VGRP-Bench: Visual Grid Reasoning Puzzle Benchmark for Large Vision-Language Models

Yufan Ren[1*] Konstantinos Tertikas[2] Shalini Maiti[3,4] Junlin Han[3,5]
Tong Zhang[1] Sabine Süsstrunk[1] Filippos Kokkinos[3]
[1]School of Computer and Communication Sciences, EPFL
[2]National and Kapodistrian University of Athens
[3] Meta GenAI
[4] University College London
[5] University of Oxford

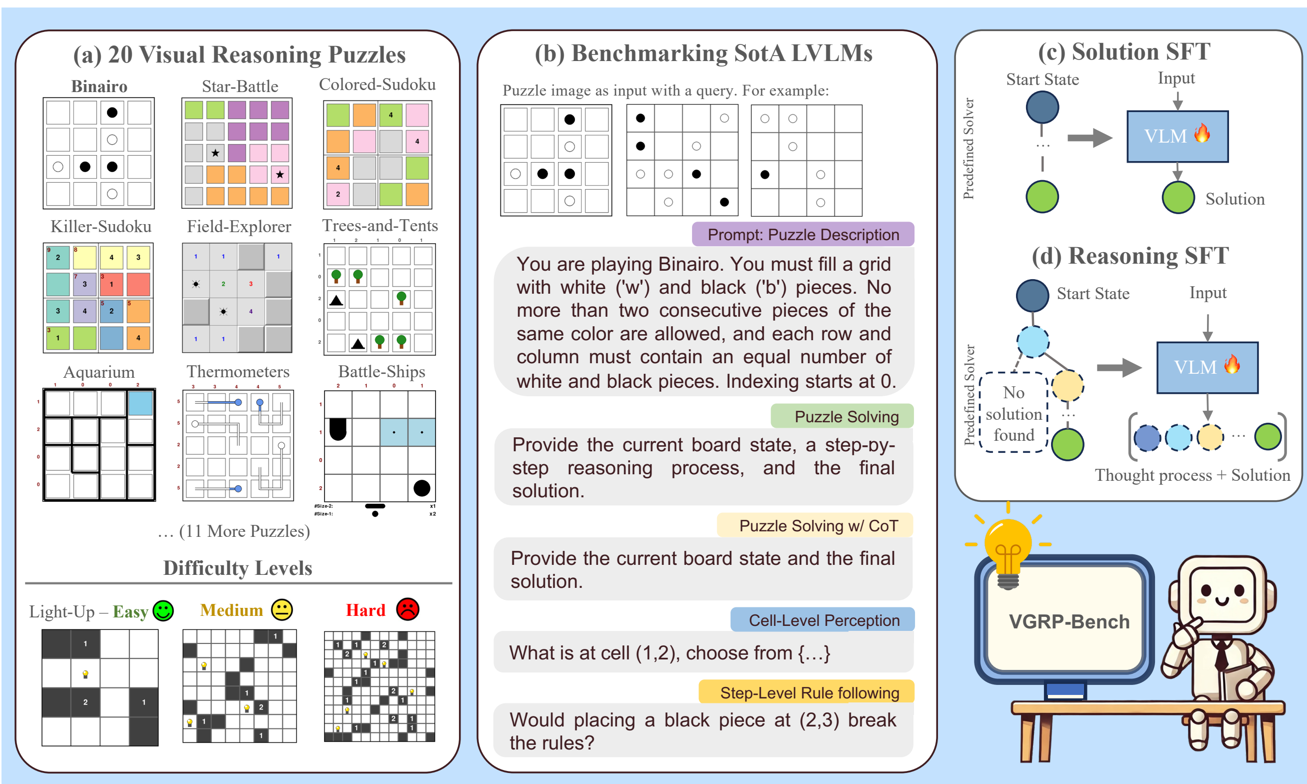

Figure 1. **Benchmark Overview.** (a) We present a benchmark for Large Vision-Language Models (LVLMs) consisting of 20 diverse visual grid reasoning puzzles (see supplementary material for complete table of per-puzzle examples and descriptions). (b) We evaluate state-of-the-art LVLMs, including closed-source models such as GPT-4o [38] and Gemini [53], open-source models like Llama 3.2 [16], and recently released reasoning models such as Gemini-Thinking, on various aspects, including perception, overall puzzle-solving, and cell-level rule-following. Additionally, to explore potential approaches for improving LVLMs' puzzle-solving abilities, we examine post-training techniques, including (c) Solution Supervised Fine-Tuning (S-SFT) and (d) Reasoning Supervised Fine-Tuning (R-SFT), where we train on thought trajectories of a predefined solver. (Best viewed on a screen when zoomed-in)

Abstract

Large Vision-Language Models (LVLMs) struggle with puzzles, which require precise perception, rule comprehension, and logical reasoning. Assessing and enhancing their performance in this domain is crucial, as it reflects their ability to engage in structured reasoning — an essential skill for real-world problem-solving. However, existing benchmarks primarily evaluate pre-trained models without additional training or fine-tuning, often lack a dedicated focus on reasoning, and fail to establish a systematic evaluation framework. To address these limitations, we introduce ***VGRP-Bench****, a Visual Grid Reasoning Puzzle Benchmark featuring 20 diverse puzzles*[1]*. VGRP-Bench spans multiple difficulty levels, and includes extensive experiments not only on existing chat LVLMs (e.g., GPT-4o), but also on reasoning LVLMs (e.g., Gemini-Thinking). Our results reveal that even the state-of-the-art LVLMs struggle with these puzzles, highlighting fundamental limitations in their puzzle-solving capabilities. Most importantly, through systematic experiments, we identify and analyze key factors influencing LVLMs' puzzle-solving performance, including the number of clues, grid size, and rule complexity. Furthermore, we explore two Supervised Fine-Tuning (SFT) strategies that can be used in post-training: SFT on solutions (S-SFT) and SFT on synthetic reasoning processes (R-SFT). While both methods significantly improve performance on trained puzzles, they exhibit limited generalization to unseen ones. We will release VGRP-Bench to facilitate further research on LVLMs for complex, real-world problem-solving.*

1. Introduction

As Large Language Models (LLMs) advance rapidly [12, 21, 46, 50, 55], researchers are extending their capabilities to multimodal tasks, leading to the rise of Large Vision-Language Models (LVLMs) [5, 16, 36, 63, 69]. While LVLMs demonstrate success in some perception tasks, they often face challenges in strategic planning, especially in visual games that require a combination of perception and multi-step reasoning [39, 59, 66].

Among the visual games, grid-like reasoning puzzles, e.g., Sudoku, Futoshiki, and Thermometers, Fig. 1, are renowned for their simple rules yet challenging solutions. They have gained widespread popularity, even being featured in annual world championships [60]. Beyond entertainment, grid puzzles also serve as structured reasoning tasks that require logical deduction, constraint satisfaction, and combinatorial search—skills that are fundamental to real-world problem-solving in domains such as robotic path planning [68], automated logistics scheduling [52], and embodied AI control [64]. Their well-defined rules and inherent complexity make them ideal for testing AI system's ability to process structured visual information and adhere to logical constraints. Nevertheless, despite their potential as benchmarks for visual reasoning, there are underused for evaluating LVLMs in existing research.

	Levels	Fine-Tuning	#Puzzles/Games	#Models
VGRP-Bench	✓	✓	**20**	**16**
ING-VP [66]	×	×	6	15
BALROG [39]	×	×	6	11
[59]	×	×	6	8

Table 1. VGRP-Bench offers a large puzzle collection for LVLM benchmarking, providing a comprehensive evaluation of state-of-the-art LVLMs across different dimensions, such as perception, rule adherence, and overall puzzle-solving, across different difficulty levels. We also investigate post-training strategies to enhance LVLMs' puzzle-solving performance.

To address this gap, we introduce the Visual Grid Reasoning Puzzle Benchmark (VGRP-Bench), the largest visual puzzle benchmark to date in terms of puzzle variety and complexity, featuring 20 diverse customizable puzzles that emphasize grid-based visual reasoning and form a taxonomy of rules, attributes, and patterns (Fig. 3). We draw inspiration from popular reasoning puzzles [42–44], and design this benchmark with different levels of difficulty, **easy** 🙂, **medium** 😐, and **hard** ☹️, depending on the grid size, the required number of reasoning steps, and the size of the decision space. We conduct extensive experiments evaluating state-of-the-art LVLMs, including their reasoning counterparts, Fig. 5. With our benchmark, we assess several aspects of LVLMs including perception, rule adherence, and overall puzzle-solving capabilities. To separate reasoning and perception, we additionally provide a text version of all puzzles. Through evaluations, we observe that our benchmark poses a huge challenge for most LVLMs, even at the easy level. For instance, GPT-4o fails to solve a simple 4×4 Sudoku consistently, even in the text-only version of the game ($< 30\%$ solving rate). We summarize several common failure cases, such as the inability to localize a number on a grid and to correctly keep track of a reasoning process. Moreover, we investigate factors that might impact an LVLM's performance, such as the difficulty level, the grid size, the number of clues, and the rules involved in a puzzle.

Beyond benchmarking off-the-shelf models following other game benchmark papers, we investigate whether post-training techniques can enhance LVLMs' puzzle-solving abilities (Tab. 1). Specifically, we explore two post-training strategies, including Solution Supervised Fine-Tuning (S-SFT) and Reasoning SFT (R-SFT). In S-SFT, we fine-

*Work done at Meta as an intern.

[1] Unlike some benchmarks that scrape fixed pre-existing puzzles from various sources, our benchmark supports sampling puzzles with different settings and difficulty levels through hyperparameters.

tune LVLMs on final solutions, typically represented as nested lists indicating the board's final state. In R-SFT, inspired by human and algorithmic approaches to puzzle solving [10, 13] such as step-by-step reasoning and process-of-elimination via rule-based deduction, we construct an SFT dataset by recording a solver's stepwise reasoning trajectory. We then fine-tune the LVLM on this dataset. We observe significant improvement in puzzle solving at the easy level, while fine-tuned models still struggle at the medium and hard levels. Additionally, recognizing the risk of overfitting to the puzzles used for finetuning, we examine the generalization capabilities of models trained with each approach in our benchmark.

In summary, we present a novel, customizable LVLM benchmark tailored for visual reasoning puzzles and conduct a systematic evaluation of LVLMs, as shown in Tab. 1. Our key contributions are as follows:

- We introduce a large LVLM customizable grid-based reasoning benchmark with systematic evaluation protocols structured around a taxonomy of diverse visual clues and rules.
- We conduct extensive experiments on state-of-the-art closed-source and open-source LVLMs using our benchmark, including fine-grained evaluations such as cell-level perception and step-wise rule understanding.
- We summarize common failure cases of LVLMs in puzzle solving and provide detailed ablation studies on various factors that impact an LVLM's puzzle solving, such as difficulty level, number of clues, and rules involved.
- To gain deeper insights into the challenges faced by LVLMs in puzzle solving, we explore two post-training strategies: Solution SFT and Reasoning SFT.

2. Related Works

2.1. General LLM/LVLM Benchmarks

The advanced capabilities of Large Language Models (LLMs) [1, 2, 53, 54] and Large Vision-Language Models (LVLMs) [30–32, 35] have inspired extensive research on benchmarking their capabilities. Prominent benchmarks like SuperGLUE [55], MMLU [21], and BigBench [50], evaluate general language understanding and multitasking text-based capabilities. Domain-specific benchmarks evaluate specialized competencies such as coding [3, 37] and mathematics [12, 22]. Notable early examples include Science QA [34], VizWiz [8], and VQAv2 [19]. Specific domains, such as image captioning, are represented by works such as [29]. More recent efforts [67], such as MMBench [33], EMMA [20], and SEED-Bench [27], offer comprehensive evaluations of multimodal reasoning and perception. BLINK [18] focuses on visual perception tasks that humans can solve in an instant. LMEvalKit [15] unifies model comparisons across various benchmarks.

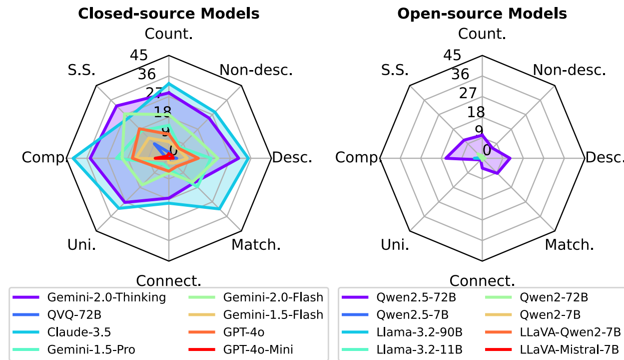

Figure 2. **Result Summary on Easy 🙂 Level.** Puzzle-solving rate of state-of-the-art chat LVLMs on easy-level puzzles associated with each rule. Please refer to the experiment section for detailed result analysis. Note that this plot's score ranges from 0 to 45%, instead of 100%. (Best viewed on a screen when zoomed in)

Our VGRP-Bench differs from other benchmarks by focusing on reasoning puzzles, a special challenge to LVLMs that requires combining perception and decision making with multi-step reasoning.

2.2. LLM/LVLM Game Benchmarks

Challenging games have long been regarded as milestones of machine intelligence as exemplified by Deep Blue [24] and AlphaGo [49]. Classical benchmarks, such as Atari [48] and the Arcade Learning Environment [7], have played a crucial role in developing reinforcement learning algorithms and improving agent capabilities. Given the natural language capabilities of LLMs, researchers have introduced benchmarks where LLM agents interact within game environments [40, 61]. [9, 23, 45, 51, 57] investigate LLMs' performance in agent-based and collaborative game environments, emphasizing interaction and teamwork skills.

Several recent studies benchmark LVLMs on visual games. ING-VP [66] shows that LVLMs still struggle with easy games. [59] proposes a benchmark with fine-grained evaluation. BALROG [39] measures LVLM games like MiniHack and NetHack. [17] proposed a puzzle RL environment, and benchmark several RL algorithms. ZeroBench [47] proposes a benchmark in which current LVLMs struggle to achieve meaningful accuracy. A concurrent work, [56], created a visual benchmark by scraping existing puzzles from online sources, resulting in a dataset of 949 instances of puzzles.

VGRP-Bench distinguishes itself by focusing on reasoning puzzles, employing customizable puzzle generators, and systematically evaluating models from inference to post-training techniques.

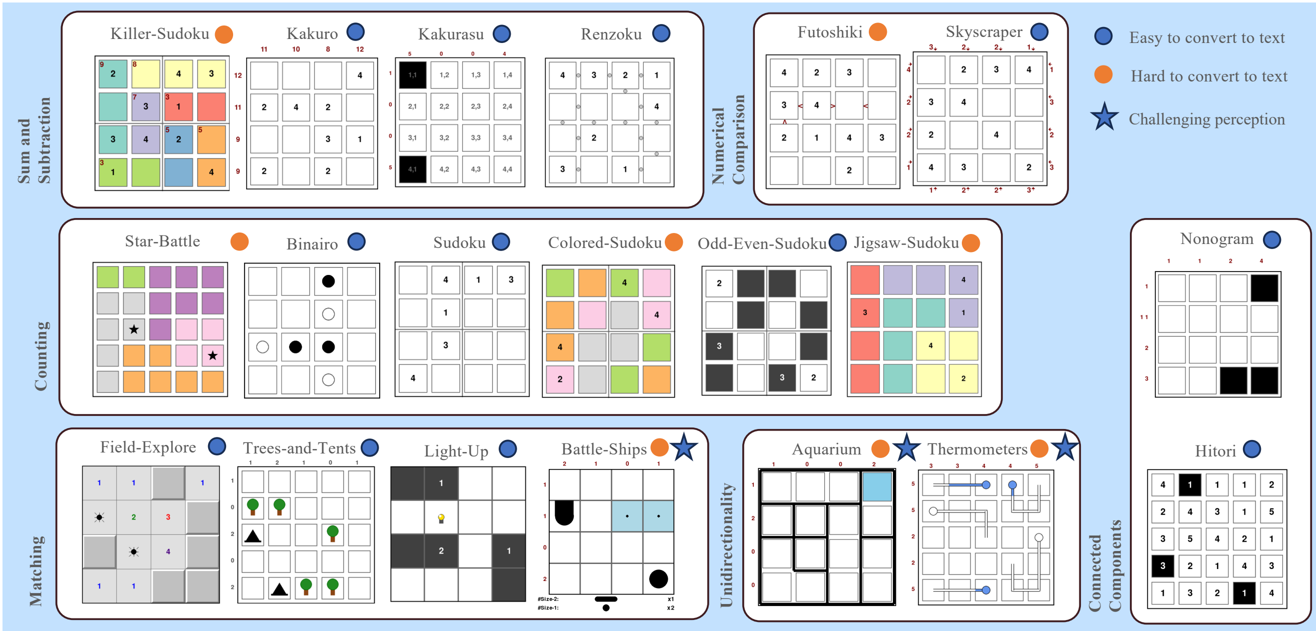

Figure 3. **Benchmark Games: Primitives and Sample Questions**. we systematically define puzzle primitives, including conditions, constraints, variables, and states, to establish a unified framework for inference and evaluation (left). This benchmark includes tasks designed to evaluate the reasoning, rule-following, and perception capabilities of state-of-the-art LVLMs. (Best viewed on a screen when zoomed in)

3. VGRP-Bench: The Benchmark

This section is organized as follows: we first present our benchmark in Sec. 3.1, along with its evaluation protocol in Sec. 3.2 and taxonomy in Sec. 3.3. In addition to benchmarking off-the-shelf models, we investigate the challenges faced by existing LVLMs in solving visual puzzles and propose strategies to address these limitations. Specifically, we use two fine-tuning strategies, Solution Supervised Fine-Tuning (S-SFT) and Reasoning SFT (R-SFT), as described in detail in Sec. 3.4.

3.1. Grid-Like Visual Reasoning Puzzles

Puzzle Selection. To form this benchmark, we select visual puzzle games based on the following criteria: requiring multi-step reasoning for decision-making and rule validation, incorporating a diverse range of visual clues, rules and interaction methods, and ultimately contributing to a structured taxonomy (Fig. 4). For example, vanilla Sudoku is purely numerical and relies on repetition-based constraints, while Trees-and-Tents demands pattern recognition, relational reasoning between trees and tents, and checking 1-to-1 matching. In contrast, Thermometers relies heavily on understanding and applying physical-world rules, e.g., thermometers must be filled starting from their base[2].

Puzzle Primitives. To ensure consistency across different puzzles and facilitate future integration of new ones, we design the benchmark around four core primitives—variables, states, constraints, and conditions—to provide a unified structure, as depicted in Fig.4 left. **Variables $\mathcal{V}$ and States $\mathcal{S}$.** Each puzzle consists of a set of variables, $\mathcal{V} = \{v_i\}_{i=1}^n$, representing cells or elements requiring value assignments. For example, a 4×4 *Sudoku* grid comprises 16 variables, with each variable taking a value from the set of possible values $\{1, 2, 3, 4\}$. The set of states $\mathcal{S} = \{s_i\}_{i=1}^n$ represents the current value assignments of the variables. **Constraints.** Constraints $\mathcal{C} = \{c_j\}_{j=1}^m$ define rules for valid puzzle state configurations. For instance, in *Sudoku*, constraints enforce the non-repetition of values in each row, column, and block. In *Trees and Tents*, constraints enforce a bijective mapping between trees and tents while adhering to row and column sums. **Conditions.** Conditions correspond to preset values or clues that define the puzzle's starting state. Examples include predefined digits that act as initial clues in *Sudoku* or row and column constraints given as clues in *Thermometers*.

3.2. Evaluation Protocol

Our benchmark evaluates LVLM performance across several capabilities, including perception, rule-following, and reasoning tasks at multiple granular levels, and on difficulty

[2] Here, Sudoku serves as an example of puzzles that could be easily converted to text, owing to its widespread popularity, while Trees-and-Tents and Thermometers represent puzzles harder to convert to text.

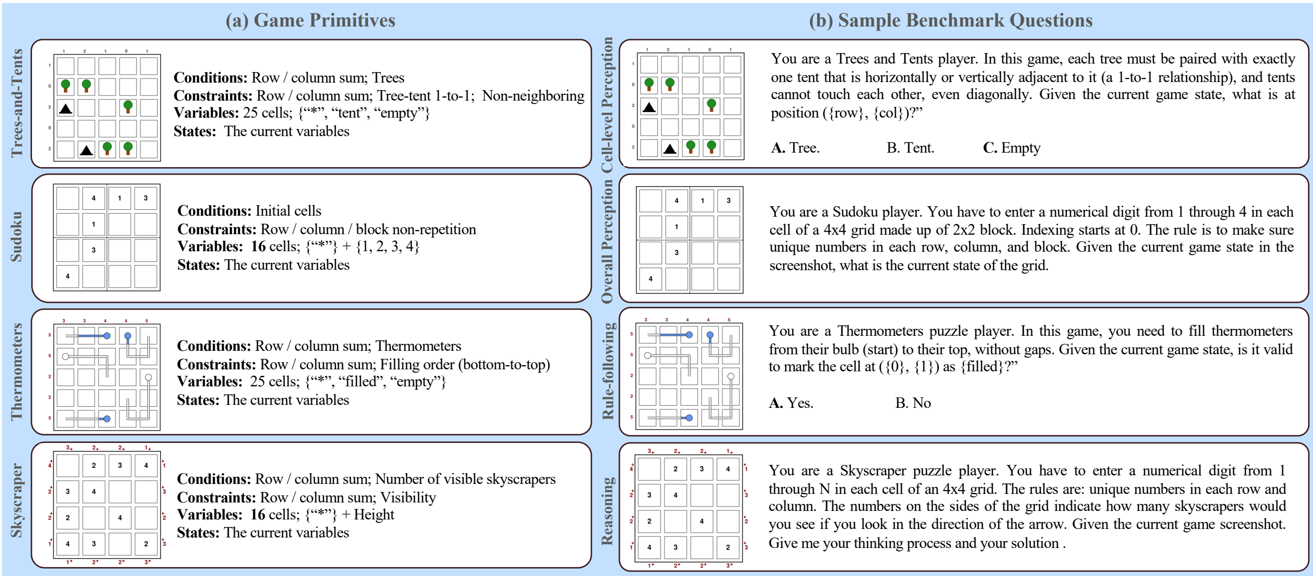

Figure 4. **Diverse Rules and Visual Patterns in VGRP-Bench.** Our benchmark includes a diverse set of rules, such as counting and mathematical calculations, and also exhibits diversity in visual patterns, encompassing text, numerical values, and objects such as trees. We highlight puzzles that are easy or difficult to convert into text.

levels, as illustrated in the right column of Fig. 4. Specifically, at the puzzle-solving level, we assess overall perception accuracy and puzzle-solving success rate by evaluating the LVLM's holistic understanding of the board and its ability to generate a correct solution. Moreover, we provide additional evaluations at finer levels of granularity, including evaluations at the cell and step level.

3.3. Puzzle Rule/Capability Taxonomy

We create a taxonomy of rule/capabilities required to solve the puzzles in our benchmark, and visualize the prominent ones in Fig. 4, as one puzzle might require multiple capabilities like counting, a basic rule in most puzzles. For example, Killer-Sudoku, Kakuro, Kakurasu, and Renzoku require mathematical calculations involving addition and subtraction. Trees-and-Tents, requiring the LVLM to understand bijective matching of trees and tents, is an example the matching rule of associating spatially or semantically relevant components. Other rules and capabilities are numerical comparison, understanding procedural order (unidirectionality) and putting connected components together.

3.4. Post-Training Techniques

Beyond assessing off-the-shelf LVLMs, we would like to take a step further to explore potential approaches to boost their performance. In this subsection, we utilize two post-training methods to tune a pretrained LVLM, i.e., Solution Supervised Fine-Tuning (S-SFT) and Reasoning Supervised Fine-Tuning (R-SFT).

S-SFT. A baseline is to use Supervised Fine-Tuning. Here, we adopt two strategies. First, we adopt a naive SFT for supervision of the LVLM to generate solutions. More specifically, we first convert the solution into a JSON-formatted text file, " {"answer": [[1, 2, 3, 4], [3, 4, 1, 2] , [2, 1, 4, 3], [4, 3, 2, 1]]}". During training, we provide a text puzzle description as prompt and a screenshot of the puzzle as input. Then we use the predefined solution as supervision for the model.

R-SFT. We introduce a SFT data creation method specific for puzzle solving. Inspired by human and algorithmic puzzle solving that feature step-by-step reasoning and per-cell rule violation checking, we propose to conduct supervised Fine-Tuning (SFT) on synthetic trajectories. In this way, we would like to supervise LVLMs to imitate step-by-step reasoning, in a similar manner to how a predefined solver solves these puzzles. To generate thought trajectories, we define the reasoning process as a trajectory through states. **A Trajectory**, $\mathcal{T} = \{s_i\}_{i=1}^{T}$, encodes key intermediate states encountered during puzzle solving. Each state s_t captures variable assignments and potential values for unassigned variables. To avoid the inefficiency of starting from a random cell, Depth-First Search (DFS) with process-of-elimination is employed, enabling systematic exploration and backtracking upon failure states. For instance, in a 4×4 Sudoku with 12 missing values, a random start often leads to excessive branching, producing trajectories that exceed the model's output window.

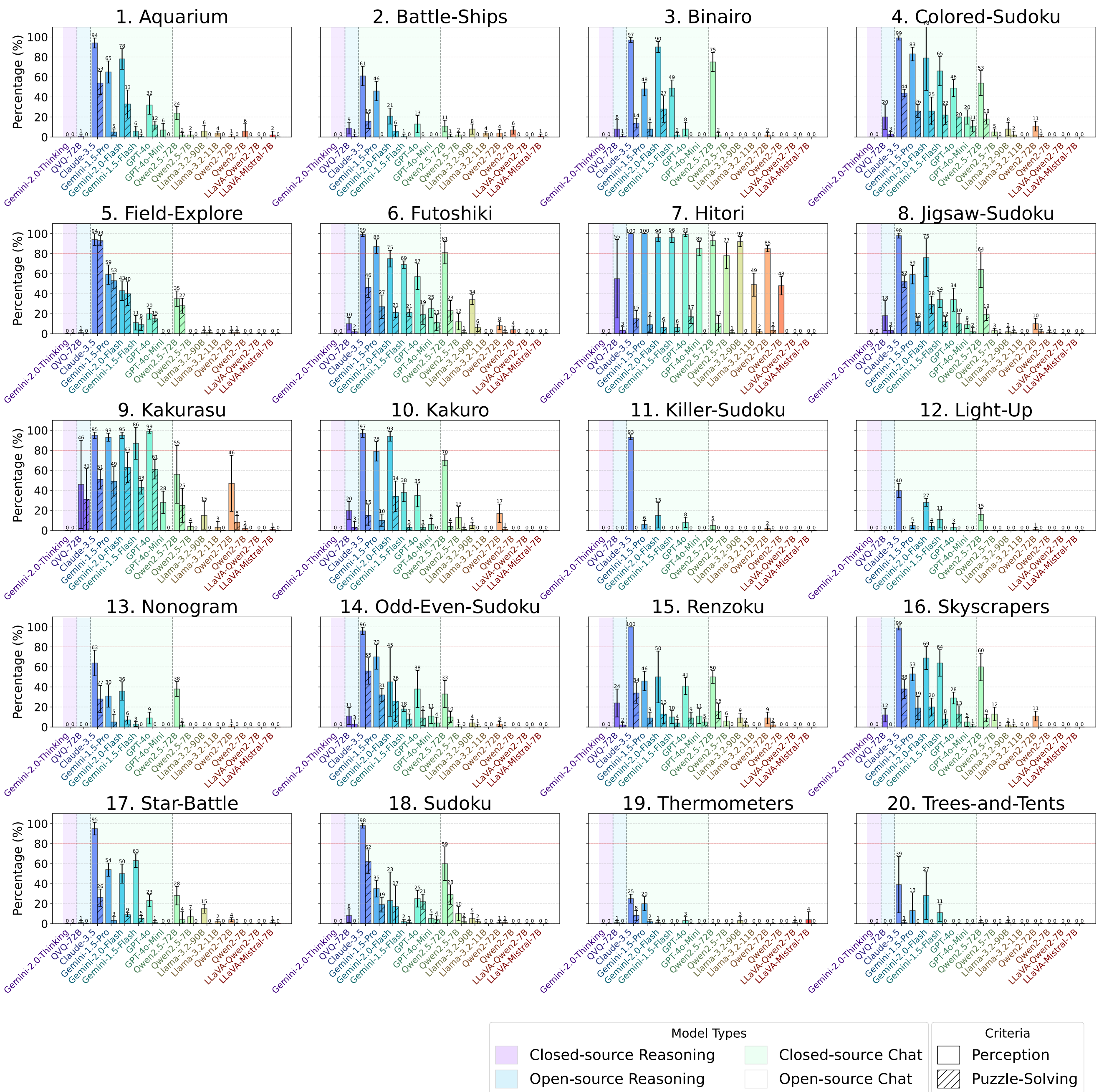

Figure 5. **Off-the-Shelf LVLMs on Level-Easy 🙂 with CoT.** We report both correct perception rate and puzzle-solving rate evaluations with closed-source / open-source and reasoning / chat models. Please refer to supplementary for additional evaluations such as finer granularity evaluations and other difficulty levels, e.g., **medium** 😐 and **hard** ☹. (Puzzle-solving in hatched bars and best viewed on a screen when zoomed in)

4. Experiments

4.1. Implementation Details

We benchmark several state-of-the-art LVLMs. For accessibility purposes, we include both closed-source and open-source models like Gemini-Pro [53] and LlaVA-OneVision-7B [28] respectively. To assess different types of models, we include both chat LVLMs and reasoning LVLMs[3]. For

[3]In the reasoning model category, we include Gemini-2.0-Thinking and Qwen-QVQ, as other reasoning models are either lacking vision capabilities, e.g., DeepSeek [14], or only accessible to high-tier users. Due to the rate limit in Gemini-2.0-Thinking, we only evaluate puzzle-solving with

evaluation, we launch 5 independent inference runs, with each run containing 20 instances, resulting in a total of 100 samples. We report the overall mean correctness and standard deviation across all sample runs. For post-training, we use Llama 3.2 Vision Instruct as the base model and conduct training on a single node equipped with 8 A100 GPUs. We ensure that the training and test splits contain no overlapping puzzles in terms of input or solution. Please refer to supplementary for more implementation details.

4.2. Off-the-Shelf LVLMs Evaluation

We present the overall perception and puzzle-solving results in Fig. 5, where all LVLMs struggle with puzzle-solving, achieving a success rate below 80%. Additional granularity and evaluation results are discussed below, and the complete evaluation on all puzzles can be found in the supplementary material. More specifically, **regarding perception, most closed-source models, except for Claude, achieve less than 50% accuracy. Among open-source models, Qwen2.5-72B performs the best.** Hitori exhibits the highest perception accuracy among all puzzles, suggesting that LVLMs struggle with grids containing missing cells. Secondly, **in terms of puzzle-solving, though all models struggle, closed-source models generally outperform open-source ones**. We also observe that larger models tend to perform better; for example, GPT-4o outperforms GPT-4o-mini. For reasoning models, we find that Gemini-2.0-Thinking performs well, whereas Qwen-QVQ underperforms compared to Qwen2.5-72B, potentially because Qwen-QVQ is a preview version.

Cell-Level Evaluation. We provide cell-level perception evaluation in Fig. 6. Similar to overall perception, closed-source models—particularly Claude and Gemini 2.0-Flash—generally achieve the highest performance. Interestingly, we notice cases when querying the LVLM for the entire board yields the correct answer, whereas querying a specific cell results in an incorrect response. This phenomenon mirrors previously observed failures in LVLMs, such as their struggles with counting tasks like "How many R's are in the word Strawberry" [62].

Step-Level Rule-Following Evaluation. Claude consistently achieves the highest performance, whereas LlaVA performs the worst among all models. Among the four puzzles shown in Fig. 7, Sudoku attains the highest accuracy, aligning with the intuition that it is a widely recognized puzzle with relatively simple and well-defined rules compared to the others.

Text Puzzles Evaluation. To understand the reasoning challenges in the text domain, we present the results of off-the-shelf models using text input in Fig. 8. Notably, while this setting eliminates vision-related losses, the puzzles remain challenging for LVLMs.

chain-of-thought prompting.

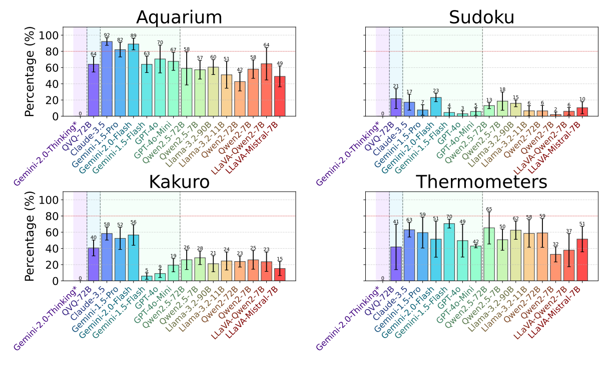

Figure 6. **Cell-level Perception Accuracy at Level-Easy.** (Best viewed on a screen when zoomed in)

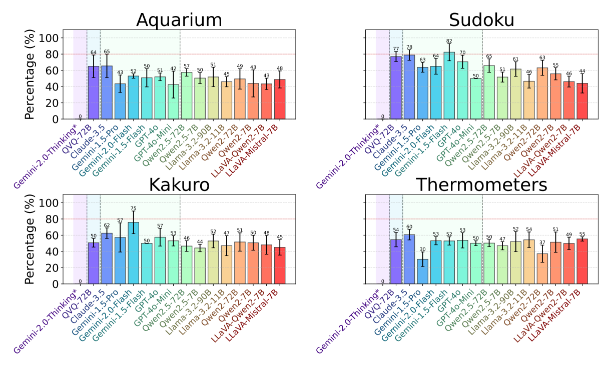

Figure 7. **Step-Level Rule-Following Accuracy at Level-Easy.** (Best viewed on a screen when zoomed in)

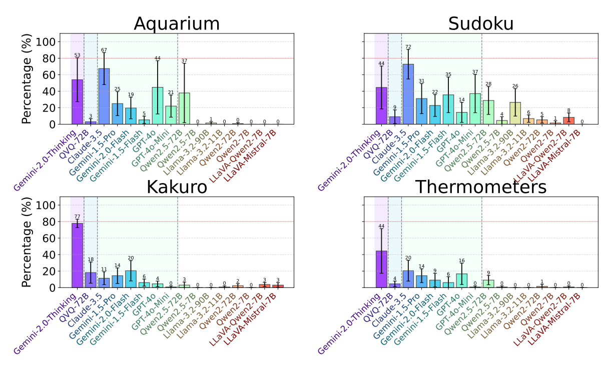

Figure 8. **Performance of Text Version Puzzles on Level-Easy.** For the text version of puzzles, the puzzle-solving rate increases significantly compared to the vision-based setting, highlighting the challenge of visual perception in our benchmark. (Best viewed on a screen when zoomed in)

Puzzle Taxonomy Analysis. The diversity of puzzles and rule types in our benchmark enables analysis through the lens of puzzle taxonomy, making it a key differentiator from other existing benchmarks. Each category includes at least two puzzles. For example, both Field-Explore and Trees-and-Tents require matching and pairing components. We present results aggregated by puzzle taxonomy in Fig. 2.

Effect of Difficulty Level. As difficulty increases, reflected in larger grids and more steps required to complete the puzzle—accuracy declines in both perception and puzzle-solving (Fig. 10). Notably, at the medium difficulty level with Thermometers, all LVLMs achieve a perception accuracy below 5% and fail to solve the puzzles completely. Performance further deteriorates at the hard difficulty level, indicating significant limitations in handling complex puzzles.

Effect of Clue Number. Intuitively, providing more clues simplifies the puzzles, leading to improved performance. This trend is evident in Fig. 9, where we also observe a corresponding increase in perception accuracy.

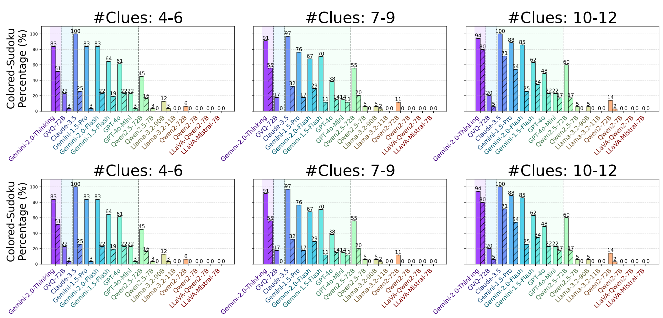

Figure 9. **Results with Different Number of Clues on Level-Easy 🙂.** When more clues are provided (to the right), puzzles become easier, resulting in a higher puzzle-solving rate. (Best viewed on a screen when zoomed in)

Common Failure Patterns. Off-the-shelf chat models exhibit several common failure cases. For instance, chat LVLMs often struggle to localize values on a grid, misinterpreting sequences like [*, 2, *,] as [, *, 2, *]. Additionally, they frequently misunderstand the roles of different components, such as mistaking a cage clue for a board number in Killer Sudoku, and they tend to repeat responses. Extensive sample outputs and common failure cases are provided in the supplementary material.

4.3. Post-Training Evaluation

We compare the pre-trained Llama 3.2 model with its fine-tuned versions after S-SFT and R-SFT in Fig. 11, with additional details provided in the supplementary material. First, **we observe that both S-SFT and R-SFT significantly enhance performance**, as the pre-trained model initially fails to produce any correct answers. This suggests that generalization to new puzzle settings is feasible. Comparing S-SFT and R-SFT, their effectiveness varies across puzzles: S-SFT outperforms R-SFT in some cases, whereas R-SFT excels in others such as Aquarium. We hypothesize that this is because R-SFT receives more supervision but is also more susceptible to compounding errors in long reasoning trajectories. We provide an evaluation on cross-puzzle generalization in the supplementary material.

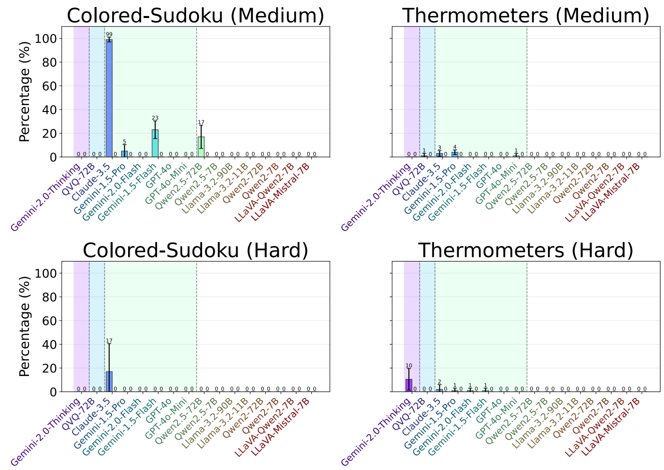

Figure 10. **Off-the-Shelf LVLMs on Level-Medium 😐 (top row) and Hard ☹ (bottom row) with CoT.** (Best viewed on a screen when zoomed in)

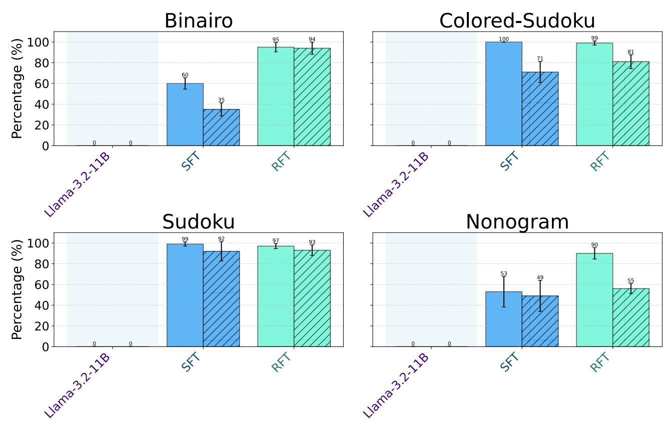

Figure 11. **Comparing S-SFT and R-SFT on Level-Easy 🙂.** Both S-SFT and R-SFT significantly improve the pretrained model's performance in perception and puzzle-solving, with R-SFT achieves slightly better results in a few puzzles such as Binairo, while being lower in puzzles like Field-Explore. (Puzzle-solving in hatched and best viewed on a screen when zoomed in)

5. Limitations and Future Work

Due to the high computational cost of fine-tuning large models (e.g., 70B parameter models), our SFT experiments are limited to smaller 11B models. Future research could explore inference-time strategies, including Monte Carlo Tree Search [49]. Another promising direction is to enhance puzzle-solving performance by integrating RL with outcome-based reward models. We report preliminary findings in the supplementary material.

6. Conclusion

In this work, we have introduced VGRP-Bench, a large visual grid puzzle benchmark with various setting, including difficulty levels and diversified puzzle rules, and systematic evaluation. We evaluated off-the-shelf LVLMs on our VGRP-Bench showing their inability of puzzle solving. Furthermore, we explore post-training for improving LVLM performance, revealing significant improvement on the trained puzzle but also a lack of generalization to unseen ones. We hope this benchmark inspires future research and advances LVLM studies for complex, real-world tasks.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 3

[2] Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*, 2023. 3

[3] Ben Athiwaratkun, Sanjay Krishna Gouda, Zijian Wang, Xiaopeng Li, Yuchen Tian, Ming Tan, Wasi Uddin Ahmad, Shiqi Wang, Qing Sun, Mingyue Shang, Sujan Kumar Gonugondla, Hantian Ding, Varun Kumar, Nathan Fulton, Arash Farahani, Siddhartha Jain, Robert Giaquinto, Haifeng Qian, Murali Krishna Ramanathan, Ramesh Nallapati, Baishakhi Ray, Parminder Bhatia, Sudipta Sengupta, Dan Roth, and Bing Xiang. Multi-lingual evaluation of code generation models. 2022. 3

[4] Stefano Baccianella. JSON Repair - A python module to repair invalid JSON, commonly used to parse the output of LLMs, 2024. 3

[5] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*, 2023. 2

[6] Evan Becker and Stefano Soatto. Cycles of thought: Measuring llm confidence through stable explanations. *arXiv preprint arXiv:2406.03441*, 2024. 4

[7] Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013. 3

[8] Jeffrey P Bigham, Chandrika Jayant, Hanjie Ji, Greg Little, Andrew Miller, Robert C Miller, Robin Miller, Aubrey Tatarowicz, Brandyn White, Samual White, et al. Vizwiz: nearly real-time answers to visual questions. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology*, pages 333–342, 2010. 3

[9] Weize Chen, Yusheng Su, Jingwei Zuo, Cheng Yang, Chenfei Yuan, Chen Qian, Chi-Min Chan, Yujia Qin, Yaxi Lu, Ruobing Xie, et al. Agentverse: Facilitating multi-agent collaboration and exploring emergent behaviors in agents. *arXiv preprint arXiv:2308.10848*, 2(4):6, 2023. 3

[10] Eric C Chi and Kenneth Lange. Techniques for solving sudoku puzzles. *arXiv preprint arXiv:1203.2295*, 2012. 3

[11] Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V Le, Sergey Levine, and Yi Ma. Sft memorizes, rl generalizes: A comparative study of foundation model post-training. *arXiv preprint arXiv:2501.17161*, 2025. 5

[12] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021. 2, 3

[13] Carlos F Daganzo. Minuet: A method to solve sudoku puzzles by hand. *arXiv preprint arXiv:1812.06778*, 2018. 3

[14] DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang.

Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. 6

[15] Haodong Duan, Junming Yang, Yuxuan Qiao, Xinyu Fang, Lin Chen, Yuan Liu, Xiaoyi Dong, Yuhang Zang, Pan Zhang, Jiaqi Wang, et al. Vlmevalkit: An open-source toolkit for evaluating large multi-modality models. In *ACMMM*, pages 11198–11201, 2024. 3

[16] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. 1, 2

[17] Benjamin Estermann, Luca Lanzendörfer, Yannick Niedermayr, and Roger Wattenhofer. Puzzles: A benchmark for neural algorithmic reasoning. *NIPS*, 37:127059–127098, 2025. 3

[18] Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A Smith, Wei-Chiu Ma, and Ranjay Krishna. Blink: Multimodal large language models can see but not perceive. *arXiv preprint arXiv:2404.12390*, 2024. 3

[19] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *CVPR*, pages 6904–6913, 2017. 3

[20] Yunzhuo Hao, Jiawei Gu, Huichen Will Wang, Linjie Li, Zhengyuan Yang, Lijuan Wang, and Yu Cheng. Can mllms reason in multimodality? emma: An enhanced multimodal reasoning benchmark. *arXiv preprint arXiv:2501.05444*, 2025. 3

[21] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020. 2, 3

[22] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021. 3

[23] Sirui Hong, Xiawu Zheng, Jonathan Chen, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, et al. Metagpt: Meta programming for multi-agent collaborative framework. *arXiv preprint arXiv:2308.00352*, 2023. 3

[24] Feng-Hsiung Hsu. *Behind Deep Blue: Building the computer that defeated the world chess champion*. Princeton University Press, 2022. 3

[25] Ryo Kamoi, Yusen Zhang, Nan Zhang, Jiawei Han, and Rui Zhang. When can llms actually correct their own mistakes? a critical survey of self-correction of llms. *Transactions of the Association for Computational Linguistics*, 12:1417–1440, 2024. 4

[26] Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023. 3

[27] Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. Seed-bench: Benchmarking multimodal llms with generative comprehension. *arXiv preprint arXiv:2307.16125*, 2023. 3

[28] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024. 6

[29] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755. Springer, 2014. 3

[30] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning, 2023. 3

[31] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning, 2023.

[32] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, 2024. 3

[33] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *ECCV*, pages 216–233. Springer, 2025. 3

[34] Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. *NIPS*, 35: 2507–2521, 2022. 3

[35] Meta AI Research. Llama 3.2: Revolutionizing edge AI and vision with open, customizable models. Technical report, Meta AI, 2024. [Online; accessed 10-Jan-2025]. 3

[36] Cade Metz. Openai unveils new ai model with advanced math and science capabilities. *The New York Times*, 2024. 2

[37] Jain Naman, Han King, Gu Alex, Li Wen-Ding, Yan Fanjia, Zhang Tianjun, Wang Sida, Solar-Lezama Armando, Sen Koushik, and Stoica Ion. Livecodebench: Holistic and contamination free evaluation of large language models for code. *arXiv preprint*, 2024. 3

[38] OpenAI. Hello gpt-4o, 2024. Accessed: 2024-12-20. 1

[39] Davide Paglieri, Bartłomiej Cupiał, Samuel Coward, Ulyana Piterbarg, Maciej Wolczyk, Akbir Khan, Eduardo Pignatelli, Łukasz Kuciński, Lerrel Pinto, Rob Fergus, et al. Balrog: Benchmarking agentic llm and vlm reasoning on games. *arXiv preprint arXiv:2411.13543*, 2024. 2, 3

[40] Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th annual acm symposium on user interface software and technology*, pages 1–22, 2023. 3

[41] Patrick Pérez, Michel Gangnet, and Andrew Blake. Poisson image editing. *TOG*, 22(3):313–318, 2003. 3

[42] Puzzle Battleships. Battleships - online puzzle game. https://www.puzzle-battleships.com/. Accessed: 2025-01-17. 2

[43] Puzzlemix. Free puzzles to play online. https://www.puzzlemix.com/menu.php. Accessed: 2025-01-17.

[44] Puzzler Media. Online puzzles, brain teasers and games. https://www.puzzler.com/online-puzzles. Accessed: 2025-01-17. 2

[45] Chen Qian, Xin Cong, Cheng Yang, Weize Chen, Yusheng Su, Juyuan Xu, Zhiyuan Liu, and Maosong Sun. Communicative agents for software development. *arXiv preprint arXiv:2307.07924*, 6(3), 2023. 3

[46] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 2

[47] Jonathan Roberts, Mohammad Reza Taesiri, Ansh Sharma, Akash Gupta, Samuel Roberts, Ioana Croitoru, Simion-Vlad Bogolin, Jialu Tang, Florian Langer, Vyas Raina, et al. Zerobench: An impossible visual benchmark for contemporary large multimodal models. *arXiv preprint arXiv:2502.09696*, 2025. 3

[48] Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, et al. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588(7839):604–609, 2020. 3

[49] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016. 3, 8

[50] Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *arXiv preprint arXiv:2206.04615*, 2022. 2, 3

[51] Weihao Tan, Ziluo Ding, Wentao Zhang, Boyu Li, Bohan Zhou, Junpeng Yue, Haochong Xia, Jiechuan Jiang, Longtao Zheng, Xinrun Xu, et al. Towards general computer control: A multimodal agent for red dead redemption ii as a case study. In *ICLR 2024 Workshop on Large Language Model (LLM) Agents*, 2024. 3

[52] Paul Mingzheng Tang, Kenji Kah Hoe Leong, Nowshad Shaik, and Hoong Chuin Lau. Automated conversion of static to dynamic scheduler via natural language. *arXiv preprint arXiv:2405.06697*, 2024. 2

[53] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 1, 3, 6

[54] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 3

[55] Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. *NIPS*, 32, 2019. 2, 3

[56] Clinton J Wang, Dean Lee, Cristina Menghini, Johannes Mols, Jack Doughty, Adam Khoja, Jayson Lynch, Sean Hendryx, Summer Yue, and Dan Hendrycks. Enigmaeval: A benchmark of long multimodal reasoning challenges. *arXiv preprint arXiv:2502.08859*, 2025. 3

[57] Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models. *arXiv preprint arXiv:2305.16291*, 2023. 3

[58] Weichuan Wang, Zhaoyi Li, Defu Lian, Chen Ma, Linqi Song, and Ying Wei. Mitigating the language mismatch and repetition issues in llm-based machine translation via model editing. *arXiv preprint arXiv:2410.07054*, 2024. 4

[59] Xinyu Wang, Bohan Zhuang, and Qi Wu. Are large vision language models good game players? In *The Thirteenth International Conference on Learning Representations*, 2025. 2, 3

[60] Wikipedia contributors. World puzzle championship — Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/World_Puzzle_Championship, 2024. [Online; accessed 6-Dec-2024]. 2

[61] Yue Wu, Xuan Tang, Tom M Mitchell, and Yuanzhi Li. Smartplay: A benchmark for llms as intelligent agents. *arXiv preprint arXiv:2310.01557*, 2023. 3

[62] Nan Xu and Xuezhe Ma. Llm the genius paradox: A linguistic and math expert's struggle with simple word-based counting problems. *arXiv preprint arXiv:2410.14166*, 2024. 7

[63] Zhenfei Yin, Jiong Wang, Jianjian Cao, Zhelun Shi, Dingning Liu, Mukai Li, Xiaoshui Huang, Zhiyong Wang, Lu Sheng, Lei Bai, et al. Lamm: Language-assisted multimodal instruction-tuning dataset, framework, and benchmark. *NIPS*, 36, 2024. 2

[64] Michał Zawalski, William Chen, Karl Pertsch, Oier Mees, Chelsea Finn, and Sergey Levine. Robotic control via embodied chain-of-thought reasoning. *arXiv preprint arXiv:2407.08693*, 2024. 2

[65] Yuexiang Zhai, Hao Bai, Zipeng Lin, Jiayi Pan, Shengbang Tong, Yifei Zhou, Alane Suhr, Saining Xie, Yann LeCun, Yi Ma, and Sergey Levine. Fine-tuning large vision-language models as decision-making agents via reinforcement learning. In *NIPS*, 2024. 5

[66] Haoran Zhang, Hangyu Guo, Shuyue Guo, Meng Cao, Wenhao Huang, Jiaheng Liu, and Ge Zhang. Ing-vp: Mllms cannot play easy vision-based games yet. *arXiv preprint arXiv:2410.06555*, 2024. 2, 3

[67] Jieyu Zhang, Weikai Huang, Zixian Ma, Oscar Michel, Dong He, Tanmay Gupta, Wei-Chiu Ma, Ali Farhadi, Aniruddha Kembhavi, and Ranjay Krishna. Task me anything. *arXiv preprint arXiv:2406.11775*, 2024. 3

[68] Hongyou Zhou, Ingmar Schubert, Marc Toussaint, and Ozgur S Oguz. Spatial reasoning via deep vision models for robotic sequential manipulation. In *IROS*, pages 11328–11335. IEEE, 2023. 2

[69] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. 2