DRIVINGVQA: A Dataset for Interleaved Visual Chain-of-Thought in Real-World Driving Scenarios

Charles Corbière* **Simon Roburin*** **Syrielle Montariol***
Antoine Bosselut **Alexandre Alahi**

EPFL, Switzerland

{firstname.lastname}@epfl.ch

https://vita-epfl.github.io/DrivingVQA/

Abstract

While chain-of-thought (CoT) prompting improves reasoning in large language models, its effectiveness in vision-language models (VLMs) remains limited due to over-reliance on textual cues and memorized knowledge. To investigate the visual reasoning capabilities of VLMs in complex real-world scenarios, we introduce DRIVINGVQA, a visual question answering dataset derived from driving theory exams, which contains 3,931 multiple-choice problems with expert-written explanations and grounded entities relevant to the reasoning process. Leveraging this dataset, we explore the benefits of incorporating entity-related information, such as entity names, spatial coordinates, and visual content, through supervised fine-tuning to enhance the model's reasoning abilities. Our experiments demonstrate that interleaving textual explanations with visual tokens extracted from entities relevant to the question improves answer accuracy by 3.1% and reasoning accuracy by 4.6% over vanilla CoT prompting. Furthermore, we demonstrate that this retrieval-based approach effectively scales to the larger A-OKVQA reasoning dataset by leveraging automatically generated pseudo-labels, outperforming CoT prompting. Code and dataset will be made publicly available.

1 Introduction

Chain-of-thought (CoT) (Wei et al., 2022) is a prompting strategy that aims at enhancing the reasoning capabilities of large language models (LLMs) (OpenAI, 2024; Touvron et al., 2023; Jiang et al., 2023) and vision-language models (VLMs) (Wang et al., 2024; Liu et al., 2024c; Alayrac et al., 2022; Lin et al., 2024). While well-suited for mathematical and logical reasoning, CoT prompting shows limited effectiveness when tasked with visual data and spatial information (Sprague et al., 2024; Zhang et al., 2024b; Shao et al., 2024). A critical challenge is VLMs' tendency to excessively rely on textual inputs and memorized knowledge rather than visual inputs, which can lead to hallucinations (Guan et al., 2024; Liu et al., 2024a; Jiang et al., 2024; Chandhok et al., 2024). Despite advances in multimodal alignment through captioning (Zhang et al., 2024a; Özdemir and Akagündüz, 2024; Li et al., 2024b), scene graphs (Mitra et al., 2024), or coordinate-based supervision (Chen et al., 2023; Lei et al., 2024; Liu et al., 2025b), existing models still struggle to reason about visual scenes with rich semantic content and intricate spatial configurations.

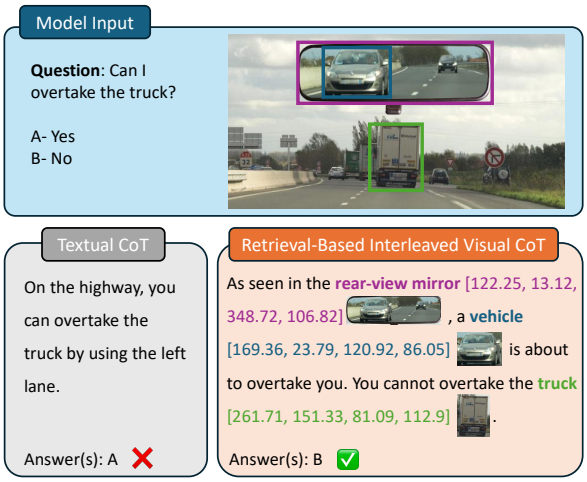


Figure 1: **Illustration of retrieval-based interleaved visual chain-of-thought in DRIVINGVQA**. The models are trained to reason using entities' coordinates and their retrieved visual patches in an interleaved manner, outperforming standard chain-of-thought.

In this work, we propose to strengthen VLMs' visual reasoning ability by implementing *visual chain-of-thought* as a process where the model leverages the visual and spatial information explicitly as part of its reasoning, jointly detecting relevant regions of interest in the scene, identifying their attributes and spatial relationships using vi-

*Equal contribution, with order determined alphabetically.

Findings of the Association for Computational Linguistics: EACL 2026, pages 3309–3333
March 24-29, 2026

sual patches and region coordinates, and deducing appropriate actions (see Fig. 1).

A key challenge in studying such reasoning lies in the lack of datasets that both evaluate and teach multimodal reasoning. Existing reasoning-oriented visual question answering (VQA) datasets either rely on synthetic or schematic visuals (Chia et al., 2024), lack grounding annotations (Lu et al., 2022; Liu et al., 2025a), or present oversimplified scenarios with a single region of interest (Shao et al., 2024). Additionally, most explanations are generated from a set of predefined templates or by using an exogenous LLM (Chia et al., 2024; Shao et al., 2024; Marcu et al., 2024). These explanations can be repetitive, error-prone, and biased, limiting their effectiveness as training signals.

Driving theory exams naturally address these gaps. They test the understanding of traffic laws, spatial awareness, and safe driving practices in realistic real-world scenarios, making them ideal for studying grounded reasoning. They require multi-entity perception and spatial reasoning (*e.g.* assessing relative positions of vehicles, signs, or lane markings) as well as rule-based logic (*e.g.*, right-of-way, overtaking conditions). Uniquely, they offer expert step-by-step didactic explanations designed to teach learners how to reason about dynamic environments.

We introduce DRIVINGVQA, a visual reasoning dataset derived from publicly available French driving theory exams. It contains 3,931 samples, each featuring one or two visual questions with multiple answer choices, alongside expert-written explanations. We propose a pipeline to extract relevant entities with bounding box coordinates and generate reasoning traces interleaved with the entities (see Figure 2), allowing models to leverage the entities' visual attributes and location as part of the reasoning process. DRIVINGVQA not only fills a critical gap in existing VQA benchmarks by introducing complex real-world driving scenes paired with grounded, expert-written explanations, but also provides a foundation for building rule-based interpretable reasoning layers in autonomous driving systems.

Using DRIVINGVQA, we explore how entity-aware reasoning benefits VLMs. While state-of-the-art models struggle in zero-shot settings, we show that training strategies incorporating relevant entities information, in particular through our *interleaved visual CoT* strategy – visual crops interleaved with textual reasoning (see Fig. 5) significantly improve both answer accuracy (+3.1 pts) and reasoning correctness (+4.6 pts) compared to standard CoT prompting. Finally, we demonstrate that our reasoning strategy coupled with DRIVINGVQA construction pipeline effectively scales to A-OKVQA (Schwenk et al., 2022), a larger and general-domain dataset without relevant entity annotations using automatically extracted pseudo-labels.

2 Related Work

2.1 Visual Reasoning Datasets

Since the seminal work of Antol et al. (2015), visual question answering (VQA) datasets have flourished over the past decade (Goyal et al., 2017; Marino et al., 2019; Hudson and Manning, 2019). With the emergence of VLMs and their enhanced reasoning capabilities, recent efforts target the development and assessment of these models' advanced reasoning skills (Lu et al., 2022; Chia et al., 2024; Shao et al., 2024; Yue et al., 2024). But due to the challenges of manually collecting detailed explanations, many rely on LLM-generated (Shao et al., 2024) or template-based (Chia et al., 2024) explanations, which often fall short of the diversity and quality of human-authored content. Additionally, some datasets rely on synthetically generated puzzle-like images, such as PuzzleVQA (Chia et al., 2024), or diagrams and schematic figures, such as ScienceQA (Lu et al., 2022); which do not adequately capture the complexity of natural images. Dataset with natural images and realistic questions reflecting complex situations often belong to specific domains such as autonomous driving, where the need for explainable driving behavior in control planning has led to the creation of many VQA datasets (Kim et al., 2018; Marcu et al., 2024; Qian et al., 2024; Sima et al., 2024). However, only a few include reasoning traces. The recently developed LingoQA (Marcu et al., 2024) comes closest to our work, featuring VQA pairs with textual descriptions of driver actions, justifications, and road observations. However, its questions and answers are synthetically generated by GPT-4 (Contributors, 2024) and do not include any bounding boxes or relevant entities to visually ground explanations.

In the general domain, VisualCoT-GQA (Shao et al., 2024) pairs relevant entity coordinates with explanations but covers only one entity per image and uses synthetic explanations. In contrast,

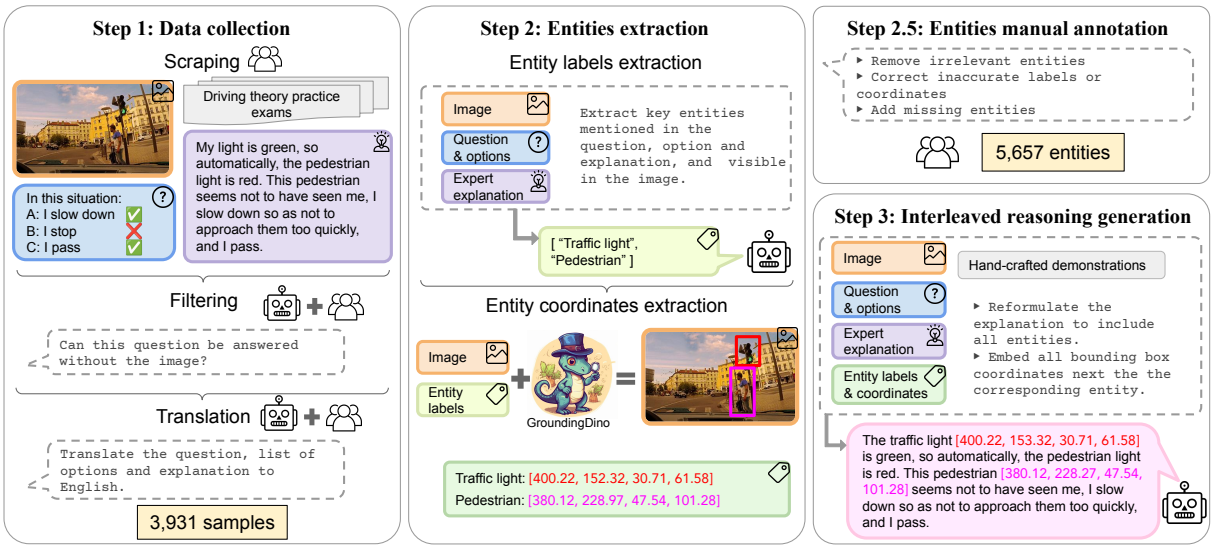


Figure 2: **DRIVINGVQA creation pipeline**. We collect and filter real-world driving theory problems along with expert explanations (*Step 1*); annotate relevant entities and their coordinates in each sample, first automatically (*Step 2*) then with manual refinement (*Step 2.5*); and generate reasoning traces interleaved with relevant entities (*Step 3*).

DRIVINGVQA combines real-world images from driving scenarios with (1) expert-written explanations and (2) multiple relevant entity annotations, interleaved inside the explanation.

2.2 Visual Chain-of-Thought in VLMs

Several methods have sought to extend CoT prompting to visual reasoning. Some generate textual scene descriptions before answering the question (Zhang et al., 2024a; Özdemir and Akagündüz, 2024; Li et al., 2024b; Qiao et al., 2025). The description may take the form of captions, a series of visual question-answer pairs that extract additional image details (Zheng et al., 2023a), or sparser representations of the input image such as scene graphs (Mitra et al., 2024). Similarly, visual programming (Gupta and Kembhavi, 2023; Surís et al., 2023; Li et al., 2025) adopts a neuro-symbolic approach that leverages off-the-shelf models to extract information from images and convert it into text for subsequent reasoning.

Another research direction focuses on explicit grounding within the input image. This includes training VLMs to generate CoT along with coordinates of relevant image regions (Liu et al., 2025b; Chen et al., 2023), or incorporating visual prompts directly into the image, to be used by the VLM when reasoning. These visual prompts can take the form of a grid (Lei et al., 2024) or a set of visual markers (Yan et al., 2024) that indicate specific areas of the image. More closely related to our work, CogCOM (Qi et al., 2024) modifies the input image before using the transformed version to answer a question, notably by zooming on a specific image region. Concurrently, Visual CoT (Shao et al., 2024) designs a visual sampler that selects a sub-region of the input image to answer a question; Gao et al. (Gao et al., 2024) proposes extracting different sub-regions of an image using visual attention patterns and generating reasoning for each sub-region. In contrast, our dataset allows VLMs to learn how to perform interleaved multimodal CoT with image patches retrieved from the input image using their bounding box coordinates, ultimately enabling more integrated visual reasoning.

3 DRIVINGVQA Dataset

We construct DRIVINGVQA through a three-step process (Figure 2): (1) collecting and filtering challenging real-world driving scenarios from driving theory tests, (2) annotating relevant entities in each sample, and (3) generating reasoning traces interleaved with relevant entities.

3.1 Data Collection

In many countries, obtaining a driver's license requires passing a theoretical test and a practical driving test. In France, the theoretical exam consists of 40 image-based multiple-choice questions (MCQs) that assess knowledge of traffic laws, road signs, and safe driving practices. Candidates must carefully analyze the image provided to answer the question. Given this emphasis on visual analysis

and the use of real-world images, we focus on French driving theory exams.

Collection. To help candidates prepare for this exam, several online platforms offer practice driving theory problems. These resources are often freely accessible and do not require registration. We obtain MCQs from such platforms. Each collected sample includes questions with two to four possible answers, where multiple answers may be correct. Some questions include two sub-questions, each with two possible answers. We standardize the format to include the image of the driving scene, the question, the list of possible answers, the correct answer(s), and an expert-annotated explanation. The explanations are written by driving theory teachers when creating the exams and are very detailed, describing the situation in the image and reasoning about it to explain the correct answer. An example of a driving theory problem is shown in Figure 2 (step 1).

Filtering. We retain only samples requiring visual understanding about driving scenes using GPT-4o (OpenAI, 2024). We perform a manual review of excluded images to address potential misclassifications. In a validation exercise involving 60 randomly selected samples, three expert annotators achieved inter-annotator agreement scores of approximately 0.95 for Krippendorff's alpha, Fleiss' kappa, and the average pairwise Cohen's kappa.

Translation. We translate all questions, answer options, and explanations from French to English using GPT-4o-mini, then manually review and refine them for consistency and fidelity.

The resulting DRIVINGVQA dataset contains 3,931 samples, divided into a training set (80%) and a test set (20%, 789 samples). Figure 3 provides an overview of the dataset statistics, including the distribution of the number of possible answers, correct answers, and explanation length in terms of word count.

3.2 Entities Extraction

We augment the collected data with annotations of relevant entities to answer the question, specifying their names and locations in the image. To reduce the burden of manual annotation, we introduce a framework, detailed in step 2 and step 2.5 of Figure 2, that identifies possible relevant entities and their location in the image.

Automated extraction. For each sample, we identify the entities from a list of domain-specific key entities that are visible in the images and referenced in the question, possible answers, and explanation using GPT-4o-mini. Then, GroundingDINO (Liu et al., 2023) is used to localize these entities in the image, obtaining (entity label, bounding box coordinates) pairs. Finally, we apply heuristics to refine these outputs, such as grouping similar labels under unified entity names. Details of this pipeline are provided in Appendix B.2.

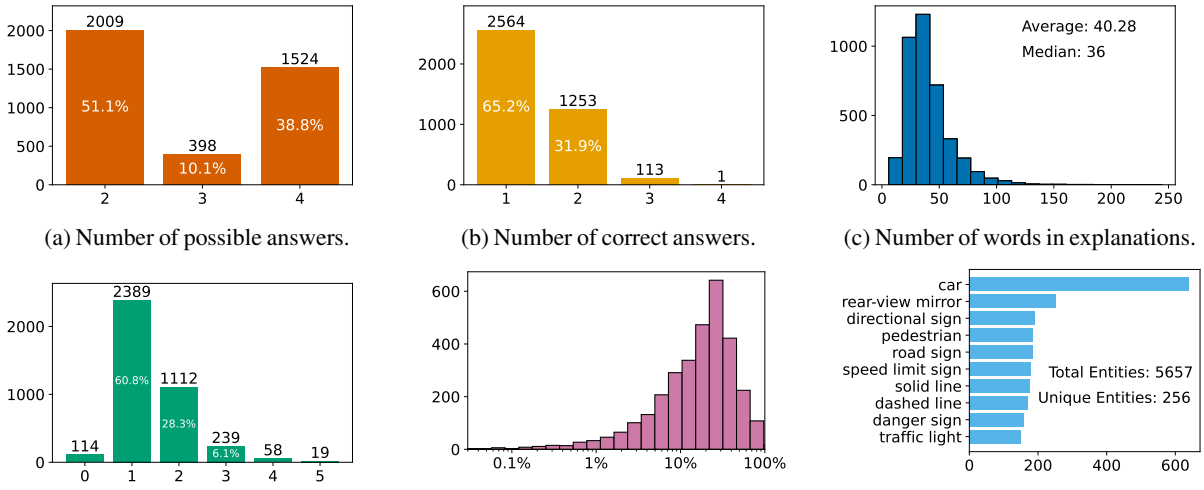
Manual annotation. In DRIVINGVQA, human experts refine the pseudo-annotated data by removing irrelevant entities, correcting inaccurate labels and coordinates, and adding missing entities. In total, 5,657 entities (spanning 256 unique labels) were annotated with precise bounding boxes, averaging 1.4 entities per image. Figure 3 shows the distribution of entities per question, their image coverage, and the top 10 most frequent entity labels.

3.3 Interleaved Reasoning Generation

Explanations in DRIVINGVQA do not systematically make explicit reference to the human-annotated relevant entities. To ground explanations with visual cues, we integrate the list of relevant entities into the explanations, creating *interleaved* explanations. We prompt GPT-4o with two manually crafted demonstrations to embed each set of bounding-box coordinates into the explanation, reformulating it slightly to refer to the entity at the beginning of each step of the reasoning; thereby achieving a more causal formulation aligned with the autoregressive nature of the training objective (see examples in Figure 1 and the step 3 of Figure 2). Hence, the resulting interleaved explanation refers to the relevant entities early in the sentences, allowing the reasoning process to be conditioned on the content of the image crops. This process is followed by light rule-based post-processing to refine the final explanation (see details and prompt in Appendix B.1).

3.4 Evaluation Metrics

Answer correctness. The performance on the DRIVINGVQA test split (789 samples) is measured with the *exam score*, analogous to the real driving theory score used to evaluate candidates: all correct answers must be selected to score a question correctly. Note that this score is equivalent to the exact match ratio. Since the task is analogous to multi-



(a) Number of possible answers. (b) Number of correct answers. (c) Number of words in explanations.

(d) Number of relevant entities. (e) Image coverage by entities, log-scale. (f) Top 10 most frequent entities.

Figure 3: **DRIVINGVQA dataset statistics.** In total, the dataset contains 3 931 examples. Each sample has 2 to 4 answer choices and up to 4 answers can be correct. Expert-written explanations are relatively long – 40 words on average – and mention up to 5 relevant entities visible in the image (1.4 on average).

label classification, we also report the F1-score.

Reasoning correctness. While a high exam score indicates the quality of answer selection, it does not necessarily reflect the correctness of the reasoning. A model may rely on heuristics or memorization, a limitation that the exam score alone does not fully capture. To address this, we assess the correctness of the model's reasoning by comparing it against DRIVINGVQA's ground truth explanations, using GPT-4o-mini as an evaluator, leveraging the LLM-as-a-judge paradigm (Zheng et al., 2023b). It is increasingly used to scale the evaluation of open-ended generations of LLMs and VLMs, and is shown to align well with human judgment in pairwise comparisons (Liu et al., 2024d; Gu et al., 2024). The prompt used for GPT-4o-mini is detailed in Appendix D.3. The model is provided with the question and answer choices, the ground truth explanation, and the reasoning generated by the VLM. It is then tasked with identifying the key arguments in both explanations, checking for missing or contradictory elements, and determining whether they align. To further confirm the reliability of LLM-as-a-judge in our context, we manually annotated 50 pairs of ground-truth and model-generated explanations from models fine-tuned under different strategies. This evaluation informed prompt refinements and yielded an F1 score of 0.83, supporting the consistency of the automated judge with human judgments.

Thus, we define the *reasoning correctness* as the proportion of samples in the test set where the judge determines that the model's reasoning matches the ground truth reasoning.

4 Experiments

In this section, we first evaluate the zero-shot performance of state-of-the-art VLMs on DRIVINGVQA. Then, we implement Retrieval-based Interleaved Visual Chain-of-Thought (*RIV-CoT*) prompting, exploring the benefits of incorporating entity-related information, such as entity names, spatial coordinates, and visual content, through supervised fine-tuning to enhance the model's reasoning abilities. Finally, we evaluate the correctness of the generated entity coordinates and of the model's reasoning.

4.1 Zero-Shot Evaluation

Figure 4 presents the zero-shot performance of popular open-source models, including the LLaVA-OneVision (LLaVA-OV, Li et al., 2024a) variants (0.5B, 7B, 72B) used in the subsequent experiments, and proprietary models on the DRIVINGVQA test set. These models are provided a prompt defining the expected format and asked to directly predict the answer (e.g. 'Answer: B,C'). We include a random baseline where responses are selected randomly from all possible answer combinations for each question. Results for the random baseline are aggregated over 1,000 runs, reporting the mean exam and F1 scores.

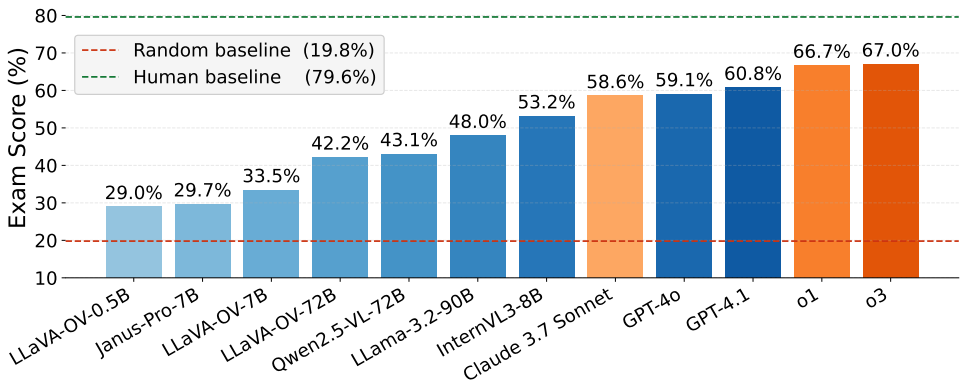


Figure 4: **Comparison of zero-shot model performance on the DRIVINGVQA test set**. The dashed lines indicate the random baseline (19.8%) and the human baseline (79.6%). Orange bars indicate large reasoning VLMs (trained using Reinforcement Learning to reason before answering).

To estimate human performance, we recruit six participants with varying driving experience. They are asked to answer batches of 40 randomly selected samples in under 20 minutes, simulating the operational driving theory exam conditions. Their average exam score of 79.6% falls below the official passing score of 87.5% (35/40). This drop reflects the increased difficulty of DRIVINGVQA, as it focuses on visually challenging questions after the data filtering process.

Larger models and reasoning models tend to perform better, with OpenAI's o3 (OpenAI, 2025) achieving the highest exam score (67.0%). Nevertheless, it remains far from the human baseline (79.6%), which illustrates the benchmark's difficulty due to its domain-specific images and knowledge and the complexity of the visual scenes. We also assess the impact of visual inputs in DRIVINGVQA by evaluating GPT-4o (OpenAI, 2024) without image information. This results in a substantial performance drop to 33.1% (-26 pts) compared to its image-enabled counterpart. It confirms that images are necessary to accurately answer DRIVINGVQA's questions, and knowledge-based shortcuts are not sufficient.

4.2 Retrieval-based Interleaved Visual CoT

We explore how to enhance VLMs' visual CoT by leveraging relevant entities through their label, bounding box and visual crops.

Experimental setup. We fine-tune LLaVA-OneVision (LLaVA-OV) 7B on the DRIVINGVQA train split. LLaVA-OV uses SigLIP (Zhai et al., 2023) as image encoder, a two-layer MLP as image-language adapter, and Qwen2 (Yang et al., 2024) as LLM backbone. All components are trained end-to-end with an autoregressive loss. Each training run is repeated five times with different random seeds to account for stochasticity in the fine-tuning process and all results are reported with standard deviation. Training is performed over 10 epochs, following the hyperparameters used by the authors of LLaVA-OV (Li et al., 2024a).

Baselines and ablations. An illustrative example of conversation formats with relevant entities can be found in Figure 5. The model takes as input the image, the question and the list of possible answers. The minimum baseline is the *Direct Answer*. Standard CoT is obtained by training the model on the expert explanations, so that it reasons before answering. Then, we successively train the model to generate elements relative to the relevant entities before reasoning (*CoT w/ entities*): the labels list, their bounding box coordinates, and their associated visual patch cropped from the input image. To leverage the visual patches, we implement two rounds of inference. The model first predicts relevant entities' labels and coordinates given the question; then, it is fed with the visual crops associated with the entities (see blue boxes in Figure 5) and uses them to reason then answer. Finally, we ablate the effect of interleaving the bounding box coordinates within the reasoning trace (*CoT w/ interleaved entities (labels+bbox)*).

Retrieval-based interleaved visual CoT (RIV-CoT). To jointly leverage the entities' labels, coordinates and visual patches as part of the reasoning process, we introduce RIV-CoT (see Figure 5, bottom right). It consists of a multi-turn conversation, breaking down the model's reasoning

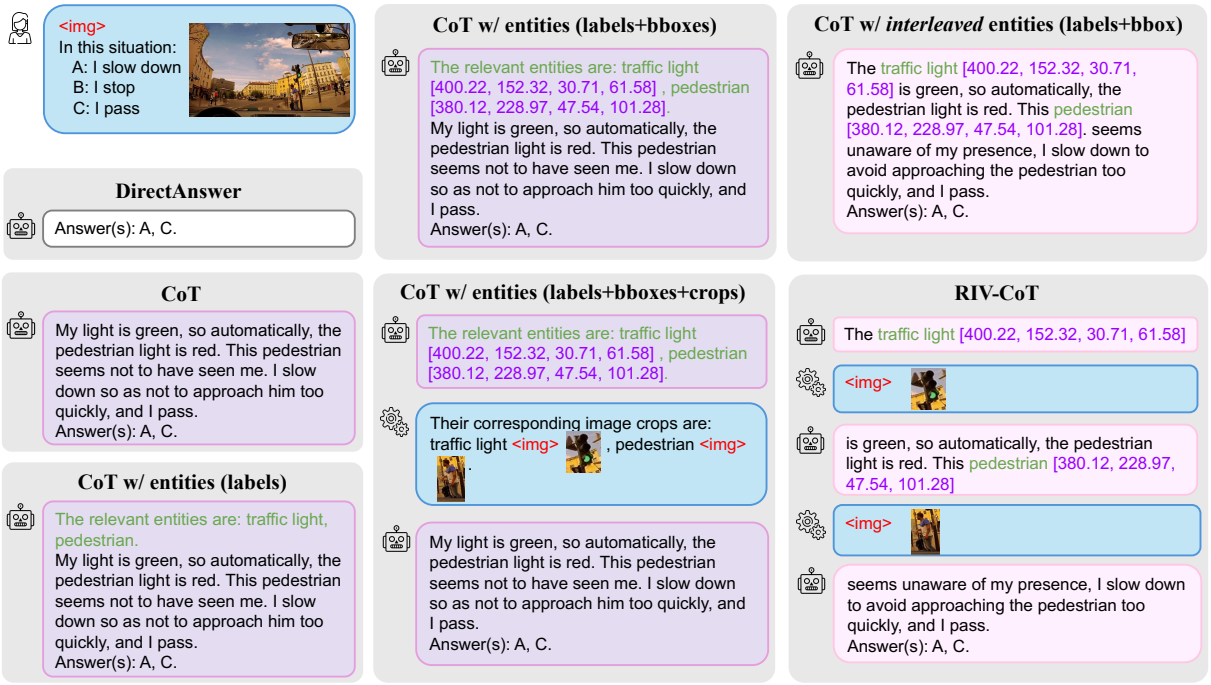


Figure 5: **Illustrations of conversation format for each fine-tuning strategy on one example**. All of them start with a prompt including an image, a question and possible answer choices.

Method	Entity Format				Answer		Reasoning
	Labels	BBoxes	Crops	Interlvd.	Exam (%)	F1-Score (%)	Acc. (%)
DirectAnswer					53.0 (±0.9)	63.3 (±0.6)	–
CoT					56.2 (±1.0)	65.8 (±0.9)	20.8 (±0.4)
CoT w/ relevant entities (labels)	✓				57.0 (±1.3)	67.1 (±1.4)	19.8 (±1.2)
CoT w/ relevant entities (labels+bboxes)	✓	✓			57.7 (±0.7)	67.3 (±0.7)	23.3 (±0.4)
CoT w/ relevant entities (labels+bboxes+crops)	✓	✓	✓		58.4 (±1.1)	67.8 (±1.1)	24.3 (±0.9)
CoT w/ *interleaved* relevant entities (labels)	✓			✓	56.4 (±0.4)	66.3 (±0.6)	24.1 (±0.3)
CoT w/ *interleaved* relevant entities (labels+bboxes)	✓	✓		✓	57.9 (±0.5)	66.8 (±0.2)	24.7 (±0.8)
RIV-CoT	✓	✓	✓	✓	**59.3** (±1.0)	**68.8** (±0.9)	**25.4** (±0.8)

Table 1: **Comparative results of fine-tuning LLaVA-OV-7B using different training strategies on DRIVINGVQA**. *Crops* stands for "visual crops". RIV-CoT is equivalent to CoT with interleaved relevant entities (label+bboxes+crops).

into sequential turns. The explanation is broken down into sequential turns, with each model turn ending with the bounding box coordinates of a relevant entity. The corresponding image crop is retrieved from the input image and fed to the model in the next turn, allowing the model to iteratively process visual and textual information in an interleaved fashion. To capture additional context and better understand the entity's attributes, we expand the detected bounding box by 50%. Training is conducted using a standard autoregressive objective. At inference time, the generation process follows an iterative approach. The model generates outputs until it predicts a bounding box; then, the image crop corresponding to the predicted coordinates, expanded by 50%, is retrieved from the input image. The cropped image is then encoded via the vision encoder and the crop tokens are inserted into the context after the bounding box, guiding the subsequent text generation. This multi-step retrieval and integration cycle continues until the final answer is produced, yielding intermediate steps that interleave image crops with textual reasoning, providing a more grounded reasoning. An illustration of RIV-CoT is available in Appendix D.2.

Method	Top-1 Acc. @ IoU 0.50 (%)		
	All	Correct	Incorrect
CoT w/ relevant entities (labels+bboxes)	68.7 (±1.4)	69.3 (±1.1)	**67.6** (±1.4)
CoT w/ relevant entities (labels+bboxes+crops)	**69.8** (±1.5)	72.3 (±1.4)	66.1 (±2.7)
CoT w/ *interleaved* relevant entities (labels+bboxes)	68.5 (±1.8)	68.9 (±1.7)	67.3 (±1.6)
RIV-CoT	69.6 (±1.7)	**72.4** (±1.2)	66.3 (±1.8)

Table 2: **Detection performance of bounding box-predicting models on DRIVINGVQA.** Top-1 accuracy at an IoU threshold of 50% is reported for all samples, correctly predicted, and incorrectly predicted samples.

Dataset	Method	MC Acc. (%)
A-OKVQA	DirectAnswer	78.2 (±0.3)
	CoT	80.6 (±0.4)
	RIV-CoT	**84.2** (±0.2)

Table 3: **MC (multiple-choice) accuracy of LLaVA-OV-7B models trained and tested on A-OKVQA (Schwenk et al., 2022)**, a general-domain VQA dataset with human-written reasoning traces.

Answer correctness results. Table 1 shows the answer and reasoning correctness of RIV-CoT along with the various baselines. As expected, fine-tuning LLaVA-OV with CoT allows the model to generate better answers than predicting answers directly. While learning to generate the list of relevant entities before generating an explanation and answer only leads to a minor improvement (+0.8 pts over CoT baseline), adding bounding box coordinates improves the exam score to 57.7% (+1.5 pts over CoT). Using visual crops further improves the performance, reaching 58.4% and showing the importance of incorporating visual information for reasoning. Finally, using explanations interleaved with visual patches, RIV-CoT achieves the best exam score (59.3%). This result shows that providing rich contextual visual information interleaved within explanations results in the most efficient way to enhance VLMs' reasoning abilities. Fine-tuning on interleaved explanation without coordinates nor visual patches leads to a score comparable to fine-tuning with the original explanations, validating that the performance gain seen with interleaving comes from the added information from the bounding boxes and visual patches.

Reasoning correctness results. We measure reasoning correctness with the evaluation metrics defined in Section 3.4. The CoT baseline reaches a reasoning correctness of 20.8%; then come models incorporating bounding boxes and image crops. Among retrieval-based approaches, RIV-CoT improves its non-interleaved counterpart by 1.1 points (25.4% vs. 24.3%) and outperforms the CoT baseline by 4.5 points, demonstrating the benefits of visual patches and interleaved formatting for reasoning accuracy.

Impact of detection performance. Table 2 shows the detection performance of models fine-tuned to predict bounding boxes of relevant entities. We measure the top-1 accuracy based on Hungarian matching at an IoU threshold of 0.50. Results are reported for all samples, as well as for samples with correct and incorrect final predictions. For correctly answered samples, the detection accuracy is slightly higher than the overall average; while samples with incorrect predictions exhibit lower accuracy. Notably, RIV-CoT achieves the highest accuracy on correctly predicted samples (72.4%) but its detection performance decreases to 66.3% on incorrect answers, indicating that detection failures may contribute to prediction errors.

To further assess the effectiveness of visual patches, we compare models using predicted entities against "oracle" entities provided as additional information in the prompt during inference. The oracle entities improve performance significantly (+3.3 pts in exam score, +3.1 pts in F1-score), highlighting a limitation in the grounding capabilities of the VLM chosen in our experiment.

To determine whether this improvement stems from relevant visual information rather than merely adding more visual tokens, we also compare with a LLaVA-OV fine-tuned with a multi-scale image patching strategy, *AnyRes*, introduced in (Liu et al., 2024b). It consists of splitting the image into N sub-patches and concatenating all their representations with the original image, and is often used to encode high-resolution images. As the maximum number of entities per sample in DRIVINGVQA is five, we use 4 splits in addition to the main image for a fair comparison. LLaVA-OV-*AnyRes* only achieves 60.7% exam score, which is -1.3 pts compared to fine-tuning with visual crops. This result confirms that carefully selected visual patches provide a stronger advantage than image splitting.

5 Scaling with Auto-Extracted Entities

DRIVINGVQA includes high-quality annotated relevant entities along with their coordinates. Such annotations are rarely available, and transferring our method to new datasets requires external tools to obtain annotations. In this section, we extend our study to a dataset where high-quality entity annotations are unavailable, demonstrating how automatically generated pseudo-annotations can serve as an alternative for improving model performance at scale.

Similar to Section 4.2, we fine-tune LLaVA-OV-7B models on A-OKVQA (Schwenk et al., 2022), a crowd-sourced dataset composed of 25K visual questions requiring commonsense and world knowledge to be answered. Each question in A-OKVQA is accompanied by multiple-choice options and rationales explaining the reasoning behind the correct answer. To obtain a list of relevant entities and their coordinates for each question, we employ an automated entity extraction pipeline based on the method introduced in Section 3.2. Specifically, we prompt GPT-4o-mini to generate potential relevant entity labels for each question. Then, we use the open-set object detector GroundingDINO (Liu et al., 2023) to localize these entities within the image (step 2 in Figure 2). Only the top 5 entities with the highest confidence scores are retained, and interleaved with the existing rationales using GPT-4o (step 3 in Figure 2).

Using these pseudo-labeled entities, we fine-tune LLaVA-OV-7B with RIV-CoT method on the train set of A-OKVQA and evaluate its performance on the available validation subset (1992 samples). As shown in Table 3, RIV-CoT achieves a multiple-choice accuracy (MC Acc.) of 84.2%, outperforming both the DirectAnswer baseline (78.2%) and vanilla CoT prompting (80.6%). These results highlight that the retrieval-based interleaved approach can largely improve the model's ability to accurately answer questions that require complex visual reasoning, even when relying on automatically extracted entities to compensate for the lack of gold-standard annotations.

6 Conclusion

In this work, we introduce DRIVINGVQA, a novel visual reasoning dataset derived from French driving theory exams, along with a manual and automated pipeline to create reasoning traces interleaved with spatial and visual information – bounding box coordinates and image crops. Then, we propose RIV-CoT, a retrieval-based interleaved visual chain-of-thought framework designed to enhance the visual reasoning abilities of VLMs. Our experiments from training on the manually annotated DRIVINGVQA and automatically annotated A-OKVQA demonstrate that RIV-CoT significantly improves both answer accuracy and reasoning correctness compared to vanilla chain-of-thought and scales effectively to datasets without additional human annotations by leveraging automatically generated pseudo-labels.

Limitations

Geographic bias and generalizability of the dataset. The dataset originates from French driving exams, which may introduce geographic bias; particularly regarding traffic laws, signage, and cultural driving norms. We selected this source due to its strong emphasis on visual analysis and the availability of real-world images, which allowed us to construct a complex VQA dataset. During preliminary exploration, we found that theoretical driving tests in other countries often rely on synthetic imagery or contain no visual components at all (e.g., the United States), and in some cases, there is no theoretical exam for licensing. Consequently, developing comparable datasets across countries proved challenging. Although traffic rules may differ internationally, the reasoning in DRIVINGVQA is not primarily tied to country-specific law: French traffic rules follow broadly standardized EU/UN conventions, and the core tasks in our dataset, hazard perception, right-of-way, pedestrian safety, blind spot awareness, interpreting speed limits or overtaking conditions, are shared across most countries. For example, the scenario in Fig. 1 reflects universal safety principles, not country-specific traffic rules.

Automated reasoning evaluation. We employ the LLM-as-a-judge framework to assess model explanations, following prior work demonstrating high alignment with human evaluation. To further confirm its reliability in our context, we validated LLM-as-a-judge performance through manual evaluation, achieving an F1 score of 0.83, which showed strong alignment between the automated and human evaluations.

Dataset quality. The dataset construction process combines both manual and automated

steps. The most demanding task, bounding box annotation for every relevant entity in each question, was too resource-intensive to assign to multiple annotators per sample. Nonetheless, annotators underwent several training rounds and followed detailed annotation guidelines to ensure quality and consistency.

Automated steps included translating samples from French to English and rewriting expert explanations to interleave them with corresponding entity coordinates. These interleaved explanations were validated through rule-based checks to eliminate hallucinations or mismatched coordinates (Appendix A.2), followed by manual spot inspections to verify alignment between visual grounding and textual reasoning. In practice, only minimal changes occurred during interleaving, typically involving the insertion of coordinate references at suitable points. To improve entity disambiguation, GPT-4o was provided with both the entity label and the corresponding image, helping differentiate entities sharing identical labels in distinct regions.

Real-life applicability of the dataset. Although DRIVINGVQA does not include action-oriented driving video QA data typical of autonomous driving datasets, it closely models real-world driving theory exams by requiring step-by-step rule-based justification rooted in visual evidence. As such, it complements existing datasets by emphasizing rule-based reasoning and grounded inference over action prediction. It thus not only serves as both a valuable benchmark that complements existing autonomous driving ones by providing a foundation for developing interpretable decision-making layers in the autonomous driving stack, but also as a testbed for demonstrating the effectiveness of interleaved multimodal explanations, which are further validated on A-OKVQA.

Training using a single model architecture. We fine-tune a single model, LLaVA-OneVision-7B on DRIVINGVQA and A-OKVQA. The selected architecture for finetuning, LLaVA-OV (visual encoder + 2-layers MLP + LLM) is highly representative of a broad family of state-of-the-art models, including Qwen-VL, InternVL, and LLaVA variants. Comprehensive ablations and diagnostic analyses (e.g., grounding accuracy, oracle comparisons) support the robustness and general validity of the reported findings.

References

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

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. 2015. VQA: Visual question answering. In *Proc. IEEE/CVF Int. Conf. Comput. Vis.*

Shivam Chandhok, Wan-Cyuan Fan, and Leonid Sigal. 2024. Response wide shut: Surprising observations in basic vision language model capabilities. *arXiv preprint arXiv:2408.06721*.

Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. 2023. Shikra: Unleashing multimodal LLMs' referential dialogue magic. *arXiv preprint arXiv:2306.15195*.

Yew Ken Chia, Vernon Toh Yan Han, Deepanway Ghosal, Lidong Bing, and Soujanya Poria. 2024. PuzzleVQA: Diagnosing multimodal reasoning challenges of language models with abstract visual patterns. *arXiv preprint arXiv:2403.13315*.

OpenAI Contributors. 2024. GPT-4 technical report. *Preprint*, arXiv:2303.08774.

Jun Gao, Yongqi Li, Ziqiang Cao, and Wenjie Li. 2024. Interleaved-modal chain-of-thought. *arXiv preprint arXiv:2411.19488*.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the V in VQA matter: Elevating the role of image understanding in visual question answering. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, and 1 others. 2024. A survey on LLM-as-a-judge. *arXiv preprint arXiv:2411.15594*.

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. 2024. HallusionBench: An advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 14375–14385.

Tanmay Gupta and Aniruddha Kembhavi. 2023. Visual programming: Compositional visual reasoning without training. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 14953–14962.

Drew A. Hudson and Christopher D. Manning. 2019. GQA: A new dataset for real-world visual reasoning and compositional question answering. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, and 1 others. 2023. Mistral 7B. *Preprint*, arXiv:2310.06825.

Chaoya Jiang, Haiyang Xu, Mengfan Dong, Jiaxing Chen, Wei Ye, Ming Yan, Qinghao Ye, Ji Zhang, Fei Huang, and Shikun Zhang. 2024. Hallucination augmented contrastive learning for multimodal large language model. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 27036–27046.

Jinkyu Kim, Anna Rohrbach, Trevor Darrell, John Canny, and Zeynep Akata. 2018. Textual explanations for self-driving vehicles. In *Proc. Eur. Conf. Comput. Vis.*

Xuanyu Lei, Zonghan Yang, Xinrui Chen, Peng Li, and Yang Liu. 2024. Scaffolding coordinates to promote vision-language coordination in large multi-modal models. *arXiv preprint arXiv:2402.12058*.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. 2024a. LLaVA-OneVision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*.

Xue Li, Yiyou Sun, Wei Cheng, Yinglun Zhu, and Haifeng Chen. 2025. Chain-of-region: Visual language models need details for diagram analysis. In *Proc. Int. Conf. Learn. Represent.*

Zhiyuan Li, Dongnan Liu, Chaoyi Zhang, Heng Wang, Tengfei Xue, and Weidong Cai. 2024b. Enhancing advanced visual reasoning ability of large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1915–1929.

Ji Lin, Hongxu Yin, Wei Ping, Pavlo Molchanov, Mohammad Shoeybi, and Song Han. 2024. ViLA: On pre-training for visual language models. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 26689–26699.

Hanchao Liu, Wenyuan Xue, Yifei Chen, Dapeng Chen, Xiutian Zhao, Ke Wang, Liping Hou, Rongjun Li, and Wei Peng. 2024a. A survey on hallucination in large vision-language models. *arXiv preprint arXiv:2402.00253*.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. 2024b. LLaVA-NeXT: Improved reasoning, OCR, and world knowledge.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024c. Visual instruction tuning. *NIPS*.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, and 1 others. 2023. Grounding DINO: Marrying DINO with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*.

Yinhong Liu, Han Zhou, Zhijiang Guo, Ehsan Shareghi, Ivan Vulić, Anna Korhonen, and Nigel Collier. 2024d. Aligning with human judgement: The role of pairwise preference in large language model evaluators. *arXiv preprint arXiv:2403.16950*.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, and 1 others. 2025a. MMBench: Is your multi-modal model an all-around player? In *Proc. Eur. Conf. Comput. Vis.*, pages 216–233. Springer.

Yuecheng Liu, Dafeng Chi, Shiguang Wu, Zhanguang Zhang, Yaochen Hu, Lingfeng Zhang, Yingxue Zhang, Shuang Wu, Tongtong Cao, Guowei Huang, and 1 others. 2025b. SpatialCoT: Advancing spatial reasoning through coordinate alignment and chain-of-thought for embodied task planning. *arXiv preprint arXiv:2501.10074*.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *NIPS*, 35:2507–2521.

Ana-Maria Marcu, Long Chen, Jan Hünemann, Alice Karnsund, Benoit Hanotte, Prajwal Chidananda, Saurabh Nair, Vijay Badrinarayanan, Alex Kendall, Jamie Shotton, Elahe Arani, and Oleg Sinavski. 2024. LingoQA: Visual question answering for autonomous driving. In *Proc. Eur. Conf. Comput. Vis.*

Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. 2019. OK-VQA: A visual question answering benchmark requiring external knowledge. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*

Chancharik Mitra, Brandon Huang, Trevor Darrell, and Roei Herzig. 2024. Compositional chain-of-thought prompting for large multimodal models. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 14420–14431.

OpenAI. 2024. GPT-4o system card.

OpenAI. 2025. OpenAI o3 system card.

Övgü Özdemir and Erdem Akagündüz. 2024. Enhancing visual question answering through question-driven image captions as prompts. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 1562–1571.

Ji Qi, Ming Ding, Weihan Wang, Yushi Bai, Qingsong Lv, Wenyi Hong, Bin Xu, Lei Hou, Juanzi Li, Yuxiao Dong, and 1 others. 2024. Cogcom: Train large vision-language models diving into details through chain of manipulations. *arXiv preprint arXiv:2402.04236*.

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

Yuxuan Qiao, Haodong Duan, Xinyu Fang, Junming Yang, Lin Chen, Songyang Zhang, Jiaqi Wang, Dahua Lin, and Kai Chen. 2025. PRISM: A framework for decoupling and assessing the capabilities of VLMs. *NIPS*, 37:111863–111898.

Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. 2022. A-OKVQA: A benchmark for visual question answering using world knowledge. In *Proc. Eur. Conf. Comput. Vis.*

Hao Shao, Shengju Qian, Han Xiao, Guanglu Song, Zhuofan Zong, Letian Wang, Yu Liu, and Hongsheng Li. 2024. Visual CoT: Advancing multi-modal language models with a comprehensive dataset and benchmark for chain-of-thought reasoning. *arXiv preprint arXiv:2403.16999*.

Chonghao Sima, Katrin Renz, Kashyap Chitta, Li Chen, Hanxue Zhang, Chengen Xie, Ping Luo, Andreas Geiger, and Hongyang Li. 2024. DriveLM: Driving with graph visual question answering. In *Proc. Eur. Conf. Comput. Vis.*

Zayne Sprague, Fangcong Yin, Juan Diego Rodriguez, Dongwei Jiang, Manya Wadhwa, Prasann Singhal, Xinyu Zhao, Xi Ye, Kyle Mahowald, and Greg Durrett. 2024. To CoT or not to CoT? chain-of-thought helps mainly on math and symbolic reasoning. *arXiv preprint arXiv:2409.12183*.

Dídac Surís, Sachit Menon, and Carl Vondrick. 2023. ViperGPT: Visual inference via Python execution for reasoning. In *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, pages 11888–11898.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, and 1 others. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, and 1 others. 2024. Qwen2-VL: Enhancing vision-language models' perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Adv. Neural Inf. Process. Syst.*

An Yan, Zhengyuan Yang, Junda Wu, Wanrong Zhu, Jianwei Yang, Linjie Li, Kevin Lin, Jianfeng Wang, Julian McAuley, Jianfeng Gao, and 1 others. 2024. List items one by one: A new data source and learning paradigm for multimodal LLMs. *arXiv preprint arXiv:2404.16375*.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, and 1 others. 2024. Qwen2 technical report. *Preprint*, arXiv:2407.10671.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, and 1 others. 2024. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. 2023. Sigmoid loss for language image pre-training. In *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, pages 11975–11986.

Zhuosheng Zhang, Aston Zhang, Mu Li, George Karypis, Alex Smola, and 1 others. 2024a. Multimodal chain-of-thought reasoning in language models. *Transactions on Machine Learning Research*.

Zhuosheng Zhang, Aston Zhang, Mu Li, Hai Zhao, George Karypis, and Alex Smola. 2024b. Multimodal chain-of-thought reasoning in language models. *Trans. Mach. Learn. Res.*

Ge Zheng, Bin Yang, Jiajin Tang, Hong-Yu Zhou, and Sibei Yang. 2023a. DDCoT: Duty-distinct chain-of-thought prompting for multimodal reasoning in language models. *NIPS*, 36:5168–5191.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023b. Judging LLM-as-a-judge with MT-bench and chatbot arena. *NIPS*, 36:46595–46623.

A Appendix Overview

This supplementary material contains the following sections:

- A description of our pipeline to annotate relevant entities (Appendix B.2);
- A description of our pipeline to generate the interleaved explanations in our dataset (Appendix B.1);
- Examples showcasing representative samples and interleaved explanations (Appendix C);
- Implementation details and hyperparameter configurations for model training (Appendix D.1);
- LLM-as-a-judge prompt for evaluation of reasoning correctness (Appendix D.3);
- Examples comparing various model outputs, showcasing the strengths and limits of our methods (Appendix D.4);

B Dataset Specifications

This section details the methodologies employed for the generation of interleaved explanations and the annotations of relevant entities.

B.1 Pipeline for Generating Interleaved Explanations

As described in Section 3.2, for each visual question in our dataset, human experts use the explanations to identify and localize the key entities in the image that are required for answering the question. This leads to an average of 1.5 entities per image, and up to 5 entities. Then, as explained in Section 3.3, we use GPT-4o to match this list of relevant entities back with the explanation to obtain an interleaved explanation. In practice, we want to interleave each key entity – whether it is the entity label, its bounding box coordinates, or the corresponding image tokens – inside the explanation. In the rest of the section, we represent the interleaved explanation with bounding box coordinates appended next to the name of the entity referred to in the explanation.

To generate the interleaved explanations, we employ a strategy that combines few-shot prompting and cleaning heuristics.

Initial interleaved explanation generation. First, we feed GPT4o with each sample: the image, question, list of options, explanation, and the list of manually annotated entities along with bounding box coordinates. Following two demonstrations and a strict set of instructions, it is asked to interleave the bounding boxes of the entities inside the explanation, with minimal change to the explanation. Here, we provide the full instructions and the two hand-crafted demonstrations fed to the model.

System prompt

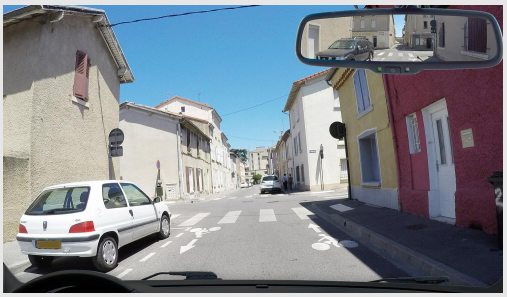
You are an expert at driving theory. You are tasked with helping a student answer questions about driving scenes.

Instructions

You are given an image of the driving theory test, a question about this image, a list of options, and the reasoning leading to the correct answer. I also give you a list of entities along with bounding box coordinates, that are relevant to answering the question. I need you to:

1. Convert the original reasoning into a clear, step-by-step reasoning that makes use of all entities in the list.
2. If the original reasoning does not mention one of the entities at all, add a short sentence referencing that entity and link it with the reasoning steps.
3. Refer to each relevant entity in the format: **entity_name** [x1, x2, y1, y2]. Replicate bounding box coordinates exactly as provided in the list.
4. Keep all other wording as close to the original reasoning as possible.
5. The entities should always be mentioned at the beginning of the sentences.

Demonstration 1



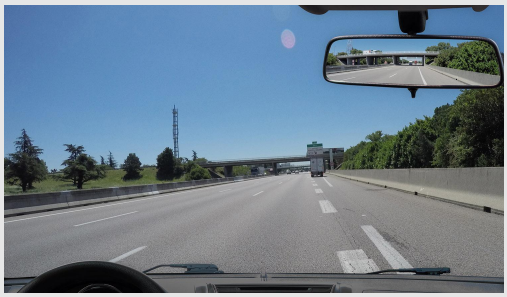
Question: ["I am turning right at the next intersection. My attention is focused on: pedestrians", "I am turning right at the next intersection. My attention is focused on: cyclists"]
Options: {A: Yes, B: No, C: Yes, D: No}
Answers: A, C
Reasoning: Before turning right, I will approach a crosswalk. My attention is therefore on the arrival of pedestrians. As I am alongside a bike lane, I will check my right blind spot before turning my wheels to ensure that no cyclist is next to me.
Entities:

- **pedestrian crossing**: [457.51, 595.61, 499.56, 62.67]
- **bike lane marking**: [893.4, 597.64, 196.2, 202.08]

Interleaved Reasoning

Before turning right at the next intersection, I will approach a **pedestrian crossing** [359.5, 337.11, 259.48, 22.09]. My attention is therefore on the arrival of pedestrians. As I am alongside a **bike lane marking** [893.4, 597.64, 196.2, 202.08], I will check my right blind spot before turning my wheels to ensure that no cyclist is next to me.

Demonstration 2



Question: By taking this exit
Options: {A: I can slow down right now, B: I must wait to brake}
Answers: A
Reasoning: A deceleration lane allows me to exit without disrupting the flow of traffic behind me. The vehicle behind is far away, so I can start slowing down right now.
Entities:

- **deceleration lane:** [933.62, 522.54, 537.35, 249.17]
- **exit sign:** [904.6, 413.47, 57.03, 36.02]
- **vehicle:** [1206.9, 156.54, 44.8, 28.56]
- **rear-view mirror:** [952.52, 82.74, 544.27, 181.17]

Interleaved Reasoning

A **deceleration lane** [933.62, 522.54, 537.35, 249.17] allows me to exit without disrupting the flow of traffic behind me. An **exit sign** [904.6, 413.47, 57.03, 36.02] indicates the upcoming exit. The **vehicle** [1206.9, 156.54, 44.8, 28.56] behind is far away, as I can see in the **rear-view mirror** [952.52, 82.74, 544.27, 181.17]. So I can start slowing down right now.

Explanations filtering and cleaning. We clean the generated interleaved explanations using regular expressions and heuristics.

- We match the bounding box coordinates in the generated interleaved explanation with the ones in the list of entities provided, correcting minor variations due to the model failing to exactly replicate the set of coordinates.
- We remove any hallucinated set of coordinates, that is absent from the provided list of entities.
- When an annotated bounding box was used twice in the interleaved explanation with different entity labels; if we can automatically identify the correct entity label, we remove the duplicated bounding box coordinates. Otherwise, we keep only the first occurrence of the set of coordinates.

B.2 Pipeline for Annotating Relevant Entities

To enrich the collected dataset with relevant entities for each sample, we developed a semi-automated entity extraction and localization pipeline that helped to accelerate the annotation process. This pipeline comprises three steps, detailed below.

Initial entity extraction. We define a taxonomy of potential entities commonly encountered in driving scenarios, organized into six groups (see Table 4). This taxonomy includes categories such as road signs, road markings, vehicles, people and other objects. Given this taxonomy, a multi-modal language model, such as GPT-4o-mini, is prompted with instructions to extract for each sample an initial list of entities relevant to the human-expert explanation. The prompt also integrates textual cues from questions, possible answers, and correct answer along with the associated image. The prompt, illustrated in Fig. 6, is designed to guide the model to prioritize visible and contextually significant entities. The extracted entities are returned in a structured list format, e.g., [`cyclist`, `pedestrian crossing`, `solid line`].

Relevant entities localization. After generating the list of relevant entities, we use a pre-trained object detection model, such as GroundingDINO (Liu et al., 2023), to localize these entities within the images. This step provides a bounding box for each detected entity. Detected entities undergo refinement such as grouping semantically similar labels under a unified category and filtering out irrelevant or erroneous detections that deviate from the predefined taxonomy.

Manual validation. The output of the automated pipeline is validated and refined by human annotators. Irrelevant entities are removed, inaccurate labels are corrected, and missing entities are added to ensure dataset consistency and accuracy.

Category	Entities
Road Signs	speed limit sign, end of speed limit sign, yield sign, directional sign, stop sign, intersection sign, mandatory right turn sign, mandatory left turn sign, mandatory straight ahead sign, no entry sign, no right turn sign, no left turn sign, no u-turn sign, no overtaking sign, end of overtaking prohibition sign, danger sign, priority sign, exit sign, dead end road sign, merge lane sign, level crossing sign, two-way traffic sign, emergency phone sign, handicapped accessible sign, parking prohibition sign, end of restrictions sign, dimension restriction sign, road narrowing sign, one-way street sign, construction detour sign, pedestrian crossing sign, pedestrian underpass sign, school crossing sign, town entry sign, town exit sign, direction sign, wild animal crossing sign, construction sign, toll road sign, weather-related sign, camping zone sign, chevron alignment marker
Road Markings	solid line, dashed line, pedestrian crossing, directional arrow, merge lane arrow, bike lane marking, stop line marking, loading zone line marking, traffic cones, temporary barrier
Road Features	speed bump, roundabout, tunnel, bridge, construction zone, accident, emergency phone, toll booth, parking lane, bus lane, bus stop area, bicycle lane, emergency lane, entry lane, exit lane
Vehicles	car, truck, motorcycle, bus, cyclist, van, motorhome, agricultural vehicle, public service vehicle, emergency vehicle
People and Animals	pedestrian, police officer, construction worker, horse rider, animal
Vehicle Parts	rear-view mirror, side-view mirror, turn signals, brake lights

Table 4: Categorized taxonomy of relevant entities for driving scenarios.

You are a driving theory expert, and your role is to extract entities from a driving scenario. These entities will be passed to an object detector for recognition.

All the possible entities are:

- road signs: (See listed entities in Table 4)
- road markings: (See listed entities in Table 4)
- road features: (See listed entities in Table 4)
- vehicles: (See listed entities in Table 4)
- people and animals: (See listed entities in Table 4)
- vehicle parts: (See listed entities in Table 4)

Instructions
Extract all entities from the scenes that are relevant to the following explanation and return them as a list. The output format should be only a list of entities, such as `[cyclist, oncoming vehicle, solid line, pedestrian crossing]`. Prioritize visible signs, markings, and vehicles directly affecting the scenario. If present in the image, always include any rear-view mirror or side-view mirror. For help, you can also refer to the questions, possible answers and true answer below, as well as the provided image attached.

Question(s): `<questions_text>`
Possible answers: `<answers_text>`
Correct answer: `<correct_answer_letters>`
Explanation: `<explanation_text>`

Figure 6: Prompt for relevant entity extraction.

C Examples

This section presents four representative examples from DRIVINGVQA dataset. Each example includes an egocentric image, one or two questions, a set of 2 to 4 answer options, the correct answers, a list of entities critical for answering the questions, and a reasoning explanation interleaved with the relevant entities. We also provide the original non-interleaved reasoning to enable a direct comparison.

The first two examples showcase where different entities have the same label, making the matching more challenging. Note that these two examples have question pairs, the first one associated with answer choices A and B, the second one having answer choices C and D.

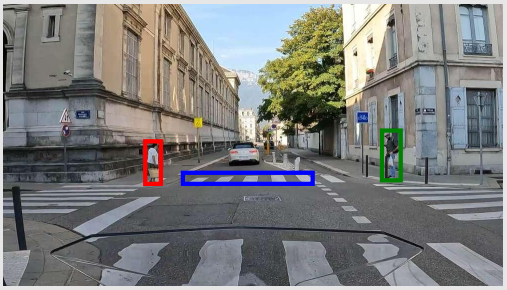
Augmented Dataset Example 1

Question: ["Can I drive at 50 km/h:", "70 km/h"]
Options: {A: Yes, B: No, C: Yes, D: No}
Answers: A, C
Entities:

- **speed limit sign**: [429.54, 50.63, 38.06, 35.67]
- **speed limit sign**: [431.53, 86.7, 33.9, 49.8]

Reasoning: The 2 speed limit signs are independent. The lower one limits the speed to 50 km/h only for vehicles designated for the transport of goods, as indicated by the category sign. The 70 km/h speed limit sign applies to all other categories of vehicles. I am not driving a goods transport vehicle, so I can drive at any speed not exceeding 70 km/h.
Interleaved Reasoning: The two speed limit signs are independent. The lower **speed limit sign** [429.54, 50.63, 38.06, 35.67] limits the speed to 50 km/h only for vehicles designated for the transport of goods, as indicated by the category sign. The **speed limit sign** [431.53, 86.7, 33.9, 49.8] applies to all other categories of vehicles. I am not driving a goods transport vehicle, so I can drive at any speed not exceeding 70 km/h.

Augmented Dataset Example 2



Question: ["I monitor the pedestrian's intention:", "I prepare to stop:"]
Options: {A: from the left, B: from the right, C: Yes, D: No}
Answers: A, B, C
Entities:

- **pedestrian**: [284.17, 274.3, 31.37, 84.98]
- **pedestrian crossing**: [359.5, 337.11, 259.48, 22.09]
- **pedestrian**: [757.46, 252.83, 37.6, 99.3]

Reasoning: The pedestrian on the left is very close to the crosswalk and is therefore preparing to cross. The pedestrian on the right is leaning over his phone and his attention is significantly diminished, so I am also monitoring him. I prepare to stop to let these two pedestrians cross.
Interleaved Reasoning: The **pedestrian** [284.17, 274.3, 31.37, 84.98] on the left is very close to the **pedestrian crossing** [359.5, 337.11, 259.48, 22.09] and is therefore preparing to cross. The **pedestrian** [757.46, 252.83, 37.6, 99.3] on the right is leaning over his phone, and his attention is significantly diminished, so I am also monitoring him. I prepare to stop to let these two pedestrians cross.

The following example shows a case where the explanation had to be modified, adding an extra sentence to include the relevant entity *dashed line* that was manually annotated by the human experts but wasn't mentioned in the original explanation.

Augemented Dataset Example 3

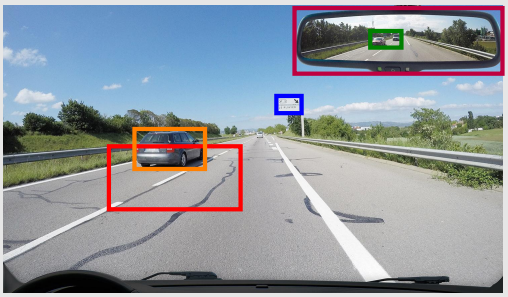
Question: Do the tradespeople run a risk if they park their van in the same way these police vans are parked?
Options: {A: Yes, B: No}
Answer: A
Entities:

- **police vehicle**: [372.0, 235.7, 1562.24, 1043.57]
- **bus lane**: [14.46, 755.5, 393.74, 996.79]
- **dashed line**: [320.34, 747.66, 214.84, 998.63]

Reasoning: These vans are not parked properly: they overflow so much from the spaces that road users have to move onto the bus lane, which is reserved for buses. We advise an average tradesperson against parking their van in this manner.
Interleaved Reasoning: These **police vehicles** [372.0, 235.7, 1562.24, 1043.57] are not parked properly: they overflow so much from the spaces that road users have to move onto the **bus lane** [14.46, 755.5, 393.74, 996.79], which is reserved for buses. The **dashed line** [320.34, 747.66, 214.84, 998.63] indicates the boundary of the lane. We advise an average tradesperson against parking their van in this manner.

This final example shows a case with 5 annotated relevant entities, which is the maximal number that can be found in our dataset.

Augmented Dataset Example 4



Question: ["The vehicle in front can still take the next exit", "I can follow this vehicle to overtake"]
Options: {A: Yes, B: No, C: Yes, D: No}
Answers: B, D
Entities:

- **dashed line**: [317.36, 423.1, 394.94, 188.64]
- **exit sign**: [819.58, 272.31, 75.71, 51.58]
- **car**: [1101.09, 77.8, 93.61, 49.79]
- **rear-view mirror**: [873.99, 9.21, 623.74, 196.48]
- **car**: [394.01, 371.19, 211.56, 119.92]

Reasoning: The marking in the middle of the two lanes of traffic only prohibits vehicles in the left lane from merging to take the exit. However, vehicles are approaching from behind, so I must give up overtaking.
Interleaved Reasoning: The **dashed line** [317.36, 423.1, 394.94, 188.64] in the middle of the two lanes of traffic only prohibits **vehicles in the left lane** [394.01, 371.19, 211.56, 119.92] from merging to take the exit indicated by the **exit sign** [819.58, 272.31, 75.71, 51.58]. However, vehicles [**1101.09, 77.8, 93.61, 49.79**] are approaching from behind, as seen in the **rear-view mirror** [873.99, 9.21, 623.74, 196.48], so I must give up overtaking.

D Experiments Details

D.1 Implementation Details

We fine-tune various LLaVA-OneVision models using different prompt formats on the DRIVINGVQA train set. All models are initialized from the pre-trained weights obtained after the final OneVision training stage.

The fine-tuning experiments are conducted using 2 NVIDIA A100 GPUs, except for experiments involving visual patches ('V') where 4 NVIDIA A100 GPUs were required. We employ the following configuration and hyper-parameters:

Table 5: Hyper-parameter configuration for fine-tuning LLaVA-OneVision on the DRIVINGVQA dataset

Hyperparameter	Value
Epochs	10
Batch Size	2
Max Seq. Length	32768
Vision Tower	SigLIP-SO400M-Patch14-384
Language Model	Qwen2-7B
Projector	MLP2x_GELU
Trainable Parts	ϕ_{vision}, θ_{proj}, θ_{LLM}
Learning Rate θ_{proj}, θ_{LLM}	$1e^{-5}$
Learning Rate ϕ_{vision}	$2e^{-6}$
Weight Decay	0
LR Scheduler	cosine
Warm-Up Ratio	0.03

D.2 RIV-CoT diagram

See Figure 7.

D.3 Model reasoning evaluation

See Figure 8.

D.4 Result Output Examples

In this section, we present test samples alongside selected model outputs to highlight their capabilities and their limitations. For each sample, we provide the question and answer choices based on the QP-EA prompt (blue box), followed by the reasoning and answers from two models, along with their prompt-specific formats where applicable (red boxes).

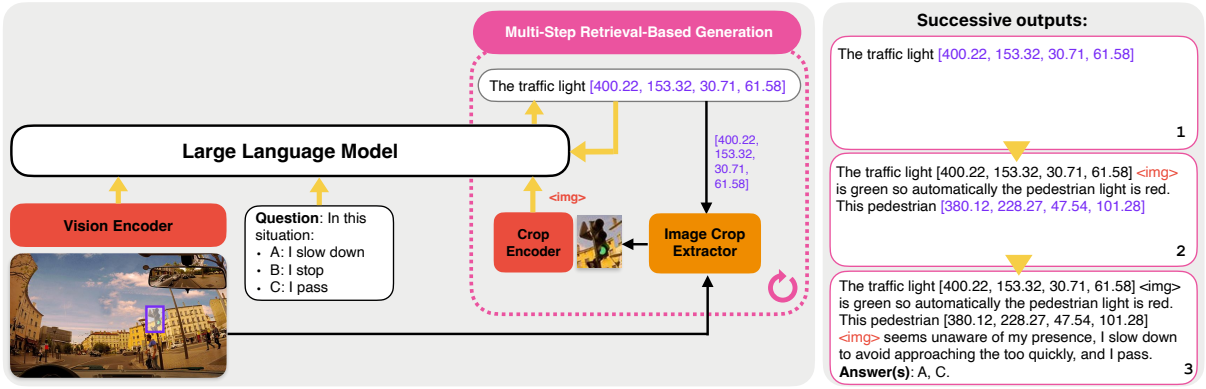


Figure 7: **Illustration of multi-step retrieval-based generation.** During inference, starting with a tokenized question and an image tokenized by the adapter on the output of the Vision Encoder, the Large Language Model generates output until it predicts a bounding box. At this point, the generation process pauses to extract the corresponding image crop based on the predicted coordinates. The image crop is encoded and adapted into an image crop token, which is then added back into the model's context along with the question, image tokens and previously generated outputs. This iterative process continues until the model produces its final answer.

Question Example 1

Human:
Unless explicitly stated otherwise, assume you are driving a car in France.
Select all correct answers to the following question from the available options. Detail your reasoning step by step based on road signs, markings, signals, and relevant driving rules. Then, provide the letters corresponding to your answer in the format: 'Answer(s): <letters>'
Question: To go straight, can I stay in this lane?
Options: {A: Yes, B No.}

Question Output 1: LLaVA-OV fine-tuned with CoT

Model:
Reasoning: If I want to go straight, the road markings indicate that I am correctly positioned, as I can either continue straight or turn right. The left lane is the lane for turning left. Therefore, I will not need to change lanes.
Answer(s): A.

Question Output 2: LLaVA-OV fine-tuned with CoT and oracle relevant entities

Human:
The relevant entities for this problem are:
directional arrow  .
Model:
Reasoning: The lane I am in is intended for users wishing to turn right. This is indicated by the directional arrow on the road markings. Therefore, I cannot stay in this lane if my intention is to go straight.
Answer(s): B.

In Example 1 above, the simple chain-of-thought model acknowledges the existence of a directional arrow as road marking, but incorrectly hallucinates its direction, leading to the wrong answer. The "Oracle" model QPRV-EA, being fed with an image patch showing the arrow, is able to identify its direction and correctly answer the question.

The ground truth explanation is the following:
"*The road markings indicate that the right lane can only be used for turning right. Therefore, I must position myself in the left lane to go straight or to the left.*"

Question Example 2

Human:
Unless explicitly stated otherwise, assume you are driving a car in France.
Select all correct answers to the following question from the available options. Detail your reasoning step by step based on road signs, markings, signals, and relevant driving rules. Then, provide the letters corresponding to your answer in the format: 'Answer(s): <letters>'
Question: This light is flashing, I must yield to this car:
Options: {A: Yes, B No.}

Question Output 1: LLaVA-OV fine-tuned with CoT

Model:
The relevant entities for this problem are: car [0.617, 0.664, 0.754, 0.798], traffic light [0.652, 0.398, 0.679, 0.521]. Reasoning: This car is stopped at the traffic light, so it does not have the right of way. Therefore, I do not need to yield to it while waiting for the light to turn green.
Answer(s): B.

Question Output 2: LLaVA-OV fine-tuned with CoT with relevant entities (labels+bbox)

Human:
The relevant entities for this problem are: traffic light [0.682, 0.344, 0.717, 0.662], vehicle [0.623, 0.640, 0.805, 0.784], stop line marking [0.616, 0.721, 0.655, 0.804], stop sign [0.640, 0.585, 0.661, 0.628], traffic light [0.555, 0.603, 0.564, 0.635].
Model:
Reasoning: The traffic light is malfunctioning, and the vehicle that wants to turn left must stop at the stop line marking. It is not a priority, so I do not have to yield to it. The stop sign is also present.
Answer(s): B.

Question Example 3

Human:
Unless explicitly stated otherwise, assume you are driving a car in France.
Select all correct answers to the following question from the available options. Detail your reasoning step by step based on road signs, markings, signals, and relevant driving rules. Then, provide the letters corresponding to your answer in the format: 'Answer(s): <letters>'
Question: After these signs, will I be able to overtake this truck:
Options: {A: Yes, B No.}

In example 2, we compare providing the entities and bounding box coordinates as input (QPRB-EA), or letting the model detect them itself (QP-RBEA). Both models reach the same answer, but the second one's reasoning is wrong: it fails to identify a key entity (the stop line marking or the stop sign), leading to an incorrect reasoning even though the answer is correct. This highlights the importance of accurate entity detection in the reasoning process. The ground truth explanation is the following:
"*This traffic light is flashing, and there is no sign under the light. However, I can see that the vehicle on the right has a stop line marking and a stop sign. There is also another traffic light visible. Therefore, I do not have to yield to the right.*"

Question Output 1: LLaVA-OV fine-tuned with CoT

Model:
Reasoning: The left lane is a storage lane that allows for left turns. It is strictly prohibited to use this type of lane for overtaking the truck. I therefore give up on making this maneuver.
Answer(s): B.

Question Output 1: LLaVA-OV fine-tuned with CoT and relevant entities (labels+bbox+crops)

Model:
The relevant entities for this problem are: speed limit sign [0.508, 0.294, 0.547, 0.401], truck [0.367, 0.304, 0.461, 0.546].
Human:
The relevant entities for this problem are: speed limit sign  , truck  .

Model:
Reasoning: The prohibition signs must be read separately. The speed limit sign of 90 km/h applies to all road users. The second sign prohibits overtaking for vehicles transporting goods over 3.5 tons. Therefore, I will be able to overtake this truck.
Answer(s): A.

Example 3 shows a case where despite an imperfect entity detection ("speed limit sign" instead of "no overtaking sign", adding the image patches allows the model to correctly identify the important sign to answer the question. This is enabled by our cropping strategy that adds 70% of the image patch to increase the context and account for imperfect entity detection. On the contrary, the baseline QP-EA hallucinates the presence of a storage lane for turning left and answers incorrectly.

The ground truth explanation is: *This no overtaking sign prohibits goods transport vehicles from overtaking. It does not apply to me. Furthermore, I am driving on a one-way road with 2 lanes, so I can use the left lane to overtake. It is clear, as confirmed by a quick glance in the rear-view mirror. I can therefore proceed with my maneuver while ensuring that I respect the speed limit of 90 km/h.*

E Impact Statement

Impact. The benchmark and method expose critical failure modes in current VLMs, such as poor utilization of localization signals, and propose a scalable alternative to address them by using automatically extracted entities. Beyond academic benchmarks, these findings have practical implications for safety-critical applications like autonomous driving and AI-assisted training systems, where explainability and visual grounding are essential. We release all data and code to encourage reproducibility and further research. Overall, we believe the dataset's release will significantly contribute to the development of more grounded, interpretable, and trustworthy vision-language systems.

Potential risks. As DRIVINGVQA is derived from real-world driving theory tests, there exists a risk that models trained on it could be exploited to automatically answer or tutor for such exams, potentially bypassing fair evaluation standards. Being also restricted to only French data, driving theory exams are also regularly updated to reflect new traffic laws, technologies, or societal expectations. Over time, certain questions may become outdated or misaligned with updated regulations.

Ethical compliance. All data are sourced from openly available French driving theory content, freely accessible online without login. No private user data was used. The dataset adheres to the NeurIPS Code of Ethics.

Instructions

You are a strict but fair driving-theory instructor. You're given:

1. A driving theory test question
2. A list of possible answer options
3. The official "correct reasoning"
4. A student's reasoning for the same question

Your task: *Assess if the student's reasoning matches the correct reasoning.*

Step-by-Step Instructions:

1. **Identify Student's Arguments**
   - List each key argument or step in the student's reasoning.
   - For each argument, briefly state whether it is correct or not, given the provided correct reasoning.
2. **Check for Missing or Contradictory Points**
   - Look at the official correct reasoning.
   - List **important points or steps** from the correct reasoning that the student **omits** or **directly contradicts**.
   - Minor omissions or differences in wording/style are acceptable.
3. **Decide on Overall Correctness**
   - If the student's reasoning is **mostly consistent** with the correct reasoning and has **no major factual errors**, then it is considered **correct**.
   - If the student's reasoning **contains significant logical or factual errors**, or **omits critical steps** from the correct reasoning, then mark it **incorrect**.

Important Note: The student's reasoning does not have to match the official reasoning exactly; it just needs to be conceptually equivalent and free of serious contradictions.

Final Output Format:

- Provide your step-by-step analysis.
- At the end, write: **Final Answer**: "1" if you judge the student's reasoning is overall correct, "0" if it is overall incorrect.

Figure 8: Prompt for evaluation of model reasoning.