vision language models (VLMs). Built on the NUSCENES dataset, SURDS comprises **41,080** vision–question–answer training instances and **9,250** evaluation samples, spanning six spatial categories:

---

[*] These authors contributed equally to this work.

[†] Corresponding author

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Track on Datasets and Benchmarks.

orientation, depth estimation, pixel-level localization, pairwise distance, lateral ordering, and front–behind relations. We benchmark leading general-purpose VLMs, including GPT, Gemini, and Qwen, revealing persistent limitations in fine-grained spatial understanding. To address these deficiencies, we go beyond static evaluation and explore whether alignment techniques can improve spatial reasoning performance. Specifically, we propose a reinforcement learning–based alignment scheme leveraging spatially grounded reward signals—capturing both perception-level accuracy (*location*) and reasoning consistency (*logic*). We further incorporate final-answer correctness and output-format rewards to guide fine-grained policy adaptation. Our GRPO-aligned variant achieves overall score of 40.80 in SURDS benchmark. Notably, it outperforms proprietary systems such as GPT-4o (13.30) and Gemini-2.0-flash (35.71). To the best of our knowledge, this is the first study to demonstrate that reinforcement learning–based alignment can significantly and consistently enhance the spatial reasoning capabilities of VLMs in real-world driving contexts. We release the SURDS benchmark, evaluation toolkit, and GRPO alignment code through: `https://github.com/XiandaGuo/Drive-MLLM`.

1 Introduction

Understanding complex spatial structures—such as object orientation, relative position, and geometric layout—within either discrete images or sequential videos [8] serves as a fundamental challenge in embodied perception and multi-modal scene understanding. Such spatial reasoning abilities are essential for a wide range of downstream tasks, including motion prediction [29, 86, 55, 32, 60], planning [65, 7, 49] and map construction [79, 19]. While we have witnessed huge progress in per-object centered recognition tasks with the assistance of various large-scale dataset [21, 30, 58], including detection [38, 39], tracking [42, 24], optical flow estimation [22, 57], depth estimation [37, 34–36] and semantic segmentation [16, 40], the inter-object spatial relation reasoning from RGB images in autonomous driving has been largely ignored even despite its vital importance in achieving fully holistic 3D scene understanding.

In the vision community, spatial relation reasoning within a single image has received increasing attention [68] by leveraging datasets such as Visual Genome [44]. However, existing works primarily focus on simple 2D positional relations (e.g., left/right, top/bottom), which fail to capture the richness and address the ambiguity of 3D spatial dependencies critical on real-world environments. Meanwhile, the recent emergence of large language models (LLMs) [61, 75] and their multimodal variants (VLMs) [1, 52, 15, 78] has opened promising avenues for high-level vision-language reasoning. Yet, the extent to which these models can perform on spatial understanding remains unclear.

Despite recent efforts on spatial question answering, such as BLINK [27], SpatialBot [11], SpatialRGPT [11], most prior studies focus on controlled indoor environments or rely on auxiliary modules such as depth estimators or object detectors. These designs limit generalization to dynamic and visually complex scenes. In contrast, spatial reasoning is required in outdoor scenarios, especially driving scenarios. At the same time, a growing number of works have begun to directly employ LLMs for planning or decision-making in autonomous driving [73, 80, 59, 23], yet such approaches often overlook a key prerequisite: without first establishing the spatial reasoning capability of these models, deploying them for real-world driving actions is inherently unreliable. This highlights the urgent need for a dedicated benchmark to systematically evaluate VLMs' spatial understanding in driving contexts.

In this work, we propose to systematically evaluate and improve the spatial reasoning capabilities of VLMs via a new large-scale benchmark: SURDS. Built on the nuScenes [10] dataset, SURDS comprises multi-view driving scenes captured from six surrounding cameras. From this, we respectively curate 41,080 training and 9,250 validation vision–question–answer (VQA) samples designed to probe fine-grained spatial understanding across six dimensions: orientation, depth, pixel-level position, pairwise distance, lateral ordering, and front–behind relations. Each query is paired with linguistically diverse, contextually grounded questions and evaluated using task-specific metrics. We compare the proposed SURDS benchmark with existing spatial understanding benchmarks in Table 1. SURDS is the first spatial understanding benchmark in driving scenarios.

Table 1: **Comparison between our work and other spatial understanding benchmarks.** Scale denotes the total number of QA pairs in the benchmark. Annotation Types indicate how spatial relations are labeled. Reasoning indicates whether the framework focuses on reasoning. Method denotes whether it proposes a specific approach to enhance spatial understanding. w/o Depth means the framework does not use depth information during training or evaluation. w/o Visual Mark indicates no visual annotations are added to the image.

Paper	Scale	Annotation Types	Data Source	Reasoning	Method	w/o Depth	w/o Visual Mark
BLINK (ECCV2024) [27]	3,807	Image QA pairs	Web	✗	✗	✓	w/ Marked point
SpatialRGPT (NeurIPS2024) [18]	1,406	Image QA pairs	Web	✓	✓	✗	w/ Mask
SpatialBot (ICRA2025) [11]	174	Image QA pairs	Web	✓	✓	✗	w/ Marked point & Bbox
VSI bench (CVPR2025) [81]	∼5,000	Video QA pairs	Indoor	✓	✗	✓	✓
SURDS (ours)	9,250	Image QA pairs	Driving	✓	✓	✓	✓

Using SURDS, we systematically evaluate the spatial reasoning capabilities of several frontier VLMs, including GPT-4o, GPT-4o-mini, Gemini, and Gemini 2.5 Pro. Our analysis reveals four converging lines of evidence that existing models still lack robust spatial grounding. Our evaluation reveals that current VLMs, including large-scale models, surprisingly struggle with spatial reasoning, showing poor absolute localization and brittle multi-object relational understanding abilities. Meanwhile, if the performance the spatial performance grows accordingly as the model scales up still remains unclear.To explore potential improvements, we first scale up synthetic spatial data via agent-centric scene construction (see Figure 3), which yields a baseline with an overall accuracy of 26.94 on our benchmark. Building on top of this, we propose a specially designed perception with reasoning process reward as shown in Figure 3 and together with the original final answer reward and format reward, we employ Group Relative Policy Optimization towards reasoning-level signals. Our GRPO-aligned model further boosts the performance to 40.80, not only reaching SoTA compared with the models at the same scale but also surpassing the most advanced large models such as GPT-4o (13.30), Gemini-2.0-flash (35.71), and Qwen2.5-VL-72B (33.47). Our experiments reveal the current spatial understanding ability for most existing SoTA works. Our benchmark exposes critical limitations in the spatial reasoning capabilities of large models within driving scenarios, and we further demonstrate that reinforcement learning–based alignment can substantially enhance these abilities. To facilitate future research, we release the comprehensive dataset, evaluation toolkit, and alignment pipeline that offer hands-on resources for advancing grounded spatial understanding in VLMs.

- We propose SURDS, the first large-scale benchmark for evaluating fine-grained spatial understanding of VLMs in realistic driving scenarios, respectively, contains 41,080 training pairs and 9,250 test pairs.
- Our evaluations on SURDS reveal fundamental spatial reasoning limitations in existing models and demonstrate that model scale alone does not ensure spatial competence.
- Comprehensive experiments with different training strategies ranging from supervised fine-tuning, reinforcement learning to post-train alignment, provide valuable insight for follow-up researchers.

2 Related Work

Large Vision Language Models (LVLM) Benefiting from the huge success in large language models (LLMs) [9, 61, 62] in recent years, a new research venue has been focusing on extending natural language-based large models (especially the GPT family LLM) to multimodal large language models (VLM) [1, 47, 48, 66, 74, 25]. Among all of them, encompassing vision into language has made dramatic progress and various vision language models (VLM) have been developed [5, 6, 47, 48, 51, 52] for various crossmodal tasks such as visual question answer (VQA) [3, 31] and crossmodal reasoning [84, 41, 28, 71], owing to the availability of various large image-text datasets [50, 43, 69, 12]. Typical VLM models include BLIP family [47, 48], LLaVA family [51, 52] and Qwen-VL family [4, 5, 76]. They either innovate in network architecture [17, 64, 47, 48] or adopt novel training strategy [5, 87]. For example, regarding the network architecture innovation, QWen-VL [5] and MiniGPT-4 [87] employ ViT [2] like network as visual encoder, LLaVA [64] instead employs CLIP ViT-L/14 [67] for visual encoding and InternVL [17] uses InternViT-6B for

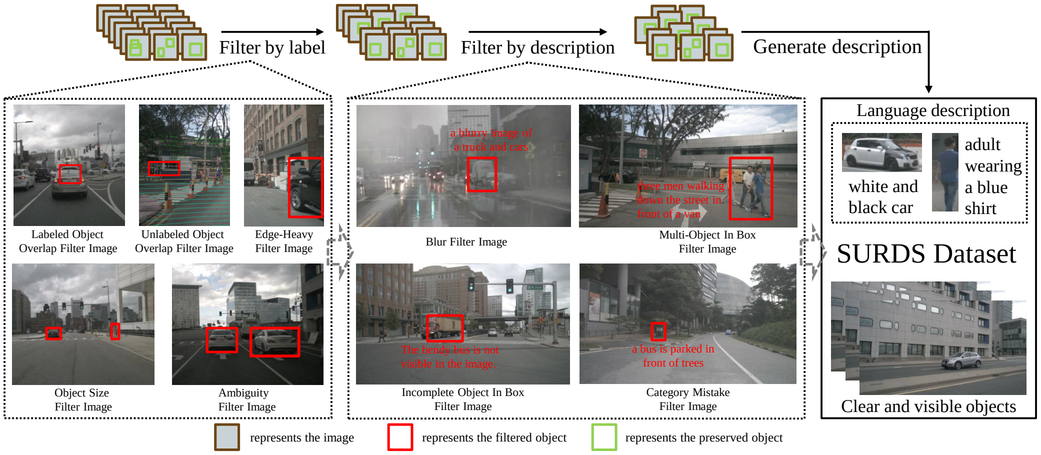

Filter by label
Filter by description
Generate description
Labeled Object Overlap Filter Image
Unlabeled Object Overlap Filter Image
Edge-Heavy Filter Image
Object Size Filter Image
Ambiguity Filter Image
a blurry image of a truck and cars
Blur Filter Image
three men walking down the street in front of a van
Multi-Object In Box Filter Image
The bendy bus is not visible in the image.
Incomplete Object In Box Filter Image
a bus is parked in front of trees
Category Mistake Filter Image
Language description
white and black car
adult wearing a blue shirt
SURDS Dataset
Clear and visible objects
represents the image
represents the filtered object
represents the preserved object

Figure 2: **Overview of the pipeline for constructing the SURDS dataset.** The visual elements shown in the image (e.g., bounding boxes and textual descriptions) are illustrative only and do not appear in the actual dataset. The system filters objects based on labels (left), applies additional filtering using text descriptions generated by a vision language model (center), and produces object descriptions (right), ensuring high-quality annotations for each image.

visual encoding. Regarding the training strategy, Qwen-VL [5] employs a three-stage strategy: first pre-train on massive image-text pairs, then multi-task pre-train over seven major tasks, and finally fine-tune with instruction on over 350,000 dialogues. MiniGPT-4 [87] adopts a two-stage training strategy by first pre-training on composite dataset including Conceptual Captions [13], LAION [70] and SBU [63] and then fine-tuning on high-quality image description dataset.

Visual-Language Benchmarks

In the pre-LLM era, most public vision-language datasets were single-task oriented, limiting their ability to holistically evaluate multimodal reasoning. Representative examples include image captioning [50], visual question answering [3, 31], and OCR [54]. With the emergence of LLMs, more comprehensive and multi-task datasets have been curated to better assess general-purpose multimodal reasoning. Among them, MME [26] focuses on *Yes/No* questions, visual perception, and language reasoning; MMBench [53] expands coverage across diverse domains with a circular evaluation design; Seed-Bench [46, 45] introduces multi-image and video inputs; and MM-Vet [82] aggregates multiple sub-tasks, including OCR, recognition, and math reasoning. Beyond recognition-centric benchmarks, recent efforts target broader cognitive abilities. MMMU [83] emphasizes domain knowledge reasoning, HallusionBench [33] investigates hallucinations and visual illusions, MathVista [56] focuses on math-based visual understanding, BLINK [27] probes holistic perception, and Mega-Bench [14] scales evaluation to 500+ real-world tasks.

3 SURDS Benchmark

Recent advancements have seen VLMs being directly employed for autonomous driving and embodied intelligence, which heavily depend on sophisticated spatial perception and reasoning. However, these works lack a detailed investigation into the spatial reasoning abilities of VLMs to demonstrate how reliable current models are on spatial information.

Data Source We construct our benchmark on the nuScenes [10] dataset, which is a large-scale public dataset specifically designed for autonomous driving research. It collects rich sensor data, including images from six cameras covering a full 360° field of view, along with LiDAR, radar, and GPS/IMU data. The dataset is captured in the urban environments of Boston and Singapore, featuring a diverse range of traffic conditions, weather scenarios, and times of day. This diversity ensures that the models are tested on a wide array of real-world driving situations, enhancing the robustness of the evaluation.

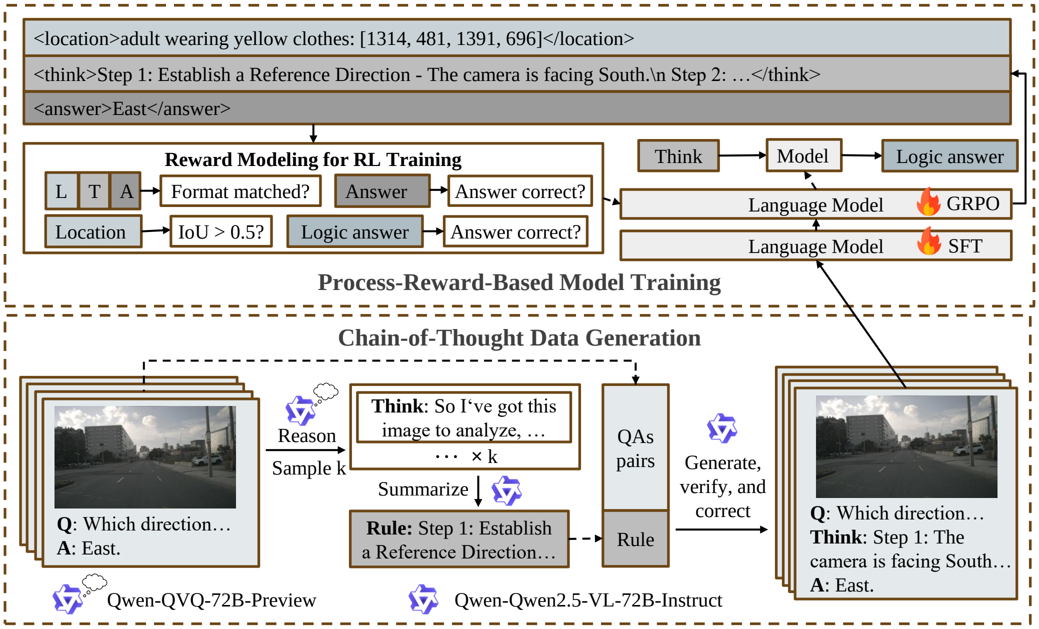

<location>adult wearing yellow clothes: [1314, 481, 1391, 696]</location>
<think>Step 1: Establish a Reference Direction - The camera is facing South.\n Step 2: …</think>
<answer>East</answer>
Reward Modeling for RL Training
L
T
A
Format matched?
Answer
Answer correct?
Location
IoU > 0.5?
Logic answer
Answer correct?
Think
Model
Logic answer
Language Model
GRPO
Language Model
SFT
Process-Reward-Based Model Training
Chain-of-Thought Data Generation
Q: Which direction…
A: East.
Reason
Sample k
Think: So I've got this image to analyze, …
… × k
Summarize
Rule: Step 1: Establish a Reference Direction…
QAs pairs
Rule
Generate, verify, and correct
Q: Which direction…
Think: Step 1: The camera is facing South…
A: East.
Qwen-QVQ-72B-Preview
Qwen-Qwen2.5-VL-72B-Instruct

Figure 3: **Overview of data generation and model training.** The lower part illustrates how COT-augmented QA data is generated: sampled QA pairs are processed by a vision reasoning model to infer solutions, followed by a vision-language model that extracts general reasoning steps and rules. These rules are used to generate COT-augmented QA pairs, which are then validated and corrected. The upper part shows the training pipeline: after SFT, the model is further optimized using process-reward-based training guided by four custom rewards.

Data Filtering To ensure that each bounding box corresponds to a clearly visible and unambiguous object, we adopt a multi-stage filtering pipeline comprising both label-based and description-based strategies, as shown in Figure 2. These filters collectively remove occluded, edge-aligned, ambiguous, and undersized objects:

- **Occlusion Removal.** We discard objects that are heavily occluded by other annotated instances when the ratio $r = \frac{|A \cap B|}{\min(|A|,|B|)}$, where A and B denote the areas of the two bounding boxes, exceeds 0.8. To further account for occlusion caused by unlabeled obstacles (e.g., fences), we project the *lidarseg* point clouds from nuScenes onto the image plane and remove any object whose bounding box contains too few LiDAR points of the correct semantic class, suggesting that the object is not meaningfully visible.
- **Edge and Size Filters.** We exclude objects whose center points lie outside image boundaries, as well as those with pixel areas below a minimal size threshold. To prevent identity ambiguity, we also remove images containing multiple objects of the same class (e.g., several pedestrians or vehicles).
- **Description-based Filtering.** Even after geometric filtering, some objects remain semantically ambiguous due to blur or annotation noise. We use *instruction-blip-13B* to generate object-level descriptions and discard any instance producing unclear or non-specific text. For retained samples, we standardize references to reduce linguistic bias—vehicles are described solely by color (e.g., "white and black car"), while pedestrians are identified by clothing (e.g., "adult wearing a blue shirt").

Starting from 28,130 training and 6,019 validation multi-view scenes (6 cameras per scene), we retain 27,152 and 5,919 images with clearly visible objects, respectively. These form the foundation for our QA benchmark, yielding 41,080 training and 9,250 validation vision–question–answer (VQA) instances.

Benchmark Construction To systematically evaluate VLMs' spatial reasoning capabilities in realistic driving contexts, we construct a large-scale QA benchmark consisting of 41,080 training and 9,250 validation vision–question–answer (VQA) instances. These are derived from filtered images captured by six cameras in the nuScenes dataset. We design six spatial tasks across two categories. The single-object subset focuses on basic spatial comprehension—yaw angle classification, pixel-level localization, and depth range estimation—each probing a distinct axis of object-centric reasoning. The multi-object subset introduces relational reasoning, including pairwise distance comparison, left–right ordering, and front–back understanding, requiring the model to analyze spatial relationships across multiple objects. The prompt templates formulated for various VQA tasks are provided in Appendix B.

4 Post-Train Alignment for Boosting Spatial Understanding

4.1 Data Generation by Test-Time CoT Scaling

RL training primarily enhances a VLM's performance based on existing knowledge [20], while SFT functions serves as a process of knowledge injection. We argue that models with smaller parameter sizes lack inherent reasoning capabilities and thus require SFT to introduce reasoning knowledge. Since the constructed dataset does not contain chain-of-thought (CoT) annotations, it is necessary to generate reasoning traces for each QA pair.

To provide more insights for the community, we employ open-source models to generate CoTs. As shown at the bottom side of Figure 3, we first use the visual reasoning model QVQ to reflect on and reason through a sampled set of k QA pairs. These thought processes are then summarized and distilled into reasoning rules consisting of generalizable solution steps using Qwen2.5-VL-72B. By feeding these rules alongside the original QA pairs into Qwen2.5-VL-72B, we generate new QA pairs annotated with CoTs. These outputs are subsequently validated and corrected by the model itself, resulting in an automated pipeline for constructing high-quality QA datasets with COT annotations. The prompts used to generate CoT reasoning for the data are provided in Appendix D. This method is motivated by two empirical observations: 1) Directly generating CoTs in batch using QVQ is computationally expensive and often results in verbose, unstructured, or format-inconsistent outputs; 2) Relying solely on Qwen2.5-VL-72B for CoT generation leads to degraded output quality and increased hallucinations.

4.2 Reinforcement Training with Reward Modeling

After obtaining the CoT-augmented data, we use it to train the model to enhance its spatial reasoning capabilities. The training pipeline is illustrated at the top side of Figure 3. We begin with **SFT as the cold start** of the full model, including the training of the visual encoder, multimodal projector, and language model, using the generated long CoT data.

Then, we apply reinforcement learning using GRPO [72] to enhance the model's capacity for spatial reasoning. In each training instance, the model is prompted to sequentially generate three components: the bounding box of the queried object, a step-by-step reasoning trace, and the final answer. Given that spatial reasoning is inherently object-centric, we assign a localization reward of 1 if the predicted bounding box achieves an IoU greater than 0.5 with the ground-truth region. To encourage output fidelity, a format reward of 1 is given if the model adheres to the prescribed output structure. Additionally, an accuracy reward of 1 is granted when the final answer is correct.

To promote logical coherence in reasoning, we introduce a logic reward inspired by Embodied-R [85], which assesses whether the reasoning trace leads to a correct answer. While the original approach uses a frozen reference model to evaluate consistency by feeding in both the reasoning trace and the question, we identify two limitations: (1) including the question can introduce answer leakage, as the trace often implicitly encodes the answer; and (2) relying on a static external model entails a trade-off between inference cost and reliability. To address both issues, we feed only the reasoning trace into the model under training, which serves as its own verifier. This design is efficient, incurs no additional computational overhead, and dynamically adapts to the model's evolving capabilities. A scalar logic reward of 1 is assigned if the inferred answer from the reasoning trace matches the originally generated final answer; otherwise, the reward is 0. This combination of localization, format, accuracy, and logic rewards ensures that the model not only produces structurally valid outputs, but also grounds its reasoning process in spatial consistency.

Table 2: **Comparison of our proposed method with other open-source and proprietary VLMs, as well as specialized spatial understanding models.** Yaw, Pixel, Depth, Dis, L/R, and F/B correspond to the six spatial reasoning tasks illustrated in Figure 1. The Score column represents the average performance across these six metrics. **Bold**: Best. Underline: Second Best.

Model	Single-object			Multi-object			Score
	Yaw	Pixel	Depth	Dis	L/R	F/B	
Random	5.73	1.12	34.27	8.76	11.57	11.89	12.22
GPT-4o	13.08	1.62	2.49	11.57	47.89	3.14	13.30
GPT-4o-mini	3.24	0.28	0.22	4.22	21.51	2.05	5.25
Gemini-1.5-pro	19.14	4.41	22.70	61.95	66.38	**22.05**	32.77
Gemini-2.0-flash	9.30	5.41	32.97	**69.30**	**77.30**	20.00	35.71
LLaVA-OV-Qwen2-72b-si	1.95	3.03	23.57	3.78	9.73	8.65	8.45
Qwen2.5-VL-72B-Instruct	11.57	6.13	44.00	58.05	66.16	14.92	33.47
Qwen2.5-VL-7B-Instruct	7.57	3.46	25.95	11.46	17.95	9.30	12.61
Qwen2.5-VL-3B-Instruct	6.27	3.81	27.68	17.84	14.81	10.49	13.48
SpatialBot [11]	0.00	0.00	12.00	0.00	0.00	0.00	2.00
SpatialRGPT [18]	1.30	0.55	10.59	1.95	0.86	7.35	3.77
Qwen2.5-VL-3B-SFT-GRPO-LocLogic	**20.97**	**44.81**	**69.84**	49.30	51.35	8.54	**40.80**

5 Experiments

5.1 Experimental Setup

Implementation Details We assess a variety of models, including state-of-the-art open-source and proprietary models, as well as models specifically designed for spatial understanding. A random baseline is also included for comparison, and the evaluated models are summarized in Table 2. All models are prompted with standardized instructions and are required to generate outputs strictly adhering to a predefined format. We employ the `sglang` framework* to accelerate inference and reduce evaluation time. All training and evaluation are conducted on eight NVIDIA A800 GPUs. For supervised fine-tuning, models are trained for 2 epochs with a learning rate of 1×10^{-6} and a warm-up ratio of 10%. GRPO training is performed for 1 epoch using a maximum prompt length of 4096 tokens, an output length of up to 512 tokens, and generating 4 samples per prompt. The structured response format is provided in Appendix C.

Evaluation Metrics

To quantitatively evaluate model performance on the SURDS benchmark, we define a set of evaluation metrics. For the *Pixel Localization Estimation* task, we adopt a centerness-based [77] metric. For other tasks, a prediction receives a score of 1 if it matches the ground-truth answer, and 0 otherwise. Given N QA pairs, the metric score for each task is computed as the average over all N pairs. The final overall score is the average of all individual task scores.

To avoid misleading results from scalar regression metrics, we adopt a range-based categorical evaluation rather than predicting precise numeric values. This design better reflects human-like spatial reasoning: for example, a person can usually tell whether a car is facing north, south, east, or west, but cannot quote its exact yaw angle from a single image. In addition, this categorical scheme prevents models from exploiting trivial solutions such as always predicting an average scalar value. Overall, the dual-query, range-based evaluation provides a more robust and honest assessment of fine-grained spatial reasoning ability.

5.2 Main Results

The evaluation results of different models are presented in Table 2. Among proprietary models, Gemini performed the best, achieving top results across several multi-object metrics. Among open-source VLMs, Qwen also showed strong performance, ranking second in many metrics. The specialized spatial understanding model performed poorly across all tasks, while the SpatialBot model could not be evaluated on several metrics due to its lack of instruction-following capability. Our proposed method achieved first place in multiple single-object metrics, with a significant margin over the second-best model—for instance, a nearly 60% improvement on the depth metric. It also

---

*`https://github.com/sgl-project/sglang`

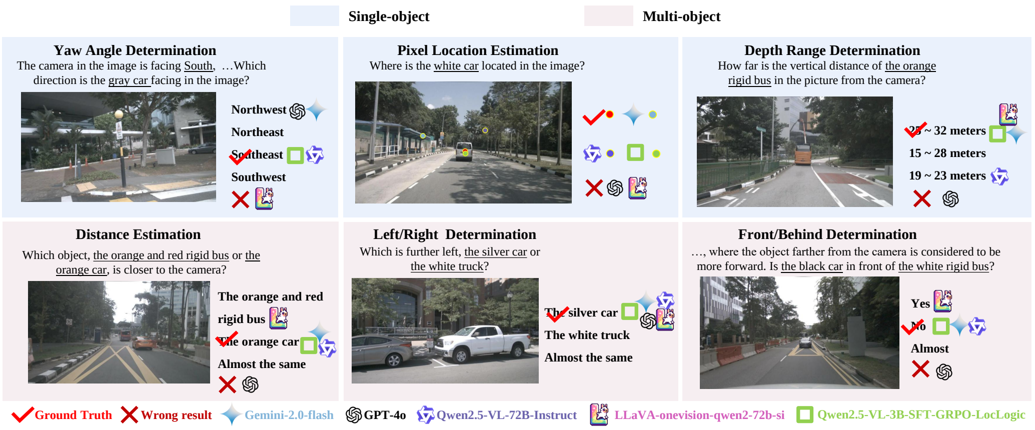

Single-object
Multi-object
Yaw Angle Determination
The camera in the image is facing South, …Which direction is the gray car facing in the image?
Northwest
Northeast
Southeast
Southwest
Pixel Location Estimation
Where is the white car located in the image?
Depth Range Determination
How far is the vertical distance of the orange rigid bus in the picture from the camera?
25 ~ 32 meters
15 ~ 28 meters
19 ~ 23 meters
Distance Estimation
Which object, the orange and red rigid bus or the orange car, is closer to the camera?
The orange and red rigid bus
The orange car
Almost the same
Left/Right Determination
Which is further left, the silver car or the white truck?
The silver car
The white truck
Almost the same
Front/Behind Determination
…, where the object farther from the camera is considered to be more forward. Is the black car in front of the white rigid bus?
Yes
No
Almost
Ground Truth
Wrong result
Gemini-2.0-flash
GPT-4o
Qwen2.5-VL-72B-Instruct
LLaVA-onevision-qwen2-72b-si
Qwen2.5-VL-3B-SFT-GRPO-LocLogic

Figure 4: **Illustrative examples of the benchmark QA pairs on both single-object and multi-object.**

achieves the highest overall score, outperforming the second-best approach by 14.25%. We visualize example responses from different models across various tasks, as shown in Figure 4. It can be seen that our proposed model performs well across multiple tasks. Furthermore, we visualize the complete outputs of our proposed model across various tasks, demonstrating its strong reasoning capabilities, as illustrated in Appendix A.

5.3 Discussion on Benchmarking Performance SOTA models

Here, we discuss the quantitative results of our benchmark with respect to four key aspects: single-object evaluation, multi-object evaluation, the relationship between model size and performance, and the impact of fine-tuning. Our benchmark offers three converging lines of evidence that current open-source VLMs still lack robust spatial grounding. **1) single-object probes** (yaw, pixel coordinates, depth bins) reveal that most models—including several 70 B-parameter variants—perform at or below random chance on orientation and depth, and seldom exceed 10% accuracy in sub-pixel localization, underscoring a persistent inability to encode absolute pose or metric information. **2) multi-object** tests show a modest uptick in accuracy for simpler comparative questions (left–right ordering, pairwise distance), yet performance collapses when the task demands non-canonical reasoning such as identifying the object "in front" under a forward-facing reference frame, indicating that relative spatial heuristics remain brittle. **3) scaling analysis** demonstrates that parameter count is not a reliable predictor of spatial competence: larger models sometimes trail lighter counterparts, implying that mere capacity expansion without explicit geometric priors does little to close the reasoning gap.

5.4 Ablation Study

Ablation of reward. We conducted ablation studies on our proposed method. The first set of experiments focused on the composition of the reward. Since the basic GRPO framework inherently includes the format reward and accuracy reward, we do not ablate these two. Instead, we examine the effects of adding or removing the location reward and logic reward. As shown in Table 3, incorporating either the location or logic reward individually led to only marginal improvements in overall performance. However, when both rewards were applied together, the model experienced a significant performance boost. These results suggest that without the supervision provided by the location reward, the model's object localization ability degrades—this ability forms the foundation of spatial reasoning. Building on this, the inclusion of logic supervision further enhances the model's consistency and spatial reasoning capability.

The second set of experiments focuses on the value settings of the rewards. We compared the conventional reward setting of 0, 1 with an alternative setting of -1, 1. The results show that using 0, 1 yields better performance. We attribute this to the sparsity of rewards during training—if the model receives a penalty every time it fails to reason correctly, it would accumulate mostly negative

Table 3: **Ablation study on model performance under different reward settings after SFT.** The four types of rewards correspond to the representations illustrated in Figure 3. †means using -1 and 1 as binary rewards instead of 0 and 1.

Base model	Training reward				Single-object			Multi-object			Score
	Format	Loc	Acc	Logic	Yaw	Pixel	Depth	Dis	L/R	F/B	
Qwen2.5-VL-3B-SFT	✗	✗	✗	✗	13.95	21.11	51.35	33.95	19.68	21.62	26.94
Qwen2.5-VL-3B-SFT	✓	✗	✓	✗	19.24	15.02	62.59	39.14	32.65	9.30	29.66
Qwen2.5-VL-3B-SFT	✓	✓	✓	✗	17.84	22.72	64.65	41.41	30.92	11.68	31.53
Qwen2.5-VL-3B-SFT	✓	✗	✓	✓	20.54	14.81	62.49	36.54	32.65	11.35	29.73
Qwen2.5-VL-3B-SFT	✓	✓	✓	✓	20.97	44.81	69.84	49.30	51.35	8.54	**40.80**
Qwen2.5-VL-3B-SFT†	✓	✓	✓	✓	18.16	22.61	62.05	39.14	32.22	15.14	31.55
Qwen2.5-VL-3B-SFT	✓	✓	✓	✓	20.97	44.81	69.84	49.30	51.35	8.54	**40.80**

Table 4: **Ablation study on different training setups of our method, analyzing the impact of SFT, GRPO, and the addition of location and logic rewards (LocLogic).** SFT denotes supervised fine-tuning, GRPO stands for Group Relative Policy Optimization.

Model	Single-object			Multi-object			Score
	Yaw	Pixel	Depth	Dis	L/R	F/B	
Qwen2.5-VL-3B	6.27	3.81	27.68	17.84	14.81	10.49	13.48
Qwen2.5-VL-3B-GRPO	14.59	3.75	29.19	35.68	39.89	22.70	24.30
Qwen2.5-VL-3B-GRPO-LocLogic	8.11	57.82	27.24	22.05	19.35	12.43	24.50
Qwen2.5-VL-3B-SFT	13.95	21.11	51.35	33.95	19.68	21.62	26.94
Qwen2.5-VL-3B-SFT-GRPO	19.24	15.02	62.59	39.14	32.65	9.30	29.66
Qwen2.5-VL-3B-SFT-GRPO-LocLogic	20.97	44.81	69.84	49.30	51.35	8.54	**40.80**

rewards. This causes gradients to remain negative or close to zero over extended periods, thereby hindering learning progress.

Ablation of training We also conducted ablation studies on the training of the base model. Specifically, we compared the standard GRPO training setup (using only format reward and accuracy reward) with our proposed reward design that incorporates format reward, location reward, accuracy reward, and logic reward. Similarly, we applied the same training configurations to the SFT-pretrained model. The results are presented in Table 4. Across both the non-SFT and SFT models, GRPO training improves overall performance. However, for the non-SFT model, adding the location and logic rewards did not lead to further improvements. In contrast, for the SFT model, incorporating location and logic rewards resulted in a significant performance boost. We attribute this difference to the weaker localization ability of the non-SFT model. Due to this limitation, the location reward remains sparse and provides limited training benefit, leading to negligible performance gains.

6 Conclusion and Limitation Discussion

We introduce SURDS, a large-scale benchmark comprising 41,080 VQA training instances and 9,250 evaluation samples that span six spatial reasoning categories—orientation, depth estimation, pixel-level localization, pairwise distance, lateral ordering, and front–behind relations. Benchmarking state-of-the-art general-purpose VLMs on SURDS exposes persistent shortcomings in fine-grained spatial understanding. To mitigate these deficiencies, we propose a reinforcement-learning framework that integrates spatially grounded reward signals with a reasoning-consistency objective. Extensive comparative and ablation experiments demonstrate that our approach yields substantial performance gains over existing VLMs while empirically validating the efficacy of both the training method and the reward design.

Nonetheless, the method remains untested on larger-scale model variants, and the benefits of linear reward scaling and multi-stage GRPO schedules have yet to be clarified. In addition, our QA construction primarily targets perception-level spatial reasoning. While these tasks naturally underpin higher-level capabilities such as trajectory prediction and planning, SURDS does not yet include such downstream QA pairs, which we leave for future work. Overall, we believe SURDS and its accompanying alignment strategy provide a solid foundation for advancing spatial reasoning in VLMs and open promising directions toward safer and more capable autonomous driving systems.

Acknowledgements. This work was supported by the Joint Funds of the National Natural Science Foundation of China under U24B20162 and the National Natural Science Foundation of China under Grant (62373356, 623B2080).

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *NeurIPS*, 35:23716–23736, 2022.

[2] Dosovitskiy Alexey. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv: 2010.11929*, 2020.

[3] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *ICCV*, pages 2425–2433, 2015.

[4] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

[5] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2309.12966*, 2023.

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

[7] Mayank Bansal, Alex Krizhevsky, and Abhijit Ogale. Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst. *arXiv preprint arXiv:1812.03079*, 2018.

[8] Istvan Barabas, Adrian Todoruţ, N Cordoş, and Andreia Molea. Current challenges in autonomous driving. In *IOP conference series: materials science and engineering*, 2017.

[9] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *NeurIPS*, 33:1877–1901, 2020.

[10] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. *arXiv preprint arXiv:1903.11027*, 2019.

[11] Wenxiao Cai, Yaroslav Ponomarenko, Jianhao Yuan, Xiaoqi Li, Wankou Yang, Hao Dong, and Bo Zhao. Spatialbot: Precise spatial understanding with vision language models. In *ICRA*, 2025.

[12] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12M: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *CVPR*, 2021.

[13] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *CVPR*, pages 3558–3568, 2021.

[14] Jiacheng Chen, Tianhao Liang, Sherman Siu, Zhengqing Wang, Kai Wang, Yubo Wang, Yuansheng Ni, Wang Zhu, Ziyan Jiang, Bohan Lyu, Dongfu Jiang, Xuan He, Yuan Liu, Hexiang Hu, Xiang Yue, and Wenhu Chen. Mega-bench: Scaling multimodal evaluation to over 500 real-world tasks. *arXiv preprint arXiv:2410.10563*, 2024.

[15] Lin Chen, Jisong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. Sharegpt4v: Improving large multi-modal models with better captions. *arXiv preprint arXiv:2311.12793*, 2023.

[16] Yuhua Chen, Wen Li, and Luc Van Gool. ROAD: Reality Oriented Adaptation for Semantic Segmentation of Urban Scenes. *CVPR*, 2018.

[17] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *CVPR*, pages 24185–24198, 2024.

[18] An-Chieh Cheng, Hongxu Yin, Yang Fu, Qiushan Guo, Ruihan Yang, Jan Kautz, Xiaolong Wang, and Sifei Liu. Spatialrgpt: Grounded spatial reasoning in vision-language models. In *NeurIPS*, 2024.

[19] Qing Cheng, Niclas Zeller, and Daniel Cremers. Vision-based large-scale 3d semantic mapping for autonomous driving applications. In *ICRA*, pages 9235–9242, 2022.

[20] Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V Le, Sergey Levine, and Yi Ma. Sft memorizes, rl generalizes: A comparative study of foundation model post-training. *arXiv preprint arXiv:2501.17161*, 2025.

[21] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The Cityscapes Dataset for Semantic Urban Scene Understanding. In *CVPR*, 2016.

[22] Qiaole Dong and Yanwei Fu. MemFlow: Optical Flow Estimation and Prediction with Memory. In *CVPR*, 2024.

[23] Yiqun Duan, Zhuoli Zhuang, Jinzhao Zhou, Yu-Cheng Chang, Yu-Kai Wang, and Chin-Teng Lin. Enhancing end-to-end autonomous driving systems through synchronized human behavior data. In *Proceedings of the 1st International Workshop on Brain-Computer Interfaces (BCI) for Multimedia Understanding*, 2024.

[24] Andreas Ess, Konrad Schindler, Bastian Leibe, and Luc Van Gool. Object Detection and Tracking for Autonomous Navigation in Dynamic Environments. *IJRR*, 2010.

[25] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva: Exploring the limits of masked visual representation learning at scale. In *CVPR*, pages 19358–19369, 2023.

[26] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023.

[27] Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A Smith, Wei-Chiu Ma, and Ranjay Krishna. Blink: Multimodal large language models can see but not perceive. *ECCV*, 2024.

[28] Xingyu Fu, Ben Zhou, Ishaan Chandratreya, Carl Vondrick, and Dan Roth. There's a time and place for reasoning beyond the image. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1138–1149, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[29] Jiyang Gao, Chen Sun, Hang Zhao, Yi Shen, Dragomir Anguelov, Congcong Li, and Cordelia Schmid. Vectornet: Encoding hd maps and agent dynamics from vectorized representation. In *CVPR*, June 2020.

[30] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013.

[31] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering. In *CVPR*, 2017.

[32] Junru Gu, Chen Sun, and Hang Zhao. Densetnt: End-to-end trajectory prediction from dense goal sets. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15303–15312, 2021.

[33] Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, Dinesh Manocha, and Tianyi Zhou. Hallusionbench: An advanced diagnostic suite for entangled language hallucination & visual illusion in large vision-language models, 2023.

[34] Xianda Guo, Juntao Lu, Chenming Zhang, Yiqi Wang, Yiqun Duan, Tian Yang, Zheng Zhu, and Long Chen. Openstereo: A comprehensive benchmark for stereo matching and strong baseline. *arXiv preprint arXiv:2312.00343*, 2023.

[35] Xianda Guo, Wenjie Yuan, Yunpeng Zhang, Tian Yang, Chenming Zhang, Zheng Zhu, Qin Zou, and Long Chen. Adjacent-view transformers for supervised surround-view depth estimation. In *IROS*, 2025.

[36] Xianda Guo, Chenming Zhang, Youmin Zhang, Ruilin Wang, Dujun Nie, Wenzhao Zheng, Matteo Poggi, Hao Zhao, Mang Ye, Qin Zou, and Long Chen. Stereo anything: Unifying stereo matching with large-scale mixed data. *arXiv preprint arXiv:2411.14053*, 2024.

[37] Xianda Guo, Chenming Zhang, Youmin Zhang, Wenzhao Zheng, Dujun Nie, Matteo Poggi, and Long Chen. Lightstereo: Channel boost is all you need for efficient 2d cost aggregation. In *ICRA*, 2025.

[38] Yuhang He, Shi Chen, Yifeng Pan, and Kai Ni. Using Edit Distance and Junction Feature to Detect and Recognize Arrow Road Marking. In *ITSC*, 2014.

[39] Yuhang He, Ziyu Pan, Lingxi Li, Yunxiao Shan, Dongpu Cao, and Long Chen. Real-Time Vehicle Detection from Short-range Aerial Image with Compressed MobileNet. In *ICRA*, 2019.

[40] Congrui Hetang, Haoru Xue, Cindy Le, Tianwei Yue, Wenping Wang, and Yihui He. Segment Anything Model for Road Network Graph Extraction. In *CVPRW*, 2024.

[41] Yushi* Hu, Hang* Hua, Zhengyuan Yang, Weijia Shi, Noah A Smith, and Jiebo Luo. Promptcap: Prompt-guided task-aware image captioning. *arXiv preprint arXiv:2211.09699*, 2022.

[42] Wei-Chih Hung, Henrik Kretzschmar, Tsung-Yi Lin, Yuning Chai, Ruichi Yu, Ming-Hsuan Yang, and Dragomir Anguelov. Soda: Multi-object tracking with soft data association. *arXiv 2008.07725*, 2020.

[43] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123:32–73, 2017.

[44] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Fei-Fei Li. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. In *IJCV*, 2017.

[45] Bohao Li, Yuying Ge, Yixiao Ge, Guangzhi Wang, Rui Wang, Ruimao Zhang, and Ying Shan. Seed-bench-2: Benchmarking multimodal large language models. *arXiv preprint arXiv:2311.17092*, 2023.

[46] Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. Seed-bench: Benchmarking multimodal llms with generative comprehension. *arXiv preprint arXiv:2307.16125*, 2023.

[47] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023.

[48] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models, 2023.

[49] Zenan Li, Fan Nie, Qiao Sun, Fang Da, and Hang Zhao. Uncertainty-aware decision transformer for stochastic driving environments. *arXiv preprint arXiv:2309.16397*, 2023.

[50] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755, 2014.

[51] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning, 2023.

[52] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, January 2024.

[53] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. Mmbench: Is your multi-modal model an all-around player?, 2023.

[54] Yuliang Liu, Zhang Li, Biao Yang, Chunyuan Li, Xucheng Yin, Cheng lin Liu, Lianwen Jin, and Xiang Bai. On the hidden mystery of ocr in large multimodal models, 2024.

[55] Yicheng Liu, Jinghuai Zhang, Liangji Fang, Qinhong Jiang, and Bolei Zhou. Multimodal motion prediction with stacked transformers. In *CVPR*, pages 7577–7586, 2021.

[56] Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. *arXiv preprint arXiv:2310.02255*, 2023.

[57] Ao Luo, Xin Li, Fan Yang, Jiangyu Liu, Haoqiang Fan, and Shuaicheng Liu. FlowDiffuser: Advancing Optical Flow Estimation with Diffusion Models. In *CVPR*, 2024.

[58] Will Maddern, Geoff Pascoe, Chris Linegar, and Paul Newman. 1 Year, 1000km: The Oxford RobotCar Dataset. *The International Journal of Robotics Research (IJRR)*, 2017.

[59] Jiageng Mao, Yuxi Qian, Junjie Ye, Hang Zhao, and Yue Wang. Gpt-driver: Learning to drive with gpt. *arXiv preprint arXiv:2310.01415*, 2023.

[60] Nigamaa Nayakanti, Rami Al-Rfou, Aurick Zhou, Kratarth Goel, Khaled S Refaat, and Benjamin Sapp. Wayformer: Motion forecasting via simple & efficient attention networks. In *ICRA*, 2023.

[61] OpenAI. Gpt-4 technical report, 2023.

[62] OpenAI. Hello gpt-4o, 2024. [Accessed: 2024-05-26].

[63] Vicente Ordonez, Girish Kulkarni, and Tamara L Berg. Im2text: Describing images using 1 million captioned photographs. In *NeurIPS*, 2011.

[64] Vicente Ordonez, Girish Kulkarni, and Tamara L Berg. Visual instruction tuning. In *NeurIPS*, 2011.

[65] Dean A Pomerleau. Alvinn: An autonomous land vehicle in a neural network. *NeurIPS*, 1, 1988.

[66] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763, 2021.

[67] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763, 2021.

[68] Mohd Nizam Saad, Zurina Muda, Noraidah Sahari Ashaari, Hamzaini Abdul Hamid, and Nur Hasanah binti Abu Hasan. The Spatial Relation Features for Describing Objects Relationships within Image. In *ICEEI*, 2015.

[69] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021.

[70] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021.

[71] Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. A-okvqa: A benchmark for visual question answering using world knowledge. In *ECCV*, pages 146–162, 2022.

[72] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

[73] Chonghao Sima, Katrin Renz, Kashyap Chitta, Li Chen, Hanxue Zhang, Chengen Xie, Ping Luo, Andreas Geiger, and Hongyang Li. Drivelm: Driving with graph visual question answering. *arXiv preprint arXiv:2312.14150*, 2023.

[74] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023.

[75] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[76] Qwen Team. Introducing qwen1.5, February 2024.

[77] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *ICCV*, 2019.

[78] Weihan Wang, Qingsong Lv, Wenmeng Yu, Wenyi Hong, Ji Qi, Yan Wang, Junhui Ji, Zhuoyi Yang, Lei Zhao, Xixuan Song, Jiazheng Xu, Bin Xu, Juanzi Li, Yuxiao Dong, Ming Ding, and Jie Tang. Cogvlm: Visual expert for pretrained language models, 2023.

[79] Xuan Xiong, Yicheng Liu, Tianyuan Yuan, Yue Wang, Yilun Wang, and Zhao Hang. Neural map prior for autonomous driving. In *CVPR*, 2023.

[80] Zhenhua Xu, Yujia Zhang, Enze Xie, Zhen Zhao, Yong Guo, Kwan-Yee K Wong, Zhenguo Li, and Hengshuang Zhao. Drivegpt4: Interpretable end-to-end autonomous driving via large language model. *RAL*, 2024.

[81] Jihan Yang, Shusheng Yang, Anjali Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in Space: How Multimodal Large Language Models See, Remember and Recall Spaces. In *CVPR*, 2025.

[82] Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. Mm-vet: Evaluating large multimodal models for integrated capabilities. *arXiv preprint arXiv:2308.02490*, 2023.

[83] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. *arXiv preprint arXiv:2311.16502*, 2023.

[84] Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. From recognition to cognition: Visual commonsense reasoning. In *CVPR*, June 2019.

[85] Baining Zhao, Ziyou Wang, Jianjie Fang, Chen Gao, Fanhang Man, Jinqiang Cui, Xin Wang, Xinlei Chen, Yong Li, and Wenwu Zhu. Embodied-r: Collaborative framework for activating embodied spatial reasoning in foundation models via reinforcement learning. *arXiv preprint arXiv:2504.12680*, 2025.

[86] Hang Zhao, Jiyang Gao, Tian Lan, Chen Sun, Ben Sapp, Balakrishnan Varadarajan, Yue Shen, Yi Shen, Yuning Chai, Cordelia Schmid, Congcong Li, and Dragomir Anguelov. Tnt: Target-driven trajectory prediction. In *CoRL*, 2021.

[87] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims made in the abstract and introduction accurately reflect the paper's contributions and scope.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: In the conclusion and limitations section.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: The paper provides the full set of assumptions and a complete proof in the method section.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: In the implementation details section.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The code is available at `https://github.com/XiandaGuo/Drive-MLLM`.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: In the implementation details section.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: In the experiment section.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: In the implementation details section.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research is with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: There is no societal impact of the work performed.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Original owners of assets are all properly cited in the paper.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [Yes]

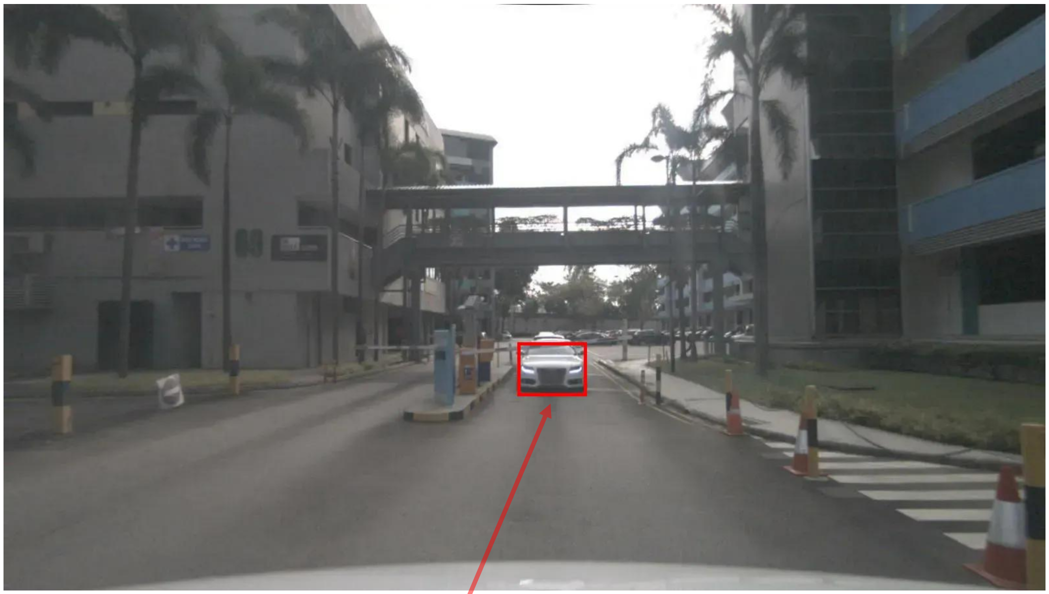
Justification: LLM is used for data processing/filtering.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

Supplementary Material on SURDS

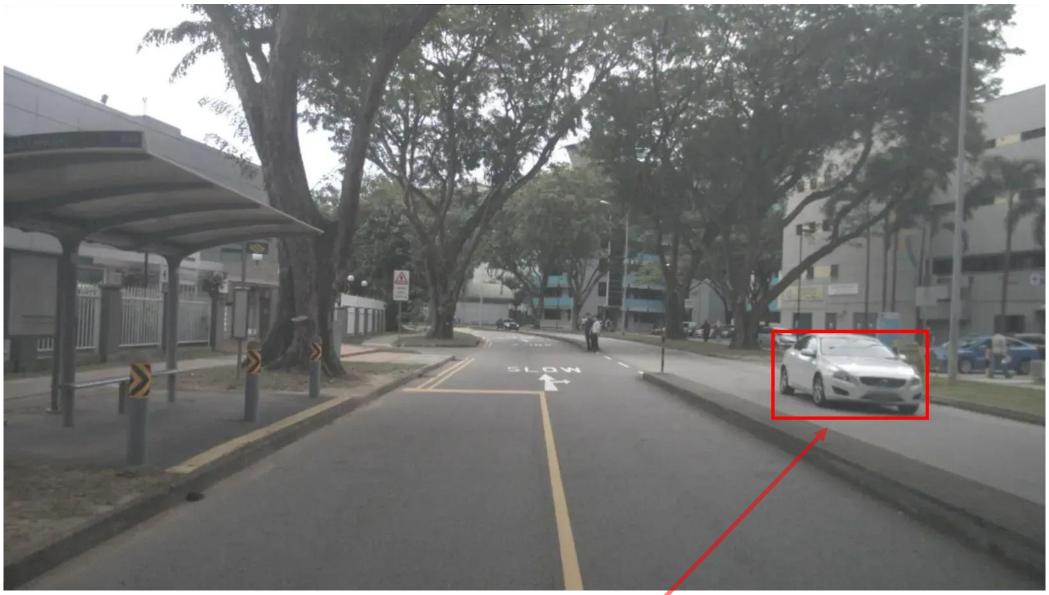
A Example Illustrating the Reasoning Process



Question:
Task Description:
The primary goal of this task is to identify the direction that the specified object is facing in the given image. **The camera in the image is facing North**, and you need to analyze the object's orientation based on this reference.
Question: **Which direction is the white car facing in the image?**
Options: \n- North\n- South\n- East\n- West
First, detect and list the target objects mentioned in the question from the image. Output their bounding boxes under the <location> tag. Next, reason carefully and step-by-step under the <think> tag to ensure logical accuracy and robustness, including any relevant error checks. Finally, provide a concise and definitive response in the <answer> tag. Use the following format: <location>[[Detected target object]: [xmin, ymin, xmax, ymax], ...]</location> <think>[Step-by-step reasoning with attention to detail and potential error checks]</think><answer>[Final answer]</answer>

Answer:
<location>white car: **[789, 519, 895, 601]**</location>
<think>
Step 1: Establish a Reference Direction
The camera is facing North, so this is our baseline direction.
Step 2: Determine Object Position Relative to Camera
The white car is positioned in front of the camera on the road, **moving towards the camera.**
Step 3: Analyze Object Orientation Based on Reference and Position
Since the car is moving towards the camera and the camera is facing North, the car must be facing South.
Step 4: Consider Additional Contextual Clues
There are no visible contextual clues such as road markings or traffic signs that would suggest a different orientation for the car.
Step 5: Verify Consistency with Given Options
The derived orientation of the car facing South is consistent with the given options.
Step 6: Formulate a Clear Final Answer
After verifying the analysis, the final answer is clear.
</think>
<answer>**South**</answer> ☑

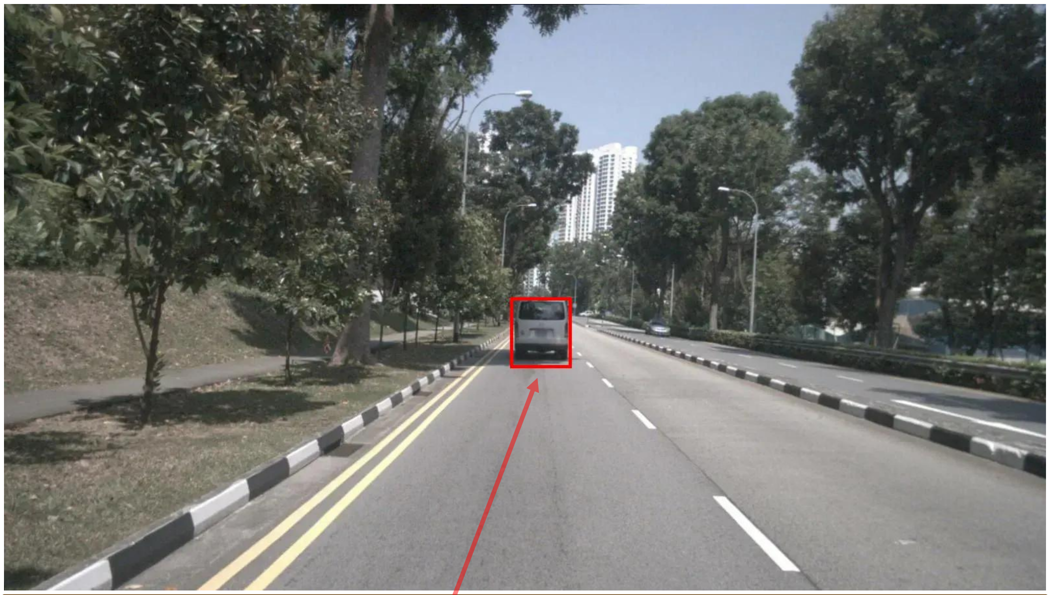
Figure 5: Example of Yaw Angle Determination task.



Question:
Task Description:
The primary goal of this task is to accurately identify and provide the coordinates of a specified object within a given image. Your task is to analyze the image, locate the object, and return its position in the form of coordinates [x, y].
Question: **Where is the white car located in the image?**
First, detect and list the target objects mentioned in the question from the image. Output their bounding boxes under the <location> tag. Next, reason carefully and step-by-step under the <think> tag to ensure logical accuracy and robustness, including any relevant error checks. Finally, provide a concise and definitive response in the <answer> tag. Use the following format: <location>[[Detected target object]: [xmin, ymin, xmax, ymax], ...]</location> <think>[Step-by-step reasoning with attention to detail and potential error checks]</think><answer>[Final answer]</answer>

Answer:
<location>white car: **[1179, 500, 1420, 637]**</location>
<think>
Step 1: The task is to identify the coordinates of the white car in the image.
Step 2: **The image shows a white car on the right side of the road, close to the curb.**
Step 3: The white car is the object of interest, and its position should be identified.
Step 4: **Estimating the coordinates, the white car appears to be around the middle-right of the image. The x-coordinate is approximately 1320, and the y-coordinate is around 551.**
Step 5: Verifying the solution, the coordinates seem plausible as they place the car near the middle-right of the image.
Step 6: No alternative scenarios are considered as the car is clearly visible and identifiable.
Step 7: The final answer is [1320, 551].
Step 8: The assumptions made are based on the visual cues and the car's position relative to the image boundaries.
</think>
<answer>**[1320, 551]**</answer> ☑

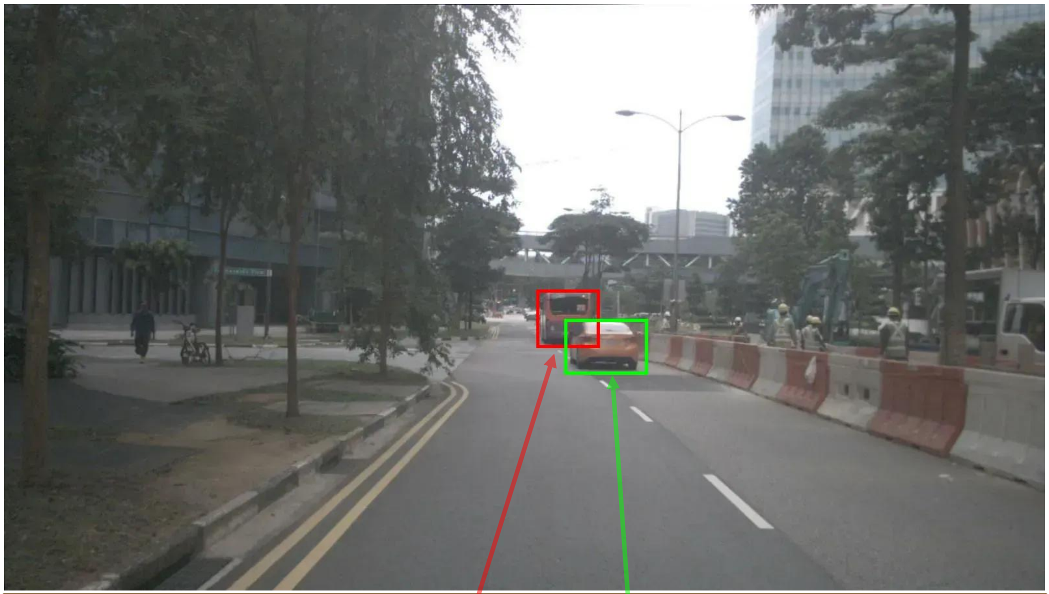
Figure 6: Example of Pixel Location Estimation task.



Question:
Task Description:
The primary goal of this task is to estimate the vertical distance of the specified object in the image from the camera, which is positioned at the origin. You need to analyze the image and choose the correct range of distance from the camera based on the visual cues provided.
Question: **How far is the vertical distance of the white car in the picture from the camera?**
Options:- Between 15 meters and 18 meters - Between 19 meters and 23 meters - Between 24 meters and 31 meters
First, detect and list the target objects mentioned in the question from the image. Output their bounding boxes under the <location> tag. Next, reason carefully and step-by-step under the <think> tag to ensure logical accuracy and robustness, including any relevant error checks. Finally, provide a concise and definitive response in the <answer> tag. Use the following format: <location>[[Detected target object]: [xmin, ymin, xmax, ymax], ...]</location> <think>[Step-by-step reasoning with attention to detail and potential error checks]</think><answer>[Final answer]</answer>

Answer:
<location>white car: **[777, 450, 871, 558]**</location>
<think>
Step 1: Understand the Context and Question
The question asks for the vertical distance of the white car from the camera, which is positioned at the origin. The options provided are ranges of distances in meters.
Step 2: Identify Visual Cues and Reference Points
The image shows a road with trees on both sides and a white car in the distance. The car appears to be a standard-sized vehicle, and the road markings can be used as reference points.
Step 3: Apply Principles of Perspective and Scaling
The white car appears smaller due to perspective, indicating it is farther away. The road markings can help estimate the distance. The car is positioned near the midpoint of the road, which suggests it is not extremely far away.
Step 4: Consider Environmental and Contextual Clues
The environment is a wide road with trees and a clear sky. The car is driving away from the camera, and the road appears to be a typical urban or suburban road.
Step 5: Make Educated Guesses Based on Reasonable Assumptions
Assuming the car is a standard-sized vehicle and using the road markings as a reference, the car appears to be between 24 meters and 31 meters away from the camera.
Step 6: Verify and Refine Estimates
Cross-checking the estimate with the given options, the range of 24 meters to 31 meters seems reasonable based on the visual cues and the typical dimensions of a standard-sized car.
Step 7: Select the Most Appropriate Option
Based on the analysis, the most appropriate option is \"Between 24 meters and 31 meters.\"
</think>
<answer>**Between 24 meters and 31 meters**</answer> ✓

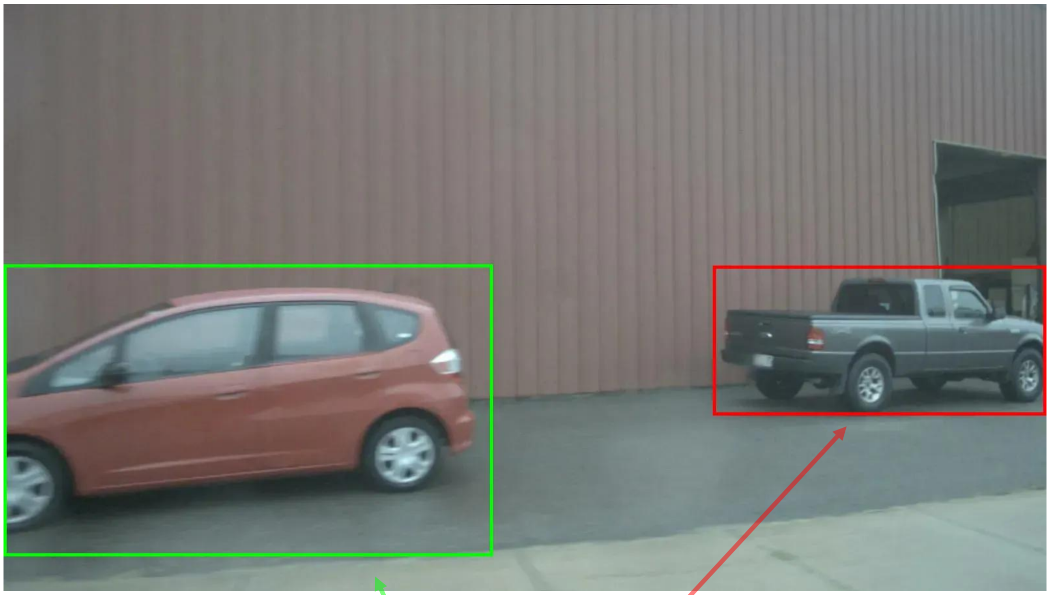
Figure 7: Example of Depth Range Determination task.



Question:
Task Description:
The primary goal of this task is to determine which of the two objects is closer to the camera that captured the image below. You need to assess the relative distance between the two objects based on the camera's perspective.
Question: **Which object, the orange rigid bus or the orange car, is closer to the camera?**
Options: - The orange rigid bus - The orange car - Almost the same
First, detect and list the target objects mentioned in the question from the image. Output their bounding boxes under the <location> tag. Next, reason carefully and step-by-step under the <think> tag to ensure logical accuracy and robustness, including any relevant error checks. Finally, provide a concise and definitive response in the <answer> tag. Use the following format: <location>[[Detected target object]: [xmin, ymin, xmax, ymax], ...]</location><think>[Step-by-step reasoning with attention to detail and potential error checks]</think><answer>[Final answer]</answer>

Answer:
<location>orange rigid bus: **[818, 438, 915, 527]**, orange car: **[861, 483, 989, 568]**</location>
<think>
1. **Identify Scene and Objects**: The scene is a road with an orange rigid bus and an orange car visible. **The bus is farther down the road, and the car is closer to the camera.**
2. **Consider Size and Perspective**: **The bus appears smaller compared to the car. Since objects closer to the camera appear larger, this suggests the car is closer to the camera.**
3. **Evaluate Clarity and Detail**: The car appears more detailed and sharper compared to the bus, which is further away and less detailed.
4. **Analyze Positioning and Overlap**: **The car is positioned closer to the camera and partially overlaps the bus, indicating it is nearer to the camera's viewpoint.**
5. **Examine Depth Cues**: **Linear perspective shows that the road narrows into the distance, and the bus is closer to this narrowing point, suggesting it is farther away.**
6. **Consider Lighting and Shadows**: The lighting and shadows on both objects are consistent with the car being closer to the camera.
7. **Visualize Camera Perspective**: Visualizing from the camera's perspective confirms that the car is closer, as it is larger, more detailed, and positioned closer in the frame.
8. **Cross-Verify Observations**: All observations consistently support the car being closer to the camera.
9. **Select the Best Answer**: Based on the analysis, the most appropriate answer is \"The orange car.\"
</think>
<answer>**The orange car**</answer> ✓

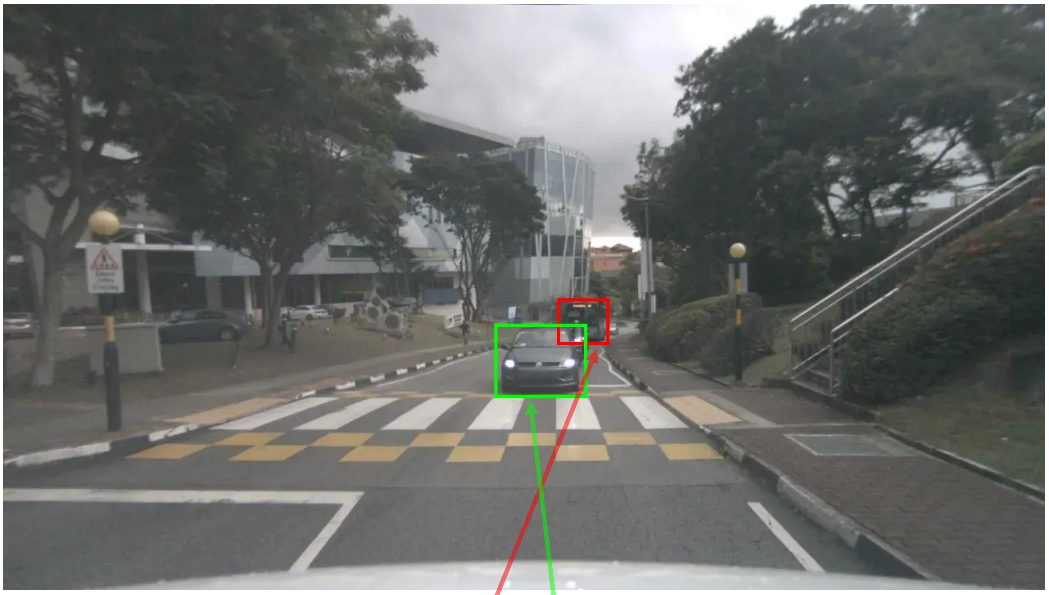
Figure 8: Example of Distance Estimation task.



Question:
Task Description:
The primary goal of this task is to determine the relative left-right positioning of the two objects from the camera's perspective.
Question: **Which is further right, the gray truck or the red car?**
Options: - The gray truck - The red car - Almost the same
First, detect and list the target objects mentioned in the question from the image. Output their bounding boxes under the <location> tag. Next, reason carefully and step-by-step under the <think> tag to ensure logical accuracy and robustness, including any relevant error checks. Finally, provide a concise and definitive response in the <answer> tag. Use the following format: <location>[[Detected target object]: [xmin, ymin, xmax, ymax], ...]</location> <think>[Step-by-step reasoning with attention to detail and potential error checks]</think><answer>[Final answer]</answer>

Answer:
<location>gray truck: **[1088, 401, 1599, 630]**, red car: **[0, 399, 750, 846]**</location>
<think>
Step 1: The image shows a gray truck and a red car parked near a building.
Step 2: The camera perspective is from a slightly elevated angle, looking towards the vehicles.
Step 3: **The gray truck is positioned to the right of the red car relative to the camera's perspective.**
Step 4: **Comparing the positions, the gray truck is clearly further to the right compared to the red car.**
Step 5: **There are no significant depth or perspective effects that would distort the apparent left-right positioning.**
Step 6: The analysis aligns with the visual evidence, confirming the gray truck is further right.
Step 7: Based on the analysis, the gray truck is the correct answer.
</think>
<answer>**The gray truck**</answer> ☑

Figure 9: Example of Left/Right Determination task.



Question:
Task Description:
The primary goal of this task is to determine the relative front-back positioning of the two objects from the camera's perspective, **where the object farther from the camera is considered to be more forward**.
Question: **Is the blue rigid bus behind the gray car?**
Options: - Yes – No - Almost the same in terms of front-back position
First, detect and list the target objects mentioned in the question from the image. Output their bounding boxes under the <location> tag. Next, reason carefully and step-by-step under the <think> tag to ensure logical accuracy and robustness, including any relevant error checks. Finally, provide a concise and definitive response in the <answer> tag. Use the following format: <location>[[Detected target object]: [xmin, ymin, xmax, ymax], ...]</location>
<think>[Step-by-step reasoning with attention to detail and potential error checks]</think><answer>[Final answer]</answer>

Answer:
<location>blue rigid bus: **[850, 452, 930, 523]**, gray car: **[754, 491, 896, 604]**</location>
<think>
Step 1: Identify the Key Objects and Their Descriptions
- The key objects are the blue rigid bus and the gray car.
- The blue rigid bus is visible in the distance, further down the road.
- The gray car is closer to the camera, positioned on the road in front of the bus.
Step 2: Understand the Camera Perspective
- The camera is positioned at a lower angle, possibly from a vehicle, and is facing forward along the road.
- The objects are positioned along the road, with the bus being farther away and the car being closer to the camera.
Step 3: Interpret Relative Positioning Terms
- According to the task, \"forward\" means farther from the camera, and \"backward\" means closer to the camera.
- The blue rigid bus is farther from the camera compared to the gray car.
Step 4: Analyze Object Positions Relative to the Camera
- The gray car is closer to the camera than the blue rigid bus.
- Therefore, the blue rigid bus is more forward relative to the camera.
Step 5: Consider Possible Scenarios and Visualize
- Visualizing the scene confirms that the gray car is in front of the blue rigid bus from the camera's perspective.
Step 6: Evaluate Given Options and Select the Best Answer
- Given the options, \"No\" is the correct answer because the blue rigid bus is not behind the gray car; it is more forward from the camera's perspective.
Step 7: Verify Accuracy and Provide Final Answer
- The reasoning aligns with the task requirements and the provided information.
</think>
<answer>**No**</answer> ✓

Figure 10: Example of Front/Behind Determination task.

B Template for Generating VQA Tasks

Template for Yaw Angle Determination Task

Task Description:
The primary goal of this task is to identify the direction that the specified object is facing in the given image. The camera in the image is facing {}, and you need to analyze the object's orientation based on this reference.

Question:
Which direction is {} facing in the image?
Options: - {} - {} - {} - {}

Template for Pixel Location Estimation Task

Task Description:
The primary goal of this task is to accurately identify and provide the coordinates of a specified object within a given image. Your task is to analyze the image, locate the object, and return its position in the form of coordinates [x, y].

Question:
Where is {} located in the image?

Template for Depth Range Determination Task

Task Description:
The primary goal of this task is to estimate the vertical distance of the specified object in the image from the camera, which is positioned at the origin. You need to analyze the image and choose the correct range of distance from the camera based on the visual cues provided.

Question:
How far is the vertical distance of {} in the picture from the camera?
Options: - {} - {} - {}

Template for Distance Estimation Task

Task Description:
The primary goal of this task is to determine which of the two objects is closer to the camera that captured the image below. You need to assess the relative distance between the two objects based on the camera's perspective.

Question:
Which object, {} or {}, is {} to the camera?
Options: - {} - {} - Almost the same

Template for Left/Right Determination Task

Task Description:
The primary goal of this task is to determine the relative left-right positioning of the two objects from the camera's perspective.

Question:
Which is further {}, {} or {}?
Options: - {} - {} - Almost the same

Template for Front/Behind Determination Task

Task Description:
The primary goal of this task is to determine the relative front-back positioning of the two objects from the camera's perspective, where the object farther from the camera is considered to be more forward.

Question:
Is {} {} {}?
Options: - Yes - No - Almost the same in terms of front-back position

C Structured Response Format for the VQA Task

Structured Response Format with Location Tag

First, detect and list the target objects mentioned in the question from the image. Output their bounding boxes under the <location> tag. Next, reason carefully and step-by-step under the <think> tag to ensure logical accuracy and robustness, including any relevant error checks. Finally, provide a concise and definitive response in the <answer> tag.

Use the following format:
<location>[[Detected target object]: [xmin, ymin, xmax, ymax], ...]</location>
<think>[Step-by-step reasoning with attention to detail and potential error checks]</think>
<answer>[Final answer]</answer>

Structured Response Format without Location Tag

Reason carefully and step-by-step under the <think> tag to ensure logical accuracy and robustness, including any relevant error checks. Finally, provide a concise and definitive response in the <answer> tag.

Use the following format:
<think>[Step-by-step reasoning with attention to detail and potential error checks]</think>
<answer>[Final answer]</answer>

D Prompts for High-Quality Chain-of-Thought Generation

Prompt for Generating Chain-of-Thought

Analyze the following task step by step to derive the best possible answer.
Task: {task}
Answer: {answer}
Please provide a detailed reasoning process, verify its accuracy, and then give your final answer clearly.

Prompt for Summarizing Rules from Examples

You are given the following reasoning examples. Analyze these examples to identify the underlying, generalizable problem-solving principles.
Examples: {examples}
Present your findings as bullet points in this format:
- Step 1: [core principle] - Step 2: [core principle] ...
Ensure these rules can be applied broadly to similar questions.

Prompt for Generating Answers Using Extracted Rules

Use the following principles to answer the question:
{rules}
Question: {question} Answer: {answer}
Provide a concise solution with key reasoning steps in the following format: <think>[Your step-by-step reasoning]</think> <answer>[Final answer]</answer>

Prompt for Verifying and Refining Reasoning and Answers

{response}
Evaluate the structured response above for logical consistency and completeness. Specifically:
1. Does the reasoning in <think> logically support the conclusion in <answer>?
2. Are there any internal contradictions, logical errors, or missing steps in the reasoning?
3. Is the reasoning chain complete and valid?
Provide your evaluation in the following format:
<reason>[A concise justification of your assessment or a brief note confirming the reasoning's validity]</reason> <validation>Valid / Invalid</validation>
Then, regardless of validity, output the full response in the following format:
- Keep <answer> unchanged.
- Modify <think> only if necessary to ensure logical soundness.
<think>[final version of reasoning steps]</think>
<answer>[original final answer]</answer>