

The Percept-V Challenge: Can Multimodal LLMs Crack Simple Perception Problems?

Samrajnee Ghosh¹, Ashish Goswami², Naman Agarwal¹, Hemanshu Garg¹,
Chinmay Mittal¹, Mausam^{1,2}, Parag Singla^{1,2}

¹Computer Science and Engineering, ²Yardi School of Artificial Intelligence
Indian Institute of Technology, Delhi

Abstract

Cognitive science research treats visual perception, the ability to understand and make sense of a visual input, as one of the early developmental signs of intelligence. Its TVPS-4 framework categorizes and tests human perception into seven skills such as visual discrimination, and form constancy. Do Multimodal Large Language Models (MLLMs) match up to humans in basic perception? Even though many benchmarks evaluate MLLMs on advanced reasoning and knowledge skills, there is limited research that focuses evaluation on simple perception. In response, we introduce Percept-V, a dataset containing 6000 program-generated uncontaminated images divided into 30 domains, where each domain tests one or more TVPS-4 skills. Our focus is on perception, so we make our domains quite simple and the reasoning and knowledge required for solving them are minimal. Since modern-day MLLMs can solve much more complex tasks, our a-priori expectation is that they will solve these domains very easily. Contrary to our belief, our experiments show a weak performance of SoTA proprietary and open-source MLLMs compared to very high human performance on Percept-V. We find that as the number of objects in the image increases, performance goes down rather fast. Our experiments also identify the perception skills that are considerably harder for all models. Fine-tuning an open-source MLLM shows considerable gains in performance, though the gains only marginally carry over to other related datasets, pointing to limitation in generalization abilities of the learned representations. Code and data are available at <https://dair-iitd.github.io/Percept-V/>

1 Introduction

Multimodal Large Language Models (MLLMs) have achieved strong performance on complex tasks, prompting some claims that they are approaching human-level intelligence (Zhou et al., 2025; King, 2023) or even MENSEA-level capabilities (Schregel, 2020). However, existing visual benchmarks evaluate image understanding jointly with reasoning and domain-specific knowledge in areas such as engineering, medicine, and art (Thanh, 2023; Yue et al., 2023). This integrated approach makes it challenging to isolate MLLMs’ visual perception capabilities from their reasoning abilities and domain-specific knowledge.

Visual perception is a foundational cognitive ability that precedes complex reasoning and knowledge-based skills in humans (Rabindran & Madanagopal, 2020). Recent research has raised concerns about MLLMs’ perceptual capabilities (Zhang et al., 2024b; Fu et al., 2024), highlighting the need for a dedicated perception benchmark. To address this, we introduce Percept-V, a dataset of 6000 programmatically generated, uncontaminated image-based problems spanning 30 domains. These tasks primarily focus on visual understanding while requiring minimal reasoning and no specialized knowledge (see Figure 1). Our dataset is grounded in cognitive science, adopting the Test of Visual Perceptual Skills (TVPS-4) framework (Saboo, 1996), which categorizes visual perception into 7 skills, including visual discrimination, and spatial relationships (Section 3). Each domain evaluates 1–3 of these skills using a standardized question prompt describing the task. Instances involve a basic

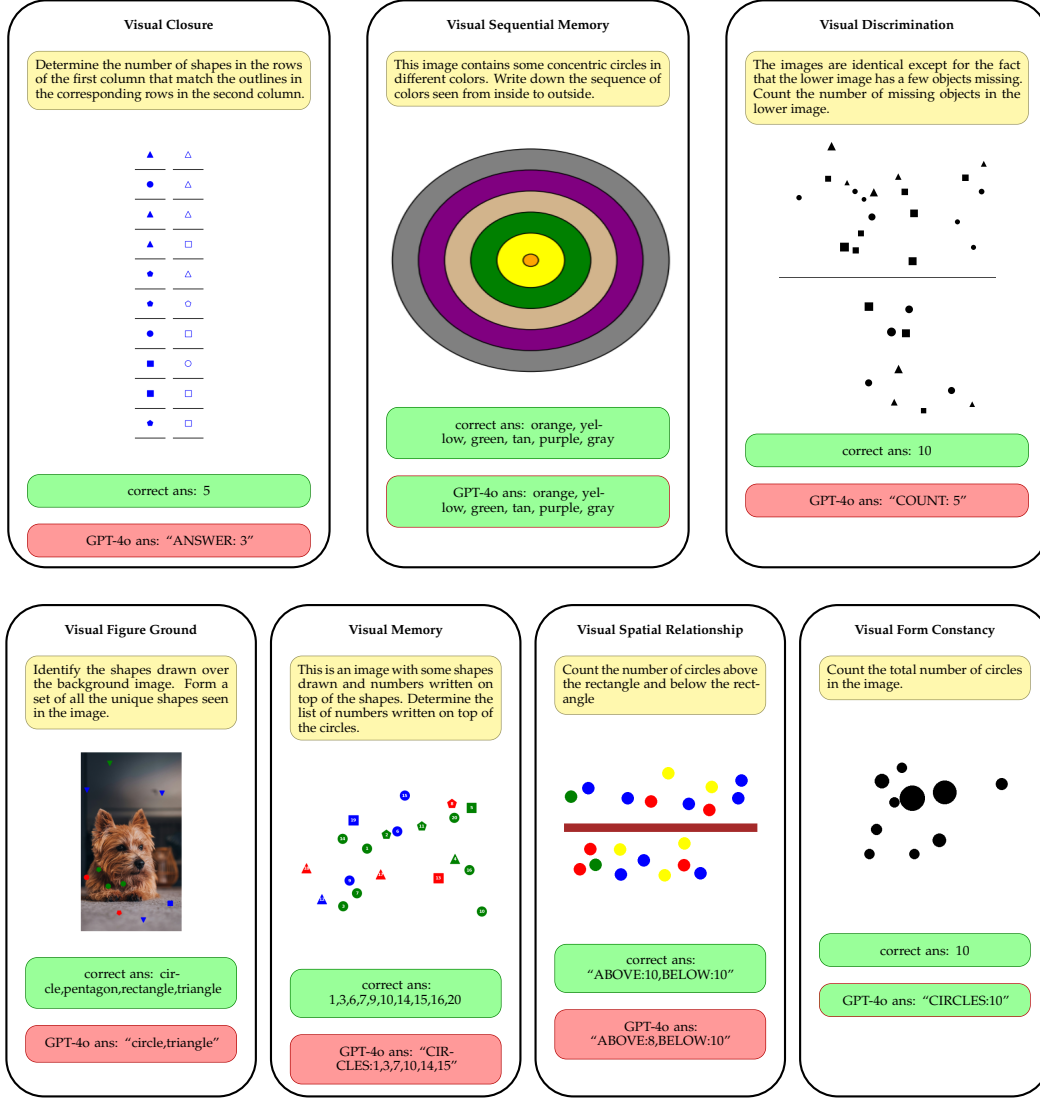


Figure 1: Sample questions from Percept-V illustrating tasks related to different visual perception skills as defined by the TVPS-4 framework. Each example shows the summarised image prompt, the correct answer, and a typical incorrect response from an MLLM.

perception question over an image with simple geometric objects (e.g., circles, triangles) varying in colour and size. Such synthetic 2D images have applications in robotics (Li et al., 2024a) and mathematics (Wang et al., 2024).

To systematically assess performance across various difficulty levels, each domain contains 200 images with varying problem size, which is operationalised through object count, grid size, or required processing steps. This enables us to evaluate MLLM performance as a function of visual complexity and provides insight into how they handle increasing perceptual demands. We evaluate 4 SOTA proprietary MLLMs (GPT-5-mini, GPT-4o, o4-mini, Gemini 2.5 Flash), 4 open-source models (Qwen 2.5 VL Instruct-7B, DeepSeek VL2 Tiny, Qwen-3VL-8B-Thinking and Gemma-4-31B-it), and also conduct a human study. Our findings show significant gaps between human and model performance, with consistent degradation across all evaluated models as problem size increases, suggesting systematic limitations in handling visual complexity. While some models perform relatively better on specific skills, overall performance spans a wide range across models, indicating broad visual perceptual limitations. We also investigate the effects of LoRA and full finetuning

on our dataset. Overall, our work is an important step towards understanding the visual perceptual limitations that must be addressed in the future to develop more robust MLLMs.

Skill	Domain
Visual Discrimination (D)	change colour, vanishing objects, comparing size, counting shapes, sort lines, sort circles, locate circles colour, locate circles shape, match shadow, match outline
Visual Memory (M)	list colours, list shapes, circle boxes, colours present, count coloured circles, numbered shapes, graph counting, sort circles, sort lines, counting shapes, identifying shapes
Visual Sequential Memory (SM)	grid path, layered colours, layered shapes, comparing size, match outline, match shadow
Visual Figure Ground (FG)	list colours, list shapes, locate circles colour, locate circles shape, layered shapes, layered colours
Visual Form Constancy (FC)	mirror image, water image, counting circles, counting shapes, numbered shapes
Visual Closure (C)	match outline, match shadow, layered shapes, layered colours
Visual Spatial Relationship (SR)	comparing size, circle location, circle right triangle, counting locations, cross and knots, inside circles, maze solving

Table 1: Domains using each skill. Note that a single domain can appear in multiple rows in the above table since multiple skills may be required to solve its Problem Instances

2 Related Work

A large number of benchmarks exist for text-only problems, and modern-day LLMs have shown excellent performance in complex reasoning tasks (Mittal et al., 2025; Arora et al., 2023). Most existing benchmarks for evaluating MLLMs require a combination of perception, reasoning, and domain knowledge. For instance, NTSEBench (Pandya et al., 2025) evaluates commonsense reasoning, MathVista (Lu et al., 2023) focuses on compositional reasoning, and MathVision (Wang et al., 2024) spans 16 mathematical disciplines. Similarly, OlympiadBench (He et al., 2024) includes advanced problems from mathematics, and science, and IQBench (Pham et al., 2025) tests MLLMs on visually-centric IQ tests. Large-scale benchmarks such as MMMU, CMMMU, and EMMA (Yue et al., 2023; Zhang et al., 2024a; Hao et al., 2025) cover fields such as engineering, medicine, humanities, and business, while SciBench (Wang et al., 2023) contains college-level science questions. Datasets like ManBench and Rebus (Zhou et al., 2025; Das et al., 2025) require complex reasoning over images, whereas other datasets require interpreting abstract figures and plots (Li et al., 2024b). All these benchmarks entangle perception with reasoning and sometimes specialized knowledge.

Recent work has attempted to disentangle these abilities. MathLENS (Chung et al., 2025) separately evaluates perception, reasoning, and integrating abilities. Other studies highlight perceptual limitations of MLLMs (Lee et al., 2025), including poor performance on small or low-quality objects and fine-grained perception (Yan et al., 2025), and introduce simple perception-focused benchmarks (Fu et al., 2024). Benchmarks like VOILA (Yilmaz et al., 2025) test perceptual understanding through image generation. However, these works typically rely on complex scenes and do not analyze the impact of problem size.

Closest to our work are contemporary benchmarks like Kanade & Ganu (2026); Huang et al. (2026); Cao et al. (2025); Kamoi et al. (2025) that focus on perception without any specialized knowledge. Compared to these papers, our work introduces a plug-and-play pipeline for generating a dataset of any size with only simple instructions. Gold answers are also generated automatically, eliminating the need for human annotation. In contrast to Kanade & Ganu (2026), our dataset provides 30 novel domains grounded in all 7 skills of the TVPS-4

framework versus their 12 and the variation in problem size in our dataset is much more fine-grained which allows for systematic scaling in task complexity. Unlike VISFACTOR (Huang et al., 2026) which extracts gold truth answers from FRCT battery and lacks a completely automated evaluation process, both our gold-answer generation and evaluation of model performance are fully done programmatically. We also provide detailed analysis of skill-specific weaknesses, scaling behaviour, answer-type difficulties and post-finetuning generalization gaps in MLLMs. Unlike VisOnlyQA which uses a mix of synthetic and real images, Percept-V uses purely synthetic images with controllable parameters and focuses specifically on basic visual perception, enabling systematic evaluation without the influence of reasoning, semantic understanding, or domain knowledge.

3 Dataset

We describe our dataset, Percept-V, which consists of 30 domains designed to evaluate *basic perception* skills based on the TVPS-4 cognitive framework (Saboo, 1996). TVPS-4, being a popular test of visual perception in humans, helps us to ground our dataset in well-established parameters for evaluating perception. TVPS-4 defines seven visual perceptual skills: (1) *Visual Discrimination* (identifying similarities/differences in size, color, shape), (2) *Visual Memory* (recalling visual traits), (3) *Visual Sequential Memory* (recalling ordered sequences), (4) *Visual Spatial Relationships* (understanding object relationships), (5) *Visual Figure Ground* (locating objects in clutter), (6) *Visual Form Constancy* (recognizing shapes under transformations), and (7) *Visual Closure* (recognizing incomplete forms).

Each domain evaluates one or more of these skills and contains multiple *problem instances* with varying perceptual complexity. Overall, 17 domains test a single skill, 7 test two skills, and 6 test three skills jointly. Table 1 lists, for each skill, the domains that require that skill to solve the problem instances correctly. Looking further at the performance on the 17 single skill domains, we found that some skills are more difficult for most models than others. Fig 42 in section A.2 shows that ‘vsm’ i.e., Visual Sequential Memory is a weakness for 4 out of 8 models as they exhibit their lowest performance in that skill. Notably, visual memory and visual sequential memory traditionally involve recalling information after it is removed from view. Since MLLMs retain all input in their memory, our setup only approximates these skills as scanning and transcription tasks (which are a subset to the original skill). Despite this simplification, model performance remains low (Table 2), indicating fundamental perceptual limitations. Please refer to appendix section A.6 for a detailed discussion.

Each domain is associated with a unique question prompt, and every problem instance is a pair (Q, I) , where Q is the domain-specific question prompt and I is a Python programme generated image.¹ The same prompt is applied across images within a domain. Instances vary in *problem size*, which captures perceptual complexity through factors such as number of objects, distractors, or required processing steps. Each domain includes 20 sizes varying from 1 to 20, with 10 instances per size, yielding 200 instances per domain and 6000 instances overall. For example, in the *counting_circles* domain, size corresponds to the number of circles, while in *numbered_shapes*, both target objects and distractors increase with size. Sample instances are shown in Figure 1.

Each question prompt Q consists of three components (in, r, op) : *in* describes the input image, *r* specifies the task in a step-by-step manner with detailed instructions, and *op* defines the required answer format. The dataset includes four answer types : (i) single-value outputs (Boolean or numeric) (11 domains), (ii) fixed-length outputs (tuples or lists of constant size) (6 domains), (iii) ordered lists (sequence-sensitive outputs whose length varies with problem size) (8 domains), and (iv) unordered sets (collections of unique elements) (5 domains). Each prompt explicitly specifies the output format to enable automatic evaluation. In practice, Most MLLMs adhere well to formatting constraints; errors primarily arise from incorrect perception rather than format violations (Section 4.5).

All images are generated using automated scripts – one script per domain. Each script takes as input a set of problem sizes and the number of instances per size, and randomizes

¹Image generation is controlled by domain-specific hyperparameters (Appendix A).

Skill / Domain	GPT-5-mini	GPT-4o	o4-mini	Gemini	Gemma4	Qwen3VL	Qwen2.5VL	DeepSeek	Avg. Acc.
— Performance by Skill —									
Visual Discrimination	55.0	23.7	55.3	51.25	68.75	34.74	11.5	4.1	38.04
Visual Memory	49.95	33.55	47.55	46.55	64.0	30.12	19.59	1.32	36.58
Visual Sequential Memory	43.5	21.5	47.5	41.92	52.58	15.75	4.67	5.25	29.08
Visual Figure Ground	64.75	25.42	63.25	62.83	59.5	40.0	7.58	0.08	40.43
Visual Form Constancy	58.7	27.3	48.9	50.3	71.0	34.67	25.6	3.0	39.93
Visual Closure	44.88	23.0	50.25	43.25	43.5	13.12	5.75	7.88	28.96
Visual Spatial Relationship	66.64	50.36	62.64	69.79	84.5	54.36	27.79	17.64	54.22
Average of skills	54.77	29.26	53.63	52.27	63.4	31.82	14.64	5.61	38.18
— Performance by Domain —									
change.colour	32.0	18.5	32.0	23.5	58.5	8.5	8.0	4.5	23.19
circle_boxes	30.0	33.0	34.0	28.0	64.5	15.0	12.5	10.5	28.44
circle_location	56.5	41.5	46.0	57.5	81.5	22.5	14.5	5.5	40.69
circle_right_triangle	100.0	82.0	99.5	99.0	98.5	94.5	25.0	50.0	81.06
colours_present	3.5	1.5	4.5	4.5	5.5	1.5	0.0	0.0	2.62
comparing_size	27.0	36.5	31.0	55.0	81.0	31.5	5.0	0.0	33.38
count_coloured_circles	51.0	44.0	37.5	44.5	99.0	38.0	39.0	2.5	44.44
counting_circles	68.0	57.5	55.0	55.5	100.0	54.5	42.5	5.5	54.81
counting_locations	44.5	20.5	36.5	33.5	98.5	37.0	15.5	0.0	35.75
counting_shapes	72.5	44.0	58.5	61.0	100.0	62.37	41.0	0.0	54.92
cross_and_knots	99.5	57.5	100.0	96.0	81.5	89.5	23.5	0.0	68.44
graph_counting	39.0	31.0	33.5	36.0	51.5	25.5	26.0	0.0	30.31
grid_path	54.5	0.5	53.0	23.5	60.5	10.5	0.0	0.0	25.31
identifying_shapes	79.5	69.0	92.5	47.0	100.0	80.0	41.5	0.0	63.69
inside_circles	96.0	87.5	95.5	93.0	97.0	94.5	96.5	68.0	91.0
layered_colours	35.0	25.0	33.5	34.0	29.5	10.0	0.5	0.0	20.94
layered_shapes	24.0	11.5	20.0	18.5	16.5	11.5	9.5	0.0	13.94
list_colours	41.0	18.0	35.5	58.0	37.0	3.5	1.5	0.5	24.38
list_shapes	88.5	70.5	90.5	92.0	90.5	31.5	21.0	0.0	60.56
locate_circles.colour	100.0	11.5	100.0	82.0	92.0	95.0	6.0	0.0	60.81
locate_circles.shape	100.0	16.0	100.0	92.5	91.5	88.5	7.0	0.0	61.94
match_outline	66.0	23.0	84.5	72.5	82.0	19.0	7.5	17.0	46.44
match_shadow	54.5	32.5	63.0	48.0	46.0	12.0	5.5	14.5	34.5
maze_solving	43.0	27.0	30.0	54.5	53.5	11.0	14.5	0.0	29.19
mirror_image	33.5	12.5	29.0	24.0	32.5	3.5	21.5	5.0	20.19
numbered_shapes	81.5	12.0	75.5	88.5	84.0	49.5	8.0	1.0	50.0
sort_circles	36.5	24.0	34.0	30.0	31.5	0.0	20.0	0.0	22.0
sort_lines	26.5	22.0	27.0	22.5	40.5	24.5	5.0	0.0	21.0
vanishing_objects	35.0	9.0	23.0	25.5	64.5	6.0	10.0	5.0	22.25
water_image	38.0	10.5	26.5	22.5	38.5	3.5	15.0	3.5	19.75
Full Percept-V Dataset	55.22	31.65	52.7	50.75	66.92	34.48	18.1	6.43	39.53

Table 2: Comparison of MLLMs’ performance across different skills and domains (in %).

object placement to ensure diversity. Images consist of simple geometric shapes (e.g., circles, triangles, squares) arranged in basic structures such as grids, rows, and mazes. Ground-truth answers are computed during generation, eliminating ambiguity or annotation errors. Our programmatic generation provides two key advantages: (1) precise control over problem complexity, enabling systematic evaluation across difficulty levels, and (2) elimination of dataset contamination, a known issue in existing benchmarks (Shojaee et al., 2025).

4 Experiments

Through our experiments, we aim to answer the following questions: (1) How well do state-of-the-art (SoTA) open and proprietary MLLMs perform on Percept-V? (2) Do some models perform better than others, particularly for specific skills? (3) How does performance vary across skills — are some skills inherently harder? (4) How does performance change with increasing problem size? (5) How does MLLM performance compare to humans? (6) How does finetuning affect the performance on our dataset?

To answer these, we evaluate 4 proprietary MLLMs: GPT-4o, Gemini 2.5 Flash, o4-mini, and GPT-5-mini, and 4 open-weight models: Qwen-2.5-VL-7B-Instruct, DeepSeek-VL2-Tiny, Qwen-3-VL-8B-Thinking and Gemma-4-31B-it. Among these, GPT-5-mini, o4-mini, Gemini, Qwen3 and Gemma-4 support “think with image” capabilities (Su et al., 2025), i.e., leverage visual information during intermediate reasoning. Experiments for open-source models are conducted on an NVIDIA A100 GPU (details in Appendix A). To ensure reproducibility, hyperparameters such as temperature and top_p are set to 0.0 wherever possible (restricted for some models). All experiments are conducted primarily in a zero-shot setting; one-shot and few-shot prompting were also tested but yielded negative results. For each domain, the same prompt (question, output format) is used across all models (see Appendix).

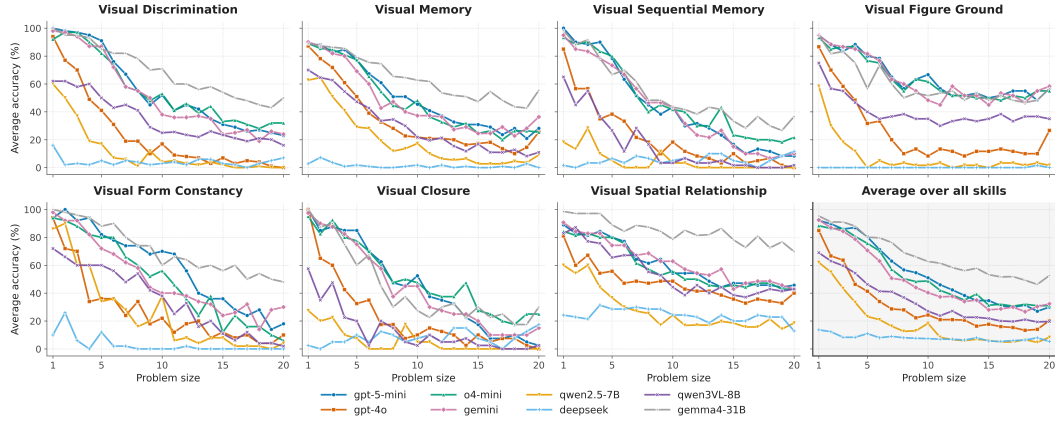


Figure 2: The overall accuracy of all models in different skills.

Domain-wise Analysis: Table 2 shows performance across domains and skills. Across most domains, MLLM performance is poor. Among proprietary models, GPT-4o (non-thinking) performs the worst, achieving $\leq 20\%$ accuracy on 11 of 30 domains and $\leq 50\%$ on 24 domains. Reasoning models perform somewhat better: o4-mini achieves $\leq 20\%$ on only 2 domains and $\leq 50\%$ on 17, with Gemini and GPT-5-mini showing similar trends. The open-source models span a far wider range than the proprietary ones: Gemma-4-31B is the strongest model in our evaluation, falling $\leq 50\%$ on only 9 domains and $\leq 20\%$ on 2, and Qwen-3VL-8B already edges out GPT-4o; at the other end Qwen-2.5VL is $\leq 20\%$ on 20 domains and exceeds 50% on only one, while DeepSeek scores exactly zero on 16. Overall, Gemma-4-31B achieves the highest average performance, but only at 66.92%.

These results highlight that both proprietary and open-source MLLMs — regardless of their reasoning capability — fail to achieve strong performance across most domains. This contrasts sharply with human performance (Section 4.3). Additionally, performance patterns are consistent across models: domains such as *inside_circles* show relatively higher performance across all models, whereas *colours_present* consistently exhibits the lowest accuracy. The latter requires identifying colours of objects, and confusion between similar colours (e.g., turquoise vs. blue), despite provided swatches and legends, may explain the difficulty. This suggests that certain domains are intrinsically harder for current MLLMs.

Skill-wise Analysis: Skill-level analysis reveals similar trends. Gemma-4-31B achieves the highest average accuracy across skills (63.40%), followed by GPT-5-mini (54.77%), o4-mini and Gemini. Even the best model achieves more than 70% on only 2 of 7 skills while none of the other models cross this threshold, indicating sub-optimal performance across skills. GPT-4o performs significantly worse (29.26%), while Qwen-2.5VL achieves only 14.64% and DeepSeek 5.61%. A significant differentiator in model performance is whether it is thinking or not - every thinking model consistently outperforms the non-thinking open models which trail far behind.

Across all models, some skills are consistently easier or harder. Most models perform best on Visual Spatial Relationships, likely due to the prevalence of spatial descriptions (e.g., “left of”, “above”) in training data. In contrast, Visual Closure is among the hardest skills, involving tasks such as recognizing partially occluded shapes or matching outlines to filled objects — a phenomena likely underrepresented in training data. This consistent pattern suggests inherent skill-specific weaknesses in MLLMs.

4.1 Performance vs Size

To analyze performance scaling (Q4), we evaluate accuracy across varying problem sizes (Figure 2). Each of the seven plots corresponds to a skill, with an additional plot (last plot of second row) showing average performance across skills.

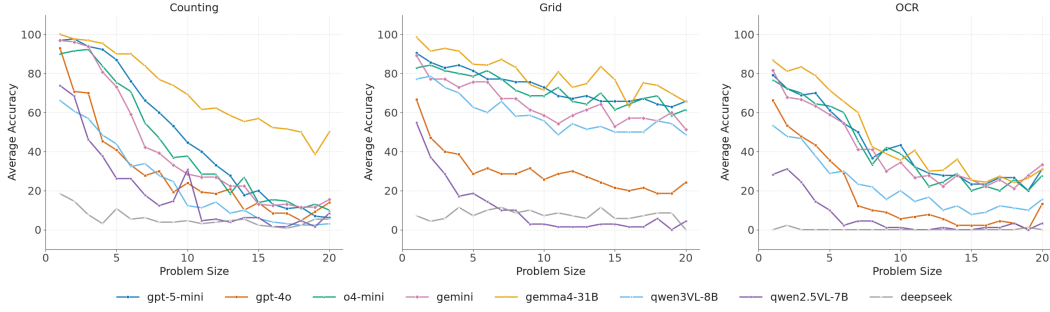


Figure 3: The overall accuracy of all models in counting, grid understanding and ocr skills.

We observe a consistent and significant drop in performance as problem size increases across all skills and models. Reasoning models (Gemma-4, o4-mini, Gemini) often achieve near-perfect accuracy on smaller instances (fewer objects or simpler layouts), but performance degrades sharply as complexity increases. Gemma-4, GPT-5-mini, o4-mini, and Gemini are relatively more robust compared to others, though still exhibit substantial declines. GPT-4o follows, while other open-source models degrade most severely. These findings indicate that MLLMs lack robustness to increasing visual complexity and fail to generalize effectively across sizes.

4.2 Alternate Skill Analysis

Beyond TVPS-4, three additional recurring skills emerge in our dataset: counting, optical character recognition (OCR), and grid understanding, appearing in 13, 9, and 7 domains respectively. While counting is a basic reasoning skill included mainly to facilitate automatic evaluation, it may distract from the main results. Comparing counting (C) vs. non-counting (NC) domains yields: GPT-4o (29.04 vs. 33.65), GPT-5-mini (47.73 vs. 60.94), o4-mini (43.00 vs. 60.12), Gemini (40.92 vs. 58.26), Gemma-4 (70.54 vs. 64.15), Qwen-3VL (23.64 vs. 42.76), Qwen-2.5VL (19.88 vs. 16.74), and DeepSeek (5.65 vs. 7.03). Counting appears to slightly increase difficulty for most of the reasoning models, with Gemma-4 the notable exception. However, poor performance on non-counting domains indicates that counting alone does not explain overall weaknesses.

Figure 3 shows performance vs. size for these skills. Both counting and OCR exhibit sharp performance degradation with increasing size. In contrast, grid-based tasks show relatively stable performance, particularly for reasoning models (Gemma-4, GPT-5-mini, o4-mini, Gemini), with near-plateau trends. Notably, GPT-5-mini achieves 100% accuracy in 3 out of 7 grid domains. This may reflect training data biases, as grid-like structures are common in tasks such as reinforcement learning with visual reasoning (RLVR).

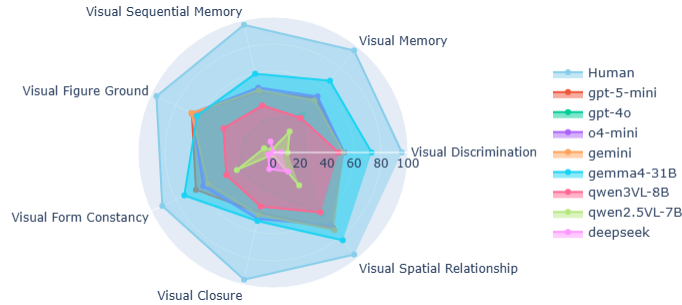


Figure 4: MLLM average accuracy vs human accuracy across the 7 TVPS-4 skills.

4.3 Human Study

We now answer Q5 through a human study where we recruit 16 college students, 6 female and 10 male. They were presented with randomly selected questions of any problem size from the 30 domains. Results of this study are presented in Figure 4. For fair comparison, the models are also evaluated on the exactly same samples, and exactly the same protocol as the humans. It can be seen that models lag severely as compared to humans. Even the best performance of the models exhibited by Gemini in Visual Spatial Relationship is around 73.08% as compared to about 96.42% accuracy shown by humans in the same skill. Overall (see Table 6 in appendix) – human performance is more than 26% points higher than the closest MLLM. Interestingly, when testing on ‘colours_present’, the domain on which all MLLMs have abysmal performance, we find the gap to be humongous – humans achieve 90% performance whereas the best MLLMs on the same questions achieve only 6.67%.

4.4 One-shot and Few-shot Results:

We additionally perform one-shot experiments on proprietary models to assess how much exemplars can help MLLMs in understanding the task better, as well as to account for prompt sensitivity. Contrary to conventional wisdom, we find that one-shot prompting *degrades* the overall performance, with average of skills accuracy reducing to 51.25% for GPT-5-mini, 17.62% for GPT-4o, 52.26% for o4-mini and 44.42% for Gemini (refer to Table 7 in Appendix). Moreover, the skill-wise graphs also show degrading trends across sizes, similar to zero-shot (see Fig. 43 in Appendix). Upon manual investigation of frequent errors made by models, we see that the models confuse the details of the in-context example with the problem to be solved. The models also commonly fail to perceive the visual details of the in-context examples and thus hallucinate wrong theories for solving the task. Further, a short few-shot experiment with 3 examples on one of the stronger proprietary models, Gemini showed that these mistakes recurred and the average accuracy on Percept-V dropped to 21.5%. Upon investigating model responses, we found that models frequently mixed details from the demonstration example with those of the target instance, particularly in tasks involving ordered outputs or spatial layouts. We believe this reveals an interesting limitation of current MLLMs in maintaining a clean separation between demonstration context and active image understanding.

4.5 Additional Experimental Analysis

Format Error Analysis: Table 3 reports the accuracy of models in following answer formats - we see that proprietary models consistently follow answer formats while open source models are worse than them. The best performing is GPT-4o (with o4-mini close second) that sticks to the output formats with 0% format errors.

Answer Type	GPT-5-mini	GPT-4o	o4-mini	Gemini	Gemma4	Qwen3VL	Qwen2.5VL	DeepSeek
single	0.14	0.00	0.00	0.00	11.95	15.09	0.00	3.68
fixed	0.00	0.00	0.17	0.83	4.08	13.00	5.18	61.58
list	2.19	0.00	0.00	0.00	5.94	14.19	0.00	22.44
set	0.00	0.00	0.10	0.20	6.4	2.4	19.10	60.00

Table 3: Format Error (%) Analysis of models

3D Percept-V: A potential criticism of Percept-V is that its images are auto-generated and may not occur naturally in model’s training data. Will the models perform better if tested on more realistic images? To further investigate this concern, we developed 3D versions of a subset of Percept-V domains using Blender in the style of the CLEVR dataset (Johnson et al., 2016). For six domains, we created 100 synthetic images spanning sizes 1–10 (10 images per size), where all 2D objects were converted into 3D objects with realistic lighting and shadows to mimic real-world photographs (Fig 41). We present the results with GPT-5-mini and Qwen2.5-7B in Table 9 which shows that even for relatively small problem sizes-where performance on the corresponding 2D domains is comparatively high-performance on the 3D versions remains sub-par for most domains. Thus, as anticipated, simply introducing

3D rendering does not resolve the underlying perception challenges. Instead, these results point toward a deeper gap in the visual perceptual capabilities of current MLLMs.

Impact of Image Resolution: Percept-V images are already reasonably high resolution (670×650 pixels on average). To test whether resolution limits performance, we re-evaluated GPT-5-mini and Qwen on all images upscaled and downscaled by a factor of two. As reported in Table 4 upscaling does not provide much benefit: GPT-5-mini moves from 54.77 to 55.62 and Qwen from 14.64 to 15.01. This hints that the information required to solve these tasks was already present in the input and the failures are not caused by lack of clarity. Downscaling does hurt, as GPT-5-mini falls by 14 points and Qwen by 2.5 points, confirming that the tasks are resolution-sensitive in the expected direction and that the models are genuinely reading the images rather than guessing. Notably, the skills that suffer most under downscaling (Visual Memory and Visual Form Constancy) are those requiring many small objects to be individuated at once.

Skill	GPT-5-mini			Qwen		
	Orig.	2×	0.5×	Orig.	2×	0.5×
Visual Discrimination	55.00	54.80	45.83	11.50	11.70	10.20
Visual Memory	49.95	49.94	32.92	19.59	22.83	17.22
Visual Sequential Memory	43.50	42.17	29.42	4.67	4.83	4.33
Visual Figure Ground	64.75	63.12	52.00	7.58	4.88	4.62
Visual Form Constancy	58.70	61.90	38.15	25.60	25.20	21.50
Visual Closure	44.88	45.75	32.75	5.75	5.25	5.75
Visual Spatial Relationship	66.64	71.64	57.07	27.79	30.36	24.07
Average of skills	54.77	55.62	41.16	14.64	15.01	12.53

Table 4: Accuracy (%) at original resolution (670×650 pixels on average), upscaled 2×, and downscaled 0.5×.

Impact of Test Phrases: We paraphrased the original prompts in two different styles - concise and verbose to understand the effects of prompt-styles on model performance. Table 8 in Appendix shows the performance of GPT-5-mini and Qwen respectively on the different styles of prompts. From Table 8, it is clear that the style of text information does not affect the performance of models too much. In fact, the original Chain-of-Thought style prompting provides better results in many cases. Also, performance of concise prompts is mostly better than verbose prompts in case of both the models.

5 Finetuning experiments

In this section we answer Q6 by exploring the performance of Qwen2.5-VL-7B when fine-tuned on Percept-V style data. We compare different configurations of LoRA (Hu et al., 2022) and full-finetuning for both in-domain performance and out-of-domain generalization. To this end, we create a large version of Percept-V which we call Percept-V_{train} for all finetuning based experiments, consisting of 60K samples with 2K samples for each domain. Evaluation is performed on Percept-V (in-domain) as well as CV-Bench (Tong et al., 2024) and Color-Bench (Liang et al., 2025) for out-of-domain assessment. We experiment with fine-tuning three different modules, the language encoder, vision encoder, and the projector (which aligns the two modalities). We perform these experiments with LoRA fine-tuning, and we also compare with fine-tuning the entire network. The results are presented in Table 5, with implementation details in A.7.

It is evident from Table 5 that both LoRA and full fine-tuning yields significant gains (over 60pts) over the pre-trained model. Interestingly, LoRA achieves performance only marginally worse than full-fine tuning. We see most of the gains coming from training the language encoder (LLM), followed by the vision encoder, which means that training even one of the modalities has a significant boost on performance. Fine-tuning both the modalities together results in further gains. Surprisingly, adding the projector to the set of modules being fine-tuned, does not benefit (last but one row in LoRA experiments), and we believe this may be due to the need to have a more careful curriculum during fine-tuning. This may

LLM	Vision Encoder	MLP Projector	Percept-V Overall	Percept-V 1-10 Problem Size	Percept-V 11-20 Problem Size	CV-Bench Combined	Colorbench Overall
<i>Baseline</i>							
—	—	—	18.10	29.81	06.35	73.92	46.28
<i>LoRA Fine-tuning</i>							
□	□	✓	49.15	65.05	33.28	74.02	47.43
□	✓	□	68.67	80.76	56.60	68.49	46.49
✓	□	□	75.18	84.33	66.05	76.70	47.54
□	✓	✓	70.20	82.49	57.91	74.02	47.43
✓	□	✓	74.63	83.80	65.49	78.06	42.72
✓	✓	□	83.42	89.47	77.38	74.77	44.71
✓	✓	✓	80.30	87.56	73.06	75.62	47.02
<i>Full Fine-tuning</i>							
□	□	✓	25.25	38.71	11.78	74.46	46.28
□	✓	□	75.80	87.5	64.09	72.53	42.61
✓	□	□	82.60	92.60	72.57	58.91	44.60
□	✓	✓	76.35	87.32	65.36	73.46	44.08
✓	□	✓	78.83	90.93	66.69	63.38	44.71
✓	✓	□	84.18	93.55	74.79	69.54	46.28
✓	✓	✓	84.30	93.33	75.27	77.56	45.97

Table 5: **Legend:** L: LLM, V: Vision Encoder, P: MLP Projector between LLM and Vision Encoder. Ordering in LoRA style Fine-tuned models (overall accuracy on Percept-V): (L+V) > (L+V+P) > (L) > (L+P) > (V+P) > (V) > (P)

further imply that models struggle to query, align and transform visual information and motivates stronger interfaces between vision and language modalities.

Table 5 also compares the performance across the test data divided across problems of sizes 1-10, and 11-20. Naturally, we observe better performance on smaller problems which is expected. Interestingly, the performance gap decreases from 23 points (first row), to 18 points (last row) for the two subsets of the data, after full-fine tuning. This means that fine-tuning is able to address some aspects of the hardness of the problem, resulting from increasing size. For most part, the fine-tuning benefits do not significantly carry over to other similar datasets, which are on real images but require basic perceptual abilities similar to ours, with only a marginal increase in one of the datasets (though there is no catastrophic forgetting either). This likely highlights the fact that fine-tuned model is primarily able to learn the patterns on the given task, and is not fully able to generalize the learned ability to other tasks requiring similar skills.

6 Conclusion

We present Percept-V, an extensive dataset of 6000 automatically generated images from 30 domains where each domain is mapped with a subset of perceptual skills from the TVPS-4 Cognitive Science framework. A variety of open and closed source MLLMs perform rather weakly on almost all domains, with best average performance of any model being 66.9%, demonstrating significant gap in their ability to solve basic perceptual tasks. Our human study experiments further show that a significant gap exists between human and LLM performance on very simple perception tasks in our dataset. Interestingly, fine-tuning on our dataset improves the performance, though does not help with generalization on other similar datasets. Ablation studies suggest that visual representation and multimodal models both require fine tuning for best performance, pointing to the limitations of the underlying representation and alignment. Two failure mode recur throughout our experiments on pre-trained models: accuracy falls steadily as problem size grows and in-context examples hurt rather than help. Both point to issues with visual capacity and the handling of multi-image context as the limitations that must be addressed to improve basic perception. Our data and code are available at <https://dair-iitd.github.io/Percept-V/>.

Ethics Statement

All human participants were shown the instructions required to be followed during the experiment, and they volunteered to participate in the study. Additionally, all humans were asked for consent and financially compensated for their contribution – details can be found in the A.8 of Appendix A. Our work only tries to expose the perceptual limitations of SoTA MLLMs, so human study did not elicit any personally identifying information – the participants were simply answering simple perception questions.

Acknowledgements

Mausam is supported by grants from Google, Microsoft, Wipro, Verisk, Tower Research and Nick McKeown Chair Professorship. Parag Singla is supported by Shanthi and K Ananth Krishnan Young Faculty Chair Professorship. Parag and Mausam are supported by the IBM AI Horizon Networks (AIHN) grant. We thank IIT Delhi HPC compute facility, Microsoft AFMR program and Google Gemini Credits grant for access to compute and LLM API calls, Yardi School of Artificial Intelligence for travel grant. Any opinions, findings, conclusions or recommendations expressed here are those of the authors and do not necessarily reflect the views or official policies of the funding agencies.

References

- Daman Arora, Himanshu Gaurav Singh, and Mausam. Have llms advanced enough? A challenging problem solving benchmark for large language models. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pp. 7527–7543. Association for Computational Linguistics, 2023.
- Xu Cao, Bolin Lai, Wenqian Ye, Yunsheng Ma, Joerg Heintz, Jintai Chen, Jianguo Cao, and J. Rehg. What is the visual cognition gap between humans and multimodal llms?, 2025. URL <https://arxiv.org/abs/2406.10424>.
- Jiwan Chung, Neel Joshi, Pratyusha Sharma, Youngjae Yu, and Vibhav Vineet. What mllms learn about when they learn about multimodal reasoning: Perception, reasoning, or their integration?, 2025. URL <https://arxiv.org/abs/2510.01719>.
- Trishanu Das, Abhilash Nandy, Khush Bajaj, and S. Deepiha.  **bus**: A large and diverse multimodal benchmark for evaluating the ability of vision-language models to understand rebus puzzles, 2025. URL <https://arxiv.org/abs/2511.01340>.
- Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A. Smith, Wei-Chiu Ma, and Ranjay Krishna. BLINK: multimodal large language models can see but not perceive. In Ales Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part XXIII*, volume 15081 of *Lecture Notes in Computer Science*, pp. 148–166. Springer, 2024. doi: 10.1007/978-3-031-73337-6_9. URL https://doi.org/10.1007/978-3-031-73337-6_9.
- Yunzhuo Hao, Jiawei Gu, Huichen Will Wang, Linjie Li, Zhengyuan Yang, Lijuan Wang, and Yu Cheng. Can mllms reason in multimodality? EMMA: an enhanced multimodal reasoning benchmark. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025. URL <https://proceedings.mlr.press/v267/hao25a.html>.
- Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Z. Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark

- for promoting agi with olympiad-level bilingual multimodal scientific problems. *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3828–3850, 2024. doi: 10.18653/v1/2024.acl-long.211.
- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL <https://openreview.net/forum?id=nZeVKeeFYf9>.
- Jen-Tse Huang, Dasen Dai, Jen-Yuan Huang, Youliang Yuan, Xiaoyuan Liu, Wenxuan Wang, Wenxiang Jiao, Pinjia He, and Zhaopeng Tu. Human cognitive benchmarks reveal foundational visual gaps in mllms, 2026. URL <https://arxiv.org/abs/2502.16435>.
- Justin Johnson, B. Hariharan, L. Maaten, Li Fei-Fei, C. L. Zitnick, and Ross B. Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Computer Vision and Pattern Recognition*, pp. 1988–1997. IEEE, 2016. doi: 10.1109/CVPR.2017.215.
- Ryo Kamoi, Yusen Zhang, Sarkar Snigdha Sarathi Das, Ranran Haoran Zhang, and Rui Zhang. Visonlyqa: Large vision language models still struggle with visual perception of geometric information, 2025. URL <https://arxiv.org/abs/2412.00947>.
- Aditya Sanjiv Kanade and Tanuja Ganu. Do you see me : A multidimensional benchmark for evaluating visual perception in multimodal llms. In Vera Demberg, Kentaro Inui, and Lluís Marquez (eds.), *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2026 - Volume 1: Long Papers, Rabat, Morocco, March 24-29, 2026*, pp. 7285–7326. Association for Computational Linguistics, 2026. doi: 10.18653/V1/2026.EACL-LONG.343. URL <https://doi.org/10.18653/v1/2026.eacl-long.343>.
- Michael King. Administration of the text-based portions of a general iq test to five different large language models. *Authorea Preprints*, 2023. doi: 10.36227/techrxiv.22645561.v1.
- Jonghyun Lee, Dojun Park, Jiwoo Lee, Hoekeon Choi, and Sung-Eun Lee. Exploring multimodal perception in large language models through perceptual strength ratings, 2025. URL <https://arxiv.org/abs/2503.06980>.
- Jinming Li, Yichen Zhu, Zhiyuan Xu, Jindong Gu, Minjie Zhu, Xin Liu, Ning Liu, Yaxin Peng, Feifei Feng, and Jian Tang. Mmro: Are multimodal llms eligible as the brain for in-home robotics?, 2024a. URL <https://arxiv.org/abs/2406.19693>.
- Lei Li, Yuqi Wang, Runxin Xu, Peiyi Wang, Xiachong Feng, Lingpeng Kong, and Qi Liu. Multimodal arxiv: A dataset for improving scientific comprehension of large vision-language models. *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14369–14387, 2024b. doi: 10.18653/v1/2024.acl-long.775.
- Yijun Liang, Ming Li, Chenrui Fan, Ziyue Li, Dang Nguyen, Kwesi Cobbina, Shweta Bhardwaj, Jiuhai Chen, Fuxiao Liu, and Tianyi Zhou. Colorbench: Can vlms see and understand the colorful world? A comprehensive benchmark for color perception, reasoning, and robustness. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/d34497330b1fd6530f7afd86d0df9f76-Abstract-Datasets_and_Benchmarks_Track.html.
- Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. *International Conference on Learning Representations*, 2023.

- Chinmay Mittal, Krishna Kartik, Mausam, and Parag Singla. Fcorebench: Can large language models solve challenging first-order combinatorial reasoning problems?, 2025. URL <https://arxiv.org/abs/2402.02611>.
- Pranshu Pandya, Vatsal Gupta, Agney S Talwarr, Tushar Kataria, Dan Roth, and Vivek Gupta. Ntsebench: Cognitive reasoning benchmark for vision language models. *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 3680–3708, 2025. doi: 10.18653/v1/2025.findings-naacl.204.
- Tan-Hanh Pham, Phu-Vinh Nguyen, D. Hung, Trong-Duong Bui, V. Thanh, Chris Ngo, T. Truong, and Truong-Son Hy. Iqbench: How “smart” are vision-language models? a study with human iq tests, 2025. URL <https://arxiv.org/abs/2505.12000>.
- R. Rabindran and D. Madanagopal. Piaget’s theory and stages of cognitive development—an overview. *Scholars Journal of Applied Medical Sciences*, 8(9):2152–2157, 2020. doi: 10.36347/sjams.2020.v08i09.034.
- Frances J. Saboo. *Test of visual perceptual skills*, volume 420. Academic Therapy Publications Novato, CA, 1996.
- Susanne Schregel. ‘the intelligent and the rest’: British mensa and the contested status of high intelligence. *History of the Human Sciences*, 33(5):12–36, 2020. doi: 10.1177/0952695120970029.
- P. Shojaei, Iman Mirzadeh, Keivan Alizadeh-Vahid, Maxwell Horton, Samy Bengio, and Mehrdad Farajtabar. The illusion of thinking: Understanding the strengths and limitations of reasoning models via the lens of problem complexity. *Advances in Neural Information Processing Systems 38*, pp. 119605–119646, 2025. doi: 10.52202/085713-3603.
- Zhaochen Su, Peng Xia, Hangyu Guo, Zhenhua Liu, Yan Ma, Xiaoye Qu, Jiaqi Liu, Yanshu Li, Kaide Zeng, Zhengyuan Yang, et al. Thinking with images for multimodal reasoning: Foundations, methods, and future frontiers, 2025. URL <https://arxiv.org/abs/2506.23918>.
- Nguyen Ha Thanh. A brief report on lawgpt 1.0: A virtual legal assistant based on gpt-3. *arXiv.org*, 2023. doi: 10.48550/arXiv.2302.05729.
- Peter Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Adithya Iyer, Sai Charitha Akula, Shusheng Yang, Jihan Yang, Manoj Middepogu, Ziteng Wang, Xichen Pan, Rob Fergus, Yann LeCun, and Saining Xie. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024. URL http://papers.nips.cc/paper_files/paper/2024/hash/9ee3a664ccfeabc0da16ac6f1f1cfe59-Abstract-Conference.html.
- Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Mingjie Zhan, and Hongsheng Li. Measuring multimodal mathematical reasoning with math-vision dataset. *Advances in Neural Information Processing Systems 37*, 37:95095–95169, 2024. doi: 10.52202/079017-3014.
- Xiaoxuan Wang, Ziniu Hu, Pan Lu, Yanqiao Zhu, Jieyu Zhang, Satyen Subramaniam, Arjun R Loomba, Shichang Zhang, Yizhou Sun, and Wei Wang. Scibench: Evaluating college-level scientific problem-solving abilities of large language models. *International Conference on Machine Learning*, 2023. doi: 10.48550/arXiv.2307.10635.
- Hao Yan, Xingchen Liu, Hao Wang, Zhenbiao Cao, Handong Zheng, Liang Yin, Xinxing Su, Zihao Chen, Jihao Wu, Minghui Liao, et al. Visuriddles: Fine-grained perception is a primary bottleneck for multimodal large language models in abstract visual reasoning, 2025. URL <https://arxiv.org/abs/2506.02537>.

- Nilay Yilmaz, Maitreya Patel, Yiran Lawrence Luo, Tejas Gokhale, Chitta Baral, Suren Jayasuriya, and Yezhou Yang. VOILA: evaluation of mllms for perceptual understanding and analogical reasoning. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL <https://openreview.net/forum?id=q5MUM1Hxpd>.
- Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Computer Vision and Pattern Recognition*, pp. 9556–9567. IEEE, 2023. doi: 10.1109/CVPR52733.2024.00913.
- Ge Zhang, Xinrun Du, Bei Chen, Yiming Liang, Tongxu Luo, Tianyu Zheng, Kang Zhu, Yuyang Cheng, Chunpu Xu, Shuyue Guo, et al. Cmmmu: A chinese massive multi-discipline multimodal understanding benchmark. *arXiv.org*, 2024a. doi: 10.48550/arXiv.2401.11944.
- Jiarui Zhang, Jinyi Hu, Mahyar Khayatkhoei, Filip Ilievski, and Maosong Sun. Exploring perceptual limitation of multimodal large language models. *arXiv.org*, abs/2402.07384, 2024b. doi: 10.48550/ARXIV.2402.07384. URL <https://doi.org/10.48550/arXiv.2402.07384>.
- Han Zhou, Qitong Xu, Yiheng Dong, and Xin Yang. Manbench: Is your multimodal model smarter than human? In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, volume ACL 2025 of *Findings of ACL*, pp. 3423–3449. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.FINDINGS-ACL.178. URL <https://doi.org/10.18653/v1/2025.findings-acl.178>.

A Appendix

A.1 Domain examples of MLLMs

A.1.1 Change Color

Input Prompt:

- These are two identical images with circles of different colours. - The colours of the circles at the same position in the two images may be different. - Count the number of differently coloured circles between the two images. - The output must be given in a single line in the form of COUNT:x

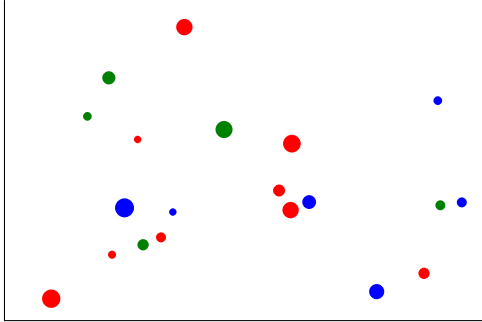


Figure 5: Change Color First Image

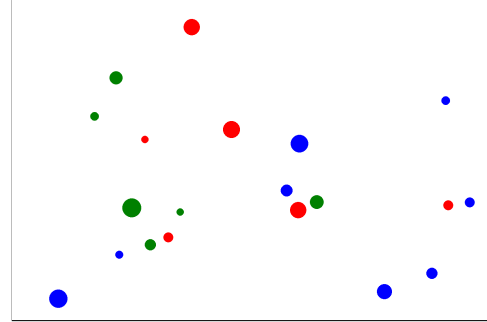


Figure 6: Change Color Second Image

A.1.2 Circle Boxes

Input Prompt:

- This is an image with some red circles divided and a blue partition that divides the image into two sides. - Some circles are on the left side of the partition whereas some are on the right. Note that, a side of the image may be completely empty as well. - Determine the number of circles to be moved from the side with more number of circles to the side with less number of circles so that each side has equal number of circles. - In case of both the sides having equal number of circles, return zero. - The output must be given in a single line in the form of COUNT:x where x is the number of circles to be transferred

A.1.3 Circle Location

Input Prompt:

- This is an image with some black circles drawn in the four quadrants - The quadrants are numbered in the default way i.e. quadrant 1 is the rightmost and topmost quadrant, quadrant 2 is the leftmost and topmost, quadrant 3 is the leftmost and lowest and quadrant 4 is the rightmost and lowest. - Determine the quadrant with the most number of black circles and the count of black circles in that quadrant. - In case of a tie, return the lesser quadrant number. - The output must be given in a single line in the form of QUADRANT:x COUNT:y where x is the quadrant number and y is the count of circles.

A.1.4 Circle Right Triangle

Input Prompt:

- This is a grid which contains some circles and triangles - Check if any triangle cell has a circle cell to its immediate right. - The last line of output should be "NO" if no such cell exists and "YES" otherwise.

A.1.5 Colours Present

Input Prompt:

- This image contains some shapes in different colours. - The colours belong to the following

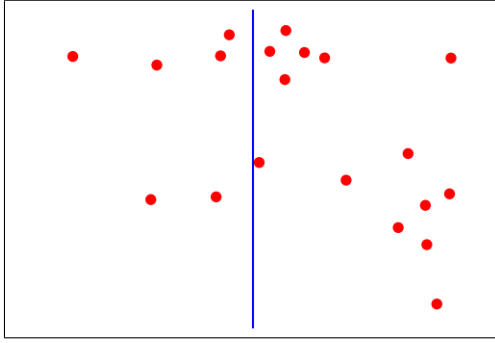


Figure 7: Circle Boxes

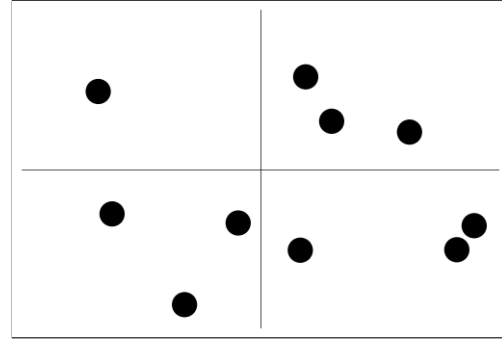


Figure 8: Circle Location

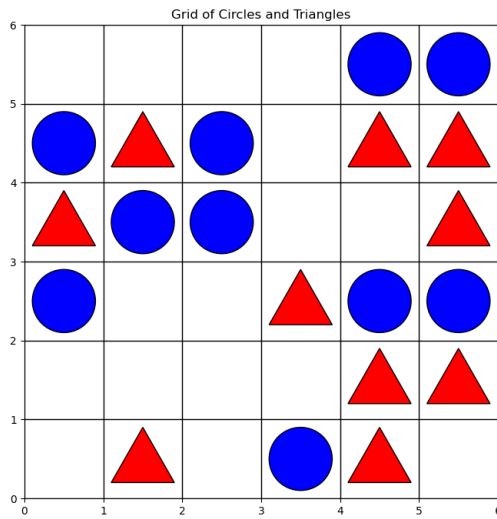


Figure 9: Circle Right Triangle

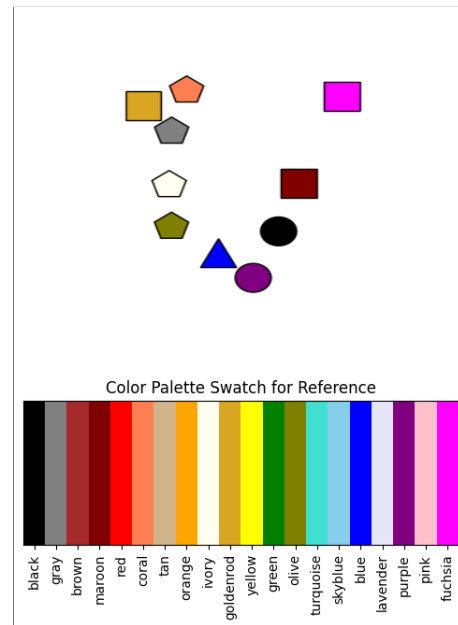


Figure 10: Colours Present

list - [black, gray, brown, maroon, red, coral, tan, orange, ivory, goldenrod, yellow, green, olive, turquoise, skyblue, blue, lavender, purple, pink, fuchsia] - No colour is repeated - Produce a list of yes or no; write yes in the list if the colour at the corresponding index of the above list is present and no if it is absent. - The output should be written in a single line as a comma-separated list of yes or no, written in order of the given list of colours. - For example, the output should be of the form ANSWER: yes , no, no, yes and so on.

A.1.6 Comparing Size

Input Prompt:

- This is an image with several rows of circles where each row is above a horizontal line. - The image may also contain a single row with a blue and a green circle. - Each row contains two colored circles (one blue and one green) of different sizes. - Analyse which circle is bigger between the two in each row. - The last line of the output should be of form "ANSWER:" followed by the space separated color answers for all the rows in the image. - For example ANSWER: Blue Green Blue

A.1.7 Count Colored Circles

Input Prompt:

- This is an image containing some colored circles - Count the total number of red circles in the image - Last line of the output must be of the form COUNT:x

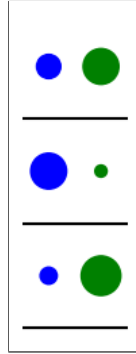


Figure 11: Comparing Size

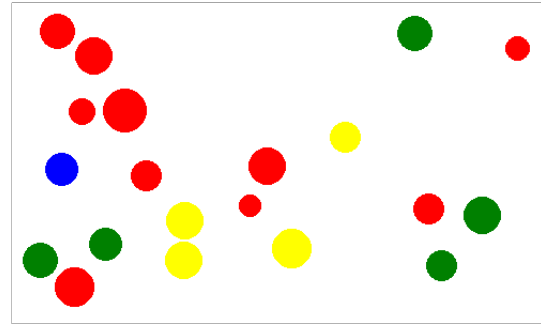


Figure 12: Count Coloured Circles

A.1.8 Counting Circles

Input Prompt:

- This is an image containing some black circles - Count the total number of circles in the image - The last line of the output should be of form COUNT:x.

A.1.9 Counting Locations

Input Prompt:

- There will be an image containing a rectangle and some circles above and below it - Count the number of circles above the rectangle and below the rectangle - The output must be of form in a single line ABOVE:x BELOW:y

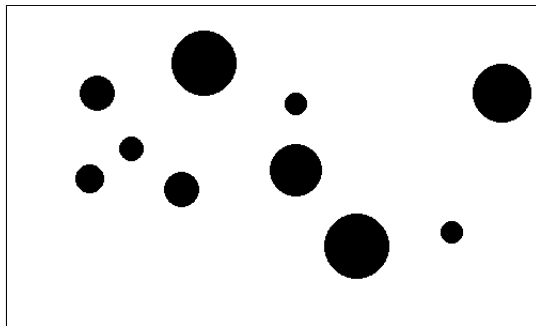


Figure 13: Counting Circles

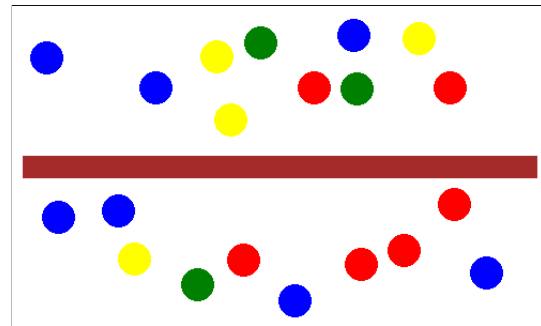


Figure 14: Count Counting Locations

A.1.10 Counting Shapes

Input Prompt:

- This is an image containing shapes such as circles, triangles and squares - Count the number of circles, triangles and squares in the image - The last line of the output must be of the form CIRCLES:x TRIANGLES:y SQUARES:z

A.1.11 Cross and Knots

Input Prompt:

- This is a grid containing some black X. - Each cell is given a coordinate (x,y) which is

(row,column) with 0 based indexing. - An adjacent cell to a cell A is any cell sharing an edge with A - A safe cell is a cell that is not adjacent to any black X. - Output the coordinates of any one safe cell as labelled in the grid. - If no such cell exists output None. - The last line of the output should just contain the cell coordinates (x,y) or None and nothing else.

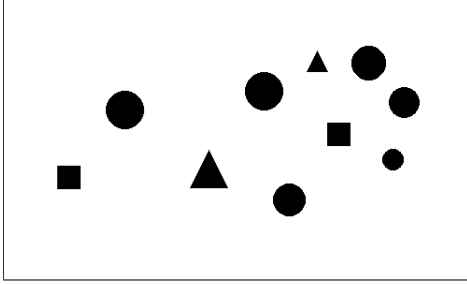


Figure 15: Counting Shapes

(0,0)	X	(0,2)	(0,3)	(0,4)	(0,5)
(1,0)	(1,1)	(1,2)	(1,3)	X	(1,5)
(2,0)	(2,1)	(2,2)	(2,3)	(2,4)	(2,5)
(3,0)	X	(3,2)	(3,3)	X	(3,5)
(4,0)	X	(4,2)	X	(4,4)	(4,5)
(5,0)	X	X	(5,3)	X	X

Figure 16: Cross and Knots

A.1.12 Graph Counting

Input Prompt:

- This is a graph where the blue colored circles are nodes and the lines joining them are edges. - The graph may contain a single blue node with no edges. - The graph may be disconnected. - Count the number of nodes and edges in the graph. - Last line of the output must be of the form NODES:x EDGES:y.

A.1.13 Grid Path

Input Prompt:

- This is a grid which contains some circles, triangles and squares. - Travel from cell labelled S to cell labelled E. - Follow the path marked by the black line and arrows. - Write down the sequence of shapes seen while moving along this path. - The shapes present in the S and E cells should also be included in the sequence. - The last line of the output should be a comma-separated list of shapes without colors, written in order of visiting of form SHAPES : Shape 1 , Shape 2 and so on.

A.1.14 Identifying Shapes

Input Prompt:

- This is an image with several rows of different shapes where each row is above a horizontal line. - Each row contains a coloured shape. - The image may also contain a single shape. - Identify the shape present in each row. - In case there is a single shape, treat it as a single row. - The output should have only a single line containing the answer. - The output format should be "ANSWER:" followed by the space separated shape names. - For example, ANSWER: Circle Triangle Circle

A.1.15 Inside Circles

Input Prompt:

- This is an image containing some black circles and a single red dot - Determine if the red dot is contained inside any circle in the image - Note that a circle containing the red dot means the centre of the dot is present inside the circumference of the circle - The last line of

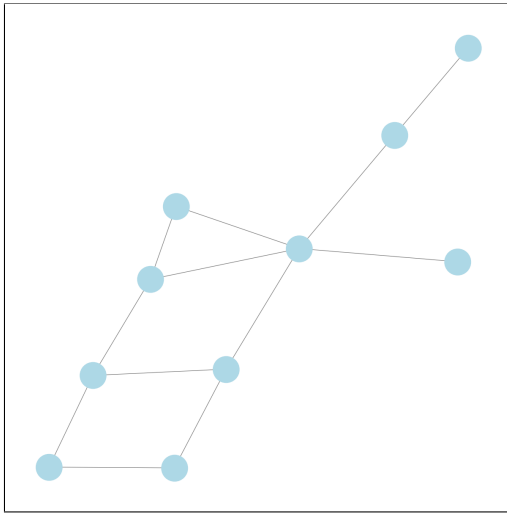


Figure 17: Graph Counting

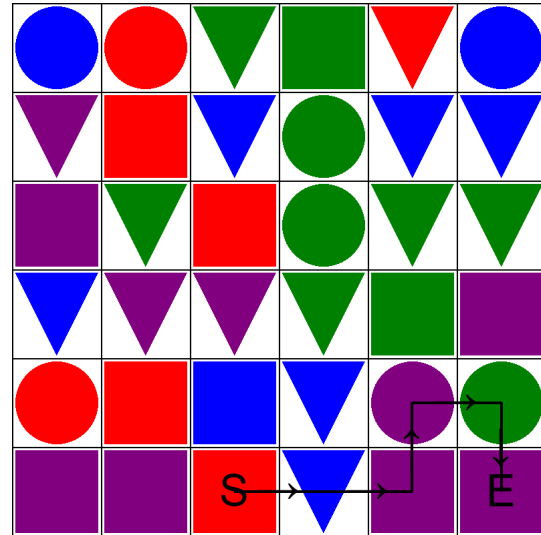


Figure 18: Grid Path

the output should be of form ANSWER: x where x is Yes if the dot is contained in any circle and No otherwise.

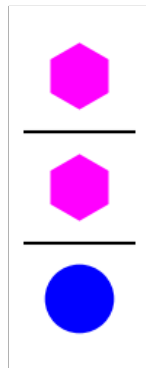


Figure 19: Identifying Shapes

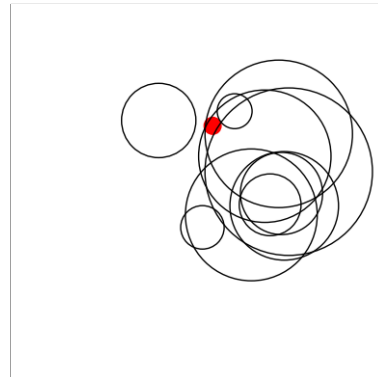


Figure 20: Inside Circles

A.1.16 Layered Colors

Input Prompt:

- This image contains some concentric circles in different colours. - The image may also contain a single circle - Write down the sequence of colours seen from inside to outside. - In case of a single circle in the image, write down the colour of that circle. - The colours belong to the following list - [black, gray, brown, maroon, red, coral, tan, orange, ivory, goldenrod, yellow, green, olive, turquoise, skyblue, blue, lavender, purple, pink, fuchsia] - No colour is repeated - The output should be written in a single line as a comma-separated list of colours, written in order from the innermost colour to the outermost colour - For example, the output should be of the form COLOURS: Colour 1 , Colour 2 and so on.

A.1.17 Layered Shapes

Input Prompt:

- This image contains some shapes that are layered on top of each other in a white background. - The shapes could be of the following types - [circle, diamond, hexagon, octagon]

- The image could also contain a single shape with no other shapes layered on top of it. - Write down the sequence of shapes from the inside layer to the outside layer. - If there is a single shape then identify it. - The last line of the output should be a comma-separated list of shapes without colors, written in order from top to bottom of form SHAPES : Shape 1 , Shape 2 and so on. - If there is a single shape then write SHAPES : shape-name where shape-name is the name of the shape seen

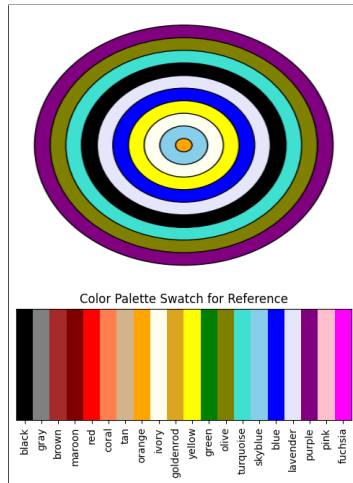


Figure 21: Layered Colours

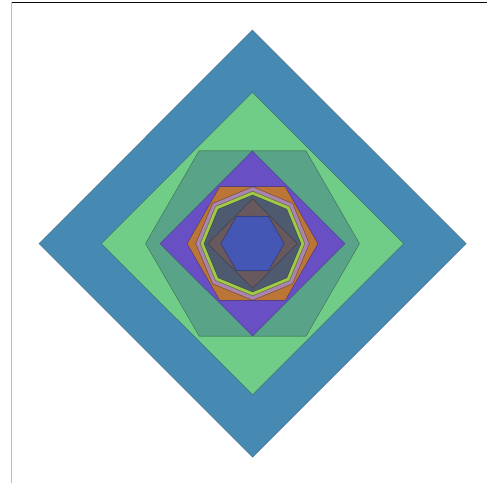


Figure 22: Layered Shapes

A.1.18 List Colors

Input Prompt:

- This is an image with some multi-coloured shapes drawn on top of it - The colours belong to the following list - [black, gray, brown, maroon, red, coral, tan, orange, ivory, goldenrod, yellow, green, olive, turquoise, skyblue, blue, lavender, purple, pink, fuchsia] - No colour is repeated - Identify the colour of the shapes drawn over the background image - Form a list of all the colours of the shapes - The output must be given in a single line in the form of a list of all the colours found

A.1.19 List Shapes

Input Prompt:

- This is an image with some multi-coloured shapes drawn on top of it - The shapes belong to the following list - [circle, triangle, square, pentagon] - Identify the shapes drawn over the background image - Form a set of all the unique shapes seen in the image - The output must be given in a single line in the form of a set of unique shapes found

A.1.20 Locate Circles Color

Input Prompt:

- This is a grid which contains some coloured circles. - The rows and columns are numbered in the grid. - The coordinate of a cell is given by the row number followed by the column number. - Write down all the coordinates of the cells which contain a green circle. - The last line of the output should be a space-separated list of the coordinates of the cells which contain a green circle. - The coordinates should be in the format (row,column) with the comma and bracket

A.1.21 Locate Circles Shape

Input Prompt:

- This is a grid which contains some coloured circles, rectangles and triangles. - The rows

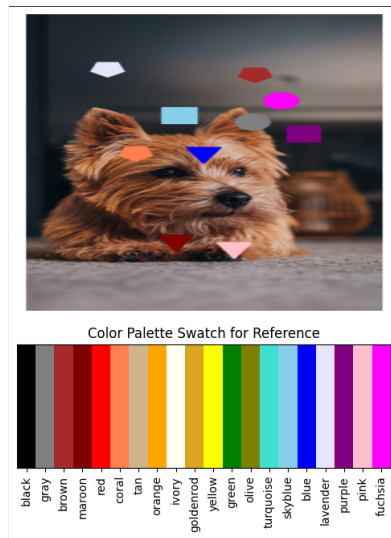


Figure 23: List Colours

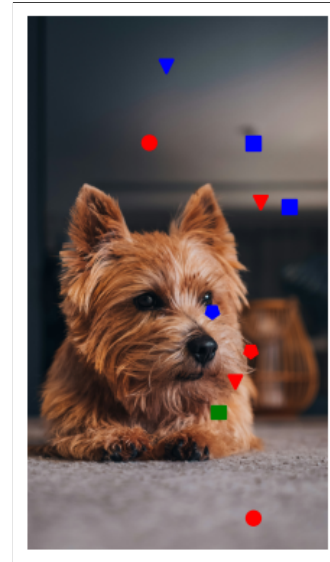


Figure 24: List Shapes

and columns are numbered in the grid. - The coordinate of a cell is given by the row number followed by the column number. - Write down all the coordinates of the cells which contain a green circle. - The last line of the output should be a space-separated list of the coordinates of the cells which contain a green circle. - The coordinates should be in the format (row,column) with the comma and bracket.

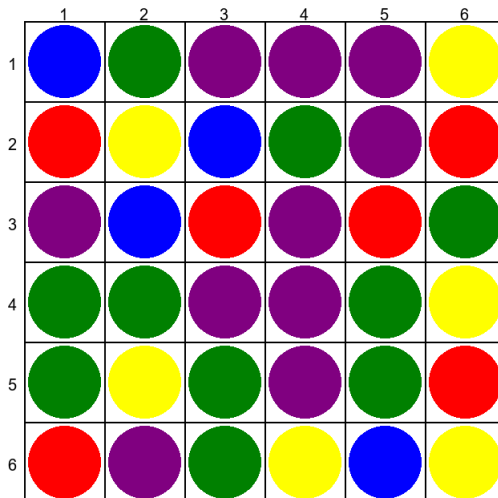


Figure 25: Locate Circles Colour

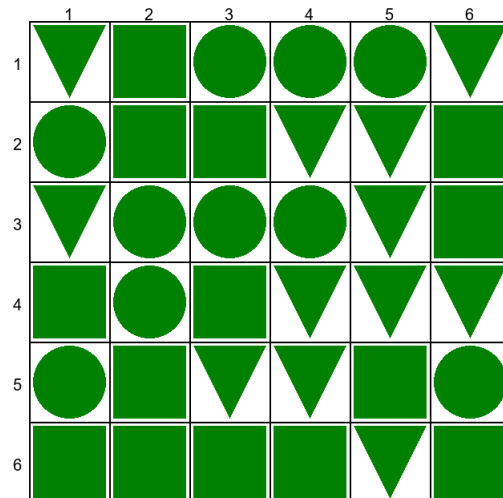


Figure 26: Locate Circles Shapes

A.1.22 Match Outline

Input Prompt:

- These are two images of some sequence of shapes separated by black horizontal lines - The two images may also contain a single shape each - The first image consists of rows of shapes and the second image consists of rows of outlines of shapes - Determine the number of shapes in the first image that match the outlines in the corresponding row in the second image. - In case each of the two images contain only a single shape, treat the single shape in

each image as a row. - The output must be given in a single line in the form of ANSWER:x where x is the number of matching shapes

A.1.23 Match Shadow

Input Prompt:

- These are two images of some sequence of shapes separated by black horizontal lines - The two images may also contain a single shape each - The first image consists of rows of shapes and the second image consists of rows of shadows of those shapes. - Determine the number of shapes in the first image that match the shadows in the corresponding row in the second image. - In case each of the two images contain only a single shape, treat the single shape in each image as a row. - The output must be given in a single line in the form of ANSWER:x where x is the number of matching shadows

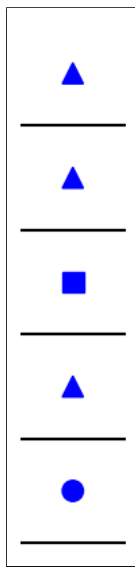


Figure 27: Match Outline 1st Image

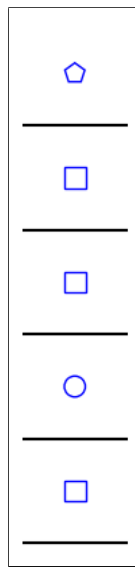


Figure 28: Match Outline 2nd Image

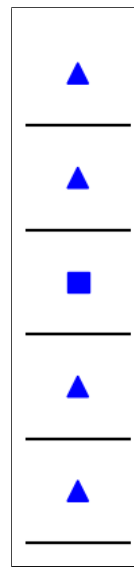


Figure 29: Match Shadow 1st Image

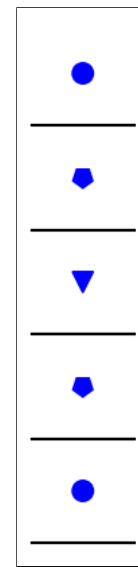


Figure 30: Match Shadow 2nd Image

A.1.24 Maze Solving

Input Prompt:

- This is a maze. - You start from the cell labelled S. - Your goal is to reach cell labelled E through the shortest path possible. - You can move up, down, left or right. - You cannot move through the black walls. - You can only move through white cells. - Give the sequence of cells taken to go from S to E. - Last line of output should be a comma separated line with cell numbers starting with S and ending at E and nothing else.

A.1.25 Mirror Image

Input Prompt:

- These are two images containing different shapes in different colours - The image consists of a vertical blue line at the centre. - Some objects of the first image are mirrored along the given vertical line in the second image. - Other objects in the second image maintain the same positions as in the first image. - Determine the number of objects that are mirrored in the second image. - The output must be given in a single line in the form of COUNT: x, where x is the number of being mirrored

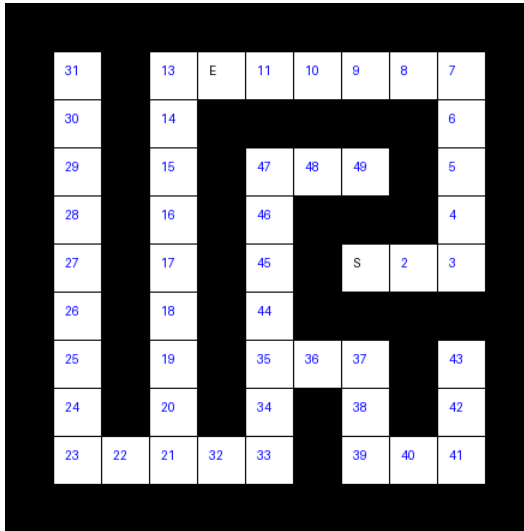


Figure 31: Maze Solving

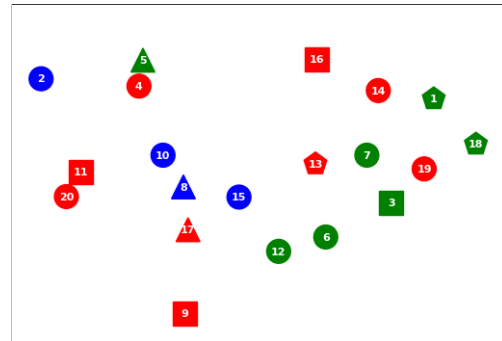


Figure 32: Numbered Shapes

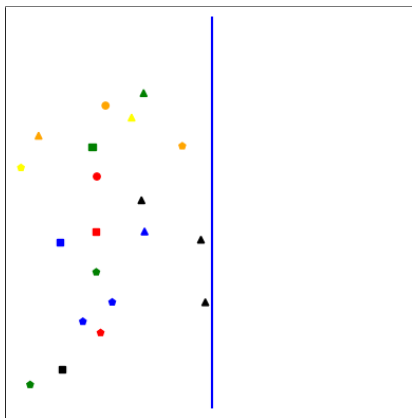


Figure 33: Mirror Image First Image

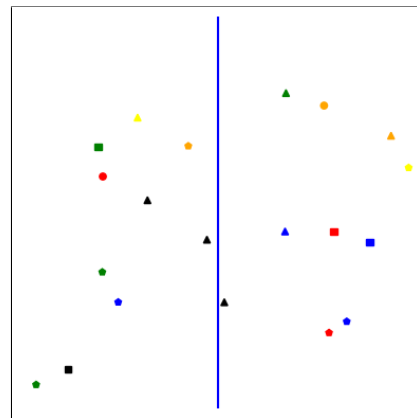


Figure 34: Mirror Image Second Image

A.1.26 *Numbered Images*

Input Prompt:

- This is an image with some shapes drawn and numbers written on top of the shapes - Determine the list of numbers written on top of the circles. - The output must be given in a single line in the form of CIRCLES:x where x is the list of numbers seen on top of the circles

A.1.27 *Sort Circles*

Input Prompt:

- There are several black circles with different sizes. - The image may also have a single black circle. - Sort these circles by size, from smallest to largest. - If there is only a single circles, give the number written on that circle. - Output must only contain the space separated labels of circles in their sorted order by size.

A.1.28 *Sort Lines*

Input Prompt:

- The image has several parallel lines of varying lengths. - The image may also contain only a single line. - Sort these lines by length, from shortest to longest. - If there is only a single

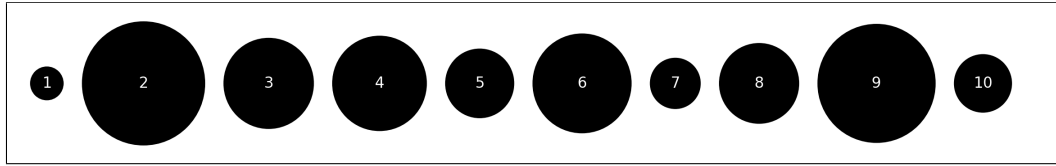


Figure 35: Sort Circles

line, give the number of that line. - Last line of the output must only contain the space separated labels of thee lines in their sorted order by length and nothing else.

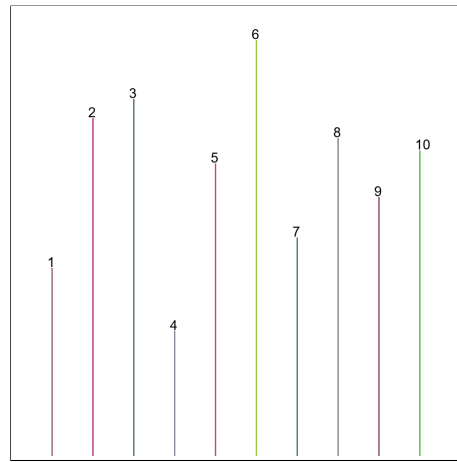


Figure 36: Sort Lines

A.1.29 Vanishing Objects

Input Prompt:

- These are two images containing shapes such as circles, triangles and squares - The images are identical except for the fact that the second image has a few objects missing - Count the number of missing objects in the second image - The output must be given in a single line in the form of COUNT:x

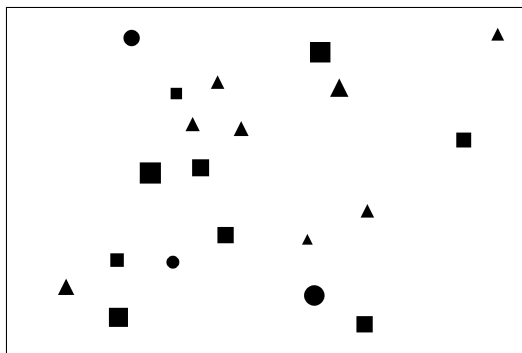


Figure 37: Vanishing Objects 1st Image

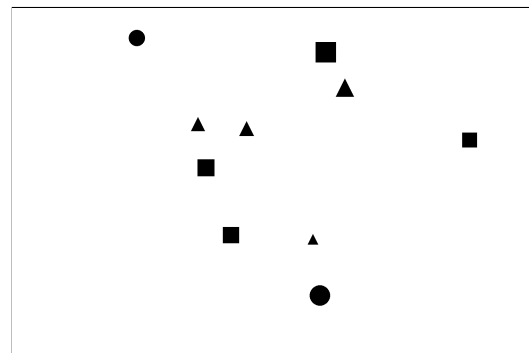


Figure 38: Vanishing Objects 2nd Image

A.1.30 Water Image

Input Prompt:

- These are two images containing different shapes in different colours - The image consists of a horizontal blue line at the centre. - Some objects of the first image are mirrored along the given horizontal line in the second image. - Other objects in the second image maintain the same positions as in the first image. - Determine the number of objects that are mirrored along the horizontal line in the second image. - The output must be given in a single line in the form of COUNT: x, where x is the number of being mirrored

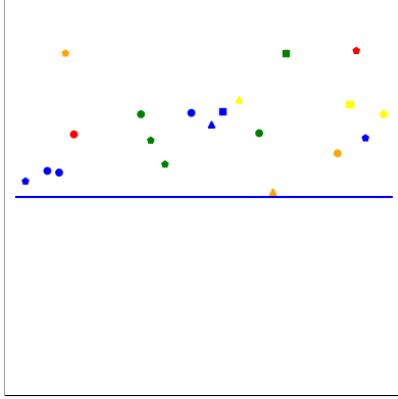


Figure 39: Water Image First Image

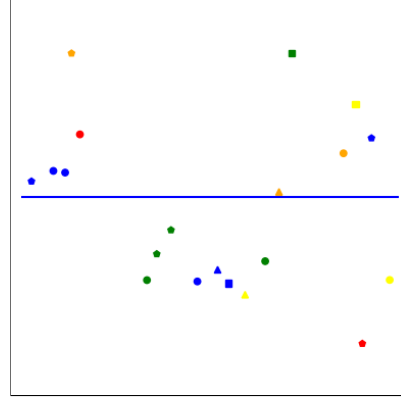


Figure 40: Water Image Second Image

A.2 Performance on Single Skill Domains

Interestingly, we see that the proprietary thinking/reasoning models perform better on 'vsm' as compared to others which indicates that this skill may require higher reasoning capabilities and thus the reasoning models fare better. On the other hand, 'vsr' i.e., Visual Spatial Relationships is an easier skill for models, with almost all of them exhibiting their best performance in it.

A.3 Overall Performance in Human Study

Table 6 shows the overall average accuracies of humans and all the evaluated models

Model	Overall performance
GPT-5-mini	56.81%
GPT-4o	31.94%
o4-mini	56.83%
Gemini	55.58%
Qwen-2.5-VL	14.69%
Deepseek	7.08%
Gemma-4-31B	68.71%
Qwen-3-VL	36.14%
Human	95.43%

Table 6: Overall performance in human study

A.4 One-shot Results

Table 7 presents the results for a subset of the models on Percept-V.

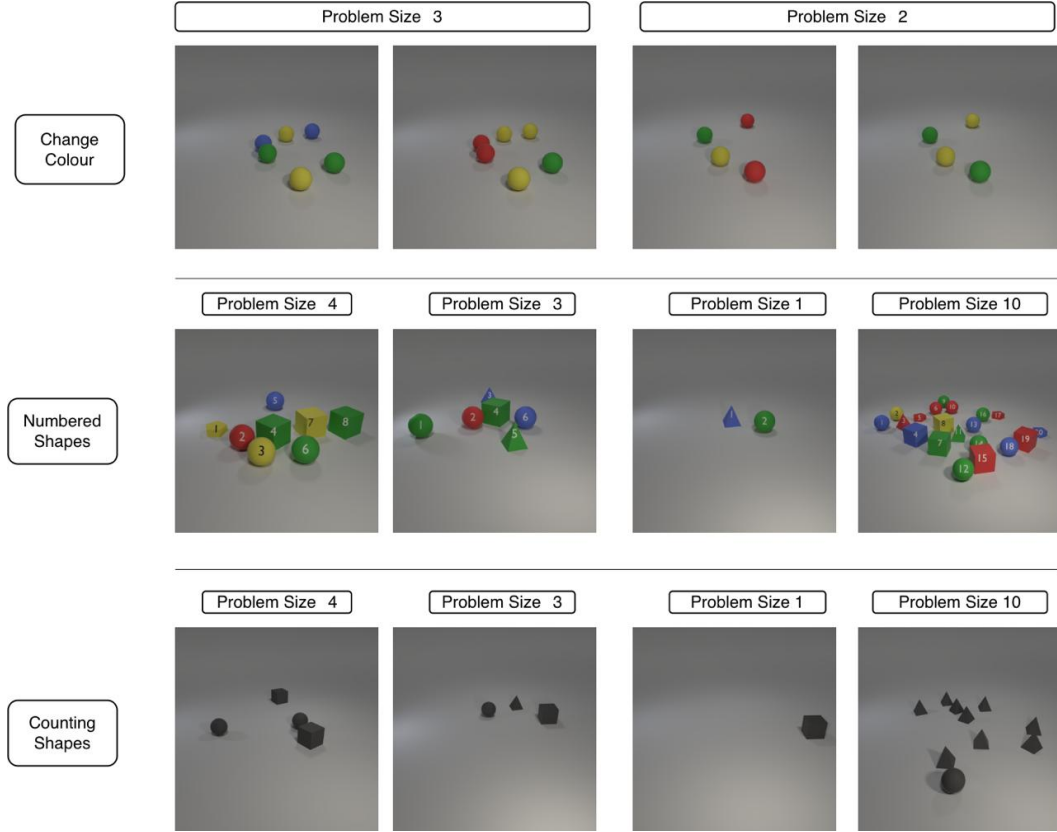


Figure 41: Examples from 3D-Percept-V created via Blender for a subset of domains.

A.5 Answer Type Result

An analysis of performance across problem size categorized by answer types is provided in Figure 44. Upon inspection, it is found that thinking models like GPT-5-mini, Gemini and o4-mini are relatively robust to size variance in unordered set answer type but the same is not true when it comes to ordered list answer type. This reveals that following an order while answering is a challenge for the evaluated models.

A.6 Discussion on Visual Memory

In this section we reflect deeper on how “memory” actually functions in humans and theorize the existence of short-term and long-term memory analogues within the MLLM architectures. A standard human (short-term) memory in TVPS-4 works as follows. The human is shown an image, and then the image is removed and a question is asked. The human’s goal is to convert the image into a mental representation without access to the question, and answer the question based on this representation. We posit that exactly this analogue happens in an MLLM forward pass. The visual encoder pre-processes the image into a question-independent sequence of latent embeddings (tokens). These representations are held in the context window, which effectively acts as the model’s transient short-term memory. The MLLM then uses these tokens to answer the question at hand. Therefore, our experiments essentially test the fidelity and capacity of this short-term representational memory. In this analogy, MLLMs will be considered as using the image (and not their memory) for answering, if the visual representations were computed in a question dependent way, i.e., the model could explicitly use the raw pixels for answering the specific question — this does not happen in a standard MLLM architecture. Conversely, the cognitive definition of “recalling a removed image” can be mapped to an MLLM retrieving visual information

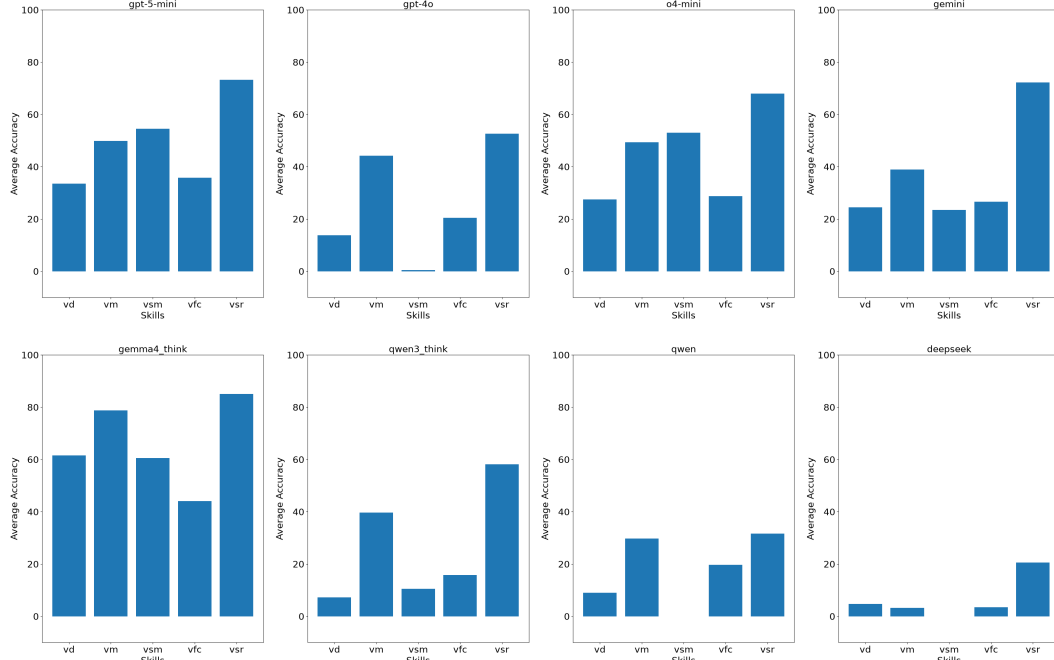


Figure 42: The average accuracies of different models on single skill domains categorised by their skills.

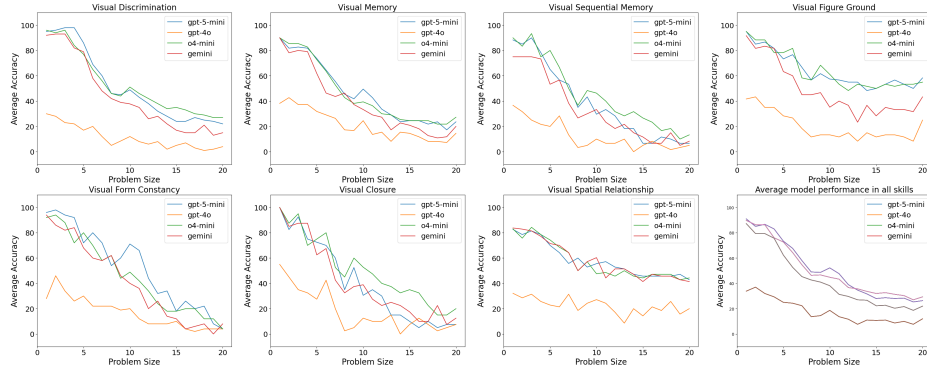


Figure 43: The overall accuracy of all models in different skills in one-shot setting.

stored in its weights during training (long-term memory). To directly test this, we designed a novel experiment to simulate long-term recall. During training phase, we forced the model to encode images into its parametric memory via fine-tuning (Image $\rightarrow$ Unique Text ID). At test time, we removed the image entirely and queried the model using only the ID (Text ID $\rightarrow$ Task Answer). Under this cognitive memory constraint, the model’s performance collapsed to just 1.23%.

Together, these results show that MLLMs fail at both maintaining short-term visual representations and retrieving long-term visual memories.

A.7 Finetuning

All the models were finetuned using qwen-vl’s official codebase. LoRA models were trained with rank 8 and batch-size of 16 on 2xA100 GPUs using $1e - 4$ learning rate with cosine-decay. Full-Finetuning runs were trained using 8xA100 GPUs with $2e - 6$ for 3 epochs.

Skill / Domain	GPT-5-mini	GPT-4o	o4-mini	Gemini	Avg. Acc.
— Performance by Skill —					
Visual Discrimination	51.46	11.17	52.51	43.57	39.68
Visual Memory	46.55	20.93	45.96	39.38	38.2
Visual Sequential Memory	38.61	12.33	45.48	33.42	32.46
Visual Figure Ground	63.9	20.0	63.9	48.83	49.16
Visual Form Constancy	53.06	16.64	45.63	40.7	39.01
Visual Closure	40.58	17.75	50.76	40.96	37.51
Visual Spatial Relationship	64.61	24.5	61.59	64.11	53.7
Average of skills	51.25	17.62	52.26	44.42	41.39
— Performance by Domain —					
change.colour	30.15	11.5	24.62	21.11	21.84
circle_boxes	26.63	21.5	28.64	24.12	25.22
circle_location	55.78	13.0	48.24	54.27	42.82
circle_right_triangle	100.0	59.0	100.0	96.48	88.87
colours_present	4.02	1.5	5.03	1.01	2.89
comparing_size	17.59	2.5	23.62	36.18	19.97
count.coloured_circles	41.71	16.5	35.68	46.73	35.16
counting_circles	57.79	23.5	48.74	53.27	45.82
counting_locations	35.68	9.0	35.18	28.64	27.12
counting_shapes	60.8	20.2	50.75	56.28	47.01
cross_and_knots	100.0	21.0	100.0	91.46	78.11
graph.counting	33.17	3.0	27.14	31.16	23.62
grid_path	51.76	0.5	46.23	0.5	24.75
identifying_shapes	67.84	41.5	91.46	51.76	63.14
inside_circles	94.97	52.0	92.96	95.48	83.85
layered.colours	31.16	25.0	34.17	24.62	28.74
layered_shapes	21.11	12.5	19.1	19.1	17.95
list.colours	40.7	12.0	40.7	36.18	32.4
list_shapes	91.96	68.5	91.96	82.91	83.83
locate_circles.colour	99.5	1.0	98.49	63.32	65.58
locate_circles_shape	98.9	1.0	98.99	66.83	66.45
match_outline	58.79	17.5	79.4	71.86	56.89
match_shadow	51.26	16.0	70.35	48.24	46.46
maze_solving	48.24	15.0	31.16	46.23	35.16
mirror_image	30.15	17.5	27.14	20.1	23.72
numbered_shapes	81.91	9.5	77.39	53.77	55.64
sort_circles	34.67	16.5	33.17	25.63	27.49
sort_lines	28.64	19.5	23.62	23.62	23.85
vanishing_objects	34.17	6.0	22.11	22.61	21.22
water_image	34.67	12.5	24.12	20.1	22.85
Full Percept-V Dataset	52.13	18.21	51.01	43.79	41.28

Table 7: MLLM performance across different skills and domains in one-shot setting (in %).

Skill / Domain	GPT-5-mini Concise	Qwen Concise	GPT-5-mini Verbose	Qwen Verbose
Visual Discrimination	55.3	13.91	53.55	13.45
Visual Memory	47.55	21.87	49.27	18.77
Visual Sequential Memory	47.5	5.5	39.83	7.25
Visual Figure Ground	63.25	4.33	63.42	5.08
Visual Form Constancy	48.9	24.31	58.5	30.1
Visual Closure	50.25	5.38	39.62	10.62
Visual Spatial Relationship	62.64	18.07	66.29	15.5
Average of skills	53.63	13.34	52.93	14.4

Table 8: Comparison of models’ performance across different prompting styles (in %).

A.8 Instructions for Participants

Thank you for participating in this experiment! This questionnaire will test your visual perception skills through some simple questions. Please follow the below instructions carefully - 1. Focus on accuracy and not on speed. There is no time bound , so you can take breaks while solving. However, you have to keep the website open on some tab/window

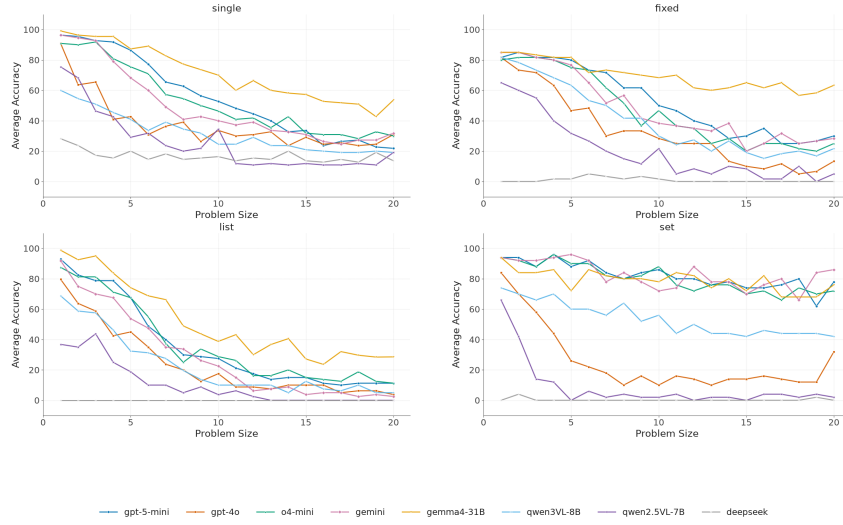


Figure 44: The overall accuracy of all models in different answer types.

Domain	GPT-5-mini		Qwen	
	2D	3D	2D	3D
change_colour	64.0	58.0	16.0	15.0
colours_present	1.0	9.0	0.0	0.0
count_coloured_circles	90.0	78.0	62.0	62.0
counting_circles	93.0	74.0	78.0	92.0
counting_shapes	95.0	94.0	67.0	81.0
numbered_shapes	98.0	19.0	16.0	17.0
Average (6 domains)	73.50	55.33	39.8	44.5

Table 9: Model accuracy (%) on six Percept-V domains rendered in 2D and in 3D, matched on problem sizes 1–10 (100 instances per domain in each condition).

otherwise, your progress will be lost. Also, take sufficient time to solve each question. 2. Please open this questionnaire on laptop/tablet. Some fonts are small, so, you may miss out if you open on smaller devices. Zoom in on images that appear tiny otherwise, this may affect your accuracy! 3. Each question follows a particular answer format. Follow the answer format strictly! Check for spelling mistakes while writing. 4. Try to attempt all the questions. If you do not understand any question, reach out to us! All the best!

A.9 Reproducibility

A.9.1 Image Generation

Image is generated using hyperparameters like `–num_image` that give the number of images to be generated per problem size and `–num_size` that give the list of problem sizes.

A.9.2 GPU Hours

GPU Hours required were around 5 for Deepseek (1B parameters) and Qwen(7B parameters). The number of parameters for o4-mini and GPT-4o are estimated to be 8B and 200B.

A.10 Limitations

This paper evaluates MLLMs only against a limited pool of adult college students. Future work may be expanded into including annotators from different age groups and diverse educational background to improve the robustness of the results. Additionally, this paper

only identifies the weakness of MLLMs in visual perceptual skills and does not investigate in detail, the rationale behind such weakness.