Methodology and Benchmark for Automated Driving Theory Test of Large Language Models

Dashuai Pei, Yiwen Wu, Jianhua He, *Senior Member, IEEE*, Kezhong Liu, Mozi Chen, Xuedou Xiao, *Member, IEEE*, Shengkai Zhang, and Jiawei Zheng

***Abstract*— Large Language Models (LLMs), with their strong generalization and inference capabilities, have been increasingly leveraged to address the challenges of handling corner cases in autonomous driving (AD). However, a critical unresolved issue remains: the lack of a comprehensive understanding and formal assessment of LLMs' driving theory knowledge and practical skills. To address this issue, we propose the first dedicated driving theory test framework and benchmark for LLMs. That is a crucial yet unexplored area in the literature, particularly for safety-critical applications in autonomous driving and driver assistance. Our framework systematically evaluates LLMs' competence in driving theory and hazard perception, akin to the official UK driving theory test, ensuring their qualification for critical driving-related tasks. To facilitate rigorous benchmarking, we construct a comprehensive dataset comprising over 700 multiple-choice questions (MCQs) and 54 hazard perception video tests sourced from the official UK driving theory examination. Additionally, we incorporate two standardized MCQ sets from the UK's Driver and Vehicle Standards Agency (DVSA). For these two types of theoretical test items, we design tailored assessment methodologies and evaluation metrics, including accuracy, recall, precision, F1-score, real-time performance, and computational efficiency. The experimental results reveal that among all LLMs tested, only GPT-4o achieved an accuracy of 88. 21% in the MCQs test, successfully passing this component. However, in hazard perception testing, none of the evaluated models met the passing criteria under the given settings, highlighting the substantial improvements required before these models can be practically deployed for real-world driving applications. Our key insight is that the specific test questions LLMs fail to answer correctly directly reflect their deficiencies in understanding and flexibly applying traffic regulations, as well as in analyzing and responding to complex driving scenarios. This provides clear directions for future improvements.**

***Index Terms*— Autonomous driving, large language model, driving theory test, hazard perception test, remote driving, mobile computing.**

I. Introduction

THE most recent data released by the World Health Organization indicates that approximately 1.3 million individuals die each year due to road traffic accidents, with an additional 50 million sustaining injuries [1]. Autonmous driving is widely viewed as a strong candidate to reduce road accidents and improve road safety. Research from the University of Michigan demonstrates that the implementation of Advanced Driver Assistance Systems (ADAS) can prevent 20% to 46% of such accidents. Additional predictions estimate that the widespread adoption of ADAS in Europe could decrease road traffic accidents by approximately 15% by 2030 [2]. Significant advancements in autonomous driving have been achieved in recent years, but existing solutions face major challenges of handling long-tail corner cases and generalization, which prevent autonomous driving from large scale deployment on the roads [3], [4]. The “long-tail problem” refers to numerous low-frequency, extreme, rare, or unforeseen situations that may arise in actual driving environments. Although these situations occur infrequently, they must be effectively handled, as autonomous driving systems need to ensure safe operation under all potential circumstances [5], [6]. Furthermore, current AD algorithms primarily rely on AI-guided decision-making processes, which suffer from limited transparency and interpretability. This insufficiency does not meet the stringent requirements of traffic safety regulations. With their extensive world knowledge and powerful reasoning abilities LLMs can help AD systems handle long-tail issues and enhance decision interpretability by generating complex scenario descriptions and natural language explanations. For example, Fu et al. address the long-tail problem in AD by utilizing GPT-3.5 that mimics human-like reasoning, interpretation, and memorization to navigate complex scenarios and

Received 24 October 2024; revised 8 March 2025 and 8 May 2025; accepted 14 May 2025. This work was supported in part by the European Union's Horizon 2020 Research and Innovation Programme under the Marie Skłodowska-Curie Actions (MSCA) under Grant 824019 and Grant 101022280; in part by the Horizon Europe MSCA Programme under Grant 101086228; in part by the U.K. Research and Innovation (UKRI) through the Engineering and Physical Sciences Research Council (EPSRC) under Grant EP/Y027787/1; in part by the UKRI EPSRC Researcher Co-Investigator Programme (RCP) under Grant 15831/DCM4480; in part by the National Natural Science Foundation of China (NSFC) under Grant 52401423 and Grant 62402353; in part by the Postdoctoral Science Foundation of Hubei Province under Grant 2024HBB079; and in part by the Fundamental Research Funds for the Central Universities WUT under Grant 104972025KFYjc0058. The Associate Editor for this article was N. Bekiaris-Liberis. *(Corresponding authors: Yiwen Wu; Jianhua He; Kezhong Liu.)*

Dashuai Pei and Xuedou Xiao are with the School of Navigation, Wuhan University of Technology (WUT), Wuhan 430062, China (e-mail: Pei.Dashuai@whut.edu.cn; xuedouxiao@whut.edu.cn).

Yiwen Wu is with the School of Computer Science, University of Electronic Science and Technology of China, Chengdu 611731, China (e-mail: YiwenWu@ieee.org).

Jianhua He is with the School of Computer Science and Electronic Engineering, University of Essex, CO4 3SQ Colchester, U.K. (e-mail: j.he@essex.ac.uk).

Kezhong Liu and Mozi Chen are with the School of Navigation and the State Key Laboratory of Maritime Technology and Safety, Wuhan University of Technology (WUT), Wuhan 430062, China (e-mail: kzliu@whut.edu.cn; chenmz@whut.edu.cn).

Shengkai Zhang is with the School of Information Engineering and the State Key Laboratory of Maritime Technology and Safety, Wuhan University of Technology (WUT), Wuhan 430062, China (e-mail: Shengkai@whut.edu.cn).

Jiawei Zheng is with the School of Electronic Engineering, State Key Laboratory of Information Photonics and Optical Communications, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: 2238486002@qq.com).

Digital Object Identifier 10.1109/TITS.2025.3571213

accumulate driving experience [7]. And Mao et al. proposed an approach that transforms the OpenAI GPT-3.5 model into a reliable motion planner for autonomous vehicles [8]. While there are increasing interests in applying LLMs for AD, they still face several major technical challenges, such as limited on-boad computing resources to run the LLMs, and a lack of comprehensive understanding and formal test of LLM's driving theoretic knowledge and skills. Generally, LLMs uses a huge number of parameters to effectively capture complex patterns and knowledge in the training data, e.g., DeepSeek-V3 has 671B parameters [9], [10]. These models require substantial computation and memory resources due to their large parameter sizes, which are difficult to accommodate by the onboard devices of autonomous vehicles. Furthermore, AD systems are expected to produce higher safety standards than human driving. Although human drivers must pass theoretical and practical road tests to qualify for driving on public roads, no rigorous evaluations have been reported on the LLMs used for AD. According to a recent report, the safety performance of LLM-empowered autonomous driving is significantly lower than that of human drivers [11]. Therefore, critical questions remain unanswered: 1) Are LLMs fundamentally qualified to be used in autonomous driving and driving assistance? 2) To what extent can we trust their decision making based on standardized driving knowledge assessments?

Considering the research potential and challenges of LLM in autonomous driving and driving methodology and benchmark of driving theory inspired by the official tests required for human drivers to obtain a license. Our framework, modeled after the UK Driver & Vehicle Standards Agency (DVSA) theory test, assesses both the driving knowledge and the hazard perception skills of LLMs. This evaluation serves as a foundational step in determining whether LLMs are fundamentally qualified for autonomous driving or assistance tasks. While strong performance on driving theory tests does not guarantee effective real-world decision-making, standardized assessments offer a structured framework for evaluating the reliability and limitations of LLMs. We cannot simply assume that future LLMs will achieve human-level generalization. However, analyzing their failures on specific test questions provides valuable insights into deficiencies in understanding traffic rules and responding to complex scenarios. These failure patterns highlight critical areas for improvement, guiding future advancements in LLM-based driving systems. An LLM capable of passing driving theory tests can be deployed on remote cloud servers, allowing CAVs to interact with it via vehicle-to-everything (V2X) communication technologies. As shown in Fig. 1, CAVs can engage with LLM hosted in roadside units (RSU) through direct vehicle-to-vehicle (V2V) communication or through vehicle-to-infrastructure (V2I) links connecting to cellations and RSUs. Alternatively, CAVs can establish communication with LLMs deployed in remote cloud environments through V2I connections to cellular base stations and the Internet, facilitating seamless access to advanced decision-making and reasoning capabilities.

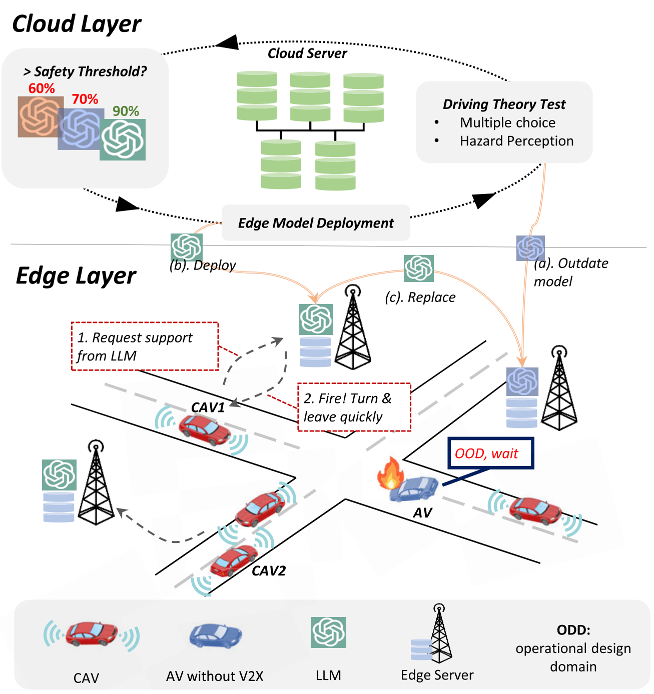


Fig. 1. Remote or local deployment of LLMs for V2X-assisted decision-making in autonomous driving. LLMs must pass driving theory tests before being utilized for real-time decision support. In the depicted scenario, an autonomous vehicle encounters an out-of-operational-design-domain (OOD) situation due to a fire hazard and comes to a stop. Nearby CAV1 request driving assistance from an LLM deployed either at a RSU or in the cloud via V2X communication. The LLM provides an immediate response instructing CAV1 to turn and leave quickly, ensuring safe and efficient navigation while mitigating further risks.

In this paper, we design and run driving theory tests for 11 proprietary LLM models (OpenAI GPT, Google Gemini, Anthropic Claude, Baidu Ernie, Zhipu glm, and Ali QWen) and 5 open source LLM models (DeepSeek-V3, Deepseek-R1, Tsinghua MiniCPM3-4B, MiniCPM-V-2.6, and MiniCPMLlama3-V2.5). We have not identified any open-source LLM specifically adapted for driving tasks. Although [12] developed the EMMA based on Gemini 1.0 Nano, it remains a closed source. To ensure a comprehensive evaluation, we included Google's Gemini series models (Gemini-1.5-Pro and Gemini-2.0-Flash) in our evaluation. The accuracy, precision, recall of the model, the F1 score, the confusion matrix, the computational efficiency and the real-time performance are measured from the experiments. Our benchmark dataset includes over 700 MCQs and 54 video hazard perception tests derived from the UK official driving theory test. Additionally, we incorporate two standardized 50 MCQ sets from the UK DVSA. This testing framework allows us to systematically evaluate the ability and effectiveness of LLMs in handling driving-related tasks, which is crucial to improve the safety of CAVs and driving assistance systems. With continuous technological advancements, further research into LLM performance in real driving environments and their potential improvements will provide valuable insights for the development of autonomous driving technology. The experimental results demonstrate that GPT-4o achieved an average accuracy of 88.21%, and Claude-3.5-sonnet attained 85.21%, both passing the multiple-choice test, whereas the other models did not. In the hazard perception test, GPT-4o and GPT-4o-mini were evaluated. The results indicated that the precision rate for detecting hazard events was below 20%, and the final score rate was below 60%, with a relatively high false positive rate, thus neither passed the test.

This study reveals the actual capabilities of current LLMs when confronted with human driving theory tests, indicating that mainstream LLMs may not yet be fully capable of handling all AD tasks. The findings of this research contribute to decision making about how to integrate LLMs into CAV applications while balancing model performance and cost. The theoretical test methodology and framework are also applicable to AV driving assistance systems utilizing locally deployed LLMs. They are complementary to the existing research works on the design and applications of LLMs for automous driving algorithms.[1]

II. Related Work

Recent research in AD has explored the application of LLMs through modular AD pipelines and end-to-end AD systems. LLMs in modular pipelines improve various stages, such as perception of the environment, localization, path planning, and decision making. Conversely, end-to-end systems leverage LLMs to directly generate control commands from sensor inputs, simplifying intermediate steps and enhancing adaptability.

A. LLM Applications in Modular Autonomous Driving Pipelines

In modular AD pipelines, various independent functional modules make up the entire AD system. LLMs have been innovatively applied in these modules. Sha et al. developed a modular autonomous driving pipeline using LLMs for high-level decision-making in complex driving scenarios, converting these decisions into mathematical instructions for Model Predictive Control (MPC) [14]. Their chain-of-thought framework enables precise vehicle control based on textual input, significantly improving performance in both single-vehicle tasks and complex multi-vehicle coordination, with quantitative experiments showing a reduction in overall cost compared to existing methods. Azarafza et al. developed a hybrid reasoning framework for AD, integrating LLMs into modular pipelines for decision-making within the CARLA simulator [15]. By combining object detection data with arithmetic and commonsense reasoning, the system calculates precise control signals for vehicle actions under varying meteorological conditions, demonstrating effective decision-making in complex scenarios. Liu et al. proposed a multi-task decision-making GPT model for AD, addressing multi-task decision-making at unsignalized intersections as a sequence modeling problem using GPT-2 [16]. They designed a training pipeline where expert models trained via reinforcement learning guide the MTD-GPT model, which shows superior or comparable performance to state-of-the-art single-task reinforcement learning models across various decision-making tasks. Wang et al. developed an LLM-based "Co-Pilot" system for human-machine co-driving, where the Co-Pilot adjusts vehicle operations based on human intentions conveyed through prompts [17]. They proposed a universal framework with a memory module for organizing task-relevant information and demonstrated the system's effectiveness in path tracking and trajectory planning tasks, showing that the Co-Pilot can align vehicle control with driver intentions without extensive numeric calculations.

B. LLM Applications in End-to-End AD Systems

End-to-end AD systems directly generated control commands from sensor inputs, simplifying the intermediate steps of traditional AD systems. Shao et al. proposed LMDrive, an end-to-end, instruction-following MLLM model for AD, which processed camera-LiDAR sensor data and natural language instructions to generate vehicle control signals [18]. They introduced a novel dataset and evaluation benchmark to train and test the model in realistic driving scenarios, incorporating complex and diverse language instructions, including misleading or safety-violating commands. Xu et al. presented DriveGPT4, an interpretable end-to-end AD system that used MLLMs to predict vehicle control signals from video input and generate natural language explanations of its decisions [19]. The system was fine-tuned on a new visual instruction dataset created with ChatGPT, outperforming baselines on the BDD-X dataset for various driving tasks. Sreeram et al. examined MLLMs for AD, revealing their limitations in reasoning across dynamic driving scenarios despite success with individual images [20]. They introduced DriveSim, a specialized simulator, and the Eval-LLM-Drive dataset to rigorously test MLLMs' capabilities in vehicle control, scene reasoning, and interactions with other road actors, highlighting the need for improved models in real-world driving environments. Wang et al. developed an end-to-end AD system by leveraging multimodal foundation models to improve generalization and reliability, focusing on latent feature extraction rather than explicit scene representations [21]. Their method enhanced decision-making by extracting per-patch features from transformer architectures and integrating language-based simulation, allowing dynamic feature manipulation and common-sense reasoning to augment training and debugging processes. You et al. proposed an end-to-end vehicle-infrastructure cooperative AD framework, V2X-VLM, which leveraged large vision-language models to enhance situational awareness and decision-making by fusing multi-source visual and textual data from vehicles and infrastructure [22]. Using contrastive learning for robust representation, their method improved trajectory planning in complex traffic scenarios, demonstrating superior performance on the DAIR-V2X dataset compared to state-of-the-art approaches.

C. Evaluating LLMs With Multiple-Choice Question

The rapid advancement of LLMs has necessitated robust evaluation methodologies to assess their capabilities and

[1] This paper is an extended version of a conference paper submitted to MobiArch 2024 [13], with more than 50% additional research content. The expanded sections include in-depth evaluations of LLMs in driving theory tests, extended discussions on model performance and cost considerations, and new insights into the application of LLMs for connected and CAV systems. The data and code of this study are publicly available at https://github.com/PeiDashuai/DVSA-test-for-LLMs

limitations. MCQs have become a popular and effective method for evaluating LLMs across various domains and tasks. Several researchers have developed comprehensive MCQ datasets for this purpose. Hendrycks et al. introduced the "Measuring Massive Multitask Language Understanding" (MMLU) benchmark, comprising 57 tasks across diverse subjects [23]. This benchmark has been widely adopted to assess the general knowledge and reasoning capabilities of LLMs. Similarly, Srivastava et al. proposed the BIG-bench benchmark, consisting of 204 tasks across diverse topics, to evaluate the capabilities of LLMs [24]. They found that while model performance improves with scale, it remains poor on complex tasks, and social bias tends to increase with larger models. Beyond general-purpose benchmarks, researchers have developed domain-specific MCQ evaluations to assess LLM performance in specialized areas. Cobbe et al. proposed GSM8K, a dataset of 8.5K grade school math word problems to evaluate language models' ability to perform multi-step mathematical reasoning [25]. They introduced training verifiers to select correct solutions, showing significant performance improvements compared to traditional fine-tuning approaches. In the medical domain, Pal et al. proposed MedMCQA, a large-scale dataset containing over 194k MCQs from medical entrance exams, covering 21 medical subjects and 2.4k healthcare topics [26]. The dataset is designed to test models' reasoning abilities and language understanding across diverse medical subjects and topics. Hendrycks et al. proposed the ETHICS dataset, a benchmark designed to assess LLMs' understanding of moral concepts such as justice, well-being, and commonsense morality [27]. Their findings indicate that while current models show promising abilities in predicting ethical judgments, they still fall short, providing a foundation for further aligning AI with shared human values.

Researchers have also used MCQs to probe specific cognitive capabilities of LLMs. Wei et al. proposed Chain-of-Thought prompting, a method that enhances reasoning in LLMs by generating intermediate reasoning steps [28]. Their experiments show significant performance improvements on arithmetic, commonsense, and symbolic reasoning tasks, with particularly strong results on the GSM8K math word problem benchmark.

III. Research Methodology

A. UK Driving Theory Test

In the UK driving theory test, candidates must complete two parts: a multiple-choice test and a hazard perception test. The multiple-choice section consists of 50 questions, each with 4 choices and one or more correct answers. These questions cover a wide range of topics, including road safety, traffic signs, vehicle handling, and environmental factors. Candidates are given 57 minutes to finish the test, and to pass, learner drivers must answer at least 43 out of 50 questions correctly, achieving an accuracy of 86%. This passing criterion is also applied to LLMs used for driving assistance evaluations.

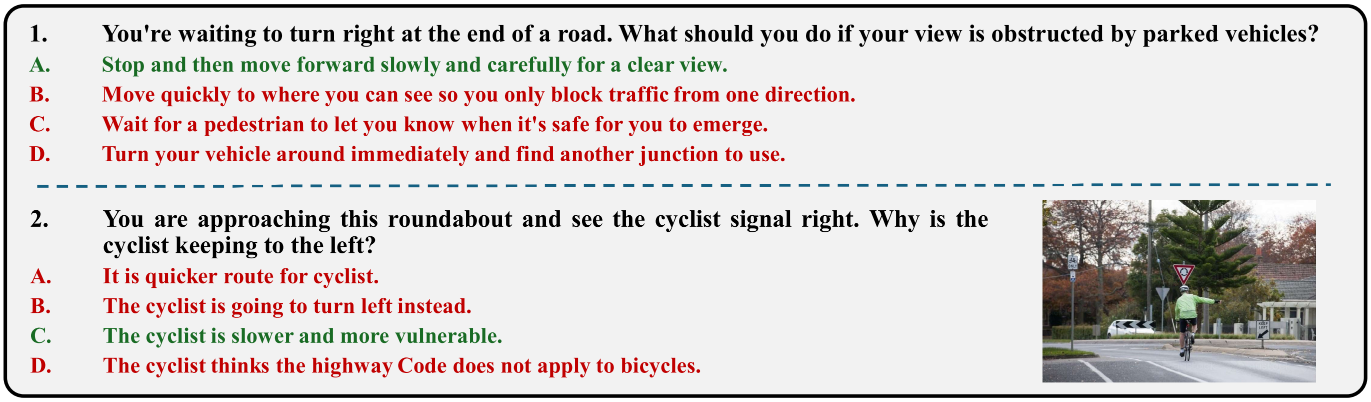
In the hazard perception test, candidates watch 14 one-minute clips (19 clips for lorry and bus candidates) filmed from the perspective of a driver or motorcyclist. The task is to identify developing hazards, which are potential dangers that require the driver to take action, such as changing speed or direction. Candidates indicate when they see a developing hazard by clicking a mouse or touching the screen. The sooner they respond after the hazard starts to develop, the higher their score. Each hazard is scored on a 5-point scale, with points decreasing the longer it takes to react. There is a maximum score of 75 for car drivers and motorcyclists and 100 for lorry and bus candidates, with a pass mark of 44 and 67 points, respectively.

B. Test Datasets

DVSA published many mock questions for MCQ test and sets of mock tests each including 50 questions [29]. These questions are close to the ones for or may appear in the official tests. Additionally, we utilized alternative test questions publicly available on the DriverInstructor website [30], which closely resemble the official DVSA theory questions. After removing duplicates, the DriverInstructor website contained 726 unique questions, of which about 109 included images of traffic scenarios and signs. For the hazard perception test, we used test materials from DVSA-published driving exam resources. We collected 54 video clips containing hazards, with official answers providing the exact start and end frames of each hazard. To optimize the clips for testing, we edited them for length while preserving the sections containing the hazards.

Given that some LLMs cannot process image input, we divided the multiple-choice questions into two datasets: one without images and one with images. The dataset without images (DS-Text) contains 617 test questions, while the dataset with images (DS-Image) includes 109 questions. Fig. 2 shows examples of two multiple-choice questions representing the DS-Text and DS-Image categories, respectively. For the hazard perception test, we converted all video clips into image sequences, segmented at 0.2-second intervals. As shown in Fig. 3, we present key frames from two hazard perception test video clips.

The ground truth data for the hazard perception test datasets are presented in two distinct formats to facilitate comprehensive evaluation: **Hazard Classification Table:** This binary classification schema assigns each frame a label of either "hazard" (denoted as 1) or "no hazard" (denoted as 0). This dichotomous approach allows for a clear delineation of hazardous and non-hazardous situations within the temporal sequence of frames. **Scoring Table:** A more nuanced evaluation mechanism is implemented through a scoring system that quantifies the timing of hazard detection. This method employs a descending numerical scale, where higher values correspond to earlier detection of hazards. For instance, a typical scoring sequence might appear as "0 0 5 5 4 4 3 3 3 2 2 1 1 0 0". This scoring paradigm serves a dual purpose: a) It provides a metric for assessing the model's efficacy in early hazard detection, which is crucial in real-world applications where prompt identification of dangers is paramount. b) It incorporates a penalty system for both missed detections (false negatives) and false alarms (false positives), thereby encouraging the development of models with high sensitivity and specificity.



1. **You're waiting to turn right at the end of a road. What should you do if your view is obstructed by parked vehicles?**
   - A. **Stop and then move forward slowly and carefully for a clear view.**
   - B. **Move quickly to where you can see so you only block traffic from one direction.**
   - C. **Wait for a pedestrian to let you know when it's safe for you to emerge.**
   - D. **Turn your vehicle around immediately and find another junction to use.**

2. **You are approaching this roundabout and see the cyclist signal right. Why is the cyclist keeping to the left?**
   - A. **It is quicker route for cyclist.**
   - B. **The cyclist is going to turn left instead.**
   - C. **The cyclist is slower and more vulnerable.**
   - D. **The cyclist thinks the highway Code does not apply to bicycles.**

Fig. 2. Example of two multiple-choice questions from the driving theory test. The first question is from the DS-Text dataset, while the second is from the DS-Image dataset. Correct answers are highlighted in green, while incorrect choices are marked in red.

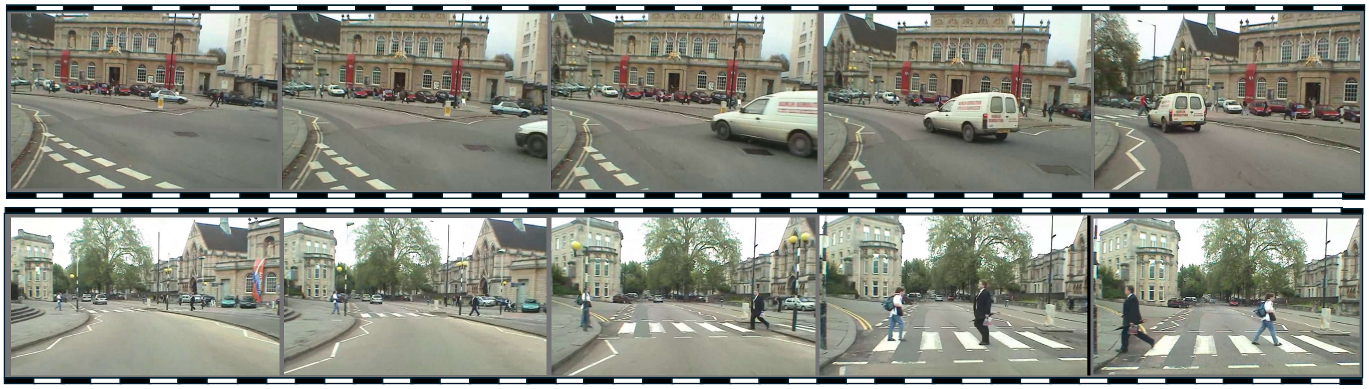


Fig. 3. Key frames from two hazard perception test video clips. (Top row) The first set of frames shows the vehicle approaching a junction from a side road, preparing to merge into a main road. The vehicle halts to allow a white van traveling on the main road to pass before merging. (Bottom row) The second set of frames depicts the vehicle stopped in front of a pedestrian crossing, waiting for pedestrians to safely cross the road on the zebra crossing.

This dual-format approach to ground truth data allows for a more comprehensive and rigorous evaluation of hazard detection models, encompassing both binary classification accuracy and the temporal precision of hazard identification.

C. LLMs Used in Theory Test

There are several powerful LLMs from leading companies such as Anthropic, OpenAI, Ali, ZhiPu, and Baidu. For the driving theory test in this paper, we selected proprietary LLMs from OpenAI, Ali, and Baidu due to their superior performance.

1) OpenAI and Anthropic LLM Models: The four selected OpenAI models—GPT-3.5-turbo, GPT-4, GPT-4o and GPT-4o-mini have different capabilities and price points. GPT-3.5-turbo is a fast and inexpensive model suitable for simpler tasks. It supports a 16K context window and is optimized for dialogue. GPT-4 was built with broad general knowledge and domain expertise, showing much stronger performance in the driving theory test. However, the GPT-4 model is much more expensive, with an input cost 60 times that of GPT-3.5-turbo. GPT-4o is OpenAI's most advanced multimodal model, which is faster and cheaper than GPT-4 and has stronger vision capabilities, supporting a 128K context window. GPT-4o mini outperforms GPT-3.5-turbo on academic benchmarks in both textual intelligence and multimodal reasoning. The model features a context window of 128,000 tokens and supports up to 16,000 output tokens per request. Thanks to the improved tokenizer shared with GPT-4o, processing non-English text is now more cost-effective.

The OpenAI API was used to call the LLMs with input test questions and to obtain the model prediction outputs. GPT-3.5 is the most cost-effective, priced at $0.50 per 1M input tokens and $1.50 per 1M output tokens, with no image support. GPT-4 has significantly higher costs at $30.00 for input and $120.00 for output, also lacking image processing capabilities. GPT-4o offers a more balanced pricing model, with $5.00 per 1M input tokens, $15.00 per 1M output tokens, and a per-image processing cost of $0.001275.

2) Alibaba, Baidu, and ZhiPu AI LLM Models: The Tongyi Qianwen (Qwen) LLM model is provided by Alibaba to the open-source community. Qwen-turbo is developed for multilingual support, including both Chinese and English. It accommodates a context length of up to 8,000 tokens, facilitating the efficient handling of long input sequences. To ensure optimal performance, the input tokens are capped at a maximum of 6,000, which guarantees smooth operation and accurate outputs.

The Ernie-4.0-turbo-8k model, developed by Baidu, is enhanced with diverse training data, particularly in the areas of the Chinese language, service applications, and knowledge. This paper utilizes Ernie-4.0-turbo-8k for inference, which supports a context length of 128,000 tokens.

GLM-4-Plus is the latest flagship pre-trained language model developed by Zhizhi AI. In the September 2024 edition of the "SuperBench Comprehensive Model Capability Evaluation Report," 24 representative large models from both domestic and international sources were evaluated. The results indicate significant progress by domestic models in areas such as alignment, agent performance, and mathematical logic. GLM-4-Plus ranks third, surpassing the Claude series models. In terms of Chinese language proficiency, GLM-4-Plus leads with a score of 8.58, surpassing o1-preview. Additionally, in the semantic understanding capability evaluation, GLM-4-Plus exceeds o1-mini by 1 point.

3) MiniCPM LLM Model: MiniCPM3-4B is an edge-side LLM developed by ModelBest Inc. and TsinghuaNLP, consisting of 4 billion parameters excluding embeddings. Its performance surpasses and GPT-3.5-Turbo-0125, and it is comparable to several models with 7B-9B parameters, such as Llama3.1-8B-Instruct, Qwen2-7B-Instruct, and GLM-4-9B-Chat. The model natively supports a 32k context length and achieves superior average scores compared to benchmark models such as GPT-4 and KimiChat on the comprehensive long-text evaluation benchmark InfiniteBench.

MiniCPMLlama3-V2.5, developed based on Meta's open-source LLM Llama3 with vision capabilities, achieves state-of-the-art performance on multiple benchmarks among models with fewer than 7 billion parameters. The MiniCPM-V-2.6 is constructed using SigLip-400M and Qwen2-7B, comprising a total of 8 billion parameters. It demonstrates a significant performance improvement over MiniCPM-Llama3-V2.5 and introduces new capabilities for multi-image and video understanding.

4) DeepSeek LLM Models: DeepSeek-V3 and DeepSeek-R1 are cutting-edge models from the DeepSeek series, designed to advance general artificial intelligence capabilities. DeepSeek-V3 is a Mixture-of-Experts language model with 671B total parameters, activating 37B per token. Trained on 14.8 trillion tokens, it incorporates Multi-head Latent Attention and an auxiliary-loss-free load balancing strategy, achieving performance comparable to the top closed-source models. DeepSeek-R1 is a reasoning-optimized model built upon DeepSeek-V3's architecture [31]. It employs a multi-stage training process, including supervised fine-tuning on "cold-start" data and reinforcement learning with rule-based rewards, to enhance reasoning capabilities. DeepSeek-R1 matches or surpasses OpenAI's o1 model in tasks involving mathematics, coding, and logical reasoning.

D. Prompts Engineering

1) Multiple-Choice Tests: In this part, we adopt a two-tiered prompt design strategy to guide the LLMs in efficiently solving UK driving theory multiple-choice test questions. The system prompt serves as the foundational role-setting instruction, framing the LLM as an "experienced driver" to align its behavior and knowledge with that of a human expert familiar with driving scenarios and theory. By specifying the task as answering UK driving theory multiple-choice test questions with multiple choices and a single correct answer, the system prompt establishes clear expectations about the nature of the task, ensuring the model's focus on accuracy and task relevance.

The user prompt is more specific, providing clear output formatting requirements. It instructs the LLMs to return only the first letter (e.g., "A" or "B") of the correct answer without additional explanation. This directive minimizes extraneous output, reducing cognitive load and optimizing the response for quick, direct evaluation. The combination of both prompts ensures that the LLMs operates with role-appropriate knowledge while delivering concise and actionable responses in line with the expectations of a multiple-choice test environment.

2) Hazard Perception Tests: We detail the design of both the system prompt and the user prompt used to guide the LLM in the hazard perception test. The system prompt is crafted to define the AI's role and key capabilities, emphasizing its specialization in real-time visual data analysis, dynamic object detection, and hazard assessment. It introduces the concept of a "developing hazard" and lists specific examples, such as vehicles merging or pedestrians crossing, to contextualize the AI's decision-making process. By outlining these competencies, the system prompt ensures that the model is primed to interpret time-stamped image sequences from a vehicle's perspective and provide actionable risk evaluations.

The user prompt complements the system prompt by specifying the tasks the AI must execute in response to input data. The prompt is structured to require binary hazard detection ("1" for hazard detected, "0" for no hazard), followed by a detailed report when a hazard is identified. This report is segmented into four components: hazard type, visual characteristics, threat assessment, and recommended action. The clear formatting and concise instructions promote rapid and accurate responses, enabling immediate decision-making in driving scenarios. Together, the system and user prompts create a well-defined framework that guides the LLM in performing hazard perception efficiently and reliably, balancing clarity, speed, and accuracy.

IV. Experiment Results and Discussions

A. Experiment Settings

All experiments in this study were conducted using Python 3.11. The open-source LLM from the MiniCPM series was deployed on a GPU server with an RTX 4090 24GB graphics card. The server environment was configured following the deployment requirements of the MiniCPM-3-4B model. Additional LLMs were accessed via APIs provided by their respective service vendors. These API calls were performed on a laptop with an i9-13900H CPU, RTX 4060 8GB graphics card, and 32 GB of RAM. Due to the use of a personal computer and network for these experiments, the response times of API-accessed LLMs were affected by random network latency. In the initial experimental setup, we set the large language model parameters to Temperature = 0 and Top_p = 1. Temperature controls output randomness. A lower value makes responses more deterministic, while a higher value increases diversity. Top_p (nucleus sampling) limits token selection to the most probable subset, ensuring a balance between diversity and coherence.

In the multiple-choice test, we sequentially input all 617 DS-Text and 109 DS-Image items into the LLM for testing, evaluating one item at a time. Finally, the test results were obtained through the average accuracy of the two datasets. To obtain sufficient historical information, we implemented a buffer that can hold up to 30 frames. Additionally, considering that the test videos provided by the officials were too long, we appropriately edited the videos without affecting the test results.

B. Evaluation Method

The evaluation of LLM-based hazard perception test results was conducted using a multi-faceted approach, encompassing three distinct perspectives to ensure a comprehensive assessment of model performance:

1) Frame-Level Evaluation: This granular analysis focuses on the binary classification accuracy of individual frames, assessing the model's capability to correctly identify the presence or absence of hazards at each time point. Performance metrics employed in this evaluation include accuracy, precision, recall, confusion matrix, and F1-score. These metrics provide a detailed quantitative assessment of the model's frame-by-frame hazard detection capabilities.

2) Event-Level Evaluation: In contrast to the frame-level approach, this method adopts a more holistic perspective, concentrating on the detection of continuous hazard events. This evaluation is particularly pertinent for higher-level hazard event detection scenarios, where the accurate identification of an entire hazardous episode is of greater importance than the precise classification of individual frames.

In the event-level evaluation, we define an event as a continuous time period during which a hazard is present. A detected event must overlap with the ground truth event in time to be considered a match. **Intersection over Union (IoU):** IoU is a metric that calculates the ratio of the overlapping duration of the detected event and the ground truth event to the duration of their union.

$$\text{IoU} = \frac{\text{Overlap}}{\text{Detected Event} + \text{Ground Truth Event} - \text{Overlap}} \tag{1}$$

Matching Criterion: An event is considered correctly detected if the IoU between the detected event and the ground truth event exceeds a specified threshold, commonly set at 0.5 or 0.7.

3) Scoring Mechanism: This approach mimics the UK official hazard perception test, emphasizing the principle that earlier detection of a hazard yields higher scores, while also incorporating mechanisms to penalize false positives and misses. This design provides a comprehensive evaluation of the model's performance in hazard recognition tasks, particularly in early detection and in scenarios involving false alarms and misses. Each hazard event is scored based on its actual starting time. The scoring rules are as follows: if the hazard level (1-5) at the start of the event is correctly detected, the corresponding hazard level is added to the score. The maximum score for each sample is 5, and the maximum score for the entire dataset is the number of samples multiplied by 5.

This tripartite evaluation framework provides a comprehensive and multidimensional assessment of LLM-based hazard perception test performance. By integrating frame-level precision, event-based detection, and a time-sensitive scoring mechanism, this approach offers a holistic view of model efficacy, encompassing both micro-level accuracy and macro-level hazard recognition capabilities. Such a diverse evaluation strategy is crucial for developing robust and reliable hazard perception systems applicable to real-world safety-critical scenarios.

C. Evaluation Metrics

In this study, several evaluation metrics are employed to assess the performance of LLMs in answering driving theory test questions. These metrics include **Accuracy**, **Precision**, **Recall**, and the **F1-Score**. Each metric is defined below:

1) Accuracy: Accuracy measures the proportion of correct answers (both correct hazard detections and correct non-hazard identifications) out of the total number of test questions. It is defined as:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

where **TP** (True Positives) represents correct hazard detections, **TN** (True Negatives) represents correct non-hazard identifications, **FP** (False Positives) represents incorrect hazard detections, and **FN** (False Negatives) represents missed hazard detections.

2) Precision: Precision quantifies the accuracy of the model in identifying hazards by measuring the proportion of correctly identified hazards out of all hazard identifications. The formula is:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3}$$

High precision indicates that the model has a low false-positive rate, meaning it rarely identifies non-hazard situations as hazards.

3) Recall: Recall measures the model's ability to correctly identify all actual hazards. It is defined as:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4}$$

High recall indicates that the model is good at identifying most hazards, minimizing missed detections.

4) F1-Score: The F1-Score is the harmonic mean of Precision and Recall. It provides a balanced metric to evaluate the model's performance by taking into account both false positives and false negatives. The F1-Score is calculated as:

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5}$$

These metrics allow for a comprehensive evaluation of how well the LLMs perform in the driving theory test, balancing the trade-offs between making too many false-positive detections (low precision) and missing actual hazards (low recall). By focusing on these metrics, we aim to measure the LLMs' effectiveness in simulating human-level driving theory knowledge.

5) Real-Time Performance Metrics: To evaluate the real-time efficiency of API-based LLMs in driving theory tests, we adopt **Throughput** as the primary metric. Throughput represents the number of MCQs processed by the LLM per second, defined as:

$$\textit{Throughput} = \frac{N}{T_{\text{total}}} \tag{6}$$

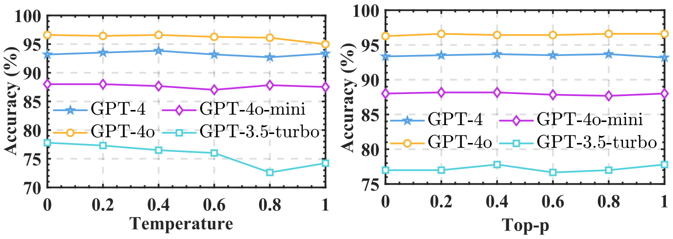
where N denotes the total number of MCQs in the test, and T_{total} is the total time taken to process all questions. This metric reflects the model's overall inference speed, taking into account potential network and system delays.

6) Computational Efficiency Metrics: To evaluate the computational efficiency of LLMs in driving theory test, we designed two key metrics: *Time per Token (TPT)* and *Cost per Token (CPT)*. TPT measures the average computational time required to process each token, calculated as the response time divided by the total number of tokens (input + output), reflecting the inference speed and responsiveness of the model in real-time applications. CPT quantifies the financial cost associated with processing each token, computed as the total API cost divided by the number of tokens processed, enabling a comparative analysis of different models in terms of cost-efficiency. By analyzing these metrics across various LLMs, we aim to assess trade-offs between processing speed, computational expense, and overall feasibility for real-world autonomous driving applications.

It is noted that since we evaluate LLMs via API calls, both real-time performance and computational efficiency are significantly influenced by network conditions. To ensure consistency, we conduct all evaluations in the same network environment as much as possible. In addition, we mitigated the impact of short-term network fluctuations by performing multiple measurements and averaging the results.

D. Multiple-Choice Test Results

The evaluation of various LLMs on the DS-Text dataset highlights their distinct capabilities in handling textual information. As shown in Tabel I, the GPT-4o model excels with a high accuracy of 96.60%, efficiently processing 617 questions in 279.65 seconds, demonstrating a strong balance between accuracy and speed. Conversely, Claude-3.5-sonnet achieves a similar accuracy of 93.35% but requires a substantially longer processing time of 689.68 seconds, suggesting a trade-off between computational demands and precision. The glm-4-plus model performs adequately with an accuracy of 89.47%. However, it has an extended processing duration of 897.87 seconds, which may be due to the complexity of generating or handling comprehensive outputs. At the lower end of the performance spectrum, MiniCPM-3-4B has the lowest accuracy of 70.83% and moderate processing time, highlighting a need for architectural enhancements in purely textual scenarios. The DeepSeek series models also demonstrated strong performance, with DeepSeek-V3 and DeepSeek-R1 achieving accuracies of 93.68% and 95.69%, respectively, on the DS-Text dataset. However, as the DeepSeek series models have not yet made image processing capabilities available to users, they could not be evaluated on the DS-Image dataset.



(a) Model Accuracy vs. Temperature (Top_p=1.0)

(b) Model Accuracy vs. Top_p (Temperature=0.0)

Fig. 4. Comparison of openai model accuracy across different parameter settings on the DS-Text. (a) Accuracy as a function of the temperature parameter with Top-p set to 1.0. (b) Accuracy as a function of the Top-p parameter with temperature set to 0.0. models compared include GPT-4, GPT-4o, GPT-4o-mini, and GPT-3.5-turbo.

In the DS-Image dataset, all models exhibit decreased accuracy, reflecting the increased complexity of integrating text and image data. As shown in Table II, GPT-4o maintains the lead with an accuracy of 79.82% and a swift processing time of 50.51 seconds, demonstrating robustness in multimodal contexts. Although Claude-3.5-sonnet performs similarly with 77.06% accuracy, its processing time extends to 78.5 seconds, indicating challenges in handling image data. The glm-4v-plus model's accuracy further drops to 69.72%, requiring 199.7 seconds, indicative of inefficiencies with multimodal input. Notably, the MiniCPM variants, V-2.6 and Llama3-V-2.5, achieve the lowest accuracies at 57.80% and 55.96%, respectively, with significantly increased processing times, highlighting substantial gaps in effectively managing complex data types.

Overall, GPT-4o consistently demonstrates high performance across both datasets, effectively balancing accuracy and efficiency in text and image-inclusive tasks. This analysis highlights the need for ongoing optimization in model architecture, particularly for handling multimodal data where current models experience performance degradation. Trade-offs among speed, accuracy, and computational cost are evident, indicating that future work should enhance model efficiency and precision, especially in complex and diverse data environments.

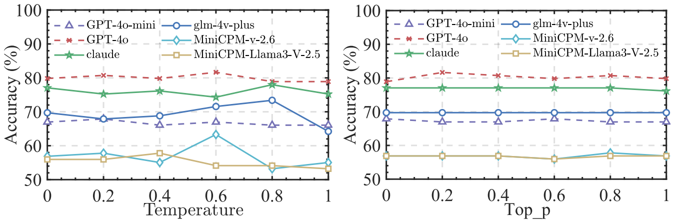
Furthermore, we examined model performance across the DS-Text and DS-Image datasets, highlighting the impact of the parameters temperature and top-p on accuracy. In the DS-Text dataset, models generally exhibit minimal accuracy variations with changes in the temperature parameter when top-p is set to 1. As shown in Fig. 4, GPT-4 and GPT-4o consistently maintain high accuracy, indicating robustness to temperature adjustments. Their performance remains near 95%, illustrating reliability in text-based tasks under stable conditions. Similarly, when examining the top-p parameter with the temperature set to 0, models display stable accuracy levels. Both GPT-4o-mini and GPT-3.5-turbo show negligible

TABLE I
PERFORMANCE COMPARISON OF DIFFERENT LLMS ON THE DS-TEXT

Model	Temperature	Top_p	Output_tokens	Input_tokens	Num_questions	Num_correct	Time	Accuracy
GPT-4	0	1	654	69911	617	575	375.57	93.19%
GPT-4o	0	1	618	69707	617	596	279.65	96.60%
Claude-3-5-sonnet	0	1	2468	72803	617	576	689.68	93.35%
glm-4-plus	0	1	1851	66601	617	552	897.87	89.47%
GPT-4o-mini	0	1	617	69707	617	543	278.14	88.01%
Ernie-4.0-turbo-8k	0.1	1	617	66261	617	533	2063.78	86.39%
GPT-3.5-turbo	0	1	629	69911	617	480	260.48	77.80%
Qwen-turbo	0	1	617	71404	617	459	604.69	74.39%
DeepSeek-V3	0	1	/	/	617	578	/	93.68%
DeepSeek-R1	0	1	/	/	441	422	/	95.69%
Meta-Llama-3-70B	0	1	/	/	617	544	/	88.17%
MiniCPM-3-4B	0.1	1	/	/	617	437	407.82	70.83%

TABLE II
PERFORMANCE COMPARISON OF DIFFERENT LLMS ON THE DS-IMAGE

Model	Temperature	Top_p	Output_tokens	Input_tokens	Num_questions	Num_correct	Time	Accuracy
GPT-4o	0	1	109	19357	109	87	50.51	79.82%
Claude-3-5-sonnet	0	1	436	21316	109	84	78.5	77.06%
glm-4v-plus	0	1	218	19200	109	76	199.7	69.72%
GPT-4o-mini	0	1	161	19357	109	73	62.57	66.97%
MiniCPM-V-2.6	0.1	1	/	/	109	63	23.05	57.80%
MiniCPM-Llama3-V-2.5	0.1	1	/	/	109	61	142.11	55.96%



(a) Model Accuracy vs. Temperature (Top_p=1.0) (b) Model Accuracy vs. Top_p (Temperature=0.0)

Fig. 5. Comparison of openai model accuracy across different parameter settings on the DS-Image. (a) Accuracy as a function of the temperature parameter with Top-p set to 1.0. (b) Accuracy as a function of the Top-p parameter with temperature set to 0.

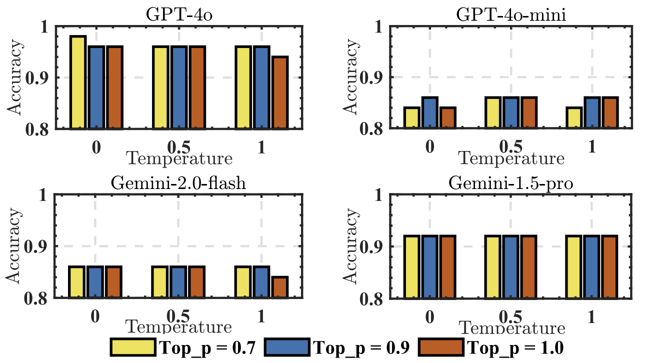
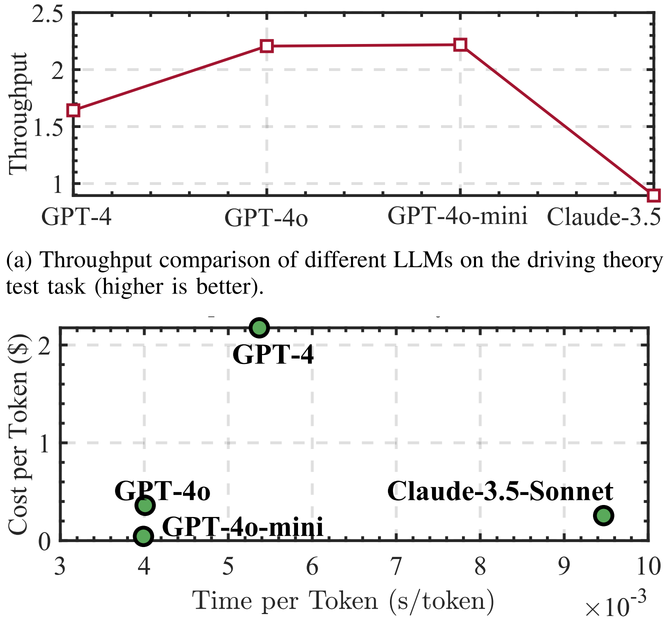


Fig. 6. The effect of temperature and Top_p on model accuracy in MCQ tasks.

fluctuations, suggesting that top-p variations do not significantly impact their performance in these scenarios.

In contrast, the DS-Image dataset shows a more pronounced impact of the temperature parameter on accuracy. As shown in Fig. 5, models like Claude and glm-4v-plus exhibit declines in performance as temperature increases, reflecting vulnerability to sampling randomness in image-inclusive tasks. However, GPT-4o-mini maintains relatively consistent accuracy, underscoring its robustness across different sampling temperatures. When examining the top-p parameter with temperature set to 0, model responses vary. Across both datasets, model performance generally remains stable with varying top-p values, as shown in Fig. 4b and Fig. 5b. However, temperature variations induce more noticeable fluctuations in accuracy, particularly for lower-performing models, as shown in Fig 4a and Fig 5a. This suggests that temperature may be a more critical parameter for optimizing model performance, especially for tasks involving multimodal inputs. The observed performance discrepancies between DS-Text and DS-Image tasks underscore the challenges associated with multimodal reasoning. The consistent superiority of GPT-4o across both tasks indicates its robust generalization capabilities. Conversely, the performance gap between GPT-4o and other models widens in the DS-Image task, highlighting the varying degrees of multimodal integration among different LLMs. Since only some models were tested on both the DS-Text and DS-Image datasets, we used the average accuracy of the two datasets as the final score for the models in the multiple-choice test. Therefore, the final test accuracy for GPT-4o is 88.21%, for GPT-4o-mini is 77.49%, and for Claude-3.5-sonnet is 85.21%.

To evaluate the combined impact of Temperature and Top_p on model performance, we conducted experiments on the standardized MCQ dataset across 4 large-language models: GPT-4o, GPT-4o-mini, Gemini-1.5-pro, and Gemini-2.0-flash. Our results indicate that within the limited test dataset, variations in Temperature and Top_p did not lead to significant differences in accuracy between models when answering the MCQs. The experimental results are shown in Fig 6. We hypothesize that this is primarily due to the structured nature of our output constraints: models were required to



(a) Throughput comparison of different LLMs on the driving theory test task (higher is better).

(b) Computational efficiency trade-off: Cost per token vs. time per token.

Fig. 7. Evaluation of real-time performance and computational efficiency of LLMs in the driving theory test. (a) Throughput comparison of different LLMs on the driving theory test task (higher is better). (b) Trade-off between inference cost and computational speed. Models evaluated include GPT-4, GPT-4o, GPT-4o-mini, and Claude-3.5-Sonnet.

generate only the final answer in a structured format. This strict output requirement significantly limited the diversity of possible responses, thereby reducing the potential influence of sampling parameters such as Temperature and Top_p.

Additionally, MCQs are inherently deterministic tasks, as they typically have a single correct answer and do not require the model to generate complex, long-form reasoning. This contrasts with open-ended generation tasks (e.g., open-domain question answering or writing), which rely more heavily on Temperature and Top_p to regulate output diversity. In MCQ tasks, the model primarily selects from a limited set of predefined options, which inherently constrains the impact of sampling strategies. These findings suggest that while Temperature and Top_p are crucial for open-ended text generation, their impact may be diminished in structured response settings, where the output space is inherently restricted. Furthermore, the deterministic nature of MCQs further limits the extent to which these parameters can influence model performance.

Next, we evaluate the real-time efficiency of mainstream LLMs on the driving theory test task, as shown in Fig. 7a. The results show that GPT-4o and GPT-4o-mini achieve the highest throughput (2.21 and 2.22 questions/sec, respectively), indicating superior inference speed. In contrast, Claude-3.5-Sonnet records the lowest throughput (0.89 questions/sec), suggesting a slower response possibly due to resource limitations or API access mechanisms. Overall, GPT-4o and GPT-4o-mini demonstrate significantly better real-time performance compared to other models.

In terms of computational efficiency, the scatter plot analysis reveals notable differences among the models. GPT-4 exhibits the highest TPT at 0.00537s and the highest CPT at $2.18, indicating substantial computational resource consumption, making it less suitable for high-efficiency inference tasks. In contrast, GPT-4o and GPT-4o-mini achieve the lowest TPT (0.00401s and 0.00399s, respectively) and the lowest CPT ($0.36 and $0.043, respectively), demonstrating superior cost-effectiveness. Claude-3.5-Sonnet falls within an intermediate range in terms of TPT (0.00947s) and CPT ($0.25), yet its lower throughput suggests a slower inference process despite its relatively moderate cost. Overall, GPT-4o-mini emerges as the most computationally efficient model, offering the fastest inference speed at the lowest cost, whereas GPT-4 incurs the highest computational cost, and Claude-3.5-Sonnet presents a trade-off between cost and efficiency.

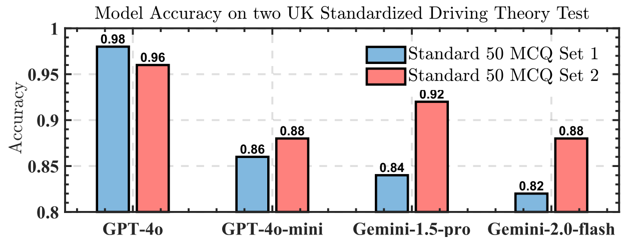


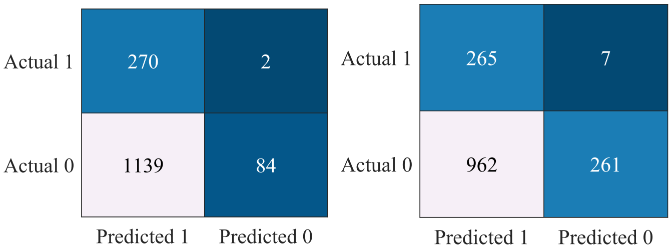
Fig. 8. Accuracy of different LLMs on two sets of UK standardized driving theory MCQs.

E. UK Standardized Driving Theory Test Performance

The standardized MCQs tests officially provided in the UK DSVA are generally designed to be more representative, ensuring a well-balanced distribution of question types, difficulty levels, and key knowledge points. This structured approach improves the comprehensiveness and fairness of the assessment. The official UK driving theory test consists of 50 multiple-choice questions. For our evaluation, we collected two sets of previousious exam papers. Additionally, we observed that Google's Gemini series models have already been utilized in previous studies to develop large language model-based autonomous driving systems [12]. Therefore, we included the latest Gemini models, Gemini-1.5-Pro and Gemini-2.0-Flash, in our evaluation. The experimental results are presented in Fig. 8. GPT-4o achieves the highest accuracy on both test sets, scoring 0.98 and 0.96, respectively. This suggests that GPT-4o demonstrates the most reliable performance in answering driving theory questions. In general, GPT-4o shows superior performance and stability, while Gemini models show greater variability between different test sets, suggesting potential differences in how these models process domain-specific driving knowledge.

F. Hazard Perception Test Results

1) Frame-Level Evaluation: The frame-level evaluation provides a detailed examination of each model's proficiency using several key performance metrics: accuracy, precision, recall, and F1-score. These metrics offer a quantitative assessment of the model's ability to detect hazards frame-by-frame. In the confusion matrix, GPT-4o achieves 270 true positives and 84 true negatives, with only 2 false negatives and



(a) GPT-4o

(b) GPT-4o-mini

Fig. 9. Frame-level confusion matrix for GPT-4o and GPT-4o-mini models. Subfigure (a) presents the confusion matrix for GPT-4o, while subfigure (b) shows the matrix for GPT-4o-mini. These matrices illustrate the models' performance in classifying frames as hazards or non-hazards.

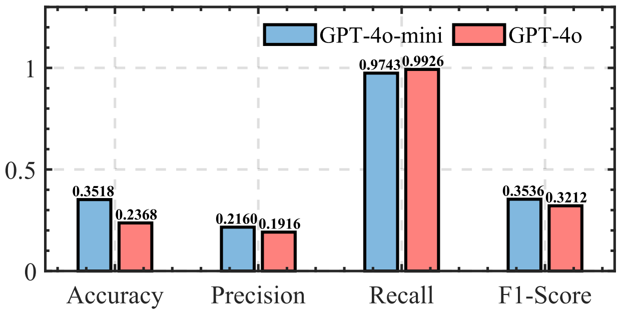


Fig. 10. Frame-level performance metrics for GPT-4o and GPT-4o-mini models. The bar graph compares the accuracy, precision, recall, and F1-score between the two models, highlighting their frame-level evaluation results.

1139 false positives. In contrast, GPT-4o-mini records 265 true positives and 261 true negatives, demonstrating improved specificity with 7 false negatives and 962 false positives, as shown in Fig. 9.

In Fig. 10, frame-level performance metrics for the GPT-4o and GPT-4o-mini models are shown. a)Accuracy: Accuracy measures the proportion of total frames that were correctly identified by the model. GPT-4o achieves an accuracy of 0.3518, indicating it correctly classifies approximately 35% of the frames. GPT-4o-mini has a lower accuracy of 23.68%, reflecting its relatively reduced capability in overall frame classification. b)Precision: Precision focuses on the model's ability to reduce false positive rates by measuring the proportion of true positive identifications among all positive identifications. With a precision of 21.6%, GPT-4o-mini excels over GPT-4o, which has a precision of 19.16%. This suggests that GPT-4o-mini is slightly better at minimizing false alarms. c)Recall: Also known as sensitivity, recall assesses the model's ability to capture actual positive instances. GPT-4o boasts a high recall rate of 0.9926, surpassing GPT-4o-mini's recall of 97.43%, underscoring its superior effectiveness in identifying hazardous frames. d)F1-Score: The F1-score balances precision and recall, providing a harmonic mean that encapsulates both metrics. GPT-4o achieves an F1-score of 32.12%, whereas GPT-4o-mini reaches 35.36%, indicating that GPT-4o-mini maintains a slightly better balance between detecting hazards and minimizing false positives.

The evaluation highlights the strengths and weaknesses of each model across various performance criteria. GPT-4o's remarkable recall demonstrates a strong capacity for detecting hazards, making it ideal for scenarios requiring high sensitivity. However, its lower precision indicates a need for refinement to enhance specificity and reduce false alarms. Conversely, GPT-4o-mini exhibits a more balanced performance, with higher precision suggesting better management of false positives, although it sacrifices some recall. This trade-off may make it advantageous in applications where minimizing false positives and conserving computational resources are priorities.

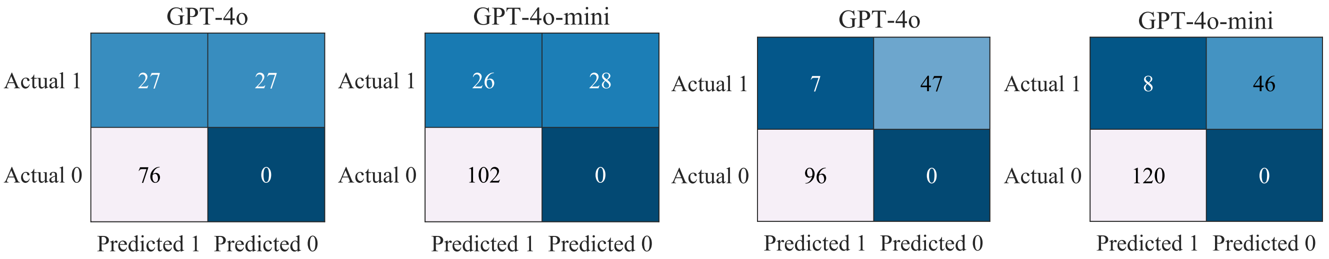
In summary, these metrics provide valuable insights into the operational characteristics of both models, guiding their application in realistic driving environments. Further optimization could enhance their ability to achieve an ideal balance between sensitivity and specificity, ultimately improving safety and reliability in hazard perception tasks.

2) Event-Level Evaluation: In the event-level evaluation, we focus on identifying continuous hazard events, a crucial aspect for scenarios where accurately detecting entire hazardous episodes is more important than classifying individual frames. This holistic approach provides insights into the models' capability to recognize ongoing hazards effectively.

Using Intersection over Union (IoU) as a metric, we evaluated the models' performance based on the overlap between detected events and ground truth events, as shown in Fig. 11. The confusion matrices provide detailed insights into their performance: GPT-4o achieves 270 true positives and 84 true negatives, with only 2 false negatives and 1139 false positives. This indicates a high sensitivity to hazard detection, reflected in impressive IoU scores, particularly at thresholds of 0.5 and 0.7. GPT-4o-mini records 265 true positives and 261 true negatives, with 7 false negatives and 962 false positives. Although slightly less sensitive than GPT-4o, it demonstrates improved specificity, handling false predictions more effectively.

Fig. 12 illustrates the comparison of metrics between GPT-4o and GPT-4o-mini across various detection thresholds, focusing on accuracy, precision, recall, and F1-score. The GPT-4o model demonstrates impressive performance at a threshold of 0.2, achieving 52.43% accuracy, 52.43% precision, 100% recall, and an F1-score of 68.79%, indicating high sensitivity and balanced precision. However, when the threshold increases to 0.85, performance declines significantly: accuracy drops to 0.64%, precision to 0.97%, recall to 1.85%, and F1-score to 1.27%, reflecting diminished detection capability under stricter classification criteria. In comparison, the GPT-4o-mini model shows lower overall accuracy at the 0.2 threshold, with 38.93% accuracy, 39.84% precision, 94.44% recall, and an F1-score of 56.04%, yet maintains good sensitivity. At a threshold of 0.85, its performance also deteriorates substantially, with accuracy at 1.11%, precision at 1.56%, recall at 3.70%, and an F1-score of 2.20%, highlighting challenges in event detection under stringent conditions.

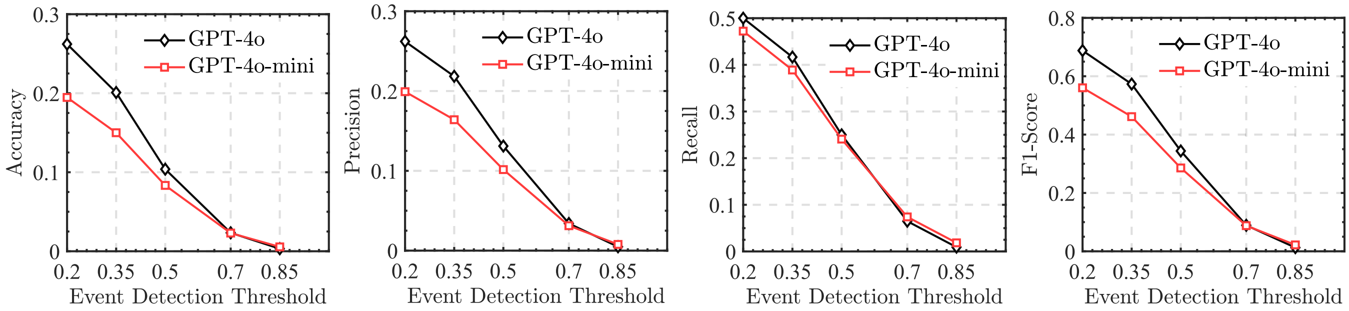
The evaluation reveals that GPT-4o excels in overall event detection accuracy and recall, making it highly effective in environments where identifying all potential hazards is critical. However, its relatively lower precision suggests that there is room for improvement in minimizing false positives. However, GPT-4o-mini, while slightly lagging in recall and accuracy, demonstrates stronger precision at higher thresholds. This



(a) Confusion Matrix (Threshold: 0.5)

(b) Confusion Matrix (Threshold: 0.7)

Fig. 11. Event-level evaluation results for GPT-4o and GPT-4o-mini models at different thresholds. Subfigure (a) shows the confusion matrices at a threshold of 0.5, while subfigure (b) presents them at a threshold of 0.7. These matrices illustrate the models' performance in classifying actual and predicted values, highlighting the effectiveness in event detection tasks.



(a) Accuracy Comparison

(b) Precision Comparison

(c) Recall Comparison

(d) F1-Score Comparison

Fig. 12. Comparison of metrics between GPT-4o and GPT-4o-mini models across various event detection thresholds. Subfigures (a) through (d) display the models' performance in terms of accuracy, precision, recall, and F1-Score, respectively, highlighting differences in effectiveness at detecting events as the threshold is varied.

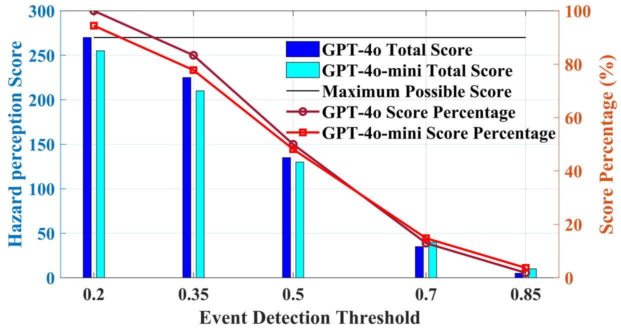


Fig. 13. Comparison of hazard perception scores and score percentages between GPT-4o and GPT-4o-mini models across various event detection thresholds. The bar graph represents the total scores for each model, while the line graph indicates the score percentages relative to the maximum possible score, illustrating model performance variations as the threshold changes.

characteristic makes it a viable option in applications where the reduction of false alarms is prioritized over the capture of every event.

The experimental results comparing GPT-4o and GPT-4o-mini models across various event detection thresholds reveal a clear performance trend, as shown in Fig. 13. At the lowest threshold of 0.2, GPT-4o achieves near-perfect performance with a score of 270 of 270 (100%), while GPT-4o-mini follows closely at 255 (94.4%). As the threshold increases, both models show a consistent decline in performance, with scores converging around 130-135 (48-50%) at a threshold of 0.5. Interestingly, at higher thresholds, the performance gap narrows, with GPT-4o-mini slightly outperforming GPT-4o at 0.7 (40 vs. 35 points) and 0.85 (10 vs. 5 points). This pattern suggests that both models excel at early hazard detection, but struggle under stricter criteria, with GPT-4o-mini showing slightly better resilience at very high thresholds. The findings indicate optimal model operation at lower thresholds (0.2 to 0.35) to balance accuracy and early detection, highlighting the trade-off between sensitivity and specificity in hazard perception tasks.

3) Results Discussion: The experimental results highlight a critical issue in the event and frame detection performance of both LLMs: low precision and high recall across different thresholds. From the confusion matrices, it is evident that both models tend to classify many instances as positive (risk events). While this behavior leads to high recall (i.e., most actual risk events are correctly identified), it also results in a significant number of false positives, leading to low precision. The high recall nature of the models suggests that they are sensitive to detecting risk events, which is beneficial in safety-critical applications. However, the low precision implies that the models produce many false alarms, which could reduce trust in their predictions and lead to unnecessary interventions. For example, as shown in the Fig. 14, when a pedestrian is present in the adjacent non-motorized lane, whether moving or stationary, the LLM perceives a potential risk of sudden intrusion into the vehicle's path. Consequently, it classifies this scenario as a hazard and responds by adopting a cautious driving strategy, maintaining vigilance, and reducing speed. The overly conservative risk assessment by LLMs can be attributed to three key factors: (1) Lack of driving-specific training, as the models have not been fine-tuned with expert-labeled traffic

a) Hazard Type: Pedestrian
b) Visual Characteristics: Group of individuals; one wearing a green jacket, on the right side on the sidewalk.
c) Threat Assessment: The pedestrian wearing the green jacket is potentially about to cross the street, posing a possible hazard due to proximity and their position near the curb.
d) Recommended Action: Slow down and be prepared to stop in case the pedestrian steps onto the road. Maintain a safe distance and monitor their movements closely.

Fig. 14. Example of LLM-based hazard perception in a driving scenario. The GPT-4o identifies a pedestrian on the sidewalk as a potential hazard due to their proximity to the curb and the possibility of crossing the street. While this cautious approach enhances safety, it also exemplifies the LLM's overly conservative hazard perception, which may lead to unnecessary slowdowns and reduced driving efficiency.

datasets; (2) Bias toward safety in RLHF optimization, where over-penalization for underestimating risks leads to excessive risk aversion; and (3) Prompt design and task framing, where the structure and wording of prompts may unintentionally encourage excessive caution, highlighting the need for refinement in hazard perception task formulation.

Furthermore, certain experimental settings may inherently influence the performance of hazard perception. For example, in our current setup, the initial and final segments of all hazard-containing videos are devoid of risk events. This design choice is consistent with the official UK driving theory test, as the hazard-containing video data is sourced from officially released datasets. However, such a configuration can limit the evaluation of an LLM's ability to detect and respond quickly to imminent hazards, particularly in high-risk, time-critical scenarios. Another key factor affecting performance is the hazardous length of the window, which is constrained by the input capacity of the LLM. For example, when accessed via API, GPT-4o has a maximum input length of 128k tokens, imposing limitations on the temporal context available for hazard assessment. Extending the hazardous time window is expected to enhance the contextual understanding of the LLM, thus improving its hazard perception capability. Recognizing these limitations, we have integrated these insights into our future research agenda, where our aim is to develop strategies to optimize LLM-based hazard perception, ultimately contributing to safer and more reliable autonomous driving systems.

V. Conclusion and Future Work

In conclusion, this paper investigated the applicability of driving theory tests to LLMs, using a benchmark composed of both MCQs and hazard perception tasks. A systematic testing methodology was designed, and various general-purpose open- and closed-source LLMs were evaluated. The experimental results revealed both the capabilities and limitations of current LLMs in the context of structured driving knowledge. While mainstream LLMs demonstrate promising generalization and reasoning abilities, they are not yet fully capable of handling all aspects of autonomous driving assistance. For example, only GPT-4o achieved a passing score (88.21%) in the full MCQ test, while other models particularly underperformed in image-based hazard perception. By contrast, many models performed well in textual MCQs, exceeding the 86% threshold, indicating that perception remains a major bottleneck. Even GPT-4o exhibited low precision and a high false positive rate in the hazard perception task, reflecting the challenges of scene understanding and dynamic risk identification.

These findings offer valuable insights for the design, evaluation, and deployment of LLMs in autonomous driving systems, especially in balancing reasoning performance with computational feasibility. The proposed driving theory test framework provides a reproducible benchmark for evaluating LLMs before integrating them into critical real-world systems. Additionally, this framework can be extended to support on-device LLM deployment and assist in assessing decision-making modules in broader autonomous driving pipelines.

As research on driving theory test evaluation for LLMs is still at an early stage, several directions can be pursued in future work. First, it is important to note that the current benchmark is based on the UK driving test, which limits the global generalizability of the results. One direction forward is to consider cross-regional adaptation and evaluation, inspired by recent work [32]. Moreover, although this work focuses on general-purpose LLMs, several domain-specific open-source models for driving have emerged, such as GPT-Driver [8], DriveGPT4 [19], and DriveLM [33]. These models provide valuable references and will be considered for inclusion in future evaluations.

Finally, we can explore techniques such as fine-tuning and retrieval-augmented generation to improve domain understanding. These methods have shown effectiveness in other domain-specific tasks [34], [35], and may support better performance on structured, high-stakes assessments. Once LLMs reach satisfactory performance in theory-based evaluations, it will be meaningful to extend the benchmark toward real-world driving decision-making scenarios, including long-tail and dynamic contexts. A comprehensive evaluation framework combining knowledge assessment and situational decision-making could then serve as a foundation for standardized testing of LLM-powered autonomous driving agents.

Acknowledgment

The authors would like to thank Driving Instructor websites for sharing their driving theory test questions.

References

[1] W. H. Org. (2023). *Global Status Report on Road Safety 2023*. [Online]. Available: https://www.who.int/teams/social-determinants-of-health/safety-and-mobility/global-status-report-on-road-safety-2023

[2] T. Seymour. (2018). *Crash Repair Market To Reduce By 17% By 2030 Due To Advanced Driver Systems, Says ICDP*. [Online]. Available: https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/autonomous-drivings-future-convenient-and-connected

[3] W. Zhou, Z. Cao, N. Deng, X. Liu, K. Jiang, and D. Yang, "Dynamically conservative self-driving planner for long-tail cases," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 3, pp. 3476–3488, Mar. 2023.

[4] E. Gao et al., "Long-tailed traffic sign detection using attentive fusion and hierarchical group softmax," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 12, pp. 24105–24115, Dec. 2022.

[5] J. He, K. Yang, and H.-H. Chen, "6G cellular networks and connected autonomous vehicles," *IEEE Netw.*, vol. 35, no. 4, pp. 255–261, Jul. 2021.

[6] A. Matin and H. Dia, "Impacts of connected and automated vehicles on road safety and efficiency: A systematic literature review," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 3, pp. 2705–2736, Mar. 2023.

[7] D. Fu et al., "Drive like a human: Rethinking autonomous driving with large language models," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. Workshops (WACVW)*, Jan. 2024, pp. 910–919.

[8] J. Mao, Y. Qian, H. Zhao, and Y. Wang, "GPT-driver: Learning to drive with GPT," in *Proc. NeurIPS Found. Models Decis. Making Workshop*, Jan. 2023, pp. 1–21. [Online]. Available: https://openreview.net/forum?id=Pvjk9lxLJK

[9] DeepSeek-AI et al., "DeepSeek-V3 technical report," 2024, *arXiv:2412.19437*.

[10] DeepSeek-AI et al., "DeepSeek LLM: Scaling open-source language models with longtermism," 2024, *arXiv:2401.02954*.

[11] L. Wen et al., "Dilu: A knowledge-driven approach to autonomous driving with large language models," in *Proc. The 12th Int. Conf. Learn. Represent.*, 2024, pp. 1–11. [Online]. Available: https://openreview.net/forum?id=OqTMUPuLuC

[12] J.-J. Hwang et al., "EMMA: End-to-end multimodal model for autonomous driving," 2024, *arXiv:2410.23262*.

[13] Z. Tang, J. He, D. Pe, K. Liu, T. Gao, and J. Zheng, "Test large language models on driving theory knowledge and skills for connected autonomous vehicles," in *Proc. Workshop Mobility Evolving Internet Archit.*, Nov. 2024, pp. 1–6.

[14] H. Sha et al., "LanguageMPC: Large language models as decision makers for autonomous driving," 2023, *arXiv:2310.03026*.

[15] M. Azarafza, M. Nayyeri, C. Steinmetz, S. Staab, and A. Rettberg, "Hybrid reasoning based on large language models for autonomous car driving," 2024, *arXiv:2402.13602*.

[16] J. Liu, P. Hang, X. Qi, J. Wang, and J. Sun, "MTD-GPT: A multi-task decision-making GPT model for autonomous driving at unsignalized intersections," in *Proc. IEEE 26th Int. Conf. Intell. Transp. Syst. (ITSC)*, Sep. 2023, pp. 5154–5161.

[17] S. Wang, Y. Zhu, Z. Li, Y. Wang, L. Li, and Z. He, "ChatGPT as your vehicle co-pilot: An initial attempt," *IEEE Trans. Intell. Veh.*, vol. 8, no. 12, pp. 4706–4721, Dec. 2023.

[18] H. Shao et al., "LMDrive: Closed-loop end-to-end driving with large language models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 15120–15130.

[19] Z. Xu et al., "DriveGPT4: Interpretable end-to-end autonomous driving via large language model," *IEEE Robot. Autom. Lett.*, vol. 9, no. 10, pp. 8186–8193, Oct. 2024.

[20] S. Sreeram, T.-H. Wang, A. Maalouf, G. Rosman, S. Karaman, and D. Rus, "Probing multimodal LLMs as world models for driving," 2024, *arXiv:2405.05956*.

[21] T.-H. Wang et al., "Drive anywhere: Generalizable end-to-end autonomous driving with multi-modal foundation models," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2024, pp. 6687–6694.

[22] J. You et al., "V2X-VLM: End-to-end V2X cooperative autonomous driving through large vision-language models," 2024, *arXiv:2408.09251*.

[23] D. Hendrycks et al., "Measuring massive multitask language understanding," 2020, *arXiv:2009.03300*.

[24] A. Srivastava et al., "Beyond the imitation game: Quantifying and extrapolating the capabilities of language models," 2022, *arXiv:2206.04615*.

[25] K. Cobbe et al., "Training verifiers to solve math word problems," 2021, *arXiv:2110.14168*.

[26] A. Pal, L. K. Umapathi, and M. Sankarasubbu, "Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering," in *Proc. PMLR*, 2022, pp. 248–260.

[27] D. Hendrycks et al., "Aligning AI with shared human values," 2020, *arXiv:2008.02275*.

[28] J. Wei et al., "Chain-of-thought prompting elicits reasoning in large language models," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 24824–24837.

[29] Driver and Vehicle Standards Agency (DVSA), *The Official DVSA Theory Test for Car Drivers*. Ford, U.K.: DVSA Books, 2024. [Online]. Available: https://www.dvsabooks.com/car-theory-book

[30] DrivingInstructor. (2024). *Uk Driving Theory Test Practice Questions and Answers*. [Online]. Available: https://www.drivinginstructorwebsites.co.uk/uk-driving-theory-test-practice-questions-and-answers

[31] DeepSeek-AI et al., "DeepSeek-r1: Incentivizing reasoning capability in LLMs via reinforcement learning," 2025, *arXiv:2501.12948*.

[32] B. Li et al., "Driving everywhere with large language model policy adaptation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 14948–14957.

[33] C. Sima et al., "DriveLM: Driving with graph visual question answering," in *Proc. Eur. Conf. Comput. Vis.*, Nov. 2024, pp. 256–274.

[34] P. Lewis et al., "Retrieval-augmented generation for knowledge-intensive NLP tasks," in *Proc. Adv. Neural Inf. Process. Syst.*, Jan. 2020, pp. 9459–9474.

[35] O. Zheng, M. Abdel-Aty, D. Wang, C. Wang, and S. Ding, "TrafficSafetyGPT: Tuning a pre-trained large language model to a domain-specific expert in transportation safety," 2023, *arXiv:2307.15311*.