What can VLMs Do for Zero-shot Embodied Task Planning?

Xian Fu [1] Min Zhang [1] Jianye Hao [1] Peilong Han [1] Hao Zhang [1] Lei Shi [1] Hongyao Tang [2]

Abstract

Recent advances in Vision Language Models (VLMs) for robotics demonstrate their enormous potential. However, the performance limitations of VLMs for embodied task planning, which require high precision and reliability, remain ambiguous, greatly constraining their potential application in this field. To this end, this paper provides an in-depth and comprehensive evaluation of VLM performance in zero-shot embodied task planning. Firstly, we develop a systematic evaluation framework encompassing various dimensions of capabilities essential for task planning for the first time. This framework aims to identify the factors that constrain VLMs in producing accurate task plans. Based on this framework, we propose a benchmark dataset called ETP-Bench to evaluate the performance of VLMs on embodied task planning. Extensive experiments indicate that the current state-of-the-art VLM, GPT-4V, achieves only 19% accuracy in task planning on our benchmark. The main factors contributing to this low accuracy are deficiencies in spatial perception and object type recognition. We hope this study can provide data support and inspire more specific research directions for future robotics research.

1. Introduction

The remarkable progress of VLMs (Zhu et al., 2023; Liu et al., 2023; Li et al., 2023b; Dai et al., 2023; OpenAI, 2023; Li et al., 2023c; Hu et al., 2024) has sparked a surge of interest in advancing their application in robotics. Currently, the mainstream approaches can be categorized into two branches. One approach involves training specific models for low-level robotic control using specific robotics data (Jiang et al., 2023; Brohan et al., 2022; 2023; Li et al., 2023d; Shah et al., 2023; Li et al., 2023a; Driess et al., 2023). The other approach employs off-the-shelf models like GPT-4V (OpenAI, 2023) as a high-level task planners and then leverage pre-trained skills to accomplish all subtask goals from these planners. This method offers flexible adaptation to various robotics scenarios in a zero-shot manner (Wake et al., 2023; Hu et al., 2023; Chen et al., 2023; Mu et al., 2024; Sun et al., 2023; Wang et al., 2024). Rooted in data-driven algorithms and specific scenarios, the former merely characterizes abstract features of limited acquisition data. Consequently, they struggle to distill nuanced insights into robotics manipulation-related knowledge from data and lack effective causal reasoning. In contrast, the latter avoids expensive data collection costs while achieving the model generalization pursued in robotics fields, thanks to the surprising reasoning and generalization capabilities of VLMs. Nevertheless, it is worth noting that these methods generally only involve straightforward application of VLMs without a deep dive into their performance in the context of embodied task planning. The lack of this exploration hinders further refinement of these approaches.

To address the issues above, leveraging the characteristics of embodied task planning, we propose a systematic evaluation framework that encapsulates four key supporting capabilities: object understanding, spatio-temporal perception, task understanding, and embodied reasoning. The framework aims to precisely analyze the specific performance of VLMs in embodied task planning. Furthermore, we propose the **E**mbodied **T**ask **P**laning **Bench**mark **(ETP-Bench)** aligned with this framework to evaluate the performance limitations of VLMs.

In summary, the main contributions of our work are as follows: (1) we propose a systematic evaluation framework tailored for embodied task planning for the first time, which provides effective guidance for improving VLM-based task planners. (2) we introduce ETP-Bench, a benchmark comprising over 1800 high-quality human-annotated test cases covering 100 embodied tasks. This benchmark will be made open source to encourage advancements in the field. (3) we evaluate six advanced VLMs and the empirical results indicate that the spatial perception and object type recognition capabilities of VLMs are critical for generating accurate task plans.

[1]College of Intelligence and Computing, Tianjin University, Tianjin, China [2]Montreal Institute of Learning Algorithms (MILA), Quebec, Canada. Correspondence to: Jianye Hao <jianye.hao@tju.edu.cn>.

Multi-modal Foundation Model meets Embodied AI Workshop at the 41st International Conference on Machine Learning, Vienna, Austria, 2024.

2. Related Work

VLMs for Task and Motion Planning Task and Motion Planning (TAMP) (Kaelbling & Lozano-Pérez, 2011; Garrett et al., 2021; Guo et al., 2023) is a framework that addresses complex, long-horizon planning tasks by integrating high-level reasoning with low-level motion planning. Recent advancements incorporate off-the-shelf VLMs (Wake et al., 2023; Hu et al., 2023; Wang et al., 2024) to enhance the framework. Their main difference lies in the design specifics of the task planning pipeline, such as whether it integrates an affordance analyzer (Wake et al., 2023), or integrates perceptual information and visual feedback (Hu et al., 2023), etc.

Benchmarking for VLMs As a significant milestone in the field of artificial intelligence, VLMs with powerful visual perception capabilities have attracted considerable attention from academia and industry. Researchers are currently exploring the capabilities of VLMs across various domains, such as commonsense tasks (Fu et al., 2023a; Yang et al., 2023; Xu et al., 2023), vision tasks (Fu et al., 2023b;a; Cao et al., 2023; Wu et al., 2023), autonomous driving (Wen et al., 2023; Mao et al., 2023), and robotics (Hu et al., 2023; Chen et al., 2023; Majumdar et al., 2024). Similar to our work, EgoPlan-Bench (Chen et al., 2023) evaluated the performance of VLMs in one-step task planning within kitchen scenarios, while OpenEQA (Majumdar et al., 2024) assessed VLMs for environmental understanding by embodied question answering. Compared to existing evaluation efforts, our work offers three key advantages: ① our evaluation framework and benchmark are purposefully designed for embodied task planning, ensuring they accurate assessment of VLMs in task planning. ② we encompass a diverse set of scenarios beyond the kitchen, facilitating a comprehensive evaluation of VLMs' task planning capabilities across different settings. ③ the capability dimensions we evaluate are more comprehensive and closely aligned with embodied task planning, covering a wide range of test case types from embodied question answering to complex task planning.

3. Evaluation Framework

In this section, we introduce our evaluation framework, which consists of four key capabilities—three fundamental capabilities, namely Object Understanding, Spatio-Temporal Perception, and Task Understanding—and an advanced capability, Embodied Reasoning. Each of these capabilities (excluding Emdbodied Reasoning) is further divided into several aspects.

Object Understanding Object understanding involves the capability to recognize the *Type* and *Property* of objects. Accurate identification of object types and properties is essential for correct task planning. For instance, in tasks such as cleaning an oven, incorrect recognition of the oven and its properties would prevent VLMs from generating accurate task plans. Moreover, this mistake could pose safety risks due to improper interactions with misidentified objects. Details of each considered property and representative objects are listed in Tab. 4 in Appendix.

Spatio-temporal Perception Compared to understanding static objects, VLMs face greater challenges in perceiving the temporal and spatial aspects of the environment. In our benchmark, for *Spatial* perception, we primarily access VLMs' spatial reasoning capabilities, including distance, and the recognition of spatial relationships between objects. For *Temporal* perception, our focus is on evaluating VLMs' ability to recognize the chronological sequence of task progress, and predict the effects of actions on the environment. Enhancing this advanced perception capability can help to better understand the environment and thereby improve task planning.

Task Understanding In the context of robotic tasks, we evaluate whether VLMs understand robotic tasks from four aspects: selecting task-relevant objects, comprehending task-related object manipulation knowledge, choosing appropriate action sequences, and assessing task completion conditions. For brevity, we refer to these aspects as *Relevant*, *Operation*, *Sequence*, and *Goal*, respectively.

Embodied Reasoning While the aforementioned three fundamental capabilities are essential for completing embodied task planning, assessing only these aspects does not fully reveal the performance constraints of VLMs in such tasks. Therefore, we introduce an advanced capability: embodied reasoning. This capability evaluates VLMs' task planning based on a general task description and visual information. General task descriptions typically provide minimal information, posing a challenging task planning problem where embodied reasoning proves invaluable in assessing the performance limitations of VLMs.

4. ETP-Bench Benchmark

In this section, we provide a detailed description of our benchmark, which consists of the data source, instruction design, and evaluation metrics.

4.1. Data Source

We chose VirtualHome (Puig et al., 2018) and BEHAVIOR-100 (Srivastava et al., 2022) as the data source for our benchmark for their complexity and realism. More detailed information about these two simulation environments can be found in Appendix B.1.

Tab. 1 presents the statistics of ETP-Bench. Our benchmark dataset encompasses a total of 1875 test cases, wherein 1605

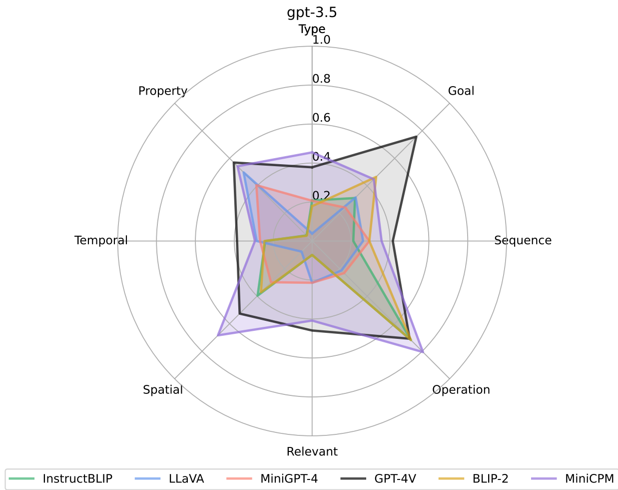

Figure 1. Radar chart of different VLMs' scores on ETP-Bench.

Table 1. Statistics of ETP-Bench benchmark.

Capabilities	**Aspects**	**Number**
Object Understanding	Type	288
	Property	259
Spatio-temporal Perception	Temporal	100
	Spatial	488
Task Understanding	Relevant	100
	Operation	100
	Sequence	170
	Goal	100
Embodied Reasoning	-	270
Total	-	**1875**

cases are dedicated to evaluating foundational capabilities, and 270 cases reserved for evaluating Embodied Reasoning. The questions and answers in our benchmark cover diverse scenarios for the three fundamental capabilities, with typical QA examples provided in Figure 2 in the Appendix. For Embodied Reasoning, a typical example is also provided in Figure 13. To ensure the data quality, our ETP-Bench benchmark is manually annotated by experienced AI researchers.

4.2. Instruction Design

In our benchmark, the complete text prompt consists of two parts: task instructions (general description of an aspect) and specific questions, as illustrated in Fig. 12 in Appendix. We meticulously designed a total of 156 text prompt types, to ensure diversity in prompts and objectivity in evaluation.

4.3. Evaluation Metrics

For fundamental capabilities, we employ a widely used metric, called *GPT-3.5* here, which facilitates semantic evaluation of the open-vocabulary outputs from VLMs, enabling more objective scores and supporting automatic and rapid evaluations. Specifically, we compare the prediction results with groundtruth, by prompting GPT-3.5 (Brown et al., 2020). We further validate the high consistency between the *GPT-3.5* and human assessment through comparative analysis. Details of the prompts and comparisons can be found in Appendix D.2. For embodied reasoning, we rely on the scores given by human annotators, mainly because its results are more intricate and challenging to evaluate automatically.

5. Experiments

We present the empirical results in two parts: Fundamental Capabilities (in Sec. 5.1), and Embodied Reasoning (in Sec. 5.2). We briefly describe the additional experiments in Appendix (in Sec. 5.3).

5.1. Results on Fundamental Capabilities

For fundamental capabilities, we evaluated GPT-4V alongside open-source VLMs: InstructBLIP (Dai et al., 2023), BLIP-2 (Li et al., 2023b), MiniGPT-4 (Zhu et al., 2023), LLaVA-1.5 (Liu et al., 2023), and MiniCPM (Hu et al., 2024) . The scores are presented in Tab. 2 and the radar chart (Fig. 1). From these results, we draw the following general conclusions: ① While GPT-4V outperforms the other models on average, MiniCPM closely approaches GPT-4V and surpasses the other open-source VLMs in all aspects except *Goal*. ② Across the three fundamental capabilities, VLMs generally perform poorly in object understanding and spatio-temporal perception. ③ All models sometimes output additional information that does not conform to the output format requirements, with MiniGPT-4 being the most severe. We analyze each fundamental capability as follows:

Object Understanding ① The MiniCPM model excels among the others in recognizing object types with an accuracy of 45.5%. ② In terms of understanding object properties, GPT-4V leads with 56.9% accuracy, followed closely by MiniCPM with 54.2%. This suggests that MiniCPM is generally more effective in object recognition tasks, while GPT-4V excels in identifying object properties.

Spatio-temporal Perception ① MiniCPM stands out in spatial perception, achieving an accuracy of 68.4%. This indicates its strong capability in interpreting spatial rela-

Table 2. Evaluation Results. We present the accuracy scores of each model in each aspect. We highlight the best scores with **bold and underline**, and use **bold only** to mark the second best scores.

Model	Type	Property	Temporal	Spatial	Relevant	Operation	Sequence	Goal	Average
GPT-4V	**37.8%**	**<u>56.9%</u>**	**<u>39%</u>**	**52.7%**	**<u>46%</u>**	71%	**<u>41.5%</u>**	**<u>76%</u>**	**<u>52.6%</u>**
InstructBLIP	20.8%	4.1%	24%	39.7%	7%	**72%**	21.2%	31%	27.5%
BLIP-2	17.9%	3.9%	25%	37.2%	7%	**72%**	29.4%	**47%**	29.9%
MiniGPT-4	20.7%	40.5%	27%	29.9%	21%	23%	29.4%	24%	26.9%
LLaVA-1.5	3.8%	49.8%	28%	7.7%	21%	21%	26.2%	32%	23.7%
MiniCPM	**<u>45.5%</u>**	**54.2%**	**29%**	**<u>68.4%</u>**	**41%**	**<u>81%</u>**	**35.7%**	45%	**50.0%**

tionships and context, surpassing GPT-4V, which achieved 52.7% accuracy. ② However, all models struggle with temporal perception, with GPT-4V achieving the highest at 39%. This highlights a general difficulty in understanding time-related concepts and underscores the need for further improvements.

Task Understanding ① In terms of task understanding, the models exhibit varying performances. MiniCPM performs best in operation understanding with 81% accuracy and also performs reasonably well in sequence understanding with 35.7% accuracy. ② GPT-4V demonstrates strong performance in goal understanding with 76% accuracy, indicating its proficiency in recognizing the objectives of tasks. ③ However, all models face challenges in recognizing related objects, highlighting an ongoing difficulty in identifying objects relevant to specific tasks.

5.2. Results on Embodied Reasoning

For embodied reasoning, we choose GPT-4V to test due to its superior capabilities. We prompt GPT-4V to generate complete task plans, which are then manually evaluated. In our benchmark, we offer 270 embodied reasoning tasks, revealing a success rate of only 19% for GPT-4V. This underscores that even the most advanced VLM struggles as a zero-shot task planner and faces significant challenges in our evaluation benchmark, highlighting its inherent difficulty.

We further conducted experiments to explore the impact of additional information in embodied reasoning. As shown in Tab. 3, we select 50 representative tasks from the original ones. For each task, we gradually inject information related to some supporting capabilities into GPT-4V through prompts, and observe corresponding changes in success rates on task planning. Our findings are as follows: ① Injecting object information, spatial information, and task knowledge can positively influence task planning; ② compared to settings where only object information (+o) or object, spatial, and task knowledge (+o.+s.+k) are injected, injecting object and spatial information (+o.+s.) notably enhances the success rate of task planning.

Table 3. Results of gradually information injection. +o. means adding object information, +s. means adding spatial information, and +k. means adding task-related knowledge.

Embodied Reasoning	general	+o.	+o.+s.	+o.+s.+k.
Success Rate	22%	34%	72%	80%

5.3. Additional Experiments

Due to space constraints, we have included additional experiments in the Appendix. Sec. C.1 presents further results on the features and limitations of GPT-4V through qualitative analysis. Sec. C.2 further highlights the importance of spatial perception and object type recognition in embodied task planning with linear regression analysis.

6. Conclusion

In this paper, we introduce a systematic evaluation framework tailored for embodied task planning for the first time. Building upon this framework, we propose ETP-Bench, an evaluation benchmark at pushing the performance boundaries of VLMs. Empirical results consistently reveal GPT-4V's challenges in effectively completing embodied planning tasks, particularly in object type recognition and spatial perception. Our comprehensive analysis provides valuable insights to propel advancements in VLMs for embodied task planning.

Limitations and future work. We utilized virtual environments as data sources for easier data collection, which may raise concerns about the generalizability of our findings to more realistic scenarios. Evaluating a broader range of models represents a potential future endeavor. While the Embodied Reasoning task remains highly challenging for current models, planning all future steps in advance remains advantageous. Utilizing closed-loop control systems like ViLa (Hu et al., 2023) for testing task planning could offer a viable compromise. Furthermore, exploring additional QA formats such as correcting past behavior and predicting subsequent actions could further enrich the research.

Acknowledgements

The authors would like to thank the reviewers who gave very useful comments.

Impact Statements

This paper presents work whose goal is to advance the field of VLMs for embodied task planning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Brohan, A., Brown, N., Carbajal, J., Chebotar, Y., Dabis, J., Finn, C., Gopalakrishnan, K., Hausman, K., Herzog, A., Hsu, J., et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

Brohan, A., Brown, N., Carbajal, J., Chebotar, Y., Chen, X., Choromanski, K., Ding, T., Driess, D., Dubey, A., Finn, C., et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901, 2020.

Cao, Y., Xu, X., Sun, C., Huang, X., and Shen, W. Towards generic anomaly detection and understanding: Large-scale visual-linguistic model (GPT-4V) takes the lead. *CoRR*, abs/2311.02782, 2023. doi: 10.48550/ARXIV.2311.02782. URL https://doi.org/10.48550/arXiv.2311.02782.

Chen, B., Xu, Z., Kirmani, S., Ichter, B., Driess, D., Florence, P., Sadigh, D., Guibas, L., and Xia, F. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. *arXiv preprint arXiv:2401.12168*, 2024.

Chen, Y., Ge, Y., Ge, Y., Ding, M., Li, B., Wang, R., Xu, R., Shan, Y., and Liu, X. Egoplan-bench: Benchmarking egocentric embodied planning with multimodal large language models. *CoRR*, abs/2312.06722, 2023. doi: 10.48550/ARXIV.2312.06722. URL https://doi.org/10.48550/arXiv.2312.06722.

Dai, W., Li, J., Li, D., Tiong, A., Zhao, J., Wang, W., Li, B., Fung, P., and Hoi, S. Instructblip: Towards general-purpose vision-language models with instruction tuning. arxiv 2023. *arXiv preprint arXiv:2305.06500*, 2023.

Driess, D., Xia, F., Sajjadi, M. S., Lynch, C., Chowdhery, A., Ichter, B., Wahid, A., Tompson, J., Vuong, Q., Yu, T., et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023.

Fu, C., Chen, P., Shen, Y., Qin, Y., Zhang, M., Lin, X., Qiu, Z., Lin, W., Yang, J., Zheng, X., Li, K., Sun, X., and Ji, R. MME: A comprehensive evaluation benchmark for multimodal large language models. *CoRR*, abs/2306.13394, 2023a. doi: 10.48550/ARXIV.2306.13394. URL https://doi.org/10.48550/arXiv.2306.13394.

Fu, C., Zhang, R., Wang, Z., Huang, Y., Zhang, Z., Qiu, L., Ye, G., Shen, Y., Zhang, M., Chen, P., Zhao, S., Lin, S., Jiang, D., Yin, D., Gao, P., Li, K., Li, H., and Sun, X. A challenger to gpt-4v? early explorations of gemini in visual expertise. *CoRR*, abs/2312.12436, 2023b. doi: 10.48550/ARXIV.2312.12436. URL https://doi.org/10.48550/arXiv.2312.12436.

Garrett, C. R., Chitnis, R., Holladay, R., Kim, B., Silver, T., Kaelbling, L. P., and Lozano-Pérez, T. Integrated task and motion planning. *Annual review of control, robotics, and autonomous systems*, 4:265–293, 2021.

Guo, H., Wu, F., Qin, Y., Li, R., Li, K., and Li, K. Recent trends in task and motion planning for robotics: A survey. *ACM Computing Surveys*, 55(13s):1–36, 2023.

Hu, S., Tu, Y., Han, X., He, C., Cui, G., Long, X., Zheng, Z., Fang, Y., Huang, Y., Zhao, W., Zhang, X., Thai, Z. L., Zhang, K., Wang, C., Yao, Y., Zhao, C., Zhou, J., Cai, J., Zhai, Z., Ding, N., Jia, C., Zeng, G., Li, D., Liu, Z., and Sun, M. Minicpm: Unveiling the potential of small language models with scalable training strategies, 2024.

Hu, Y., Lin, F., Zhang, T., Yi, L., and Gao, Y. Look before you leap: Unveiling the power of GPT-4V in robotic vision-language planning. *CoRR*, abs/2311.17842, 2023. doi: 10.48550/ARXIV.2311.17842. URL https://doi.org/10.48550/arXiv.2311.17842.

Jiang, Y., Gupta, A., Zhang, Z., Wang, G., Dou, Y., Chen, Y., Fei-Fei, L., Anandkumar, A., Zhu, Y., and Fan, L. Vima: Robot manipulation with multimodal prompts. 2023.

Kaelbling, L. P. and Lozano-Pérez, T. Hierarchical task and motion planning in the now. In *2011 IEEE International Conference on Robotics and Automation*, pp. 1470–1477. IEEE, 2011.

Li, J., Gao, Q., Johnston, M., Gao, X., He, X., Shakiah, S., Shi, H., Ghanadan, R., and Wang, W. Y. Mastering robot manipulation with multimodal prompts through pretraining and multi-task fine-tuning, 2023a.

Li, J., Li, D., Savarese, S., and Hoi, S. BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, 2023b.

Li, J., Pan, K., Ge, Z., Gao, M., Zhang, H., Ji, W., Zhang, W., Chua, T.-S., Tang, S., and Zhuang, Y. Empowering vision-language models to follow interleaved vision-language instructions. *arXiv preprint arXiv:2308.04152*, 2023c.

Li, X., Liu, M., Zhang, H., Yu, C., Xu, J., Wu, H., Cheang, C., Jing, Y., Zhang, W., Liu, H., et al. Vision-language foundation models as effective robot imitators. *arXiv preprint arXiv:2311.01378*, 2023d.

Lin, C.-Y. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pp. 74–81, 2004.

Liu, H., Li, C., Li, Y., and Lee, Y. J. Improved baselines with visual instruction tuning. 2023.

Majumdar, A., Ajay, A., Zhang, X., Putta, P., Yenamandra, S., Henaff, M., Silwal, S., Mcvay, P., Maksymets, O., Arnaud, S., et al. Openeqa: Embodied question answering in the era of foundation models. In *2nd Workshop on Mobile Manipulation and Embodied Intelligence at ICRA 2024*, 2024.

Mao, J., Qian, Y., Zhao, H., and Wang, Y. Gpt-driver: Learning to drive with gpt. *arXiv preprint arXiv:2310.01415*, 2023.

Mu, Y., Zhang, Q., Hu, M., Wang, W., Ding, M., Jin, J., Wang, B., Dai, J., Qiao, Y., and Luo, P. Embodiedgpt: Vision-language pre-training via embodied chain of thought. *Advances in Neural Information Processing Systems*, 36, 2024.

OpenAI. Gpt-4 technical report. 2023. URL https://api.semanticscholar.org/CorpusID:257532815.

Puig, X., Ra, K., Boben, M., Li, J., Wang, T., Fidler, S., and Torralba, A. Virtualhome: Simulating household activities via programs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8494–8502, 2018.

Shah, R., Martín-Martín, R., and Zhu, Y. Mutex: Learning unified policies from multimodal task specifications. *arXiv preprint arXiv:2309.14320*, 2023.

Shao, W., Hu, Y., Gao, P., Lei, M., Zhang, K., Meng, F., Xu, P., Huang, S., Li, H., Qiao, Y., et al. Tiny lvlm-ehub: Early multimodal experiments with bard. *arXiv preprint arXiv:2308.03729*, 2023.

Srivastava, S., Li, C., Lingelbach, M., Martín-Martín, R., Xia, F., Vainio, K. E., Lian, Z., Gokmen, C., Buch, S., Liu, K., et al. Behavior: Benchmark for everyday household activities in virtual, interactive, and ecological environments. In *Conference on Robot Learning*, pp. 477–490. PMLR, 2022.

Sun, J., Zhang, Q., Duan, Y., Jiang, X., Cheng, C., and Xu, R. Prompt, plan, perform: Llm-based humanoid control via quantized imitation learning, 2023.

Wake, N., Kanehira, A., Sasabuchi, K., Takamatsu, J., and Ikeuchi, K. Gpt-4v (ision) for robotics: Multimodal task planning from human demonstration. *arXiv preprint arXiv:2311.12015*, 2023.

Wang, J., Wu, Z., Li, Y., Jiang, H., Shu, P., Shi, E., Hu, H., Ma, C., Liu, Y., Wang, X., et al. Large language models for robotics: Opportunities, challenges, and perspectives. *arXiv preprint arXiv:2401.04334*, 2024.

Wen, L., Yang, X., Fu, D., Wang, X., Cai, P., Li, X., Ma, T., Li, Y., Xu, L., Shang, D., Zhu, Z., Sun, S., Bai, Y., Cai, X., Dou, M., Hu, S., Shi, B., and Qiao, Y. On the road with gpt-4v(ision): Early explorations of visual-language model on autonomous driving. *CoRR*, abs/2311.05332, 2023. doi: 10.48550/ARXIV.2311.05332. URL https://doi.org/10.48550/arXiv.2311.05332.

Wu, W., Yao, H., Zhang, M., Song, Y., Ouyang, W., and Wang, J. Gpt4vis: What can GPT-4 do for zero-shot visual recognition? *CoRR*, abs/2311.15732, 2023. doi: 10.48550/ARXIV.2311.15732. URL https://doi.org/10.48550/arXiv.2311.15732.

Xu, P., Shao, W., Zhang, K., Gao, P., Liu, S., Lei, M., Meng, F., Huang, S., Qiao, Y., and Luo, P. Lvlm-ehub: A comprehensive evaluation benchmark for large vision-language models. *CoRR*, abs/2306.09265, 2023. doi: 10.48550/ARXIV.2306.09265. URL https://doi.org/10.48550/arXiv.2306.09265.

Yang, Z., Li, L., Lin, K., Wang, J., Lin, C., Liu, Z., and Wang, L. The dawn of lmms: Preliminary explorations with gpt-4v(ision). *CoRR*, abs/2309.17421, 2023. doi: 10.48550/ARXIV.2309.17421. URL https://doi.org/10.48550/arXiv.2309.17421.

Zhu, D., Chen, J., Shen, X., Li, X., and Elhoseiny, M. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023.

A. Appendix Overview

The appendix includes the following content:

1. Details information about the proposed ETP-Bench benchmark. (Sec. B)
2. Additional experimental results (Sec. C)
3. Full prompt and result for each dimension and *GPT-3.5* metric (Sec. D)
4. Detailed information of VLMs (Sec. E).
5. Accessibility (Sec. F)

B. Details of ETP-Bench Benchmark

B.1. BEHAVIOR-100 and VirtualHome

BEHAVIOR-100 (Srivastava et al., 2022) simulates 100 everyday household tasks for embodied AI, with a distribution similar to the full space of simulatable tasks in the American Time Use Survey (ATUS). It aims to create realistic and complex scenarios for AI testing. It introduce a predicate logic-based language (BDDL) for defining tasks, and possesses simulator-agnostic features for versatile application. Despite its comprehensive setup, state-of-the-art AI still struggles with the benchmark's challenges.

VirtualHome (Puig et al., 2018) uses "programs" to model complex family activities, which are a series of actions and interactions executed by agents. These programs are crowdsourced through a game-like interface, originating from natural language and video input. VirtualHome is implemented in Unity3D, enabling human agents to perform tasks in simulated environments, which helps create video datasets for training video comprehension models and demonstrating task execution based on language descriptions.

For BEHAVIOR-100, the vision input (images) for each test case comes from 100 videos provided by the project homepage. For VirtualHome, we first design some tasks through "programs", and save the videos for choosing vision input.

B.2. Data Details

For Embodied Reasoning, we have provided at least one question for each task; For the other capabilities (Fundamental Capabilities), we let human annotators decide whether to construct the corresponding data and the number of data. Note that for the Embodied Reasoning dataset, only the question section is explicitly provided because the answer may take multiple forms, so it is not specifically provided.

Details of the data for Fundamental Capabilities We present typical QA pairs examples for Fundamental capabilities in Fig. 2. Among the subaspects of QA pairs, the object type, and relevant object are designed as open-ended questions, while the others are designed as choice questions.

Details of Embodied Reasoning data From Fig. 13, we can see the whole prompt structure for task planning. First, the vision input usually consists of several frames of the task. Among them, the first frame indicates the current position of the robot, and the others displays other task-related objects. Then, we carefully designed the text prompt, which mainly includes three parts: task instruction, task goal, and predefined action function. We organize the natural language description of the task goals according to the goals of each task (from the bddl file of each task). Note Tab. 5 details all action functions. Finally, we define examples of VLMs task planning results, which include identification results of task-related object types and spatial relationships between objects, as well as complete task planning results.

B.3. Full object properties and action list

We list the full **object properties** in Tab. 4, and the full **action list** in Tab. 5.

Table 4. List of object properties.

Property	Annotation	example
breakable	Mark if the object is brittle, that is, it can be broken into smaller pieces by a human dropping it on the floor.	wine bottle, room light
cleaningTool	Is a [object] designed to clean things?	scrub brush
cookable	Can a [object] be cooked?	biscuit, pizza
grabbable	If an object has this attribute, it is usually lightweight and can be potentially grabbed and picked up by the robot.	apple, bottle, rag, plate
openable	Mark if the object is designed to be opened.	mixer, keg
sliceable	Can a [object] be sliced easily by a human with a knife?	sweet corn, sandwich
slicingTool	Can a [object] slice an apple?	blade, razor
toggleable	The object can be switched between a finite number of discrete states and is designed to do so.	hot tub, light bulb
waterSource	where you can get water	sink

C. Additional Experimental Results

C.1. Further qualitative analysis of GPT-4V

Object Understanding GPT-4V performs poorly in object type recognition, achieving only a 38% accuracy across 288 QA pairs, which we attribute to poor perspective (with the best egocentric perspective we can obtain, objects are still difficult to identify), limitations of simulation environments (low resolution or poor realism), and limitations in GPT-4V itself. An interesting phenomenon is that GPT-4V scores higher in object properties than in object type recognition , indicating that GPT-4V can judge the objects' properties without accurately identifying their types, that is, object types are not a strict prerequisite for object properties. From Fig. 14, we can see that GPT-4V sometimes incorrectly recognizes the type and properties of objects.

Spatio-temporal Perception Previous works often mentioned that GPT-4V lacks spatial understanding ability (Wen et al., 2023; Chen et al., 2024), which is consistent with our evaluation results. In our experiments, we found that GPT-4V often fail to perceive distance and relative position between objects (e.g., left and right) given 2D images. (as in Fig. 15) Also, we found that GPT-4V performs poorly in terms of temporal perception as well, which is reflected in its inability to accurately sort images according to the task completion process. (as in Fig. 16)

Task Understanding GPT-4V also lacks understanding of tasks, mainly manifested in the inability to accurately select task related objects and select reasonable action execution sequences (see Fig. 17), which are essential knowledge for task planning.

Embodied Reasoning We analyzed the corresponding results and found some common issues: firstly, task related object recognition faces challenges, including insufficient recognition, incorrect recognition, and chaotic spatial relationships when there are a large number of objects. Secondly, the system's spatiotemporal perception ability is not strong, manifested in insufficient understanding of the current position, as well as the poor ability to perceive whether an object is within the operating range, which often leads to providing unnecessary or missing navigation actions. In terms of action selection, the system may wrongly or repeatedly call actions, or perform actions beyond predefined ones. In addition, GPT-4V may give some difficult to execute conditional branch statements, or assume a condition when identifying key information unclear. These issues comprehensively affect the efficiency and accuracy of the system, and require specific optimization measures to improve. Finally, GPT-4V cannot provide accurate planning for tasks involving too many objects.

Fig. 13 shows the whole task planning results from GPT-4V for the task preserving food. GPT-4V identified the wrong object type "bowl_of_fruit", which does not exist in this task. Regarding task planning, the step after *place_inside (potato_chips_package, cupboard)* is to directly grasp the chocolate bar far from the cupboard. The step *navigate_to(dinning table)* is missing. We have marked multiple similar errors in red. In contrast, GPT-4V demonstrate better task understanding capability and the results have no irrelevant objects or incorrect task operation knowledge. The results show that object type recognition capabilities and spatiotemporal perception capabilities are crucial to make correct task planning.

Table 5. List of actions for task planning.

Action	Annotation
navigate_to(arg1)	Navigate to the arg1, which can be a object or a room. If it's a object, you should get to a place where arg1 is reachable for the robot.
grasp(arg1):	Grasp arg1. Preconditions: arg1 is within reachable distance and no object is currently held. Postconditions: arg1 is being held.
place_onTop(arg1, arg2):	Place arg1 on top of arg2. Preconditions: arg1 is currently being held, and arg2 is reachable. Postconditions: arg1 is put on top of arg2.
place_inside(arg1, arg2):	Place arg1 inside of arg2. Preconditions: arg1 is currently being held, and arg2 is reachable. Postconditions:arg1 is put inside of arg2.
place_under(arg1, arg2):	Place arg1 under arg2. Preconditions: arg1 is currently being held, and arg2 is reachable. Postconditions: arg1 is put under arg2.
place_onLeft(arg1, arg2):	Place arg1 on left of arg2. Preconditions: arg1 is currently being held, and arg2 is reachable . Postconditions: arg1 is put on left of arg2.
place_onRight(arg1, arg2):	Place arg1 on right of arg2. Preconditions: arg1 is currently being held, and arg2 is reachable . Postconditions: arg1 is put on right of arg2.
open(arg1):	Open arg1. Preconditions: Arg1 is closed, and arg1 is reachable. Postconditions: Arg1 is open.
close(arg1):	Close arg1. Preconditions: Arg1 is open, and arg1 is reachable. Postconditions: Arg1 is closed.
slice(arg1):	Slice arg1, the item needs to be placed on the countertop. Preconditions: Arg1 is not sliced, and arg1 is reachable. Postconditions: Arg1 is sliced.
wipe(arg1, arg2):	Wipe across the surface of arg2 with arg1. Preconditions: Arg1 is currently being held, and arg2 is reachable. Postconditions: Arg1 continues to be held, arg2 holds state unchanged.
wait(arg1):	Wait for arg1 seconds. Preconditions: None. Postconditions: arg1 second(s) has(have) passed.
toggle(arg1):	Press the button of arg1 to turn it on or off, Preconditions: Arg1 is open or closed, and arg1 is reachable. Postconditions: Arg1 is closed or open.

Table 6. Linear regression coefficients for each aspects.

Aspect	Type	Property	Temporal	Spatial	Relevant	Operation	Sequence	Goal
Coefficient	0.14	0.01	-0.1	0.4	-0.04	0.07	0.09	0.06

C.2. Linear regression analysis of the impact of each aspects on task planning

In this section, we will use linear regression coefficients as proxies for the impact of various factors on embodied reasoning (task planning). Specifically, for each robotic task, we use the binary result of embodied reasoning as the dependent variable and mark successful cases as 1 and failed cases as 0. Take the basic capabilities corresponding to all aspects as a multivariate independent variable, and use their accuracy as the value. The obtained coefficients are shown in the Tab. 6. It can be concluded that spatial perception is the most important for embodied reasoning, with a coefficient of 0.4, followed by 0.14 for object recognition, and those of other aspects are relatively closer to 0, indicating that their impact on embodied reasoning is not significant. Note that some aspects, such as the coefficient of time perception, are negative, which does not necessarily indicate that a good ability in that aspect will have a negative impact on embodied reasoning. It is more likely that this is because the planning success rate is low, and many times when planning fails, GPT-4V can also get relative high scores on these aspects.

D. Full prompt example for Each dimension and *GPT-3.5* metric

D.1. Full prompt example

Regarding the object type and object property test cases, we ask about the object types and object properties in the red box or marked with numbers respectively. When an object contains multiple properties, we require the properties of the object to be output sequentially in the order of the properties list.

For each capability dimension of ETP-Bench, we have provided the corresponding prompts and results of different models

Table 7. Evaluation metric comparison. *Human* means using scores given by several human annotators, *GPT-3.5* metric utilizes GPT-3.5 for automated evaluation, and *Rouge-L* is a rule-based metric based on the longest common subsequence.

Metric	Type	Property	Temporal	Spatial	Relevant	Operation	Sequence	Goal
Human	35.3	40.8	29.4	69.0	27.0	77.5	38.1	77.0
GPT-3.5	37.8	56.9	38.6	52.7	45.9	70.9	41.5	75.7
Rouge-L	19.8	61.4	62.4	38.8	58.5	76.7	54.5	77.2

below:

The full prompt and results on ***object type*** is shown in Fig. 4. For this aspect, we have the model recognize one or more objects enclosed by numerical annotations or boxes.

The full prompt and results on ***property*** is shown in Fig. 5. For this aspect, like *object type*, we have the model recognize the properties of single objects enclosed by numerical annotations or boxes. For the full property list, refer to Tab. 4.

The full prompt and results on ***spatial perception*** is shown in Fig. 6.

The full prompt and results on ***temporal perception*** is shown in Fig. 7. For this aspect, we have the model sort the task process in chronological order, or predict the success of actions based on visual observations.

The full prompt and results on ***relevant object*** is shown in Fig. 8. For this aspect, we have the model recognize one object related to the task, such as using plates to load food.

The full prompt and results on ***operation*** is shown in Fig. 9. For this aspect, we let the model select appropriate actions based on current observations and tasks. For the full property list, refer to Tab. 5.

The full prompt and results on ***sequence*** is shown in Fig. 10. For this aspect, we have the model select appropriate action sequences based on current observations and tasks (there may be multiple correct answers).

The full prompt and results on ***goal*** is shown in Fig. 11. For this aspect, we let the model judge whether the task objectives have been achieved based on current observations and tasks.

D.2. *GPT-3.5* metric

We hope to evaluate whether VLMs can understand and answer the question, rather than outputting results that are completely consistent with the groundtruth, which leads us to seek help from the model-based method (Shao et al., 2023) to evaluate the open-vocabulary output of VLMs, namely *GPT-3.5* metric in Sec. 4.3. The evaluation criteria is specified through the prompts shown in Fig. 3. Unlike (Shao et al., 2023), we did not provide specific questions as we found that this can sometimes lead to misunderstandings of the model.

Here, we further demonstrate that GPT-3.5 metric aligns better with human evaluation results compared to rule-based metrics (Lin, 2004). Tab. 7 displays the results. The ICC1 (Intraclass Correlation Coefficient 1) values of 0.808 for GPT-3.5 metric and 0.402 for Rouge-L metric indicate that compared to rule-based methods like Rouge-L, evaluation with GPT-3.5 has higher consistency with human evaluation.

E. Detailed Information of VLMs

We have provided the specific versions of each model used in this paper in Table 8.

GPT-4V (OpenAI, 2023) We use the *gpt-4-vision-preview* version of GPT-4V(ision).

InstructBLIP (Dai et al., 2023) We use the *Flan-T5-XL* version of InstructBLIP.

BLIP-2 (Li et al., 2023b) We use the *Flan-T5-XL* version of BLIP-2.

MiniGPT-4 (Zhu et al., 2023) We use the *Vicuna-7B* version of MiniGPT-4.

LLaVA-1.5 (Liu et al., 2023) We use the *LLaMA-7B* version of LLaVA-1.5.

Table 8. Model version.

Model	Version
GPT-4V	gpt-4-vision-preview
InstructBLIP	Flan-T5-XL
BLIP-2	Flan-T5-XL
MiniGPT-4	Vicuna-7B
LLaVA-1.5	LLaMA-7B
MiniCPM	MiniCPM-Llama3-V-2_5

MiniCPM (Hu et al., 2024) We use the *MiniCPM-Llama3-V-2_5* version of MiniCPM.

F. Accessibility

URL to dataset The proposed ETP-Bench benchmark is available at: https://huggingface.co/datasets/cccccyan/ETP-Bench.

Author Statement We accept complete accountability for any infringement of rights that may arise during the utilization or dissemination of the data presented in this work. We commit to undertaking necessary measures, including the modification or deletion of any data involved in such violations, to address these concerns promptly. The data provided is designed for academic purposes.

License ETP-Bench is released under the MIT License.

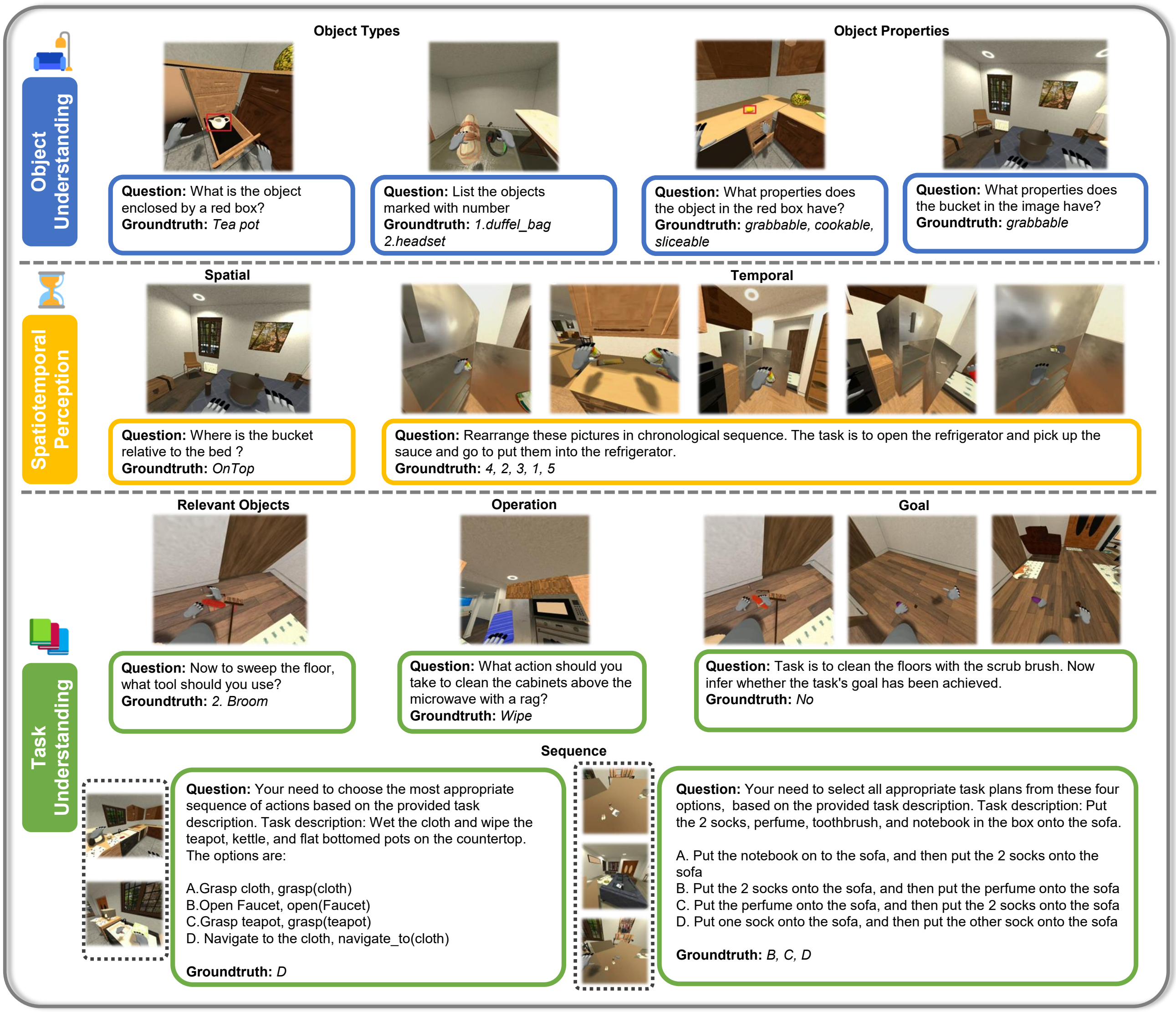

Figure 2. QA examples from our ETP-Bench benchmark, comprising 3 distinct subsets: Object Understanding, Spatio-temporal Perception, Task Understanding.

Task instance:

Question: I will give you a picture. For each picture, you need to identify the name of the object according to the requirements of the question. Neglect the robot, robot arm, and robot hand when answering. What is the object enclosed by a red box? Answer with its name only, like 'Apple'. Your answer is:

Groundtruth: *Hat*

Prompt:
Now, you will be presented with a correct response, and a student's answer to a question about the type of objects. Your job is to compare the student's answer to the correct one and assign a score based on the following rules: If the student's answer is semantically correct, give it a score of '1'. (The student's answer is very close to the correct answer, for example, they are all objects of the same type, with similar appearance and purposes) If the answer is incorrect, give it a '0'. If the answer contains correct and incorrect objects, give it a '0.5'. If the answer is correct but contains the other information for further correct and relevant explaination, assign it a '1'. Begin your evaluation with an 'Assessment:' paragraph, where you elaborate on your thought process. Conclude with 'Final Score: 1(or 0, or 0.5)', which is your final judgement. Output in JSON format. For instance: {\"Assessment\": \"xxxxx\", \"Final Score\": \"1(or 0, or 0.5)\"}. The correct response and student's answer is provided below.
The Student's Answer is: "Fedora"
The Correct Answer is: "Hat"

Output:

```
{
 "Assessment": " The student's answer 'fedora' is semantically correct as it is a type of hat. Although 'hat' is the general category, 'fedora' falls under this category, hence the student's response is very close to the correct answer. Therefore, the score will be 1. ",
 "Final Score": "1"
}
```

Figure 3. The full prompt and result of *GPT-3.5* metric. We provide model predictions and groundtruth answers and the GPT outputs scores. We highlight the most relevant task information in blue, and the effectiveness of *GPT-3.5* metric in green.

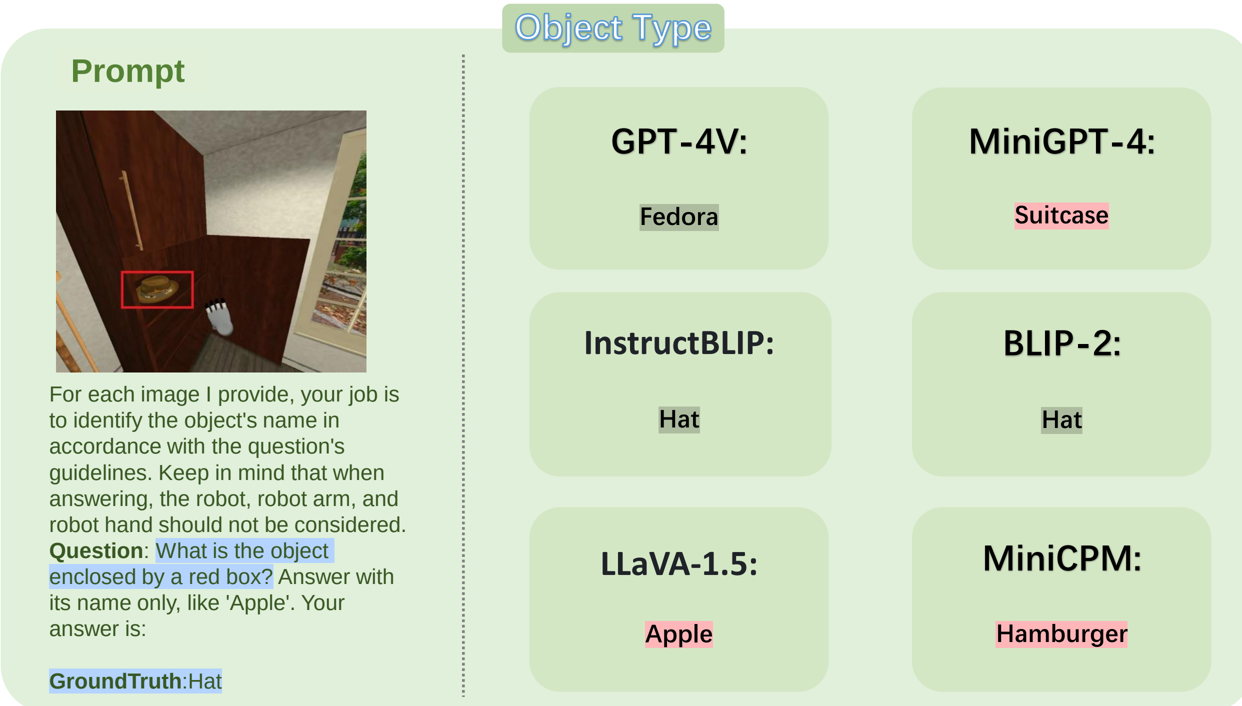

Figure 4. Full prompt and results on Object Type.

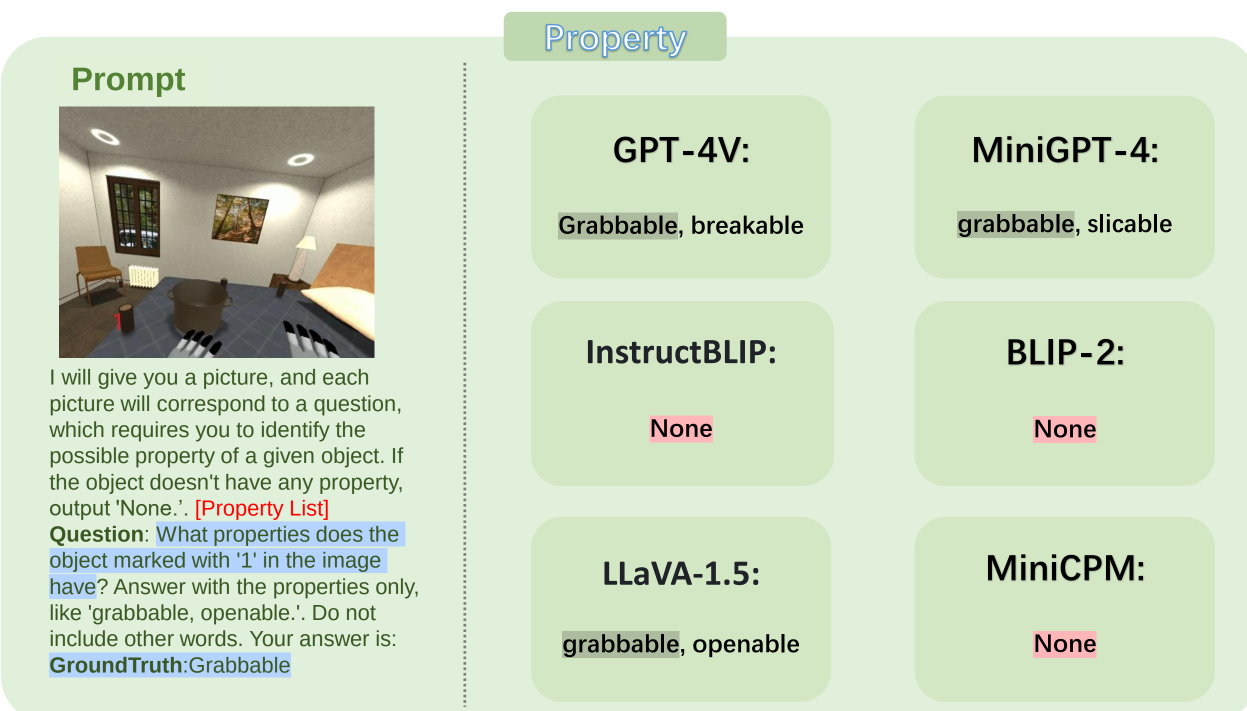

Figure 5. Full prompt and results on Property. For the full property list, refer to Tab. 4.

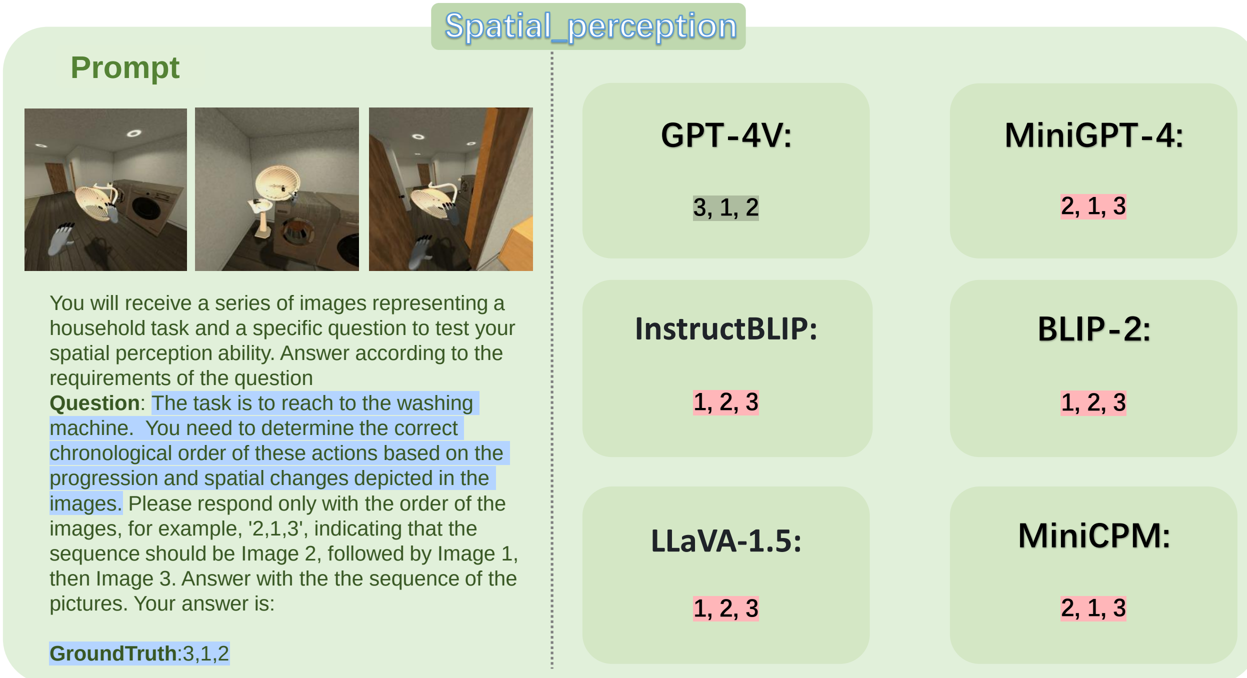

Figure 6. Full prompt and results on Spatial Perception.

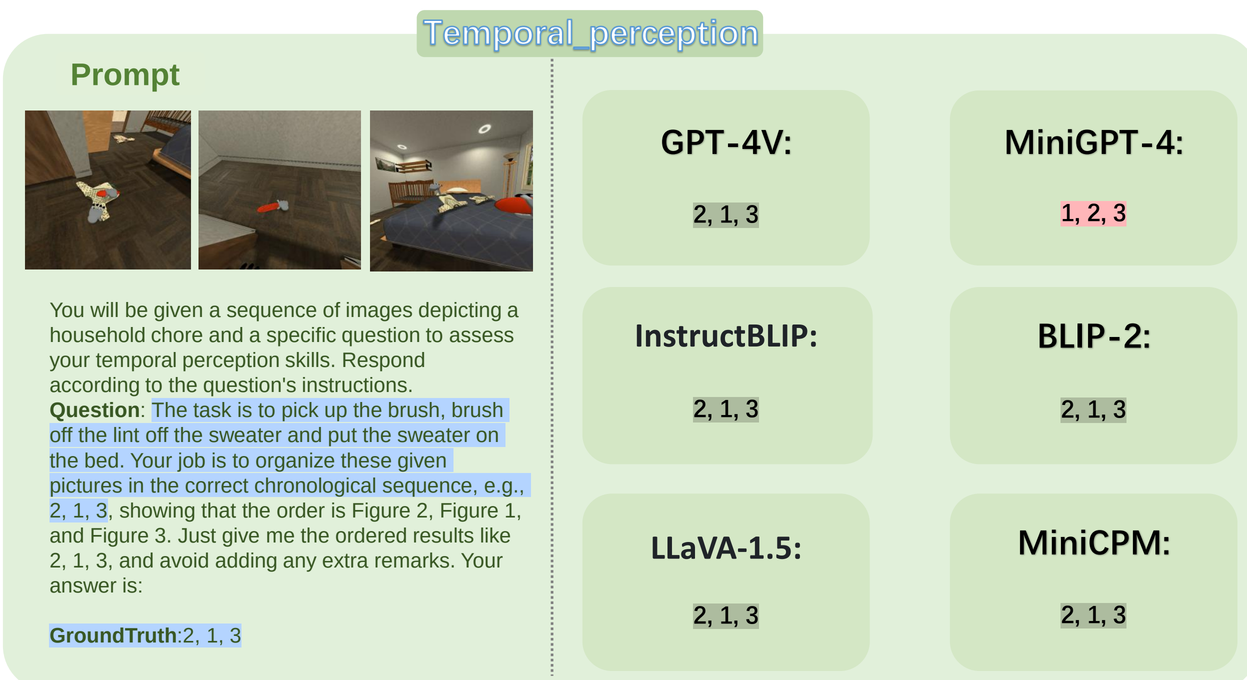

Figure 7. Full prompt and results on Time Perception.

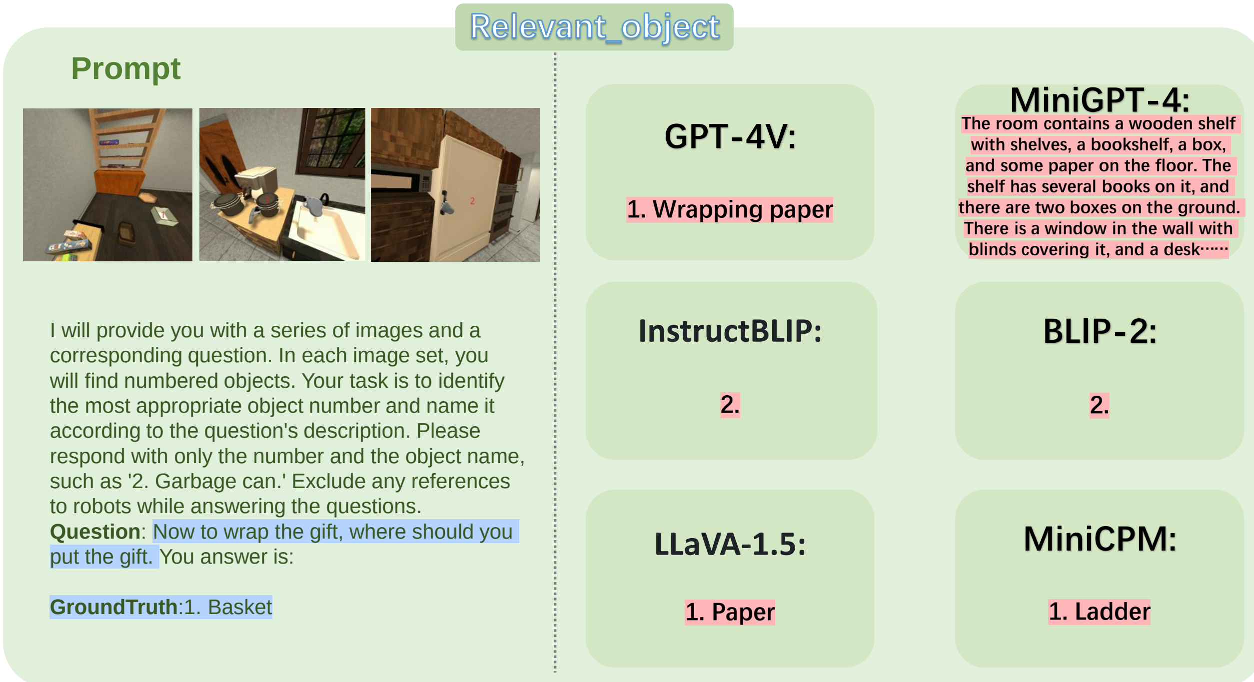

Figure 8. Full prompt and results on Relevant Object.

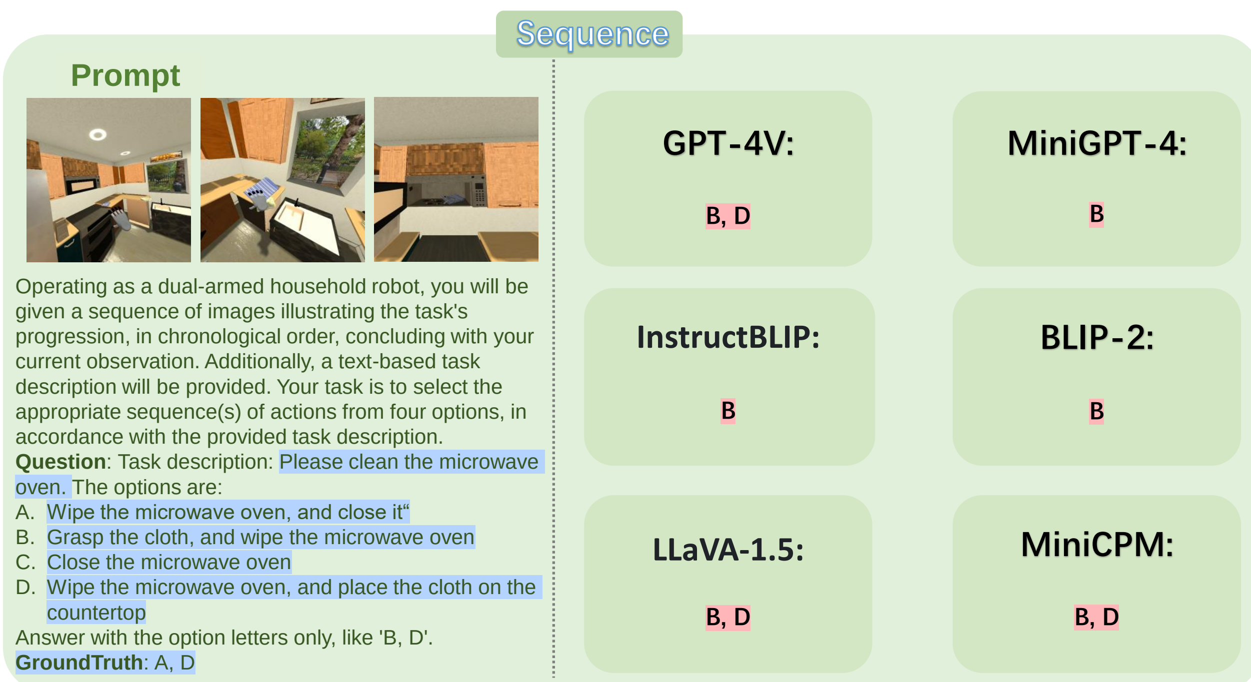

Figure 9. Full prompt and results on Operation. For the full action list, refer to Tab. 5.

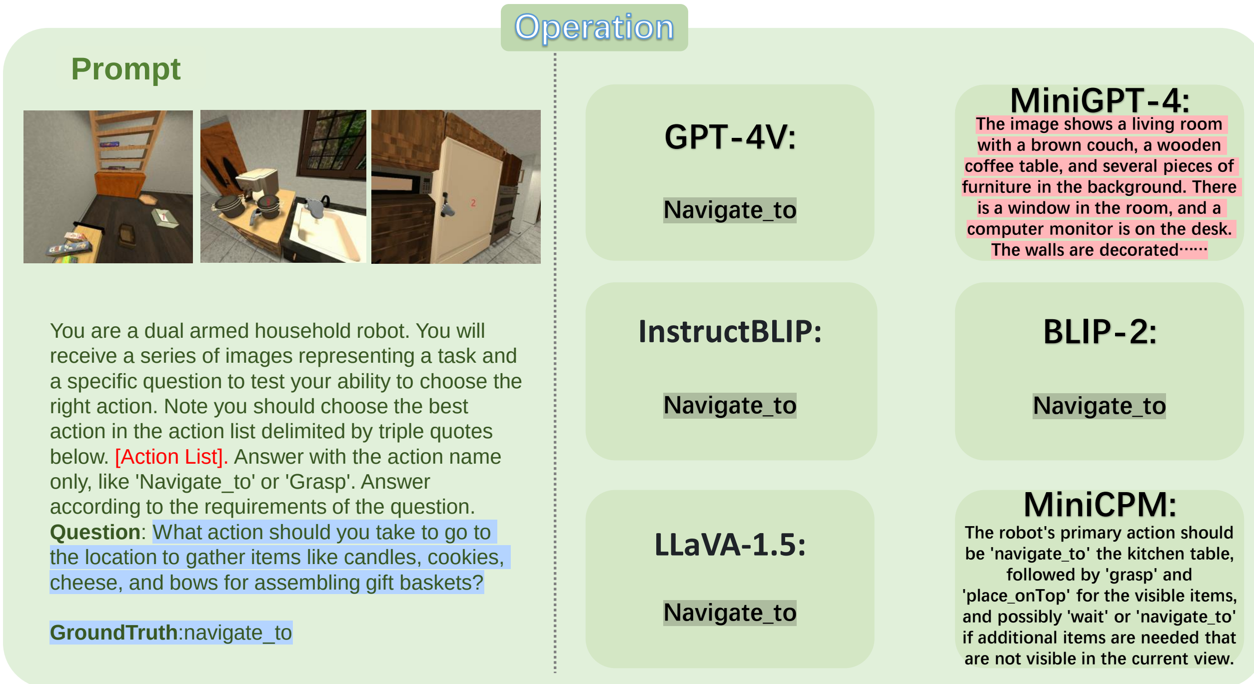

Figure 10. Full prompt and results on Sequence.

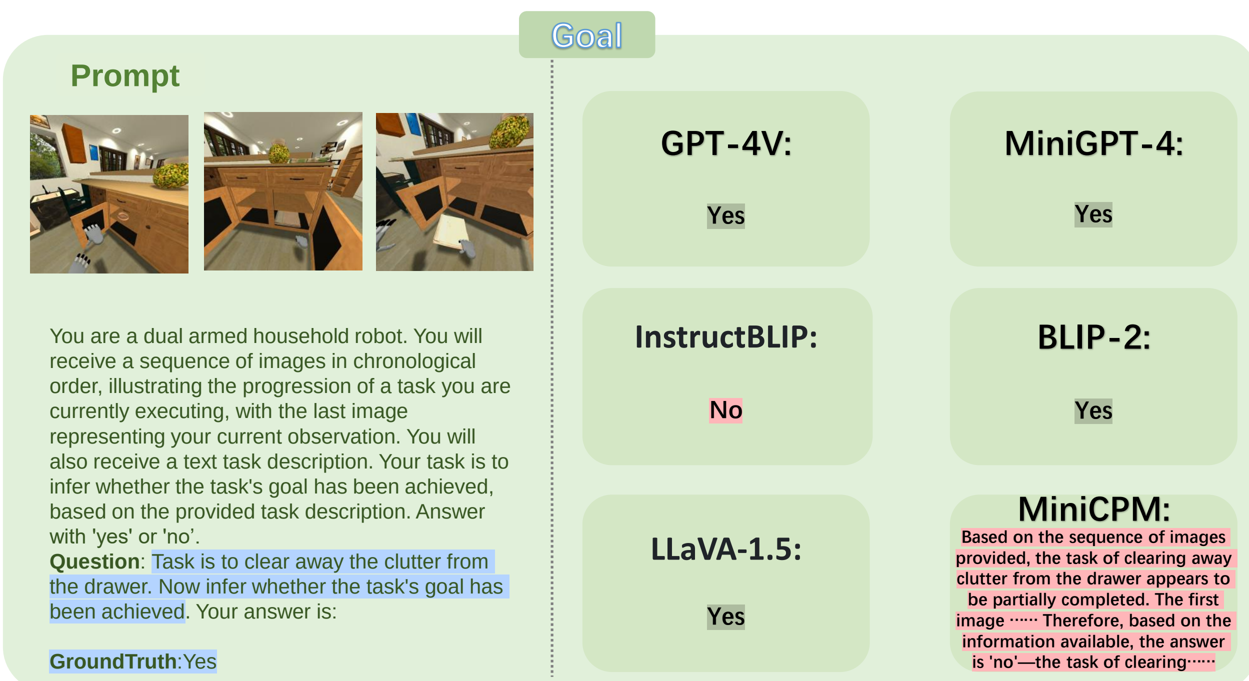

Figure 11. Full prompt and results on Goal aspect.

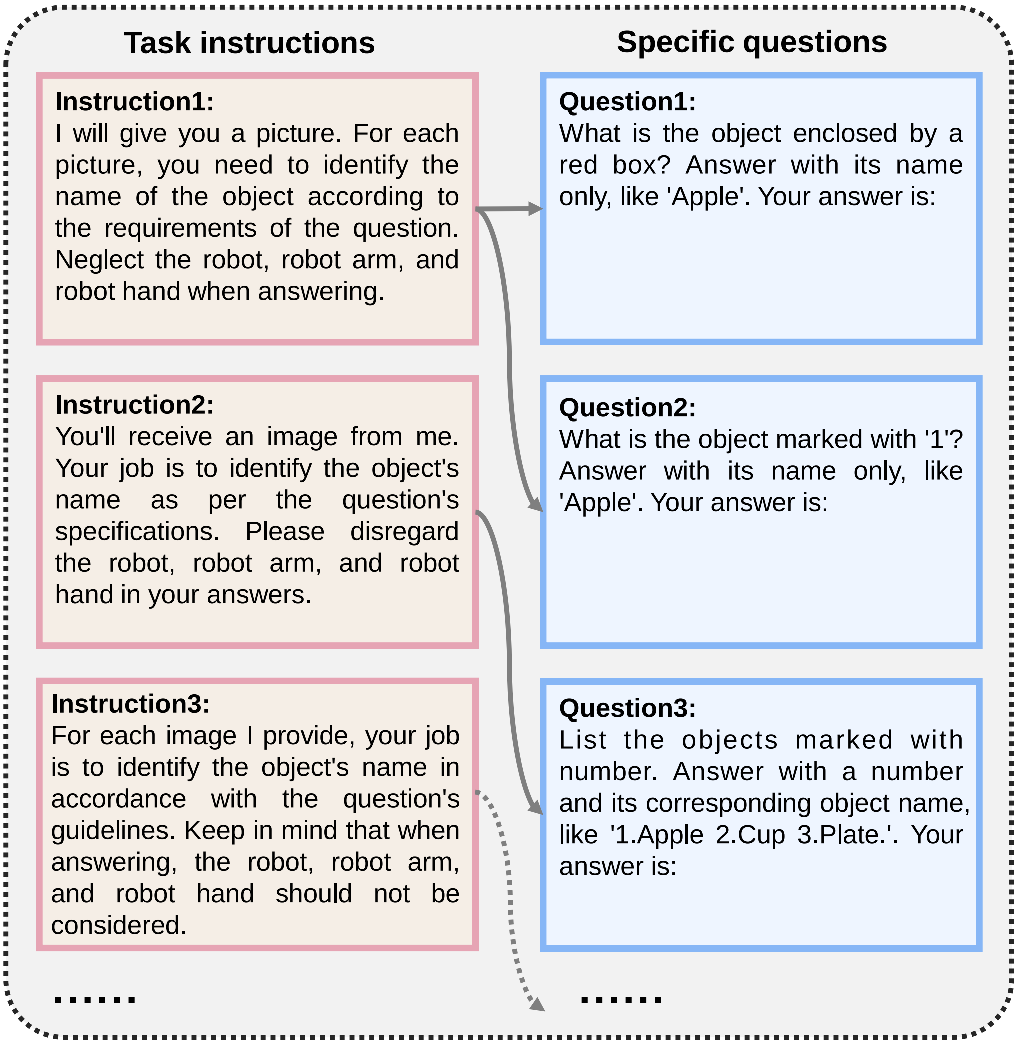

Figure 12. Illustration of the diversity of prompt. Here we choose Object Types QA as an example. The final input text prompt will be the concatenation of task instructions (in red boxes) and specific questions (in blue boxes).

Prompt:

There is a dual armed humanoid robot (agent) that can perform a variety of household tasks just like a human. The robot is currently ready to perform a household task called "preserve_food".
The language description of the task goal is:
Preserve the fruit from the dinner table into the fridge, the other foods from the dinner table into the cabinet.
I'll give you an image. Note that the image represents robot's current perspective. Please complete the following one steps according to the given image and the above task goal.

Step 1: The given image represents robot's current perspective. To ensure that the robot can accomplish the above task goal, output the correct action plan for the robot to execute using a set of predefined action functions. At the same time, output the specific preconditions and postconditions of each action function in the action plan. If the arguments of action functions are objects, specify the location of the objects.
To output the correct action plan to enable the robot to complete the task, please accurately identify the task-related objects involved in the current perspective and the objects not in the current perspective but related to the task.
To output the correct action plan, please consider the initial position of the robot and the spatial position relationship between the robot and various objects in robot's current perspective. For example, when the robot is in the initial position, some objects in robot's current perspective may or may not be within reachable distance for the robot. However, when the robot is in the initial position, objects outside robot's current perspective are not be within reachable distance for the robot. The robot position may change as the robot gradually performs the action according to the action plan. (For example, when the robot have completed the action, place_inside(pie, fridge), the robot position moves from its previous position to next to the fridge. If robot want to return to the previous position, robot need to perform the navigation action) When planning each action, consider the position of the robot and its position relationship to the relevant objects.

It is worth noting that the following predefined action functions can be adopted in the action plan if and only if their preconditions are met. For example, if the output action plan has a step, navigate_to(arg1), it means that the "arg1" is not within reachable distance for the robot before executing this step.

Predefined action functions: **{action_list}**

Below I provide you with an output example.

1. navigate_to (jar.1)
Preconditions: jar.1 is on the kitchen counter and it is not within reachable distance for the robot.
Postconditions: jar.1 is within reachable distance for the robot.

2. grasp (jar.1)
Preconditions: jar.1 is on the kitchen counter. jar.1 is within reachable distance and no object is currently held.
Postconditions: jar.1 is being held.

GPT-4V:

1. grasp (potato_chips_package)
Preconditions: The potato chips package is on the dining table and within reachable distance for the robot, and no object is currently held.
Postconditions: The potato chips package is being held by the robot.
2. navigate_to (cupboard)
Preconditions: The cupboard is not within reachable distance for the robot.
Postconditions: The cupboard is within reachable distance for the robot.
3. open (cupboard)
Preconditions: The cupboard is closed and within reachable distance.
Postconditions: The cupboard is open.
4. place_inside (potato_chips_package, cupboard) Preconditions: The potato chips package is being held, and the cupboard is reachable and open.
Postconditions: The potato chips package is inside the cupboard.
5. grasp(chocolate_bar) Preconditions: The chocolate bar is on the dining table and within reachable distance for the robot, and no object is currently held.
Postconditions: The chocolate bar is being held by the robot.
6. place_inside (chocolate_bar, cupboard) Preconditions: The chocolate bar is being held, and the cupboard is reachable and open. Postconditions: The chocolate bar is inside the cupboard.
7. close (cupboard) Preconditions: The cupboard is open and within reachable distance. Postconditions: The cupboard is closed.
8. navigate_to (fridge)
Preconditions: The fridge is not within reachable distance for the robot.
Postconditions: The fridge is within reachable distance for the robot.
9. open (fridge)
Preconditions: The fridge is closed and within reachable distance.
Postconditions: The fridge is open.
10. grasp (pie)
Preconditions: The pie is on the dining table and within reachable distance for the robot, and no object is currently held.
Postconditions: The pie is being held by the robot.
11. place_inside (pie, fridge)
Preconditions: The pie is being held, and the fridge is reachable and open.
Postconditions: The pie is inside the fridge.
12. grasp (bowl_of_fruit)
Preconditions: The bowl of fruit is on the dining table and within reachable distance for the robot, and no object is currently held.
Postconditions: The bowl of fruit is being held by the robot.
13. place_inside (bowl_of_fruit, fridge)
Preconditions: The bowl of fruit is being held, and the fridge is reachable and open. Postconditions: The bowl of fruit is inside the fridge.
14. close (fridge)
Preconditions: The fridge is open and within reachable distance.
Postconditions: The fridge is closed.

Figure 13. Task planning example. We highlight the task relevant information in blue, the right answer in green., and the wrong answer in red. For the full action list, refer to Tab. 5.

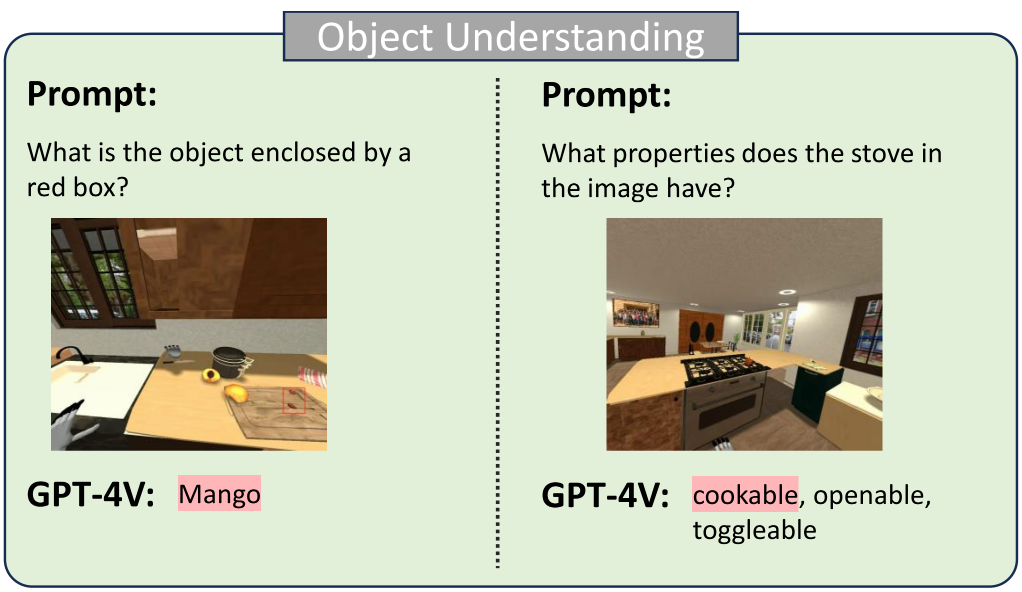

Figure 14. Qualitative Results of GPT-4V in Object Understanding.

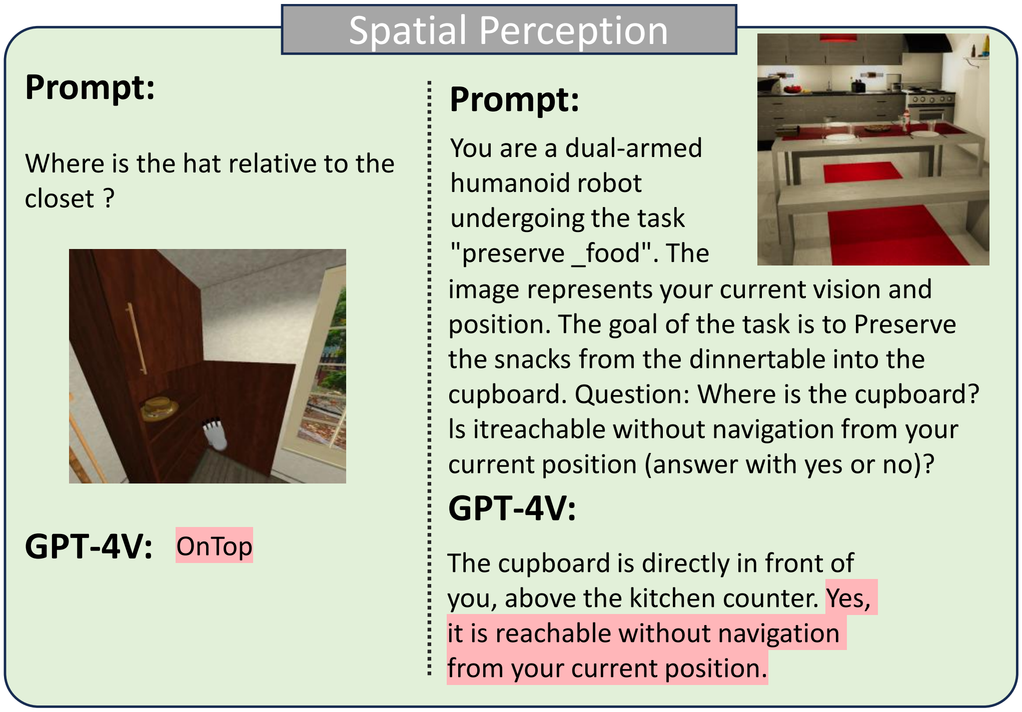

Figure 15. Qualitative Results of GPT-4V in Spatial Perception.

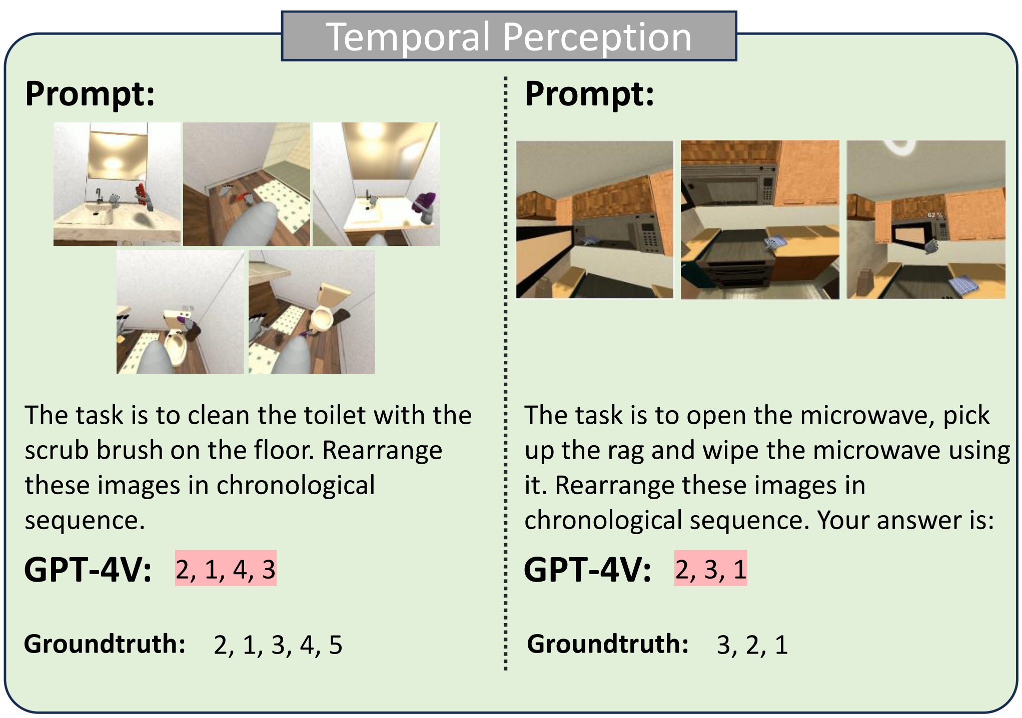

Figure 16. Qualitative Results of GPT-4V in Temporal Perception.

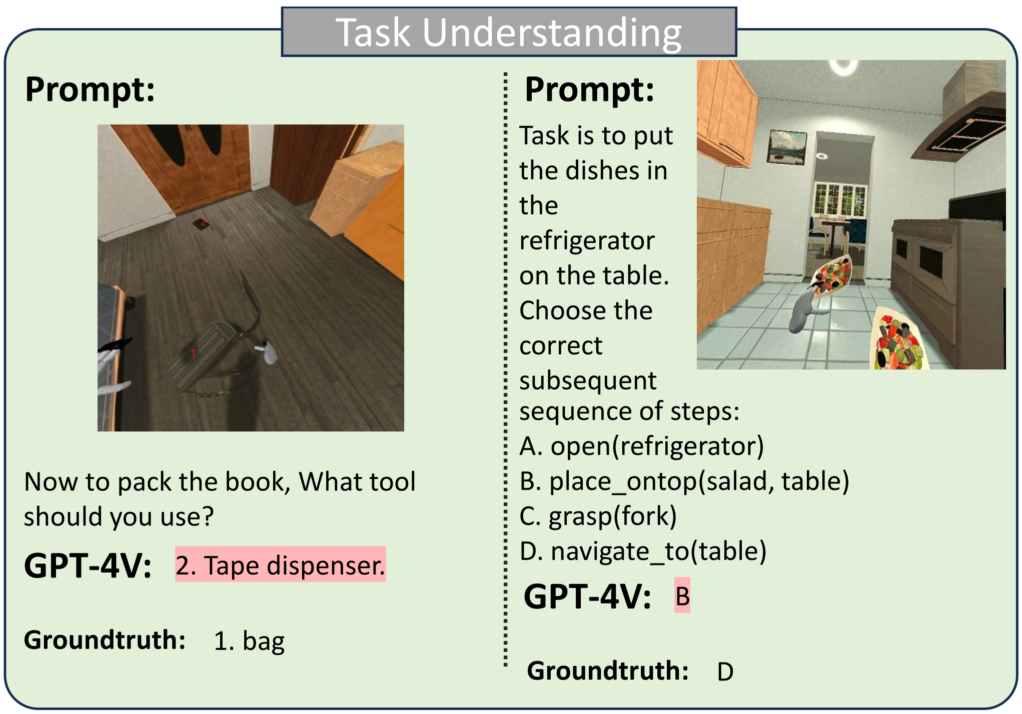

Figure 17. Qualitative Results of GPT-4V in Task Understanding.