

# 000 001 002 003 004 005 006 007 008 009 010 011 012 013 014 015 016 017 018 019 020 021 022 023 024 025 026 027 028 029 030 031 032 033 034 035 036 037 038 039 040 041 042 043 044 045 046 047 048 049 050 051 052 053 V-MAGE: A GAME EVALUATION FRAMEWORK FOR ASSESSING VISION-CENTRIC CAPABILITIES IN MULTIMODAL LARGE LANGUAGE MODELS

Anonymous authors

Paper under double-blind review

## ABSTRACT

Recent advancements in Multimodal Large Language Models (MLLMs) have demonstrated impressive capabilities in visual-text processing. However, existing static image-text benchmarks are insufficient for evaluating their dynamic perception and interactive reasoning abilities. We introduce **Vision-centric Multiple Abilities Game Evaluation (V-MAGE)**, a novel game-based evaluation framework designed to systematically assess MLLMs' visual reasoning in interactive, continuous-space environments. V-MAGE features five distinct video games comprising over 30 carefully constructed evaluation scenarios. These scenarios are set in free-form, visually complex environments that require models to interpret dynamic game states and make decisions based solely on visual input, thereby closely reflecting the conditions encountered by human players. To ensure robust and interpretable comparisons across models, V-MAGE employs a dynamic ELO-based ranking system that accounts for varying difficulty levels and task diversity. Benchmarking state-of-the-art MLLMs against human baselines reveals that while leading models approach human-level performance in simple tasks, their performance drops significantly in complex scenarios requiring advanced reasoning and task orchestration. This persistent performance gap highlights fundamental limitations in current MLLMs' ability to perform real-time, vision-grounded interactions. Through extensive analyses, we demonstrate the utility of V-MAGE in uncovering these limitations and providing actionable insights for improving the visual and reasoning capabilities of MLLMs in dynamic, interactive settings.

## 1 INTRODUCTION

Building on the success of Large Language Models (LLMs) in text-based tasks(Bai et al. (2023); Cai et al. (2024); OpenAI (2023b)), researchers have extended their capabilities to visual-text multimodal tasks through Multimodal Large Language Models (MLLMs)(OpenAI (2023a); Liu et al. (2023a); Team (2023); Yang et al. (2023); Li et al. (2024); Wang et al. (2024); Bai et al. (2025a); Zhu et al. (2025)). Various multimodal evaluation benchmarks, such as MME(Fu et al. (2023)), MM-Bench(Liu et al. (2023b)), SEED-Bench(Li et al. (2023)) have driven improvements in MLLM performance. With improving model capabilities, researchers are shifting toward open-world, dynamic, multi-round tasks beyond static benchmarks with fixed image-text inputs, as these better reflect real-world interaction and reasoning challenges. Among the promising approaches for evaluating models in such dynamic settings, game-based evaluation has emerged as a promising alternative, offering a more natural and interactive assessment of a model's perception and reasoning abilities.

While progress has been made in game-based MLLM benchmarks, current approaches predominantly rely on text-based (Costarelli et al. (2024); Hu et al. (2024); Duan et al. (2024)) or grid-based(Zhang et al. (2024); Wang et al. (2025); Paglieri et al. (2024)) games. In such settings, limited visual reasoning demands and static, fully textually renderable content restrict evaluation of spatial, temporal, and dynamic complexities crucial for real-world problem-solving. In contrast, the rich visual information inherent in video games presents a valuable opportunity to assess MLLMs' genuine visual reasoning capabilities, potentially addressing the limitations of current methodologies.

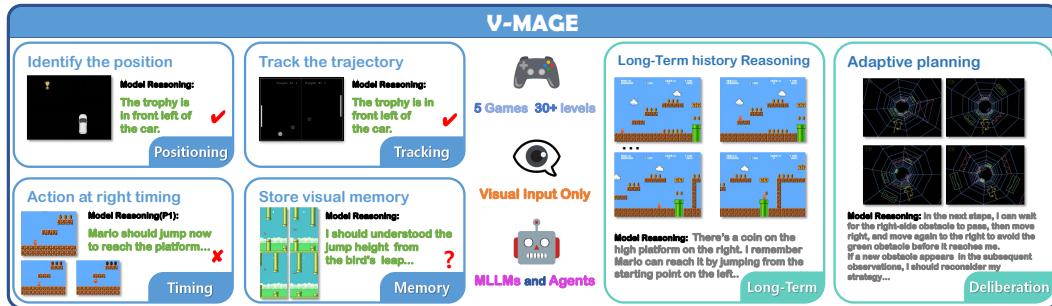


Figure 1: The overview of the V-MAGE benchmark, designed to evaluate vision-centric capabilities and higher-level reasoning of MLLMs across 5 free-form games with 30+ levels. V-MAGE assesses critical abilities in visual reasoning, providing a comprehensive evaluation of model performance in complex, dynamic environments.

To address the lack of vision-centric video game benchmarks, we present **Visual-centric Multiple Abilities Game Evaluation (V-MAGE)**, which allows for a thorough assessment of diverse model and agent abilities within dynamic, interactive game environments and addresses key limitations in current game-based evaluations of MLLM capabilities.

With V-MAGE, we evaluate leading MLLMs across five interactive games across 30+ levels. Results highlight significant challenges posed by the dynamic visual interaction environment for MLLMs. The results reveal that current MLLMs, despite excelling in static benchmarks, lack the perception, multi-step reasoning, and task orchestration required for human-level gameplay in dynamic settings.

Our contributions are summarized as follows:

- We established V-MAGE, an interactive and visually rich evaluation framework focused on dynamic interaction and vision-centric reasoning. It also serves as a sandbox environment conducive to vision agent development.
- We evaluated various publicly available MLLMs with V-MAGE, measuring model performance with ELO scores and highlighting the significant gap between model performance and human-level proficiency on complex tasks.
- Through the evaluation results of V-MAGE, we further analyzed the reasons for the suboptimal performance of current MLLMs on video game tasks, including deficiencies in several fundamental visual capabilities, challenges in reasoning during prolonged interactions, and issues such as anchoring bias, among others.

## 2 RELATED WORK

**MLLMs and Multimodal Agents.** As LLMs(Qwen et al. (2025); Cai et al. (2024); OpenAI (2023b)) advance, MLLMs have emerged to handle multimodal tasks by integrating text and visual inputs(Zhu et al. (2025); Bai et al. (2025b); Chen et al. (2024c); Wang et al. (2024); Liu et al. (2023a)). Open-source models like InternVL and QwenVL are narrowing the gap (Chen et al. (2024b)) with closed-source models such as GPT-4o(OpenAI (2024)), and Gemini(Team (2023)).

MLLMs are evolving into interactive multimodal agents, finding applications in areas such as robotics(Driess et al. (2023)), virtual assistants(Brohan et al. (2023; 2022)), GUI automation(Xu et al. (2024); Bonatti et al. (2024); Zhang et al. (2023)), and game agents(Tan et al. (2024); Chen et al. (2024a)). These domains necessitate capabilities like sequential reasoning, memory, and adaptability, which are not adequately captured by static benchmarks.

**MLLM Benchmarks.** Classic MLLM benchmarks have focused on tasks like Visual Question Answering (VQA)(Antol et al. (2015); Goyal et al. (2017); Li et al. (2018); Marino et al. (2019)) and image captioning(Chen et al. (2015); Agrawal et al. (2019); Sidorov et al. (2020)). More comprehensive benchmarks, such as MME(Fu et al. (2023)), MMBench(Liu et al. (2023b)), SEED-Bench(Li et al. (2023)), MMMU(Yue et al. (2024)), and MM-Vet(Yu et al. (2023; 2024)), introduce broader assessments across multiple domains.

108 Most of these evaluations rely on structured multiple-choice and VQA-style tasks, limiting their  
 109 ability to measure real-world problem-solving and interactive reasoning. Recent multimodal agent  
 110 benchmarks like OSWorld(Xie et al. (2024)), Windows Agent Arena(Bonatti et al. (2024)), and  
 111 COMMA(Ossowski et al. (2024)) assess broader capabilities such as open-ended tasks in real envi-  
 112 ronments, OS interaction, and multi-agent collaboration.

113 **Evaluating MLLMs in Games.** Recent work(Tan et al. (2024); Chen et al. (2024a); Ruoss et al.  
 114 (2024)) has explored MLLMs in interactive gaming environments. Meanwhile, game-based eval-  
 115 uation has evolved from text-only benchmarks(Costarelli et al. (2024); Hu et al. (2024); Duan et al.  
 116 (2024)) to vision-integrated tests(Zhang et al. (2024); Wang et al. (2025); Paglieri et al. (2024)).  
 117 However, most existing benchmarks rely on grid-based games (e.g., Tic-Tac-Toe, Chess)(Zhang  
 118 et al. (2024); Wang et al. (2025); Paglieri et al. (2024)), which can be fully represented in text. These  
 119 evaluations primarily test game-state recognition, and in some cases, additional visual input even  
 120 confuses models, reducing performance(Paglieri et al. (2024)). Consequently, by primarily focus-  
 121 ing on or being reducible to text-based representations, these benchmarks offer limited insights into  
 122 MLLMs’ visual perception and reasoning abilities, providing little guidance for improving vision-  
 123 centric skills.

### 124 3 V-MAGE BENCHMARK

125 We present V-MAGE, a benchmark built on video game environments designed to evaluate the  
 126 comprehensive performance of MLLMs, with a focus on vision-centric capabilities. Its defining  
 127 features are as follows:

- 130 • **Vision Centric Gameplay.** Models receive only visual input, requiring pixel-level scene under-  
 131 standing, object tracking, and spatial-temporal reasoning. V-MAGE features continuous-space  
 132 environments, allowing models to explore the almost infinite state space. Each game is designed  
 133 with different difficulty levels that target various skill dimensions.
- 134 • **Extensible Evaluation Framework.** V-MAGE extends beyond model evaluation to assess  
 135 agentic skills that are out-of-scope for current MLLMs. Our game-agent-model three-module  
 136 evaluation pipeline allows optimizations in both MLLMs and their agent strategies.
- 137 • **Adaptive ELO-based Ranking.** V-MAGE uses a dynamic ELO system to provide a unified  
 138 and interpretable metric across diverse games and difficulty levels. Unlike raw scores, which  
 139 vary in scale across tasks, the ELO rating captures relative skill levels by modeling win-loss  
 140 dynamics between model performances on shared levels.

#### 141 3.1 EVALUATION PIPELINE

142 V-MAGE separates the game environment from the MLLM, ensuring that all information is con-  
 143veyed solely through visual input. The MLLM interacts with games in a human-like manner: it  
 144 observes real-time screen states and generates actions based on continuous visual interpretation,  
 145 mirroring human gameplay dynamics.

146 As depicted in Figure 2, the system operates through iterative action cycles, comprising three se-  
 147 quentially linked components. The Game Module serves as the environment interface, executing  
 148 game logic, capturing real-time screenshots of the current game state, and transmitting these visual  
 149 frames to subsequent modules. The Agent Module functions as the perceptual-cognitive processor,  
 150 integrating three critical data streams: (1) raw visual inputs from the current frame, (2) tempo-  
 151 ral context from past observations, and (3) task-specific textual prompts such as game rules. This  
 152 synthesized input is structured into a multimodal format compatible with the MLLM’s processing  
 153 requirements. The Model Execution Phase completes the cycle, wherein the MLLM generates an  
 154 action command that undergoes semantic validation by the Agent Module before being relayed back  
 155 to the Game Module for environmental state updates.

156 To prioritize unbiased evaluation of core MLLM capabilities, V-MAGE’s architecture adopts a delib-  
 157 erately minimalist design, avoiding auxiliary subsystems that might obscure model performance.  
 158 The framework simultaneously retains modular extensibility, allowing researchers to modify agent  
 159 strategies without altering core evaluation protocols. This dual emphasis on streamlined standardiza-  
 160 tion and controlled customization ensures methodological rigor in benchmarking while maintaining  
 161 compatibility with specialized investigative requirements.

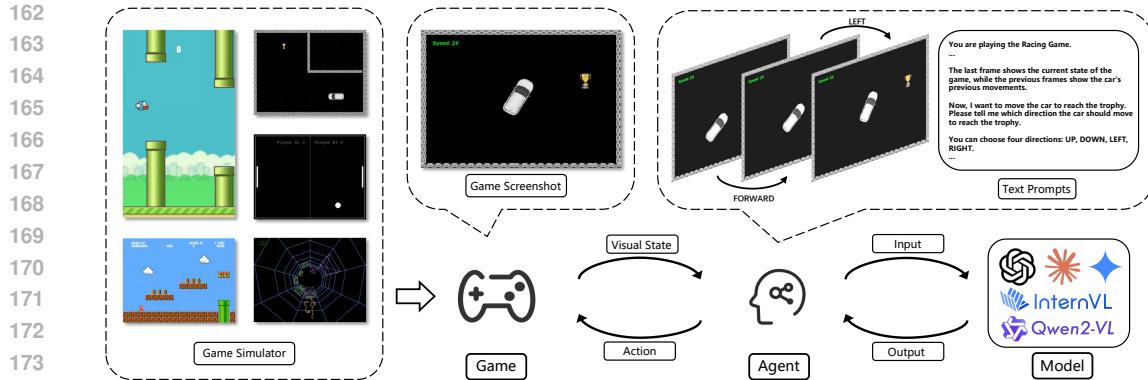


Figure 2: V-MAGE games and evaluation pipeline. V-MAGE employs five distinct games, each with several levels, to facilitate a decomposed evaluation of model performance. These games include **FlappyBird**, **Race**, **SuperMario**, **Pong** and **TempestRun**. During the evaluation process, the **Agent** module receives visual game state information directly from the **Game** module, primarily in the form of screenshots. The **Agent** module then structures these screenshots, combined with prompts containing the game rules, into the appropriate input format for MLLMs. Subsequently, the model’s output is processed by the **Agent** module to generate executable actions, which are then transmitted back to the **Game** module to update the environment state.

### 3.2 GAMES AND LEVELS

V-MAGE incorporates five human-playable video games (Figure 2), each featuring 3 to 10 levels, culminating in over 30 distinct evaluation environments. In contrast to traditional grid-based evaluation setups, V-MAGE selects games based on specific principles. The games feature free-form or continuous-space visual environments, facilitating more nuanced and flexible model movement and interaction. Crucially, to effectively assess vision-centric capabilities, the game environments are designed to be **visually irreducible**. This characteristic ensures that the system state cannot be fully discretized or textually summarized without significant information loss, thereby necessitating continuous visual grounding throughout the reasoning process. Detailed discussions regarding the game selection criteria and sources are provided in Appendix C.

Existing game-based benchmarks indicate that MLLMs frequently struggle to achieve meaningful scores at standard human-level difficulties in conventional game-based benchmarks (Zhang et al. (2024); Wang et al. (2025)). This limits their discriminative power for fine-grained capability assessment and inter-model comparisons in complex tasks. To address this, V-MAGE introduces a multi-level assessment framework that evaluates models across various skill dimensions and provides granular performance diagnostics through difficulty-stratified tasks. Specifically, levels are designed for each game with gradually increasing complexity, varying control paradigms and perceptual challenges. For instance, Figure 3 illustrates the level design in Race. Detailed information on the level design for all games can be found in Appendix C.2.

### 3.3 EVALUATION METRICS

V-MAGE employs a robust implementation of the ELO rating system to evaluate MLLMs, building on recent advancements in LLM benchmarking(Duan et al. (2024); Jiang et al. (2024)). This com-

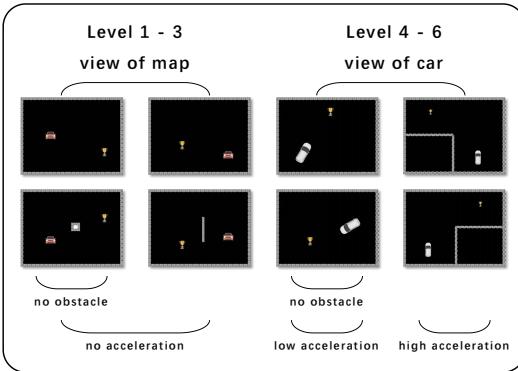


Figure 3: Race level design. Six levels progressively increase in difficulty while sharing the core objective: navigating a car to a trophy. Detailed Race level configurations are provided in Appendix Table19.

petitive evaluation framework ensures reliable model rankings by leveraging competitive evaluation mechanisms, ensuring robustness through dynamically balanced interactions.

Games demonstrate diverse scales and difficulty thresholds. The ELO system mitigates this variability by providing a standardized metric for comparing model performance across heterogeneous environments. Moreover, ELO is inherently sensitive to performance consistency. A model that achieves a high average score through a mix of exceptional successes and frequent failures may still be ranked lower, as its instability would likely lead to more losses in direct, pairwise matchups. This allows the ELO system to reward reliable performance over erratic, high-variance gameplay, which simple score averaging might otherwise obscure.

Furthermore, it skillfully measures incremental advancement in games featuring non-linear scoring plateaus, where minor improvements can differ greatly depending on the performance range. For instance, it can differentiate between progress made from 80 to 85 and significant breakthroughs from 95 to 100.

We introduce an ELO-based ranking system to assess model performance by means of competitive pairwise comparisons. In each game level, models are randomly matched in pairs for up to 100 evaluation rounds. The outcomes are determined based on their gameplay scores and the rates of valid actions taken. Detailed mathematical formulations of the pairing mechanism, rating updates, and stabilization process are provided in Appendix D.

### 3.4 COMPARISON TO EXISTING WORKS

Humans play dynamic games using visual perception and intuitive reasoning, a process largely uncaptured by most existing MLLM game benchmarks. Many of these rely on grid-based games (Wang et al. (2025); Zhang et al. (2024)) where states are textually representable. While such benchmarks assess text-based reasoning, similar to traditional LLM tasks (Paglieri et al. (2024)), they offer limited insights into MLLMs’ visual intuitive reasoning. Models often bypass genuine visual perception here, acting more like OCR converters, which hinders assessing and improving vision-centric abilities. Park et al. (2025) also employed video games as a testbed, wherein visual information remained auxiliary, and the game’s state and information were accessed chiefly through text-based inputs.

In contrast to this paradigm, V-MAGE shifts the evaluation focus by embedding models in dynamic visual environments that fundamentally necessitate real-time perception and action grounded in visual input. V-MAGE deliberately adopts environments lacking rigid grid structures, where the states of characters and objects cannot be easily simplified into sparse, coordinate-based textual descriptions. This design compels models to continuously leverage the visual modality throughout the reasoning process, rather than discarding it after an initial conversion.

Furthermore, unlike benchmarks where decisions are made based on a single static frame, such as in many chess-like environments, V-MAGE requires sophisticated temporal reasoning across sequences of frames to make informed decisions, more closely mirroring human gameplay dynamics.

By shifting evaluation to more naturalistic and visually complex dynamic game environments, V-MAGE provides a more rigorous and representative test of MLLM capabilities, particularly in assessing their visual intuitive reasoning. A holistic comparison between V-MAGE and existing game benchmarks is presented in Table 1.

Table 1: The comparison of V-MAGE with existing game-based evaluation benchmarks. \*Text in V-MAGE only represents the instructions for game rules and output format.

Game Benchmarks	Game Type	Input	Reasoning Type	Level Design
GameBench(Costarelli et al. (2024))	Word	Text-Only	Text Reasoning	✗
GameArena(Hu et al. (2024))	Word	Text-Only	Text Reasoning	✗
GTBench(Duan et al. (2024))	Word	Text-Only	Text Reasoning	✗
ING-VP(Zhang et al. (2024))	Grid Based	Single-Image-Text	Visual Aid	✓
LVLM-Playground(Wang et al. (2025))	Grid Based	Single-Image-Text	Visual Aid	✓
BALROG(Paglieri et al. (2024))	Word / Grid Based	Single-Image-Text	Text / Visual Aid	✗
Orak(Park et al. (2025))	Video	Single-Image-Text	Text / Visual Aid	✗
<b>V-MAGE</b>	<b>Video</b>	<b>Multi-Images-Text*</b>	<b>Vision-Centric Reasoning</b>	<b>✓</b>

270 

## 4 EXPERIMENTS

272 As the baseline settings for the V-MAGE benchmark, we evaluate state-of-the-art MLLMs using  
 273 full-precision models under a minimal naive agent strategy (Appendix B.1.2) to ensure a fair com-  
 274 parison. The naive agent utilizes the most recent  $k$  frames (typically  $k=3$ ) for reasoning, integrating  
 275 them with reasoning history, action decisions, and game rules as input for the models. Detailed  
 276 experiments settings and prompts can be found in Appendix B and C.4.

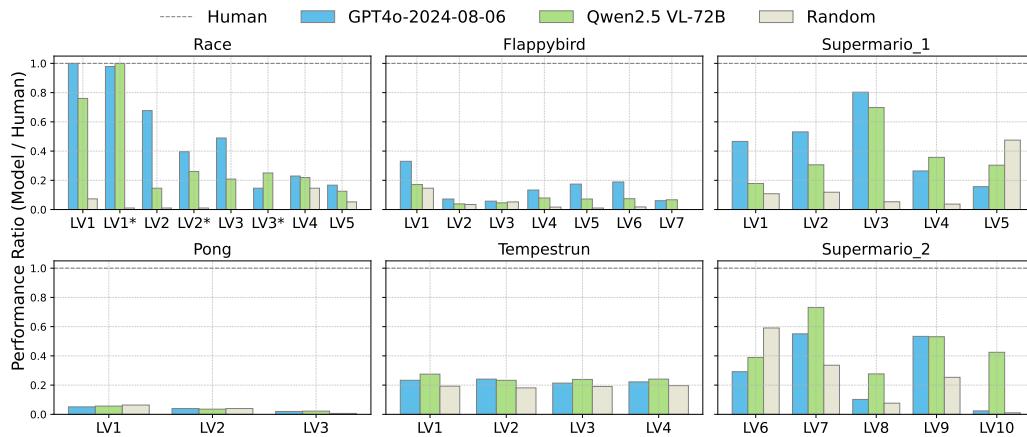
278 

### 4.1 MAIN RESULT

280 Table 2: Performance comparison across different games based on the ELO ranking system. The  
 281 Random baseline refers to randomly selecting actions from the predefined action space during  
 282 decision-making phases. Average performance ratio, abbreviated as **Avg. Ratio**, refers to the aver-  
 283 age percentage of the model’s score compared to the human baseline score.

Model	Flappybird	Pong	Race	Supermario	Tempestrun	Avg. ELO Score	Avg. Ratio (%)
GPT-4o	<b>1618</b>	1531	<b>1716</b>	1582	1548	<b>1599</b>	<b>26.6</b>
Gemini-2.0-Flash-Thinking	1579	<b>1552</b>	1648	<b>1631</b>	1525	1587	22.6
Gemini-2.0-Flash	1559	1541	1582	1561	1541	1557	16.7
Qwen2.5-VL-72B-Instruct	1563	1525	1624	1620	<b>1559</b>	1578	21.5
InternVL2.5-78B	1529	1539	1577	1614	1541	1560	19.2
Qwen2-VL-72B-Instruct	1490	1527	1587	1576	1561	1548	16.5
InternVL2.5-8B	1521	1530	1556	1438	1506	1510	12.9
Qwen2.5-VL-7B-Instruct	1469	1530	1548	1476	1494	1503	12.7
Random	1493	1516	1561	1490	1456	1503	11.0
LLaVA-Onevision-Qwen2-7B	1489	1495	1522	1448	1538	1498	13.0
Keye-VL-8B-Preview	1487	1518	1566	1401	1513	1497	13.1
Qwen2-VL-7B-Instruct	1484	1506	1529	1426	1518	1493	11.4
LLaVA-v1.6-Mistral-7B	1513	1512	1470	1396	1385	1455	9.2
Phi-4-multimodal-instruct	1441	1510	1388	1502	1389	1446	13.7
LLaVA-1.5-7B	1425	1304	1214	1473	1356	1354	14.1

298 **Scores and Rankings.** The evaluation results clearly demonstrate a performance gradient across  
 299 models ranging from 7B to 70B+ parameters. This also highlights that the dynamic visual reasoning  
 300 tasks we propose represent a universal challenge for current MLLMs. We note that rankings from  
 301 ELO scores and the Average Ratio may occasionally differ. This discrepancy arises because the ELO  
 302 system rewards performance consistency (penalizing unstable, high-variance results) and provides  
 303 a more balanced, holistic assessment across games with varying score scales. In contrast, the Avg  
 304 Ratio metric can be skewed when averaging across tasks with imbalanced performance levels. More  
 305 detailed analysis are provided in Appendix B.2 and B.3.



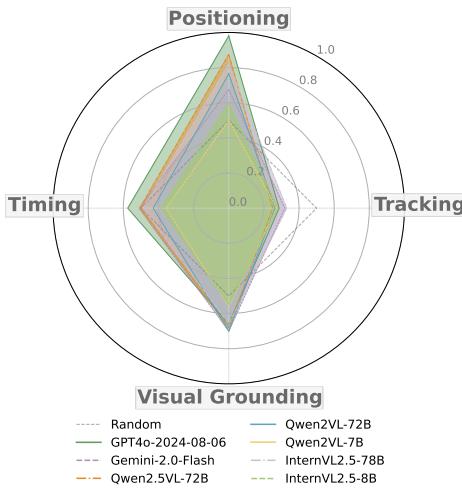
320 Figure 4: **The MLLM trails humans by a large margin in all six games.** The levels with an  
 321 asterisk (\*) represent ‘no history’. Detailed performance metrics for each model across individual  
 322 game levels are provided in Appendix B.2 (Tables 5-9).  
 323

324 **Significant Performance Gap Between MLLMs and Humans in Complex Scenes.** We in-  
 325 vited 5 human participants to play the game in the same environment as the MLLMs and calculated  
 326 their average score, which served as the baseline for human performance. Figure 4 compares the  
 327 performance of leading MLLMs and human players across different game levels. The significant  
 328 performance gap observed as task complexity increases underscores a critical limitation of current  
 329 MLLMs in dynamic environments: they struggle to integrate real-time visual perception with the  
 330 sophisticated reasoning and planning necessary for human-level gameplay, particularly in tasks de-  
 331 manding temporal understanding and flexible strategic adaptation.

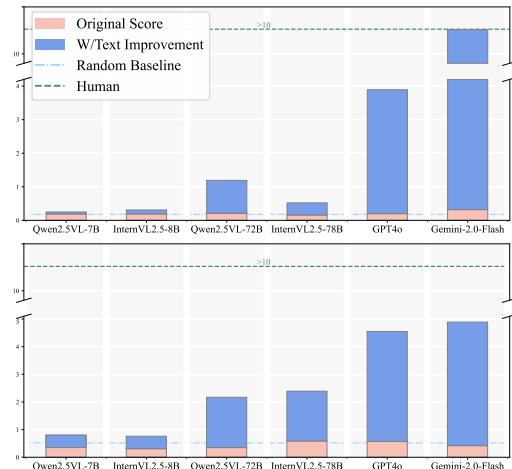
## 332 4.2 FURTHER ANALYSIS

333 **Unit Tests for Core Visual Abilities.** We devised a unit test for vision-centric abilities by extract-  
 334 ing foundational levels from V-MAGE. Figure 5 presents the capability profiles of various models  
 335 across four core visual competencies. Scores near or below baseline suggest little effective relevant  
 336 reasoning, while higher scores indicate a greater likelihood of correct reasoning. For each capabili-  
 337 ty, effective reasoning was evaluated by calculating the percentage of model scores that exceeded a  
 338 random baseline score on the corresponding unit test levels (as defined in Appendix F.1).

339 As depicted, most models substantially outperform the random baseline in **Positioning** and **Vi-  
 340 sual Grounding**, indicating a degree of proficiency in single-frame image comprehension and basic  
 341 visual information perception. However, performance notably declines in **Tracking** and **Timing**,  
 342 which require processing continuous frame information and executing precise spatiotemporal judg-  
 343 ments. For the **Tracking** task, nearly all models fail to significantly surpass the random baseline.



362 Figure 5: Capability maps of the underlying  
 363 visual capabilities of each model.



364 Figure 6: Model performance with vs. with-  
 365 out text information on Pong Level 2 (top) and  
 366 Flappy Bird Level 3 (bottom).

367 **Limitations Beyond Visual Perception.** While visual perception constitutes a critical assessment  
 368 dimension in V-MAGE’s game tasks, our experiments revealed additional limitations and defi-  
 369 ciencies in other aspects. To validate this, we conducted supplementary experiments in relatively simple  
 370 levels providing textual descriptions of important game state information, thereby bypassing the  
 371 perception process.

372 As shown in Figure 6, providing textual descriptions of the game state significantly improved the  
 373 performance of most evaluated models, with this gain being particularly prominent in larger models  
 374 such as Gemini and in games requiring precise state understanding like Pong. This notable per-  
 375 formance increase when perception is bypassed strongly suggests that limitations in processing visual  
 376 information are indeed a significant bottleneck for current MLLMs.

377 However, despite this substantial performance gain, the models’ scores still remained considerably  
 378 lower than the human baseline in most cases. This persistent gap indicates that while visual per-  
 379 ception challenges are critical, the models’ limitations extend beyond merely “seeing” the state

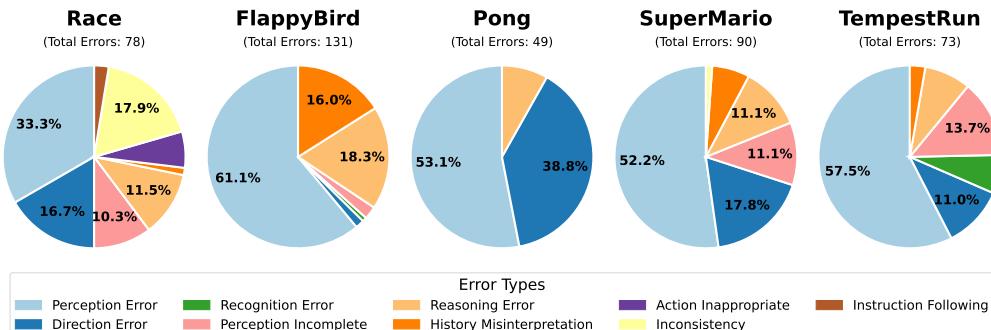
378 accurately. It highlights that significant bottlenecks also exist in the downstream processes responsible  
 379 for robust interpretation of information (even when provided textually or perceived imperfectly),  
 380 strategic planning, and effective action generation in complex and dynamic environments. Furthermore,  
 381 the less pronounced performance improvement observed in smaller models (like Qwen2.5VL  
 382 7B) suggests that inherent limitations in their core reasoning capabilities may also act as a performance  
 383 bottleneck. Check Appendix F.2 for more experimental details.

384 Table 3: Average number of rounds for each model to generate different responses.  
 385

Model	Race	FlappyBird	Pong	TempestRun	Avg.
Qwen2VL 7B	4.3	25.9	13.7	7.3	12.8
Qwen2.5VL 72B	2.3	19.3	2.6	5.3	7.4
InternVL2.5 8B	2.0	6.9	6.7	8.0	5.9
InternVL2.5 78B	6.8	16.0	2.0	3.0	7.0
GPT4o	<b>1.0</b>	<b>1.6</b>	<b>1.0</b>	<b>1.0</b>	<b>1.1</b>
<b>PCC <math>r</math></b> (Avg. Rounds vs. ELO)	-0.57	-0.71	-0.87	-0.72	-0.72

394 **Anchoring Bias in Model Inference Processes.** When provided with historical information to aid  
 395 reasoning in dynamic game progression, MLLMs often exhibit anchoring bias, particularly when  
 396 processing similar consecutive frames. This bias manifests as an undue influence of prior inferences  
 397 on current reasoning, hindering the accurate identification of subtle visual changes and unique  
 398 frame details. Models tend to favor relying on historical textual descriptions over nuanced visual  
 399 input, making them less sensitive to fine-grained visual updates, consequently leading to unchanged  
 400 reasoning content over extended game sequences.

401 As shown in Table 3, models vary significantly in their responsiveness; for instance, in FlappyBird,  
 402 Qwen2.5VL 72B altered its reasoning only once every 19.3 responses on average, significantly  
 403 less frequently than GPT-4o (1.6 responses). The Pearson correlation coefficients (**PCC  $r$** ) reveal a  
 404 consistent negative correlation between the average rounds to change response and ELO score across  
 405 games, with an average  $r$  of **-0.72**. This highlights a critical challenge in maintaining responsiveness  
 406 to dynamic visual input and its direct impact on task success. To investigate the impact of settings  
 407 within the pipeline (e.g., frame sampling and decision frequency) on anchoring bias, we conducted  
 408 corresponding experiments, with results presented in Appendix F.3.

420 Figure 7: Error type probability distribution for GPT4o across 494 samples.  
 421

422 **Analysis of Model Errors in V-MAGE.** For GPT-4o’s complete inputs and responses across all  
 423 game levels after one to two rounds of gameplay, we uniformly sampled 494 interaction sets for  
 424 manual annotation and categorized the primary error types. The visualization results depicting the  
 425 distribution of these errors are presented in Figure 7. The annotation method for error distribution  
 426 and the details of statistical data are provided in Appendix F.4.

427 The predominant error type was *perception error*, with *direction error* being particularly prevalent.  
 428 In such cases, the model frequently misidentified directional cues in visual content, leading to incor-  
 429 rect inferences. Another major category was *reasoning error*, primarily involving logical flaws and  
 430 decision-making failures, including misinterpretation of historical information (*history misinterpre-  
 431 tation*) and selection of suboptimal actions (*action inappropriate*). Figure 8 presents case examples  
 illustrating perception error and reasoning error.

432	FlappyBird	<b>GPT4o</b>	Description: The bird is <b>slightly below the center of the pipe gap</b> . The gap between the upper and lower pipe appears vertically aligned to allow safe passage if the bird is centered. Currently, the bird is a bit too low relative to the middle of the gap.	Reasoning: Since the bird is positioned slightly below the center of the gap, and the goal is to move horizontally through the gap without touching either the top or bottom pipe, the bird needs to ascend slightly to align with the center of the gap. There is no gravity, so unless it is moved manually, the bird will stay at the same height.	Action: <b>UP</b>	✓
433		<b>InternVL2.5-78B</b>	Description: The bird is currently positioned <b>below the upper pipe and above the lower pipe</b> , which means it is aligned well enough with the gap between the pipes. There is sufficient space for the bird to pass through without needing an adjustment.	Reasoning: Since the bird is already correctly positioned within the gap between the pipes, taking an action like UP or DOWN would unnecessarily risk moving out of alignment with the gap. Therefore, the best course of action is to maintain its current position until it successfully passes through the gap.	Action: <b>KEEP</b>	✗
434					Perception Error	✗
435						
436		<b>Gemini-2.0-Flash</b>	Description: The trophy is located to the right and slightly below the car.	Reasoning: There is a checkered finish line obstacle situated vertically between the car and the trophy. <b>Moving directly right would result in a collision</b> with this obstacle and a game over. To reach the trophy, the car needs to move downwards first to avoid the finish line, and then move right.	Action: <b>DOWN</b>	✓
437	Race		Description: The trophy is positioned to the right and slightly below the car. There is a vertical line of blocks <b>between the car and the trophy</b> , which acts as an obstacle.	Reasoning: To reach the trophy, the car needs to <b>move to the right first to bypass the obstacle</b> . Moving directly down or up would not help in reaching the trophy due to its current position relative to the car. Once the car moves to the right past the obstacle, it can then adjust its path if necessary to align with the trophy's position.	Action: <b>RIGHT</b>	✗
438					Reasoning Error	✗
439						
440	Level 2					
441						
442						
443						
444						
445						
446						
447	Level 3					
448						

Figure 8: Case examples illustrating Perception Error and Reasoning Error in FlappyBird and Race. The FlappyBird example shows a *Perception Error* where the model misjudges the bird’s vertical position relative to the pipe gap. The Race example illustrates a *Reasoning Error* where the model fails to plan a path around an obstacle between the car and the trophy, resulting in a suboptimal action.

Additionally, *perception incomplete* errors were commonly observed, where the model failed to fully extract useful information from visual inputs, resulting in partial information acquisition. *Inconsistency* errors mainly occurred in scenarios permitting multiple valid solutions, where the model exhibited unstable decision-making by frequently revising its choices, ultimately leading to timeout failures due to excessive deliberation. Notably, *instruction following* errors were virtually absent, as GPT-4o consistently adhered to the provided prompts. Additional case study analyses are documented in Appendix G.

**V-MAGE Poses Significant Challenges to MLLMs.** Unlike conventional static VQA or text-reducible grid-based benchmarks, V-MAGE necessitates real-time interaction within dynamic, vision-centric game environments, demanding human-like gameplay capabilities. The framework effectively exposes significant challenges and persistent limitations in current MLLMs. MLLMs demonstrate difficulty in processing and integrating information across sequences of dynamic frames, which impacts critical tasks like tracking, temporal reasoning, and trajectory understanding. This difficulty may contribute to anchoring bias, as models overly rely on prior inferences due to an insensitivity to subtle visual changes in consecutive frames, hindering their ability to adapt reasoning to dynamic game states. Furthermore, MLLMs demonstrate fundamental reasoning deficiencies that affect complex planning, strategic decision-making, and optimal action generation. These limitations persist even when initial visual processing challenges are mitigated, highlighting that deficiencies in the core reasoning process itself extend beyond perception.

## 5 CONCLUSION

This paper introduces V-MAGE, a pioneering game-based evaluation framework designed to assess the vision-centric capabilities of MLLMs in dynamic, interactive environments. Utilizing over 30 levels across 5 distinct games, our evaluation reveals significant limitations in current MLLMs. Specifically, models exhibit insufficient multi-image perception, leading to issues like anchoring bias, and demonstrate fundamental deficiencies in complex reasoning and strategic planning that persist even when perceptual challenges are mitigated. These findings highlight critical needs for future research, primarily in enhancing multi-frame visual processing and advancing higher-level reasoning capabilities. By systematically diagnosing these core deficiencies, V-MAGE sets a new and more demanding standard, challenging the field to develop MLLMs with robust, human-like visual intelligence for dynamic interactions.

486 ETHICS STATEMENT  
487488 Research involving human subjects in this paper was limited to inviting a small number of participants (N=5) to perform tasks within the V-MAGE game environments for the purpose of establishing  
489 a human performance baseline. The research has always been conducted under the guidance  
490 and supervision of our institution's Institutional Review Board (IRB) and in full compliance with its  
491 policies. To formally document this compliance for publication, our research protocol was reviewed  
492 by the IRB committee. The committee confirmed the study's classification as 'minimal risk' and has  
493 approved our research protocol.  
494495  
496 REPRODUCIBILITY STATEMENT  
497498 The experiments presented in this work are entirely reproducible. The code used to evaluate and  
499 reproduce our findings is available in the supplementary materials. Appendices B and C detail  
500 the necessary configuration environment, sample game scenes, and all input examples used in the  
501 evaluation that could potentially influence the outcomes. We affirm that the provided resources are  
502 sufficient to fully reproduce our experimental results.  
503504 REFERENCES  
505506 Harsh Agrawal, Karan Desai, Yufei Wang, Xinlei Chen, Rishabh Jain, Mark Johnson, Dhruv Batra,  
507 Devi Parikh, Stefan Lee, and Peter Anderson. Nocaps: Novel object captioning at scale. In  
508 *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 8948–8957, 2019.510 Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zit-  
511 nick, and Devi Parikh. Vqa: Visual question answering. In *ICCV*, 2015.512 Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang  
513 Zhou, and Jingren Zhou. Qwen-vl: A frontier large vision-language model with versatile abilities.  
514 *arXiv preprint arXiv:2308.12966*, 2023.  
515516 Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang,  
517 Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan,  
518 Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng,  
519 Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv*  
520 *preprint arXiv:2502.13923*, 2025a.521 Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang,  
522 Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan,  
523 Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng,  
524 Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report, 2025b.  
525 URL <https://arxiv.org/abs/2502.13923>.  
526527 Rogerio Bonatti, Dan Zhao, Francesco Bonacci, Dillon Dupont, Sara Abdali, Yinhe Li, Yadong  
528 Lu, Justin Wagle, Kazuhito Koishida, Arthur Bucker, Lawrence Jang, and Zack Hui. Windows  
529 agent arena: Evaluating multi-modal os agents at scale, 2024. URL <https://arxiv.org/abs/2409.08264>.  
530531 Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn,  
532 Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, Julian Ibarz, Brian  
533 Ichter, Alex Irpan, Tomas Jackson, Sally Jesmonth, Nikhil Joshi, Ryan Julian, Dmitry Kalash-  
534 nikov, Yuheng Kuang, Isabel Leal, Kuang-Huei Lee, Sergey Levine, Yao Lu, Utsav Malla, Deek-  
535 sha Manjunath, Igor Mordatch, Ofir Nachum, Carolina Parada, Jodilyn Peralta, Emily Perez,  
536 Karl Pertsch, Jornell Quiambao, Kanishka Rao, Michael Ryoo, Grecia Salazar, Pannag Sanketi,  
537 Kevin Sayed, Jaspiar Singh, Sumedh Sontakke, Austin Stone, Clayton Tan, Huong Tran, Vincent  
538 Vanhoucke, Steve Vega, Quan Vuong, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and  
539 Brianna Zitkovich. Rt-1: Robotics transformer for real-world control at scale. In *arXiv preprint*  
*arXiv:2212.06817*, 2022.

540 Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choro-  
 541 manski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, Pete Florence, Chuyuan Fu,  
 542 Montse Gonzalez Arenas, Keerthana Gopalakrishnan, Kehang Han, Karol Hausman, Alex Her-  
 543 zog, Jasmine Hsu, Brian Ichter, Alex Irpan, Nikhil Joshi, Ryan Julian, Dmitry Kalashnikov,  
 544 Yuheng Kuang, Isabel Leal, Lisa Lee, Tsang-Wei Edward Lee, Sergey Levine, Yao Lu, Hen-  
 545 ryk Michalewski, Igor Mordatch, Karl Pertsch, Kanishka Rao, Krista Reymann, Michael Ryoo,  
 546 Grecia Salazar, Pannag Sanketi, Pierre Sermanet, Jaspiar Singh, Anikait Singh, Radu Soricu-  
 547 tran, Huong Tran, Vincent Vanhoucke, Quan Vuong, Ayzaan Wahid, Stefan Welker, Paul Wohlhart,  
 548 Jialin Wu, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. Rt-  
 549 2: Vision-language-action models transfer web knowledge to robotic control. In *arXiv preprint*  
 550 *arXiv:2307.15818*, 2023.

551 Zheng Cai, Maosong Cao, Haojong Chen, Kai Chen, Keyu Chen, Xin Chen, Xun Chen, Zehui  
 552 Chen, Zhi Chen, Pei Chu, Xiaoyi Dong, Haodong Duan, Qi Fan, Zhaoye Fei, Yang Gao, Jiaye  
 553 Ge, Chenya Gu, Yuzhe Gu, Tao Gui, Aijia Guo, Qipeng Guo, Conghui He, Yingfan Hu, Ting  
 554 Huang, Tao Jiang, Penglong Jiao, Zhenjiang Jin, Zhikai Lei, Jiaxing Li, Jingwen Li, Linyang Li,  
 555 Shuaibin Li, Wei Li, Yining Li, Hongwei Liu, Jiangning Liu, Jiawei Hong, Kaiwen Liu, Kuikun  
 556 Liu, Xiaoran Liu, Chengqi Lv, Haijun Lv, Kai Lv, Li Ma, Runyuan Ma, Zerun Ma, Wenchang  
 557 Ning, Linke Ouyang, Jiantao Qiu, Yuan Qu, Fukai Shang, Yunfan Shao, Demin Song, Zifan Song,  
 558 Zhihao Sui, Peng Sun, Yu Sun, Huanze Tang, Bin Wang, Guoteng Wang, Jiaqi Wang, Jiayu Wang,  
 559 Rui Wang, Yudong Wang, Ziyi Wang, Xingjian Wei, Qizhen Weng, Fan Wu, Yingtong Xiong,  
 560 Chao Xu, Ruiliang Xu, Hang Yan, Yirong Yan, Xiaogui Yang, Haochen Ye, Huaiyuan Ying, Jia  
 561 Yu, Jing Yu, Yuhang Zang, Chuyu Zhang, Li Zhang, Pan Zhang, Peng Zhang, Ruijie Zhang, Shuo  
 562 Zhang, Songyang Zhang, Wenjian Zhang, Wenwei Zhang, Xingcheng Zhang, Xinyue Zhang, Hui  
 563 Zhao, Qian Zhao, Xiaomeng Zhao, Fengzhe Zhou, Zaida Zhou, Jingming Zhuo, Yicheng Zou,  
 564 Xipeng Qiu, Yu Qiao, and Dahu Lin. Internlm2 technical report, 2024.

565 Peng Chen, Pi Bu, Jun Song, Yuan Gao, and Bo Zheng. Can vlms play action role-playing games?  
 566 take black myth wukong as a study case, 2024a. URL <https://arxiv.org/abs/2409.12889>.

567 Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and  
 568 C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv*  
 569 *preprint arXiv:1504.00325*, 2015.

570 Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong,  
 571 Kongzhi Hu, Jiapeng Luo, Zheng Ma, et al. How far are we to gpt-4v? closing the gap to  
 572 commercial multimodal models with open-source suites. *Science China Information Sciences*, 67  
 573 (12):220101, 2024b.

574 Zhe Chen, Jiannan Wu, Wenhui Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong  
 575 Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning  
 576 for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer*  
 577 *Vision and Pattern Recognition*, pp. 24185–24198, 2024c.

578 Anthony Costarelli, Mat Allen, Roman Hauksson, Grace Sodunke, Suhas Hariharan, Carlson Cheng,  
 579 Wenjie Li, Joshua Clymer, and Arjun Yadav. Gamebench: Evaluating strategic reasoning abilities  
 580 of llm agents, 2024. URL <https://arxiv.org/abs/2406.06613>.

581 Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter,  
 582 Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar,  
 583 Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc  
 584 Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. Palm-e: An embodied  
 585 multimodal language model. In *arXiv preprint arXiv:2303.03378*, 2023.

586 Jinhao Duan, Renming Zhang, James Diffenderfer, Bhavya Kailkhura, Lichao Sun, Elias Stengel-  
 587 Eskin, Mohit Bansal, Tianlong Chen, and Kaidi Xu. Gtbench: Uncovering the strategic reasoning  
 588 limitations of llms via game-theoretic evaluations, 2024. URL <https://arxiv.org/abs/2402.12348>.

589 Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Zhenyu Qiu, Wei  
 590 Lin, Jinrui Yang, Xiawu Zheng, et al. Mme: A comprehensive evaluation benchmark for multi-  
 591 modal large language models. *arXiv preprint arXiv:2306.13394*, 2023.

594 Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa  
 595 matter: Elevating the role of image understanding in visual question answering. In *Proceedings*  
 596 *of the IEEE conference on computer vision and pattern recognition*, pp. 6904–6913, 2017.  
 597

598 Lanxiang Hu, Qiyu Li, Anze Xie, Nan Jiang, Ion Stoica, Haojian Jin, and Hao Zhang. Gamearena:  
 599 Evaluating llm reasoning through live computer games, 2024. URL <https://arxiv.org/abs/2412.06394>.  
 600

601 Dongfu Jiang, Max Ku, Tianle Li, Yuansheng Ni, Shizhuo Sun, Rongqi Fan, and Wenhui Chen. Genai  
 602 arena: An open evaluation platform for generative models, 2024. URL <https://arxiv.org/abs/2406.04485>.  
 603

604 Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. Seed-bench: Bench-  
 605 marking multimodal llms with generative comprehension, 2023.  
 606

607 Chunyuan Li, Zhe Gan, Zhengyuan Yang, Jianwei Yang, Linjie Li, Lijuan Wang, Jianfeng Gao, et al.  
 608 Multimodal foundation models: From specialists to general-purpose assistants. *Foundations and*  
 609 *Trends® in Computer Graphics and Vision*, 16(1-2):1–214, 2024.  
 610

611 Qing Li, Qingyi Tao, Shafiq Joty, Jianfei Cai, and Jiebo Luo. Vqa-e: Explaining, elaborating, and  
 612 enhancing your answers for visual questions. In *Proceedings of the European Conference on*  
 613 *Computer Vision (ECCV)*, pp. 552–567, 2018.  
 614

615 Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv*  
 616 *preprint arXiv:2304.08485*, 2023a.  
 617

618 Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan,  
 619 Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around  
 620 player? *arXiv preprint arXiv:2307.06281*, 2023b.  
 621

622 Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Ok-vqa: A visual  
 623 question answering benchmark requiring external knowledge. In *Conference on Computer Vision*  
 624 *and Pattern Recognition (CVPR)*, 2019.  
 625

626 OpenAI. Gpt-4v(ision) system card. 2023a. URL [https://cdn.openai.com/papers/GPTV\\_System\\_Card.pdf](https://cdn.openai.com/papers/GPTV_System_Card.pdf).  
 627

628 OpenAI. Gpt-4 technical report, 2023b.  
 629

630 OpenAI. Hello gpt-4o. *OpenAI Blog*, 2024. URL <https://openai.com/index/hello-gpt-4o/>.  
 631

632 Timothy Ossowski, Jixuan Chen, Danyal Maqbool, Zefan Cai, Tyler Bradshaw, and Jun-  
 633 jie Hu. Comma: A communicative multimodal multi-agent benchmark. *arXiv preprint*  
 634 *arXiv:2410.07553*, 2024.  
 635

636 Davide Paglieri, Bartłomiej Cupiał, Samuel Coward, Ulyana Piterbarg, Maciej Wolczyk, Akbir  
 637 Khan, Eduardo Pignatelli, Łukasz Kuciński, Lerrel Pinto, Rob Fergus, Jakob Nicolaus Foerster,  
 638 Jack Parker-Holder, and Tim Rocktäschel. Balrog: Benchmarking agentic llm and vlm reasoning  
 639 on games, 2024. URL <https://arxiv.org/abs/2411.13543>.  
 640

641 Dongmin Park, Minkyu Kim, Beongjun Choi, Junhyuck Kim, Keon Lee, Jonghyun Lee, Inkyu Park,  
 642 Byeong-Uk Lee, Jaeyoung Hwang, Jaewoo Ahn, Ameya S. Mahabaleshwarkar, Bilal Kartal, Pri-  
 643 tam Biswas, Yoshi Suhara, Kangwook Lee, and Jaewoong Cho. Orak: A foundational benchmark  
 644 for training and evaluating llm agents on diverse video games. 2025. arXiv:2506.03610.  
 645

646 Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan  
 647 Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang,  
 648 Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin  
 649 Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li,  
 650 Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang,  
 651 Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025.  
 652 URL <https://arxiv.org/abs/2412.15115>.  
 653

648 Anian Ruoss, Fabio Pardo, Harris Chan, Bonnie Li, Volodymyr Mnih, and Tim Genewein. Lmact: A  
 649 benchmark for in-context imitation learning with long multimodal demonstrations. *arXiv preprint*  
 650 *arXiv:2412.01441*, 2024.

651

652 Oleksii Sidorov, Ronghang Hu, Marcus Rohrbach, and Amanpreet Singh. Textcaps: a dataset for  
 653 image captioning with reading comprehension. In *ECCV*, pp. 742–758, 2020.

654

655 Weihao Tan, Wentao Zhang, Xinrun Xu, Haochong Xia, Ziluo Ding, Boyu Li, Bohan Zhou, Junpeng  
 656 Yue, Jiechuan Jiang, Yewen Li, Ruyi An, Molei Qin, Chuqiao Zong, Longtao Zheng, Yujie Wu,  
 657 Xiaoqiang Chai, Yifei Bi, Tianbao Xie, Pengjie Gu, Xiyun Li, Ceyao Zhang, Long Tian, Chaojie  
 658 Wang, Xinrun Wang, Börje F. Karlsson, Bo An, Shuicheng Yan, and Zongqing Lu. Cradle: Em-  
 659 powering foundation agents towards general computer control. *arXiv preprint arXiv:2403.03186*,  
 660 2024.

661

662 Google Gemini Team. Gemini: A family of highly capable multimodal models, 2023.

663

664 Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu,  
 665 Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng  
 666 Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen-2-vl: Enhancing vision-language model's  
 667 perception of the world at any resolution, 2024. URL <https://arxiv.org/abs/2409.12191>.

668

669 Xinyu Wang, Bohan Zhuang, and Qi Wu. Are large vision language models good game players? In  
 670 *International Conference on Learning Representations*, 2025.

671

672 Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing  
 673 Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, Yitao Liu, Yiheng Xu, Shuyan Zhou, Silvio  
 674 Savarese, Caiming Xiong, Victor Zhong, and Tao Yu. Osworld: Benchmarking multimodal agents  
 675 for open-ended tasks in real computer environments, 2024.

676

677 Tianqi Xu, Linyao Chen, Dai-Jie Wu, Yanjun Chen, Zecheng Zhang, Xiang Yao, Zhiqiang Xie,  
 678 Yongchao Chen, Shilong Liu, Bochen Qian, Philip Torr, Bernard Ghanem, and Guohao Li. Crab:  
 679 Cross-environment agent benchmark for multimodal language model agents, 2024. URL <https://arxiv.org/abs/2407.01511>.

680

681 Zhengyuan Yang, Linjie Li, Kevin Lin, Jianfeng Wang, Chung-Ching Lin, Zicheng Liu, and Li-  
 682 juan Wang. The dawn of lmms: Preliminary explorations with gpt-4v (ision). *arXiv preprint*  
 683 *arXiv:2309.17421*, 2023.

684

685 Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang,  
 686 and Lijuan Wang. Mm-vet: Evaluating large multimodal models for integrated capabilities, 2023.  
 687 URL <https://arxiv.org/abs/2308.02490>.

688

689 Weihao Yu, Zhengyuan Yang, Linfeng Ren, Linjie Li, Jianfeng Wang, Kevin Lin, Chung-Ching Lin,  
 690 Zicheng Liu, Lijuan Wang, and Xinchao Wang. Mm-vet v2: A challenging benchmark to evaluate  
 691 large multimodal models for integrated capabilities. *arXiv preprint arXiv:2408.00765*, 2024.

692

693 Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens,  
 694 Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multi-  
 695 modal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF*  
 Conference on Computer Vision and Pattern Recognition, pp. 9556–9567, 2024.

696

697 Chi Zhang, Zhao Yang, Jiaxuan Liu, Yucheng Han, Xin Chen, Zebiao Huang, Bin Fu, and Gang Yu.  
 698 Appagent: Multimodal agents as smartphone users, 2023. URL <https://arxiv.org/abs/2312.13771>.

699

700 Haoran Zhang, Hangyu Guo, Shuyue Guo, Meng Cao, Wenhao Huang, Jiaheng Liu, and Ge Zhang.  
 701 Ing-vp: Mllms cannot play easy vision-based games yet, 2024. URL <https://arxiv.org/abs/2410.06555>.

702 Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen  
703 Duan, Weijie Su, Jie Shao, Zhangwei Gao, Erfei Cui, Xuehui Wang, Yue Cao, Yangzhou Liu,  
704 Xingguang Wei, Hongjie Zhang, Haomin Wang, Weiye Xu, Hao Li, Jiahao Wang, Nianchen  
705 Deng, Songze Li, Yinan He, Tan Jiang, Jiapeng Luo, Yi Wang, Conghui He, Botian Shi,  
706 Xingcheng Zhang, Wenqi Shao, Junjun He, Yingtong Xiong, Wenwen Qu, Peng Sun, Penglong  
707 Jiao, Han Lv, Lijun Wu, Kaipeng Zhang, Huipeng Deng, Jiaye Ge, Kai Chen, Limin Wang, Min  
708 Dou, Lewei Lu, Xizhou Zhu, Tong Lu, Dahua Lin, Yu Qiao, Jifeng Dai, and Wenhui Wang. In-  
709 ternvl3: Exploring advanced training and test-time recipes for open-source multimodal models,  
710 2025. URL <https://arxiv.org/abs/2504.10479>.

711

712

713

714

715

716

717

718

719

720

721

722

723

724

725

726

727

728

729

730

731

732

733

734

735

736

737

738

739

740

741

742

743

744

745

746

747

748

749

750

751

752

753

754

755

756 APPENDIX OVERVIEW  
757758 In the appendix, we provide the following contents:  
759760 Sec A: Provides additional discussion of current limitations and directions for future work.  
761762 Sec B: Introduces experimental settings and provides detailed information on the experiments, mod-  
763  
764  
765  
766  
767  
768  
769  
770  
771  
772  
773  
774  
775  
776  
777  
778  
779  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809  
810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863  
864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
900  
901  
902  
903  
904  
905  
906  
907  
908  
909  
910  
911  
912  
913  
914  
915  
916  
917  
918  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
970  
971  
972  
973  
974  
975  
976  
977  
978  
979  
980  
981  
982  
983  
984  
985  
986  
987  
988  
989  
990  
991  
992  
993  
994  
995  
996  
997  
998  
999  
1000  
1001  
1002  
1003  
1004  
1005  
1006  
1007  
1008  
1009  
10010  
10011  
10012  
10013  
10014  
10015  
10016  
10017  
10018  
10019  
10020  
10021  
10022  
10023  
10024  
10025  
10026  
10027  
10028  
10029  
10030  
10031  
10032  
10033  
10034  
10035  
10036  
10037  
10038  
10039  
10040  
10041  
10042  
10043  
10044  
10045  
10046  
10047  
10048  
10049  
10050  
10051  
10052  
10053  
10054  
10055  
10056  
10057  
10058  
10059  
10060  
10061  
10062  
10063  
10064  
10065  
10066  
10067  
10068  
10069  
10070  
10071  
10072  
10073  
10074  
10075  
10076  
10077  
10078  
10079  
10080  
10081  
10082  
10083  
10084  
10085  
10086  
10087  
10088  
10089  
10090  
10091  
10092  
10093  
10094  
10095  
10096  
10097  
10098  
10099  
100100  
100101  
100102  
100103  
100104  
100105  
100106  
100107  
100108  
100109  
100110  
100111  
100112  
100113  
100114  
100115  
100116  
100117  
100118  
100119  
100120  
100121  
100122  
100123  
100124  
100125  
100126  
100127  
100128  
100129  
100130  
100131  
100132  
100133  
100134  
100135  
100136  
100137  
100138  
100139  
100140  
100141  
100142  
100143  
100144  
100145  
100146  
100147  
100148  
100149  
100150  
100151  
100152  
100153  
100154  
100155  
100156  
100157  
100158  
100159  
100160  
100161  
100162  
100163  
100164  
100165  
100166  
100167  
100168  
100169  
100170  
100171  
100172  
100173  
100174  
100175  
100176  
100177  
100178  
100179  
100180  
100181  
100182  
100183  
100184  
100185  
100186  
100187  
100188  
100189  
100190  
100191  
100192  
100193  
100194  
100195  
100196  
100197  
100198  
100199  
100200  
100201  
100202  
100203  
100204  
100205  
100206  
100207  
100208  
100209  
100210  
100211  
100212  
100213  
100214  
100215  
100216  
100217  
100218  
100219  
100220  
100221  
100222  
100223  
100224  
100225  
100226  
100227  
100228  
100229  
100230  
100231  
100232  
100233  
100234  
100235  
100236  
100237  
100238  
100239  
100240  
100241  
100242  
100243  
100244  
100245  
100246  
100247  
100248  
100249  
100250  
100251  
100252  
100253  
100254  
100255  
100256  
100257  
100258  
100259  
100260  
100261  
100262  
100263  
100264  
100265  
100266  
100267  
100268  
100269  
100270  
100271  
100272  
100273  
100274  
100275  
100276  
100277  
100278  
100279  
100280  
100281  
100282  
100283  
100284  
100285  
100286  
100287  
100288  
100289  
100290  
100291  
100292  
100293  
100294  
100295  
100296  
100297  
100298  
100299  
100300  
100301  
100302  
100303  
100304  
100305  
100306  
100307  
100308  
100309  
100310  
100311  
100312  
100313  
100314  
100315  
100316  
100317  
100318  
100319  
100320  
100321  
100322  
100323  
100324  
100325  
100326  
100327  
100328  
100329  
100330  
100331  
100332  
100333  
100334  
100335  
100336  
100337  
100338  
100339  
100340  
100341  
100342  
100343  
100344  
100345  
100346  
100347  
100348  
100349  
100350  
100351  
100352  
100353  
100354  
100355  
100356  
100357  
100358  
100359  
100360  
100361  
100362  
100363  
100364  
100365  
100366  
100367  
100368  
100369  
100370  
100371  
100372  
100373  
100374  
100375  
100376  
100377  
100378  
100379  
100380  
100381  
100382  
100383  
100384  
100385  
100386  
100387  
100388  
100389  
100390  
100391  
100392  
100393  
100394  
100395  
100396  
100397  
100398  
100399  
100400  
100401  
100402  
100403  
100404  
100405  
100406  
100407  
100408  
100409  
100410  
100411  
100412  
100413  
100414  
100415  
100416  
100417  
100418  
100419  
100420  
100421  
100422  
100423  
100424  
100425  
100426  
100427  
100428  
100429  
100430  
100431  
100432  
100433  
100434  
100435  
100436  
100437  
100438  
100439  
100440  
100441  
100442  
100443  
100444  
100445  
100446  
100447  
100448  
100449  
100450  
100451  
100452  
100453  
100454  
100455  
100456  
100457  
100458  
100459  
100460  
100461  
100462  
100463  
100464  
100465  
100466  
100467  
100468  
100469  
100470  
100471  
100472  
100473  
100474  
100475  
100476  
100477  
100478  
100479  
100480  
100481  
100482  
100483  
100484  
100485  
100486  
100487  
100488  
100489  
100490  
100491  
100492  
100493  
100494  
100495  
100496  
100497  
100498  
100499  
100500  
100501  
100502  
100503  
100504  
100505  
100506  
100507  
100508  
100509  
100510  
100511  
100512  
100513  
100514  
100515  
100516  
100517  
100518  
100519  
100520  
100521  
100522  
100523  
100524  
100525  
100526  
100527  
100528  
100529  
100530  
100531  
100532  
100533  
100534  
100535  
100536  
100537  
100538  
100539  
100540  
100541  
100542  
100543  
100544  
100545  
100546  
100547  
100548  
100549  
100550  
100551  
100552  
100553  
100554  
100555  
100556  
100557  
100558  
100559  
100560  
100561  
100562  
100563  
100564  
100565  
100566  
100567  
100568  
100569  
100570  
100571  
100572  
100573  
100574  
100575  
100576  
100577  
100578  
100579  
100580  
100581  
100582  
100583  
100584  
100585  
100586  
100587  
100588  
100589  
100590  
100591  
100592  
100593  
100594  
100595  
100596  
100597  
100598  
100599  
100600  
100601  
100602  
100603  
100604  
100605  
100606  
100607  
100608  
100609  
100610  
100611  
100612  
100613  
100614  
100615  
100616  
100617  
100618  
100619  
100620  
100621  
100622  
100623  
100624  
100625  
100626  
100627  
100628  
100629  
100630  
100631  
100632  
100633  
100634  
100635  
100636  
100637  
100638  
100639  
100640  
100641  
100642  
100643  
100644  
100645  
100646  
100647  
100648  
100649  
100650  
100651  
100652  
100653  
100654  
100655  
100656  
100657  
100658  
100659  
100660  
100661  
100662  
100663  
100664  
100665  
100666  
100667  
100668  
100669  
100670  
100671  
100672  
100673  
100674  
100675  
100676  
100677  
100678  
100679  
100680  
100681  
100682  
100683  
100684  
100685  
100686  
100687  
100688  
100689  
100690  
100691  
100692  
100693  
100694  
100695  
100696  
100697  
100698  
100699  
100700  
100701  
100702  
100703  
100704  
100705  
100706  
100707  
100708  
100709  
100710  
100711  
100712  
100713  
100714  
100715  
100716  
100717  
100718  
100719  
100720  
100721  
100722  
100723  
100724  
100725  
100726  
100727  
100728  
100729  
100730  
100731  
100732  
100733  
100734  
100735  
100736  
100737  
100738  
100739  
100740  
100741  
100742  
100743  
100744  
100745  
100746  
100747  
100748  
100749  
100750  
100751  
100752  
100753  
100754  
100755  
100756  
100757  
100758  
100759  
100760  
100761  
100762  
100763  
100764  
100765  
100766  
100767  
100768  
100769  
100770  
100771  
100772  
100773  
100774  
100775  
100776  
100777  
100778  
100779  
100780  
100781  
100782  
100783  
100784  
100785  
100786  
100787  
100788  
100789  
100790  
100791  
100792  
100793  
100794  
100795  
100796  
100797  
100798  
100799  
100800  
100801  
100802  
100803  
100804  
100805  
100806  
100807  
100808  
100809  
100810  
100811  
100812  
100813  
100814  
100815  
100816  
100817  
100818  
100819  
100820  
100821  
100822  
100823  
100824  
100825  
100826  
100827  
100828  
100829  
100830  
100831  
100832  
100833  
100834  
100835  
100836  
100837  
100838  
100839  
100840  
100841  
100842  
100843  
100844  
100845  
100846  
100847  
100848  
100849  
100850  
100851  
100852  
100853  
100854  
100855  
100856  
100857  
100858  
100859  
100860  
100861  
100862  
100863  
100864  
100865  
100866  
100867  
100868  
100869  
100870  
100871  
100872  
100873  
100874  
100875  
100876  
100877  
100878  
100879  
100880  
100881  
100882  
100883  
100884  
100885  
100886  
100887  
100888  
100889  
100890  
100891  
100892  
100893  
100894  
100895  
100896  
100897  
100898  
100899  
100900  
100901  
100902  
100903  
100904  
100905  
100906  
100907  
100908  
100909  
100910  
100911  
100912  
100913  
100914  
100915  
100916  
100917  
100918  
100919  
100920  
100921  
100922  
100923  
100924  
100925  
100926  
100927  
100928  
100929  
100930  
100931  
100932  
100933  
100934  
100935  
100936  
100937  
100938  
100939  
100940  
100941  
100942  
100943  
100944  
100945  
100946  
100947  
100948  
100949  
100950  
100951  
100952  
100953  
100954  
100955  
100956  
100957  
100958  
100959  
100960  
100961  
100962  
100963  
100964  
100965  
100966  
100967  
100968  
100969  
100970  
100971  
100972  
100973  
100974  
100975  
100976  
100977  
100978  
100979  
100980  
100981  
100982  
100983  
100984  
100985  
100986  
100987  
100988  
100989  
100990  
100991  
100992  
100993  
100994  
100995  
100996  
100997  
100998  
100999  
1001000  
1001001  
1001002  
1001003  
1001004  
1001005  
1001006  
1001007  
1001008  
1001009  
1001010  
1001011  
1001012  
1001013  
1001014  
1001015  
1001016  
1001017  
1001018  
1001019  
1001020  
1001021  
1001022  
1001023  
1001024  
1001025  
1001026  
1001027  
1001028  
1001029  
1001030  
1001031  
1001032  
1001033  
1001034  
1001035  
1001036  
1001037  
1001038  
1001039  
1001040  
1001041  
1001042  
1001043  
1001044  
1001045  
1001046  
1001047  
1001048  
1001049  
1001050  
1001051  
1001052  
1001053  
1001054  
1001055  
1001056  
1001057  
1001058  
1001059  
1001060  
1001061  
1001062  
1001063  
1001064  
1001065  
1001066  
1001067  
1001068  
1001069  
1001070  
1001071  
1001072  
1001073  
1001074  
1001075  
1001076  
1001077  
1001078  
1001079  
1001080  
1001081  
1001082  
1001083  
1001084  
1001085  
1001086  
1001087  
1001088  
1001089  
1001090  
1001091  
1001092  
1001093  
1001094  
1001095  
1001096  
1001097  
1001098  
1001099  
1001100  
1001101  
1001102  
1001103  
1001104  
1001105  
1001106  
1001107  
1001108  
1001109  
1001110  
1001111  
1001112  
1001113  
1001114  
1001115  
1001116  
1001117  
1001118  
1001119  
1001120  
1001121  
1001122  
1001123  
1001124  
1001125  
1001126  
1001127  
1001128  
1001129  
1001130  
1001131  
1001132  
1001133  
1001134  
1001135  
1001136  
1001137  
1001138  
1001139  
1001140  
1001141  
1001142  
1001143  
1001144  
1001145  
1001146  
1001147  
1001148  
1001149  
1001150  
1001151  
1001152  
1001153  
1001154  
1001155  
1001156  
1001157  
1001158  
1001159  
1001160  
1001161  
1001162  
1001163  
1001164  
1001165  
1001166  
1001167  
1001168  
1001169  
1001170  
1001171  
1001172  
1001173  
1001174  
1001175  
1001176  
1001177  
1001178  
1001179  
1001180  
1001181  
1001182  
1001183  
1001184  
1001185  
1001186  
1001187  
1001188  
1001189  
1001190  
1001191  
1001192  
1001193  
1001194  
1001195  
1001196  
1001197  
1001198  
1001199  
1001200  
1001201  
1001202  
1001203  
1001204  
1001205  
1001206  
1001207  
1001208  
1001209  
1001210  
1001211  
1001212  
1001213  
1001214  
1001215  
1001216  
1001217  
1001218  
1001219  
1001220  
1001221  
1001222  
1001223  
1001224  
1001225  
1001226  
1001227  
1001228  
1001229  
1001230  
10

810 model inference. This ensures that the game environment remains static while the model processes  
 811 visual inputs and generates actions, effectively decoupling timing evaluation from raw inference  
 812 speed.

813 Regarding real-time execution and inference delays, the frame-pausing mechanism ensures fairness  
 814 across models. While current models may not consistently achieve real-time inference due to API  
 815 or GPU limitations, V-MAGE automatically pauses the game environment during model process-  
 816 ing. This approach isolates the model’s temporal reasoning ability (strategic “when to act”) from  
 817 infrastructure-related delays, enabling a focused assessment of capabilities such as Timing. In V-  
 818 MAGE, Timing is explicitly designed to evaluate the model’s ability to choose the optimal timing of  
 819 actions, not the system’s response speed.

820 The frames per second (FPS) for all our games is 30. In our standard benchmark setting, we use  
 821 a sample rate of 3. This corresponds to the model making a decision approximately every 100  
 822 milliseconds (30 FPS / 3) in the game.

823

824

### 825 B.1.2 AGENT MODULE

826 Researchers can modify the agent’s operational mode by adjusting the configuration file. This in-  
 827 cludes altering historical strategies, such as employing a longer history of steps or sparsely sampling  
 828 more distant historical records to provide different contextual information to the models. However,  
 829 for fairness in evaluation and due to resource constraints, we present results obtained using only the  
 830 naive configuration as a baseline in our main findings. This allows for standardized comparisons  
 831 across different MLLMs.

832 In the baseline agent strategy, we utilize historical information comprising three preceding screen-  
 833 shots and the model’s reasoning outputs associated with these screenshots, combined with the cur-  
 834 rent game screenshot, to form the `<history>` and `<image_current_step>` components (as  
 835 shown in Sec. C.4) that are incorporated into the model’s input framework.

836

837 The `<history>` component is structured as follows:

838 This screenshot is `<step>` steps before the current step of the game. After this frame, your rea-  
 839 soning message was `<history_reasoning>`. After the action was executed, the game info was  
 840 `<history_action_info>`.

841

842 Upon obtaining the model’s output, the system parses the structured response format specified in  
 843 the prompt to systematically extract both the cognitive reasoning log and action parameters. These  
 844 extracted elements are subsequently stored and fed back to the game module through a standard-  
 845 ized data pipeline. This process ensures consistent interpretation of the model’s decision-making  
 846 trajectory while maintaining synchronization with the game state.

847

848 Taking an input from the actual evaluation process in the **Tempest Run** as an example:

849

850 Assume you are playing a PC game called ‘Tempest Run’.

851 You need to control a character who moves through a three-dimensional space inside a futuristic  
 852 tunnel filled with various obstacles and enemies. Your goal is to navigate through the tunnel, avoid  
 853 or overcome obstacles, and run as far as possible. Avoid colliding with red spikes, purple walls, or  
 854 failing to deal with green enemies.

855 Use the optimal combination of movements to progress through the tunnel smoothly and efficiently.  
 856 Monitor the character’s position relative to obstacles and react appropriately to avoid losing progress.  
 857 Now, I will give you some history screenshots in the game for decision making.

858 `<image>`

859 This screenshot is three steps before the current step of the game. After this frame, your reasoning  
 860 message was “observation”: ‘The character is running forward in a narrow tunnel with blue lines  
 861 forming the walls and ceiling. Red spikes are visible on the ground ahead, and green enemies are  
 862 further down the tunnel. The path appears clear with no immediate obstacles or enemies to avoid.’,  
 863 ‘reasoning’: ‘Maintaining the current trajectory with no action is optimal to continue moving straight  
 864 towards the green enemy and prepare for a potential slide to eliminate it.’, ‘action’: ‘NONE’’. After  
 865 the action was excuted, the game info was “Game is running.”

866 ...

864 <image>  
 865 This screenshot represents the current step of the game.  
 866 The last frame shows the current state of the game, while the previous frames show the character's  
 867 previous movements.  
 868 Important notes: 1. Use JUMP to jump over red spikes on the ground. 2. Use SLIDE to duck and  
 869 kick green enemies to eliminate them.  
 870 ...  
 871 You should think step by step and respond with the following format, remember to respond with  
 872 plain text without any special characters or symbols, DO NOT respond in markdown or Latex or  
 873 any other format.  
 874 Response:  
 875 Observation: ... (Describe the character's current position and nearby obstacles or enemies.)  
 876 Reasoning: ... (Think step by step and explain how you choose the action.)  
 877 Action: ... (Choose ONE of the six actions to control the character. Do NOT add any other words.)  
 878

### 879 B.1.3 MODEL MODULE

880 The Model module is primarily responsible for model deployment and parameter control. In addition  
 881 to closed-source models accessed via APIs, we deployed open-source models on an **8xV100 GPU**  
 882 **Azure cluster** for our experiments, utilizing the **vLLM** library for efficient serving. For text output  
 883 generation across all models, we standardized the decoding parameters by setting **top-p=0.9** and  
 884 **temperature=0.8**.

885 The following models are involved in the V-MAGE evaluation:  
 886

887 Table 4: Models involved in V-MAGE.

889 Model	890 Organization	891 Website	892 Open Source
GPT-4o-2024-08-06	OpenAI	<a href="https://openai.com/index/hello-gpt-4o/">https://openai.com/index/hello-gpt-4o/</a>	No
Gemini-2.0-Flash	Google DeepMind	<a href="https://deepmind.google/technologies/gemini/flash/">https://deepmind.google/technologies/gemini/flash/</a>	
InternVL2.5-78B	Shanghai AI Lab	<a href="https://huggingface.co/OpenGVLab/InternVL2_5-78B">https://huggingface.co/OpenGVLab/InternVL2_5-78B</a>	
InternVL2.5-8B		<a href="https://huggingface.co/OpenGVLab/InternVL2_5-8B">https://huggingface.co/OpenGVLab/InternVL2_5-8B</a>	
Qwen2.5VL-72B-Instruct	Alibaba Cloud	<a href="https://huggingface.co/Qwen/Qwen2.5-VL-72B-Instruct">https://huggingface.co/Qwen/Qwen2.5-VL-72B-Instruct</a>	Yes
Qwen2VL-72B-Instruct		<a href="https://huggingface.co/Qwen/Qwen2-VL-72B-Instruct">https://huggingface.co/Qwen/Qwen2-VL-72B-Instruct</a>	
Qwen2VL-7B-Instruct		<a href="https://huggingface.co/Qwen/Qwen2-VL-7B-Instruct">https://huggingface.co/Qwen/Qwen2-VL-7B-Instruct</a>	

## 900 B.2 DETAILED STATISTICS

### 901 B.2.1 SCORE-BASED PERFORMANCE

902 Cross-task result analysis reveals the limitations of parameter scaling: In RaceGame Level 1 (with  
 903 historical frame input), Qwen2VL showed a 429% improvement in score when scaling from 7B to  
 904 72B (from 10.43 to 55.19), reaching about 55% of the human baseline score. However, in more  
 905 complex tasks such as Tempestrun Level 4, InternVL2.5-78B (199.78 points) only improved by  
 906 14.4% compared to its 8B version (174.58 points), still achieving only 25% of the human score  
 907 (800 points). This suggests that parameter scaling cannot compensate for the inherent ability gap in  
 908 complex dynamic tasks. The detailed scores are presented below:  
 909

### 910 B.2.2 ADDITIONAL INDICATORS

911 Due to the dynamic game environments inherent in the V-MAGE evaluation process, certain lev-  
 912 els may necessitate a considerable number of tokens during assessment. In this section, using the  
 913 **Qwen2.5VL-72B** model deployed with **vLLM** as an illustrative example, we provide the statisti-  
 914 cally averaged frame counts (equal to the number of frames between two neighboring interactions  
 915 multiplied by the number of interactions) and the average input and output token consumption per  
 916 game round, serving as a reference.  
 917

918  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
970  
971  
Table 5: Performance analysis based on average scores in Race

Level	GPT-4o	Gemini 2.0-flash	Qwen2.5VL 72B	Qwen2VL 7B	Qwen2VL 72B	InternVL2.5 8B	InternVL2.5 78B	Random	Human
Level1	<b>99.99</b>	35.41	76.01	10.43	55.19	28.12	64.56	7.30	100.00
Level1 No History	97.87	98.91	99.95	87.46	97.87	89.54	<b>99.99</b>	1.06	100.00
Level2	<b>67.68</b>	7.30	14.59	0.00	1.06	3.14	15.63	1.06	100.00
Level2 No History	<b>39.57</b>	22.92	26.04	1.06	23.96	5.22	26.04	1.06	100.00
Level3	<b>48.94</b>	5.22	20.84	4.18	7.30	6.26	11.47	0.02	100.00
Level3 No History	14.59	4.18	<b>25.00</b>	4.18	16.67	11.47	22.92	0.00	100.00
Level4	<b>22.92</b>	6.26	21.88	0.02	2.10	0.00	2.10	14.59	100.00
Level5	16.67	8.34	12.51	13.55	4.18	<b>20.84</b>	7.30	5.22	100.00

Table 6: Performance analysis based on average scores in Pong

Level	GPT-4o	Gemini 2.0-flash	Qwen2.5VL 72B	Qwen2VL 7B	Qwen2VL 72B	InternVL2.5 8B	InternVL2.5 78B	Random	Human
Level1	0.51	0.54	0.56	0.59	0.54	0.68	<b>0.77</b>	0.63	10.00
Level2	0.39	<b>0.41</b>	0.35	0.31	0.33	0.31	0.38	0.39	10.00
Level3	0.19	<b>0.32</b>	0.21	0.18	0.20	0.18	0.15	0.06	10.00

Table 7: Performance analysis based on average scores in Supermario

Level	GPT-4o	Gemini 2.0-flash	Qwen2.5VL 72B	Qwen2VL 7B	Qwen2VL 72B	InternVL2.5 8B	InternVL2.5 78B	Random	Human
Level10	18.77	108.30	<b>339.57</b>	12.51	29.16	14.61	80.19	8.36	800.00
Level11	<b>372.85</b>	109.41	142.76	33.41	216.67	69.83	203.12	86.50	800.00
Level2	<b>424.92</b>	127.17	244.78	102.12	338.47	102.12	186.48	94.83	800.00
Level3	<b>802.99</b>	429.10	697.91	188.54	565.46	286.44	610.26	53.19	1000.00
Level4	369.76	251.07	<b>499.89</b>	112.53	346.84	151.09	447.84	52.15	1400.00
Level5	125.08	258.33	242.72	232.29	192.75	209.41	<b>433.23</b>	380.13	800.00
Level6	233.36	325.96	311.36	324.96	296.79	267.70	344.74	<b>472.78</b>	800.00
Level7	440.66	527.96	<b>585.21</b>	161.48	490.48	220.86	491.52	268.74	800.00
Level8	91.75	211.43	<b>248.96</b>	51.13	179.20	76.09	168.74	68.79	900.00
Level9	693.56	594.67	690.46	162.62	508.24	201.12	<b>756.02</b>	329.19	1300.00

Table 8: Performance analysis based on average scores in Flappybird

Level	GPT-4o	Gemini 2.0-flash	Qwen2.5VL 72B	Qwen2VL 7B	Qwen2VL 72B	InternVL2.5 8B	InternVL2.5 78B	Random	Human
Level1	<b>3.30</b>	2.38	1.70	0.76	0.47	1.20	1.54	1.45	10.00
Level2	<b>0.71</b>	0.47	0.38	0.20	0.12	0.36	0.39	0.34	10.00
Level3	<b>0.57</b>	0.41	0.45	0.20	0.35	0.33	0.43	0.52	10.00
Level4	1.33	1.50	0.79	<b>1.52</b>	0.38	1.43	0.64	0.16	10.00
Level5	<b>1.74</b>	1.38	0.71	1.44	0.51	1.20	0.49	0.10	10.00
Level6	<b>1.88</b>	1.05	0.73	1.62	0.56	1.14	0.66	0.17	10.00
Level7	0.60	0.07	<b>0.66</b>	0.03	0.14	0.00	0.13	0.00	10.00

972 Table 9: Performance analysis based on average scores in Tempestrun  
973

974 Level	975 GPT-4o	976 Gemini 2.0-flash	977 Qwen2.5VL 72B	978 Qwen2VL 7B	979 Qwen2VL 72B	980 InternVL2.5 8B	981 InternVL2.5 78B	982 Random	983 Human
984 Level1	985 466.25	986 478.35	987 <b>549.98</b>	988 446.92	989 519.22	990 444.71	991 475.22	992 385.72	993 2000.00
994 Level2	995 361.44	996 356.05	997 349.06	998 352.76	999 <b>370.13</b>	1000 327.38	1001 333.37	1002 271.65	1003 1500.00
1004 Level3	1005 213.73	1006 197.91	1007 <b>238.74</b>	1008 208.75	1009 220.21	1010 197.71	1011 216.64	1012 190.71	1013 1000.00
1014 Level4	1015 177.60	1016 <b>201.67</b>	1017 192.79	1018 182.91	1019 195.19	1020 174.58	1021 199.78	1022 157.17	1023 800.00

Depending on the differences in the models and the randomness of the games and reasoning, as well as other further experiments, the full research project may require **more** compute than the experiments reported here. The time of execution of the experiment depends on the network environment and computational power.

984 Table 10: **SuperMario** Average Frames and Tokens Consumed  
985

986 Metric	987 Level 1	988 Level 2	989 Level 3	990 Level 4	991 Level 5	992 Level 6	993 Level 7	994 Level 8	995 Level 9	996 Level 10	997 All
998 Average Frames	999 400	1000 655.56	1001 1000	1002 641.2	1003 234.00	1004 300	1005 300	1006 148.06	1007 504.35	1008 950.45	1009 5133.62
1010 Average Prompt Tokens	1011 150004.78	1012 253457.38	1013 379649.92	1014 266125.09	1015 100595.30	1016 112241.57	1017 111288.00	1018 54231.22	1019 192642.31	1020 361208.32	1021 1981443.89
1022 Average Completion Tokens	1023 10054.86	1024 18437.53	1025 25428.29	1026 18242.65	1027 7075.53	1028 7639.30	1029 7314.46	1030 3710.36	1031 13492.79	1032 24197.61	1033 135593.38

998 Table 11: **Race** Average Frames and Tokens Consumed  
999

1000 Metric	1001 Level 1	1002 Level 2	1003 Level 3	1004 Level 1	1005 Level 2	1006 Level 3	1007 Level 4	1008 Level 5	1009 Level 6	1010 All
	1011 No History	1012 No History	1013 No History	1014 Level 1	1015 Level 2	1016 Level 3	1017 Level 4	1018 Level 5	1019 Level 6	1020 All
1021 Average Frames	1022 12.66	1023 15.39	1024 16.66	1025 29.20	1026 30.69	1027 32.14	1028 58.07	1029 98.06	1030 32.46	1031 325.33
1032 Average Prompt Tokens	1033 1738.83	1034 2309.85	1035 2562.15	1036 12317.22	1037 14044.87	1038 14934.32	1039 31164.03	1040 54346.81	1041 17399.59	1042 255136.23
1043 Average Completion Tokens	1044 275.35	1045 531.41	1046 595.91	1047 693.11	1048 937.66	1049 1060.77	1050 2243.55	1051 3898.42	1052 1517.55	1053 20798.72

1004 Table 12: **FlappyBird** Average Frames and Tokens Consumed  
1005

Metric	Level 1	Level 2	Level 3	Level 4	Level 5	Level 6	Level 7	All
Average Frames	224.73	133.34	76.49	153.11	153.87	152	143.28	1036.82
Average Prompt Tokens	98273.78	57332.59	32326.91	65853.39	66500.27	65322.27	56528.63	442137.84
Average Completion Tokens	9979.17	5772.39	3319.76	7142.32	7309.31	7082.73	6259.62	46865.30

1014 Table 13: **TempestRun** and **PongGame** Average Frames and Tokens Consumed  
1015

Metric	Level 1	Level 2	Level 3	Level 4	All	Level 1	Level 2	Level 3	All
Average Frames	173.58	92.70	38.98	28.72	333.98	221.79	83.98	47.00	352.77
Average Prompt Tokens	108291.56	57096.18	33218.84	22874.80	237820.07	136254.76	50056.53	26981.30	213292.59
Average Completion Tokens	7000.84	3799.28	2316.48	1619.33	15942.53	10998.40	4064.67	2208.39	17271.46

1026  
1027

## B.2.3 MORE MODELS

1028  
1029  
1030  
1031

In addition to the models mentioned in Table B.1.3, we also evaluated the more recent **Claude-3.7-sonnet**<sup>1</sup>. Due to budgetary constraints, we were only able to conduct approximately 5 to 10 rounds of testing. When we included Claude-3.7-sonnet in the ELO calculation, the results are shown in Table 14.

1032  
1033  
1034

Table 14: Performance comparison across different games based on the elo ranking system.

Model	Pong	Race	Flappybird	Tempestrun	SuperMario	Average
<i>Closed-Source Models</i>						
<b>Claude-3.7-sonnet</b>	<b>1607</b>	<b>1626</b>	<b>1578</b>	1513	<b>1601</b>	<b>1591</b>
<b>GPT-4o</b>	1487	1582	1573	1514	1512	1526
<b>Gemini-2.0-Flash</b> (Thinking)	1518	1550	1533	1498	1588	1553
<b>Gemini-2.0-Flash</b>	1502	1498	1513	1515	1512	1510
<i>Open-Source Models</i>						
<b>Qwen2VL-7B</b>	1464	1417	1438	1488	1361	1412
<b>Qwen2VL-72B</b>	1479	1527	1521	1530	1580	1543
<b>Qwen2.5VL-72B</b>	1485	1489	1440	<b>1531</b>	1509	1494
<b>InternVL2.5-8B</b>	1489	1442	1481	1471	1372	1428
<b>InternVL2.5-78B</b>	1492	1447	1481	1514	1546	1510
<i>Baseline</i>						
Random	1477	1424	1440	1424	1419	1431

1055

It is important to note that these results may be biased because the number of evaluation rounds is incomplete compared to other models, which is why the **Claude-3.7-sonnet** model was not included in the main results discussed.

1056  
1057

Nevertheless, based on the current findings, it is one of the best-performing models on V-MAGE to date.

1061

## B.3 INCONSISTENCY BETWEEN ELO AND PERFORMANCE RATIO RANKINGS

1063  
1064

As shown in Table 2 in the main text, ELO and Performance Ratio sometimes do not align in rankings.

1065  
1066  
1067

We examine **Keye-VL-8B-Preview** and **Qwen2.5-VL-7B-Instruct**, with **LLaVA-v1.6-Mistral-7B** as a control.

1068  
1069

Table 15: Elo Scores and Average Performance Ratios (E/R) Across Games.

	Race(E/R)	SuperMario(E/R)	Pong(E/R)	FlappyBird(E/R)	Tempestrun(E/R)
Qwen2.5-VL-7B-Instruct	1487/0.120	1459/0.239	<b>1503/0.035</b>	1431/0.030	1485/0.210
Keye-VL-8B-Preview	1487/0.118	1430/0.217	<b>1495/0.039</b>	1450/0.044	1513/0.239
LLaVA-v1.6-Mistral-7B	1462/0.051	1374/0.127	<b>1494/0.035</b>	1489/0.077	1379/0.169

1076  
1077  
1078  
1079

In Pong, Qwen shows higher ELO but lower average ratio. We analyzed level-wise scores and variances to explore this. The variance is calculated as: variance =  $\frac{\sum_{i=1}^n (\text{score}_i - \text{score})^2}{n}$ .

<sup>1</sup>Anthropic, <https://www.anthropic.com/clause/sonnet>

1080  
1081  
1082 Table 16: Pong Scores by Level (Avg: average score, Var: variance).  
1083  
1084  
1085  
1086  
1087  
1088

	<b>L1 Avg</b>	<b>L1 Var</b>	<b>L2 Avg</b>	<b>L2 Var</b>	<b>L3 Avg</b>	<b>L3 Var</b>
Qwen2.5-VL-7B-Instruct	0.48	0.50	0.37	0.25	0.20	0.18
Keye-VL-8B-Preview	0.68	<b>0.67</b>	0.26	<b>0.33</b>	0.23	<b>0.36</b>
LLaVA-v1.6-Mistral-7B	0.48	0.58	0.29	0.26	0.29	0.34

1089  
1090  
1091  
1092  
1093  
Keye’s higher variance across all Pong levels indicates unstable performance, where high-scoring  
1094  
1095  
1096  
1097  
1098  
1099  
outliers mask frequent weak results. In the ELO system, this instability leads to more losses against  
a consistent opponent, resulting in a lower rating despite a competitive average score.

1100  
1101  
1102  
1103  
Additionally, current models perform poorly on Pong, with ratios tightly clustered in the 0–10%  
1104  
1105  
1106  
1107  
1108  
range. When calculating the performance ratio by averaging across games, minor differences in  
1109  
1110  
1111  
1112  
1113  
1114  
1115  
Pong (3.5% vs. 3.9%) are overshadowed by larger gaps in other games(21% vs. 24%). The ELO  
1116  
1117  
1118  
1119  
1120  
system, in contrast, is based on the aggregate outcomes of all pairwise matchups. The ELO rating  
1121  
1122  
1123  
1124  
1125  
1126  
1127  
1128  
1129  
1130  
1131  
1132  
1133  
boost from a consistent pattern of wins in Pong is just as significant as from wins in any other game.  
This demonstrates that ELO is more robust in **fairly** assessing a model’s holistic capabilities across  
tasks with imbalanced performance levels.

1104  
1105  
1106  
1107  
1108  
1109  
1110  
1111  
1112  
1113  
1114  
1115  
1116  
1117  
1118  
1119  
1120  
1121  
1122  
1123  
1124  
1125  
1126  
1127  
1128  
1129  
1130  
1131  
1132  
1133  
We also observed that in terms of Response Format Accuracy, GPT-4o is slightly lower than Gemini  
model (by 0.04%), and InternVL2.5-78B is slightly lower than Qwen2-VL-72B (by 0.25%). This  
may also be an influencing factor.

1134 **C GAMES IN V-MAGE**  
11351136 **C.1 PRINCIPLES AND STANDARDS FOR GAME SELECTION**  
11371138 **Simplified and unrealistic considerations.** While the simplified visuals in these games differ  
1139 from real-world scenes, empirical evidence demonstrate that MLLMs comprehend core game seman-  
1140 tics (objectives, rules, entities) despite stylistic simplifications. Performance limitations primar-  
1141 ily emerge from perceptual inaccuracies (e.g., dynamic object tracking) and multi-step reasoning  
1142 deficiencies rather than misinterpretation. V-MAGE therefore focuses more on **precise evaluation**  
1143 than **visual realism** to drive targeted improvements in visual reasoning.1144 **Selection criteria.** The five games in V-MAGE share critical characteristics (e.g., non-textual  
1145 state representation, free-form gameplay, and continuous-space environments) while offering di-  
1146 verse challenges.1147 Our current minimal set covers four **2D** game types through this matrix:  
11481149 Table 17: 2D Game Taxonomy in V-MAGE  
1150

	<b>XY-axis</b>	<b>XZ-axis</b>
<b>Linear Process</b>	PongGame	FlappyBird
<b>Open Planning</b>	RaceGame	SuperMario

1151 The Linear Process implies that the game’s progression is, to some extent, enforced. In PongGame,  
1152 the ball’s movement direction is determined by the game environment, requiring the model to move  
1153 paddles on both sides to catch the ball, while in FlappyBird, the forward movement of the bird  
1154 is compulsory, with the model controlling the height to navigate through pipes. OpenPlanning,  
1155 in contrast, is relatively more open-ended. In RaceGame, the model can freely control the car’s  
1156 movement and direction on a plane to reach a trophy. In SuperMario, the model can move and jump  
1157 in a relatively open environment to collect rewards and earn points.1158 For **3D** environments, we selected **Tempest Run** for its streamlined visual elements.  
11591160 V-MAGE’s flexible framework allows seamless integration of new PyGame-based environments.  
1161 For instance, Tempest Run (one of our five games) was sourced from PyWeek[3], a community-  
1162 driven game jam with thousands of open-source entries. This demonstrates our framework’s capacity  
1163 to incorporate externally developed, human-designed games. We provide APIs to wrap new games  
1164 into V-MAGE’s evaluation pipeline. This allows researchers to easily integrate additional games.1165 We will continue expanding the benchmark with more diverse titles that meet our selection criteria  
1166 (e.g., Player vs Player (PVP) games) and will open-source both the codebase and detailed docu-  
1167 mentation to facilitate community contributions.  
11681169 **C.2 DESIGN AND IMPLEMENTATION**  
11701171 As previously mentioned, V-MAGE enhances the diversity of the evaluation environment by expand-  
1172 ing it through level design. Tables in this section detail the settings, rewards, and design objectives  
1173 for each game’s levels. For more comprehensive visual comparisons and prompt information, please  
1174 refer to Appendix C.4.1175 **Race Game** is a skill-based driving game where the objective is to control a car through a maze-like  
1176 track to reach the trophy while avoiding obstacles. The car is represented by a red or white vehicle  
1177 with a visible front and back, while the trophy is shown as a golden cup icon. The surrounding  
1178 white-lined boundaries represent walls, which the car must avoid. For the overall observation and  
1179 action spaces of the game, including the task and reward definitions, please refer to Table 18.  
11801181 Each level in Race has a different set of rules and challenges. As presented in Table 19, we manually  
1182 designed six levels. Levels 1–3 use a *map-view perspective* (“map” view), where models adjust ab-  
1183 solute coordinates. The four types of movement operations directly translate the vehicle on the map  
1184

Table 18: Race Environment Details (\* means potentially requires observation).

Observation space	Action Space	Task	Reward
Car Position	UP, DOWN,	Move the car	+100 Success
Trophy Position	LEFT, RIGHT	to reach the trophy	+0 Timeout
Obstacle Position*			+0 Destroyed
Speed*			
Acceleration*			
Facing Angle*			

according to the direction of action. Conversely, Levels 4–6 shift to a *first-person perspective* (“car” view), the observation is centered on the vehicle, and movements are performed based on the vehicle’s perspective, Requiring real-time interpretation of velocity vectors and acceleration constraints. Furthermore, acceleration is introduced in the high-difficulty levels, which further expands the observation space. This requires the model to extract more information from the visual input, including current speed and acceleration, in order to perform rational reasoning.

Table 19: Race Level Configurations

Level	View	bstacle	Initial Direction	Acceleration	Max Rounds	Sample Frames
1	Map	No	-	No	100	1
2	Map	Yes	-	No	150	1
3	Map	Yes	-	No	150	1
4	Car	No	Vertical(up)	Low	150	3
5	Car	No	Horizontal(random)	Mid	150	3
6	Car	Yes	Vertical(up)	Mid	150	1

**SuperMario** is a two-dimensional side-scrolling platformer where the player controls the character Mario navigating through environments populated with various platforms, enemies, and obstacles. The goal is to traverse the level, collect coins, evade or defeat enemies, and reach the flagpole at the stage’s conclusion. Players must avoid falling off platforms, colliding with enemies, or being struck by obstacles. Successful gameplay involves employing optimal movement combinations for smooth and efficient progression, alongside monitoring Mario’s position relative to environmental elements. Task and reward definition is shown in Table 20.

Table 20: SuperMario Environment Details.

Observation space	Action Space	Task	Reward
Mario Position	UP, UP+LEFT,	Collect coins and	+100 for collecting a coin
Platforms Position	UP+RIGHT, LEFT,	evade or defeat	+100 for defeating a Goomba
Enemies Position	RIGHT, NONE	enemies	Penalties for falling or collisions
Obstacles Position			

Table 21: SuperMario Level Configurations

Level	Enemy count	Coin Count	CoinBox Count	Max Rounds	Gameplay
1	0	6	2	400	Common
2	2	6	2	1000	Common
3	0	17	4	1000	Long History (Two ways)
4	2	17	4	1000	Long History (Two ways)
5	3	8	0	300	Left or Right
6	0	13	0	300	Left or Right
7	0	8	0	300	Left or Right
8	0	12	0	1000	Jump Only
9	5	8	0	1000	Jump and Enemy
10	12	0	9	5000	Classic W1-1

To provide a comprehensive evaluation of MLLMs' visual reasoning and planning capabilities, SuperMario features ten levels with configurations detailed in Table 21. These levels vary in enemy count, coin and coinbox quantities, maximum allowed rounds, and specific gameplay mechanics or focuses. Of these, level 10 serves as a standard human-difficulty benchmark, providing a 1:1 replica of the original Super Mario game's World 1-1 stage.

**Flappy Bird** is a widely recognized side-scrolling mobile game serving as a common benchmark in reinforcement learning. The objective is to control a bird's vertical movement to navigate through a continuous series of horizontal gaps within vertically oriented pipes. Successful traversal of a pipe pair increments the player's score, while collision with any pipe or the ground constitutes a terminal state, ending the game. The game mechanic involves a constant downward gravitational pull, counteracted by discrete upward 'flaps' initiated by the player.

Table 22: Flappy Bird Environment Details (\* means only available at certain levels).

Observation space	Action Space	Task	Reward
Bird Position	UP	Maneuver the bird to avoid hitting the pipes	+1 per pipe pair passed
Bird Velocity	NONE		+0 Collision
Next Pipe Distance	DOWN*		
Gap VerticalPosition	KEEP*		

Given the high difficulty of human-standard levels for MLLMs, we designed seven levels with progressive difficulty. Specifically, as presented in Table 23, levels 1-3 constitute a simplified game environment where the gravity factor is removed, and height is controlled via UP and DOWN actions to navigate through the pipes. Levels 4-6 are based on the standard difficulty but incorporate a 'KEEP' option, enabling the model to maintain the bird's altitude through this action. Within the same difficulty tier, levels are differentiated by varying the bird's forward speed and the pipe gap width. Level 7 represents the standard human game difficulty, retaining the original game settings.

Table 23: FlappyBird Level Configurations

Level	Gravity	Availability of "DOWN"	Availability of "KEEP"	Others
1 - 3	No	Yes	Yes	Distinguished by gap clearance and speed
4 - 6	Yes	No	Yes	
7	Yes	No	No	Human Standard

**Pong Game** is a classic two-player adversarial game. The objective is to control the paddles on the left and right sides of the screen to return the ball, preventing it from passing one's own paddle while simultaneously attempting to make the ball pass the opponent's paddle. One point is awarded to the player for each successful return of the ball. The final score is the sum of both players' scores. Task and reward definition is shown in Table 24.

Table 24: Pong Game Environment Details.

Observation space	Action Space	Task	Reward
Left Paddle Position	LEFTUP	Track the ball's trajectory	+1 per successful hit
Right Paddle Position	LEFTDOWN	and maneuver the left and right paddles to intercept	+0 if ball passes paddle
Ball Position	RIGHTUP		
Ball Trajectory	RIGHTDOWN	and return the ball.	
	NONE		

Table 25: Pong Game Level Configurations

Level	Paddle Width	Ball Speed	Ball Size	Others
1	Big	Slow	Big	Ball initial position randomly changes.
2	Mid	Mid	Mid	
3	Small	Fast	Small	

Considering the challenges MLLMs face in tracking and temporal tasks, we designed levels with varying difficulty. As shown in Table 25, difficulty for Levels 1-3 is differentiated by adjusting the

1296 paddle width and the speed of the ball. Within the same level, the initial position of the ball is  
 1297 randomized, but the relative difficulty remains consistent.  
 1298

Table 26: Tempest Run Environment Details.

Observation space	Action Space	Task	Reward
Current Character State	JUMP, LEFT, RIGHT,	Perform corresponding	Score increases
Nearby Obstacles Position	SLIDE, RISE, NONE	actions to avoid or	with distance run.
Nearby Obstacles Type		destroy obstacles.	
Visual Information Quantity			

1300  
 1301 **Tempest Run** is a third-person perspective 3D runner game where the player controls a character  
 1302 moving within a futuristic tunnel filled with various obstacles and enemies. The objective is to  
 1303 navigate through the tunnel, avoiding or overcoming impediments, and to run as far as possible.  
 1304 Players must specifically avoid colliding with red spikes, purple walls, or failing to manage green  
 1305 enemies. Successful gameplay requires employing optimal combinations of movements for smooth  
 1306 and efficient tunnel traversal, alongside monitoring the character’s position relative to obstacles and  
 1307 reacting appropriately. Task and reward definition is shown in Table 26.  
 1308

1309 To evaluate MLLMs’ visual comprehension and reactive capabilities within a dynamic 3D environ-  
 1310 ment, Tempest Run includes four levels of varying difficulty. As outlined in Table 27, Levels 1-4 are  
 1311 primarily differentiated by parameters including role speed, cell length (denoting the distance be-  
 1312 tween environmental segments), and random rate (controlling obstacle spawning frequency). These  
 1313 parameters collectively influence the pace of barrier generation and the overall visual complexity of  
 1314 the tunnel environment, thereby varying the level of challenge. Within the same level, the position-  
 1315 ing of environmental elements is randomized, while maintaining consistent relative difficulty.  
 1316

Table 27: Tempest Run Level Configurations

Level	Role Speed	Cell Length	Random Rate	Others
1	Slow	Large	Low	Environmental elements initial positions randomly change.
2	Medium	Medium	Medium-Low	
3	Fast	Small	Medium-High	
4	Very Fast	Small	High	

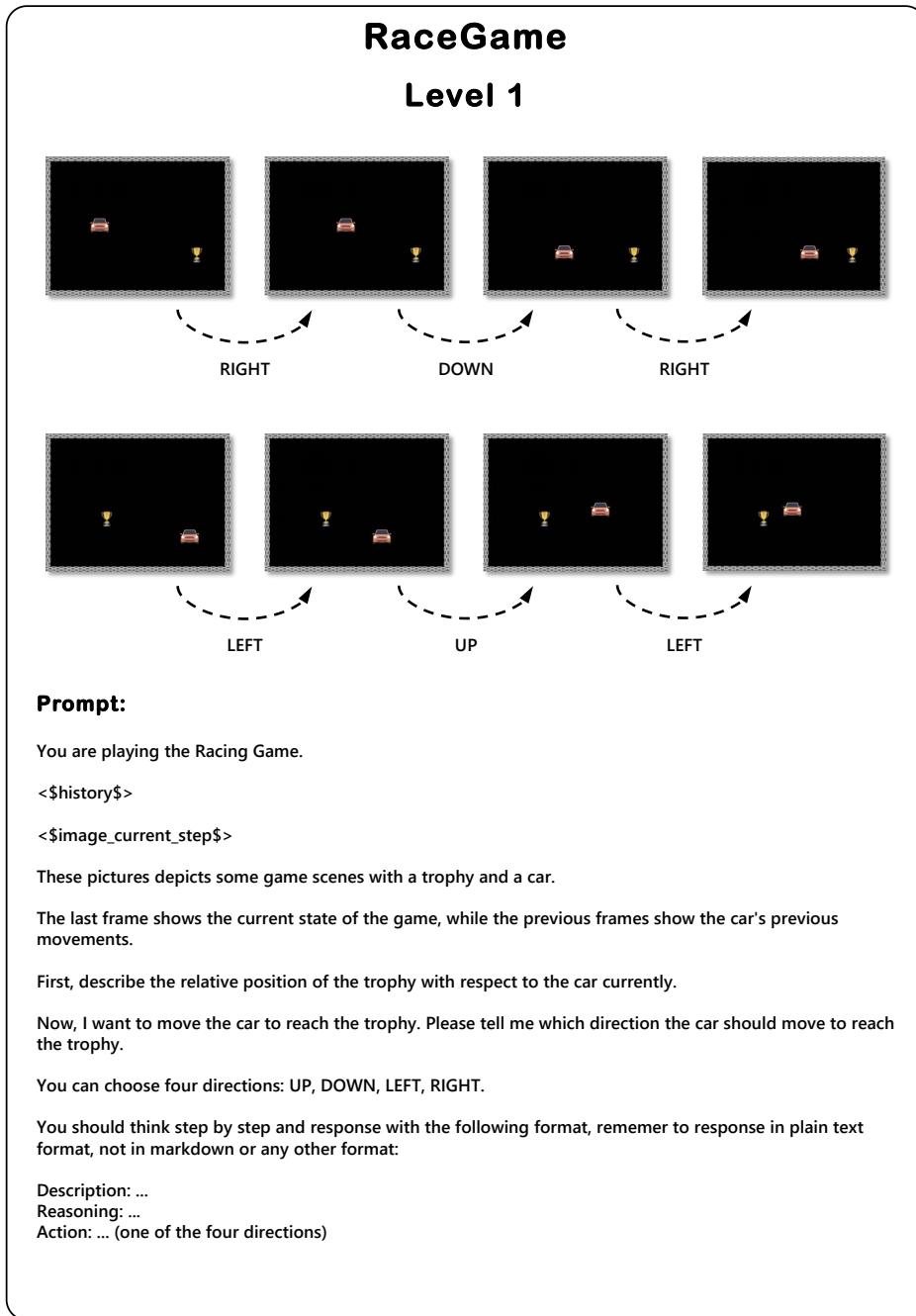
### C.3 ORIGINAL SOURCES

1328 Thanks to the open-source community, we are able to leverage existing game codebases to build our  
 1329 benchmark. Here are the codebases we used:  
 1330

Table 28: Game Codebase Sources

Game	Codebase
Race	<a href="https://github.com/tdostilio/Race_Game">https://github.com/tdostilio/Race_Game</a>
FlappyBird	<a href="https://github.com/agneay/pygame-projects/tree/master/Flappy%20Bird">https://github.com/agneay/pygame-projects/tree/master/Flappy%20Bird</a>
Pong	<a href="https://github.com/pyGuru123/Python-Games/tree/master/Pong">https://github.com/pyGuru123/Python-Games/tree/master/Pong</a>
SuperMario	<a href="https://github.com/mx0c/super-mario-python">https://github.com/mx0c/super-mario-python</a>
Tempest Run	<a href="https://github.com/davidpendergast/pygame-summer-team-jam">https://github.com/davidpendergast/pygame-summer-team-jam</a>

1344 In most cases, the original codebases lacked comprehensive difficulty settings and level designs  
 1345 suitable for systematic evaluation. We therefore modified the default human-oriented game config-  
 1346 urations to adapt them for benchmarking purposes, while meticulously designing a diverse set of  
 1347 challenging levels to ensure rigorous assessment.  
 1348

1350 C.4 GAMES AND PROMPTS  
13511352 All the games have been modified based on publicly available code. The detailed design is provided  
1353 below:  
1354  
13551399 Figure 9: **RaceGame Level 1: Level Design and Prompt Overview.** The images showcase the  
1400 scene from Level 1, illustrating the level design and corresponding prompt. Elements in the same  
1401 level will randomly change their initial positions while maintaining consistent relative difficulty.  
1402  
1403

1404  
1405  
1406  
1407  
1408  
1409  
1410  
1411  
1412  
1413  
1414  
1415  
1416  
1417  
1418  
1419  
1420  
1421  
1422  
1423  
1424  
1425  
1426  
1427  
1428  
1429  
1430  
1431  
1432  
1433  
1434  
1435  
1436  
1437  
1438  
1439  
1440  
1441  
1442  
1443  
1444  
1445  
1446  
1447  
1448  
1449  
1450  
1451  
1452  
1453  
1454  
1455  
1456  
1457

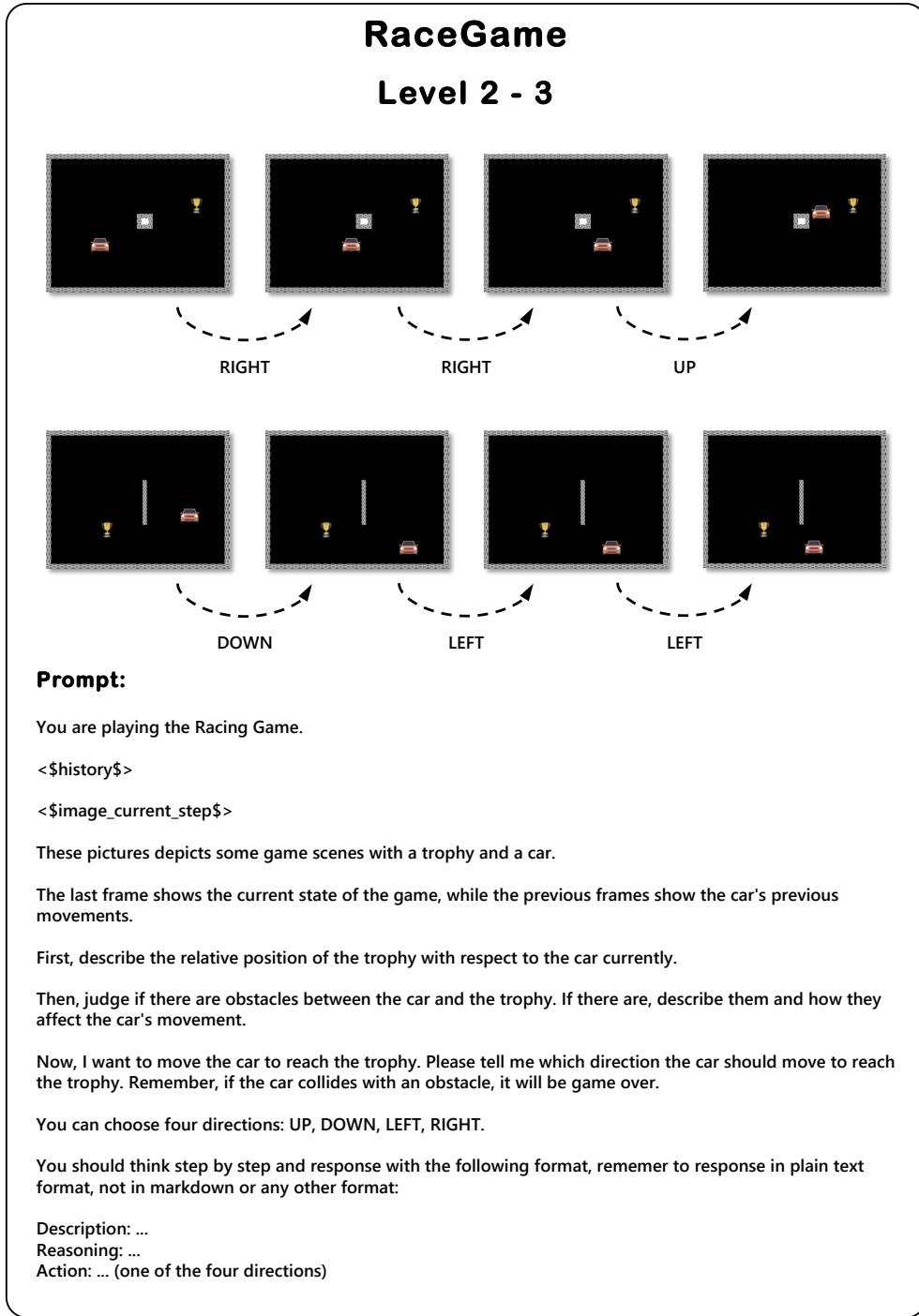


Figure 10: **RaceGame Level 2-3: Level Design and Prompt Overview.** The images showcase the scene from Level 2-3, illustrating the level design and corresponding prompt. Elements in the scene will randomly change their initial positions while maintaining consistent relative difficulty.

1458

1459

1460

1461

1462

1463

1464

1465

1466

1467

1468

1469

1470

1471

1472

1473

1474

1475

1476

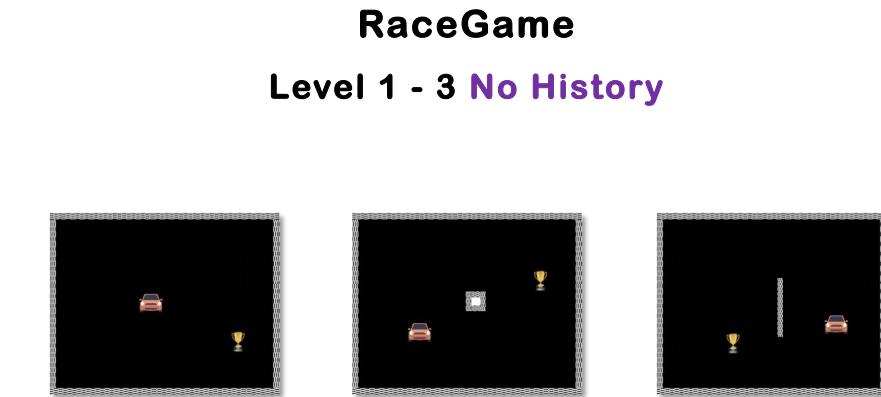
1477

1478

1479

1480

1481



**Prompt:**

You are playing the Racing Game.

<\$image\_current\_step\$>

These pictures depicts some game scenes with a trophy and a car.

The last frame shows the current state of the game, while the previous frames show the car's previous movements.

First, describe the relative position of the trophy with respect to the car currently.

(Only for Level 2/3)

Then, judge if there are obstacles between the car and the trophy. If there are, describe them and how they affect the car's movement.

Now, I want to move the car to reach the trophy. Please tell me which direction the car should move to reach the trophy. Remember, if the car collides with an obstacle, it will be game over.

You can choose four directions: UP, DOWN, LEFT, RIGHT.

You should think step by step and response with the following format, rememer to response in plain text format, not in markdown or any other format:

Description: ...

Reasoning: ...

Action: ... (one of the four directions)

1506

1507

1508

1509

1510

1511

Figure 11: **RaceGame Level 1-3 No History: Level Design and Prompt Overview.** The images showcase the scene from Level 1-3 No History, illustrating the level design and corresponding prompt. Elements in the scene will randomly change their initial positions while maintaining consistent relative difficulty. Same as the original levels except the input sequence has been changed to the single image.

1512  
1513  
1514  
1515  
1516  
1517  
1518  
1519  
1520  
1521  
1522  
1523  
1524  
1525  
1526  
1527  
1528  
1529  
1530  
1531  
1532  
1533  
1534  
1535  
1536  
1537  
1538  
1539  
1540  
1541  
1542  
1543  
1544  
1545  
1546  
1547  
1548  
1549  
1550  
1551  
1552  
1553  
1554  
1555  
1556  
1557  
1558  
1559  
1560  
1561  
1562  
1563  
1564  
1565

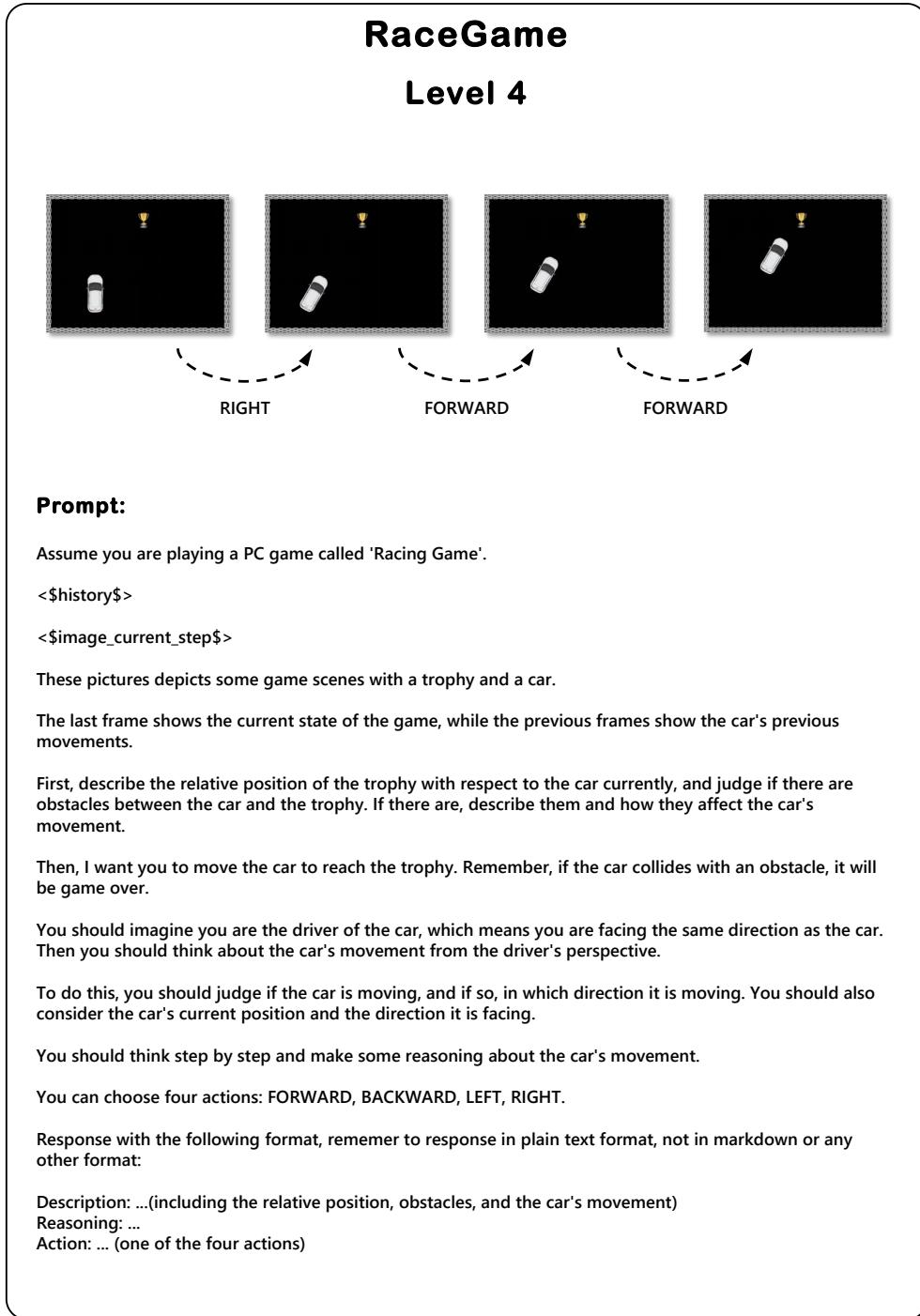
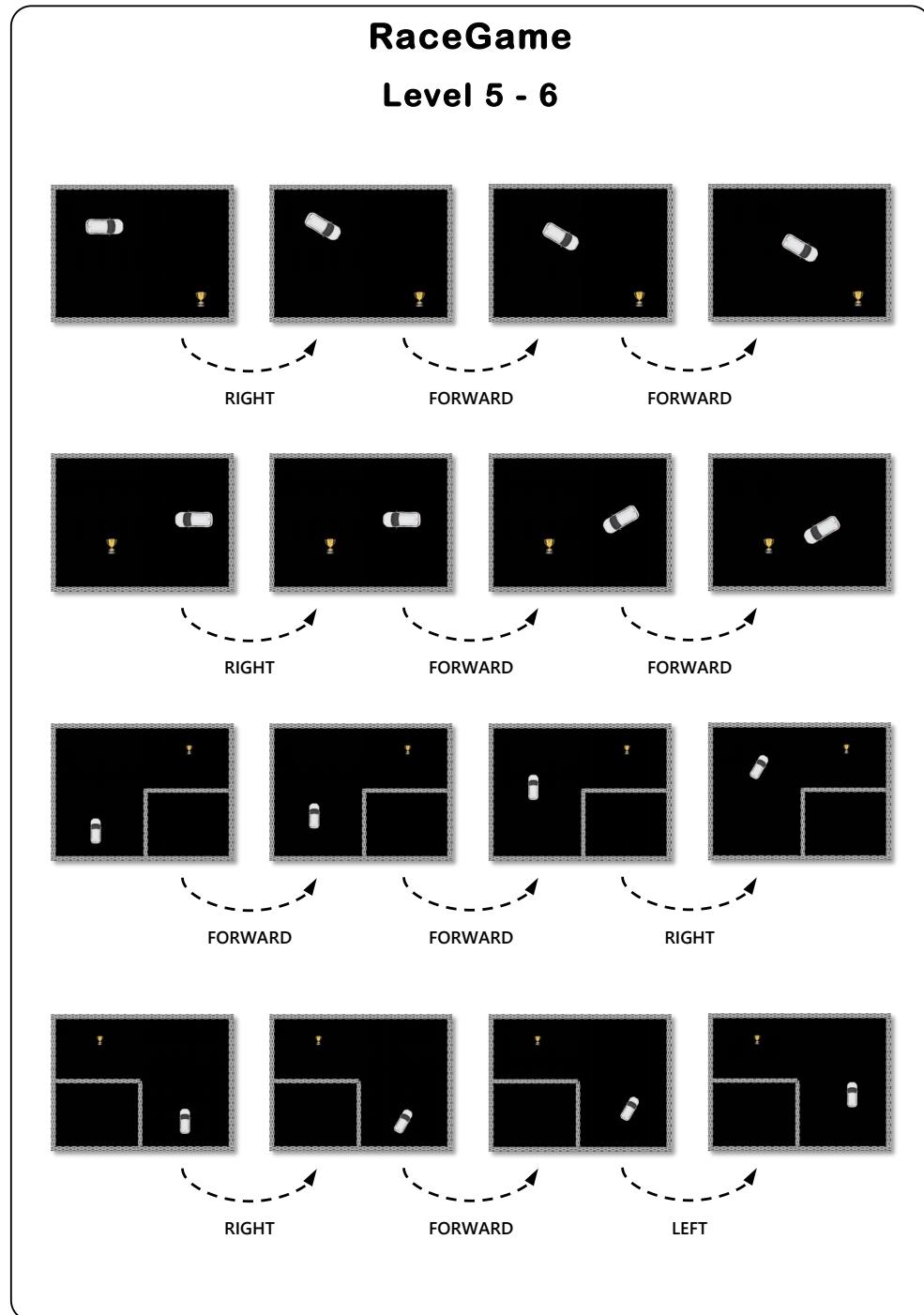


Figure 12: **RaceGame Level 4: Level Design and Prompt Overview.** The images showcase the scene from Level 4, illustrating the level design and corresponding prompt. Elements in the same level will randomly change their initial positions while maintaining consistent relative difficulty.

1566  
 1567  
 1568  
 1569  
 1570  
 1571  
 1572  
 1573  
 1574  
 1575  
 1576  
 1577  
 1578  
 1579  
 1580  
 1581  
 1582  
 1583  
 1584  
 1585  
 1586  
 1587  
 1588  
 1589  
 1590  
 1591  
 1592  
 1593  
 1594  
 1595  
 1596  
 1597  
 1598  
 1599  
 1600  
 1601  
 1602  
 1603  
 1604  
 1605  
 1606  
 1607  
 1608  
 1609  
 1610  
 1611  
 1612  
 1613  
 1614



1615 **Figure 13: RaceGame Level 5-6: Level Design and Prompt Overview.** The images showcase the  
 1616 scene from Level 5-6, illustrating the level design and corresponding prompt. Elements in the same  
 1617 level will randomly change their initial positions while maintaining consistent relative difficulty.  
 1618 **The prompt is the same as in Level 4.**

1619

1620

1621

1622

1623

1624

1625

1626

1627

1628

1629

1630

1631

1632

1633

1634

1635

**Prompt:**

1637 Assume you are playing a PC game called 'Super Mario'.

1638  
1639 You need to control Mario, who can move in a two-dimensional space consisting of various platforms, enemies, and obstacles. Your goal is to navigate through the level, collect coins, avoid or defeat enemies, and reach the flagpole at the end of the stage. Avoid falling off the platforms, colliding with enemies, or being hit by obstacles.1640 Use the optimal combination of movements to progress through the level smoothly and efficiently. Monitor  
1641 Mario's position relative to obstacles, platforms, and enemies.1642  
1643 <\$history\$>1644  
1645 <\$image\_current\_step\$>

1646 This screenshot represents the current step of the game.

1647 The last frame shows the current state of the game, while the previous frames show Mario's previous  
1648 movements.1649  
1650 \*\*Important notes:\*\*1651 1. Mario can jump (actions involving UP) only if he is on the ground or on a solid surface like a platform or  
1652 pipe.  
2. If Mario is in mid-air, he can only use LEFT or RIGHT to adjust his position, or NONE to continue falling or  
1653 moving with momentum.

1654 You can make six types of actions to control Mario:

1655 1. UP: Makes Mario jump upward (only available when Mario is on the ground or solid platforms).  
2. LEFT: Moves Mario left.  
3. RIGHT: Moves Mario right.  
4. UP+LEFT: Makes Mario jump upward and left simultaneously (only available when on the ground or solid  
1656 platforms).  
5. UP+RIGHT: Makes Mario jump upward and right simultaneously (only available when on the ground or  
1657 solid platforms).  
6. NONE: No new action is performed; Mario continues to be affected by gravity (if airborne) or momentum  
1658 from previous movements.

1659 Note that DOWN has no effect and cannot be used, so you should never attempt to use it.

1660 You should think step by step and respond with the following format, remember to respond with plain text  
1661 without any special characters or symbols, DO NOT respond in markdown or Latex format.1662  
1663 Response:1664  
1665 Observation: ... (Describe Mario's current position, nearby platforms, enemies, and obstacles.)

1666 Reasoning: ... (Think step by step and explain how you choose the action.)

1667 Action: ... (Choose one of the six actions to control Mario. Do NOT add any other words.)

1668

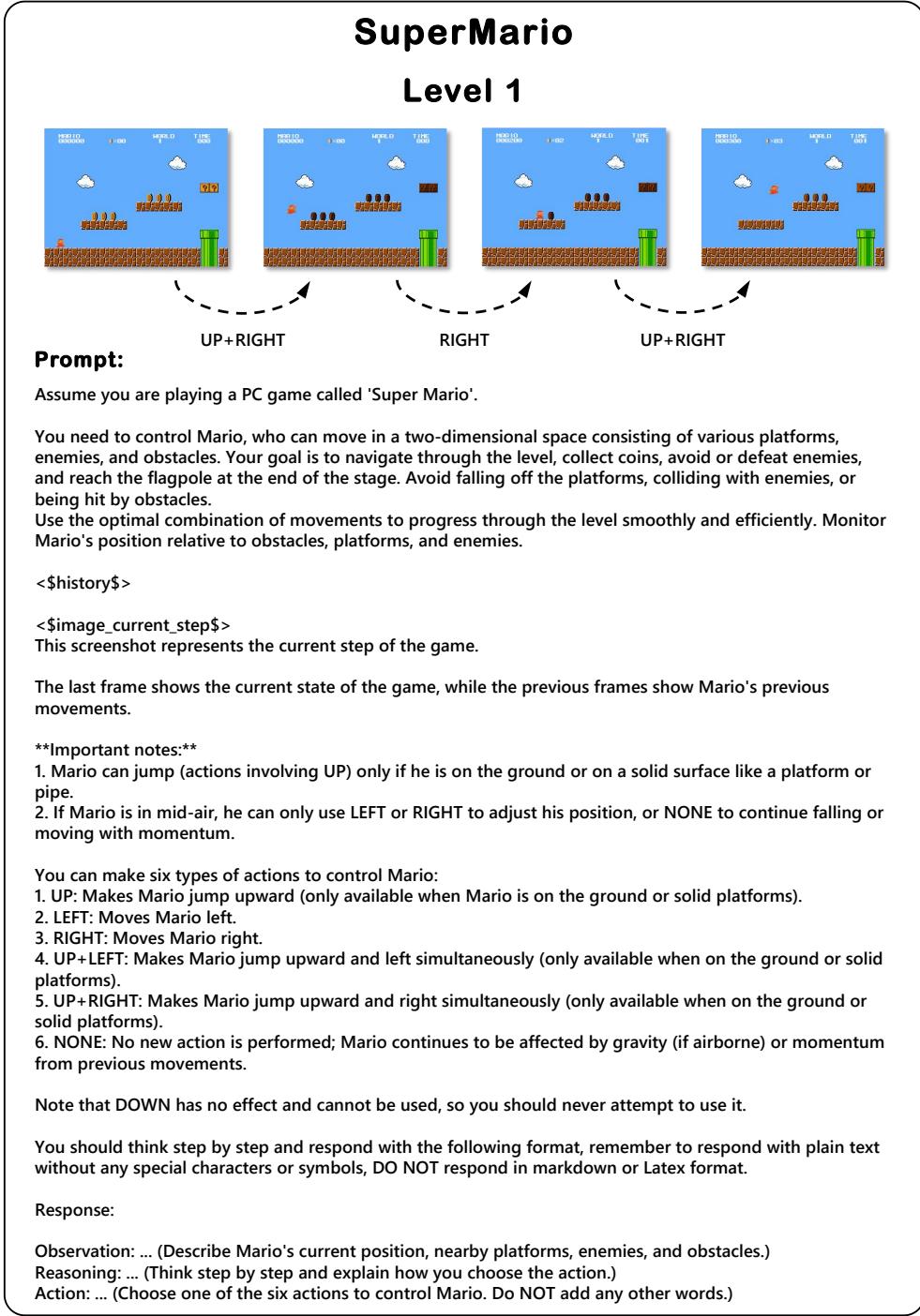
1669

1670

1671

1672

1673



1674

1675

1676

1677

1678

1679

1680

1681

1682

1683

1684

1685

1686

1687

1688

1689

1690

1691

1692

1693

1694

1695

1696

1697

1698

1699

1700

1701

1702

1703

1704

1705

1706

1707

1708

1709

1710

1711

1712

1713

1714

1715

1716

1717

1718

1719

1720

1721

1722

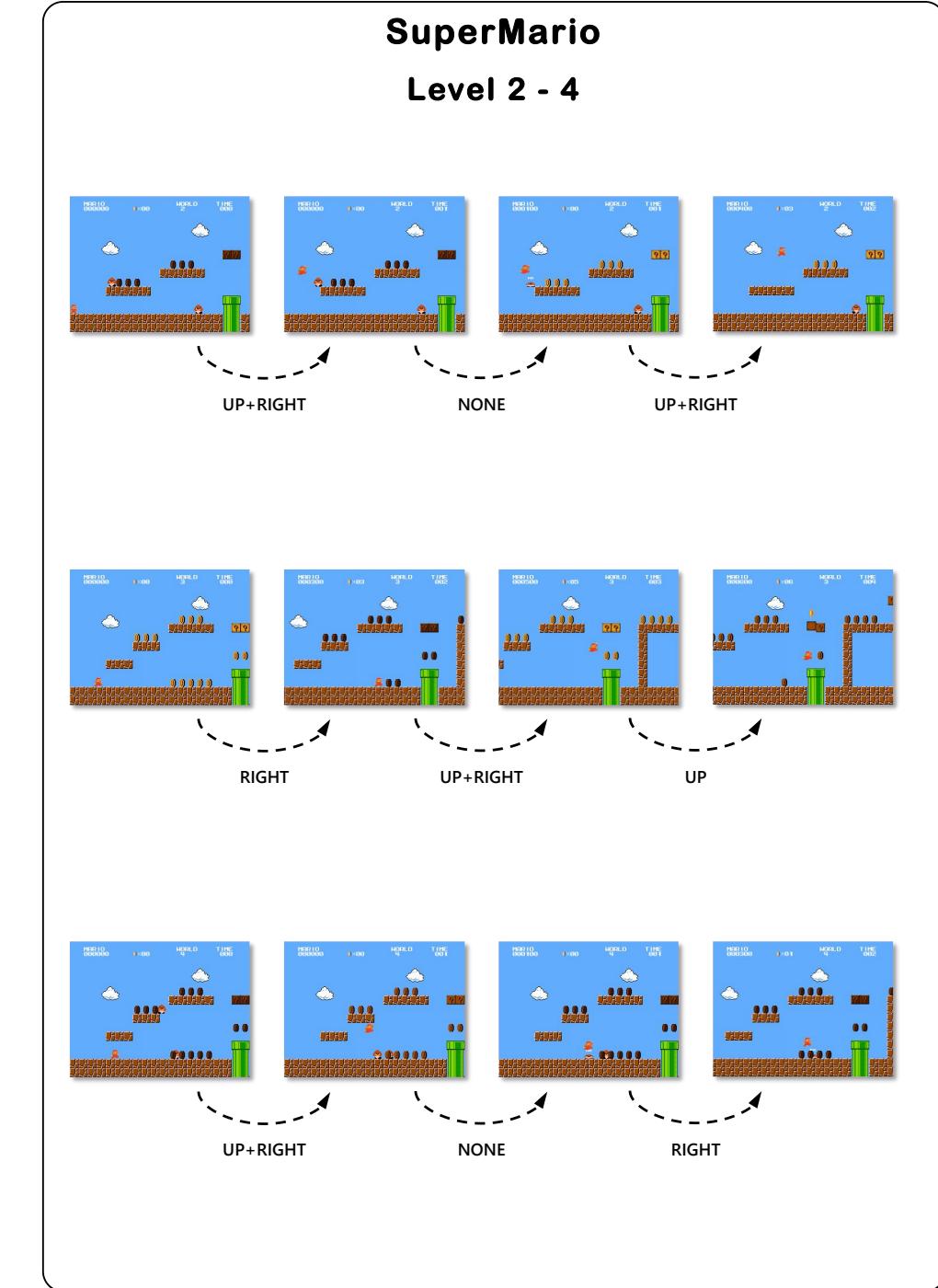


Figure 15: **SuperMario Level 2-4: Level Design and Prompt Overview.** The images showcase the scene from Level 2-4, illustrating the level design and corresponding prompt. **The prompt is the same as in Level 4.**

1724

1725

1726

1727

1728

1729

1730

1731

1732

1733

1734

1735

1736

1737

1738

1739

1740

1741

1742

1743

1744

1745

1746

1747

1748

1749

1750

1751

1752

1753

1754

1755

1756

1757

1758

1759

1760

1761

1762

1763

1764

1765

1766

1767

1768

1769

1770

1771

1772

1773

1774

1775

1776

1777

1778

1779

1780

1781

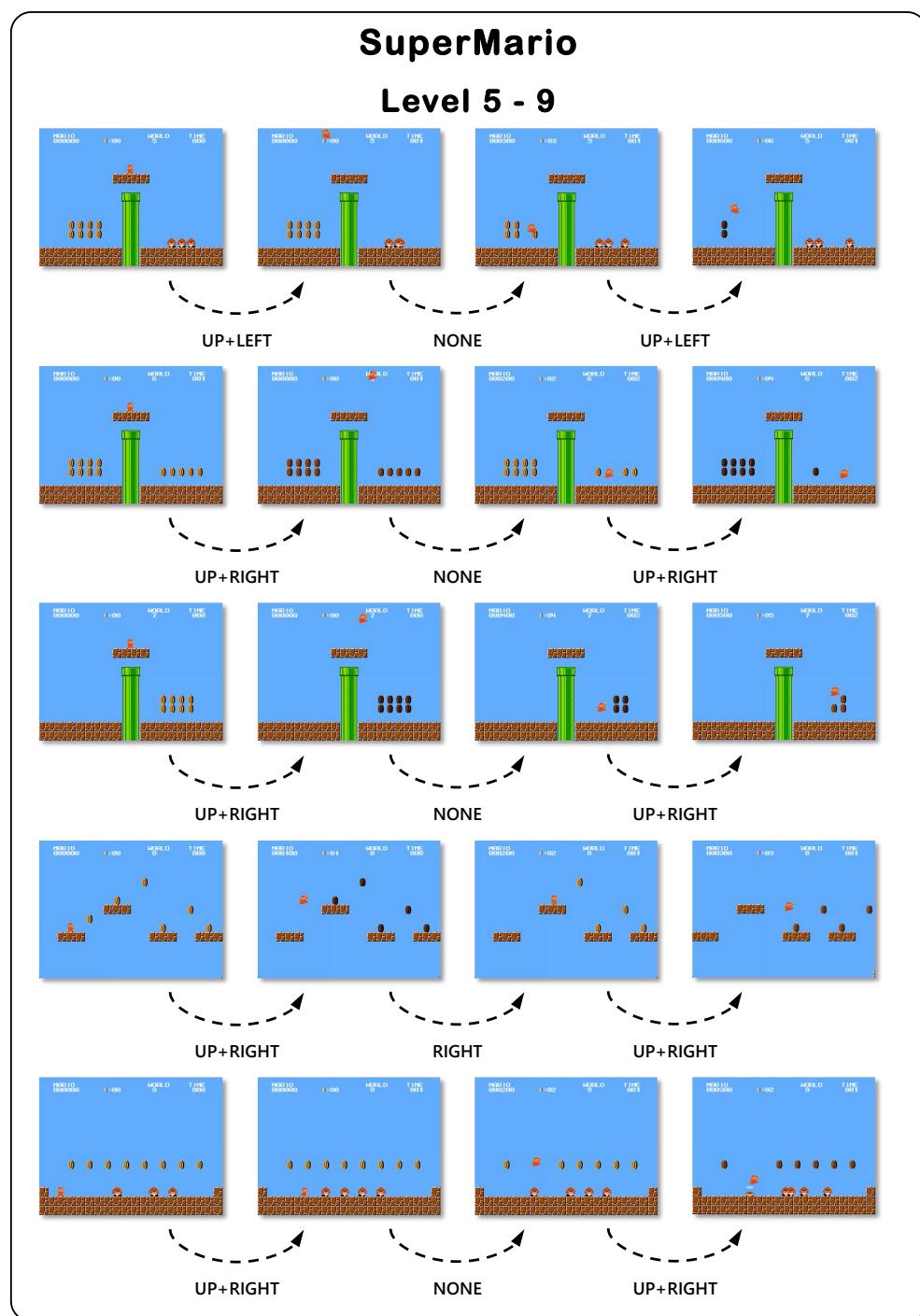


Figure 16: **SuperMario Level 5-9: Level Design and Prompt Overview.** The images showcase the scene from Level 5-9, illustrating the level design and corresponding prompt. **The prompt is the same as in Level 4.**

1782

1783

1784

1785

1786

1787

1788

1789

1790

1791

1792

1793

1794

1795

1796

1797

1798

1799

1800

1801

1802

1803

1804

1805

1806

1807

1808

1809

1810

1811

1812

1813

1814

1815

1816

1817

1818

1819

1820

1821

1822

1823

1824

1825

1826

1827

1828

1829

1830

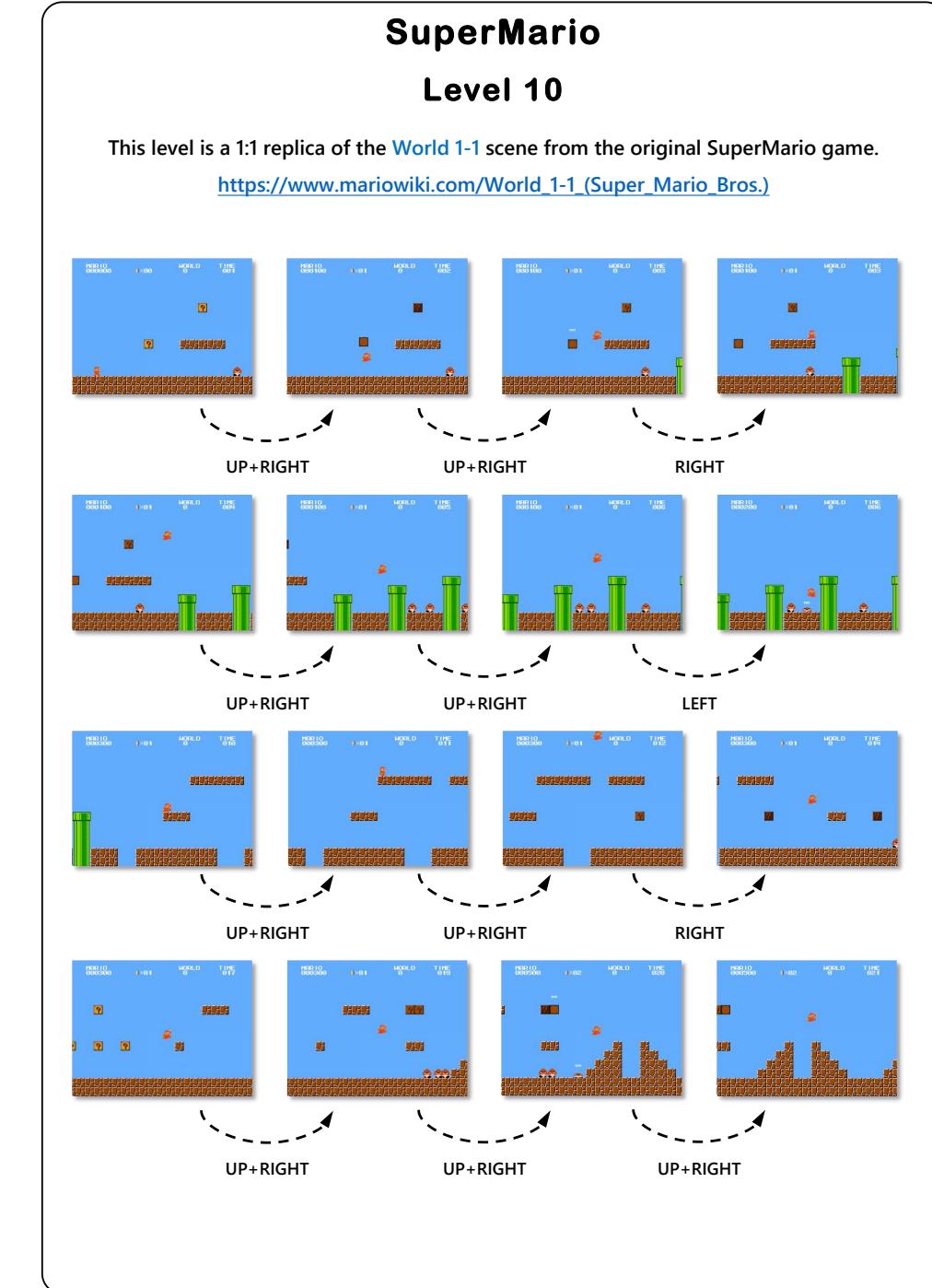


Figure 17: **SuperMario Level 10 (Standard Level): Level Design and Prompt Overview.** The images showcase the scene from Level 10, illustrating the level design and corresponding prompt. This is The standard level that matches the difficulty of the human game. **The prompt is the same as in Level 4.**

1831

1832

1833

1834

1835

1836

1837

1838

1839

1840

1841

1842

1843

1844

1845

1846

1847

1848

1849

1850

1851

1852

1853

1854

1855

1856

1857

1858

1859

1860

1861

1862

1863

1864

1865

1866

1867

1868

1869

1870

1871

1872

1873

1874

1875

1876

1877

1878

1879

1880

1881

1882

1883

1884

1885

1886

1887

1888

1889

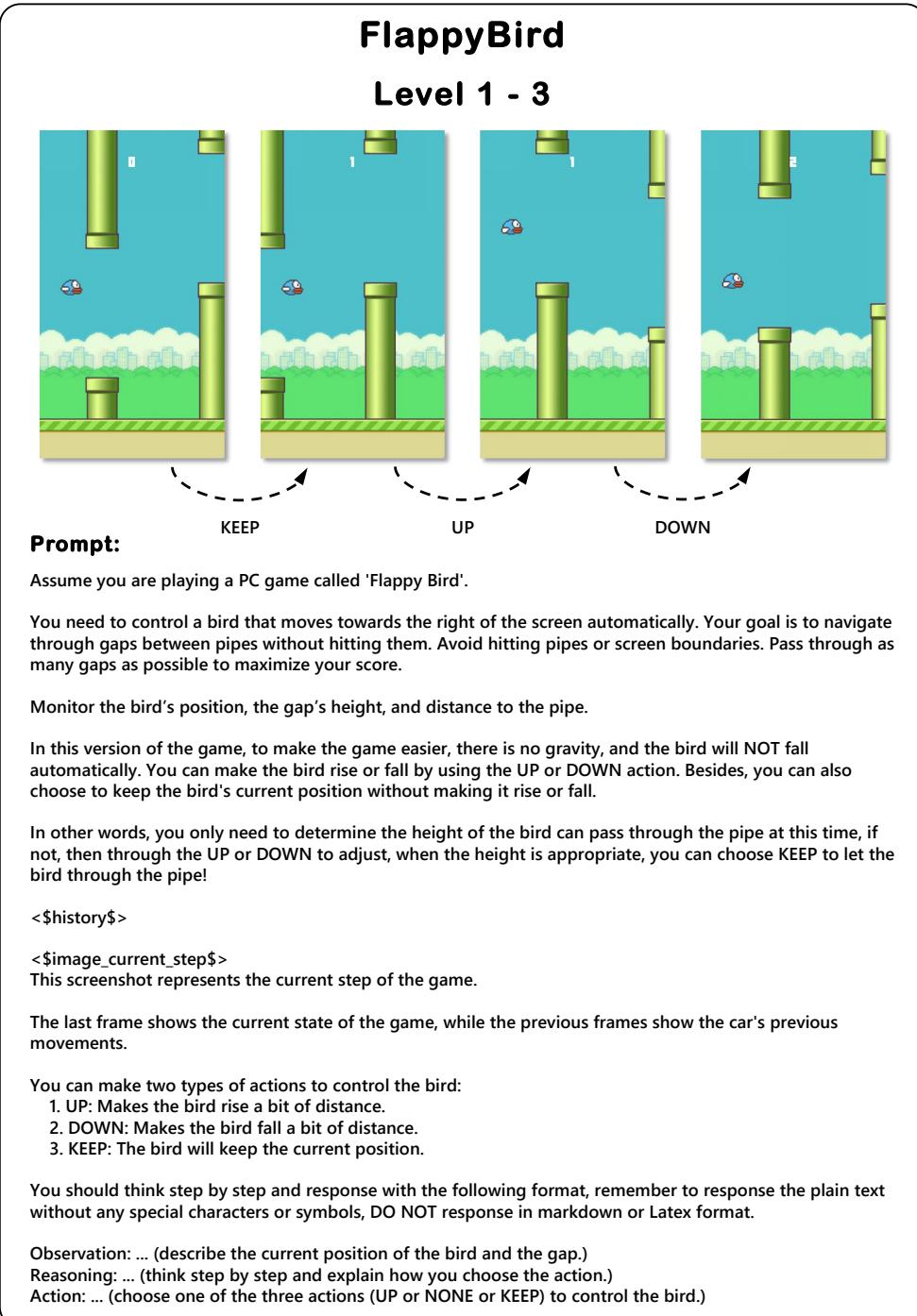


Figure 18: **FlappyBird Level 1-3: Level Design and Prompt Overview.** The images showcase the scene from Level 1, illustrating the level design and corresponding prompt. Levels are differentiated by the pipe gap width and the bird's forward speed. Elements in the same level will randomly change their initial positions while maintaining consistent relative difficulty.

1890

1891

1892

1893

1894

1895

1896

1897

1898

1899

1900

1901

1902

1903

1904

1905

1906

1907

1908

1909

1910

1911

1912

1913

1914

1915

1916

1917

1918

1919

1920

1921

1922

1923

1924

1925

1926

1927

1928

1929

1930

1931

1932

1933

1934

1935

1936

1937

1938

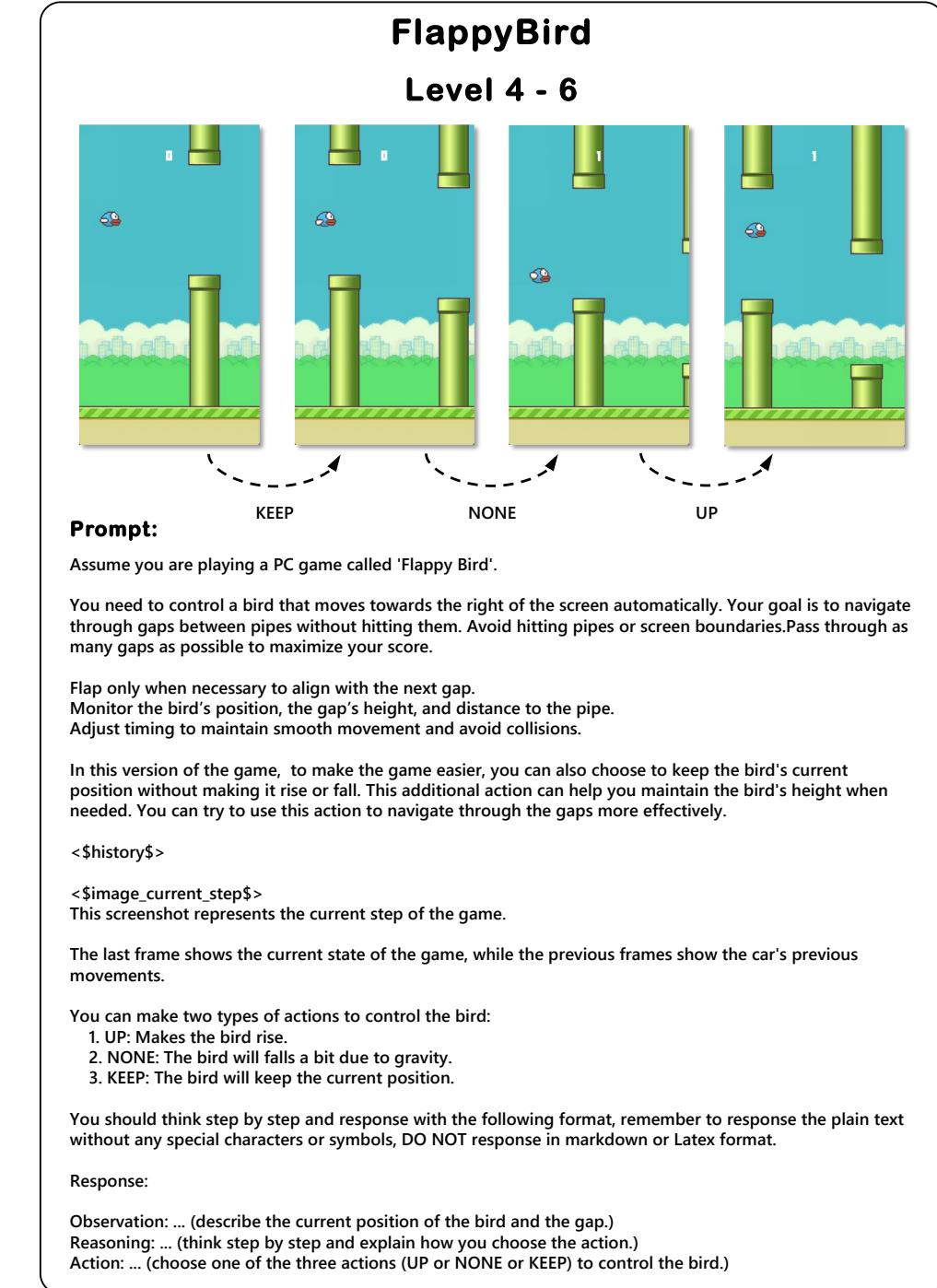


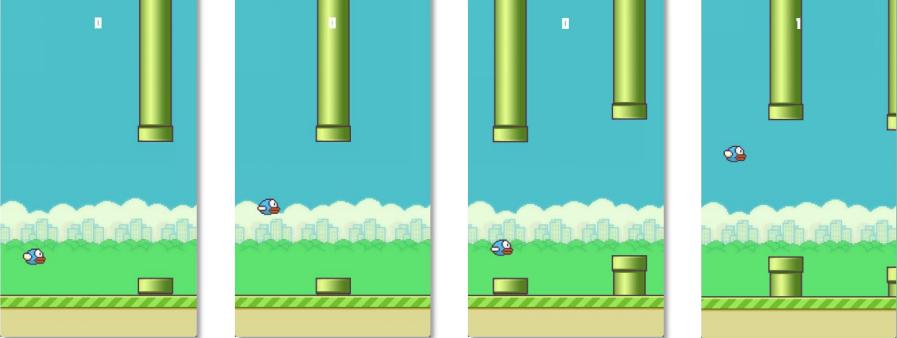
Figure 19: **FlappyBird Level 4-6: Level Design and Prompt Overview.** The images showcase the scene from Level 4, illustrating the level design and corresponding prompt. Levels are differentiated by the pipe gap width and the bird's forward speed. Elements in the same level will randomly change their initial positions while maintaining consistent relative difficulty.

1943

1944  
1945  
1946  
1947  
1948  
1949  
1950  
1951  
1952  
1953  
1954  
1955  
1956  
1957  
1958  
1959  
1960  
1961  
1962  
1963  
1964  
1965  
1966  
1967  
1968  
1969  
1970  
1971  
1972  
1973  
1974  
1975  
1976  
1977  
1978  
1979  
1980  
1981  
1982  
1983  
1984  
1985  
1986  
1987  
1988  
1989  
1990  
1991  
1992  
1993  
1994  
1995  
1996  
1997

## FlappyBird

### Level 7



**Prompt:**

Assume you are playing a PC game called 'Flappy Bird'.

You need to control a bird that moves towards the right of the screen automatically. Your goal is to navigate through gaps between pipes without hitting them. Avoid hitting pipes or screen boundaries. Pass through as many gaps as possible to maximize your score.

Flap only when necessary to align with the next gap.  
 Monitor the bird's position, the gap's height, and distance to the pipe.  
 Adjust timing to maintain smooth movement and avoid collisions.

<\$history\$>

<\$image\_current\_step\$>  
 This screenshot represents the current step of the game.

The last frame shows the current state of the game, while the previous frames show the car's previous movements.

You can make two types of actions to control the bird:

1. UP: Makes the bird rise.
2. NONE: The bird may fall a bit due to gravity.

You should think step by step and response with the following format, remember to response the plain text without any special characters or symbols, DO NOT response in markdown or Latex format.

**Response:**

Observation: ... (describe the current position of the bird and the gap.)  
 Reasoning: ... (think step by step and explain how you choose the action.)  
 Action: ... (choose one of the two actions (UP or NONE) to control the bird.)

Figure 20: **FlappyBird Level 7 (Standard Level): Level Design and Prompt Overview.** The images showcase the scene from Level 7, illustrating the level design and corresponding prompt. Elements in the same level will randomly change their initial positions while maintaining consistent relative difficulty. This is The standard level that matches the difficulty of the human game.

1998

1999

2000

2001

2002

2003

2004

2005

2006

2007

2008

2009

2010

2011

2012

2013

2014

2015

2016

2017

2018

2019

2020

2021

2022

2023

2024

2025

2026

2027

2028

2029

2030

2031

2032

2033

2034

2035

2036

2037

2038

2039

2040

2041

2042

2043

2044

2045

2046

2047

2048

2049

2050

2051

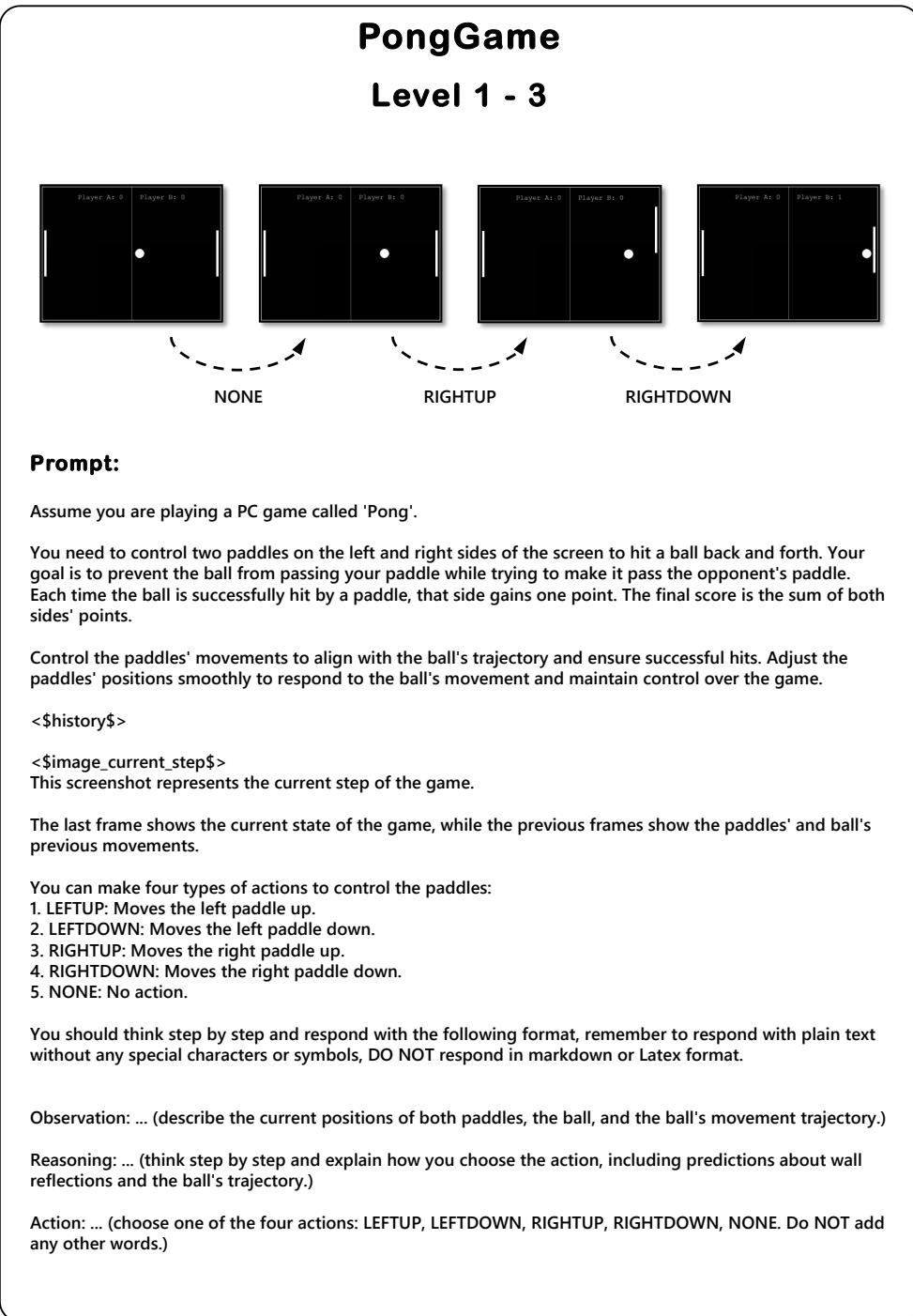


Figure 21: **PongGame Level 1-3: Level Design and Prompt Overview.** The images showcase the scene from Level 1, illustrating the level design and corresponding prompt. Levels are differentiated by the paddle width and the ping pong ball's speed. The ping pong ball in the same level will randomly change its initial position while maintaining consistent relative difficulty.

2052

2053

2054

2055

2056

2057

2058

2059

2060

2061

2062

2063

2064

2065

2066

2067

2068

2069

2070

2071

2072

2073

2074

2075

2076

2077

2078

2079

2080

2081

2082

2083

2084

2085

2086

2087

2088

2089

2090

2091

2092

2093

2094

2095

2096

2097

2098

2099

2100

2101

2102

2103

2104

2105

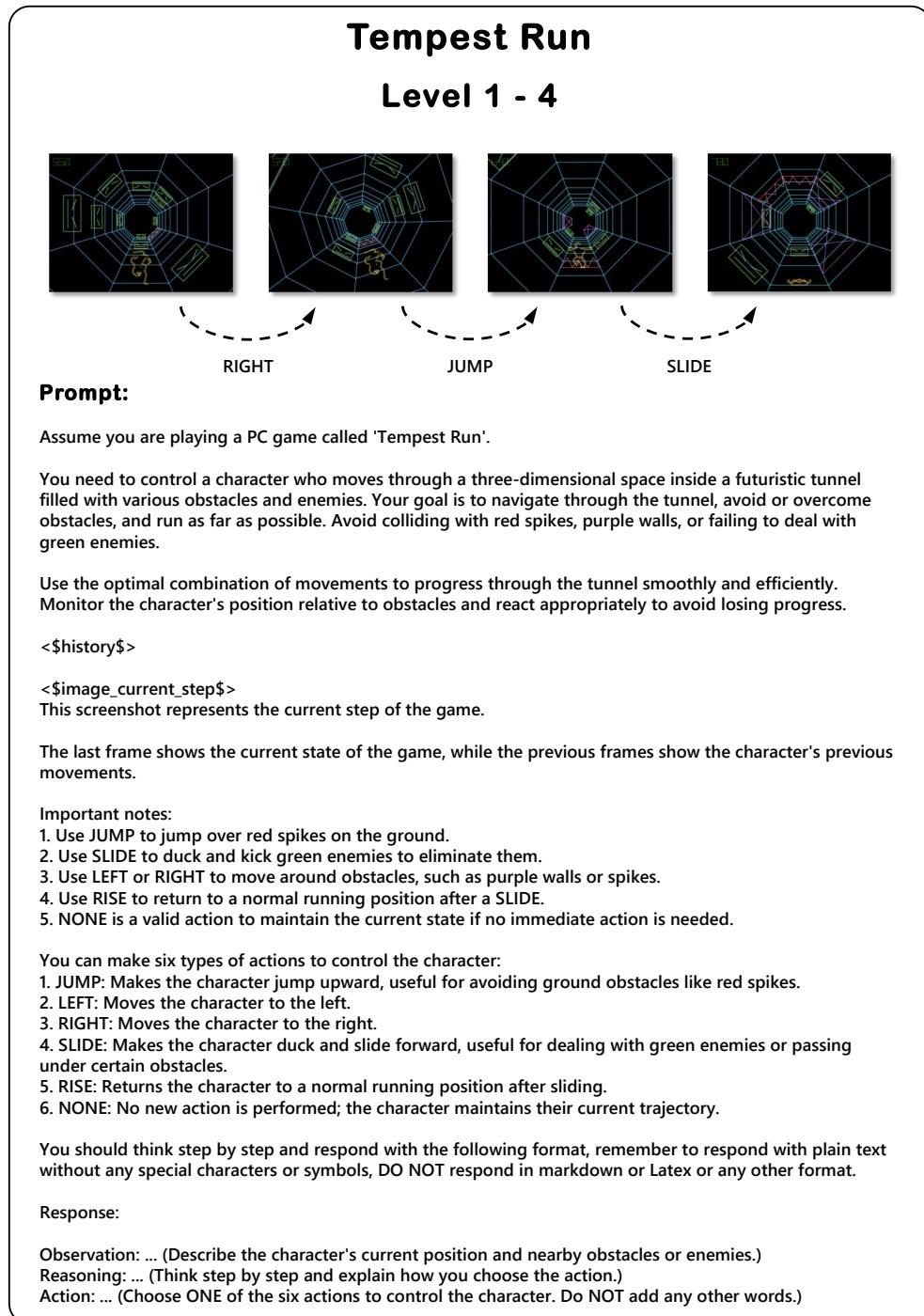


Figure 22: **Tempest Run Level 1-4: Level Design and Prompt Overview.** The images showcase the scene from Level 1, illustrating the level design and corresponding prompt. Levels are differentiated by the speed of barrier generation and the amount of visual information available. Elements in the same level will randomly change their initial positions while maintaining consistent relative difficulty.

## 2106 D ELO PERFORMANCE COMPARISON PROTOCOL DETAILS

2108 The core methodology for metrics evaluation in V-MAGE comprises two interconnected components: *performance comparison* and *statistical stabilization*.

### 2111 Performance Comparison Protocol.

2113 Each model begins with an initial Elo rating  $R_m^{(0)} = 1500$ , where  $m \in \mathcal{M}$  represents the set of  
 2114 candidate models. We conducted 100 evaluation rounds for each game level  $\ell$  where models were  
 2115 randomly paired in each round through a shuffle operation:

$$2117 \quad \mathcal{P}_t = \{(A_t, B_t) \mid A_t, B_t \stackrel{\text{rand}}{\sim} \mathcal{M}^\ell, A_t \neq B_t\} \quad (1)$$

2119 where  $A_t$  and  $B_t$  denote the paired models in round  $t$ .

2121 For paired models  $(A, B)$ , their game scores  $\text{score}_A$  and  $\text{score}_B$  are compared next. We first construct

$$2123 \quad f(m) = (\text{score}_m, \text{valid\_rate}_m) \quad (2)$$

2124 where  $\text{valid\_rate}_m$  represents the proportion of actions output by model  $m$  in this game round that  
 2125 conform to the formatting requirements. The outcome  $(S_A, S_B) \in \{(1, 0), (0, 1), (0.5, 0.5)\}$  is  
 2126 determined by:

$$2128 \quad (S_A, S_B) = \begin{cases} (1, 0) & \text{if } f(A) \succ f(B) \\ (0, 1) & \text{if } f(A) \prec f(B) \\ (0.5, 0.5) & \text{otherwise} \end{cases} \quad (3)$$

2132 The rating update mechanism follows the classical Elo formulation with enhanced stability controls.  
 2133 For models  $A$  and  $B$  with pre-match ratings  $R_A$  and  $R_B$ , their expected performance is calculated  
 2134 as:

$$2136 \quad E_A = \frac{1}{1 + 10^{(R_B - R_A)/400}}, \quad E_B = \frac{1}{1 + 10^{(R_A - R_B)/400}} \quad (4)$$

2138 where the denominator base 10 and scaling factor 400 establish a logarithmic relationship between  
 2139 rating differences and expected outcomes. The post-match ratings become:

$$2141 \quad R'_A = R_A + K(S_A - E_A) \quad (5)$$

$$2142 \quad R'_B = R_B + K(S_B - E_B)$$

2144 where  $K$  is a constant determining the sensitivity of the rating system (typically set to 32),

### 2146 Stabilization through Randomized Iteration.

2148 To ensure the robustness of rating updates, V-MAGE implements multi-pass stabilization protocol.  
 2149 All historical comparisons are aggregated into a win/loss pool:

$$2151 \quad \mathcal{W} = \bigcup_{g \in \mathcal{G}} \bigcup_{\ell \in \mathcal{L}_g} \bigcup_{t=1}^N (A_{g\ell t}, B_{g\ell t}, S_A^{g\ell t}, S_B^{g\ell t}) \quad (6)$$

2154 which undergoes  $T = 10^4$  independent shuffles. For each permutation  $\pi_i(\mathcal{W})$ , complete rating  
 2155 recalculation yields  $R_m^{(i)}$ . The final stabilized rating combines these trials:

$$2158 \quad \bar{R}_m = \frac{1}{T} \sum_{i=1}^T R_m^{(i)} \quad (7)$$

2160 E ABLATION STUDY ON PIPELINE SETTINGS  
21612162 E.1 IMPACT OF HISTORY SAMPLING CONFIGURATION  
2163

2164 we conducted supplementary experiments on the **Qwen2.5-VL-7B** and **Qwen2.5-VL-72B** models  
2165 using various history strategies (including increasing the number of history steps and altering sam-  
2166 pling methods). The results are presented in the tables below. These scores were calculated as a  
2167 percentage of model scores versus human performance in a manner similar to Figure 4.

2168 In the default setting of our main experiments, the history sampling configuration is one where  
2169 decisions are made using information from the most recent **3steps**, and the game screen is sampled  
2170 every **3frames**.

2172 Table 29: Performance comparison of different history strategies for Qwen2.5-VL 7B and 72B  
2173 models.

Game	Qwen2.5-VL-7B					Qwen2.5-VL-72B				
	3steps_2sample	3steps_5sample	<b>3steps</b>	5steps	8steps	3steps_2sample	3steps_5sample	<b>3steps</b>	5steps	8steps
<b>race</b>	11.20	11.20	12.60	11.20	12.40	30.00	29.00	29.60	32.60	33.60
<b>supermario</b>	20.10	22.10	22.60	22.80	21.20	34.50	33.90	42.10	36.40	39.80
<b>pong</b>	3.30	4.30	3.70	4.00	4.30	4.50	3.90	4.10	4.80	5.00
<b>flappybird</b>	6.70	11.20	3.40	2.10	5.00	17.70	13.10	8.10	13.30	13.60
<b>tempestrun</b>	18.80	17.80	21.10	18.80	17.80	22.00	21.10	24.80	22.70	23.70
<b>average</b>	<b>12.02</b>	<b>13.32</b>	<b>12.68</b>	<b>11.78</b>	<b>12.14</b>	<b>21.74</b>	<b>20.20</b>	<b>21.74</b>	<b>21.96</b>	<b>23.14</b>

2185 Table 30: Performance comparison of different frame sampling strategies for Qwen2.5-VL 7B and  
2186 72B models.

Game	Qwen2.5-VL-7B				Qwen2.5-VL-72B			
	<b>8frames</b>	<b>5frames</b>	<b>3frames</b>	<b>1frames</b>	<b>8frames</b>	<b>5frames</b>	<b>3frames</b>	<b>1frames</b>
<b>race</b>	13.80	11.60	12.60	10.40	19.60	27.20	29.60	26.00
<b>pong</b>	4.00	3.90	3.70	4.50	4.80	5.90	4.10	7.60
<b>flappybird</b>	3.40	5.70	3.40	7.60	10.80	14.00	8.10	14.10
<b>tempestrun</b>	19.00	19.30	21.10	16.50	23.40	18.50	24.80	26.60
<b>average</b>	<b>10.05</b>	<b>10.13</b>	<b>10.20</b>	<b>9.75</b>	<b>14.65</b>	<b>16.40</b>	<b>16.65</b>	<b>18.58</b>

2199 The experimental results show that simply increasing the length of the history window (e.g., from 3  
2200 to 8 steps) does not yield significant performance gains. This finding supports our core argument:  
2201 the bottleneck for current MLLMs lies **not in the quantity** of historical information they receive,  
2202 but in their ability to **understand and utilize** this dynamic visual information.

2204 Therefore, we chose a 3-frame history as our baseline configuration. This provides the necessary  
2205 temporal context while establishing a fair, simple, and effective standard for exposing the models'  
2206 core deficiencies, without confounding the evaluation with complex agent strategies.

2207 As mentioned in the main text, to investigate the impact of settings within the sampling strategies  
2208 on anchoring bias, we also conducted relevant experiments, with the results presented in Appendix  
2209 F.3.2.

2210 E.2 IMPACT OF INPUT RESOLUTION  
2211

2213 To systematically investigate the impact of input resolution on model performance, we conducted a  
new set of experiments, testing the **Qwen2.5-VL 7B** and **72B** models on four different resolutions.

The resolutions from 120 to 480 refer to images with heights of 120 to 480 pixels, respectively, with the width scaled according to the original aspect ratio. We selected these four resolutions to cover different levels of visual detail, from low to high. In the `default` setting of our main experiments, the model’s input resolution was **360** pixels height. The scores were calculated as a percentage of model scores versus human performance in a manner similar to Figure 4. The results are presented in Table 31.

Table 31: Performance comparison of Qwen2.5-VL 7B and 72B models across different input resolutions (height in pixels). Scores are percentages relative to human performance.

Game	Qwen2.5-VL 7B				Qwen2.5-VL 72B			
	120 (7B)	240 (7B)	<b>360</b> (7B)	480 (7B)	120 (72B)	240 (72B)	<b>360</b> (72B)	480 (72B)
<b>race</b>	9.80	10.60	12.60	11.00	15.80	23.60	29.60	28.20
<b>supermario</b>	17.50	21.70	22.60	17.20	38.90	44.90	42.10	47.90
<b>pong</b>	4.00	3.90	3.70	3.60	3.50	3.50	4.10	3.60
<b>flappybird</b>	4.40	5.40	3.40	9.80	7.70	12.70	8.10	12.00
<b>tempestrun</b>	19.60	19.10	21.10	18.50	19.60	24.30	24.80	22.50
<b>average</b>	<i>11.06</i>	<i>12.14</i>	<b>12.68</b>	<i>12.02</i>	<i>17.10</i>	<i>21.80</i>	<i>21.74</i>	<b>22.84</b>

This data reveals a nuanced relationship: for the more capable 72B model, the overall performance trend improves with higher resolution, peaking at 480px. This suggests it can benefit from the finer details in higher-resolution images. However, for the smaller 7B model, performance peaks at our default setting of 360px and declines at the higher 480px resolution.

This indicates that the relationship between model performance and input resolution is **not simply linear**. For less capable models, excessive resolution might introduce ‘noise’ that they struggle to filter effectively, thereby interfering with their decision-making process.

2268 **F ADDITIONAL EXPERIMENTAL DETAILS**  
22692270 **F.1 UNIT TESTS FOR CORE VISUAL ABILITIES EXPERIMENT**  
22712272 Table 32: Basic visual capabilities and their corresponding simple game levels.  
2273

2274 Visual Abilities	2275 Game	2276 Levels
2276 Tracking	2277 Pong	2278 1, 2, 3
2277 Positioning	2278 Race	2279 1, 1_no_history
2278 Visual Grounding	2279 TempestRun	2280 1
2279 Timing	2280 FlappyBird	2281 1, 2, 3

2281 The unit testing framework conducts a systematic assessment of fundamental visual capabilities by  
2282 drawing from the comprehensive V-MAGE benchmark. In each carefully designed level of a game, a  
2283 random baseline score is first determined by averaging scores from random actions. Following this,  
2284 the performance of each evaluated model on the said level is quantified by calculating the percentage  
2285 of rounds where the model’s score outperforms this established random baseline. The specific game  
2286 levels used for assessing each ability are listed in Table 32.

2287 As illustrated in Figure 5, model performances across representative levels for four fundamental vi-  
2288 sual competencies reveal critical insights: In tracking tasks requiring cross-frame analysis, nearly all  
2289 models underperform random baselines. This indicates that while current models achieve reasoning  
2290 through caption-based approaches in single-frame tasks, they struggle to extract discriminative fea-  
2291 tures in multi-frame scenarios requiring fine-grained spatiotemporal comparisons. The quantitative  
2292 results for each model across the four core visual abilities are presented in Table 33.

2293 Table 33: Performance of MLLMs on Core Visual Ability Unit Tests (% Exceeding Random Base-  
2294 line)  
2295

2297 Model	2298 Positioning	2299 Tracking	2300 Visual Grounding	2301 Timing
2300 Qwen2VL 7B	0.50	0.27	0.56	0.36
2301 Qwen2VL 72B	0.76	0.26	0.70	0.43
2302 Qwen2.5VL 72B	0.88	0.25	0.68	<b>0.51</b>
2303 InternVL2.5 78B	0.82	<b>0.33</b>	0.66	0.49
2304 InternVL2.5 8B	0.60	0.28	0.55	0.39
2305 Gemini-2.0-Flash	0.68	0.32	<b>0.70</b>	<b>0.51</b>
2306 GPT4o	<b>0.98</b>	0.29	0.66	0.58

2309 It is important to interpret the results of these unit tests within their intended scope. Designed to  
2310 assess fundamental visual competencies, these tests utilize a random baseline score as the primary  
2311 reference point. While a model significantly outperforming this random baseline indicates a degree  
2312 of relevant reasoning ability in that specific task dimension, it does not necessarily imply a high level  
2313 of overall competence. The random baseline represents minimal performance, and even achieving  
2314 scores far exceeding it on these foundational tests serves primarily to diagnose basic capabilities  
2315 rather than validate advanced mastery required for complex gameplay.

2317 **F.2 PERCEPTUAL SKIPPING EXPERIMENT**  
2318

2319 To further investigate the interplay between visual perception and reasoning, we conducted supple-  
2320 mentary experiments where textual descriptions of the game state were provided, effectively bypassing  
2321 the visual perception module (see Table 34 for detailed results on Flappy Bird Level 3 and Pong  
Level 2).

Table 34: Model performance on simple levels with and without textual state information.

Model	Flappy Bird		Pong	
	w/o Text	w/ Text	w/o Text	w/ Text
Qwen2.5VL 7B	0.8	0.35	0.19	0.25
InternVL2.5 8B	0.31	0.76	0.19	0.31
Qwen2.5VL 72B	0.35	2.17	0.21	1.19
InternVL2.5 78B	0.59	2.39	0.16	0.52
GPT4o	0.57	4.55	0.20	3.89
Gemini-2.0-Flash	0.42	4.89	0.32	>10
random	0.52		0.18	
human	> 10		> 10	

The results indicate that alleviating the perceptual challenge generally improves performance, particularly for larger models like GPT-4o and the 72B/78B parameter models, supporting the hypothesis that visual perception is a significant bottleneck. However, even with this intervention, model scores remained substantially lower than the human baseline ( $>10$ ), underscoring the presence of critical reasoning and planning deficiencies beyond visual perception, as discussed earlier.

Notably, the performance gains from text input were more pronounced for larger models, suggesting their enhanced capacity to leverage structured textual information for reasoning, whereas smaller models exhibited less consistent benefits or even performance degradation in some cases. This finding further highlights that while perception is a challenge, fundamental reasoning limitations persist across models and are not fully overcome even when provided with simplified, textual state representations.

### F.3 ANCHORING BIAS EXPERIMENTS

#### F.3.1 DETAILS AND EXAMPLES

Table 35: Average number of rounds for each model to generate different responses.

Model	Race	FlappyBird	Pong	TempestRun	Avg.
Qwen2VL 7B	4.3	25.9	13.7	7.3	12.8
Qwen2.5VL 72B	2.3	19.3	2.6	5.3	7.4
InternVL2.5 8B	2.0	6.9	6.7	8.0	5.9
InternVL2.5 78B	6.8	16.0	2.0	3.0	7.0
GPT4o	<b>1.0</b>	<b>1.6</b>	<b>1.0</b>	<b>1.0</b>	<b>1.1</b>
<b>PCC <math>r</math></b> (Avg. Rounds vs. ELO)	-0.63	-0.86	-0.88	-0.64	-0.75

Due to the possibility of models receiving identical visual inputs over multiple rounds in Super Mario (e.g., being stuck in a corner), we conducted a statistical analysis using lots of rounds of responses from each model across the other four games. This was done by iterating through the recorded interactions for each level and measuring the number of sequential rounds where the model’s output response remained unchanged. The average of these durations across interactions within a game provides the metric presented in Table 35.

The results indicate that GPT-4o updates its responses more actively and frequently when the visuals change, while other models do so less frequently. This may suggest that GPT-4o is more sensitive to subtle visual updates, enabling it to make timely inferences and more accurately track game progress.

In the example of Figure 23, GPT-4o’s response during the final interaction is as follows:



Figure 23: In the prior reasoning processes, GPT-4o accurately identified the car’s orientation relative to the trophy’s position. However, in the final reasoning instance, due to anchoring bias, the model misinterpreted previous historical information and incorrectly assumed that the car’s orientation was already directed toward the trophy. Consequently, despite correctly determining the direction, the model executed an erroneous action.

Description: The car is currently facing diagonally towards the **bottom-right** of the screen. The trophy is positioned near the car, slightly to the **right**. There are no visible obstacles between the car and the trophy, indicating a clear path. The car has been moving consistently forward in the previous steps, maintaining its alignment towards the trophy.

Reasoning: Given the car’s orientation towards the trophy and the absence of obstacles, the car’s current position and direction remain favorable for reaching the trophy. Continuing to move forward is the most efficient strategy to maintain the direct path to the trophy. The car’s current trajectory should allow it to reach the trophy without any adjustments.

Action: FORWARD

### F.3.2 IMPACT OF FRAME SAMPLING AND DECISION FREQUENCY

To further quantify the relationship between perceptual sensitivity and anchoring bias, we have conducted a deeper quantitative exploration of the relationship between anchoring bias and model performance. To more objectively measure a model’s reaction to dynamic changes in the game world, we introduced a new metric: **Average Response Game Frames** (abbreviated as ‘avg frames’ in the results). This metric is calculated by:

$$(\text{sampling interval}) \times (\text{average rounds to generate different responses})$$

and represents **how many game frames, on average, have elapsed before a model makes a substantive change in its reasoning**.

We performed a series of experiments with different sampling strategies, first testing the Qwen2.5VL-72B model. The results are as follows:

Table 36: Performance of Qwen2.5VL-72B under different frame sampling strategies. The top section shows response frequency metrics, while the bottom shows game scores.

Game	8frames	5frames	3frames(default)	1frames
<i>Average Response Game Frames</i>				
<b>race</b>	5	1.8	2.3	10.8
<b>pong</b>	1.7	1.7	2.6	19.7
<b>flappybird</b>	1.5	2.3	19.3	64.2
<b>tempestrun</b>	1.8	14.2	5.3	23.2
<b>avg request</b>	2.5	5	7.4	29.5
<b>avg frames</b>	<b>20</b>	<b>25</b>	<b>22.2</b>	<b>29.5</b>
<i>Game Score</i>				
<b>race</b>	19.60	27.20	29.60	26.00
<b>pong</b>	4.80	5.90	4.10	7.60
<b>flappybird</b>	10.80	14.00	8.10	11.90
<b>tempestrun</b>	23.40	18.50	24.80	26.60
<b>avg score</b>	<b>14.65</b> <span style="color:red">↓2.00</span>	<b>16.40</b> <span style="color:red">↓0.25</span>	<b>16.65</b> <span style="color:green">↑0.00</span>	<b>18.03</b> <span style="color:green">↑1.38</span>

**Stable ‘visual reaction threshold’ in strong models:** From the avg frames metric, the Qwen2.5VL-72B model demonstrates remarkable consistency across different sampling strategies, with its av-

verage response time stabilizing within a narrow range of **20-30** game frames. This suggests that the model possesses a relatively constant intrinsic reaction threshold, where a certain amount of accumulated visual change triggers a shift in its reasoning.

**Regarding task score (avg score):** The 72B model’s performance clearly improves as the sampling interval decreases, with the highest score achieved at the highest decision frequency (1-frame interval). Under such high-frequency decision-making, the model can capture crucial task timings with the highest precision. As the decision frequency decreases, the opportunities for the model to take appropriate action at the right moment are reduced, thus leading to a drop in performance.

Next, we compared the **Qwen2.5VL-7B** and **72B** models under the same sampling strategies:

Table 37: Comparison of Qwen2.5VL-7B and 72B models across sampling strategies. The 72B model shows a consistently lower reaction threshold (avg frames) and higher scores.

Game	8frames	5frames	3frames	1frames
<b>Qwen2.5VL-7B</b>				
avg request	13.5	12.5	34.7	97.8
avg frames	<b>108.0</b>	<b>62.5</b>	<b>104.1</b>	<b>97.8</b>
avg score	<b>10.1</b>	<b>10.1</b>	<b>10.2</b>	<b>9.8</b>
<b>Qwen2.5VL-72B</b>				
avg request	2.5	5	7.4	29.5
avg frames	<b>20.0</b> $\downarrow$ <b>88.0</b>	<b>25.0</b> $\downarrow$ <b>37.5</b>	<b>22.2</b> $\downarrow$ <b>81.9</b>	<b>29.5</b> $\downarrow$ <b>68.3</b>
avg score	<b>14.7</b> $\uparrow$ <b>4.6</b>	<b>16.4</b> $\uparrow$ <b>6.3</b>	<b>16.7</b> $\uparrow$ <b>6.5</b>	<b>18.6</b> $\uparrow$ <b>8.8</b>

The correlation between the "visual reaction threshold" and "task score" remains clear when comparing across models. The 72B model exhibits a **lower** Average Response Game Frames (indicating higher perceptual sensitivity) and a **higher** task score, while the 7B model shows the opposite. This is consistent with the conclusions about anchoring bias: a more powerful model possesses greater sensitivity to dynamic visual perception, which forms the basis for more accurate decision-making in interactive tasks.

#### F.4 ANALYSIS OF GPT4O ERRORS IN V-MAGE

We have collected **2,351** prompt-response pairs generated by GPT-4o while completing all levels for 1 to 5 rounds. From these, **494** examples were randomly and uniformly sampled for manual error annotation. The frequency of occurrence for various error types is presented in Table 38.

Table 38: Error count by error type and game environment

Error Type	FlappyBird	Pong	Race	SuperMario	TempestRun
no error	30	18	54	88	21
perception error	80	26	26	47	42
direction error	2	19	13	16	8
recognition error	1	0	0	0	5
perception incomplete	3	0	8	10	10
reasoning error	24	4	9	10	6
history misinterpretation	21	0	1	6	2
action inappropriate	0	0	5	0	0
inconsistency	0	0	14	1	0
instruction following	0	0	2	0	0

The definitions of each error type are presented as follows:

- **no error:** There is no error in the response.
- **perception error:** Description misinterpreted elements.

2484     • **direction error:** A type of perception error. Confused directions (e.g., LEFT/RIGHT)  
2485     • **recognition error:** A type of perception error. Failed to identify key objects/elements.  
2486     • **perception incomplete:** Description missed important elements in the scene.  
2487     • **reasoning error:** Flawed logic in the reasoning section for the chosen action.  
2488     • **history misinterpretation:** A type of reasoning error. Misunderstood the game history.  
2489     • **action inappropriate:** A type of reasoning error. The chosen Action is clearly wrong given  
2490       the Observation/Reasoning.  
2491     • **inconsistency:** Inconsistent action plans in multiple response processes.  
2492     • **instruction following:** Failed to follow instructions in the prompt.  
2493  
2494  
2495  
2496  
2497  
2498  
2499  
2500  
2501  
2502  
2503  
2504  
2505  
2506  
2507  
2508  
2509  
2510  
2511  
2512  
2513  
2514  
2515  
2516  
2517  
2518  
2519  
2520  
2521  
2522  
2523  
2524  
2525  
2526  
2527  
2528  
2529  
2530  
2531  
2532  
2533  
2534  
2535  
2536  
2537

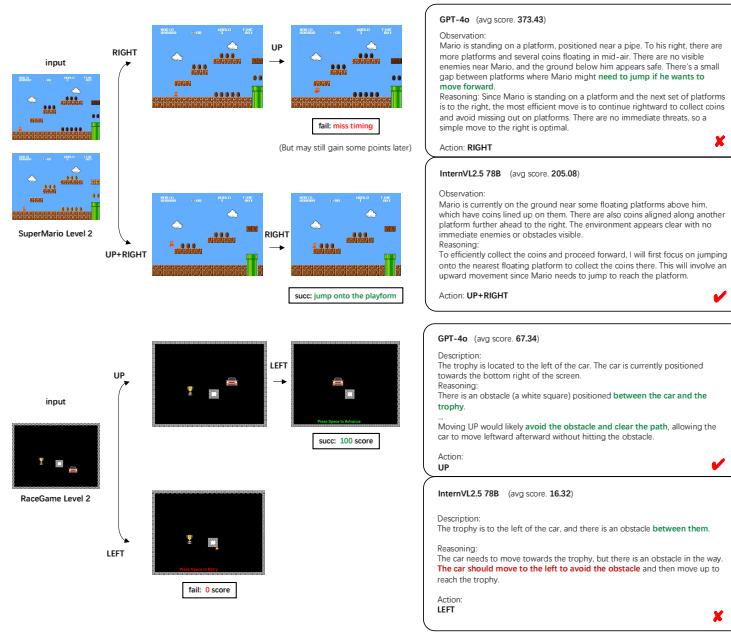
2538 G CASE STUDY  
2539  
2540  
2541  
2542  
2543  
2544  
2545  
2546  
2547  
2548  
2549  
2550  
2551  
2552  
2553  
2554  
2555  
2556  
2557  
2558  
2559  
2560  
2561  
2562  
2563  
2564  
2565  
2566  
2567  
2568  
2569  
2570  
2571  
2572  
2573  
2574  
2575  
2576  
2577  
2578  
2579  
2580  
2581  
2582  
2583  
2584  
2585  
2586  
2587  
2588  
2589  
2590  
2591

Figure 24: case studies group 1.

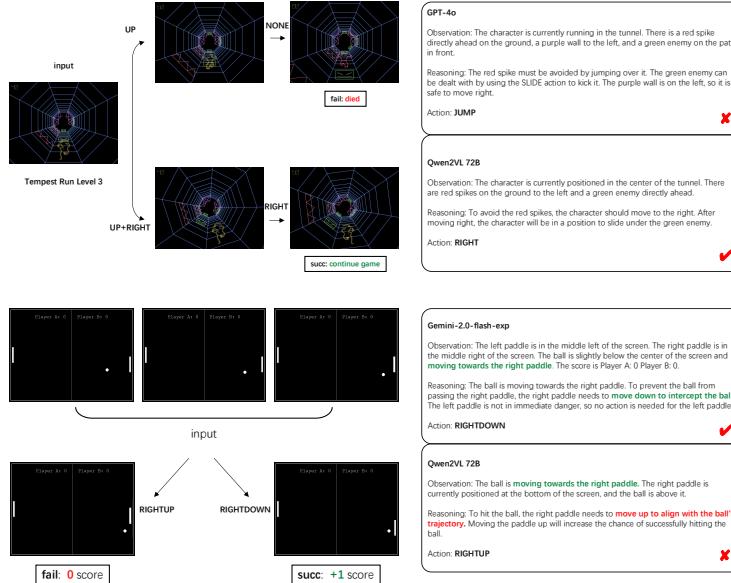


Figure 25: case studies group 2.

2592  
2593  
2594  
2595

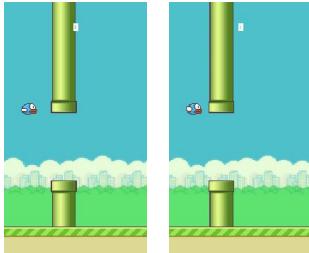
2596 **History Misinterpretation**



2597  
2598  
2599  
2600  
2601  
2602  
2603  
2604  
2605  
2606 Observation: The bird is positioned slightly below the middle of the screen. The gap in the pipes is directly ahead and is approximately at the same vertical level as the bird's current position.  
2607  
2608  
2609 Reasoning: The bird has maintained its position over the past several steps, which indicates it has been consistently aligned with the gap. As there has been no change in the relative positions of the bird and the gap, and since the game continues without any issues, there is no need to make any vertical adjustments. Keeping the current altitude will allow the bird to safely pass through the gap.  
2610  
2611  
2612  
2613  
2614 Action: KEEP  
2615  
2616  
2617  
2618  
2619  
2620  
2621  
2622  
2623  
2624  
2625  
2626  
2627  
2628  
2629  
2630  
2631  
2632  
2633  
2634  
2635  
2636  
2637  
2638  
2639  
2640  
2641  
2642  
2643  
2644  
2645

Observation: The bird is positioned slightly below the middle of the screen. The gap in the pipes is directly ahead and is approximately at the same vertical level as the bird's current position.  
Reasoning: In the previous steps, the bird was well-aligned with the gap, and **the choice to keep the bird's position allowed it to pass safely through the pipes**. Since the bird's position relative to the gap has remained consistent and aligned over these steps, maintaining the current altitude should continue to be a successful strategy. There is no indication of any necessary adjustment as the gap is still aligned with the bird's current trajectory.  
Action: KEEP

**Perception error**



Observation: The bird is currently positioned slightly below the center of the gap in the upcoming pipe. The alignment seems appropriate for passing through the gap without hitting the pipes.  
Reasoning: In the previous steps, the bird's position was maintained because it was already at an **optimal height to pass through the gap**. The current frame shows that the bird is still **well-aligned with the center of the gap**. There has been no indication of any change in the gap's position or size that would necessitate an adjustment. Since there is no gravity affecting the bird, it will maintain its current height, ensuring it passes through without any collision risk.  
Action: KEEP

**Incomplete perception**



Description: The trophy is currently located to the left of the car.  
Reasoning: The car is positioned to the right of the trophy, and there are no visible obstacles directly between the car and the trophy.  
Action: LEFT  
(No obstacle locations observed)

Figure 26: Error analysis in GPT4o cases.

2646

2647

2648

2649

2650

2651

2652

2653

2654

2655

2656

2657

2658

2659

2660

2661

2662

2663

2664

2665

2666

2667

2668

2669

2670

2671

2672

2673

2674

2675

2676

2677

2678

2679

2680

2681

**Inconsistency**

Description: The trophy is currently positioned to the right of the car.

Reasoning: There is an obstacle, which appears to be a wall, located between the car and the trophy. The obstacle is directly in the path if the car attempts to move **RIGHT** towards the trophy. Therefore, the car should avoid moving directly towards the obstacle and instead find an alternative path.

Action: **UP**



Description: The trophy is located to the right and slightly below the car.

Reasoning: There is an obstacle (a square block) directly between the car and the trophy. Moving directly to the right would result in a collision with the obstacle. Therefore, the car must first navigate around the obstacle by moving **DOWNWARD**, then move to the right to approach the trophy safely.

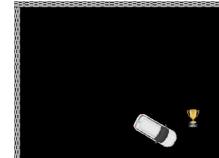
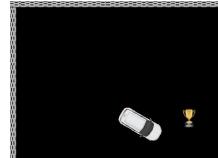
Action: **DOWN**

**Action Inappropriate**

Description: The trophy is to the right of the car.

Reasoning: The trophy is directly to the right of the car. There is a vertical obstacle (a wall) between the car and the trophy. The car needs to **avoid colliding with the obstacle** to reach the trophy.

Action: **RIGHT**

**Perception error**

Description: The car is currently facing diagonally towards the bottom-right of the screen. The trophy is positioned in front of the car, slightly to the right. There are no visible obstacles between the car and the trophy, indicating a clear path. The car has been moving consistently forward in the previous steps, maintaining its alignment towards the trophy.

Reasoning: Given the car's orientation towards the trophy and the absence of obstacles, the car's current position and direction remain favorable for reaching the trophy. Continuing to move forward is the most efficient strategy to maintain the direct path to the trophy. The car's current trajectory should allow it to reach the trophy without any adjustments.

Action: **FORWARD**

Figure 27: Error analysis in GPT4o cases.

2700

2701

2702

2703

2704

2705

2706

2707

2708

2709

2710

2711

2712

2713

2714

2715

2716

2717

2718

2719

2720

2721

2722

2723

2724

2725

2726

2727

2728

2729

2730

2731

2732

2733

2734

2735

2736

2737

2738

2739

2740

2741

2742

2743

2744

2745

2746

2747

2748

2749

2750

2751

2752

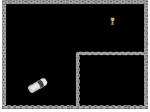
2753

**Direction Error**

Description: The trophy is located to the top-left of the car. The car is currently facing diagonally upwards to the right side of the screen. There is an obstacle (a wall) directly to the left and slightly above the car, which separates it from the trophy. To reach the trophy, the car must navigate around this wall without colliding with any obstacles. The car is currently not moving but is positioned at an angle facing towards the top-right corner of the screen.

Reasoning: Since the car is facing diagonally upwards to the right, the immediate path to the trophy is obstructed by the wall directly to the left and slightly above. The car needs to move upwards and then turn left to navigate around the wall towards the trophy. From the driver's perspective, the car needs to first orient itself to face upwards more directly. The immediate action should be to adjust the car's direction by turning LEFT, which will help the car face upwards, allowing it to maneuver around the wall and approach the trophy.

Action: LEFT

**Action Inappropriate**

Description: The trophy is positioned in the upper-right corner of the frame, relative to the car's current position. The car is facing directly towards the right side of the frame. There is a vertical wall directly in front of the car, which extends upwards and prevents the car from moving forward directly towards the trophy. This wall creates a barrier between the car and the trophy, so the car cannot reach the trophy by moving straight ahead.

Reasoning: The car is positioned in such a way that if it moves forward, it will collide with the wall, resulting in game over. Since the car is facing right and the path to the trophy is blocked by the wall, the car needs to navigate around the obstacle to reach the trophy. The car must first turn right to begin moving around the wall. By turning right, the car can begin to reposition itself and align with a path that allows it to bypass the wall and proceed towards the trophy.

Action: RIGHT

**Instruction Following**

(In the prompt, we ask the model to output only one word in the Action, with no other content attached)



Description: The trophy is located to the right of the car.

Reasoning: There is a vertical obstacle (a white line) directly between the car and the trophy. The car cannot move directly to the right without colliding with this obstacle.

Action: UP (to move the car up and around the obstacle)

Figure 28: Error analysis in GPT4o cases.

2754

2755

2756

2757

2758

2759

2760

2761

2762

2763

2764

2765

2766

2767

2768

2769

2770

2771

2772

2773

2774

2775

2776

2777

2778

2779

2780

2781

2782

2783

2784

2785

2786

2787

2788

2789

2790

2791

2792

2793

2794

2795

2796

2797

2798

2799

2800

2801

2802

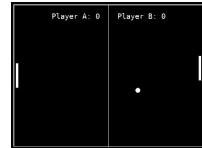
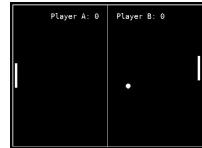
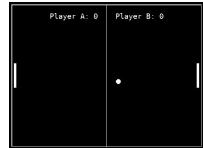
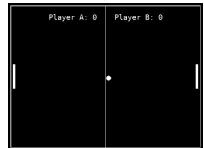
2803

2804

2805

2806

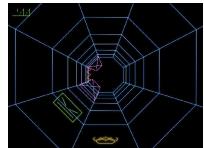
2807

**Direction Error**

Observation: The left paddle is positioned in the middle of the left side. The right paddle is slightly above the center on the right side. The ball is moving towards the right side of the screen and is slightly above the center line, continuing its trajectory towards the right paddle.

Reasoning: The ball is heading towards the right paddle and is slightly **above** the center line. Given the ball's trajectory, it appears to be on a direct path towards the **upper part of the right paddle**. The right paddle is already positioned slightly above the center, which aligns well with the ball's current path. To ensure the ball is intercepted effectively, the right paddle should be moved up slightly to account for any upward drift in the ball's trajectory. The left paddle does not need any adjustment since the ball is moving away from it.

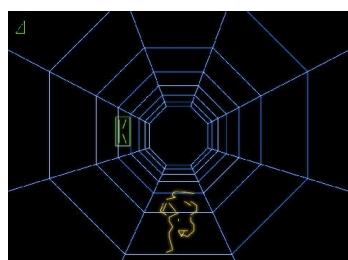
Action: **RIGHTUP**

**Direction Error**

Observation: The character is currently in a normal running position at the bottom center of the tunnel. There is a purple wall approaching in the center of the tunnel.

Reasoning: The character is in a normal running position, and there is a purple wall **directly ahead**. To avoid the purple wall, the character should move around it by using the **LEFT** action to shift to the left. This will help the character to safely navigate through the tunnel without colliding with the wall.

Action: **LEFT**

**Action Inappropriate**

Observation: The character is currently running in the center of the tunnel, and there is a green enemy approaching in the **left section** of the tunnel ahead.

Reasoning: The green enemy is approaching on the left side, and to eliminate it, the character **needs to perform a slide**. Sliding will allow the character to duck and kick the green enemy. Since the character is currently running in the center, sliding will position them correctly to deal with the green enemy.

Action: **SLIDE**

Figure 29: Error analysis in GPT4o cases.

---

**2808 H MISCELLANEOUS MATERIAL**  
2809**2810 H.1 LLM USAGE STATEMENT**  
2811

2812 Our research methodology centered on the evaluation of various MLLMs. Models such as GPT-4o  
2813 served as the subjects within our V-MAGE pipeline, generating the outputs that form the basis of  
2814 our analysis and conclusions on MLLM performance. The role of these MLLMs was strictly limited  
2815 to this evaluation phase. The conceptualization and implementation of the V-MAGE framework and  
2816 its software were carried out entirely by the authors.

2817 For the manuscript preparation, we employed LLMs for the sole purpose of improving grammar  
2818 and polishing the language. All scientific contributions, including the research ideas, experimental  
2819 design, and results interpretation, originate exclusively from the authors.

**2820 H.2 IMPACTS STATEMENT**  
2821

2822 This research contributes to the field of multimodal models by providing a novel and challenging  
2823 benchmark for evaluating vision-centric capabilities in dynamic environments. The primary positive  
2824 impact is facilitating the diagnosis of limitations in current MLLMs and guiding future research  
2825 towards developing more capable, robust, and potentially safer AI systems for real-world interaction.  
2826 As our work focuses on foundational evaluation in simulated environments and does not involve the  
2827 deployment of high-risk models or the collection of sensitive personal data, the potential for negative  
2828 societal impacts is considered minimal and indirect at this stage. We believe that developing better  
2829 evaluation tools is a crucial step towards building more reliable and trustworthy AI.

2830

2831

2832

2833

2834

2835

2836

2837

2838

2839

2840

2841

2842

2843

2844

2845

2846

2847

2848

2849

2850

2851

2852

2853

2854

2855

2856

2857

2858

2859

2860

2861