THE CARTESIAN SHORTCUT: RE-EVALUATE VISION REASONING IN POLAR COORDINATE SPACE

Anonymous authors
Paper under double-blind review

ABSTRACT

As current Multimodal Large Language Models rapidly saturate canonical visual reasoning benchmarks, a key question emerges: do these strong scores genuinely reflect robust visual understanding? We identify a pervasive vulnerability, the **Cartesian Shortcut**: models frequently discretize the orthogonal grid-based layouts prevalent in visual reasoning benchmarks into explicit textual coordinates, offloading reasoning from visual perception to text-based deduction. To re-evaluate visual reasoning when this shortcut is unavailable, we introduce **Polaris-Bench**, which re-formulates 53 visual reasoning tasks in Polar coordinate space with paired Cartesian counterparts that preserve task semantics, disrupting the orthogonal structure that models exploit. Comprehensive evaluation across 14 state-of-the-art MLLMs reveals that frontier models achieving 69–83% on Cartesian layouts collapse to 31–39% on Polar equivalents. Moreover, thinking gains largely vanish on Polar layouts, prompting interventions fail to close the gap, and comparable drops arise on other non-orthogonal layouts. These findings reveal that current MLLMs' visual reasoning performance is strongly coupled to orthogonal grid structure, a fragility that is consistent across model families but far smaller in humans, who maintain 88.8% accuracy on Polar layouts.

1 INTRODUCTION

Recent Multimodal Large Language Models (MLLMs) (Gemini Team, 2025; Singh et al., 2025; Anthropic, 2024) achieve remarkable, often human-level performance across canonical visual reasoning benchmarks (Yue et al., 2025; Chen et al., 2025; Hao et al., 2025; Tong et al., 2024a). However, as these datasets rapidly saturate, a critical question emerges: have models genuinely acquired visual reasoning abilities, or are they merely exploiting text-based heuristics (Asadi et al., 2026; Xu et al., 2025b; Chen et al., 2024; Kanade & Ganu, 2026)? In this study, we expose a pervasive systemic vulnerability that creates an illusion of profound visual capability: the **Cartesian Shortcut**.

Orthogonal grid-based layouts, which underlie many prominent visual reasoning benchmarks, can be readily discretized into explicit textual coordinates. We find that MLLMs systematically exploit this property: across a synthetic subset of 9 prominent benchmarks (e.g., Hao et al. (2025); Jia et al. (2026)), frontier model explicitly invoke Cartesian coordinates (e.g., "row 2", "(x,y)") in over 56% of intermediate Chain-of-Thought (Wei et al., 2022) reasoning, converting visual layouts into textual representations to assist problem-solving. By substantially offloading reasoning from visual perception to text-based deduction, this shortcut significantly confounds the evaluation of visual reasoning on grid-based testbeds.

Crucially, performance drops sharply once the orthogonal layout is distorted: on a logically identical Sudoku puzzle rendered in Polar coordinates, frontier models that correctly solve the Cartesian version fail on its coordinate equivalent (Figure 1). Drawing on this observation, we introduce **Polaris-Bench**, comprising 53 procedurally generated visual reasoning tasks in Polar coordinate space, each accompanied by a Cartesian counterpart as controlled reference under consistent logical constraints. The Cartesian-to-Polar coordinate transformation breaks the orthogonal structure underlying the Cartesian shortcut, disrupting models' ability to easily discretize visual layouts into textual coordinates. This design provides a controlled diagnostic testbed to re-evaluate visual reasoning under coordinate transformation, with the consistent logical constraints enabling direct cross-topology evaluation and analysis.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

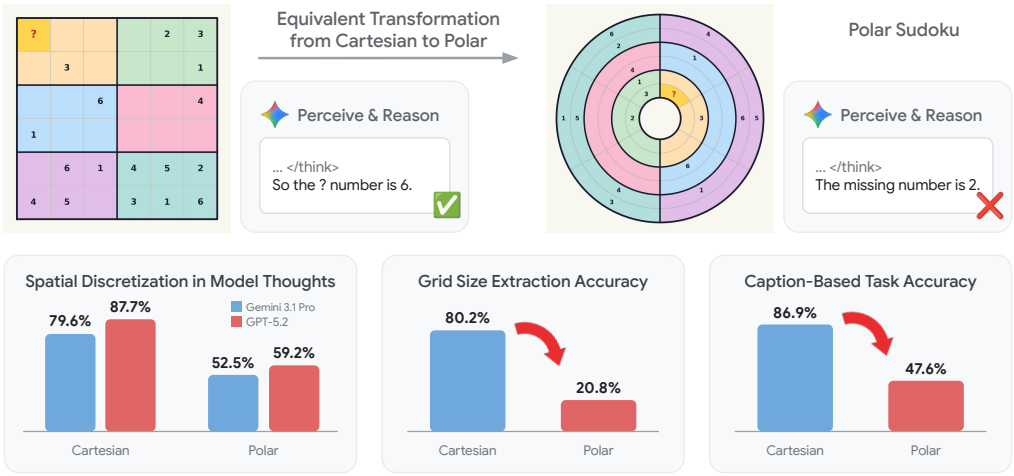


Figure 1: Illustration and empirical evidence of the Cartesian Shortcut. *(Top)* An equivalent sudoku question in Cartesian and Polar spaces (row to ring, column to angular sector). The model correctly solves the orthogonal layout but fails on its polar counterpart. *(Bottom Left)* The percentage of intermediate CoT that explicitly invokes textual coordinates. Gemini-3.1-Pro and GPT-5.2 both exhibit high frequencies on Cartesian layouts. *(Bottom Middle & Right)* Two-stage perception and reasoning analysis on Gemini-3.1-Pro: grid size extraction accuracy estimates visual perception capability (*Middle*), while task accuracy is evaluated on correctly perceived instances (*Right*). High Cartesian accuracy indicates the effectiveness of the shortcut, while the sharp decline on Polar layouts is consistent with the corruption of orthogonal structure.

We comprehensively evaluate Polaris-Bench across 14 state-of-the-art MLLMs spanning both proprietary and open-source families. The results reveal a striking and consistent pattern that frontier models achieving 69–83% accuracy on Cartesian layouts see their performance collapse to 31–39% on Polar equivalents, with the largest degradation approaching 47 points. Comparatively, human performance drop only 5.7 points on Polar layout. Critically, the degradation persists even on tasks where the transformation preserves complete logical equivalence (i.e., identical rules, constraints, and ground truths), confirming that the coordinate transformation alone suffices to disrupt model reasoning. Moreover, while enabling high reasoning mode yields substantial accuracy gains on Cartesian layouts, these gains are severely diminished on Polar equivalents. Further analyses show that comparable drops arise on non-Polar layouts (hexagonal tilings, wave and arc deformations), and the gap could not be recovered by prompting intervention (conversion hints, in-context examples).

Our main contributions are:

- We identify and empirically validate the Cartesian Shortcut, whereby MLLMs systematically discretize orthogonal grid-based layouts into explicit textual coordinates, offloading visual reasoning onto text-based deduction and confounding existing visual reasoning evaluation.
- We introduce Polaris-Bench, which re-formulates 53 visual reasoning tasks in Polar coordinate space to systematically disrupt the Cartesian Shortcut, with paired Cartesian counterparts enabling cross-topology evaluation.
- Our comprehensive evaluation across 14 state-of-the-art MLLMs reveals drastic and consistent performance collapse on Polar equivalents, with degradation persisting across diverse model families, scales, and reasoning configurations, while remaining far smaller in human participants.

2 Rethinking Vision Reasoning: The Cartesian Shortcut

Current MLLMs excel on vision reasoning benchmarks, where many of these tasks are built upon synthetic, grid-based layouts (Tang & Kejriwal, 2025; Ren et al., 2025). To investigate how models approach them, we analyzed over 3,800 synthetic-based questions across 9 prominent visual reasoning benchmarks (e.g., Hao et al. (2025); Jia et al. (2026)). Our Chain-of-Thought (CoT) (Wei et al., 2022) analysis reveals a striking pattern: in over 56% examples, frontier models (e.g., Gemini-3-Flash)

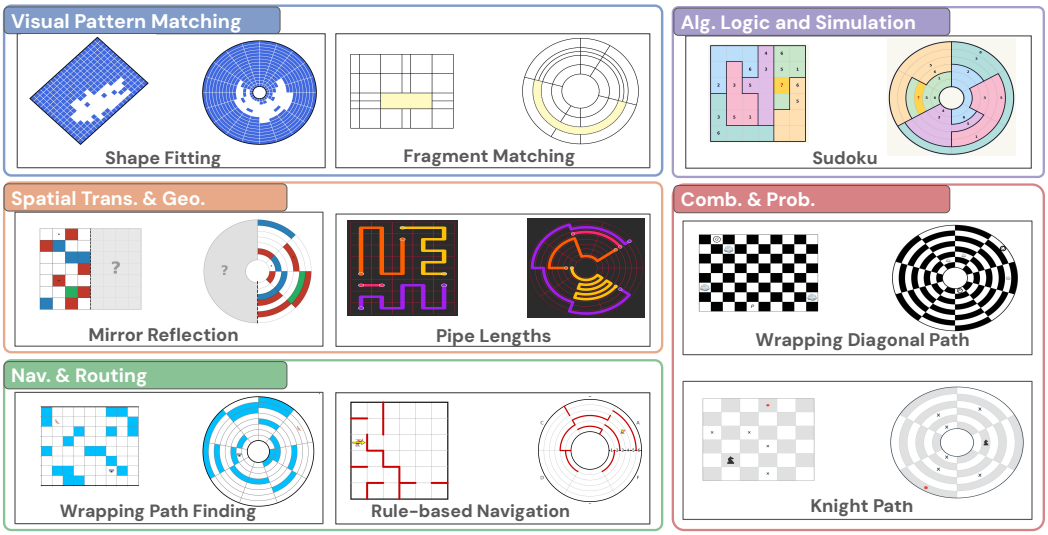


Figure 2: Representative examples for each of the five core taxonomies in Polaris-Bench.

explicitly discretize the input image into Cartesian coordinates, generating text such as "*starting at (2,3), moving to (3,5).*" By translating these visual layouts into discrete matrices, models heavily offload visual logic onto text-based deductive reasoning, reducing the need for persistent visual grounding. We define this phenomenon as **Cartesian Shortcut**. While this behavior demonstrates robust deductive capability, it exposes a critical flaw in current evaluations: models can leverage textual discretization to assist visual reasoning, projecting an inflated estimate of visual capability.

To further empirically validate this Cartesian Shortcut, we analyzed model behavior on our paired Cartesian-Polar benchmark. Intermediate CoT from frontier models reveal a deep dependency on textual discretization, explicitly invoking Cartesian coordinates in 80%-88% of Cartesian examples, and still attempting to in over half of Polar ones (Figure 1(left)). We further probed this dependency by a two-stage experiment on Gemini-3.1-Pro (Gemini Team, 2025). The model first generates a descriptive caption, then solves the task based on the caption. As shown in Figure 1(middle, right), the model achieves 80.2% accuracy on visual layout perception, estimated via grid dimension extraction from model-generated captions, and task accuracy on the subset of correctly perceived instances reaches 86.9%. This indicates that Cartesian Shortcut is not merely a behavioral pattern but a highly effective problem-solving strategy for MLLMs. This effectiveness raises a fundamental concern: benchmarks built on orthogonal grids may systematically overestimate visual reasoning capabilities.

Drawing on the duality of coordinate systems (Arfken et al., 2011; Zetzsche et al., 1999), we introduce Polaris-Bench to dismantle the Cartesian shortcut. Polaris-Bench re-formulates visual reasoning tasks in Polar coordinate space, pairing each instance with a Cartesian counterpart as controlled reference. This mapping is a visual and coordinate transformation that preserves the underlying logical constraints, rather than a mathematical translation. The coordinate transformation disrupts the orthogonal grid structure that enables the Cartesian Shortcut. Applying the same two-stage analysis to Polar layouts reveals a stark contrast: visual layout perception accuracy falls to 20.8%, and task accuracy on correctly perceived instances to 47.6% (Figure 1). This is consistent with the observation: on Polar layouts, models can neither reliably discretize the grid into text nor maintain high accuracy.

3 Polaris-Bench

3.1 Benchmark Overview

Polaris-Bench comprises 53 procedurally generated visual reasoning tasks, each re-formulated in Polar coordinate space with paired Cartesian counterparts as controlled reference, yielding over 10,600 core test cases. This paired design is central to our methodology: by presenting each task in two coordinate systems that share identical rules, question context, and identical ground truths for the

majority, we construct a controlled diagnostic testbed where performance differences directly measure sensitivity to the coordinate transformation. To ensure valid spatial logic across coordinate systems, physical concepts are contextually adapted (e.g., Cartesian grid distances are translated to unit edge counting in Polar space). All task outputs are constrained to deterministic formats (multiple-choice options, exact digits, coordinates, string, and lists) to facilitate rigorous automated evaluation.

Topology Alignment. The paired Cartesian-Polar evaluation introduces a design constraint: the angular dimension of Polar coordinates is inherently cyclic, with 360 wrapping seamlessly to 0. This cylindrical connectivity can alter valid solutions for adjacency-sensitive tasks, creating a potential confound between perceptual distortion and genuine topological divergence.

We address this through two boundary conditions. Under the *bounded* condition, boundary rings and radial seams act as impenetrable barriers, producing a closed Polar grid coordinate equivalent to its Cartesian counterpart with identical ground truths. Under the *wrapping* condition, we preserve the natural cyclic connectivity of Polar space, retaining its intrinsic topological properties as an additional dimension of the evaluation; for a subset of these tasks, we additionally introduce cyclic boundaries in the Cartesian version (e.g., exit left will enter right) to both align ground truths and examine whether models can handle wrapping within standard grid layouts. This yields two alignment categories: the large majority are *fully aligned* where coordinate transformation is the only variance, while a deliberately preserved subset remains *partially aligned*, where cylindrical connectivity alters the solution space. This stratification enables controlled ablation: gaps on fully aligned tasks reflect the coordinate transformation itself, with rules and ground truths held fixed; while gaps on partially aligned tasks additionally capture failures under altered topology.

Task Taxonomy. To evaluate vision reasoning across diverse cognitive dimensions, we organize the 53 tasks into five categories based on their core cognitive operation (Figure 3). *Visual Pattern Matching* (11 tasks) evaluates pattern recognition and structural discrimination within organized visual layouts. *Spatial Transformation and Geometry* (13 tasks) evaluates geometric transformation reasoning (e.g., rotation, reflection, folding) and quantitative property inference. *Navigation and Routing* (14 tasks) evaluates sequential positional state tracking, requiring models to execute or trace multi-step movements under environmental constraints. *Combinatorics and Probability* (9 tasks) evaluates systematic combinatorial enumeration and probabilistic reasoning. *Algorithmic Logic and Simulation* (6 tasks) evaluates constraint satisfaction and deterministic forward simulation. This categorization enables fine-grained analysis of how the Cartesian Shortcut differentially affects distinct reasoning capabilities.

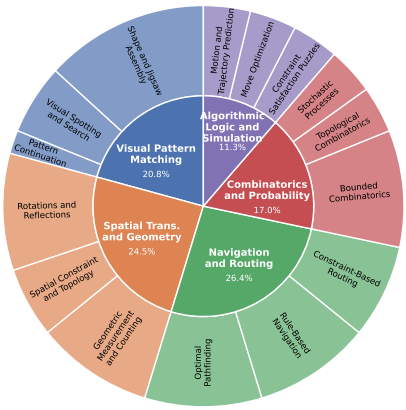


Figure 3: Distribution of 53 tasks across the five core categories.

3.2 Benchmark Construction

Given the scarcity of Polar tasks in the wild, we systematically generated our benchmark data instead of collecting it manually. Specifically, we synthesize Polaris-Bench through a four-stage pipeline illustrated in Figure 4: task curation from existing benchmarks, task design with cross-topology alignment, example generation, and multi-stage quality assurance. Key facts are provided below with comprehensive details in Appendix.

Task Curation. We first surveyed prominent visual reasoning benchmarks (e.g., Yue et al. (2025); Lu et al. (2024); Hao et al. (2025)) to identify tasks that heavily rely on grid-based spatial logic. Using a combination of LLM-assisted filtering and manual review, we selected tasks whose core reasoning logic can be faithfully re-formulated across different coordinate topologies(Section 3.1).

Task Design and Cross-Topology Alignment. For each curated task, we designed paired Cartesian and Polar implementations that preserve equivalent evaluation conditions. This involves three interrelated design considerations. First, *logical alignment*: for each task, we analyzed and designed the wrapping behavior and topology alignment to ensure consistent evaluation across coordinate systems (Section 3.1). Second, *visual calibration*: since the Polar mapping introduces nonlinear distortion, we calibrate the rendering of each task so that size-sensitive visual elements remain as

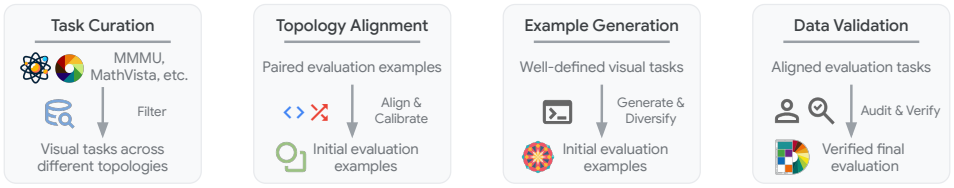


Figure 4: An overview of the construction process of the Polaris-Bench.

comparable as possible across the two layouts, ensuring that performance differences stem from the coordinate transformation rather than perceptual disadvantage. Third, *narrative design*: beyond adapting existing task formats, we substantially redesigned task rules, visual presentations, and problem structures to introduce richer reasoning demands, and contextualized them within diverse textual scenarios (e.g., robotic navigation, physics simulations).

Procedural Generation. Each task is implemented as a self-contained generative script capable of producing unlimited unique instances. We heavily randomize both visual and logical axes (e.g., grid dimensions, starting positions, path topologies, distractors, color palettes), while enforcing programmatic solvability checks to prevent degenerate configurations. Each task contributes 100 paired Cartesian–Polar instances to the final benchmark.

Quality Assurance and Human Validation. We apply a multi-stage verification pipeline to ensure the correctness and clarity of generated data. (1) We iteratively refined the rendering to ensure visual clarity and legibility across both Cartesian and Polar layouts. (2) The authors verified task logic through collaborative review and by independently spot-checking and solving sampled instances of every tasks. (3) We employed LLM-assisted audits of the procedural code to identify unhandled edge cases. Steps (1)–(3) were repeated until no further issues identified. (4) As a final safeguard, we conducted a human validation study with independent raters manually solved 20 randomly sampled Cartesian–Polar pairs per task, confirming the correctness and quality of the final benchmark data.

4 Experiments

4.1 Experimental setup

Evaluated Models. To comprehensively evaluate visual reasoning capabilities, we test a diverse suite of 14 state-of-the-art MLLMs spanning both proprietary and open-source families. For proprietary models, we include the *Gemini family* (Gemini Team, 2025), *GPT-5.2* (Singh et al., 2025), *Claude-Sonnet-4.6* (Anthropic, 2024), and *Grok-4 family* (xAI, 2025). For open-source models, we evaluate *Gemma-4-31B*, *Gemma-4-26B* (DeepMind, 2026), *Kimi-k2.5* (Team et al., 2026), *Qwen-3.5-397B-A17B* (Qwen Team, 2026), and *Mistral-Small-3.1-24B* (AI, 2025). This diverse model selection spans different model families at varying architectures, scales, and training recipes. All models are accessed via public APIs with default parameter settings and maximum supported context length.

Reasoning Modes. For the subset of models supporting configurable reasoning, we evaluate under two inference settings: (1) a *high reasoning* mode that activates the model's highest supported Chain-of-Thought setting; and (2) a *non-reasoning* mode that uses the minimal reasoning configuration.

Evaluation Metrics. Since ground-truth answers in Polaris-Bench are deterministic and span well-defined formats (option labels, digits, coordinate positions, strings and lists), we employ exact-match accuracy as our primary metric. For the responses whose format deviates from the expected structure, we apply LLM-as-a-judge (i.e., Gemini-3.1-Pro) as a calibration fallback to ensure fair evaluation (Zheng et al., 2023; Gu et al., 2024). To contextualize model performance, we additionally report a human performance reference alongside a theoretical random baseline. All models and tasks share the same prompt templates, with full setup details provided in Appendix.

4.2 Main results

The main results in Table 1 report overall and per-category accuracy for all evaluated models under high reasoning mode, comparing Cartesian (C) and Polar (P) layouts. Figure 5 examines topology

Table 1: Main results on Polaris-Bench. All models are evaluated under *high* reasoning mode. C and P denote Cartesian and Polar accuracy(%). Accuracy is averaged over tasks in each category. ($-\Delta$) indicates the C→P accuracy drop. The **highest** and second-highest scores are highlighted.

Model	Algorithmic Logic & Simulation		Combinatorics & Probability		Navigation & Routing		Spatial Trans. & Geometry		Visual Pattern Matching		Overall	
	C	P ($-\Delta$)	C	P ($-\Delta$)	C	P ($-\Delta$)	C	P ($-\Delta$)	C	P ($-\Delta$)	C	P ($-\Delta$)
Human	92.5	87.5 (−5.0)	77.8	64.4 (−13.4)	97.5	92.5 (−5.0)	98.8	94.6 (−4.2)	100.0	97.3 (−2.7)	94.5	88.8 (−5.7)
Random	17.0	17.0	6.7	6.7	18.4	18.4	16.2	16.2	19.0	19.0	15.8	15.8
Close Source MLLMs												
Gemini-3.1-Pro	**85.0**	**52.6** (−32.4)	**92.4**	24.9 (−67.6)	**83.4**	39.4 (−44.0)	**83.7**	31.4 (−52.3)	**71.0**	**36.6** (−34.4)	**82.6**	35.9 (−46.7)
Gemini-3-Flash	68.8	41.5 (−27.3)	74.8	20.6 (−54.2)	76.2	41.9 (−34.4)	68.7	29.8 (−38.8)	65.0	34.8 (−30.2)	71.0	33.8 (−37.2)
Gemini-3-Flash-lite	58.0	28.3 (−29.7)	40.4	15.9 (−24.6)	54.6	28.8 (−25.8)	40.6	21.7 (−18.9)	43.3	27.7 (−15.5)	46.8	24.6 (−22.2)
Gemini-2.5-Pro	54.3	30.2 (−24.2)	23.3	12.7 (−10.7)	45.6	31.1 (−14.5)	33.8	23.9 (−9.8)	38.4	27.0 (−11.4)	38.4	25.3 (−13.2)
Gemini-2.5-Flash	51.3	27.7 (−23.7)	15.9	9.4 (−6.4)	38.7	26.8 (−11.9)	26.4	18.5 (−7.9)	33.8	22.7 (−11.1)	32.2	21.1 (−11.2)
Gpt-5.2	70.7	46.7 (−24.0)	84.8	**30.7** (−54.1)	82.2	**42.8** (−39.4)	76.5	**41.6** (−34.9)	70.0	34.5 (−35.5)	77.4	**39.2** (−38.2)
Claude-Sonnet-4.6	60.0	30.3 (−29.7)	42.7	16.8 (−25.9)	45.0	28.5 (−16.4)	42.3	25.2 (−17.0)	38.9	28.3 (−10.6)	44.4	25.9 (−18.5)
Grok-4-0709	53.2	32.8 (−20.4)	20.8	12.0 (−8.8)	38.0	24.9 (−13.1)	29.9	22.2 (−7.7)	29.5	19.6 (−9.9)	33.0	21.8 (−11.2)
Grok-4-Fast-Reasoning	49.7	30.8 (−18.8)	17.7	12.3 (−5.3)	37.7	27.1 (−10.6)	28.4	21.5 (−6.8)	26.1	20.5 (−5.5)	31.0	22.3 (−8.7)
Open Source MLLMs												
Qwen3.5-397B-A17B	72.2	**43.2** (−29.0)	65.2	18.1 (−47.1)	**77.6**	**40.9** (−36.7)	**77.2**	**39.5** (−37.8)	**68.5**	31.5 (−37.0)	**72.9**	**35.0** (−37.9)
Kimi-k2.5	**73.5**	37.3 (−36.1)	**68.7**	16.2 (−52.5)	73.0	38.3 (−34.7)	71.7	31.1 (−40.6)	58.4	30.8 (−27.5)	69.0	31.1 (−37.8)
Gemma-4-31B	67.7	36.0 (−31.7)	62.3	**19.4** (−42.8)	63.9	37.6 (−26.2)	60.1	28.5 (−31.6)	51.5	**32.5** (−19.1)	60.5	31.0 (−29.5)
Gemma-4-26B	56.0	29.5 (−26.5)	45.8	14.8 (−30.9)	48.9	26.5 (−22.4)	48.5	19.4 (−29.2)	40.0	25.4 (−14.6)	47.2	22.9 (−24.4)
Mistral-Small-2503	25.0	20.7 (−4.4)	9.6	10.9 (1.3)	22.6	21.2 (−1.4)	19.5	21.2 (1.7)	20.3	19.4 (−0.9)	19.4	19.0 (−0.4)

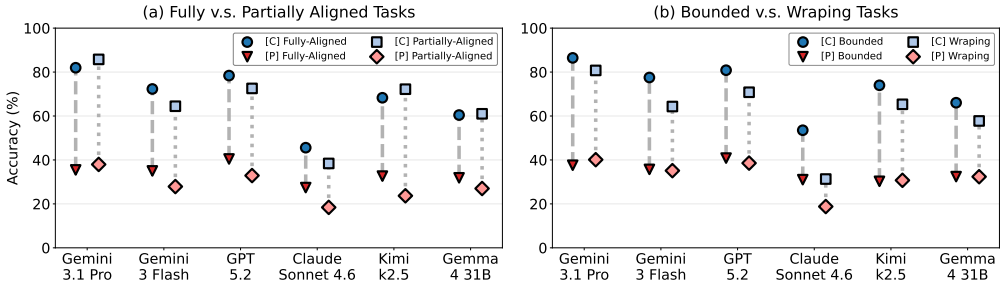


Figure 5: Cartesian (C) and Polar (P) accuracy under two partitions of the 53 tasks: *(a)* fully vs. partially aligned; *(b)* bounded vs. wrapping. The C→P drop persists in every subset; wrapping mainly lowers Cartesian accuracy.

alignment conditions. Table 2 reports reasoning vs. non-reasoning inference for configurable models. Figure 6 visualize the comparison in Table 1,2. Figure 7(a) provides a representative failure case.

Models consistently collapse on Polar layouts. Table 1 and Figure 6(a) present the Cartesian and Polar accuracy across all evaluated models. Frontier models achieving 69–83% on Cartesian layouts collapse to 31–39% on Polar equivalents, with the largest drop over 45 points, whereas human accuracy decreases by only 5.7 points from 94.5% to 88.8%. Notably, while Cartesian accuracy varies widely across model families (e.g., 82.6% for Gemini-3.1-Pro, 33.0% for Grok-4-0709), Polar scores compress into a narrow band, suggesting a shared performance floor that transcends model-specific capacity, consistent with a common dependence on orthogonal grid structure. Per-category analysis reveals that this collapse is non-uniform: *Combinatorics & Probability* suffers the steepest degradation (e.g., 92.4 → 24.9 for Gemini-3.1-Pro, 68.7 → 16.2 for Kimi-k2.5), which we hypothesize relates to the tasks' deeper dependency on structured grid enumeration. *Algorithmic Logic & Simulation* retains comparatively higher Polar accuracy (e.g., 52.6% for Gemini-3.1-Pro). This performance collapse stay consistency across diverse model families, scales, and training paradigms, suggests that the performance collapse reflects a shared dependence on orthogonal structure rather than a model-specific limitation.

Performance collapse persists across fully and partial alignment conditions. Figure 5 examines whether the collapse is attributable to the coordinate transformation itself or to confounds introduced by altered solution spaces or boundary conditions (Section 3). On *fully aligned* tasks (Figure 5(a)), where the coordinate transformation is the sole variable, all models still exhibit severe collapse (e.g.,

Table 2: Model performance across categories, comparing reasoning modes. $(-\Delta)$ are shown alongside the Polar column scores. The third row for each model represents the thinking gain.

Model	Reason	Algorithmic Logic & Simulation			Combinatorics & Probability			Navigation & Routing			Spatial Trans. & Geometry			Visual Pattern Matching			Overall		
		C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$
Gemini-3.1-Pro	✓	85.0	52.6	(−32.4)	92.4	24.9	(−67.6)	83.4	39.4	(−44.0)	83.7	31.4	(−52.3)	71.0	36.6	(−34.4)	82.6	35.9	(−46.7)
	×	66.7	35.6	(−31.1)	83.4	22.7	(−60.7)	82.9	42.6	(−40.3)	86.2	31.8	(−54.4)	71.3	37.6	(−33.7)	79.5	34.8	(−44.8)
		18.3	16.9		9.1	2.2		0.5	-3.2		-2.5	-0.4		-0.3	-1.0		3.1	1.1	
Gemini-3-Flash	✓	68.8	41.5	(−27.3)	74.8	20.6	(−54.2)	76.2	41.9	(−34.4)	68.7	29.8	(−38.8)	65.0	34.8	(−30.2)	71.0	33.8	(−37.2)
	×	34.2	19.3	(−14.8)	14.2	11.7	(−2.6)	42.9	37.1	(−5.9)	28.5	24.1	(−4.5)	36.7	24.0	(−12.7)	32.2	24.8	(−7.4)
		34.7	22.2		60.6	8.9		33.3	4.8		40.2	5.8		28.3	10.8		38.7	9.0	
Gemini-3-Flash-lite	✓	58.0	28.3	(−29.7)	40.4	15.9	(−24.6)	54.6	28.8	(−25.8)	40.6	21.7	(−18.9)	43.3	27.7	(−15.5)	46.8	24.6	(−22.2)
	×	26.5	19.2	(−7.3)	9.9	9.3	(−0.6)	36.7	33.0	(−3.7)	23.7	19.3	(−4.4)	36.7	26.5	(−10.3)	27.8	22.7	(−5.1)
		31.5	9.2		30.6	6.6		17.9	-4.2		16.9	2.4		6.5	1.3		19.0	1.9	
GPT-5.2	✓	70.7	46.7	(−24.0)	84.8	30.7	(−54.1)	82.2	42.8	(−39.4)	76.5	41.6	(−34.9)	70.0	34.5	(−35.5)	77.4	39.2	(−38.2)
	×	32.9	28.0	(−5.0)	11.2	10.7	(−0.5)	29.3	28.6	(−0.8)	24.1	25.2	(1.1)	35.4	28.1	(−7.3)	26.6	24.5	(−2.1)
		37.8	18.7		73.6	20.0		52.8	14.2		52.4	16.5		34.6	6.4		50.8	14.6	
Claude-Sonnet-4.6	✓	60.0	30.3	(−29.7)	42.7	16.8	(−25.9)	45.0	28.5	(−16.4)	42.3	25.2	(−17.0)	38.9	28.3	(−10.6)	44.4	25.9	(−18.5)
	×	45.4	26.9	(−18.5)	42.6	16.7	(−25.9)	45.5	25.5	(−20.0)	29.6	24.0	(−5.6)	37.5	28.4	(−9.1)	39.4	24.4	(−15.0)
		14.6	3.4		0.2	0.2		-0.5	3.0		12.6	1.2		1.5	-0.0		4.9	1.5	
Grok-4	✓	49.7	30.8	(−18.8)	17.7	12.3	(−5.3)	37.7	27.1	(−10.6)	28.4	21.5	(−6.8)	26.1	20.5	(−5.5)	31.0	22.3	(−8.7)
	×	18.2	17.8	(−0.4)	10.5	10.7	(0.2)	35.3	36.8	(1.5)	19.1	18.9	(−0.3)	21.5	18.1	(−3.4)	22.3	21.9	(−0.4)
		31.5	13.0		7.1	1.6		2.4	-9.7		9.2	2.7		4.6	2.4		8.6	0.3	
Kimi-k2.5	✓	73.5	37.3	(−36.1)	68.7	16.2	(−52.5)	73.0	38.3	(−34.7)	71.7	31.1	(−40.6)	58.4	30.8	(−27.5)	69.0	31.1	(−37.8)
	×	44.5	25.4	(−19.1)	20.9	8.0	(−12.9)	37.2	29.9	(−7.4)	27.8	23.2	(−4.6)	40.9	27.7	(−13.2)	33.7	23.6	(−10.2)
		29.0	12.0		47.8	8.2		35.7	8.4		43.9	8.0		17.5	3.1		35.2	7.6	
Gemma-4-31B	✓	67.7	36.0	(−31.7)	62.3	19.4	(−42.8)	63.9	37.6	(−26.2)	60.1	28.5	(−31.6)	51.5	32.5	(−19.1)	60.5	31.0	(−29.5)
	×	32.5	21.8	(−10.7)	19.3	11.3	(−8.0)	44.8	38.1	(−6.7)	24.2	22.2	(−2.0)	38.2	30.9	(−7.3)	32.7	26.3	(−6.3)
		35.2	14.2		42.9	8.1		19.1	-0.4		35.8	6.2		13.4	1.5		27.9	4.7	

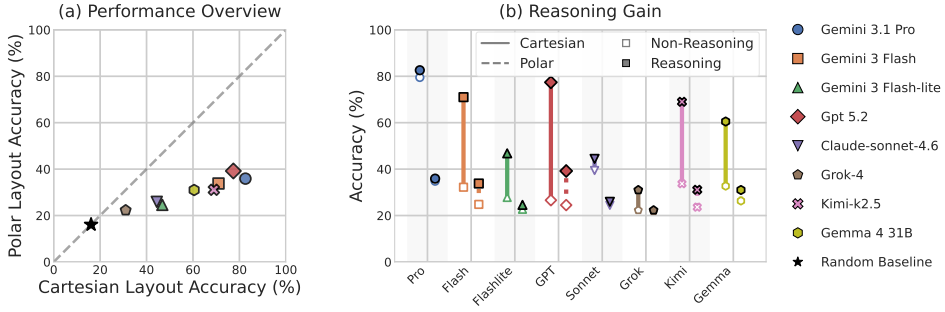


Figure 6: Performance overview and reasoning mode effect. *(a)* Overall Cartesian vs. Polar accuracy for each model (reasoning: high). *(b)* Reasoning effect on Cartesian and Polar accuracy. Connected pairs show the reasoning gains on Cartesian layouts consistently diminish on Polar equivalents.

Gemini-3.1-Pro: $82.0 \rightarrow 35.5$, Kimi-k2.5: $68.3 \rightarrow 32.6$) comparable in magnitude to the overall and partially aligned results. This shows that the performance collapse persists even when the transformation preserves complete logical equivalence, indicating that altered solution spaces are not a primary confound. *Partially aligned* tasks show marginally larger gaps (e.g., Gemini-3.1-Pro: $85.8 \rightarrow 38.0$), yet the collapse trend remains consistent. The *bounded* vs. *wrapping* comparison (Figure 5(b)) similarly shows consistent C to P drops under both conditions; wrapping mainly lowers Cartesian accuracy. Together, these results show that the performance collapse is robust across topological configurations, with boundary conditions (bounded vs. wrapping) exerting limited additional influence.

Thinking gains on Cartesian layouts are severely diminished on Polar equivalents. Table 2 and Figure 6(b) compare model performance under high reasoning and non-reasoning modes. On Cartesian layouts, enabling reasoning yields substantial accuracy improvements, with gains up to 50.8 points (GPT-5.2) and several models exceeding 30 points. However, on Polar layouts these gains are systematically compressed, with most models improving by fewer than 10 points: Gemini-3-Flash's gain shrinks from 38.7 to 9.0, Kimi-k2.5 from 35.2 to 7.6. At the category level, this pattern mirrors the findings in Table 1: *Combinatorics & Probability*, which benefits more from reasoning on Cartesian (e.g., Gemini-3-Flash: 60.6 points), sees these gains collapse more severely on Polar (8.9

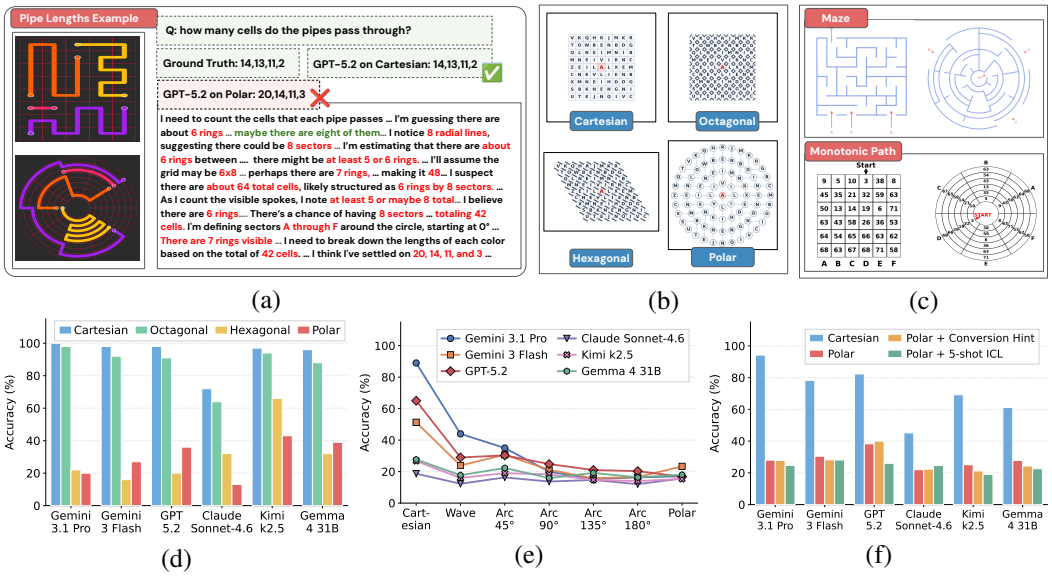


Figure 7: Case study and analyses for the discussion. *(a)* Pipe Lengths case study: GPT-5.2 answers Cartesian correctly but repeatedly revises its ring and sector counts on Polar. *(b)* Word Search in Cartesian, Octagonal, Hexagonal, and graph-based Polar layouts. *(c)* Maze and Monotonic Path in both layouts. *(d)* Word Search accuracy on the four layouts in (b). *(e)* Accuracy under progressive deformation of Cartesian layouts, averaged over 3 tasks. *(f)* Accuracy on Cartesian, Polar, and Polar with a conversion hint or 5-shot ICL, averaged over 6 tasks. (d-f) report 6 representative models.

points), while *Algorithmic Logic & Simulation* retains comparatively more of its reasoning benefit. This asymmetry shows that thinking gains are strongly coupled to the presence of orthogonal grid structure: reasoning substantially amplifies performance on Cartesian layouts, but this amplification diminishes severely on Polar equivalents.

5 Discussions

We further characterize the Cartesian-Polar gap along three axes: whether it is specific to Polar layouts; whether it can be closed by test-time guidance; whether it appears uniformly across tasks.

The gap is not specific to Polar layouts. On Word Search, we compare Cartesian and graph-based Polar layouts with Octagonal and Hexagonal tilings (Figure 7(b)). Octagonal tilings, which preserve local orthogonal adjacency, stay close to Cartesian accuracy, whereas Hexagonal and Polar layouts show large drops for most models (Figure 7(d)). We further apply progressive geometric deformations to the Cartesian images of three tasks (Figure 7(e)). For models with substantial Cartesian accuracy, a sine-wave deformation, which introduces neither wrap-around nor radial compression, already causes a large drop (e.g., Gemini 3.1 Pro: 89.0% to 44.0%), a 45 arc has a similar effect, and accuracy approaches the Polar level by the 90 arc. Models near floor on Cartesian for these tasks (Sonnet 4.6, Kimi K2.5, Gemma 4 31B) vary little across conditions. Comparable drops therefore arise without Polar-specific properties, on several layouts that depart from the standard orthogonal grid.

The gap is not closed by test-time guidance. We evaluate two interventions on six tasks with large drops across six models (Figure 7(f)): (1) a conversion hint instructing the model to re-map the Polar layout onto a Cartesian grid before solving, and (2) 5-shot in-context learning (Dong et al., 2024) with solved Polar examples. Conversion hints change accuracy by less than 4 points relative to the Polar baseline, and 5-shot examples lower accuracy for five of the six models (e.g., GPT-5.2: 38.3% to 26.0%). Test-time instruction and demonstrations therefore do not close the gap between familiar orthogonal layouts and Polar layouts, which humans solve with a modest drop (Table 1).

Two tasks show no Cartesian-Polar gap. In *Monotonic Path*, leading models exceed 80% on both Cartesian and Polar layouts; while in *Maze*, most models score below 40% on both (Figure 7(c)).

Inspection of thinking traces on Monotonic Path reveals that models consistently reference the numerical labels in cells during reasoning (e.g., "currently at 35, move to 45"); Maze contains no such labels. These cases show that the gap does not appear uniformly whenever the layout changes: models can reach high accuracy on Polar layouts and can score low on Cartesian ones. They are consistent with the use of textual representations observed in Section 2, although we present them as illustrative cases rather than a systematic analysis.

6 Related works

Vision reasoning in MLLMs. The rapid advancement of MLLMs (Gemini Team, 2025; Singh et al., 2025; Team et al., 2026; Qwen Team, 2026; Anthropic, 2024) has driven a proliferation of visual reasoning benchmarks evaluating mathematical reasoning (Lu et al., 2024; Hao et al., 2025), multi-discipline understanding (Yue et al., 2025; 2024), spatial reasoning (Jia et al., 2026), and broader, deeper vision reasoning (Chen et al., 2025; Xu et al., 2025a; Meng et al., 2024; Tong et al., 2024a; Zou et al., 2026; Wang et al., 2024b;a), with several adopting structured grid-based layouts for precise assessment of spatial and algorithmic problem-solving (Tang & Kejriwal, 2025; Ren et al., 2025). Alongside this progress, growing evidence questions whether current models genuinely reason over visual content. One line of work reveals that MLLMs fail at perceptual tasks humans find trivial (Fu et al., 2024; Tong et al., 2024b; Ye et al., 2026; Chen et al., 2026; Saxena et al., 2025). Another line questions whether strong benchmark scores genuinely reflect visual understanding (Asadi et al., 2026; Chen et al., 2024; Xu et al., 2025b): evaluation practices may overestimate true capability (Chen et al., 2024), reasoning modes can amplify hallucination rather than improve visual grounding (Xu et al., 2025b), and frontier models generate elaborate reasoning for images never provided (Asadi et al., 2026). These concerns connect to shortcut learning (Geirhos et al., 2020; Xia et al., 2025): models exploiting superficial regularities that generalize within benchmarks but collapse under distribution shift. However, the specific mechanisms by which models circumvent genuine visual reasoning remain underexplored. In this work, we identify the Cartesian Shortcut, a pervasive pattern in which models discretize orthogonal layouts into textual coordinates, and show that model performance collapses consistently when this structure is disrupted.

Coordinate systems and geometric foundations. Standard neural architectures are built upon grid-structured representations with inherent translational symmetry (Bronstein et al., 2021), an architectural prior that may contribute to model's Cartesian deduction behavior. The equivalence and duality between different coordinate representations has long been studied in mathematics (Arfken et al., 2011) and vision science (Zetzsche et al., 1999). To our knowledge, this is the first work to leverage coordinate transformation as a systematic diagnostic probe for MLLMs visual reasoning.

7 Conclusion

In this work, we identify the Cartesian Shortcut, a pervasive pattern whereby MLLMs systematically discretize orthogonal grid layouts into textual coordinates, offloading reasoning from visual perception to text-based deduction. We introduce Polaris-Bench, a controlled diagnostic benchmark of 53 paired Cartesian-Polar visual reasoning tasks. We choose the Cartesian Polar pair because the bijective mapping preserves task semantics while cleanly disrupting orthogonal structure, enabling a rigorous paired evaluation. Our experiments reveal that current MLLMs' visual reasoning performance is strongly coupled to orthogonal grid structure: when this structure is disrupted, performance collapses consistently across model families, is not recovered by prompting interventions, and is far smaller in humans. We do not claim that Polar conversion is either necessary or sufficient for performance collapse: several non-Polar deformations produce comparable drops (Section 5), and Polar tasks with textual anchors show no gap. Rather, this work establishes the scope and severity of models' dependence on orthogonal structure, and provides a controlled testbed for studying it. Separating the contributions of familiarity, geometry, and orthogonal structure through training-level interventions, extending to 3D and naturally occurring non-orthogonal visuals, and developing visual reasoning that generalizes across layout geometries all remain important open directions.

AI USE STATEMENT

The discovery of the Cartesian Shortcut and the Cartesian-Polar paired evaluation design are the authors' own. Generative AI was not used for brainstorming, formulating hypotheses, or designing experiments. To source candidate tasks, we used generative AI to help survey existing benchmarks for problems defined on Cartesian coordinates and to run pilot evaluations on these candidates. The authors made all final decisions on which tasks to include, balancing coverage and diversity. The authors also designed, task by task, the transformation from Cartesian to Polar: whether a task admits a meaningful Polar counterpart, its rules and constraints, its visual layout, and its question and ground-truth answer. Polaris-Bench consists of procedurally generated synthetic data (Section 3.2; Appendix B.2). From these author designs, an LLM coding assistant helped implement the generation scripts over multiple iterative rounds, in which the authors checked the code and its outputs for correctness and directed revisions. The scripts were additionally examined through an LLM-assisted code audit (Appendix B.4). Model outputs are scored by exact match, and Gemini-3.1-Pro serves as an LLM judge only when exact match cannot determine the correctness of a response (Section 4.1). Generative AI assisted with the code for plotting experimental results and generating result tables, which the authors reviewed; the conceptual illustration figures (Figures 1 and 2) were drawn by the authors without AI. Generative AI was also used as an aid in examining and interpreting experimental results, and every reported finding was verified by the authors against the raw results. We also used generative AI tools to search for and summarize related literature, with every cited work read and verified by the authors; and to draft, polish, and condense passages from the authors' own content. We take full responsibility for the final content of this paper, including text, claims, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We have taken the following steps to support the reproducibility of this work. *Benchmark data.* Polaris-Bench consists of 10,600 fixed Cartesian-Polar instances (53 tasks $\times$ 100 pairs) and 200 additional instances (Word Search: Octagonal, Hexagonal) with deterministic ground-truth answers, and all main results are computed on these instances. The complete benchmark has been publicly released; the link is omitted here to preserve anonymity and will be provided in the camera-ready version. *Benchmark construction.* Section 3 and Appendix A specify all 53 tasks, including their rules, answer formats, alignment, and boundary conditions (Tables 3–7); Appendix B details procedural generation, visual calibration, rendering, and quality assurance; and Appendix E provides representative Cartesian–Polar examples. The generation scripts will be added to the same public release upon completion of an ongoing institutional review; since evaluation is performed on the released fixed benchmark instances, reproducing the reported results does not depend on these scripts. *Evaluation.* Section 4 and Appendix C.1 list all evaluated models, their access routes, and reasoning configurations; Appendix C.3 gives the exact prompt templates for the standard and intervention settings; Appendix C.4 describes the exact-match parsers and the LLM-judge fallback and its validity; Appendix C.2 details the Cartesian Shortcut analyses; and Appendices B.5 and C.5 describe the human study protocol. Complete per-task results and the setups of all discussion experiments are reported in Appendix D.

REFERENCES

Mistral AI. Mistral-small-3.1-24b-instruct-2503, March 2025. URL https://mistral.ai/news/mistral-small-3-1.

AI Anthropic. The claude 3 model family: Opus, sonnet, haiku. *Claude-3 Model Card*, 1(1):4, 2024.

George B Arfken, Hans J Weber, and Frank E Harris. *Mathematical methods for physicists: a comprehensive guide*. Academic press, 2011.

Mohammad Asadi, Jack W. O'Sullivan, Fang Cao, Tahoura Nedaee, Kamyar Rajabalifardi, Fei-Fei Li, Ehsan Adeli, and Euan Ashley. Mirage: The illusion of visual understanding, 2026. URL https://arxiv.org/abs/2603.21687.

Michael M Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *arXiv preprint arXiv:2104.13478*, 2021.

Jiacheng Chen, Tianhao Liang, Sherman Siu, Zhengqing Wang, Kai Wang, Yubo Wang, Yuansheng Ni, Wang Zhu, Ziyan Jiang, Bohan Lyu, Dongfu Jiang, Xuan He, Yuan Liu, Hexiang Hu, Xiang Yue, and Wenhu Chen. Mega-bench: Scaling multimodal evaluation to over 500 real-world tasks. 2025.

Liang Chen, Weichu Xie, Yiyan Liang, Hongfeng He, Hans Zhao, Zhibo Yang, Zhiqi Huang, Haoning Wu, Haoyu Lu, Y. charles, Yiping Bao, Yuantao Fan, Guopeng Li, Haiyang Shen, Xuanzhong Chen, Wendong Xu, Shuzheng Si, Zefan Cai, Wenhao Chai, Ziqi Huang, Fangfu Liu, Tianyu Liu, Baobao Chang, Xiaobo Hu, Kaiyuan Chen, Yixin Ren, Yang Liu, Yuan Gong, and Kuan Li. Babyvision: Visual reasoning beyond language, 2026. URL https://arxiv.org/abs/2601.06521.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, et al. Are we on the right way for evaluating large vision-language models? *Advances in Neural Information Processing Systems*, 37:27056–27087, 2024.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Google DeepMind. Gemma 4: Our most intelligent open models to date. https://blog.google/innovation-and-ai/technology/developers-tools/gemma-4/, 2026. Accessed: 2026-05-01.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Tianyu Liu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. A survey on in-context learning, 2024. URL https://arxiv.org/abs/2301.00234.

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

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

Gemini Team. Gemini 3 pro technical report. 2025. URL https://deepmind.google/models/gemini/pro/.

Gemini Team and Google DeepMind. Gemini 3 flash: Frontier intelligence built for speed. Technical blog / model release, Google, 12 2025. URL https://blog.google/products-and-platforms/products/gemini/gemini-3-flash/.

Gemini Team and Google DeepMind. Gemini 3.1 flash-lite: Built for intelligence at scale. Technical blog / model release, Google, 3 2026. URL https://blog.google/innovation-and-ai/models-and-research/gemini-models/gemini-3-1-flash-lite/.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, et al. A survey on llm-as-a-judge. *The Innovation*, 2024.

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 *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=v26vwjxOEz.

Mengdi Jia, Zekun Qi, Shaochen Zhang, Wenyao Zhang, XinQiang Yu, Jiawei He, He Wang, and Li Yi. Omnispatial: Towards comprehensive spatial reasoning benchmark for vision language models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=6nZKT2rL0H.

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 (Volume 1: Long Papers)*, pp. 7285–7326, Rabat, Morocco, March 2026. Association for Computational Linguistics. ISBN 979-8-89176-380-7. doi: 10.18653/v1/2026.eacl-long.343. URL https://aclanthology.org/2026.eacl-long.343/.

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. In *The Twelfth International Conference on Learning Representations*, 2024.

Fanqing Meng, Jin Wang, Chuanhao Li, Quanfeng Lu, Hao Tian, Jiaqi Liao, Xizhou Zhu, Jifeng Dai, Yu Qiao, Ping Luo, Kaipeng Zhang, and Wenqi Shao. Mmiu: Multimodal multi-image understanding for evaluating large vision-language models, 2024. URL https://arxiv.org/abs/2408.02718.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

Yufan Ren, Konstantinos Tertikas, Shalini Maiti, Junlin Han, Tong Zhang, Sabine Süsstrunk, and Filippos Kokkinos. Vgrp-bench: Visual grid reasoning puzzle benchmark for large vision-language models. *arXiv preprint arXiv:2503.23064*, 2025.

Rohit Saxena, Aryo Pradipta Gema, and Pasquale Minervini. Lost in time: Clock and calendar understanding challenges in multimodal llms, 2025. URL https://arxiv.org/abs/2502.05092.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, et al. Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*, 2025.

Zhisheng Tang and Mayank Kejriwal. Grasp: A grid-based benchmark for evaluating commonsense spatial reasoning, 2025. URL https://arxiv.org/abs/2407.01892.

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, SH Cai, Yuan Cao, Y Charles, HS Che, Cheng Chen, Guanduo Chen, et al. Kimi k2. 5: Visual agentic intelligence. *arXiv preprint arXiv:2602.02276*, 2026.

Shengbang Tong, Ellis L Brown II, Penghao Wu, Sanghyun Woo, ADITHYA JAIRAM 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 *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024a. URL https://openreview.net/forum?id=Vi8AepAXGy.

Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9568–9578, 2024b.

Fei Wang, Xingyu Fu, James Y Huang, Zekun Li, Qin Liu, Xiaogeng Liu, Mingyu Derek Ma, Nan Xu, Wenxuan Zhou, Kai Zhang, et al. Muirbench: A comprehensive benchmark for robust multi-image understanding. *arXiv preprint arXiv:2406.09411*, 2024a.

Zirui Wang, Mengzhou Xia, Luxi He, Howard Chen, Yitao Liu, Richard Zhu, Kaiqu Liang, Xindi Wu, Haotian Liu, Sadhika Malladi, Alexis Chevalier, Sanjeev Arora, and Danqi Chen. Charxiv: Charting gaps in realistic chart understanding in multimodal llms. *arXiv preprint arXiv:2406.18521*, 2024b.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

xAI. Grok-4 model card. 2025. URL https://x.ai/news/grok-4.

Jiaer Xia, Yuhang Zang, Peng Gao, Sharon Li, and Kaiyang Zhou. Visionary-r1: Mitigating shortcuts in visual reasoning with reinforcement learning, 2025. URL https://arxiv.org/abs/2505.14677.

Weiye Xu, Jiahao Wang, Weiyun Wang, Zhe Chen, Wengang Zhou, Aijun Yang, Lewei Lu, Houqiang Li, Xiaohua Wang, Xizhou Zhu, Wenhai Wang, Jifeng Dai, and Jinguo Zhu. Visulogic: A benchmark for evaluating visual reasoning in multi-modal large language models. *arXiv preprint arXiv:2504.15279*, 2025a. URL https://arxiv.org/abs/2504.15279.

Zhongxing Xu, Chengzhi Liu, Qingyue Wei, Juncheng Wu, James Zou, Xin Eric Wang, Yuyin Zhou, and Sheng Liu. More thinking, less seeing? assessing amplified hallucination in multimodal reasoning models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025b.

Junyan Ye, Dongzhi Jiang, Jun He, Baichuan Zhou, Zilong Huang, Zhiyuan Yan, Hongsheng Li, Conghui He, and Weijia Li. BLINK-twice: You see, but do you observe? a reasoning benchmark on visual perception. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2026. URL https://openreview.net/forum?id=g0AMmWiHCq.

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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9556–9567, 2024.

Xiang Yue, Tianyu Zheng, Yuansheng Ni, Yubo Wang, Kai Zhang, Shengbang Tong, Yuxuan Sun, Botao Yu, Ge Zhang, Huan Sun, et al. Mmmu-pro: A more robust multi-discipline multimodal understanding benchmark. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15134–15186, 2025.

Christoph Zetzsche, Gerhard Krieger, and Bernhard Wegmann. The atoms of vision: Cartesian or polar? *Journal of the Optical Society of America A*, 16(7):1554–1565, 1999.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

Kai Zou, Ziqi Huang, Yuhao Dong, Shulin Tian, Dian Zheng, Hongbo Liu, Jingwen He, Bin Liu, Yu Qiao, and Ziwei Liu. Uni-mmmu: A massive multi-discipline multimodal unified benchmark, 2026. URL https://arxiv.org/abs/2510.13759.

A BENCHMARK DETAILS

A.1 TASK TAXONOMY

The Polaris-Bench comprises 53 procedurally generated visual reasoning tasks, each yielding 200 QA pairs (100 Cartesian + 100 Polar), totaling 10,800 QA pairs across the suite (Word Search additionally includes Hexagonal and Octagonal variants, contributing 400 QA pairs). We organize these tasks into five cognitive domains based on the core reasoning ability each task primarily evaluates:

- **Visual Pattern Matching** (11 tasks) evaluates pattern recognition and structural discrimination, including shape assembly, visual search, and sequential pattern continuation.
- **Spatial Transformation and Geometry** (13 tasks) evaluates geometric transformation reasoning (e.g., rotation, reflection, folding) alongside quantitative geometric measurement such as area counting and curve estimation.
- **Navigation and Routing** (14 tasks) evaluates sequential positional state tracking, requiring models to execute or trace multi-step movement paths under environmental constraints such as walls, wrapping boundaries, and monotonicity rules.
- **Combinatorics and Probability** (9 tasks) evaluates systematic combinatorial enumeration and probabilistic reasoning, including path counting under topological boundary conditions and stochastic trajectory analysis.
- **Algorithmic Logic and Simulation** (6 tasks) evaluates constraint satisfaction, discrete optimization, and deterministic forward simulation of physical or rule-based dynamics.

Tables 3–7 provide a complete per-task breakdown. For each task, we report its sub-category, answer type, a one-sentence description, and two topological properties. *Alignment* indicates whether the Cartesian and Polar versions share identical ground-truth answers (✓ = Fully Aligned) or intentionally diverge due to intrinsic coordinate differences (✗ = Partially Aligned). *Boundary* specifies the Polar layout's boundary condition: *Bounded* (closed, non-wrapping edges), *Wrapping* (360° angular wrap-around), or "—" (tasks unaffected by the boundary condition). Figure 8 provides a statistic of task distribution among answer type, alignment between Cartesian and Polar, and boundary condition.

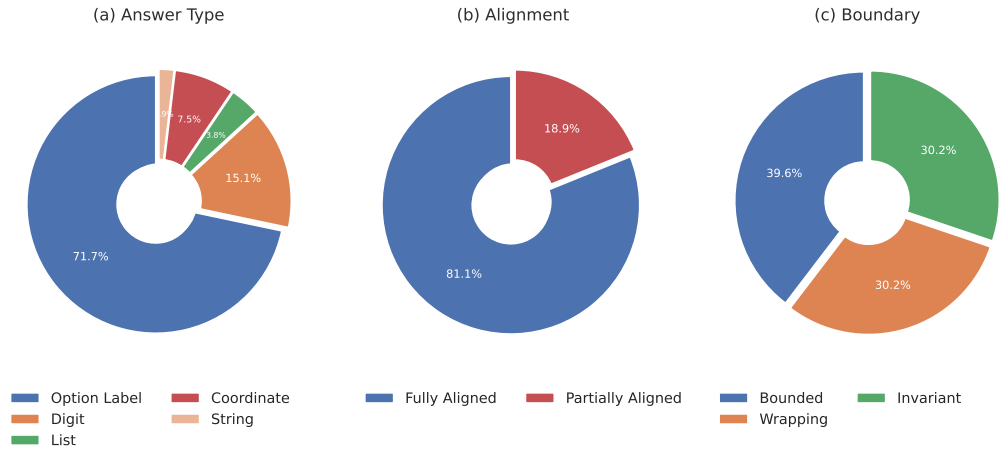


Figure 8: Task distribution of Polaris-Bench across *(a)* answer type, *(b)* alignment, *(c)* boundary condition.

Partially Aligned Tasks Below we provide task-specific explanation to partial aligned tasks, where the Cartesian and Polar instances share same problem rules, but the valid solution and ground-truth answers are different:

- Impossible Shape. Finding a perfectly identical shape correspondence between Cartesian (rectangular cells) and Polar (sector-shaped cells) layouts is inherently infeasible. We

therefore control for identical grid dimensions and overall shape silhouette, but due to the geometric differences between cell types, the set of valid rotation and assembly configurations may differ.

- Knight Paths. Under identical problem setups, the Polar layout permits angular wrapping, granting the knight additional boundary-crossing moves that do not exist on a bounded Cartesian grid. This results in more valid paths to reach the destination in Polar.
- Longest Path. The set of candidate paths and the correct answer are identical across both versions. However, to prevent candidate paths from inadvertently connecting head-to-tail through angular wrapping, the Polar layout uses one additional sector compared to the number of Cartesian columns.
- Maximum Collection. The Polar layout permits angular wrapping, expanding the reachable region along the trajectory. Consequently, more numbers can be collected in Polar under the same movement rules, yielding a higher optimal collection value.
- Minimum Flips. Angular wrapping makes the first and last columns effectively adjacent, introducing additional parity constraints on the boundary. This alters the minimum number of flips required to achieve the target pattern.
- Random Walk. The problem formulation and correct answer are kept identical across both versions. To achieve answer alignment despite wrapping, the Polar layout uses a slightly larger number of sectors, with the key intersections (A, B, C) placed within a contiguous sector region whose internal connectivity mirrors that of the Cartesian grid.
- Rotation Center. To ensure that the answer and key point positions align correctly between the two coordinate systems, the Polar grid uses a different grid size from the Cartesian version.
- Wrapping Diagonal Path. Angular wrapping in Polar allows diagonal paths to wrap around the angular boundary, creating additional valid trajectories that do not exist in the bounded Cartesian grid. This increases the total count of valid diagonal paths.

Table 3: Details of 11 tasks for the **Visual Pattern Matching** domain. Each task yields 200 QA pairs (100 Cartesian + 100 Polar) unless otherwise noted.

Task Name	Sub-Category	Answer Type	Description	#QA	Alignment	Boundary
Pattern Completion	Shape and Jigsaw Assembly	Option label	Fill a missing contiguous region in a bi-color diagonally partitioned layout	200	✓	—
Shape Fitting	Shape and Jigsaw Assembly	Option label	Choose the piece that exactly fills an irregular hole in a cell layout	200	✓	Bounded
Layer Completion	Shape and Jigsaw Assembly	Option label	Select the piece that fills a missing chunk in a concentric-layer colored grid	200	✓	Bounded
Fragment Matching	Shape and Jigsaw Assembly	Option label	Match a grid fragment to a hole based on non-uniform line spacing patterns	200	✓	—
Template Matching	Shape and Jigsaw Assembly	Option label	Find which puzzle piece (with rotation/flip) is hidden in a symbol grid	200	✓	Wrapping
Jigsaw Matching	Shape and Jigsaw Assembly	Option label	Determine which two of four pieces combine to form a complete shape	200	✓	Bounded
Shape Completion	Shape and Jigsaw Assembly	Option label	Identify which piece fills the missing gap to complete a solid shape	200	✓	Wrapping
Odd Piece Out	Visual Spotting and Search	Option label	Find the imposter piece that doesn't fit when assembling 4 pieces into a shape	200	✓	—
Anomaly Detection	Visual Spotting and Search	Coordinate	Find the one subtly different item in a large grid of identical items	200	✓	—
Letter Collection	Visual Spotting and Search	String	Walk along a spiral path and collect letters only on your right-hand side	200	✓	Wrapping
Pattern Prediction	Pattern Continuation	Option label	Predict which point lies on a continuing periodic bump pattern	200	✓	Wrapping

B Benchmark Construction Details

B.1 Visual Calibration

The Polar coordinate mapping introduces nonlinear spatial distortion that, without mitigation, could confound the comparison by adding irrelevant legibility differences. We apply task-specific visual

Table 4: Task details for the **Spatial Transformation and Geometry** domain (13 tasks). Notation follows Table 3.

Task Name	Sub-Category	Answer Type	Description	#QA	Alignment	Boundary
Rotation Matching	Rotations and Reflections	List	Match 3 rotated versions of a colored ring/polygon pattern to 6 candidates	200	✓	Wrapping
Mirror Reflection	Rotations and Reflections	Option label	Identify the correct mirror reflection of a half-filled colored layout	200	✓	Bounded
Grid Rotation	Rotations and Reflections	Option label	Identify a grid's appearance after a 90° rotation from 5 options	200	✓	—
Rotation Center	Rotations and Reflections	Option label	Determine which point is the center of rotation between two segments	200	✗	—
Pivot Rotation	Rotations and Reflections	Option label	Identify the appearance of a shape after 180° rotation around a marked pivot	200	✓	—
Impossible Shape	Spatial Constraint and Topology	Option label	Find which combined grid cannot be made by inserting a piece into a frame	200	✗	Wrapping
Grid Folding	Spatial Constraint and Topology	Option label	Fold a grid along a line; find which labeled cell is not covered	200	✓	Bounded
Four-Color	Spatial Constraint and Topology	Option label	Insert a colored connected piece into a gap without violating adjacency rules	200	✓	—
Area Counting	Geometric Measurement and Counting	Digit	Count the number of gray shaded cells forming a connected region	200	✓	Bounded
Pipe Lengths	Geometric Measurement and Counting	List	Count the lengths of colored pipes filling a layout; report in descending order	200	✓	—
Uncut Cells	Geometric Measurement and Counting	Option label	Count cells not intersected by two boundary-spanning diagonal cuts	200	✓	Bounded
Area Balancing	Geometric Measurement and Counting	Option label	Choose the cell pattern that makes total black area equal total white area	200	✓	—
Curve Length	Geometric Measurement and Counting	Option label	Calculate the total length of a snake tracing cell boundary edges	200	✓	—

Table 5: Task details for the **Navigation and Routing** domain (14 tasks). Notation follows Table 3.

Task Name	Sub-Category	Answer Type	Description	#QA	Alignment	Boundary
Maze	Optimal Pathfinding	Option label	Find which labeled entrance leads to the exit through a maze	200	✓	Bounded
Shortest Path	Optimal Pathfinding	Option label	Compare 4 diagonal zigzag routes and find the shortest (or all-equal)	200	✓	—
Longest Path	Optimal Pathfinding	Option label	Compare 5 paths on grids and identify which is the longest	200	✗	Bounded
Bounded Path Finding	Optimal Pathfinding	Option label	Select the correct move sequence for a penguin reaching a fish on a bounded grid	200	✓	Bounded
Wrapping Path Finding	Optimal Pathfinding	Option label	Select the correct move sequence on a cylindrical wrapping grid	200	✓	Wrapping
Wrapping Navigation	Rule-Based Navigation	Option label	Trace an absolute-direction movement sequence on a wrapping grid to find the target animal	200	✓	Wrapping
Egocentric Navigation	Rule-Based Navigation	Option label	Mouse traces egocentric commands (Fwd/Back/Turn) through cacti to find food	200	✓	Bounded
Absolute Navigation	Rule-Based Navigation	Option label	Turtle traces absolute directions through cacti to find flowers (bounded)	200	✓	Bounded
Wall Follower	Rule-Based Navigation	Coordinate	Simulate a robot that turns right at walls; find where it stops or loops	200	✓	Bounded
Rule-Based Navigation	Rule-Based Navigation	Coordinate	Simulate a drone with alternating R/L banking rules; find where it lands	200	✗	Bounded
Monotonic Path	Constraint-Based Routing	Option label	Navigate rooms with strictly increasing numbers from START to the correct exit	200	✓	Wrapping
Turn Counting	Constraint-Based Routing	Option label	Identify the path with the exact specified number of left and right turns	200	✓	Bounded
Word Search	Constraint-Based Routing	Digit	Count valid adjacent-cell paths that spell a target word from the center	400	✓	—
Largest Number Path	Constraint-Based Routing	Option label	Trace digit paths through a number grid; identify which forms the largest number	200	✓	—

calibrations to ensure that critical visual elements remain perceptually comparable across Cartesian and Polar layouts.

For example, for tasks where font size is a key perceptual factor (e.g., word search, odd one out), we enforce matched letter sizes across both layouts. Since Polar cells near the pole are significantly

Table 6: Task details for the **Combinatorics and Probability** domain (9 tasks). Notation follows Table 3.

Task Name	Sub-Category	Answer Type	Description	#QA	Alignment	Boundary
Path Counting	Bounded Combinatorics	Option label	Count distinct shortest paths from P to Q avoiding wall obstacles	200	✓	Bounded
Bounded Diagonal Paths	Bounded Combinatorics	Digit	Count diagonal paths from P to Q with a wall blocking Polar wrap-around	200	✓	Bounded
Bounded Knight Paths	Bounded Combinatorics	Digit	Count knight paths from start to target with a wall blocking sector wrap-around	200	✓	Bounded
Checkpoint Paths	Bounded Combinatorics	Digit	Count right/down paths from X to Y passing through checkpoint ∇, avoiding gaps	200	✓	Bounded
Lattice Paths	Bounded Combinatorics	Digit	Count paths on a dot grid allowing right, down, and diagonal-down-right moves	200	✓	Bounded
Wrapping Diagonal Paths	Topological Combinatorics	Digit	Count diagonal paths from P to Q on an alternating-color layout with wrapping	200	✗	Wrapping
Knight Paths	Topological Combinatorics	Digit	Count paths for a chess knight from start to target in exactly K steps	200	✗	Wrapping
Edge Counting	Stochastic Processes	Option label	Count the number of cell edges an ant traverses from P to Q	200	✓	—
Random Walk	Stochastic Processes	Option label	Calculate the probability of passing through point C on a random walk A→B	200	✗	Wrapping

Table 7: Task details for the **Algorithmic Logic and Simulation** domain (6 tasks). Notation follows Table 3.

Task Name	Sub-Category	Answer Type	Description	#QA	Alignment	Boundary
N-Queens	Constraint Satisfaction	Option label	Place missing queens on a partial N-Queens board; deduce the target coordinate	200	✓	Bounded
Sudoku	Constraint Satisfaction	Option label	Solve for the value of a highlighted cell in a partially filled Sudoku	200	✓	—
Minimum Flips	Move Optimization	Option label	Find minimum moves to achieve an alternating two-color pattern by flipping 3-cell strips	200	✗	Wrapping
Maximum Collection	Move Optimization	Option label	Find the maximum cheese a mouse can eat on a non-repeating maze path	200	✗	Wrapping
Collision Detection	Motion and Trajectory Prediction	Option label	Determine which pair of moving cars will crash first based on velocity arrows	200	✓	Wrapping
Bouncing Point	Motion and Trajectory Prediction	Coordinate	Compute coordinates after N steps of a bouncing/looping point	200	✗	Wrapping

compressed, we adjust the Polar rendering to use cells large enough for legible text. To maintain size parity, we correspondingly introduce additional white space in the Cartesian layout, forcing the core visual elements (i.e., the letters) to occupy a comparable rendered area rather than filling the entire grid. This ensures that any performance gap reflects reasoning difficulty rather than a legibility disadvantage.

For tasks involving geometric shapes or icons, we apply analogous area-matching calibrations: element sizes in Cartesian layouts are scaled to approximate the rendered area of their Polar counterparts, rather than naively filling the available cell space. These calibrations are tuned on a per-task basis through manual inspection.

B.2 Procedural Generation

Each of the 53 tasks is implemented as a self-contained Python script that procedurally generates paired Cartesian–Polar instances. The generation process involves several carefully tuned components.

Randomization Tuning. We randomize multiple axes per task, including grid dimensions, starting positions, path topologies, target placements, distractor counts, and color palettes. The randomization bounds for each axis were manually calibrated through iterative testing to satisfy two constraints simultaneously: (1) all generated instances must be solvable with a unique correct answer, and (2) no configuration should produce visual elements that are too small or cluttered to be perceived, particularly under Polar distortion. This tuning process was performed by the authors through systematic manual review.

Solvability and Corner Case Resolution. Each script includes programmatic solvability checks that verify the generated instance has a valid, unique solution before it is accepted. Beyond these automated checks, we conducted extensive manual and LLM-assisted reviews to identify and resolve corner cases. For example, we discovered through literature review that N-Queens under wrapping (i.e., toroidal) boundary conditions has no valid solution for most board sizes; we therefore constrained this task to use bounded topology exclusively. Similar task-specific adaptations were applied across the benchmark wherever edge cases were identified.

Distractor Validation. For tasks with multiple-choice or enumeration-based answers, we implement explicit distractor validation to ensure that incorrect answer options are genuinely incorrect. The generative scripts verify that each distractor does not accidentally coincide with the ground-truth answer under alternative interpretations or boundary conditions. This prevents ambiguous evaluation where a model might select a "wrong" answer that is in fact defensible.

B.3 Rendering and Legibility

Rendering Polar layouts introduces several unique visual challenges that require dedicated engineering solutions.

Anti-Collision Constraints. We enforce strict anti-collision rules to prevent overlapping visual elements. Grid lines, labels, icons, and path traces are rendered with minimum separation guarantees. Under Polar distortion, cells near the pole are significantly smaller than those near the boundary; we dynamically adjust label placement and font sizing to prevent overlap in these compressed regions.

Boundary Wrapping Artifacts. Certain visual patterns that appear straightforward in Cartesian layouts create rendering artifacts under Polar mapping. For instance, a polyline that spans both the left and right edges of a Cartesian grid will form a closed circle in Polar space, fundamentally altering its visual appearance. We address such cases through multiple strategies: constraining the generative script to produce polylines that touch at most one boundary, adding an extra column to the Polar layout to break the visual closure, or programmatically detecting and excluding instances where unwanted closed loops arise.

Inner Ring Radius Optimization. If the inner ring radius is too small, elements near the pole become illegibly compressed; if too large, the outer rings waste space. We tune the inner ring radius on a per-task basis to find the optimal balance between visual clarity near the pole and efficient use of the rendering canvas. This parameter is adjusted through manual inspection of representative instances for each task.

B.4 Quality Assurance Pipeline

Our quality assurance process consists of three iterative stages followed by a final human validation study.

Author Review. The authors collaboratively reviewed the logical design of each task, verifying that the rules, constraints, and ground-truth computations are correct for both Cartesian and Polar versions. In addition, we independently spot-checked randomly sampled instances for each of the 53 tasks by manually solving them and comparing our answers against the computed ground truths.

LLM-Assisted Code Audit. We used Gemini-3.1-Pro to systematically audit the procedural generation code, prompting the model to identify potential edge cases, off-by-one errors, and boundary condition failures. This process surfaced several subtle bugs that were not apparent during manual review, including incorrect wrapping behavior at grid boundaries and degenerate configurations for specific grid dimensions.

Iterative Convergence. The above two stages were repeated in multiple rounds. Each round produced a list of identified issues, which were resolved before the next round of review. This process continued until no further issues were identified across all 53 tasks.

Representative Examples of Resolved Issues. To illustrate the depth of this process, we highlight several representative cases:

- *N-Queens wrapping:* Literature review revealed that N-Queens under toroidal (wrapping) boundary conditions has no valid solution for most board sizes. We adapted this task to use bounded topology exclusively.
- *Polyline boundary closure:* Polylines spanning both edges of a Cartesian grid form closed circles under Polar mapping. We constrained generation to avoid this artifact (Section B.3).
- *Distractor collision:* In several combinatorial tasks, randomly generated distractors occasionally coincided with the correct answer under alternative boundary interpretations. We added explicit distractor validation checks to eliminate these cases.

Representative Design Refinements. Beyond resolving correctness issues, we iteratively refined task designs in this stage based on observations and results, for more comprehensive evaluation and reduce possible shortcut for answering. Below are some representative examples:

- *Four Color:* This task originally used numerical labels (1–4) to fill cells, presenting a layout close to the Monotonic Path task. Pilot evaluation revealed that both tasks exhibited similarly high accuracy across Cartesian and Polar layouts. After comparative analysis on their CoT reasoning chain, we retained numerical labels for monotonic path and switched four color to use color-filled cells, creating a controlled contrast pair. Under this design, four color exhibits sharp performance collapse on Polar layouts, further corroborating our observation that Monotonic Path does not show this drop (Section 5).
- *Letter Collection:* This task was originally framed as multiple-choice. We observed that models could exploit an elimination strategy: identifying letters unique to each option, finding them on the image and verifying their positioning correctness, rather than genuinely tracing the path to collect letters. We reformulated the task to require sequential string output, forcing models to actually traverse the path and aggregate the collected letters in order. This adjustment caused a sharp performance decline, confirming that the original format permitted a reasoning shortcut. Notably, under the multiple-choice setting, increasing the length of candidate strings does not degrade performance, indicating that the difficulty of the underlying task is not really be evaluated in the multi-choice setting. After converting to string output format, we subsequently reduced task difficulty in multiple iterations to ensure performance remained sufficiently above the random baseline to yield informative signal.

B.5 Human Validation Study

As a final safeguard, we conducted a human validation study (pool of over 50 human raters) to confirm the quality of the benchmark data and simultaneously establish a human performance reference (Appendix C.5). For each of the 53 tasks, we randomly sampled 20 Cartesian–Polar paired instances (2,120 individual questions in total) and presented them independently to general-audience human raters, with the Cartesian and Polar versions of each pair assigned separately. For each instance, raters were required to evaluate three criteria and provide a complete step-by-step derivation:

1. **Visual clarity:** Is the rendered image visually clear and legible, such that all relevant elements (labels, shapes, paths, grid lines) can be unambiguously perceived?
2. **Logical correctness:** Is the task logic (rules, constraints, problem formulation) consistent and well-defined?
3. **Problem solving and solution trace:** Raters were required to independently solve the question and write out a detailed, logically complete step-by-step solution trace justifying their final answer (or select *I don't know* if unable to reach a solution with confidence).

To guarantee annotation quality, we employed a two-stage annotator and quality-checker protocol. An independent quality checker reviewed every submission to verify that the written reasoning chain was complete, self-contained, and consistent with the submitted answer; any submission with incomplete reasoning or unjustified claims or incomplete response was returned to the rater for revision until accepted. Finally, for all instances where a rater's answer differed from the ground truth (*Incorrect*) or

where the rater opted out (*I don't know*), the authors manually solved and re-audited each case. This audit confirmed that all such cases stemmed from rater reasoning errors or opt-outs on challenging problems rather than flaws in the benchmark, verifying the visual clarity, logical soundness, and ground-truth accuracy across all sampled instances.

C Evaluation Setup Details

C.1 Model Query Details

All models are evaluated via their publicly available APIs using default generation configurations to ensure optimal performance. We set the maximum output token limit to each model's maximum supported value. For the Gemini, GPT, Grok and Gemma model families, we query their respective first-party APIs. For all remaining models (e.g., Qwen, Kimi, Mistral), we access them through third-party serving platforms, specifically TogetherAI and OpenRouter.

Evaluated Models. Here is the full list of models we evaluated. For proprietary models, we include the Gemini family (*Gemini-3.1-Pro*, *Gemini-3-Flash*, *Gemini-3.1-Flash-Lite*, *Gemini-2.5-Pro*, *Gemini-2.5-Flash*) (Comanici et al., 2025; Gemini Team, 2025; Gemini Team & Google DeepMind, 2025; 2026), *GPT-5.2* (Singh et al., 2025), *Claude-Sonnet-4.6* (Anthropic, 2024), and *Grok-4* family (*Grok-4-0709*, *Grok-4-Fast-Reasoning*, *Grok-4-Fast-Non-Reasoning*) (xAI, 2025). For open-source models, we evaluate *Gemma-4-31B*, *Gemma-4-26B* (DeepMind, 2026), *Kimi-k2.5* (Team et al., 2026), *Qwen-3.5-397B-A17B* (Qwen Team, 2026), and *Mistral-Small-3.1-24B* (AI, 2025).

Reasoning Mode Configuration. For models supporting configurable reasoning, we evaluate under two inference settings: (1) a *high reasoning* mode that activates the model's highest supported reasoning budget, and (2) a *non-reasoning* mode that uses the minimal reasoning configuration. We note the following model-specific exceptions:

- **Gemini-3.1-Pro** (Gemini Team, 2025): Gemini-3.1-Pro does not support a fully non-reasoning setting. We therefore report its non-reasoning results under the low reasoning budget (reasoning set to LOW).
- **Claude-Sonnet-4.6** (Anthropic, 2024): Due to query timeouts under the extended high reasoning mode, the high reasoning results reported in this paper are obtained using the default adaptive high reasoning mode.
- **Grok-4-Fast-Reasoning** (xAI, 2025): We pair Grok-4-Fast-Reasoning and Grok-4-Fast-Non-Reasoning as the high reasoning and non-reasoning variants, respectively, in Table 2.

C.2 Evaluation Setup for the Cartesian Shortcut

Here we provide implementation details for the empirical analyses presented in Section 2.

Chain-of-Thought Discretization Analysis. To quantify how frequently models discretize visual layouts into textual coordinates during reasoning, we conduct a Chain-of-Thought (CoT) analysis across 9 prominent visual reasoning benchmarks: BabyVision (Chen et al., 2026), OmniSpatial (Jia et al., 2026), EMMA (Hao et al., 2025), MegaBench (Chen et al., 2025), MMMU-Pro (Yue et al., 2025), BLINK (Fu et al., 2024), CharXiv (Wang et al., 2024b), MUIRBench (Wang et al., 2024a), and MMIU (Meng et al., 2024). We use Gemini-3-Flash as a representative model. To isolate synthetic, grid-based visual reasoning tasks, we first prompt the model to filter out photographic images and examples with domain-specific backgrounds (e.g.,, medical imaging, charts), yielding over 3,800 qualifying examples. We then evaluate these examples under the high reasoning mode of Gemini-3-Flash and collect the intermediate Chain-of-Thought outputs. To systematically detect spatial discretization, we provide the generated reasoning traces to Gemini-3.1-Pro as a judge, prompting it to identify whether the reasoning explicitly invokes Cartesian coordinates or axis-aligned positional references (e.g.,, "row 2", "column 3", "(x,y)"). Results show that 56.58% of examples exhibit explicit Cartesian coordinate usage in their reasoning traces, confirming the prevalence of the Cartesian Shortcut.

Two-Stage Perception-Reasoning Disentanglement. To further isolate the role of visual perception from deductive reasoning, we conduct a two-stage experiment on Gemini-3.1-Pro (Figure 1). In the first stage, we provide only the task image and prompt the model to generate a detailed descriptive caption, explicitly informing the model that this caption will subsequently be used to answer a question. In the second stage, we provide the original question together with the generated caption (without the image) and prompt the model to answer based solely on the textual description. To evaluate caption quality, we use the grid dimensions as a proxy for perceptual accuracy: we extract the ground-truth grid size from each example's metadata and prompt Gemini-3.1-Pro to verify whether the caption correctly reports the grid dimensions. We clarify that comprehensively evaluating caption fidelity would require manual human inspection; grid size verification serves as a coarse but scalable proxy for perceptual accuracy rather than a precise measure. A caption is marked as incorrect if it explicitly mentions a grid size that contradicts the ground truth. As reported in Figure 1, grid size extraction accuracy reaches 80.2% on Cartesian layouts but drops to only 20.8% on Polar layouts. Crucially, when evaluating task accuracy exclusively on the subset of correctly perceived instances, Cartesian examples maintain 86.9% accuracy while Polar examples collapse to 47.6%, demonstrating that the performance gap persists even when visual perception is controlled for.

C.3 Evaluation Prompt Template

```
Please answer the following question based on the image.
Output **only the final correct answer directly**, nothing else is allowed in the
output. e.g., for multi-choise, it can only contain option letter (e.g., A). For
text answers, it can only contain the required answer values.


[Question]
{Question}
<img>
```

Figure 9: Prompt template used in the standard evaluation in our experiments.

```
Please answer the following question based on the image.
Put your analysis before the correct answer. In the last line, make sure provide
only the correct answer, nothing else is allowed in last line. **Last line must
only contain correct answer as question asked.**

[Task Hint]
The image displays a polar (radial) layout. This is topologically equivalent to a
standard rectangular grid: each concentric ring corresponds to a row, and each
angular sector corresponds to a column. To solve this problem, first reconstruct
the equivalent Cartesian grid by reading the content of each cell (ring, sector)
and mapping it into the corresponding (row, column) position in a matrix. Then,
solve the problem on this reconstructed matrix using standard
grid-based reasoning.


[Question]
{Question}
<img>
```

Figure 10: Prompt template used in conversion hint experiment.

All tasks across both Cartesian and Polar layouts share a unified evaluation prompt template to ensure fair comparison. The template consists of: (1) a task-specific narrative providing contextual framing and rules, (2) the visual input image, and (3) the question with its answer format specification. The exact prompt template is provided in Figure 9.

For the prompting interventions described in Section 5, we employ two additional prompt variants:

```
You are a visual reasoning expert. Below are examples of how to analyze and solve
specific spatial tasks.

[Examples]
{few_shot_text}


[Task Instruction]
Please solve the following new question based on the provided question image.
1. Think step-by-step about the spatial layout and rules. Use above examples as
reference.
2. Output your final answer clearly on the very last line.

Ensure your response strictly follows this structure:
<Analysis>
(Your step-by-step reasoning here)
</Analysis>
(Your final exact answer here, on the absolute last line)


[Question]
{Question}
<img>
```

Figure 11: Prompt template used in the few-shot in context learning.

- **Conversion Hint:** The prompt is augmented with an explicit instruction asking the model to first re-map the Polar layout into an equivalent Cartesian grid representation before attempting to solve the task.
- **5-Shot In-Context Learning:** The prompt is prepended with five solved Polar examples from the same task, each consisting of the Polar image, the question, and the correct answer, to scaffold the model's reasoning on Polar layouts.

The exact prompts for these two variants are provided in Figure 10 and Figure 11 respectively..

C.4 Evaluation Metrics

All tasks in the Polaris-Bench are designed with deterministic, well-defined answer formats, enabling reliable automated evaluation. Depending on the task, ground-truth answers take one of the following standardized forms: multiple-choice labels, exact integers, coordinate pairs or axis positions, ordered lists, and directional sequences, as stated in Section A.1. In this case, we employ rule-based programmatic parsers as the primary evaluation method. For each answer format, a dedicated parser extracts the model's predicted answer from its response and performs an exact-match comparison against the ground truth. Across all evaluated models, the vast majority of responses are successfully parsed and evaluated through this automated pipeline. For responses where programmatic parsing fails, typically due to models not following the specified output format instructions, we apply an LLM-as-a-judge (Zheng et al., 2023; Gu et al., 2024) as a calibration fallback. Specifically, we prompt Gemini-3.1-Pro with both the ground-truth answer and the model's raw response, and ask it to determine whether the response is correct. This two-tier evaluation strategy ensures that models are not unfairly penalized for minor formatting deviations while maintaining rigorous evaluation standards.

Random Baseline. We additionally report a theoretical random baseline computed from the statistical expectation of each task's answer format. For multiple-choice tasks, this corresponds to $1/k$ where k is the number of options; for integer and coordinate tasks, the baseline is derived from the uniform distribution over the valid answer space. This baseline serves as a lower-bound reference to contextualize model performance.

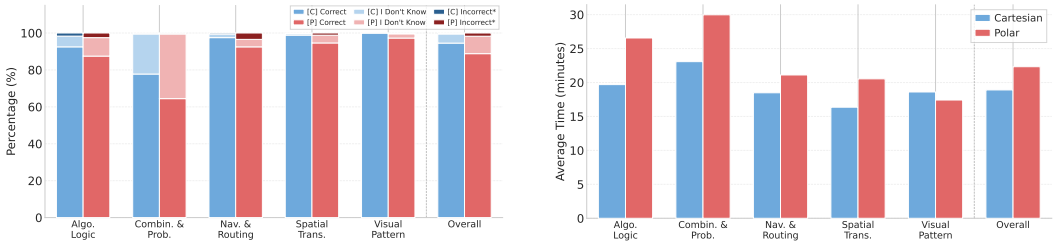
LLM-as-a-judge. Across all evaluated models, the overall fallback rate to the LLM judge (Gemini-3.1-Pro) is 14.47% (e.g., Gemini-3.1-Pro: 5.0%, Grok-4: 5.1%, Qwen3.5-397B-A17B: 9.7%, Gemma-

4-26B: 15.3%, GPT-5.2: 17.8%, Claude-Sonnet-4.6: 29.6%, and Kimi-k2.5: 47.3%), primarily reflecting differences in how strictly each model adheres to the requested output format; notably, Gemini-3.1-Pro itself exhibits one of the lowest fallback rates, meaning the judge is rarely invoked on its own outputs. The responses routed to the fallback judge are predominantly surface-level format mismatches—for instance, ground truth "7" versus model output "seven", "(18,2)" versus "18 and 2", or "C: 62cm" versus "62cm", where the prediction is semantically correct but fails string-level exact matching. Because the judge only needs to determine whether the predicted answer semantically matches a deterministic ground truth, this verification task lies well within the reliable regime of frontier LLMs. To empirically validate the judge's reliability, we applied it to model predictions already scored by exact matching; across three independent runs on GPT-5.2's outputs those of which could be evaluated by exact match, the judge achieved 100% agreement with exact-match scoring, and random human spot-checks on fallback outputs identified no grading errors. Combining this low self-fallback rate, the 100% validation agreement, and the observation that Gemini-3.1-Pro exhibits the largest Cartesian-to-Polar accuracy drop among all evaluated models (Table 1), demonstrating that this fallback mechanism does not introduce systematic bias favoring the judge model.

C.5 Human Annotation Baseline

Table 8: Human performance and time cost on Polaris-Bench by task category. $-\Delta Acc$ denotes the accuracy gap between Cartesian and Polar.

	Correct (%)		I Don't Know (%)		Incorrect (%)			Avg. Time (min)	
Category	C	P	C	P	C	P	$-\Delta$ Acc.	C	P
Algo. Logic	92.5	87.5	5.8	10.0	1.7	2.5	-5.0	19.7	26.6
Combin. & Prob.	77.8	64.4	21.7	35.0	0.6	0.6	-13.3	23.1	30.0
Nav. & Routing	97.5	92.5	1.9	4.1	0.6	3.4	-5.0	18.5	21.1
Spatial Trans.	98.8	94.6	0.8	4.2	0.4	1.2	-4.2	16.4	20.5
Visual Pattern	100.0	97.3	0.0	2.3	0.0	0.5	-2.7	18.6	17.4
Overall	94.5	88.8	4.9	9.5	0.5	1.7	-5.7	18.9	22.3



(a) (b)

Figure 12: Human performance on Polaris-Bench across task categories. (a) Answer correctness distribution, with stacked segments showing Correct, I Don't Know, and Incorrect rates for Cartesian (blue) and Polar (red) layouts. (b) Average annotation time per instance in minutes.

To establish a human performance reference, we analyze rater performance from our human validation study (Appendix B.5), covering 20 randomly sampled Cartesian–Polar question pairs for each of the 53 tasks in Polaris-Bench. Raters were general-audience annotators without specialized STEM backgrounds, and Cartesian and Polar versions were evaluated separately. For unfamiliar problem types (e.g., Sudoku validity checking, N-Queens constraint verification), raters were permitted to perform basic web searches to understand task rules, but were required to independently derive and write a complete step-by-step solution trace. Raters were instructed to submit a final answer only when fully confident in their derivation, and to select "I don't know" otherwise. Since author verification confirmed the ground-truth validity of all sampled instances (Appendix B.5), instances where a rater's complete derivation arrived at the ground truth are recorded as "Correct," while "Incorrect" and "I don't know" reflect rater solving errors and opt-outs, respectively.

Table 8 and Figure 12 report the results by task category. Overall, human raters achieve 94.5% accuracy on Cartesian instances and 88.8% on Polar instances, representing a modest 5.7 percentage

point reduction, while the "I don't know" rate increases moderately from 4.9% to 9.5%. The average solving time is 18.9 minutes per Cartesian instance and 22.3 minutes per Polar instance; these reported times measure the rater's active problem-solving duration on the final accepted submission, excluding quality-checker review and intermediate revision cycles. Across individual categories, the Cartesian-to-Polar accuracy gap remains consistently small. *Visual Pattern Matching* maintains near-ceiling accuracy on both layouts (100.0% and 97.3%) and requires less time on Polar instances (17.4 vs. 18.6 minutes), aligning with the intuition that visual pattern recognition is a fast, high-confidence process for humans. Conversely, *Combinatorics and Probability* exhibits the lowest human accuracy (77.8% and 64.4%) and the highest "I don't know" rates (21.7% and 35.0%), reflecting the demanding mathematical counting and enumeration required for general-audience raters. Even in this most challenging category, however, the Cartesian-to-Polar gap remains moderate (-13.3 points) compared to the severe degradation seen in models.

Overall, human performance exhibits only a minor accuracy reduction and a modest 3.4-minute increase in solving time when transitioning from Cartesian to Polar layouts, in stark contrast to the 30–47 percentage point collapse observed across frontier MLLMs (Table 1). This provides a clear human reference point contextualizing the model evaluation results in Section 4.

C.6 Evaluation Results Statistics

All results reported in this work are based on a single evaluation run per model, because repeated runs are very costly. The 53 tasks in Polaris-Bench require complex visual reasoning, and under high reasoning mode models often produce long reasoning traces, so a single query can take several minutes and consume a large output token budget. Two situations required re-querying. First, when a response reached the maximum output token limit without a final answer, we re-ran that instance under the same setting. Second, some API requests failed (e.g., server errors or timeouts). If fewer than 5 of the 100 instances of a task failed, we computed accuracy over the completed responses; otherwise, we re-sent the failed requests. Both rules were applied in the same way to Cartesian and Polar instances.

Spot-check verification. To estimate run-to-run variation, we repeated the evaluation of selected models on randomly sampled tasks under identical settings. Task-level accuracy varied by a few percentage points (approximately within 3) across runs. For example, two repeated runs of Gemini-3.1-Pro on Four Color gave 92% and 90% on Cartesian and 31% and 30% on Polar.

The following factors further support the reliability of our single-run results:

- **Large sample size.** Each task has 100 paired Cartesian–Polar instances, and aggregate results are computed over 5,300 instances per coordinate condition, which reduces instance-level noise.
- **Large effect sizes.** For frontier models, the Cartesian-to-Polar drops range from 37 to 47 percentage points (Table 1). Run-to-run differences of a few points at the task level, as observed in our spot checks, are small relative to these gaps and do not change the conclusions in the main text.
- **Cross-model consistency.** Every evaluated model scores lower on Polar than on Cartesian layouts, across both proprietary and open-source models. The only negligible drop is Mistral-Small-2503 (0.4 points), which is close to the random baseline on both layouts.
- **Consistency across conditions.** For frontier models, the drop appears in all five task categories (Table 1). For all six models in Figure 5, it appears in both fully and partially aligned tasks and under both bounded and wrapping conditions.

D Complete Experimental Results

D.1 Overall Performance Analysis

We provide fine-grained radar chart visualizations to complement the aggregate results reported in Table 1 and Table 2. These charts display per-category accuracy across the five task taxonomies (i.e., visual pattern matching, spatial trasnformation and geometry, navigation and routing, combinatorics and probability, and algorithmic logic and simulation) enabling direct visual comparison of model performance profiles.

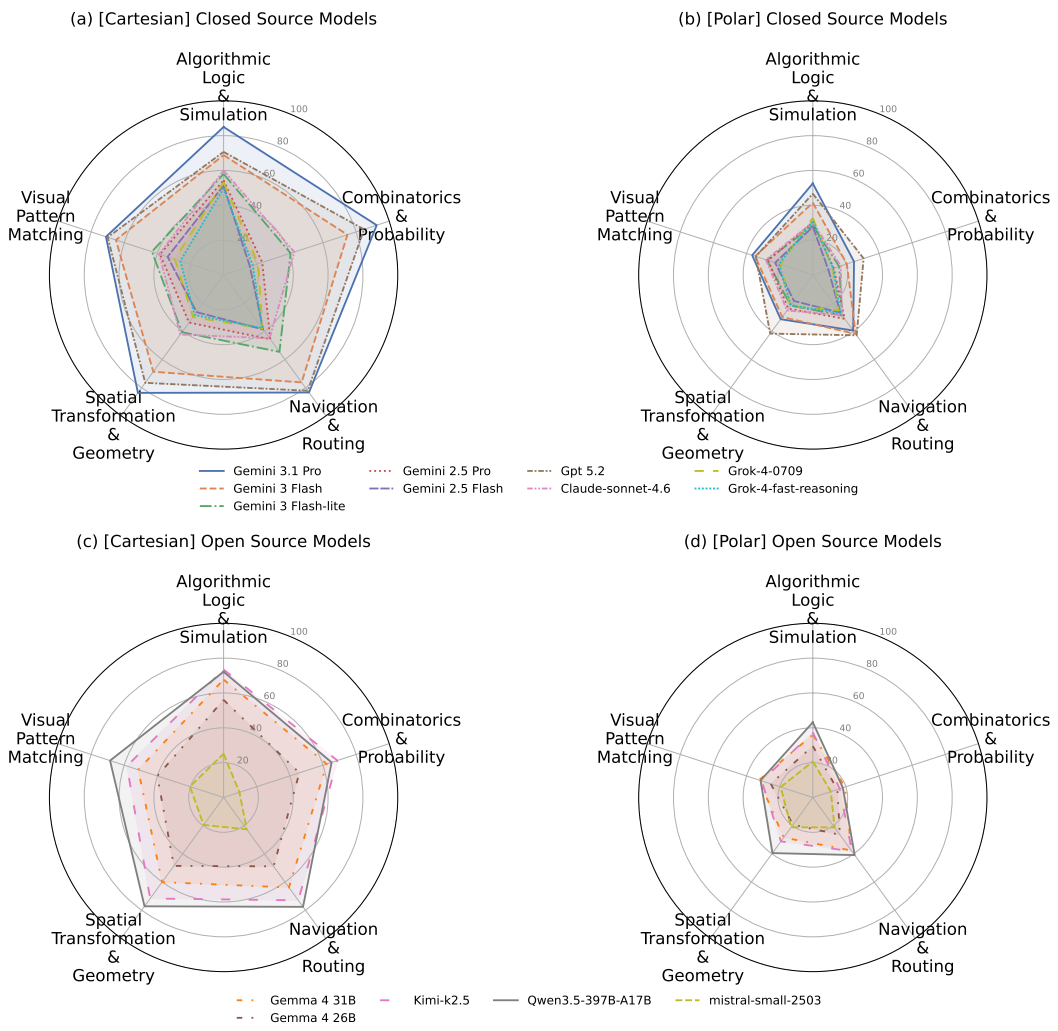


Figure 13: Per-category radar chart comparison of all evaluated models under high reasoning mode, corresponding to Table 1. *(a)* Closed-source models on Cartesian layouts. *(b)* Closed-source models on Polar layouts. *(c)* Open-source models on Cartesian layouts. *(d)* Open-source models on Polar layouts. Each axis represents one of the five task categories, and the radial scale indicates accuracy (%).

Figure 13 presents the per-category performance of all evaluated models under high reasoning mode, separated into closed-source (top row) and open-source (bottom row) groups. The left column shows Cartesian accuracy and the right column shows Polar accuracy. On Cartesian layouts, closed-source models exhibit relatively expanded and overlapping radar profiles, with Gemini-3.1-Pro achieving the largest coverage. On Polar layouts, all radar profiles contract substantially and cluster closer to the center. Among open-source models, a similar contraction pattern is observed, with Polar profiles notably smaller and more compressed than their Cartesian counterparts across all five categories.

Figures 14 and 15 present the effect of reasoning mode on per-category performance for closed-source and open-source models, respectively. Each figure contains four panels: the top row shows non-reasoning mode results on Cartesian (left) and Polar (right) layouts, while the bottom row shows the corresponding high reasoning mode results. For closed-source models (Figure 14), the radar profiles expand visibly from non-reasoning to reasoning on Cartesian layouts, whereas the expansion on Polar layouts is comparatively modest. For open-source models (Figure 15), the reasoning mode similarly produces larger radar profiles on Cartesian layouts, while the Polar profiles remain relatively compact across both reasoning settings.

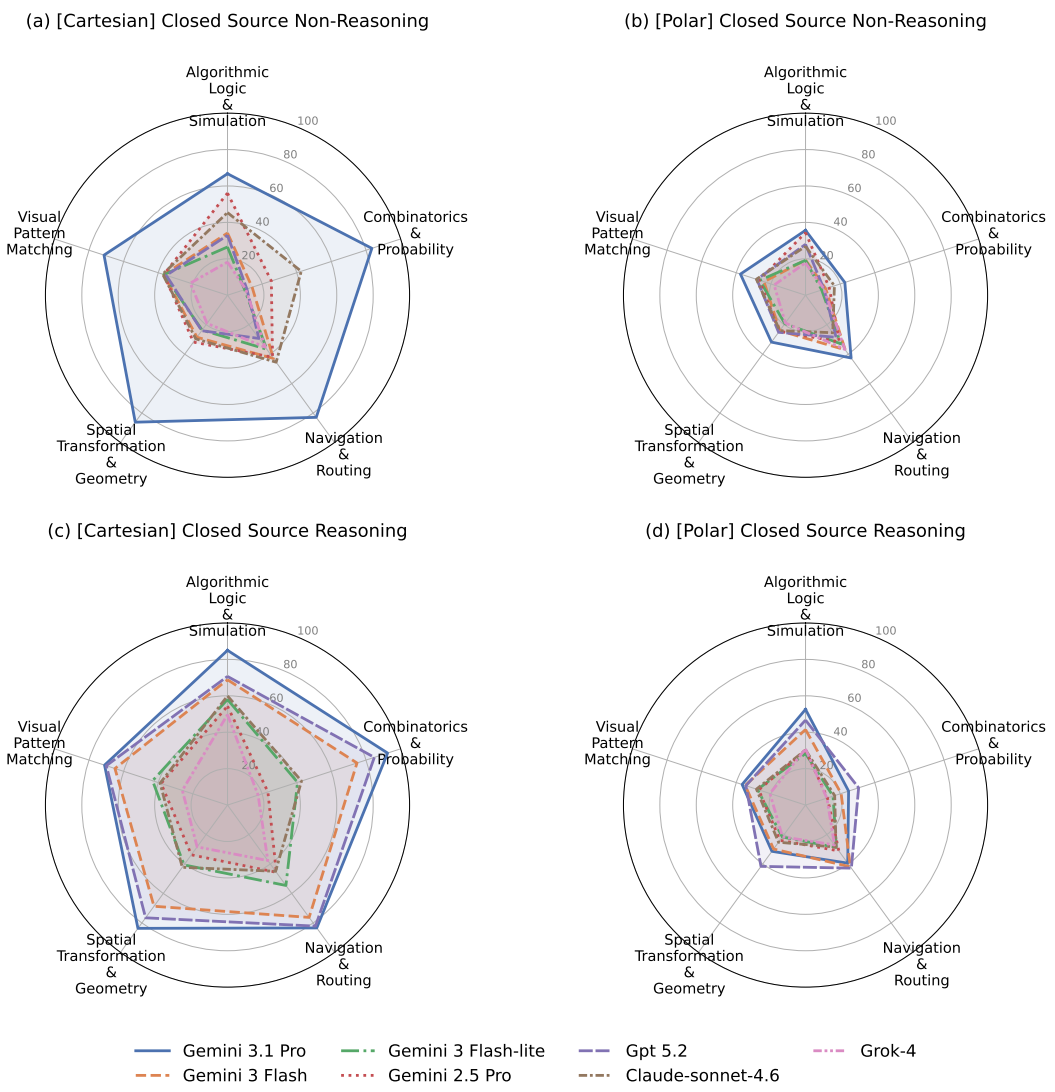


Figure 14: Effect of reasoning mode on per-category performance for closed-source models, corresponding to Table 2. *(a)* Non-reasoning mode on Cartesian layouts. *(b)* Non-reasoning mode on Polar layouts. *(c)* High reasoning mode on Cartesian layouts. *(d)* High reasoning mode on Polar layouts.

Figure 16 further disaggregates model performance by answer type rather than task category, with axes corresponding to Coordinate, Digit, String, Option Label, and List. On Cartesian layouts, both closed-source and open-source models exhibit relatively expanded radar profiles with visible inter-model variation across answer types. Upon transition to Polar layouts, all profiles contract substantially, and the degree of contraction varies across answer types, indicating that the Cartesian-to-Polar performance gap is not uniform across output formats. Among open-source models, the Polar profiles are particularly compressed, with reduced inter-model differentiation compared to their Cartesian counterparts.

D.2 Per-Catagory Breakdown Performance

We report detailed per-task performance for all five task categories, with results organized by subcategory where applicable.

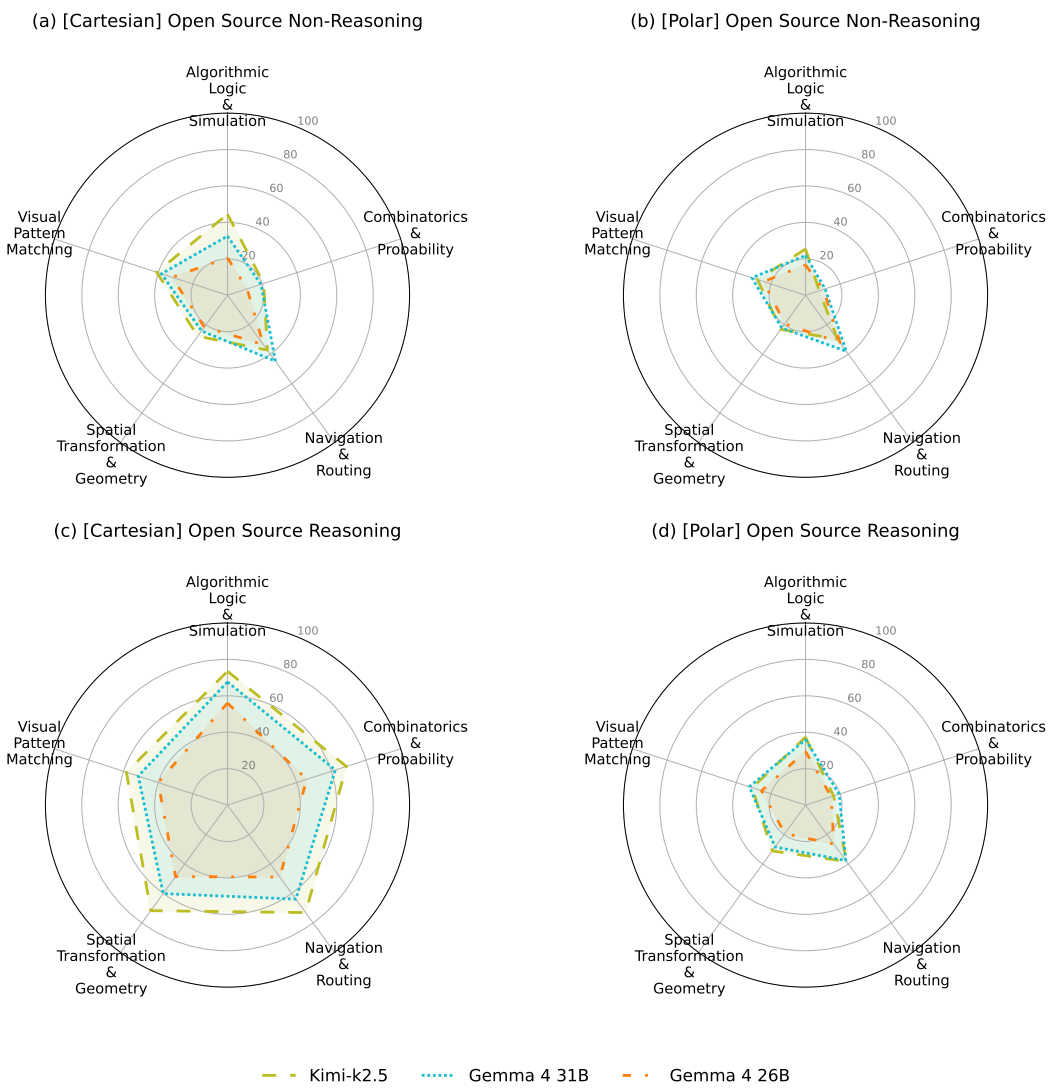


Figure 15: Effect of reasoning mode on per-category performance for open-source models, corresponding to Table 2. *(a)* Non-reasoning mode on Cartesian layouts. *(b)* Non-reasoning mode on Polar layouts. *(c)* High reasoning mode on Cartesian layouts. *(d)* High reasoning mode on Polar layouts.

Navigation and Routing. Tables 9, 10, and 11 present the per-task results under this category, grouped by the Rule-Based Navigation, Optimal Pathfinding, and Constraint-Based Routing subcategories, respectively.

Visual Pattern Matching. Tables 12 and 13 present the per-task results, with the latter merging subcategories that contain fewer tasks into a single table.

Spatial Transformation and Geometry. Tables 14, 15, and 16 present the per-task results, grouped by the Geometric Measurement and Counting, Spatial Constraints and Topology, and Rotations and Reflections subcategories, respectively.

Algorithmic Logic and Simulation. Table 17 presents the per-task results for this category. Due to the small number of tasks per subcategory, all results are consolidated into a single table.

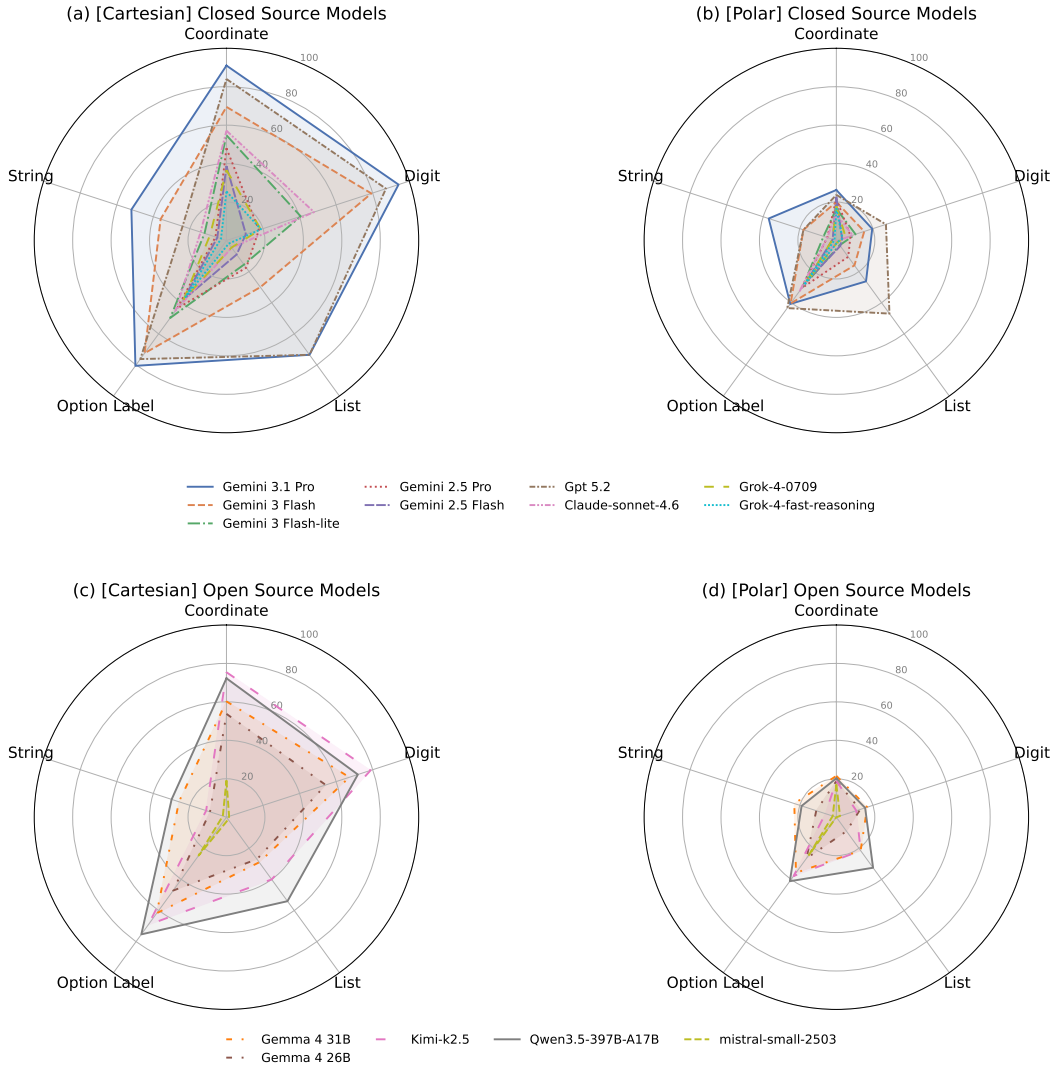


Figure 16: Per-answer-type radar chart comparison of all evaluated models under high reasoning mode. *(a)* Closed-source models on Cartesian layouts. *(b)* Closed-source models on Polar layouts. *(c)* Open-source models on Cartesian layouts. *(d)* Open-source models on Polar layouts. Each axis represents one of the five answer types (Coordinate, Digit, String, Option Label, List), and the radial scale indicates accuracy (%).

Table 9: Per-task results under the Navigation and Routing category, subcategory Rule-Based Navigation. All models are evaluated under high reasoning mode. C and P denote Cartesian and Polar accuracy (%), respectively. $(-\Delta)$ indicates the accuracy change from Cartesian to Polar. Task abbreviations: Abs. Nav. = Absolute Navigation; Ego. Nav. = Egocentric Navigation; Rule Nav. = Rule-Based Navigation; Wall Fol. = Wall Follower; Wrap. Nav. = Wrapping Navigation.

Model	Abs. Nav.			Ego. Nav.			Rule Nav.			Wall Fol.			Wrap. Nav.		
	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$
Human	90.0	75.0	(−15.0)	90.0	75.0	(−15.0)	100.0	60.0	(−40.0)	95.0	75.0	(−20.0)	100.0	100.0	(0.0)
Random	16.7	16.7	(0.0)	16.7	16.7	(0.0)	20.0	20.0	(0.0)	20.0	20.0	(0.0)	16.7	16.7	(0.0)
Closed-Source MLLMs															
Gemini-3.1-Pro	96.0	31.0	(−65.0)	94.0	32.0	(−62.0)	95.0	12.0	(−83.0)	89.0	52.5	(−36.5)	99.0	24.5	(−74.5)
Gemini-3-Flash	81.0	16.0	(−65.0)	76.0	21.0	(−55.0)	88.0	29.0	(−59.0)	63.0	19.0	(−44.0)	76.0	25.0	(−51.0)
Gemini-3-Flash-lite	47.0	15.0	(−32.0)	34.0	10.0	(−24.0)	80.0	17.0	(−63.0)	48.0	31.0	(−17.0)	40.0	19.0	(−21.0)
Gemini-2.5-Pro	23.0	16.0	(−7.0)	13.0	7.0	(−6.0)	85.0	15.0	(−70.0)	48.0	47.0	(−1.0)	36.0	25.0	(−11.0)
Gemini-2.5-Flash	20.0	13.0	(−7.0)	8.0	12.0	(4.0)	64.0	16.0	(−48.0)	52.0	50.0	(−2.0)	15.0	15.0	(0.0)
GPT-5.2	91.0	29.0	(−62.0)	71.0	27.0	(−44.0)	96.0	32.0	(−64.0)	88.0	29.0	(−59.0)	93.0	36.0	(−57.0)
Claude-Sonnet-4.6	75.3	23.2	(−52.1)	43.8	17.4	(−26.5)	24.7	19.8	(−5.0)	42.7	23.7	(−19.0)	52.0	21.2	(−30.8)
Grok-4-0709	19.0	13.0	(−6.0)	9.0	6.0	(−3.0)	61.6	10.0	(−51.6)	47.0	42.0	(−5.0)	16.0	16.0	(0.0)
Grok-4-Fast-Reasoning	25.0	25.0	(0.0)	21.0	28.0	(7.0)	56.0	31.0	(−25.0)	23.0	23.0	(0.0)	21.0	28.0	(7.0)
Open-Source MLLMs															
Gemma-4-31B	75.0	15.0	(−60.0)	52.0	22.0	(−30.0)	75.0	12.0	(−63.0)	55.0	48.0	(−7.0)	71.0	28.0	(−43.0)
Gemma-4-26B	52.0	8.0	(−44.0)	13.1	9.0	(−4.1)	66.0	11.0	(−55.0)	44.0	37.0	(−7.0)	59.0	10.0	(−49.0)
Kimi-k2.5	94.9	27.4	(−67.5)	93.0	16.3	(−76.7)	83.0	26.0	(−57.0)	53.0	28.0	(−25.0)	95.0	26.0	(−69.0)
Qwen3.5-397B-A17B	95.0	17.0	(−78.0)	95.0	25.0	(−70.0)	80.0	14.0	(−66.0)	73.0	40.0	(−33.0)	91.0	33.0	(−58.0)
Mistral-Small-2503	23.9	21.3	(−2.6)	6.1	8.1	(2.0)	7.0	17.0	(10.0)	51.0	51.0	(0.0)	20.0	34.0	(14.0)

Table 10: Per-task results under the Navigation and Routing category, subcategory Optimal Pathfinding. All models are evaluated under high reasoning mode. C and P denote Cartesian and Polar accuracy (%), respectively. $(-\Delta)$ indicates the accuracy change from Cartesian to Polar. Task abbreviations: Bound. Path = Bounded Path Finding; Long. Path = Longest Path; Short. Path = Shortest Path; Wrap. Path = Wrapping Path Finding.

Model	Bound. Path			Long. Path			Maze			Short. Path			Wrap. Path		
	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$
Human	95.0	100.0	(5.0)	100.0	100.0	(0.0)	100.0	100.0	(0.0)	100.0	100.0	(0.0)	95.0	100.0	(5.0)
Random	20.0	20.0	(0.0)	20.0	20.0	(0.0)	31.3	31.3	(0.0)	20.0	20.0	(0.0)	20.0	20.0	(0.0)
Closed-Source MLLMs															
Gemini-3.1-Pro	98.0	37.0	(−61.0)	78.0	55.0	(−23.0)	34.0	37.5	(3.5)	69.7	55.6	(−14.1)	98.0	48.0	(−50.0)
Gemini-3-Flash	86.0	63.0	(−23.0)	77.0	56.0	(−21.0)	37.5	40.5	(3.0)	93.0	73.0	(−20.0)	92.0	74.0	(−18.0)
Gemini-3-Flash-lite	64.0	29.0	(−35.0)	56.0	48.0	(−8.0)	37.0	35.5	(−1.5)	68.0	70.0	(2.0)	51.0	27.0	(−24.0)
Gemini-2.5-Pro	33.0	23.0	(−10.0)	69.0	46.0	(−23.0)	32.0	37.5	(5.5)	65.0	54.0	(−11.0)	29.0	18.0	(−11.0)
Gemini-2.5-Flash	29.0	21.0	(−8.0)	48.0	32.0	(−16.0)	39.5	39.0	(−0.5)	46.0	43.0	(−3.0)	31.0	20.0	(−11.0)
GPT-5.2	86.0	31.0	(−55.0)	80.0	69.0	(−11.0)	44.5	36.5	(−8.0)	82.0	87.0	(5.0)	97.0	32.0	(−65.0)
Claude-Sonnet-4.6	59.8	49.0	(−10.8)	48.5	44.4	(−4.0)	34.7	33.7	(−1.0)	65.0	56.1	(−8.9)	56.0	54.1	(−1.9)
Grok-4-0709	20.0	25.0	(5.0)	43.4	40.8	(−2.6)	39.5	32.0	(−7.5)	40.0	43.0	(3.0)	29.0	33.0	(4.0)
Grok-4-Fast-Reasoning	29.0	25.0	(−4.0)	37.0	40.0	(3.0)	33.5	39.0	(5.5)	41.0	40.0	(−1.0)	22.0	13.0	(−9.0)
Open-Source MLLMs															
Gemma-4-31B	72.0	41.0	(−31.0)	71.0	50.0	(−21.0)	42.0	34.0	(−8.0)	62.0	66.0	(4.0)	72.0	43.0	(−29.0)
Gemma-4-26B	48.0	20.2	(−27.8)	48.0	26.0	(−22.0)	27.5	35.0	(7.5)	49.0	52.0	(3.0)	43.4	30.0	(−13.4)
Kimi-k2.5	93.0	47.0	(−46.0)	68.0	40.8	(−27.2)	43.2	35.7	(−7.5)	68.0	65.0	(−3.0)	90.0	51.0	(−39.0)
Qwen3.5-397B-A17B	92.0	52.0	(−40.0)	76.0	57.0	(−19.0)	38.5	42.5	(4.0)	66.0	58.0	(−8.0)	88.0	45.0	(−43.0)
Mistral-Small-2503	17.2	8.1	(−9.1)	30.0	28.0	(−2.0)	53.5	26.5	(−27.0)	40.0	40.0	(0.0)	19.0	16.0	(−3.0)

Combinatorics and Probability. Tables 18 and 19 present the per-task results, with Table 18 covering the Bounded Combinatorics subcategory and Table 19 merging the Stochastic Processes and Topological Combinatorics subcategories.

Notably, while the overall Cartesian-to-Polar accuracy gap remains substantial across all models, each model exhibits a small subset of tasks on which its performance drop is comparatively moderate relative to other models evaluated on the same task. These task-level exceptions are largely *model-specific*: the particular tasks on which a given model shows a smaller gap differ considerably from one model to another, with little consistent overlap across the evaluated model set. We think this may stem from idiosyncratic biases in individual models' training distributions. Moreover, the rare cases in which Polar accuracy approaches Cartesian accuracy almost exclusively occur when both values

Table 11: Per-task results under the Navigation and Routing category, subcategory Constraint-Based Routing. All models are evaluated under high reasoning mode. C and P denote Cartesian and Polar accuracy (%), respectively. $(-\Delta)$ indicates the accuracy change from Cartesian to Polar. Task abbreviations: Lrg. Num. = Largest Number Path; Mono. Path = Monotonic Path; Turn Cnt. = Turn Counting; Word Srch. = Word Search.

Model	Lrg. Num.			Mono. Path			Turn Cnt.			Word Srch.		
	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$
Human	100.0	100.0	(0.0)	100.0	100.0	(0.0)	100.0	100.0	(0.0)	95.0	95.0	(0.0)
Random	20.0	20.0	(0.0)	16.7	16.7	(0.0)	20.0	20.0	(0.0)	0.0	0.0	(0.0)
Closed-Source MLLMs												
Gemini-3.1-Pro	71.0	29.0	(−42.0)	89.0	91.0	(2.0)	57.0	27.0	(−30.0)	100.0	20.0	(−80.0)
Gemini-3-Flash	48.0	26.0	(−22.0)	87.0	94.0	(7.0)	65.0	23.0	(−42.0)	98.0	27.0	(−71.0)
Gemini-3-Flash-lite	34.0	22.0	(−12.0)	90.0	52.0	(−38.0)	18.0	13.0	(−5.0)	97.0	14.0	(−83.0)
Gemini-2.5-Pro	27.0	23.0	(−4.0)	89.0	80.0	(−9.0)	37.0	21.0	(−16.0)	53.0	23.0	(−30.0)
Gemini-2.5-Flash	28.0	23.0	(−5.0)	90.0	63.0	(−27.0)	21.0	21.0	(0.0)	50.0	7.0	(−43.0)
GPT-5.2	69.0	29.0	(−40.0)	84.0	93.0	(9.0)	71.0	33.0	(−38.0)	98.0	36.0	(−62.0)
Claude-Sonnet-4.6	25.3	19.0	(−6.3)	1.0	0.0	(−1.0)	29.0	24.2	(−4.8)	71.6	13.4	(−58.2)
Grok-4-0709	17.0	22.0	(5.0)	82.0	45.9	(−36.1)	16.0	7.0	(−9.0)	92.0	13.0	(−79.0)
Grok-4-Fast-Reasoning	23.0	18.0	(−5.0)	86.0	43.0	(−43.0)	21.0	22.0	(1.0)	89.0	4.0	(−85.0)
Open-Source MLLMs												
Gemma-4-31B	31.0	22.0	(−9.0)	84.0	90.0	(6.0)	36.0	17.0	(−19.0)	96.0	39.0	(−57.0)
Gemma-4-26B	27.0	27.0	(0.0)	89.0	77.0	(−12.0)	20.2	14.0	(−6.2)	98.0	14.1	(−83.9)
Kimi-k2.5	35.7	24.0	(−11.7)	83.0	87.0	(4.0)	25.0	18.0	(−7.0)	96.9	43.4	(−53.5)
Qwen3.5-397B-A17B	68.0	34.0	(−34.0)	82.0	94.0	(12.0)	54.0	21.0	(−33.0)	88.0	40.0	(−48.0)
Mistral-Small-2503	21.0	19.0	(−2.0)	0.0	0.0	(0.0)	23.0	16.0	(−7.0)	5.0	12.0	(7.0)

Table 12: Per-task results under the Visual Pattern Matching category, subcategory Visual Spotting and Search / Pattern Continuation. All models are evaluated under high reasoning mode. C and P denote Cartesian and Polar accuracy (%), respectively. $(-\Delta)$ indicates the accuracy change from Cartesian to Polar. Task abbreviations: Anom. Det. = Anomaly Detection; Letter Col. = Letter Collection; Odd Piece = Odd Piece Out; Pat. Pred. = Pattern Prediction.

Model	Anom. Det.			Letter Col.			Odd Piece			Pat. Pred.		
	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$
Human	100.0	85.0	(−15.0)	100.0	100.0	(0.0)	100.0	100.0	(0.0)	100.0	95.0	(−5.0)
Random	0.9	1.0	(0.0)	0.0	0.0	(0.0)	20.0	20.0	(0.0)	19.7	19.6	(−0.0)
Closed-Source MLLMs												
Gemini-3.1-Pro	83.0	10.0	(−73.0)	52.0	37.0	(−15.0)	69.0	42.0	(−27.0)	90.0	27.3	(−62.7)
Gemini-3-Flash	78.0	4.0	(−74.0)	36.0	18.0	(−18.0)	65.0	36.0	(−29.0)	83.0	31.0	(−52.0)
Gemini-3-Flash-lite	57.0	2.0	(−55.0)	12.0	7.0	(−5.0)	37.0	40.0	(3.0)	32.0	26.0	(−6.0)
Gemini-2.5-Pro	27.0	0.0	(−27.0)	6.0	5.0	(−1.0)	36.0	46.0	(10.0)	37.0	17.0	(−20.0)
Gemini-2.5-Flash	20.0	1.0	(−19.0)	5.0	2.0	(−3.0)	36.0	41.0	(5.0)	26.0	29.0	(3.0)
GPT-5.2	73.0	2.0	(−71.0)	24.0	18.0	(−6.0)	88.0	49.0	(−39.0)	54.0	13.0	(−41.0)
Claude-Sonnet-4.6	74.8	1.0	(−73.7)	14.1	5.1	(−9.1)	26.0	36.0	(10.0)	19.0	14.0	(−5.0)
Grok-4-0709	16.2	2.1	(−14.0)	9.0	2.0	(−7.0)	26.0	29.0	(3.0)	26.0	23.0	(−3.0)
Grok-4-Fast-Reasoning	6.0	0.0	(−6.0)	3.0	1.0	(−2.0)	29.0	23.0	(−6.0)	22.0	34.0	(12.0)
Open-Source MLLMs												
Gemma-4-31B	63.0	5.0	(−58.0)	26.0	23.0	(−3.0)	39.0	32.0	(−7.0)	41.0	21.0	(−20.0)
Gemma-4-26B	63.0	4.1	(−58.9)	9.0	11.0	(2.0)	26.3	29.0	(2.7)	17.0	14.1	(−2.9)
Kimi-k2.5	75.0	4.0	(−71.0)	11.1	6.0	(−5.1)	53.0	47.0	(−6.0)	19.0	13.0	(−6.0)
Qwen3.5-397B-A17B	78.0	2.0	(−76.0)	30.0	19.0	(−11.0)	74.0	37.0	(−37.0)	49.0	24.0	(−25.0)
Mistral-Small-2503	14.1	0.0	(−14.1)	2.0	2.0	(0.0)	27.0	38.0	(11.0)	18.0	27.0	(9.0)

are near the random baseline, indicating that the model is effectively guessing under both layouts rather than maintaining high accuracy across the layout change.

D.3 Experimental Details for Discussion

Below we provide more details of the experiments discussed in Section 5.

Prompting interventions. We randomly select six tasks from those exhibiting large Cartesian-to-Polar accuracy gaps: Bouncing Point, Edge Counting, Four Color, Grid Rotation, Pattern Prediction, and Rule-Based Navigation. For each task we generate four variation runs, each containing 100 Polar instances (400 instances per task in total). We evaluate six representative models spanning closed-source and open-source families: Gemini-3.1-Pro, Gemini-3-Flash, GPT-5.2, Claude-Sonnet-4.6,

Table 13: Per-task results under the Visual Pattern Matching category, subcategory Shape and Jigsaw Assembly. All models are evaluated under high reasoning mode. C and P denote Cartesian and Polar accuracy (%), respectively. $(-\Delta)$ indicates the accuracy change from Cartesian to Polar. Task abbreviations: Frag. Match. = Fragment Matching; Jigsaw = Jigsaw Matching; Layer Comp. = Layer Completion; Pat. Comp. = Pattern Completion; Shape Comp. = Shape Completion; Shape Fit. = Shape Fitting; Tmpl. Match. = Template Matching.

Model	Frag. Match.			Jigsaw			Layer Comp.			Pat. Comp.			Shape Comp.			Shape Fit.			Tmpl. Match.		
	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$
Human	100.0	100.0	(0.0)	100.0	100.0	(0.0)	100.0	95.0	(−5.0)	100.0	100.0	(0.0)	100.0	100.0	(0.0)	100.0	100.0	(0.0)	100.0	95.0	(−5.0)
Random	20.0	20.0	(0.0)	20.0	20.0	(0.0)	25.0	25.0	(0.0)	25.0	25.0	(0.0)	25.0	25.0	(0.0)	33.0	33.0	(0.0)	20.0	20.0	(0.0)
Closed-Source MLLMs																					
Gemini-3.1-Pro	59.0	31.0	(−28.0)	80.0	44.0	(−36.0)	93.0	69.7	(−23.3)	55.0	26.0	(−29.0)	43.0	29.3	(−13.7)	60.0	33.0	(−27.0)	97.0	53.5	(−43.5)
Gemini-3-Flash	53.0	25.0	(−28.0)	74.0	50.0	(−24.0)	91.0	70.0	(−21.0)	41.0	20.0	(−21.0)	33.0	22.0	(−11.0)	61.0	45.0	(−16.0)	100.0	62.0	(−38.0)
Gemini-3-Flash-lite	24.0	30.0	(6.0)	40.0	34.0	(−6.0)	58.0	47.0	(−11.0)	29.0	19.0	(−10.0)	49.0	29.0	(−20.0)	46.0	34.0	(−12.0)	92.0	37.0	(−55.0)
Gemini-2.5-Pro	21.0	19.0	(−2.0)	34.0	34.0	(0.0)	76.0	52.0	(−24.0)	29.0	27.0	(−2.0)	28.0	26.0	(−2.0)	43.0	39.0	(−4.0)	85.0	32.0	(−53.0)
Gemini-2.5-Flash	23.0	21.0	(−2.0)	18.0	16.0	(−2.0)	71.0	36.0	(−35.0)	30.0	28.0	(−2.0)	37.0	24.0	(−13.0)	30.0	33.0	(3.0)	76.0	19.0	(−57.0)
GPT-5.2	95.0	30.0	(−65.0)	54.0	46.0	(−8.0)	88.0	64.0	(−24.0)	94.0	24.0	(−70.0)	52.0	40.0	(−12.0)	48.0	34.0	(−14.0)	100.0	59.0	(−41.0)
Claude-Sonnet-4.6	35.0	18.6	(−16.5)	39.2	30.9	(−8.2)	67.0	74.8	(7.8)	31.6	24.0	(−7.6)	41.0	29.3	(−11.7)	49.5	42.4	(−7.1)	31.0	35.7	(4.7)
Grok-4-0709	26.0	21.0	(−5.0)	19.0	17.0	(−2.0)	38.0	26.0	(−12.0)	27.0	18.0	(−9.0)	32.0	21.0	(−11.0)	38.0	34.0	(−4.0)	67.0	22.6	(−44.4)
Grok-4-Fast-Reasoning	15.0	17.0	(2.0)	22.0	17.0	(−5.0)	52.0	26.0	(−26.0)	25.0	23.0	(−2.0)	24.0	26.0	(2.0)	38.0	37.0	(−1.0)	51.0	22.0	(−29.0)
Open-Source MLLMs																					
Gemma-4-31B	34.0	19.0	(−15.0)	50.0	47.0	(−3.0)	80.0	60.0	(−20.0)	50.0	34.0	(−16.0)	29.0	26.0	(−3.0)	60.0	39.0	(−21.0)	95.0	51.0	(−44.0)
Gemma-4-26B	31.0	19.0	(−12.0)	34.3	26.0	(−8.3)	64.0	54.0	(−10.0)	29.0	27.0	(−2.0)	22.2	22.2	(0.0)	52.5	37.0	(−15.5)	91.9	36.0	(−55.9)
Kimi-k2.5	70.4	31.0	(−39.4)	46.0	28.3	(−17.7)	91.0	66.0	(−25.0)	81.6	25.0	(−56.6)	37.0	29.0	(−8.0)	63.0	40.0	(−23.0)	95.0	50.0	(−45.0)
Qwen3.5-397B-A17B	84.0	33.0	(−51.0)	69.0	55.0	(−14.0)	96.0	49.0	(−47.0)	89.0	30.0	(−59.0)	43.0	19.0	(−24.0)	54.0	23.0	(−31.0)	88.0	56.0	(−32.0)
Mistral-Small-2503	31.3	11.3	(−20.0)	14.0	17.0	(3.0)	7.0	17.0	(10.0)	26.0	19.0	(−7.0)	30.0	30.0	(0.0)	38.0	33.0	(−5.0)	16.0	19.0	(3.0)

Table 14: Per-task results under the Spatial Transformation and Geometry category, subcategory Geometric Measurement and Counting. All models are evaluated under high reasoning mode. C and P denote Cartesian and Polar accuracy (%), respectively. $(-\Delta)$ indicates the accuracy change from Cartesian to Polar. Task abbreviations: Area Bal. = Area Balancing; Area Cnt. = Area Counting; Curve Len. = Curve Length; Pipe Len. = Pipe Lengths.

Model	Area Bal.			Area Cnt.			Curve Len.			Pipe Len.			Uncut Cells		
	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$
Human	95.0	75.0	(−20.0)	100.0	100.0	(0.0)	100.0	70.0	(−30.0)	95.0	100.0	(5.0)	100.0	100.0	(0.0)
Random	20.0	19.9	(−0.1)	0.1	0.1	(−0.0)	20.0	20.0	(0.0)	0.0	0.0	(0.0)	20.0	20.0	(0.0)
Closed-Source MLLMs															
Gemini-3.1-Pro	66.0	22.0	(−44.0)	94.0	2.0	(−92.0)	69.0	20.0	(−49.0)	75.0	1.0	(−74.0)	89.0	44.0	(−45.0)
Gemini-3-Flash	51.0	22.0	(−29.0)	80.0	4.0	(−76.0)	59.0	39.0	(−20.0)	12.0	2.0	(−10.0)	63.0	41.0	(−22.0)
Gemini-3-Flash-lite	29.0	22.0	(−7.0)	8.0	6.0	(−2.0)	26.0	26.0	(0.0)	4.0	0.0	(−4.0)	40.0	35.0	(−5.0)
Gemini-2.5-Pro	20.0	18.0	(−2.0)	6.0	4.0	(−2.0)	38.0	40.0	(2.0)	1.0	1.0	(0.0)	45.0	34.0	(−11.0)
Gemini-2.5-Flash	23.0	16.0	(−7.0)	1.0	2.0	(1.0)	23.0	15.0	(−8.0)	0.0	2.0	(2.0)	34.0	28.0	(−6.0)
GPT-5.2	46.0	21.0	(−25.0)	88.0	7.0	(−81.0)	35.0	29.0	(−6.0)	56.0	7.0	(−49.0)	50.0	41.0	(−9.0)
Claude-Sonnet-4.6	18.2	10.0	(−8.2)	37.4	1.0	(−36.4)	75.8	87.6	(11.8)	2.0	0.0	(−2.0)	43.3	47.4	(4.1)
Grok-4-0709	20.0	20.0	(0.0)	0.0	2.0	(2.0)	60.0	67.0	(7.0)	0.0	0.0	(0.0)	34.7	36.4	(1.7)
Grok-4-Fast-Reasoning	17.0	13.0	(−4.0)	1.0	0.0	(−1.0)	94.0	90.0	(−4.0)	0.0	0.0	(0.0)	38.0	43.0	(5.0)
Open-Source MLLMs															
Gemma-4-31B	48.0	24.0	(−24.0)	53.0	6.0	(−47.0)	46.0	25.0	(−21.0)	1.0	0.0	(−1.0)	62.0	41.0	(−21.0)
Gemma-4-26B	20.0	22.0	(2.0)	40.0	5.0	(−35.0)	9.1	5.0	(−4.0)	11.0	0.0	(−11.0)	52.0	20.2	(−31.8)
Kimi-k2.5	46.0	22.4	(−23.6)	98.0	3.0	(−95.0)	52.0	70.7	(18.7)	12.0	1.0	(−11.0)	58.6	36.0	(−22.6)
Qwen3.5-397B-A17B	52.0	20.0	(−32.0)	72.0	7.0	(−65.0)	47.0	32.0	(−15.0)	27.0	3.0	(−24.0)	75.0	32.0	(−43.0)
Mistral-Small-2503	20.0	27.0	(7.0)	0.0	0.0	(0.0)	54.7	74.0	(19.3)	0.0	0.0	(0.0)	19.0	17.0	(−2.0)

Kimi-k2.5, and Gemma-4-31B. Each model is evaluated under three conditions: (1) vanilla Polar (the standard Polar prompt), (2) Polar + Conversion Hint, where the prompt is augmented with an explicit instruction asking the model to first re-map the Polar layout into an equivalent Cartesian grid before solving the task (Figure 10), and (3) Polar + 5-shot ICL, where the prompt is prepended with five solved Polar examples from the same task (Figure 11). The Cartesian baseline uses the same 100 instances per run under the standard prompt. Table 20 reports each model's accuracy averaged across the six tasks and four runs. Figure 7(f) visualizes the same data. The six selected tasks span all five task categories: Visual Pattern Matching (Pattern Prediction), Spatial Transformation & Geometry (Four Color, Grid Rotation), Navigation & Routing (Rule-Based Navigation), Combinatorics & Probability (Edge Counting), and Algorithmic Logic & Simulation (Bouncing Point).

Geometry deformation. To test whether the performance gap is specific to the Polar transformation, we construct a gradient of intermediate geometric deformations applied to Cartesian images. We randomly select three tasks—Pattern Prediction, Pipe Lengths, and Path Counting—and apply five deformation levels of increasing severity: (1) **Wave**: a sinusoidal distortion applied to the vertical axis

Table 15: Per-task results under the Spatial Transformation and Geometry category, subcategory Spatial Constraints and Topology. All models are evaluated under high reasoning mode. C and P denote Cartesian and Polar accuracy (%), respectively. $(-\Delta)$ indicates the accuracy change from Cartesian to Polar. Task abbreviations: Grid Fold. = Grid Folding; Imp. Shape = Impossible Shape.

Model	Four Color			Grid Fold.			Imp. Shape		
	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$
Human	100.0	95.0	(−5.0)	100.0	100.0	(0.0)	100.0	100.0	(0.0)
Random	20.0	20.0	(0.0)	20.0	20.0	(0.0)	20.0	20.0	(0.0)
Closed-Source MLLMs									
Gemini-3.1-Pro	91.9	28.3	(−63.6)	100.0	54.1	(−45.9)	62.0	20.0	(−42.0)
Gemini-3-Flash	86.0	19.0	(−67.0)	99.0	47.0	(−52.0)	41.0	20.0	(−21.0)
Gemini-3-Flash-lite	44.0	18.0	(−26.0)	86.0	41.0	(−45.0)	28.0	22.0	(−6.0)
Gemini-2.5-Pro	30.0	20.0	(−10.0)	60.0	42.0	(−18.0)	14.0	10.0	(−4.0)
Gemini-2.5-Flash	24.0	25.0	(1.0)	40.0	26.0	(−14.0)	23.0	22.0	(−1.0)
GPT-5.2	98.0	27.0	(−71.0)	100.0	59.0	(−41.0)	43.0	22.0	(−21.0)
Claude-Sonnet-4.6	45.4	19.2	(−26.2)	100.0	37.1	(−62.9)	19.2	17.2	(−2.0)
Grok-4-0709	34.0	25.0	(−9.0)	45.0	23.0	(−22.0)	24.0	24.0	(0.0)
Grok-4-Fast-Reasoning	28.0	28.0	(0.0)	51.0	18.0	(−33.0)	21.0	15.0	(−6.0)
Open-Source MLLMs									
Gemma-4-31B	65.0	32.0	(−33.0)	97.0	44.0	(−53.0)	33.0	22.0	(−11.0)
Gemma-4-26B	51.5	19.0	(−32.5)	98.0	36.0	(−62.0)	25.5	13.0	(−12.5)
Kimi-k2.5	81.8	26.0	(−55.8)	99.0	39.0	(−60.0)	56.0	13.0	(−43.0)
Qwen3.5-397B-A17B	92.0	43.0	(−49.0)	100.0	64.0	(−36.0)	82.0	26.0	(−56.0)
Mistral-Small-2503	18.2	17.0	(−1.2)	20.0	21.2	(1.2)	18.6	28.3	(9.7)

Table 16: Per-task results under the Spatial Transformation and Geometry category, subcategory Rotations and Reflections. All models are evaluated under high reasoning mode. C and P denote Cartesian and Polar accuracy (%), respectively. $(-\Delta)$ indicates the accuracy change from Cartesian to Polar. Task abbreviations: Grid Rot. = Grid Rotation; Mirror Ref. = Mirror Reflection; Pivot Rot. = Pivot Rotation; Rot. Center = Rotation Center; Rot. Match. = Rotation Matching.

Model	Grid Rot.			Mirror Ref.			Pivot Rot.			Rot. Center			Rot. Match.		
	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$
Human	100.0	100.0	(0.0)	100.0	100.0	(0.0)	100.0	100.0	(0.0)	95.0	90.0	(−5.0)	100.0	100.0	(0.0)
Random	20.0	20.0	(0.0)	25.0	25.0	(0.0)	20.0	20.0	(0.0)	25.0	25.0	(0.0)	0.0	0.0	(0.0)
Closed-Source MLLMs															
Gemini-3.1-Pro	98.0	31.0	(−67.0)	84.0	42.0	(−42.0)	87.0	51.5	(−35.5)	100.0	40.8	(−59.2)	72.0	51.5	(−20.5)
Gemini-3-Flash	99.0	35.0	(−64.0)	76.0	47.0	(−29.0)	83.0	62.0	(−21.0)	96.0	20.0	(−76.0)	48.0	30.0	(−18.0)
Gemini-3-Flash-lite	73.0	23.0	(−50.0)	33.0	36.0	(3.0)	52.0	36.0	(−16.0)	78.0	11.0	(−67.0)	27.0	6.0	(−21.0)
Gemini-2.5-Pro	30.0	20.0	(−10.0)	38.0	35.0	(−3.0)	48.0	33.0	(−15.0)	75.0	34.0	(−41.0)	34.0	20.0	(−14.0)
Gemini-2.5-Flash	27.0	23.0	(−4.0)	32.0	27.0	(−5.0)	32.0	22.0	(−10.0)	66.0	27.0	(−39.0)	18.0	5.0	(−13.0)
GPT-5.2	100.0	87.0	(−13.0)	98.0	58.0	(−40.0)	94.0	68.0	(−26.0)	96.0	28.0	(−68.0)	91.0	87.0	(−4.0)
Claude-Sonnet-4.6	53.1	19.2	(−33.9)	39.0	43.0	(4.0)	40.0	32.0	(−8.0)	68.7	13.3	(−55.4)	7.2	1.0	(−6.2)
Grok-4-0709	34.0	21.0	(−13.0)	35.0	32.0	(−3.0)	33.0	21.2	(−11.8)	59.6	15.0	(−44.6)	9.0	2.0	(−7.0)
Grok-4-Fast-Reasoning	21.0	18.0	(−3.0)	25.0	25.0	(0.0)	28.0	13.0	(−15.0)	41.0	15.0	(−26.0)	4.0	2.0	(−2.0)
Open-Source MLLMs															
Gemma-4-31B	97.0	28.0	(−69.0)	64.0	35.0	(−29.0)	76.0	44.0	(−32.0)	82.0	26.0	(−56.0)	57.0	43.0	(−14.0)
Gemma-4-26B	92.0	25.2	(−66.8)	62.0	40.0	(−22.0)	57.0	42.0	(−15.0)	70.0	6.1	(−63.9)	43.0	18.0	(−25.0)
Kimi-k2.5	100.0	29.0	(−71.0)	91.9	53.5	(−38.4)	78.0	52.5	(−25.5)	90.7	16.7	(−74.0)	68.0	42.0	(−26.0)
Qwen3.5-397B-A17B	100.0	73.0	(−27.0)	94.0	56.0	(−38.0)	89.0	60.0	(−29.0)	93.0	35.0	(−58.0)	81.0	62.0	(−19.0)
Mistral-Small-2503	23.0	15.3	(−7.7)	27.0	34.0	(7.0)	13.0	20.0	(7.0)	37.0	22.0	(−15.0)	3.0	0.0	(−3.0)

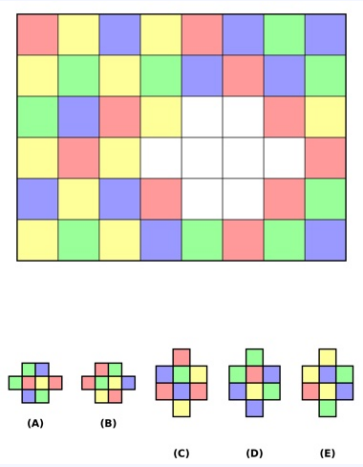
of the Cartesian grid, which introduces curvature without radial compression or angular wrapping; and (2) **Arc-$\boldsymbol{\theta}$** ($\theta \in \{45, 90, 135, 180\}$): the Cartesian grid is bent into a circular arc of the specified angular extent, progressively departing from orthogonal structure. All deformations are generated programmatically from the same Cartesian source images while ensuring that task-relevant visual elements (labels, grid lines, paths) remain legible. The same six models are evaluated on each deformation level, as well as on the original Cartesian and full Polar versions for reference. Table 21 reports accuracy averaged over the three tasks. Figure 7(e) visualizes the per-deformation-level accuracy for each model.

E Task Examples

We provide examples with complete image and question, answer from more tasks in Polaris-Bench.

Four Color

Cartesian

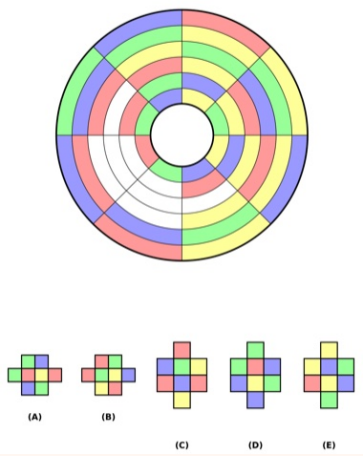


In the diagram below, two neighbouring cells are never allowed to have the same color. Which puzzle piece has to be placed in the gap so that this rule is followed? Note: You can rotate the pieces.

(A) (B) (C) (D) (E)

Answer: E

Polar



In the diagram below, two neighbouring cells are never allowed to have the same color. Which puzzle piece has to be placed in the gap so that this rule is followed? Note: You can rotate the pieces.

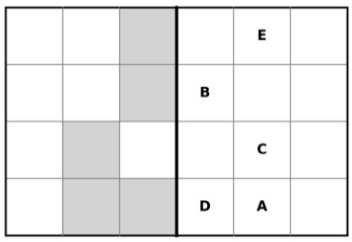
(A) (B) (C) (D) (E)

Answer: E

Figure 17: Example from four color task.

Grid Folding

Cartesian

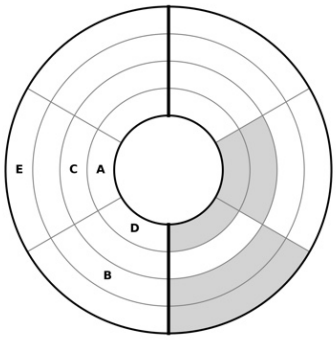


A page is folded along the thick line as shown. Which letter will NOT be covered by a grey square?

(A) (B) (C) (D) (E)

Answer: E

Polar



A page is folded along the thick line as shown. Which letter will NOT be covered by a grey cell?

(A) (B) (C) (D) (E)

Answer: E

Figure 18: Example from grid folding task.

Table 17: Per-task results under the Algorithmic Logic and Simulation category. All models are evaluated under high reasoning mode. C and P denote Cartesian and Polar accuracy (%), respectively. $(-\Delta)$ indicates the accuracy change from Cartesian to Polar. Task abbreviations: Bounce = Bouncing Point; Collision = Collision Detection; Max. Coll. = Maximum Collection; Min. Flips = Minimum Flips.

Model	Bounce			Collision			Max. Coll.			Min. Flips			N-Queens			Sudoku		
	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$
Human	85.0	70.0	(−15.0)	95.0	95.0	(0.0)	95.0	95.0	(0.0)	85.0	80.0	(−5.0)	95.0	95.0	(0.0)	100.0	90.0	(−10.0)
Random	1.8	1.8	(−0.0)	20.0	20.0	(0.0)	20.0	20.0	(0.0)	20.0	20.0	(0.0)	20.0	20.0	(0.0)	20.0	20.0	(0.0)
Closed-Source MLLMs																		
Gemini-3.1-Pro	97.0	30.6	(−66.4)	58.0	34.0	(−24.0)	66.0	31.0	(−35.0)	88.9	85.7	(−3.2)	100.0	74.0	(−26.0)	100.0	60.0	(−40.0)
Gemini-3-Flash	49.0	28.0	(−21.0)	67.0	31.0	(−36.0)	30.0	20.0	(−10.0)	68.0	53.0	(−15.0)	100.0	50.0	(−50.0)	99.0	67.0	(−32.0)
Gemini-3-Flash-lite	34.0	16.0	(−18.0)	54.0	27.0	(−27.0)	27.0	18.0	(−9.0)	37.0	32.0	(−5.0)	100.0	33.0	(−67.0)	96.0	44.0	(−52.0)
Gemini-2.5-Pro	32.0	25.0	(−7.0)	48.0	29.0	(−19.0)	18.0	17.0	(−1.0)	36.0	24.0	(−12.0)	99.0	34.0	(−65.0)	93.0	52.0	(−41.0)
Gemini-2.5-Flash	19.0	24.0	(5.0)	46.0	25.0	(−21.0)	20.0	14.0	(−6.0)	31.0	31.0	(0.0)	99.0	35.0	(−64.0)	93.0	37.0	(−56.0)
GPT-5.2	79.0	32.0	(−47.0)	61.0	40.0	(−21.0)	43.0	17.0	(−26.0)	41.0	30.0	(−11.0)	100.0	90.0	(−10.0)	100.0	71.0	(−29.0)
Claude-Sonnet-4.6	86.3	9.6	(−76.8)	51.0	32.3	(−18.7)	20.6	33.3	(12.7)	18.9	22.7	(3.7)	99.0	34.7	(−64.2)	83.8	49.0	(−34.8)
Grok-4-0709	21.9	17.2	(−4.7)	38.4	29.0	(−9.4)	26.0	29.3	(3.3)	55.7	56.1	(0.5)	96.0	29.0	(−67.0)	81.0	36.0	(−45.0)
Grok-4-Fast-Reasoning	17.0	13.0	(−4.0)	30.0	25.0	(−5.0)	34.0	30.0	(−4.0)	57.0	55.0	(−2.0)	76.0	26.0	(−50.0)	84.0	36.0	(−48.0)
Open-Source MLLMs																		
Gemma-4-31B	48.0	21.0	(−27.0)	53.0	26.0	(−27.0)	52.0	27.0	(−25.0)	55.0	36.0	(−19.0)	100.0	56.0	(−44.0)	98.0	50.0	(−48.0)
Gemma-4-26B	42.0	23.0	(−19.0)	39.0	14.0	(−25.0)	14.1	8.0	(−6.1)	43.0	36.0	(−7.0)	100.0	46.0	(−54.0)	98.0	50.0	(−48.0)
Kimi-k2.5	90.6	17.8	(−72.8)	56.0	32.0	(−24.0)	36.8	22.3	(−14.5)	59.4	45.2	(−14.2)	99.0	53.7	(−45.3)	99.0	53.0	(−46.0)
Qwen3.5-397B-A17B	58.0	25.0	(−33.0)	58.0	37.0	(−21.0)	70.0	29.0	(−41.0)	56.0	51.0	(−5.0)	100.0	46.0	(−54.0)	91.0	71.0	(−20.0)
Mistral-Small-2503	4.5	0.0	(−4.5)	31.0	35.7	(4.7)	22.0	18.0	(−4.0)	15.8	24.2	(8.4)	38.0	20.0	(−18.0)	39.0	26.0	(−13.0)

Table 18: Per-task results under the Combinatorics and Probability category, subcategory Bounded Combinatorics. All models are evaluated under high reasoning mode. C and P denote Cartesian and Polar accuracy (%), respectively. $(-\Delta)$ indicates the accuracy change from Cartesian to Polar. Task abbreviations: Bnd. Diag. = Bounded Diagonal Paths; Bnd. Knight = Bounded Knight Paths; Chkpt. Path = Checkpoint Paths; Lattice = Lattice Paths; Path Cnt. = Path Counting.

Model	Bnd. Diag.			Bnd. Knight			Chkpt. Path			Lattice			Path Cnt.		
	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$
Human	65.0	50.0	(−15.0)	75.0	45.0	(−30.0)	75.0	45.0	(−30.0)	95.0	90.0	(−5.0)	85.0	70.0	(−15.0)
Random	0.0	0.0	(0.0)	0.0	0.0	(0.0)	0.0	0.0	(0.0)	0.0	0.0	(0.0)	20.0	20.0	(0.0)
Closed-Source MLLMs															
Gemini-3.1-Pro	100.0	64.3	(−35.7)	92.0	4.0	(−88.0)	99.0	10.0	(−89.0)	88.0	19.0	(−69.0)	82.0	23.0	(−59.0)
Gemini-3-Flash	97.0	39.0	(−58.0)	76.0	6.0	(−70.0)	81.0	5.0	(−76.0)	76.0	13.0	(−63.0)	59.0	34.0	(−25.0)
Gemini-3-Flash-lite	53.0	42.0	(−11.0)	23.0	1.0	(−22.0)	55.0	1.0	(−54.0)	40.0	4.0	(−36.0)	38.0	27.0	(−11.0)
Gemini-2.5-Pro	29.0	23.0	(−6.0)	4.0	0.0	(−4.0)	25.0	1.0	(−24.0)	9.0	2.0	(−7.0)	32.0	28.0	(−4.0)
Gemini-2.5-Flash	12.0	11.0	(−1.0)	1.0	1.0	(0.0)	9.0	0.0	(−9.0)	6.0	1.0	(−5.0)	20.0	18.0	(−2.0)
GPT-5.2	98.0	78.0	(−20.0)	88.0	33.0	(−55.0)	71.0	0.0	(−71.0)	83.0	9.0	(−74.0)	85.0	22.0	(−63.0)
Claude-Sonnet-4.6	74.7	38.0	(−36.7)	66.7	9.5	(−57.2)	62.5	4.3	(−58.2)	29.4	3.1	(−26.2)	27.6	32.0	(4.5)
Grok-4-0709	17.2	13.0	(−4.1)	4.5	1.0	(−3.5)	7.3	0.0	(−7.3)	10.0	2.0	(−8.0)	28.0	25.0	(−3.0)
Grok-4-Fast-Reasoning	27.0	13.0	(−14.0)	9.0	0.0	(−9.0)	5.0	0.0	(−5.0)	2.0	2.0	(0.0)	31.0	32.0	(1.0)
Open-Source MLLMs															
Gemma-4-31B	91.0	52.0	(−39.0)	56.0	8.0	(−48.0)	48.0	7.0	(−41.0)	75.0	7.0	(−68.0)	41.0	22.0	(−19.0)
Gemma-4-26B	80.0	50.0	(−30.0)	51.0	4.1	(−46.9)	22.0	5.0	(−17.0)	48.0	9.0	(−39.0)	10.2	11.2	(1.0)
Kimi-k2.5	96.9	16.7	(−80.2)	79.4	4.1	(−75.3)	51.6	1.1	(−50.6)	52.0	5.0	(−47.0)	49.0	27.6	(−21.4)
Qwen3.5-397B-A17B	99.0	35.0	(−64.0)	41.0	9.0	(−32.0)	81.0	4.0	(−77.0)	77.0	6.0	(−71.0)	8.0	4.0	(−4.0)
Mistral-Small-2503	0.0	0.0	(0.0)	2.1	1.0	(−1.1)	3.5	0.0	(−3.5)	0.0	1.0	(1.0)	23.0	27.0	(4.0)

Table 19: Per-task results under the Combinatorics and Probability category, subcategories Stochastic Processes and Topological Combinatorics. All models are evaluated under high reasoning mode. C and P denote Cartesian and Polar accuracy (%), respectively. $(-\Delta)$ indicates the accuracy change from Cartesian to Polar. Task abbreviations: Edge Cnt. = Edge Counting; Rand. Walk = Random Walk; Knight Path = Knight Paths; Wrap. Diag. = Wrapping Diagonal Paths.

Model	Edge Cnt.			Rand. Walk			Knight Path			Wrap. Diag.		
	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$	C	P	$(-\Delta)$
Human	100.0	95.0	(−5.0)	70.0	80.0	(10.0)	70.0	45.0	(−25.0)	65.0	60.0	(−5.0)
Random	20.0	20.0	(0.0)	20.0	20.0	(0.0)	0.0	0.0	(0.0)	0.0	0.0	(0.0)
Closed-Source MLLMs												
Gemini-3.1-Pro	91.0	25.0	(−66.0)	100.0	40.8	(−59.2)	90.0	2.0	(−88.0)	90.0	35.7	(−54.3)
Gemini-3-Flash	65.0	34.0	(−31.0)	93.0	26.0	(−67.0)	74.0	5.0	(−69.0)	52.0	23.0	(−29.0)
Gemini-3-Flash-lite	28.0	29.0	(1.0)	78.0	23.0	(−55.0)	28.0	1.0	(−27.0)	21.0	15.0	(−6.0)
Gemini-2.5-Pro	29.0	27.0	(−2.0)	66.0	24.0	(−42.0)	3.0	1.0	(−2.0)	13.0	8.0	(−5.0)
Gemini-2.5-Flash	21.0	26.0	(5.0)	68.0	25.0	(−43.0)	2.0	2.0	(0.0)	4.0	1.0	(−3.0)
GPT-5.2	67.0	36.0	(−31.0)	99.0	44.0	(−55.0)	88.0	28.0	(−60.0)	84.0	26.0	(−58.0)
Claude-Sonnet-4.6	40.4	39.4	(−1.0)	42.5	23.2	(−19.4)	18.8	0.0	(−18.8)	22.1	2.1	(−20.0)
Grok-4-0709	39.0	40.0	(1.0)	62.2	17.2	(−45.0)	5.2	1.0	(−4.2)	14.0	9.0	(−5.0)
Grok-4-Fast-Reasoning	41.0	36.0	(−5.0)	26.0	20.0	(−6.0)	7.0	0.0	(−7.0)	11.0	8.0	(−3.0)
Open-Source MLLMs												
Gemma-4-31B	41.4	18.0	(−23.4)	89.0	45.0	(−44.0)	66.0	7.0	(−59.0)	53.0	9.0	(−44.0)
Gemma-4-26B	22.2	14.0	(−8.2)	83.0	23.0	(−60.0)	62.0	4.0	(−58.0)	33.3	13.0	(−20.3)
Kimi-k2.5	41.0	34.0	(−7.0)	92.6	43.3	(−49.3)	85.6	7.1	(−78.5)	70.0	7.0	(−63.0)
Qwen3.5-397B-A17B	68.0	31.0	(−37.0)	96.0	49.0	(−47.0)	53.0	6.0	(−47.0)	64.0	19.0	(−45.0)
Mistral-Small-2503	38.5	41.2	(2.7)	18.0	26.0	(8.0)	0.0	2.0	(2.0)	1.0	0.0	(−1.0)

Table 20: Prompting intervention accuracy (%) across 6 representative models on 6 randomly selected tasks (4 runs $\times$ 100 instances per task). Neither conversion hints nor 5-shot ICL recovers Polar accuracy.

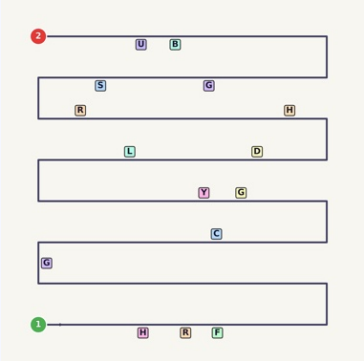
Model	Cartesian	Polar	Polar+Conversion Hint	Polar+5-shot ICL
Gemini-3.1-Pro	94.2	28.0	27.8	24.7
Gemini-3-Flash	78.3	30.5	28.2	28.1
GPT-5.2	82.3	38.3	40.0	26.0
Claude-Sonnet-4.6	45.2	22.0	22.3	24.6
Kimi-k2.5	69.2	25.1	21.2	19.1
Gemma-4-31B	61.2	27.8	24.3	22.6

Table 21: Complete results (accuracy %) under geometry deformation across 6 evaluated models (averaged over 3 tasks: Pattern Prediction, Pipe Lengths, and Path Counting).

Model	Cartesian	Wave	Arc-45°	Arc-90°	Arc-135°	Arc-180°	Polar
Gemini-3.1-Pro	89.0	44.0	35.0	20.0	15.3	16.3	17.1
Gemini-3-Flash	51.3	23.9	30.7	21.1	15.8	16.5	23.3
GPT-5.2	65.0	29.0	30.3	24.8	21.0	20.3	16.7
Claude-Sonnet-4.6	18.7	12.3	16.3	13.7	14.7	12.0	15.7
Gemma-4-31B	27.7	17.7	22.3	15.9	19.2	16.3	18.0
Kimi-k2.5	26.7	16.0	19.0	18.3	14.7	14.0	15.3

Letter Collection

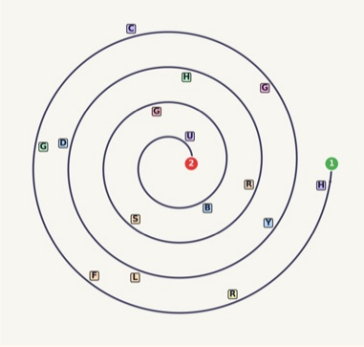
Cartesian



Imagine walking along the continuous path from the start (point 1) to the end (point 2). As you walk forward along the path, collect all the letters that appear on your right-hand side relative to your walking direction. What is the final sequence of letters you collect? Provide only the continuous letter string as your answer.

Answer: HRFGGYHRSG

Polar



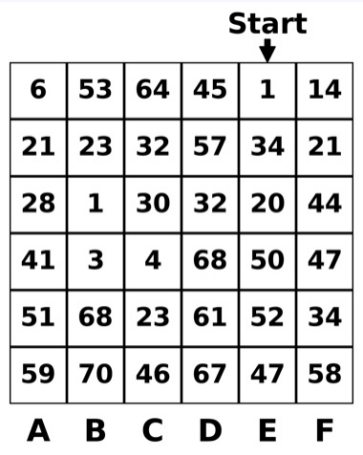
Imagine walking along the continuous path from the start (point 1) to the end (point 2). As you walk forward along the path, collect all the letters that appear on your right-hand side relative to your walking direction. What is the final sequence of letters you collect? Provide only the continuous letter string as your answer.

Answer: HRFGGYHRSG

Figure 19: Example from letter collection task.

Monotonic Path

Cartesian

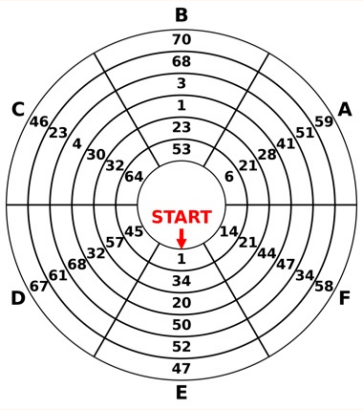


The rooms in the house/maze are numbered. Eva enters through the START entrance. She has to walk through the rooms in such a way that each room she enters has a number strictly higher than the previous one. Through which door does Eva leave?

(A) A (B) B (C) C (D) D (E) E (F) F

Answer: D

Polar



The rooms in the house/maze are numbered. Eva enters through the START entrance. She has to walk through the rooms in such a way that each room she enters has a number strictly higher than the previous one. Through which door does Eva leave?

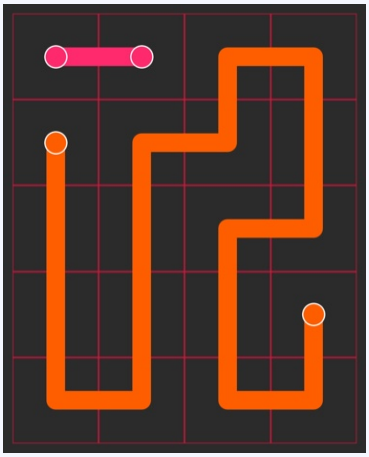
(A) A (B) B (C) C (D) D (E) E (F) F

Answer: D

Figure 20: Example from monotonic path task.

Pipe Lengths

Cartesian



There are multiple pipes of different colors in the image, and the pipes fill the entire grid. Counting the start and end cells, how many cells do the pipes pass through, in descending order? (Answer format is like: 5,3,3,1)

Answer: 18,2

Polar



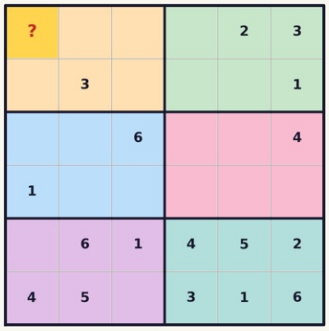
There are multiple pipes of different colors in the image, and the pipes fill the entire grid. Counting the start and end cells, how many cells do the pipes pass through, in descending order? (Answer format is like: 5,3,3,1)

Answer: 18,2

Figure 21: Example from pipe lengths task.

Sudoku

Cartesian

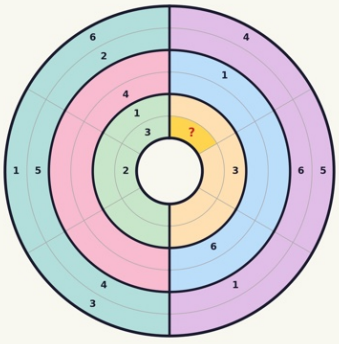


In this 6x6 Sudoku puzzle, each row, each column, and each outlined box must contain all numbers from 1 to 6 exactly once. Rows are numbered 1 to 6 from top to bottom, and columns are numbered 1 to 6 from left to right.
What number should replace the '?' in the highlighted cell (row 1, column 1)?

(A) 1 (B) 2 (C) 3 (D) 5 (E) 6

Answer: E

Polar



In this polar Sudoku puzzle (6 rings Ã— 6 sectors), each ring, each sector, and each outlined region must contain all numbers from 1 to 6 exactly once. The rings are numbered 1 to 6 from the innermost to the outermost ring. The sectors are numbered 1 to 6 clockwise starting from the top (12 o'clock).
What number should replace the '?' in the highlighted cell (ring 1, sector 1)?

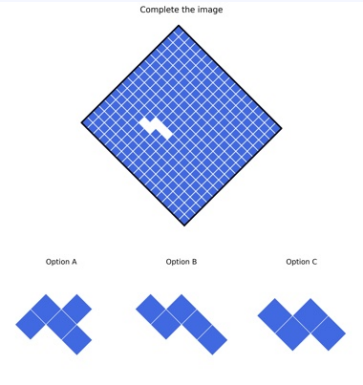
(A) 1 (B) 2 (C) 3 (D) 5 (E) 6

Answer: E

Figure 22: Example from sudoku task.

Shape Fitting

Cartesian

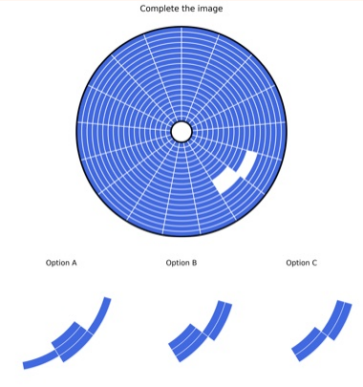


Choose the appropriate option to fill in the blank, so that the image is complete.

A B C

Answer: B

Polar



Choose the appropriate option to fill in the blank, so that the image is complete.

A B C

Answer: B

Figure 23: Example from shape fitting task.

Word Search

Cartesian



In the grid of characters below, by how many different paths can one spell 'WIND'? Beginning at the 'W' in the middle, a path allows only moves from one cell to an adjacent (above, below, left, or right) cell.

Answer: 7

Octagonal



In the octagonal grid below, by how many different paths can one spell 'WIND'? Beginning at the 'W' in the middle, a path allows only moves from one cell to an adjacent cell through the light open boundaries (up, down, left, right), without crossing the thick dark diagonal walls.

Answer: 7

Hexagonal



In the honeycomb grid below, by how many different paths can one spell 'WIND'? Beginning at the 'W' in the middle, a path allows only moves from one cell to an adjacent cell through the light open boundaries, without crossing the thick dark walls.

Answer: 7

Polar (Graph)



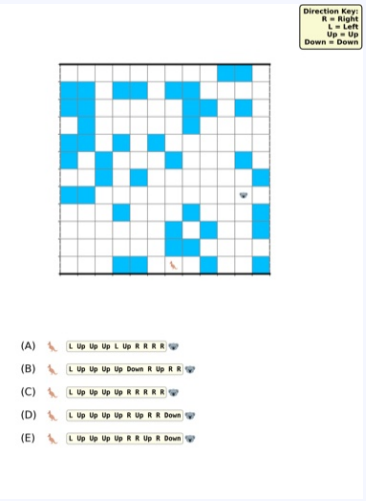
In the node-link graph below, by how many different paths can one spell 'WIND'? Beginning at the 'W' in the center node, a path allows only moves from one node to an adjacent node connected directly by a line.

Answer: 7

Figure 24: Example from word search task.

Wrapping Path Finding

Cartesian

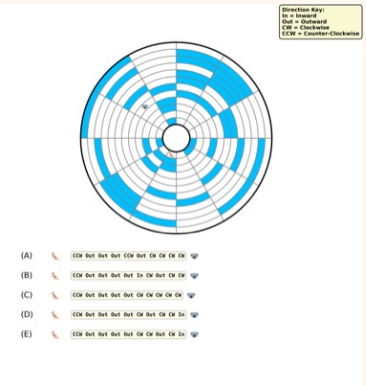


The kangaroo wants to reach the koala. It can move one cell at a time: Up, Down, Left (L), or Right (R). It cannot enter blue (water) cells. The top and bottom edges (thick black borders) are impassable, but the left and right edges wrap around (shown by dashed lines) -- moving left from the first column enters the last column, and vice versa. Which sequence of moves takes the kangaroo to the koala?

(A) (B) (C) (D) (E)

Answer: C

Polar



The kangaroo wants to reach the koala. It can move one cell at a time: Inward (In), Outward (Out), Clockwise (CW), or Counter-Clockwise (CCW). It cannot enter blue (water) cells. The thick black inner and outer circles are impassable boundaries, but the grid wraps around in the angular direction -- the kangaroo can move continuously clockwise or counter-clockwise. Which sequence of moves takes the kangaroo to the koala?

(A) (B) (C) (D) (E)

Answer: C

Figure 25: Example from wrapping path finding task.

Wrapping Navigation

Cartesian

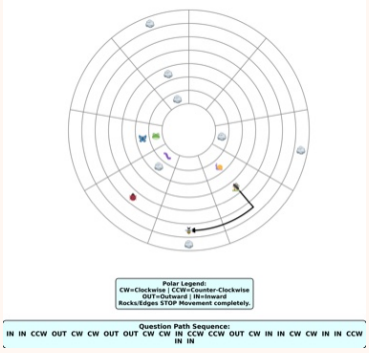


The Ant starts at the House. The black arrow line shows its path to the Bee. Rocks block movement. If a move command directs the Ant into a rock or a solid wall, it bumps into it, STOPS COMPLETELY, and ignores ALL remaining commands in the sequence. The board wraps horizontally (moving off the right edge brings the Ant to the left edge). Top and bottom edges are solid walls. If it instead attempts to follow the absolute movement sequence shown in the light yellow box, which animal does it permanently stop at?

(A) Bee
(B) Frog
(C) Butterfly
(D) Caterpillar
(E) Ladybug
(F) Snail

Answer: B

Polar



The Ant starts at the House. The black arrow line shows its path to the Bee. Rocks block movement. If a move command directs the Ant into a rock or a solid wall, it bumps into it, STOPS COMPLETELY, and ignores ALL remaining commands in the sequence. If it instead attempts to follow the movement sequence shown in the light cyan box at the bottom, which animal does it permanently stop at?

(A) Bee
(B) Frog
(C) Butterfly
(D) Caterpillar
(E) Ladybug
(F) Snail

Answer: B

Figure 26: Example from wrapping navigation task.