Understanding the Effects of Distractors on Reasoning Vision-Language Models

Anonymous ACL submission

Abstract

How does irrelevant information (i.e., distractors) affect test-time scaling in vision-language models (VLMs)? Prior work on text-only language models has shown that textual distractors can intensify inverse scaling, causing models to reason longer but less effective reasoning traces. In this work, we investigate whether similar phenomena arise in multimodal settings. We introduce Idis (Images with distractors), a visual question-answering dataset that systematically varies distractors along semantic and numerical dimensions. Our analyses reveal that visual distractors affect reasoning VLMs in a fundamentally different way from textual distractors: although inverse scaling still emerges, visual distractors reduce accuracy without increasing reasoning length. We further show that attribute counts extracted from reasoning traces provide key insights into how distractors interact with reasoning length and accuracy. As a sanity check, we propose a simple prompting strategy that mitigates distractor-driven predictions in reasoning vision-language models.

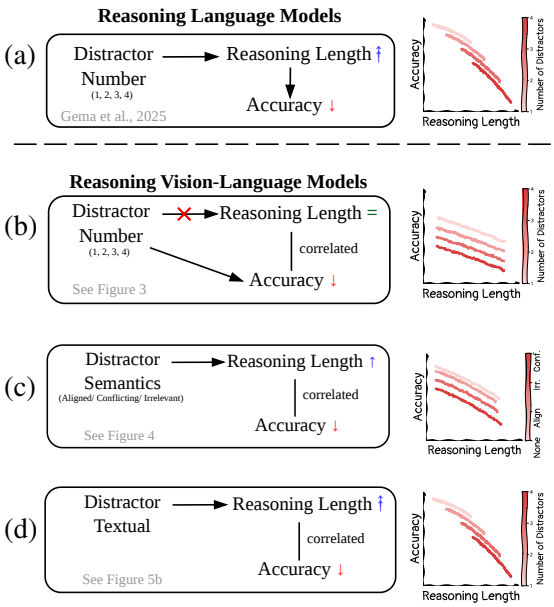


Figure 1: **Test-time scaling of reasoning VLMs.** Adding visual distractors decreases accuracy without increasing reasoning length, shifting the entire length–accuracy curve downward, unlike reasoning LMs. In contrast, textual distractors inserted into the prompt intensify the test-time inverse scaling pattern.

1 Introduction

Increasing test-time computation—e.g., generating more tokens during inference—has proven to be an effective strategy for improving the prediction quality of language models (LMs), and similar benefits have been observed for vision-language models (VLMs). In particular, reasoning VLMs equipped with long chain-of-thought traces have shown impressive performance across tasks requiring multimodal understanding, ranging from visual question-answering (VQA) to mathematical reasoning and embodied tasks (Qwen Team, 2025; Intern-S1 Team, 2025; GLM-V Team, 2025).

However, is longer reasoning always beneficial? Unfortunately, the answer is no. Reasoning models are prone to several failure modes: they may ***overthink***—producing lengthy reasoning traces without improving upon shorter, non-reasoning outputs (Chen et al., 2025; Sui et al., 2025)—or even exhibit ***inverse scaling***, where increased test-time computation consistently degrades output quality (Cuadron et al., 2025; Hassid et al., 2025; Gema et al., 2025). These observations highlight the need for a clearer understanding of the factors that drive such scaling failures. Yet, systematic analyses of these phenomena remain limited.

Recent findings on text-only LMs provide an important clue regarding the cause of scaling failures (Gema et al., 2025). In particular, prior work highlights the role of ***textual distractors*** in triggering inverse scaling, identifying two consistent patterns: First, the presence of irrelevant information in the context (i.e., distractors) consistently

induces the scaling failure. Second, adding more distractors lengthens the reasoning process, which in turn reduces accuracy (Fig. 1a). These observations suggest that longer reasoning may amplify flawed heuristics introduced by distractors.

In this work, we study whether ***distractors*** introduced through ***visual*** and ***linguistic*** channels induce analogous failure modes in reasoning VLMs. Rather than focusing only on a single distractor modality, we examine a broader class of distractors, including objects or rendered text inserted into the image and irrelevant information appended to the question prompt. Our motivation is twofold. First, real multimodal inputs often contain irrelevant information in both the visual and linguistic modalities, such as background clutter, contextual objects, or distracting descriptions. Second, VLMs may rely on visual and textual evidence in different ways, raising the question of whether distractors affect test-time scaling through their content itself or through the modality in which they are presented.

To systematically examine the effect of visuo-linguistic distractors on the test-time scaling behavior of reasoning VLMs, we construct Idis (Images with distractors), a new VQA benchmark suite built from two task families: perception-centric object classification (Idis-perception) and reasoning-centric visual mathematics problems (Idis-math). Idis comprises over 277k natural and synthetic images featuring target objects alongside one or more distractors, while preserving the original target information and answer.

Using Idis, we reveal that the scaling pattern differs substantially depending on the modality of distraction. Visual distractors degrade accuracy without substantially increasing reasoning length, shifting the length-accuracy curve downward (Figure 1b,c). This effect further depends on distractor severity, including the number of distractors and their semantic relationship to the target information. In contrast, textual distractors appended to the question prompt more closely resemble the pattern reported in reasoning LMs (Gema et al., 2025): they increase reasoning length and thereby intensify test-time inverse scaling (Figure 1d).

To better understand the behavior of reasoning VLMs, we analyze the visual attributes verbalized in reasoning traces. We find that distractor attribute ratio is strongly associated with final accuracy, and attention-blocking experiments further suggest that attributes in the trace directly affect answer formation. Motivated by this finding, we test a simple prompt-based sanity check that encourages models to focus on the target object and reason from its own visual attributes. This reduces reliance on distractor-related cues and improves robustness on Idis. We further show that the same insight extends to Waterbirds (Sagawa et al., 2020), where distractors arise from background context rather than localized inserted objects.

Our key contributions are as follows:

- We introduce **Idis**, a VQA benchmark suite that systematically varies distractor modality, number, and semantic relationship across both perception-centric and reasoning-centric tasks.
- We identify the **modality-dependent effect of distractors** on scaling that departs from prior findings on reasoning LMs (Gema et al., 2025).
- Our **attribute-centric analysis** reveals that the distractor attribute ratio—not reasoning length—drives accuracy for visual distractors, motivating an attribute-guiding prompt strategy that improves robustness on Idis and Waterbirds.

2 Related work

Test-time scaling and distractors. Recent work shows that reasoning models often overthink, spending excessive tokens for only marginal accuracy gains (Sui et al., 2025; Chen et al., 2025; Yeo et al., 2025; Luo et al., 2025). More recent studies further report *inverse scaling*, where accuracy decreases as more test-time token usage increases. Cuadron et al. (2025) observe this phenomenon in interactive settings, such as agentic tasks, with related trends reported in subsequent studies (Hassid et al., 2025; Marjanović et al., 2025; Pham et al., 2025). Most closely related to our work, Gema et al. (2025) show that inverse scaling can arise reliably in the presence of distractors, such as irrelevant text or code. We extend this line of work to reasoning VLMs by formalizing the notion of visual and textual distractors in VQA and studying their impact on test-time scaling.

Distractors in VLMs. Prior work has shown that reasoning VLMs are vulnerable to diverse distractors, including GUI pop-ups (Ma et al., 2025), textual overlays (Deng et al., 2025), and unrelated images (Cai et al., 2025), which can substantially degrade accuracy. Our work differs in three ways. First, we study how distractors affect scaling behavior, rather than accuracy alone. Second, we systematically vary distractor severity along multiple dimensions—including semantic type, number,

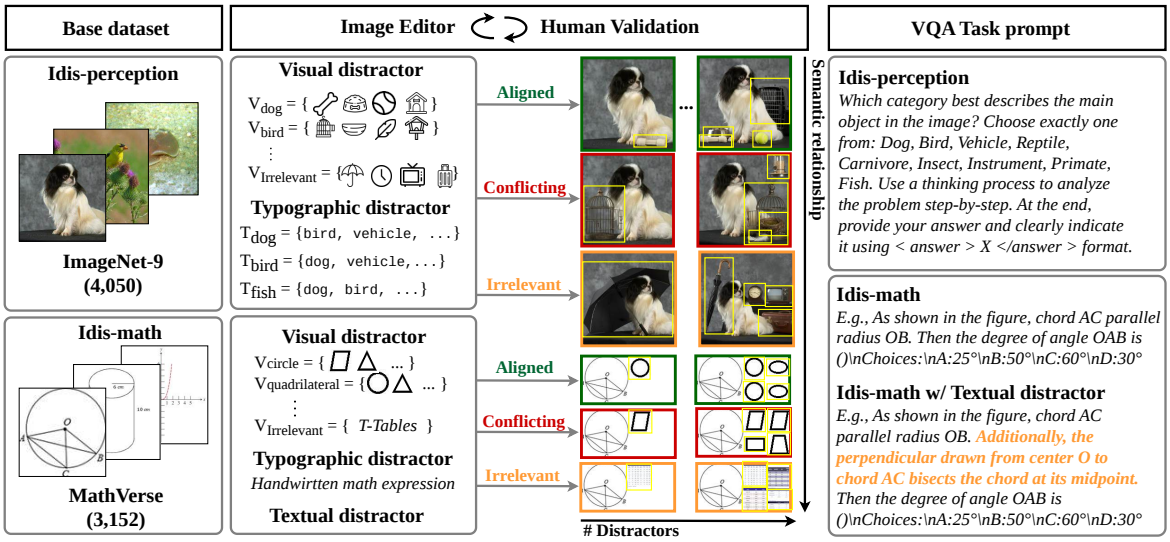


Figure 2: **Idis construction pipeline and VQA task prompt.** Idis is a VQA benchmark suite that aims to evaluate how visual and linguistic distractors affect the test-time scaling behavior of reasoning VLMs. Distractor-augmented images are produced through generative editing or deterministic compositing and manually validated through a human-in-the-loop process. Yellow boxes indicate distractor regions (not included in original images).

and text placement—and compare text distractors embedded in the image with those appended to the prompt. Third, we extend the analysis to reasoning-centric mathematical tasks, providing, to the best of our knowledge, the first systematic study of how distractors affect visual mathematical reasoning.

3 Idis: Images with distractors

To study how distractors affect test-time scaling in reasoning VLMs, we introduce ***Idis*** (Images with distractors), a VQA benchmark that evaluates how visual and linguistic distractors shape the scaling trend. Each sample requires a model to answer a visual question while exposed to distractors to the original problem. Idis comprises two task families: *Idis-perception*, an object classification task based on ImageNet-9 (Xiao et al., 2021), and *Idis-math*, a visual math reasoning task based on MathVerse (Zhang et al., 2024). Together, Idis allows us to study distractor effects in both perception-centric and reasoning-centric settings.

3.1 Distractor design

We design distractors along two broad axes: visual and linguistic. Visual distractors are additional objects inserted into the image. Linguistic distractors, in contrast, introduce distracting linguistic information embedded as an image or as text. We distinguish between typographic distractor, where text is rendered in the image, and textual distractor, where text is inserted into the question prompt.

Semantics of visual distractors. We categorize visual distractors as *aligned*, *conflicting*, or *irrelevant*, according to their semantic relationship with the target information needed to answer the question. For Idis-perception, this relationship is defined at class level: we follow the debiasing literature's notion of correlated features (Nam et al., 2020) and consider whether the distractor is spuriously associated with the target object class. For Idis-math, we use the same terminology analogously, but at the concept level, describing the relationship between the distractor and the mathematical concept required by the original problem.

- *Aligned.* The distractor is positively associated with the target class or belongs to the same concept family as the original problem: In Idis-perception, a bird cage for "bird"; in Idis-math, another circle figure for a circle-related problem.
- *Conflicting.* The distractor is associated with a different class or lies outside the relevant mathematical concept space: In Idis-perception, a bird cage for "vehicle"; in Idis-math, or a quadrilateral for a circle-related problem.
- *Irrelevant.* The distractor has no strong association with the target class or concept: In Idis-perception, a TV; In Idis-math, a table.

For Idis-perception, aligned distractors have the positive semantic relationship with the target, while conflicting distractors have the negative one. Thus, aligned distractors may reinforce the model's correct decision, whereas conflicting distractors may

bias the model toward the distractor-associated class. Irrelevant distractors, in contrast, test robustness to semantically unrelated visual information.

For Idis-math, however, aligned distractors are not necessarily helpful: because distractors are not part of the original problem, they are unnecessary for deriving the correct answer. Here, the aligned/-conflicting/irrelevant taxonomy instead captures the concept-level relationship between the distractor and the mathematical concept needed by the problem. In particular, irrelevant distractors are drawn from the table subset of LogicVista (Xiao et al., 2024), lying outside the geometry-focused problem space of MathVerse (Zhang et al., 2024).

Number. We add one to four distractors to each sample. When multiple distractors are included, they are drawn from the same semantic category (e.g., aligned), allowing us to isolate the effect of distractor quantity while holding the semantic relationship fixed. The construction varies by task. For Idis-perception, multiple distractors differ in fine-grained class; for example, an image containing a dog may be augmented with two aligned distractors such as a dog bowl and a wooden kennel.

Linguistic distractors. Linguistic distractors fall into two types: typographic and textual.

- *Typographic.* A text segment is rendered directly into the image, making the distracting information part of the visual input. In Idis-perception, we insert the class name of non-target classes; in Idis-math, we insert handwritten math expressions (HumynLabs, 2025; Gervais et al., 2025).
- *Textual.* A text segment, generated by Sonnet 4.5 (Anthropic, 2025), is inserted into the question prompt as natural language. This is used only for Idis-math, as the textual prompt is fixed for all samples in Idis-perception.

3.2 Dataset construction pipeline

We construct Idis by systematically introducing distractors into the same underlying VQA examples. Starting from task-specific base datasets, we create distractor-conditioned variants by inserting controlled visual or linguistic distractors while preserving the original target information and answer. Throughout this process, we incorporate human verification to ensure that the target remains unchanged, the answer is preserved, and each distractor matches its intended semantic category.

Crucially, we generate multiple distractor-conditioned samples from each base example. The target object or original math problem is kept fixed, while only the distractor type, semantic relationship, or number is varied. This controlled intervention enables direct comparison between the original and distractor-augmented examples, allowing changes in model predictions and reasoning behavior to be attributed to the inserted distractors.

Idis-perception. We use the "original" split of ImageNet-9 (Xiao et al., 2021) as the base dataset, which contains 4,050 curated images from ImageNet classification benchmark (Deng et al., 2009). These images contain a single salient foreground object with minimal background clutter, enabling precise control over the number and type of distracting objects inserted into each image.

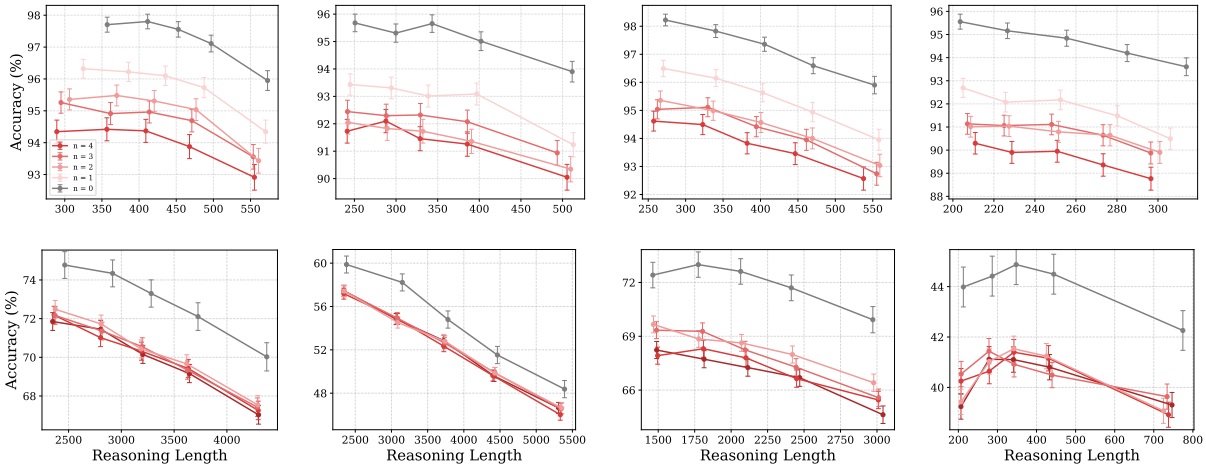
Naïvely compositing distractors into the base image, such as by direct overlay, can introduce artifacts that VLMs detect instead of reasoning about the image content, leading to traces such as *"the image might be a composite or edited photo."* To produce realistic edits while preserving the original target object, we use Gemini 2.5 Flash Image (Google, 2025) for generative editing. For each image, we create a distractor-conditioned variant by sampling k distinct distractor classes from one semantic category: aligned, conflicting, or irrelevant. We then prompt Gemini 2.5 Flash Image to insert the selected distractors while preserving the identity, location, and appearance of the target object. For typographic distractors, we instead overlay the distractor class name directly onto the image.

Idis-math. We construct *Idis-math* by adding controlled distractors to visual math problems from the MathVerse (Zhang et al., 2024), while preserving the original problem structure and answer.

For visual distractors, we deterministically composite distractors around the target diagram on a fixed canvas using PIL, rather than relying on generative image editing. We preserve the original resolution and aspect ratio, and place distractors to avoid overlap with the target figure.

For typographic distractors, we use the same rendering protocol, but insert handwritten math expressions (HumynLabs, 2025; Gervais et al., 2025) instead of geometric figures. These expressions are placed around the target diagram as visually salient but answer-irrelevant distractors.

For textual distractors, we leave the diagram unchanged and insert mathematically plausible but answer-irrelevant sentences into the question prompt. These sentences are constructed to be



(a) Qwen3-VL-Thinking (b) Intern-S1-mini (c) GLM-4.1V-Thinking (d) R1-OneVision

Figure 3: **Test-time scaling vs. #distractors.** Inserting more visual distractors does not extend the reasoning length of reasoning VLMs, but instead shifts the length-accuracy curve downward, in contrast to reasoning LMs. The first and second rows report results on Idis-perception and Idis-math, respectively.

consistent with the diagram, but not necessary for deriving the correct answer.

Human verification. To ensure dataset quality, we perform iterative human verification until all samples in the benchmark satisfy subtask-specific acceptance criteria. Samples that fail verification are either regenerated or excluded. Appendix B details the verification protocol, including selection criteria and filtering statistics.

Other details. We provide omitted details, including the dataset statistics, prompts, and human verification details in Appendix B.

4 Experimental setup

Following Gema et al. (2025), we study *sequential scaling*, where models scale by producing longer reasoning trace rather than aggregating parallel reasoning traces. For each question, we sample five responses, rank them by reasoning length, and compute the accuracy of each rank across all questions. This repeated sampling is deliberate: it lets us compare responses to the same underlying problem, controlling for question difficulty and image complexity. Thus, the resulting length-accuracy curves reflect within-question variation in reasoning length, rather than across-question differences in problem hardness that could otherwise confound the analysis.

Models. We evaluate four open-weight frontier reasoning VLMs in the 7–9B parameter range, spanning diverse architectures and training recipes: Qwen3-VL-8B-Thinking (Qwen Team, 2025), GLM-4.1V-9B-Thinking (GLM-V Team, 2025), Intern-S1-mini (Intern-S1 Team, 2025), and R1-OneVision-7B-RL (Yang et al., 2025). For brevity, we omit parameter counts when referring to these models hereafter.

Reasoning budget. Unlike text-only reasoning models, reasoning VLMs typically lack an explicit mechanism to control the reasoning budget.[1] Thus, we mainly focus on the setting of "natural overthinking," i.e., the models naturally generating extended reasoning (Gema et al., 2025).

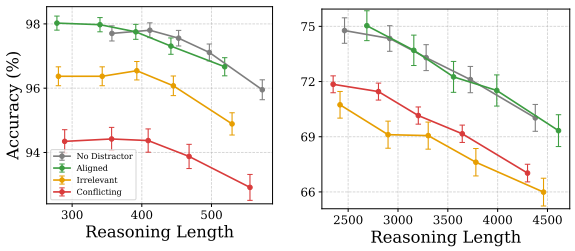
Metrics. Following Gema et al. (2025), we primarily focus on the interplay between three elements: (1) the accuracy; (2) the reasoning length; (3) the number of distractors added to the sample.

5 Results

We summarize our main findings as follows:

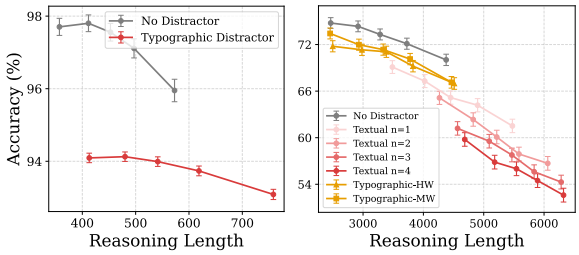
- *Visual distractors* lower the length-accuracy curves without substantially increasing reasoning length; the effect depends on the number of distractors and their semantics (Section 5.1).
- *Linguistic distractors* have modality-dependent effects: typographic distractors mainly shift the curve downward (as visual distractors do), whereas textual distractors reduce accuracy by increasing reasoning length (Section 5.2).

[1] We have also attempted controlling the reasoning length via prompting, which turned out to be ineffective; see Appendix E for details.



(a) Idis-perception (b) Idis-math

Figure 4: **Test-time scaling vs. distractor semantics.** Conflicting distractors cause the largest downward shift, indicating that reasoning VLMs are vulnerable to distractors that semantically conflict with the target object. Results are on Qwen3, when we add four distractors at a time; see Appendix F.1 for other VLMs.



(a) Idis-perception (b) Idis-math

Figure 5: **Test-time scaling vs. linguistic distractors.** Typographic distractors, rendered as an image, has a similar effect with visual distractors of shifting the curve downward with limited increase in reasoning length. Textual distractors, inserted in the question prompt, also reduces the accuracy by increasing the reasoning length. Results are on Qwen3; see Appendix F.1 for other VLMs.

5.1 Scaling vs. visual distractors

Number of distractors. We first put particular focus on how the number of the distractors affects the scaling trend. In reasoning LMs (Gema et al., 2025), adding more textual distractors intensifies test-time inverse scaling. In contrast, Fig. 3 shows that additional visual distractors consistently shift the length-accuracy curve downward without substantially changing the range of reasoning lengths. On Idis-math (bottom row), we observe similar trends, except that the marginal effect of having more than one distractor is notably smaller. We also draw particular attention to the curves for R1-OneVision. Here, the overall shape of the curve is maintained despite the shifts.

Distractor semantics. Next, we study the interplay between the scaling behavior and the semantic properties of the distractors. In Fig. 4, we observe that distractor semantics also tend to shift the scaling curve downward along the y-axis, similar to the effect of increasing the number of distractors. For Idis-perception, the downward shift is large when the distractors are semantically conflicting with the target, while aligned distractors cause little to no degradation compared to the no-distractor baseline. For Idis-math, interestingly, irrelevant samples give the largest drop; this may be due to the fact that "irrelevant" distractors are much OOD-ish for Idis-math, leading to the largest confusion.

Discussion: Severity of distraction. It is noteworthy that both aspects (numerics and semantics) have a very similar effect; the severity of distraction increases as we have greater number of distractors or more confusing ones. In Appendix F.2, we show that "size" also has a similar effect, where the curve shifts down more for larger distractors.

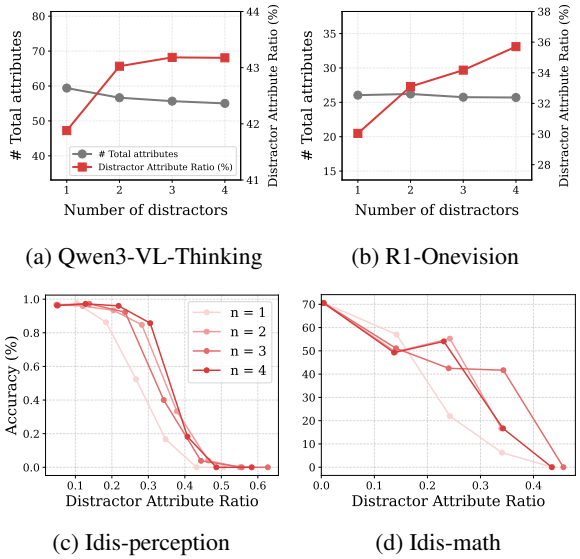
5.2 Scaling vs. linguistic distractors

We next examine linguistic distractors for two reasons. First, they connect directly to prior findings in reasoning LMs, where irrelevant text amplifies test-time inverse scaling (Gema et al., 2025). Second, in multimodal tasks, the same distraction can be introduced visually, by rendering text in the image, or linguistically, by inserting it into the prompt. This lets us test whether reasoning VLMs respond differently depending on the distractor modality.

Typographic distractor. As shown in Figure 5, typographic distractors mirror the pattern observed for visual distractors: they reduce accuracy without substantially expanding the range of reasoning lengths. This trend is less pronounced on Idis-perception, but becomes clearer on Idis-math, where we can compare against textual distractors.

Textual distractor. This pattern does not hold for textual distractors inserted into the question prompt. As shown in Figure 5b, increasing the number of textual distractors induces a clear test-time inverse-scaling pattern, similar to that observed by Gema et al. (2025) in text-only reasoning LMs: much accuracy drop is driven by longer reasoning traces.

Discussion: Modality matters. Together, these observations suggest that the "modality of distraction" has a dominant effect in test-time scaling, instead of whether the distractor has been a visual object or linguistic information.



(a) Qwen3-VL-Thinking (b) R1-Onevision

(c) Idis-perception (d) Idis-math

Figure 6: **Distractor attributes as a meaningful indicator.** In the top row, we find that as we increase the number of distractors, the number of total attributes remain similar but the fraction of distractor attributes consistently increases; see Appendix F.1 for more plots. The bottom row shows that the distractor attribute ratio is strongly correlated with the model accuracy. The plots are for Qwen 3, when we add conflicting distractors. Plots on other models are given in Appendix F.1.

6 Analysis

To better understand this behavior, we conduct an ***attribute-centric*** analysis of reasoning traces (Deng et al., 2025; Shojaee et al., 2025; Qian et al., 2025). Our analysis reveals a potential limitation of reasoning VLMs: *Final predictions are strongly influenced by the raw fraction of distractor attributes in the trace, which itself increases with the number and size of distractors in the image* (Section 6.2).

This observation suggests that the distractor-robust VLM reasoning may require mechanisms that suppress distractor attributes during trace generation. As a sanity check on this insight, we also explore a prompt-based mitigation strategy that can be applied at the inference time.

6.1 Tools

For our analysis, we utilize the following tools.

Attribute extraction. We parse visual attributes mentioned in each trace and link them to the corresponding target or distractor object. We use DeepSeek-V3.2-Exp (DeepSeek-AI, 2025) with structured instructions; see Appendix B.1 for procedural details and Appendix B.5 for the validation of this extraction process.

Object area. We measure the pixel area of target and distractor objects and analyze its relation to accuracy and distractor attributes. We obtain masks with LangSAM (Medeiros, 2025), using each object description as the prompt, and compute area by counting masked pixels.

Detailed formula. see Appendix B.2

6.2 The role of distractor attributes

The upper row of Figure 6 shows that (i) the number of total attributes in each trace remains similar as we vary the number of distractors;[2] (ii) the fraction of distractor attributes, on the other hand, steadily grows as we increase the number of distractors.

The lower row of Figure 6 shows that the distractor attribute ratio is strongly correlated with the final prediction. The accuracy drops to near zero whenever the distractor attribute ratio exceeds 50%, yet remains above 97% whenever the ratio is below 20%. We observe a similar pattern in Idis-math, where accuracy drops to zero when the distractor attribute ratio exceeds 0.4.

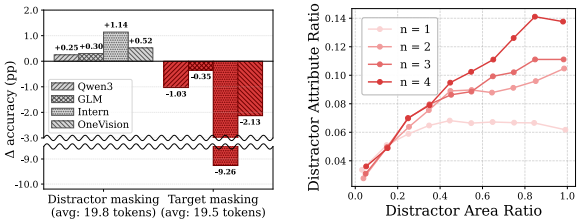
Discussion: Which causes which? Together, the observations suggest that the distractor attribute ratio is highly correlated with the final prediction. Is final prediction guiding the reasoning trace, or is the reasoning trace guiding the final prediction? The attention blocking experiment (Fig. 7a) suggests that the latter is more plausible than the former. By blocking the attention going to the distractor (or target) attribute tokens, one can dramatically change the model accuracy.

What affects distractor attribute ratio? Figure 7b suggests that the distractor attribute ratio is correlated with not only the number of distractors, but also the distractor area ratio. In Appendix F.2, we supplement this observation with the experiments on a dataset where we manually control the size of the distractors (dubbed *Idis-manual*), which confirm the effect of distractor sizes on the distractor attribute ratio.

6.3 Sanity check: Prompt-based mitigation

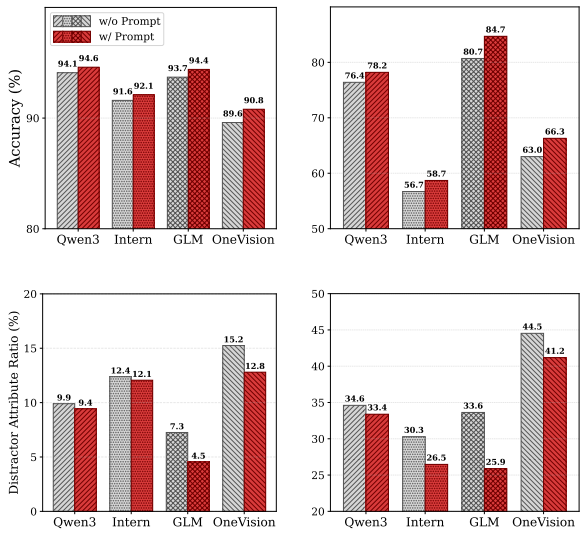
This observation suggests that the VLM reasoning under distractor-prone environments may benefit from the mechanisms that suppress distractor attributes and facilitate target attributes during the reasoning trace generation.

[2]This is well-expected since the reasoning length remains similar (Fig. 3). Indeed, the number of attributes is tightly correlated with the reasoning length; see Appendix F.1



(a) Token attention blocking (b) Distractor Area Ratio

Figure 7: **Closer look at the distractor attributes.** (a) Reasoning trace attributes causally shape answers: blocking the attention of the final answer tokens to distractor-related attributes improves accuracy, while blocking target-related attributes degrades it. (b) Distractor attributes increase with distractor area and saturate at a higher ratio when a greater number of distractors are present in the image.



(a) Idis-perception (b) Waterbirds

Figure 8: **Specialized prompts can improve accuracy and reduce focusing on distractors.** Top row: The prompting strategy consistently improves the accuracy across all models, on both Idis-perception and Waterbirds. Bottom row: The prompt decreases the fraction of distractor-related attributes in the reasoning trace, which is likely to have caused the accuracy gain.

To validate this idea, we design a simple sanity check based on ***prompting.*** Concretely, we prompt the model in two ways: (i) the original prompts, and (ii) a prompt that explicitly instructs the reasoning model to base its prediction on the target object itself rather than distractor or background. For example, to avoid focusing on non-target objects, one can add sentences like "First identify the main object. Base your reasoning only on that object's own visual attributes." to the original prompt.

Experimental setup. We test this idea on two datasets: Idis-perception, and Waterbirds (Sagawa et al., 2020). Waterbirds is a standard benchmark in the debiasing literature, where the task is to correctly classify waterbirds from landbirds, where the background may be either the water or the land; the model predictions are often overly reliant on the background cues, making erroneous predictions on images with bias-conflicting backgrounds, e.g., waterbirds on land. We design prompts that instructs the model to focus on the target (or foreground) objects for each of these tasks; see Appendix B.4 for the actual prompts.

Results. The experimental results are given in Figure 8. The top row provides comparisons on the the accuracy of reasoning VLMs with and without specialized prompts. On the Idis-perception dataset, we observe a slight yet consistent increase in accuracy, ranging from 0.5%p to 1.2%p. The accuracy gain is more pronounced on the Waterbirds dataset, ranging from 1.8%p to 4.0%p.

The results on the bottom row of Figure 8 demonstrates that such an accuracy gain is well-correlated with the decrease in the fraction of distractor-related attributes inside the reasoning trace. The distractor attribute ratio decreases quite consistently over all models and on both datasets.

Takeaway. The proposed prompting strategy *per se* is somewhat limited in its ability to improve the model accuracy. However, the results suggest that it might be possible to mitigate the reasoning model's vulnerability toward blindly extracting and distractor-related attributes in reasoning traces.

7 Conclusion

We study the test-time scaling behavior of reasoning VLMs by introducing Idis, a VQA benchmark suite for analyzing visual and linguistic distractors across perception-centric and reasoning-centric tasks. We show that distractor modality shapes scaling behavior: visual and typographic distractors reduce accuracy without substantially increasing reasoning length, whereas textual distractors intensify the test-time inverse scaling. Our attribute-level analysis further shows that reasoning VLMs are strongly influenced by distractor-related attributes verbalized in their traces. Based on this insight, we explore a attribute-guide prompt that improves robustness on both Idis and Waterbirds. We hope this work informs future efforts to build distractor-robust reasoning VLMs.

Limitations

Our analyses focus on a VQA domain, which provides a controlled setup to analyze the effects of visual and textual distractors. Extending this framework to more complex reasoning-heavy tasks, such as agentic decision making or multi-step planning, remains challenging and is an essential next step. Furthermore, our findings suggest several opportunities for designing mitigation strategies that guide models toward task-related evidence, but developing such methods in a more systematic and generalizable way remains future work.

References

Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K Arora, Yu Bai, Bowen Baker, Haiming Bao, and 1 others. 2025. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*.

Anthropic. 2025. Claude sonnet 4.5 system card. https://www.anthropic.com/claude-sonnet-4-5-system-card.

Rui Cai, Bangzheng Li, Xiaofei Wen, Muhao Chen, and Zhe Zhao. 2025. Diagnosing and mitigating modality interference in multimodal large language models. *arXiv preprint arXiv:2505.19616*.

Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu, Mengfei Zhou, Zhuosheng Zhang, and 1 others. 2025. Do NOT think that much for 2+3=? on the overthinking of o1-like LLMs. In *ICML*.

Alejandro Cuadron, Dacheng Li, Wenjie Ma, Xingyao Wang, Yichuan Wang, Siyuan Zhuang, Shu Liu, Luis Gaspar Schroeder, Tian Xia, Huanzhi Mao, and 1 others. 2025. The danger of overthinking: Examining the reasoning-action dilemma in agentic tasks. *arXiv preprint arXiv:2502.08235*.

DeepSeek-AI. 2025. Deepseek-v3.2-exp: Boosting long-context efficiency with deepseek sparse attention.

Ailin Deng, Tri Cao, Zhirui Chen, and Bryan Hooi. 2025. Words or vision: Do vision-language models have blind faith in text? In *CVPR*.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *CVPR*.

Aryo Gema, Alexander Hägele, Runjin Chen, Andy Arditi, Jacob Goldman-Wetzler, Kit Fraser-Taliente, Henry Sleight, Linda Petrini, Julian Michael, Beatrice Alex, and 1 others. 2025. Inverse scaling in test-time compute. *TMLR*.

Philippe Gervais, Anastasiia Fadeeva, and Andrii Maksai. 2025. Mathwriting: A dataset for handwritten mathematical expression recognition. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 5459–5469.

GLM-V Team. 2025. Glm-4.5v and glm-4.1v-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning.

Google. 2025. Gemini 2.5 flash image. https://developers.googleblog.com/en/gemini-2-5-flash-image-now-ready-for-production-with-new-aspect-ratios/.

Michael Hassid, Gabriel Synnaeve, Yossi Adi, and Roy Schwartz. 2025. Don't overthink it. preferring shorter thinking chains for improved llm reasoning. *arXiv preprint arXiv:2505.17813*.

HumynLabs. 2025. English handwritten math notes dataset. Hugging Face Datasets. CC BY 4.0 license.

Intern-S1 Team. 2025. Intern-s1: A scientific multimodal foundation model. *arXiv preprint arXiv:2508.15763*.

Haotian Luo, Li Shen, Haiying He, Yibo Wang, Shiwei Liu, Wei Li, Naiqiang Tan, Xiaochun Cao, and Dacheng Tao. 2025. o1-pruner: Length-harmonizing fine-tuning for o1-like reasoning pruning. *arXiv preprint arXiv:2501.12570*.

Xinbei Ma, Yiting Wang, Yao Yao, Tongxin Yuan, Aston Zhang, Zhuosheng Zhang, and Hai Zhao. 2025. Caution for the environment: Multimodal LLM agents are susceptible to environmental distractions. In *ACL*.

Sara Vera Marjanović, Arkil Patel, Vaibhav Adlakha, Milad Aghajohari, Parishad BehnamGhader, Mehar Bhatia, Aditi Khandelwal, Austin Kraft, Benno Krojer, Xing Han Lù, and 1 others. 2025. Deepseek-r1 thoughtology: Let's think about llm reasoning. *arXiv preprint arXiv:2504.07128*.

Luca Medeiros. 2025. Language segment-anything. https://github.com/luca-medeiros/lang-segment-anything.

Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. 2020. Learning from failure: Training debiased classifier from biased classifier. In *NeurIPS*.

Thinh Pham, Nguyen Nguyen, Pratibha Zunjare, Weiyuan Chen, Yu-Min Tseng, and Tu Vu. 2025. SealQA: Raising the bar for reasoning in search-augmented language models. *arXiv preprint arXiv:2506.01062*.

Chen Qian, Dongrui Liu, Haochen Wen, Zhen Bai, Yong Liu, and Jing Shao. 2025. Demystifying reasoning dynamics with mutual information: Thinking tokens are information peaks in llm reasoning. In *NeurIPS*.

Qwen Team. 2025. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*.

Qwen Team. 2026. Qwen3.5: Towards native multimodal agents.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. 2020. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *ICLR*.

Parshin Shojaee, Iman Mirzadeh, Keivan Alizadeh, Maxwell Horton, Samy Bengio, and Mehrdad Farajtabar. 2025. The illusion of thinking: Understanding the strengths and limitations of reasoning models via the lens of problem complexity. In *NeurIPS*.

Yang Sui, Yu-Neng Chuang, Guanchu Wang, Jiamu Zhang, Tianyi Zhang, Jiayi Yuan, Hongyi Liu, Andrew Wen, Shaochen Zhong, Na Zou, Hanjie Chen, and Xia Hu. 2025. Stop overthinking: A survey on efficient reasoning for large language models. *TMLR*.

Kai Xiao, Logan Engstrom, Andrew Ilyas, and Aleksander Madry. 2021. Noise or signal: The role of image backgrounds in object recognition. In *ICLR*.

Yijia Xiao, Edward Sun, Tianyu Liu, and Wei Wang. 2024. Logicvista: Multimodal llm logical reasoning benchmark in visual contexts. *arXiv preprint arXiv:2407.04973*.

Yi Yang, Xiaoxuan He, Hongkun Pan, Xiyan Jiang, Yan Deng, Xingtao Yang, Haoyu Lu, Dacheng Yin, Fengyun Rao, Minfeng Zhu, and 1 others. 2025. R1-OneVision: Advancing generalized multimodal reasoning through cross-modal formalization. In *ICCV*.

Edward Yeo, Yuxuan Tong, Morry Niu, Graham Neubig, and Xiang Yue. 2025. Demystifying long chain-of-thought reasoning in llms. *arXiv preprint arXiv:2502.03373*.

Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou, Pan Lu, Kai-Wei Chang, Yu Qiao, and 1 others. 2024. Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems? In *ECCV*.

N	Resolution	Images per semantic category			
		Conflicting	Irrelevant	Aligned	Total
0	224×224	–	–	–	4,050
Visual distractor					
1	1024×1024	4,050	4,050	4,050	12,150
2	1024×1024	4,050	4,050	4,050	12,150
3	1024×1024	4,050	4,050	4,050	12,150
4	1024×1024	4,050	4,050	4,050	12,150
Linguistic distractor: Typographic					
1	1024×1024	32,400	–	–	32,400
2	1024×1024	32,400	–	–	32,400
3	1024×1024	32,400	–	–	32,400
4	1024×1024	32,400	–	–	32,400

Table 1: **Dataset statistics of *Idis-perception***. The no-distractor setting corresponds to the original ImageNet-9 split. Visual distractors are generated for each semantic category and distractor number. Typographic distractors are constructed by rendering the class names of non-target categories directly into the image.

N	Samples per distractor category						Total
	Aligned	Conflicting	Irrelevant	Handwritten	MathWriting	Textual	
0	–	–	–	–	–	–	3,152
Visual distractor							
1	2,684	7,676	3,152	–	–	–	13,512
2	2,684	7,676	3,152	–	–	–	13,512
3	2,684	7,676	3,152	–	–	–	13,512
4	2,684	7,676	3,152	–	–	–	13,512
Linguistic distractor: Typographic							
1	–	–	–	3,152	3,152	–	6,304
2	–	–	–	3,152	3,152	–	6,304
3	–	–	–	3,152	3,152	–	6,304
4	–	–	–	3,152	3,152	–	6,304
Linguistic distractor: Textual							
1	–	–	–	–	–	3,152	3,152
2	–	–	–	–	–	3,152	3,152
3	–	–	–	–	–	3,152	3,152
4	–	–	–	–	–	3,152	3,152

Table 2: **Dataset statistics of *Idis-math***. The no- distractor setting corresponds to the original MathVerse. Visual distractors are organized by their semantic relationship to the target class and distractor number. Textual distractors are inserted into the question prompt rather than the image.

A Pipeline details of Idis

A.1 Dataset statistic

We summarize the composition of *Idis* in Tables 1 and 2. The benchmark consists of two task families, *Idis-perception* and *Idis-math*.

Idis-perception. *Idis-perception* is built from 4,050 images in the original split of ImageNet-9. There are a total of nine classes in the dataset (Precisely, the classes are: bird, carnivore, dog, fish, insect, instrument, primate, reptile, and vehicle.), with 450 samples for each class. The nine classes are the coarse-grained superclasses for the ImageNet classes determined based on the WordNet hierarchy. We use these images as the no-distractor baseline, where each image contains a single salient target object at a resolution of 224×224. For visual-object distractors, we vary the number of distractors from $N = 1$ to $N = 4$. For each distractor number and each semantic category—aligned, conflicting, and irrelevant—we generate one augmented image for every base image, resulting in 4,050 images per semantic category and 12,150 images per distractor-number setting. All distractor-augmented images are generated at a resolution of 1024×1024. The typographic distractor condition inserts non-target class-name text into the image, producing eight augmented variants for each sample, one for each non-target class.

Idis-math. *Idis-math* is built from MathVerse and contains both visual and linguistic distractor conditions. MathVerse consists of 2,612 high-quality mathematical questions, each paired with variants that provide different degrees of multimodal information. As our base dataset, we use the Test Mini split under four settings—Text Dominant, Text Lite, Vision Intensive, and Vision Dominant—which contain approximately 3,152 samples. We consider six distractor conditions: aligned, conflicting, irrelevant, handwritten, mathwriting, and textual. The first three correspond to visual-object distractors, whereas handwritten, mathwriting, and textual distractors are treated as linguistic distractors.

For visual distractors, we vary the distractor density from $N = 1$ to $N = 4$. The resulting visual subset contains 40,536 augmented question-image pairs in total: 2,684 aligned, 7,676 conflicting, 3,152 irrelevant samples. To construct irrelevant visual distractors, we use LogicVista (Xiao et al., 2024) as an external source pool. LogicVista (Xiao et al., 2024) evaluates the logical reasoning abilities of VLMs across spatial, deductive, inductive, numeric, and mechanical reasoning, and contains 448 visual multiple-choice questions. We use the the table-capability subset as the seed pool.

Linguistic distractors fall into two types: typographic and textual. For textual distractors, we keep the original image unchanged and insert answer-irrelevant sentences into the question prompt. Typographic distractors are rendered directly into the image. We insert handwritten mathematical expressions (HumynLabs, 2025; Gervais et al., 2025). Lastly, following Gema et al. (2025), we generate textual distractors with Sonnet 4.5 (Anthropic, 2025) using the prompt in Table 5, and insert them into the question prompt as natural-language statements.

	Objects			
Class	Object 1	Object 2	Object 3	Object 4
Dog	dog bone chew toy	dog bowl	tennis ball	kennel
Bird	birdcage	nest	feather	bird feeder
Wheeled Vehicle	tire	steering wheel	license plate	bumper
Reptile	terrarium rock	heat lamp	log hideout	shed skin
Carnivore	toy fang	fake blood stain	fake chunk of meat	toy skeletal animal carcass
Insect	trash bag	empty net designed for catching insects	fruit peel	flower
Musical Instrument	chalkboard with music notes	music stand	metronome	sheet music
Primate	patch of jungle foliage	banana	coconut	vine
Fish	fishing rod	large empty nylon fishing net	life jacket	aquarium coral ornament
Irrelevant	umbrella	clock	TV	suitcase

Table 3: **List of defined objects per class.** Each semantic class is associated with four representative objects used in distractor generation.

A.2 Distractor lists and pipeline

Defined object. As shown in Table 3, we define a set of class-relevant objects for each class, as well as all class-irrelevant objects. For class-relevant objects, we first prompted GPT-5 to generate candidate items associated with each category using the following prompt.

> ***Prompt***: What objects are typically aligned with {class}? I'm curious about which objects commonly co-occur with {class} in images. Please focus on objects that are suitable for segmentation.

Through human evaluation, these candidates were filtered and refined, and the four most appropriate objects were finally selected. For the irrelevant category, we followed the same procedure, using object candidates from the MS-COCO (COCO 2017) dataset.

A.3 Human validation and statistics

Dataset construction was carried out with human-in-the-loop validation at every stage. Generated images were manually inspected, and images that did not satisfy our quality criteria were regenerated and re-evaluated by human annotators. We repeated this iterative validation process until all examples in the final dataset met the required standards. We summarize the validation criteria and the statistics of filtered cases below.

Idis-perception. (a) Criteria

- The target object disappears or is not preserved.
- The distractor differes from the intended concept.
- An object from other class is generated.
- The distractor occludes the target object.
- Class-indicative text appears in the image and could serve as a hint.

(b) Statistics

- Regeneration cases: 367/48,600 (0.76%)

Idis-math. To enable the construction of aligned and conflicting visual distractors, we first categorize each source diagram into a coarse mathematical concept family, such as triangle, quadrilateral, circle family, or three-dimensional shape family. The labels are produced through automatic labeling, model-assisted annotation, and human verification, and only human-confirmed labels are used. Based on these labels, aligned distractors are selected from the same concept family as the source diagram, whereas conflicting distractors are selected from different concept families.

A.4 Prompts for dataset generation

To generate each distractor-conditioned sample in the Idis-perception, we provide the image generation model with a structured text prompt describing the target class and the distractor type to be inserted. The generic prompt template is shown below:

> ***Prompt***: Using the provided image of a {Class}, please add {Aligned / Conflicting / Irrelevant Distractors} to the scene. Ensure the changes are seamlessly integrated into the natural setting. Do not modify or obscure the {Class}.

For example, if the target class is `dog` and the distractor category is the conflicting class `bird`, the instantiated prompt becomes:

> ***Prompt***: Using the provided image of a **dog**, please add **birdcage**, **nest**, **feather**, **bird feeder** to the scene. Ensure the changes are seamlessly integrated into the natural setting. Do not modify or obscure the **dog**.

Comparison	Pearson r	Notes
Ours vs DeepSeek-V3.2-Exp + alternative prompt	0.803	Prompt sensitivity
Ours vs Qwen3.5-27B	0.785	Alternative LLM
Ours vs Strict lexical matcher	0.476	Non-LLM baseline

Table 4: **Robustness of attribute extraction across alternative extractors.**

B Implementation details

B.1 Experimental protocol

Extract area of each object. To quantify the spatial prominence of visual entities, we estimate the area of both target and distractor regions using text-conditioned segmentation. We employ LangSAM to produce binary masks conditioned on class-level textual prompts (e.g., "feather", "birdcage"), allowing predicted mask labels to be matched against either the target class or the distractor object list. All masks whose labels belong to the target class are merged via pixelwise union into a single target mask, and likewise, all masks whose labels correspond to the distractor object list are merged into a single distractor mask. The area of each region is computed as the number of pixels in the resulting aggregated mask, which we use in downstream analyses such as distractor–target area ratios and their effects on accuracy and reasoning length.

Attribute extraction procedure. To characterize how visual evidence is used within the model's reasoning process, we extract fine-grained visual attributes directly from the generated reasoning traces. For each trace, we provide the full text together with structured, class-aware instructions to a specialized large language model (DeepSeek-V3.2-Exp) (See Tables 6 and 7). The model is guided to operate purely as an evidence extractor: it must identify literal words or phrases in the reasoning text that denote observable attributes, or class-related features. For Idis-perception, we define 10 attribute categories corresponding to the nine semantic classes plus an 'other' category. For Idis-math, we instead define three attribute categories: target-related, distractor-related, and other. To ensure consistency, the extractor follows a small set of rules. It only selects literal words or phrases that explicitly appear in the reasoning trace, without paraphrasing or inference. Each extracted phrase is then assigned to exactly one of the ten predefined categories. Multi-word expressions such as long tail are treated as a single attribute.

An analogous extraction procedure is conducted for predictions on the Waterbirds dataset. In this case, we use a separate set of instruction prompts tailored to the ecological and environmental cues that are central to Waterbirds, as shown in Table 8.

B.2 Metrics.

To quantify how visual distractors affect the model's reasoning behavior, we define two metrics as follows.

Distractor Attribute Ratio measures the proportion of extracted attributes that are allocated to distractors in the reasoning trace:

$$r_{\text{attr}} = \frac{\#\text{Distractor Attributes}}{\#\text{Total Attributes}}. \quad (1)$$

Distractor Area Ratio quantifies distractor spatial size relative to the target:

$$r_{\text{area}} = \frac{A_{\text{dist}}}{A_{\text{dist}} + A_{\text{target}}}. \quad (2)$$

Here, A_{dist} denotes the distractor area, and A_{target} denotes the target object area.

B.3 Hardware

All inference experiments were conducted on a single NVIDIA RTX A6000 and RTX A6000 Ada GPU using BF16 precision.

B.4 Prompts

For Idis-math evaluation, we follow the VLMEvalKit protocol and use gpt-oss-120b (Agarwal et al., 2025) as the judge model.

Inference prompt. We employ different inference VQA task prompts depending on the dataset:

Prompts for Idis-perception

- **VQA Task prompt:** [Question] Which category best describes the main object in the image? Choose exactly one from: Dog, Bird, Vehicle, Reptile, Carnivore, Insect, Instrument, Primate, Fish. Use a thinking process to analyze the problem step-by-step. At the end, provide your answer and clearly indicate it using $< answer > X < /answer >$ format.
- **Prompt strategy:** [Question] Which category best describes the main object in the image? Choose exactly one from: Dog, Bird, Vehicle, Reptile, Carnivore, Insect, Instrument, Primate, Fish. First identify the main object. Base your reasoning only on that ob-

ject's own visual attributes, and determine which of the nine categories best matches the object. Do not assume that the most visually dominant object is the main object. Do not infer the category from surrounding objects or from items that are merely associated with another category. At the end, provide your answer as $< answer > X < /answer >$ format.

Prompts for Waterbirds

- **Reasoning model prompt:** [Question] Is the bird in the image a waterbird or a landbird? Use a thinking process to analyze the problem step-by-step. At the end, provide your answer and clearly indicate it using $< answer > X < /answer >$ format.
- **Prompt strategy:** [Question] Is the bird in the image a waterbird or a landbird? Think step by step based on the foreground bird's attributes. At the end, select your answer from the provided options and clearly indicate it using $< answer > X < /answer >$ format.

Attribute extraction prompt. We use two types of extraction prompts, depending on the dataset

- Idis attribute extraction: The class-aware attribute extraction prompt shown in Table 6.
- Waterbirds attribute extraction: The biological/environmental attribute extraction prompt shown in Table 8.

B.5 Validation of the attribute-extraction pipeline.

we evaluated the robustness of our findings across alternative extraction pipelines. Our extracted scores show strong correlation with both (1) an alternative prompting strategy (Pearson r = 0.803; see Table 4 for the prompt) and (2) a different base LLM, Qwen3.5-27B (Qwen Team, 2026) (r = 0.785). In contrast, a strict lexical matching baseline yields substantially weaker correlation (Pearson r = 0.476), suggesting that simple word matching is insufficient for this analysis. Taken together, these results indicate that our findings are robust to reasonable variations in the extraction pipeline while still requiring semantic parsing beyond a non-LLM baseline.

Prompts for Waterbirds

- **Alternative prompt:** You are analyzing a chain-of-thought reasoning trace.

  Task: Extract every concrete noun phrase that literally appears in the text, and classify each into one of these 10 categories based on what the noun phrase typically refers to: dog, bird, vehicle, reptile, carnivore, insect, instrument, primate, fish, other.

  The "other" category is for noun phrases that don't naturally belong to any of the nine main classes.

  Rules: - Only use words/phrases that literally appear in the text. Do not paraphrase or infer. - Multi-word noun phrases count as one item (e.g., "long tail" -> one attribute). - "Dog", "bird", and other class words themselves count if they literally appear. - Each extracted item must be assigned to exactly one of the 10 categories.

  Expected JSON output: [SAME with original prompt]

System prompt for textual distractor generation

You are a math education expert. Your task is to insert distractor sentences into a math question to test whether a solver can ignore irrelevant information and focus on what actually matters.

I have a math question with an accompanying figure. I want you to add `{num_distractors}` distractor sentences into the question text. The distractors should be inserted between the setup/given conditions and the final question, blending naturally into the text.

```
<original_question>
{question}
</original_question>

<correct_answer>
{answer}
</correct_answer>

<subject>
{subject} / {subfield}
</subject>
```

Rules:

1. Distractors MUST be mathematically related to the question's domain, such as geometry or algebra, but MUST NOT affect the correct answer.
2. Distractors should introduce plausible-sounding information, such as extra angles, lengths, relationships, theorems, or observations, that seems relevant but is actually irrelevant to solving the problem.
3. The distractors should reward careful thinking about what information is actually needed.
4. The correct answer MUST remain exactly `{answer}` after inserting distractors.
5. Distractors should blend naturally into the question text; they should read as if they were always part of the problem statement.
6. Do NOT change, remove, or rephrase the original sentences. Only INSERT new distractor sentences.
7. After generating, verify that the answer is still `{answer}` by mentally solving the augmented question. If it is not, regenerate.
8. Output ONLY the complete augmented question with distractors inserted. Do not output the answer, explanation, tags, or anything else.
9. Keep the Choices section exactly as-is at the end.

Instruction: Only output the complete augmented question. No explanations or extra text.

Table 5: **System prompt for textual distractor generation in Idis-math.**

System prompt for class-wise attribute extraction

You are an expert in analyzing a model's chain-of-thought. Extract literal evidence words or phrases from the text and classify them into 10 categories: nine main classes (`dog`, `bird`, `vehicle`, `reptile`, `carnivore`, `insect`, `instrument`, `primate`, `fish`) and one "`other`" category for anything else. For each main class, include attributes or objects directly related to it, considering morphology, taxonomy, features, shape, size, or adaptations.

Representative related objects:

- **dog_attributes:** dog bone chew toy, dog bowl, tennis ball, kennel
- **bird_attributes:** birdcage, nest, feather, bird feeder
- **vehicle_attributes:** tire, steering wheel, license plate, bumper
- **reptile_attributes:** terrarium rock, heat lamp, log hideout, shed skin
- **carnivore_attributes:** fang, blood stain, chunk of meat, skeletal animal carcass
- **insect_attributes:** trash bag, insect net, fruit peel, flower
- **instrument_attributes:** chalkboard with music notes, music stand, metronome, sheet music
- **primate_attributes:** jungle foliage, banana, coconut, vine
- **fish_attributes:** fishing rod, fishing net, life jacket, aquarium coral ornament
- **other_attributes:** unrelated attributes or objects (e.g., umbrella, clock, TV, suitcase)

Rules:

1. Use only literal words/phrases from the text (case-insensitive match for listed objects).
2. Multi-word phrases (e.g., "long tail") count as one attribute.
3. Do not infer or paraphrase.
4. "Taxonomic labels" like "bird" or "dog" are valid only if they literally appear.
5. Each extracted attribute must belong to exactly one of the ten categories.

Expected JSON output:

```
{
"dog_attributes": [...],
"bird_attributes": [...],
"vehicle_attributes": [...],
"reptile_attributes": [...],
"carnivore_attributes": [...],
"insect_attributes": [...],
"instrument_attributes": [...],
"primate_attributes": [...],
"fish_attributes": [...],
"other_attributes": [...],
"counts": { "dog": <int>, "bird": <int>, "vehicle": <int>, "reptile": <int>,
"carnivore": <int>, "insect": <int>, "instrument": <int>, "primate": <int>,
"fish": <int>, "other": <int> }
}
```

Instruction: Only output the JSON object. No explanations or extra text.

Table 6: **System prompt for Idis-perception.**

System prompt for target–distractor attribute extraction

You are analyzing a model's mathematical visual reasoning trace for the MathVerse conflicting setting, where a TARGET geometric figure (the figure the original question is actually about) coexists with one or more DISTRACTOR figures (extra shapes inserted into the image that are NOT the subject of the question).

The user message provides four pieces of context for this sample:

TARGET shapes: `<comma-separated shape names>`
DISTRACTOR shapes: `<comma-separated shape names>`
QUESTION: `<the original problem statement, text-only version>`
Raw text: `a single sentence from the model's reasoning trace.`

Use the QUESTION as the authoritative description of the TARGET. Any entity, point, line, segment, angle, length, value, relation, or option in the QUESTION belongs to the TARGET context; when the same entity or a clear anaphoric reference appears in Raw text, classify it as `target_related`.

Task: Extract every concrete mathematical, diagrammatic, or visual attribute phrase that literally appears in Raw text, and assign each phrase to exactly one of `target_related`, `distractor_related`, or `other`.

Definitions:

- **`target_related`:** phrases referring to the TARGET shape; any entity, property, measurement, relation, value, or option mentioned in the QUESTION; or a part/property/operation clearly attached to the TARGET.
- **`distractor_related`:** phrases referring to a DISTRACTOR shape or to a part/property/measurement/relation/operation explicitly tied to a DISTRACTOR in Raw text.
- **`other`:** concrete attribute phrases that cannot be confidently assigned to either TARGET or DISTRACTOR, including generic math terms, unrelated entities, or ambiguous mentions.

Rules:

1. Use only words/phrases that literally appear in Raw text; do not infer, translate, or paraphrase. The QUESTION is for grounding only.
2. Assign `target_related` if the phrase names or clearly refers to anything introduced by the QUESTION or to the TARGET shape. Anaphora counts only when unambiguous.
3. Assign `distractor_related` only when Raw text explicitly links the phrase to a DISTRACTOR shape.
4. If ambiguous, assign `other`. Multi-word phrases count as one item.
5. Keep duplicates only when they have meaningfully different wording.
6. Do not extract generic discourse words such as "step," "approach," "question," or "answer" unless tied to a concrete option or shape.
7. Respond strictly as one JSON object and nothing else.

Expected JSON output:

```
{
"target_related": [...],
"distractor_related": [...],
"other": [...],
"counts": {
"target_related": <int>,
"distractor_related": <int>,
"other": <int>
}
}
```

Instruction: Only output the JSON object. No explanations or extra text.

Table 7: **System prompt for Idis-math.**

System prompt for bio/env attribute extraction

You are an expert in analyzing a model's chain-of-thought. Your job is to pull out the concrete evidence words or phrases the model itself cites and sort them into two buckets.

Buckets

- **bio attribute**: many morphological part, taxonomic label, features, adaptations or size/shape descriptor of the foreground object (e.g., wings, webbed feet, long legs, body shape, long tail, petrel).
- **env attribute**: physical background or habitat terms that locate the scene (e.g., forest path, reeds, lake, ocean, coastal zone, sky, sand).

Respond strictly in this JSON format:

```
{
"bio_attributes": [ . . . ],
"env_attributes": [ . . . ],
"bio_count":     <integer>,
"env_count":     <integer>
}
```

Rules for extracting attributes

1. A multi-word phrase like "long neck" counts as one attribute.
2. Do not invent attributes; use only words or phrases literally present in the model output.

Examples

Example Input 1:

"The bird has a thick body, similar to a juvenile albatross, which are seabirds adapted to marine environments. They spend most of their time at sea and rely on oceanic ecosystems."

Output: `{"bio_attributes": ["thick body","juvenile albatross","seabirds","adapted to marine environments"], "env_attributes": ["sea","oceanic ecosystems"], "bio_count": 4, "env_count": 2}`

Example Input 2:

"The image shows a small animal with a light-colored face, dark eyes, and a body that's mostly light brown or beige. It has a small head with pointed ears, and its front paws are visible."

Output: `{"bio_attributes": ["light-colored face","dark eyes","light brown","beige","small head","pointed ears","front paws"], "env_attributes": [], "bio_count": 7, "env_count": 0}`

Example Input 3:

"The background has a body of water, like a pond or lake, and the bird is near that. Also, waterbirds often have adaptations for aquatic life, like webbed feet (though here it's a statue, but the context). The setting with water suggests it's a waterbird."

Output: `{"bio_attributes": ["adaptations for aquatic life","webbed feet","waterbird"], "env_attributes": ["background","body of water","pond","lake","water"], "bio_count": 3, "env_count": 5}`

Input:

{model output text here}

Output: {JSON object only; no additional text}

Table 8: **System prompt for Waterbirds dataset.**

C Qualitative examples

C.1 Idis-perception: Visual distractors

All qualitative examples in this section are generated by Qwen3-VL-8B-Thinking.

Prompt

[Question] Which category best describes the main object in the image? Choose exactly one from: Dog, Bird, Vehicle, Reptile, Carnivore, Insect, Instrument, Primate, Fish.
Use a thinking process to analyze the problem step-by-step.
At the end, provide your answer and clearly indicate it using <answer>X</answer> format.

Expected Answer

Fish

(Left: no distractor. Right: adding one conflicting distractor from the vehicle class.)

Response without distractor - Correct

Model Output:

So, let's try to figure out what the main object in the image is. ...
It's **a flat, disc-shaped creature with a tail**. Let's look at the options: ...
the creature in the image has a flat body, which is typical of rays. Rays are a type of fish, specifically cartilaginous fish (like sharks, but rays are different). **So the main object is a fish.** ...
To determine the category of the main object in the image, follow this step - by - step analysis: ...
2. **Analyze the creature's physical traits**: The object has a flat, disc - shaped body with a tail, which is characteristic of **rays**. Rays are a type of **fish** (specifically, cartilaginous fish, related to sharks).
...
Confirm the best fit: The creature's morphology and habitat align with the definition of a fish, making this the only category that accurately describes the main object. **<answer>Fish</answer>**

Response with one visual distractor - Incorrect

Model Output:

So, let's look at the image. The main objects are **a sea slug (or maybe a type of mollusk) and a tire.** ...
First, identify the main object. **The tire is a vehicle part, but the other object is a sea creature.** ...
Wait, maybe the tire is a vehicle part, but the question says main object. ...
So if the main object is the tire, then Vehicle. But is the tire the main object? **The image has both, but the tire is more prominent.** ...
To determine the correct category, we analyze the image and the given options:
1. **Identify the main objects**: The image contains a sea slug (a marine invertebrate) and a tire. ...
so the tire (as a vehicle component) is the most relevant object for the given options.
<answer>Vehicle</answer>

C.2 Attribute-level reasoning behavior

Example 1
Target attributes. instrument, ceramic object
Distractor attributes. fishing rod, fishing, fish-shaped ornament
Correct answer. Musical Instrument
Model output. *The image shows a hand holding a turquoise ceramic object with some holes and a design, and there's a fishing rod next to it. ... Wait, is the main object a fish? No ...But is it a real fish? No ... but maybe the question is considering the context of fishing. ... Instrument? Maybe ...*
<answer>Fish</answer>
Distractor. fishing rod (from Fish class)

Example 2
Target attributes. flower, insect
Distractor attributes. bone-shaped toy, dog bed, metal food bowl, tennis ball
Correct answer. Insect
Model output. *So, let's look at the image. Wait, maybe the main object is the dog-related items, ... Wait, the image has a flower, ... Insect is unrelated ... The items in the image are all dog-related, so the category that best describes the primary subject (the context of the objects) is Dog.*
<answer>Dog</answer>
Distractor. dog bone chew toy, dog bowl, tennis ball, kennel (from Dog class)

These examples illustrate how Qwen3-VL-Thinking extracts attributes and reasons on Idis samples with visual distractors. When distractor-related attributes outnumber target-object attributes, the model's reasoning shifts toward distractor context and produces an incorrect answer. In particular, when distractors occupy a larger image area than the target object, the model sometimes predicts the distractor's class as the final label, highlighting biased attribute allocation toward distractors.

C.3 Effect of prompt strategy

Prompt

[Question] Is the bird in the image a waterbird or a landbird? Use a thinking process to analyze the problem step-by-step.
At the end, provide your answer and clearly indicate it using <answer>X</answer> format.

Expected Answer

waterbird

(Bias-conflicting sample with land background)

Response without prompt strategy - Incorrect

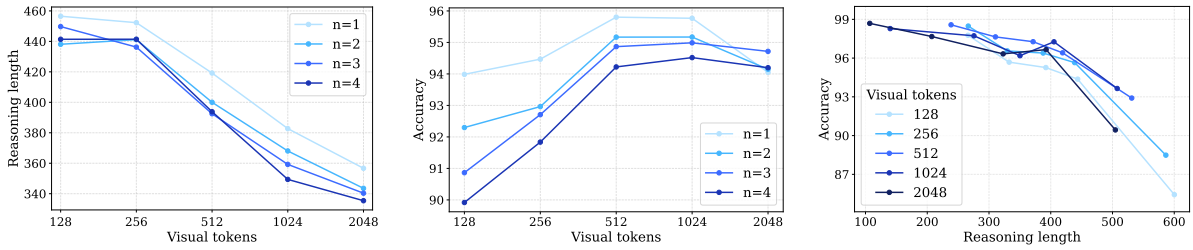
Model Output:

So, let's try to figure out if the bird in the image is a waterbird or a landbird. ...
the bird is flying near some yellow flowers, which are probably goldenrod (a common plant in meadows or fields). ...
the context of the habitat is important. Since the bird is in a field with flowers, not near water, that's a clue. ...
So maybe it's a landbird. ...
The bird looks like a Black-bellied Plover or maybe a similar shorebird, but wait, shorebirds are waterbirds. ...
Wait, the key is the habitat. If the bird is in a terrestrial environment (land), then it's a landbird. ...
Since the bird is in a terrestrial habitat with no evidence of aquatic adaptation or proximity to water, it is classified as a **landbird**.
<answer>landbird</answer>

Response with prompt strategy - Correct

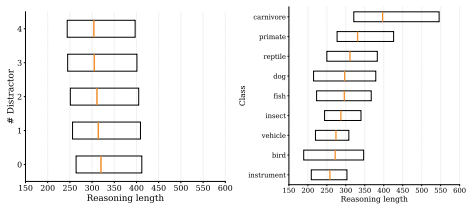
Model Output:

So, let's figure out if the bird is a waterbird or a landbird. ...
Looking at the image: the bird is flying near yellow flowers (maybe goldenrod). ...
Wait, maybe it's a tern? Terns are waterbirds. Wait, but the image shows it flying over land with flowers. Hmm. Wait, **maybe the key is the habitat.** ...
Wait, **the question says "based on the foreground bird's attributes".** ...
Wait, **let's check the bird's features.** The bird has a long, pointed beak, which is common in seabirds that catch fish. Also, the wings are long and narrow, which is typical for birds that fly over water. So maybe it's a waterbird.
Since the bird has the physical traits of a seabird (long wings, pointed beak for fishing), it's likely a waterbird.
So the answer should be waterbird. ...
Thus, **based on the bird's physical attributes (long wings, pointed beak)** and typical ecological role, it is a waterbird.
<answer>waterbird</answer>



(a) Visual tokens & Reasoning Length (b) Visual tokens & Accuracy (c) Reasoning length & Accuracy

Figure 9: **Impact of number of visual tokens.** (a) Increasing the number of visual tokens consistently shortens the model's reasoning traces, indicating that higher-resolution visual inputs reduce the need for long reasoning chains. (b) Accuracy generally improves as the number of visual tokens increases, eventually reaching a plateau as additional visual detail yields diminishing returns. (c) The degree of inverse-scaling with respect to reasoning length is similar across visual-token settings; however, the absolute reasoning lengths differ substantially, reflecting the effect of visual-token granularity on the model's reasoning process.



(a) Number of distractors (b) Target object class

Figure 10: **Average reasoning length distribution across distractor counts and image classes.** (a) shows that reasoning lengths remain relatively consistent across different numbers of distractors, whereas (b) reveals substantial variation in length distributions across nine classes of the Idis dataset.

D Beyond distractors: Exploring reasoning length determinants

In this section, we conduct further investigation into the factors that influence reasoning length, given that the presence of distractors did not yield significant changes in this metric.

We further analyze the distributions of reasoning lengths across three dimensions: image class, number of visual tokens, and sampling variability.

D.1 Impact of image class

We compare how reasoning length changes with the number of distractors to how it changes across object classes. We perform inference using Qwen3-VL-Thinking. As shown in Fig. 10, variations in reasoning length across object classes are substantially larger than those induced by additional distractors. This suggests that reasoning length is primarily governed by the intrinsic properties of the target object in the image, rather than by the presence of auxiliary visual elements.

D.2 Impact of number of visual tokens

The Idis dataset comprises images at a native resolution of 1024×1024 pixels. To analyze how varying the number of visual tokens affects model reasoning length, we resized images to 128×128, 256×256, 512×512, 1024×1024, and 2048×2048, corresponding to visual-token counts of 128, 256, 512, 1024, and 2048, respectively. We perform inference using Qwen3-VL-Thinking across all token counts.

Overall, we find that the number of visual tokens primarily determines a sample's position along a shared inverse-scaling curve: fewer visual tokens induce longer, lower-accuracy reasoning sequences, whereas moderate visual token counts shorten chains and improve accuracy, with diminishing returns at very high token counts.

In detail, as shown in Fig. 9a, reasoning length decreases consistently as the number of visual tokens increases, indicating a strong inverse relationship. Larger token budgets provide richer visual information, reducing the need for extended reasoning and mitigating overthinking.

Fig. 9b reveals a parallel trend in accuracy. Performance is lowest at 128–256 tokens, peaks around 512–1024 tokens, and slightly declines at 2048 tokens. When combined with the length–accuracy curves in Fig. 9c, the underlying mechanism becomes clear: across all visual-token settings, accuracy monotonically decreases as reasoning length increases, and the curves for different token counts substantially overlap. This alignment suggests that the inverse-scaling relationship holds regardless of token count.

Consequently, the accuracy patterns in Fig. 9b

are largely mediated by where each sample lands on the shared length–accuracy curve. Low token counts push samples into a long-chain, low-accuracy region, while moderate token counts move them into shorter, higher-accuracy regimes. The slight degradation at 2048 tokens indicates diminishing returns, where additional visual tokens no longer meaningfully improve accuracy despite further shortening the reasoning chains.

These trends likely reflect a simple balance. With too few visual tokens, the model cannot fully perceive the scene, leading to overthinking with unnecessarily long reasoning. With too many tokens, the model shortens its chains, suggesting a diminishing effect on reasoning length as visual information becomes increasingly abundant. Thus, the number of visual tokens naturally modulates how much the model needs to reason.

D.3 Impact of sampling

To quantify the variability inherent in autoregressive generation, we sampled each image in the Idis dataset 64 times using the Qwen3-VL-Thinking, with standard settings of temperature 0.7 and Top-p (nucleus sampling) 0.95.

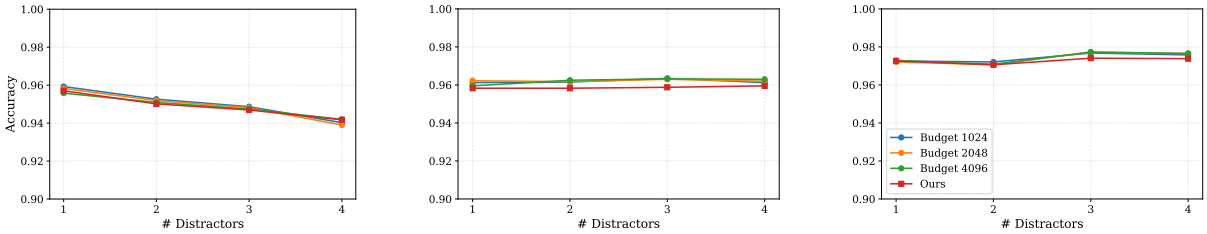
Across the four distractor conditions, the average response lengths were $\{397.4, 378.5, 370.7, 361.0\}$ tokens for 1 through 4 distractors, respectively. At the single-image level, this corresponds to an average per-step slope of 11.8 tokens per additional distractor and an average reduction of 36.6 tokens when increasing the number of distractors. Overall, these extents indicate that the impact of the distractor count on reasoning length is relatively slight.

However, within-image sampling variability is substantial: the mean sampling standard deviation is 122.4 tokens. Consequently, sampling noise overwhelms the distractor effect at the single-image level, where stochastic decoding noise (SD $\approx$ 122 tokens) is roughly $10\times$ larger than the per-step effect and $3\times$ larger than the full reduction when increasing from 1 to 4 distractors. Concretely, if we randomly sample one response from a 1-distractor image and one from a 4-distractor image, the first response is longer only 40% of the time—barely better than chance.

While sampling noise dominates at the single-image level, its impact diminishes when aggregating over many images. When averaging across 4,050 images per condition, the standard error drops to 3.4 tokens (95% CI: ± 6.7), yielding an overall effect of $\Delta_{4-1} = -36.3 \pm 9.5$ tokens. Thus, the random variability that obscures distractor effects in individual images largely cancels out in aggregate, allowing prior analyses to reliably measure average trends across the whole dataset.

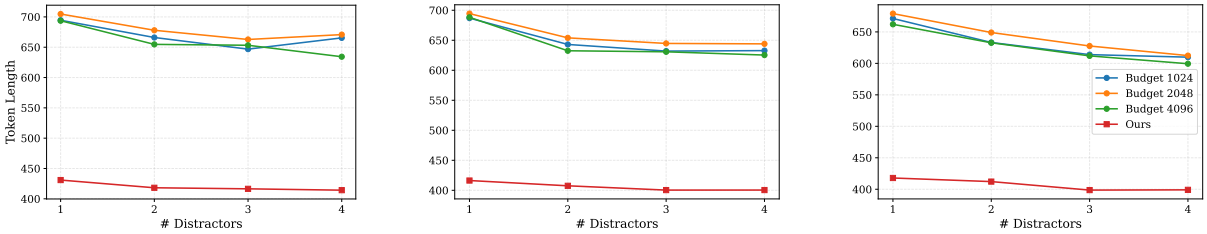
E Controlled reasoning budgets

We additionally consider a controlled overthinking setting where we explicitly cap the thinking length via prompting, in contrast to the natural overthinking setting used in our main experiments. Concretely, we prepend an instruction that fixes the maximum number of reasoning tokens to 1024, 2048, or 4096 and evaluate the model on the Idis dataset. As shown in Fig. 11 and Fig. 12, varying this budget barely changes either accuracy or reasoning length across all distractor counts and semantic types, and the three budgeted variants almost overlap in both metrics. Consistent with our main results, Fig. 11 also shows that accuracy drops the most under conflicting distractors. These results suggest that, unlike in reasoning LMs, the test-time behavior of reasoning VLMs is largely insensitive to such prompt-level budget control, so we conduct all main analyses under the natural overthinking setting.



(a) Conflicting (b) Irrelevant (c) Aligned

Figure 11: **Reasoning budgets do not meaningfully affect accuracy in reasoning VLMs.** Across all distractor counts (from 1 to 4), both the controlled overthinking setting that adjusts the thinking budget via prompting (1024, 2048, 4096 tokens) and the natural overthinking setting ("Ours") yield nearly identical performance. This contrasts with reasoning language models, where longer budgets typically alter the scaling curve. Accuracy decreases most noticeably only in the conflicting distractor condition, whereas aligned and irrelevant distractors maintain stable accuracy regardless of distractor count, demonstrating that semantic conflict—not budget size—drives performance degradation.



(a) Conflicting (b) Irrelevant (c) Aligned

Figure 12: **Reasoning budgets do not meaningfully affect reasoning length in reasoning VLMs.** Across all distractor conditions—i.e., conflicting, irrelevant, and aligned—the controlled overthinking settings with prompting-based budgets (1024, 2048, 4096 tokens) produce nearly identical reasoning lengths, mirroring the stability observed in accuracy. However, these controlled settings consistently generate substantially longer reasoning traces than the natural overthinking setting ("Ours").

	Qwen3-VL-Thinking	Intern-S1-mini	GLM-4.1V-Thinking	R1-OneVision
Size	Acc.	Acc.	Acc.	Acc.
Small	96.1	94.4	96.3	92.86
Medium	95.5	93.1	95.6	91.68
Large	94.0	90.9	94.7	88.25

Table 9: **Accuracy results on the Idis-manual dataset.** Accuracy across three distractor-size conditions (Small, Medium, Large) for four reasoning VLMs.

F Additional experimental results

F.1 Full quantitative results on Idis dataset

In this section, we provide additional quantitative results on the Idis dataset across all four reasoning VLMs (Qwen3-VL-Thinking, Intern-S1-mini, GLM-4.1V-Thinking, and R1-OneVision). Fig. 13 shows that, across all models, conflicting distractors cause the largest declines with downward shifts, indicating that reasoning VLMs are most vulnerable to distractors that semantically conflict with the target object. As illustrated in Fig. 14, typographic distractors rendered as images exhibit a similar trend to visual distractors across all four models, shifting the length–accuracy curve downward with only a limited increase in reasoning length. As shown in Fig. 15, textual distractors inserted into the question prompt reduce accuracy across all four models by inducing longer reasoning traces. Fig. 16 shows the relationship between reasoning length and the number of generated visual attributes. All models exhibit a strong linear positive correlation: longer traces systematically produce more attributes, and this trend holds regardless of the number of distractors. Fig. 17 presents that increasing the number of distractors does not substantially change the total number of attributes, but consistently increases the fraction of distractor-related attributes. Finally, Fig. 18 illustrates that the distractor-related attribute ratio is negatively correlated with the accuracy. As the fraction of attributes assigned to distractors increases, accuracy monotonically decreases, and when distractor attributes dominate, accuracy effectively collapses. Taken together, these full quantitative results support our main finding that performance degradation on Idis is driven by how attributes are allocated to visual distractors rather than the target object, not by an overall expansion of the reasoning length.

F.2 Results on Idis-manual dataset

While Gemini 2.5 Flash Image enables flexible image editing, it does not allow precise control over the size of generated distractors. To enable fine-grained manipulation of distractor size, we construct Idis-manual by directly over-laying background-masked segments of distracting objects onto the original images. Specifically, we use Language Segment Anything (LangSAM) (Medeiros, 2025) to extract image segments from the textual description of each distractor. The extracted segments are then resized according to predefined small, medium, or large configurations and overlaid onto the target image.

As shown in Fig. 19, increasing the distractor size from Small to Large consistently raises the distractor-related attribute ratio for all four reasoning VLMs, indicating that larger distractors capture more of the model's attention and receive a greater portion of the generated attributes. Table 9 further shows that this shift in attribute allocation is accompanied by a clear drop in accuracy as distractor size grows. Taken together, these results suggest a direct link between distractor size and performance degradation, where larger distractors lead reasoning VLMs to produce more distractor-related attributes, which in turn yields lower accuracy.

Model	Aligned		Conflicting		Overall	
	Acc.	Len.	Acc.	Len.	Acc.	Len.
Qwen3-VL-Thinking	93.6	635.9	76.4	829.9	88.3	695.8
+ Prompt Strategy	94.1	474.7	78.2	652.1	89.2	529.5
Intern-S1-mini	93.2	493.2	56.7	672.5	81.9	548.6
+ Prompt Strategy	92.4	574.2	58.7	711.9	82.0	616.7
GLM-4.1V	94.4	340.9	80.7	498.2	90.1	389.5
+ Prompt Strategy	92.9	253.9	84.7	358.7	90.4	286.3
R1-OneVision	93.9	300.5	63.0	310.5	84.4	303.6
+ Prompt Strategy	93.9	233.0	66.3	244.7	85.4	236.6

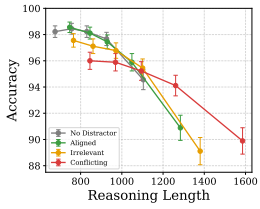
Table 10: **Table results of four vision-language models on the Waterbirds dataset.** We report accuracy and average reasoning length for aligned, conflicting, and overall conditions across non-reasoning VLMs, reasoning VLMs, and prompt-strategy settings.

F.3 Detailed results for debiasing experiments

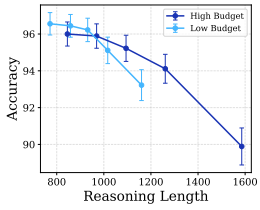
Table 10 presents the full results of our debiasing experiments on the Waterbirds dataset, extending the summary trends shown in Fig. 8b. All four reasoning VLMs with the prompt strategy show better performance on the conflicting. These detailed results confirm that our debiasing prompt can mitigate spurious-correlation failures on Waterbirds by steering the reasoning VLMs to reason primarily based on attributes of the target object rather than spurious cues.

G Experiments on proprietary model

To examine whether test-time inverse scaling generalizes beyond open-weight models, we conduct additional experiments on Gemini 3 Flash, a proprietary SOTA model, using the Idis dataset.



We observe similar inverse scaling trends, with accuracy degrading as reasoning length increases. Notably, the degree of performance degradation varies depending on distractor semantics, consistent with our findings on open-weight models.



Furthermore, by varying Gemini 3 Flash's native thinking budget, we find that the "High budget" setting, which encourages longer reasoning, consistently yields lower accuracy than the "Low budget" setup across all reasoning lengths.

H Artifact Licenses and Intended Use

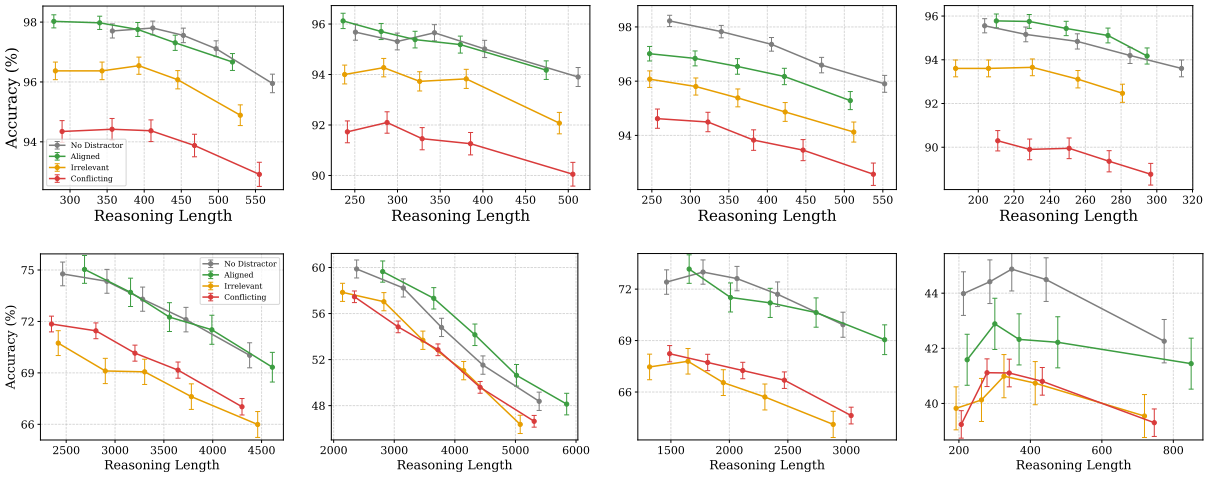
In this section, we provide the licenses for all the pretrained models and datasets utilized in our study. Our use of all artifacts strictly adheres to their original licenses and is entirely consistent with their intended use for academic research and evaluation.

H.1 Pretrained Models

- **Qwen3-VL-8B-Thinking.** Apache License Version 2.0.
- **GLM-4.1V-9B-Thinking.** MIT.
- **Intern-S1-mini.** Apache License Version 2.0.
- **R1-OneVision-7B-RL.** Apache License Version 2.0.
- **Gemini 2.5 Flash Image.** Google Gemini API Additional Terms of Service and Google APIs Terms of Service.
- **Sonnet 4.5.** Anthropic Commercial Terms of Service.
- **GPT-OSS-120B.** Apache License Version 2.0.

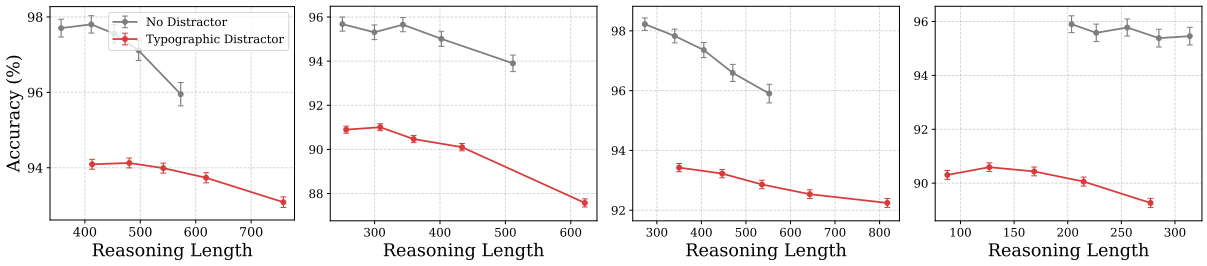
H.2 Datasets and Benchmarks

- **ImageNet-9.** ImageNet Terms of Access.
- **MathVerse.** MIT.
- **LogicVista.** Apache License Version 2.0.
- **Waterbirds.** MIT.



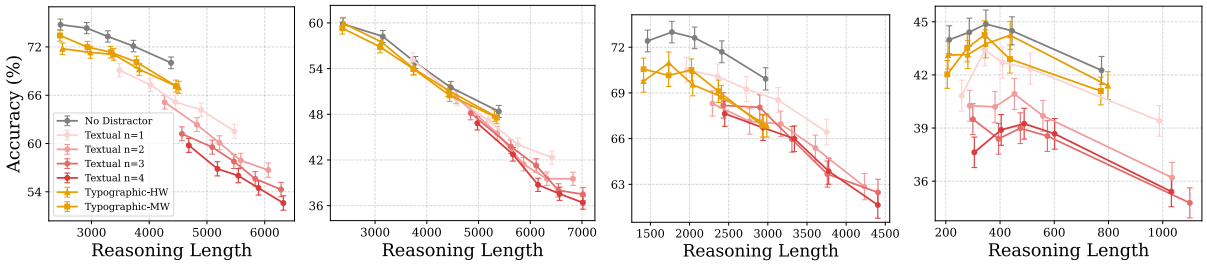
(a) Qwen3-VL-Thinking (b) Intern-S1-mini (c) GLM-4.1V-Thinking (d) R1-OneVision

Figure 13: **Scaling behavior in reasoning VLMs, with various types of semantic relationship of visual distractors to the target object.**



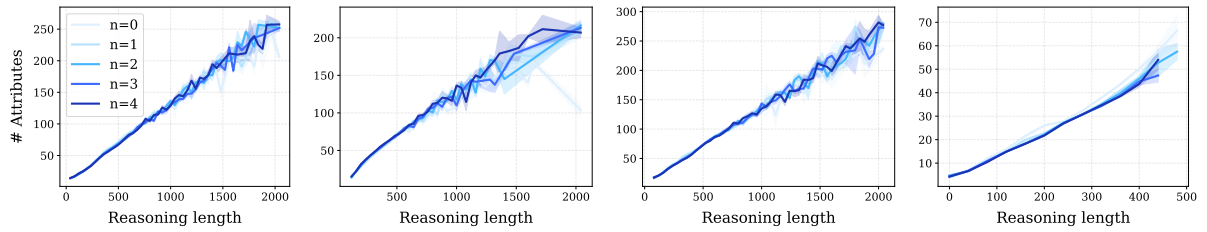
(a) Qwen3-VL-Thinking (b) Intern-S1-mini (c) GLM-4.1V-Thinking (d) R1-OneVision

Figure 14: **The effect of typographic distractors on Idis-perception.**



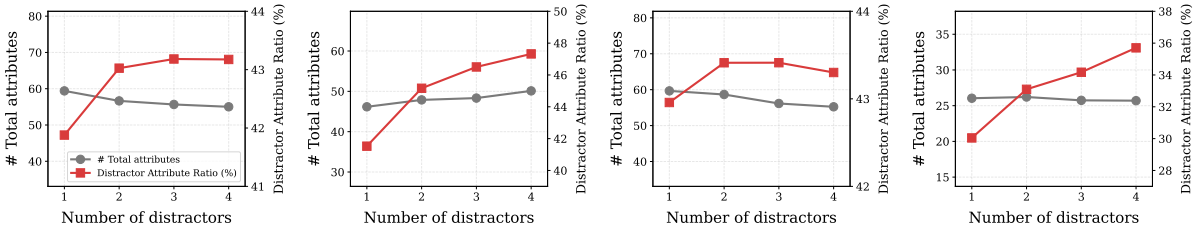
(a) Qwen3-VL-Thinking (b) Intern-S1-mini (c) GLM-4.1V-Thinking (d) R1-OneVision

Figure 15: **The effect of typographic and textual distractors on Idis-math.**



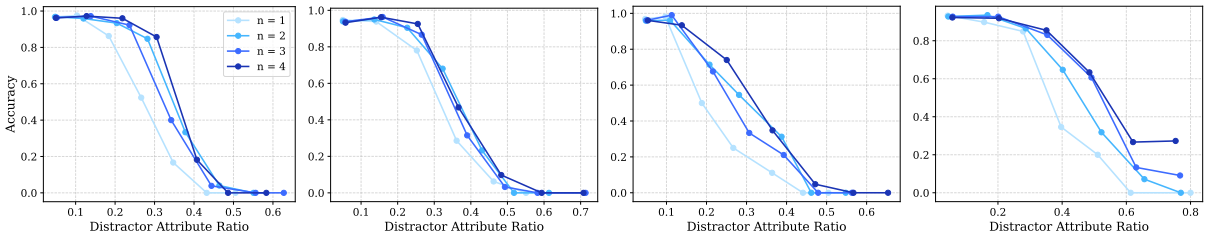
(a) Qwen3-VL-Thinking (b) Intern-S1-mini (c) GLM-4.1V-Thinking (d) R1-OneVision

Figure 16: **A strong linear correlation between reasoning length and the number of attributes.**



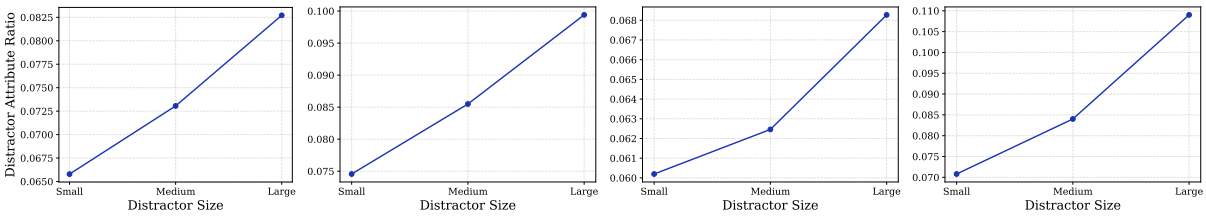
(a) Qwen3-VL-Thinking (b) Intern-S1-mini (c) GLM-4.1V-Thinking (d) R1-OneVision

Figure 17: **The effect of typographic and textual distractors on Idis-math.**



(a) Qwen3-VL-Thinking (b) Intern-S1-mini (c) GLM-4.1V-Thinking (d) R1-OneVision

Figure 18: **The distractor attribute ratio is negatively correlated with the accuracy.**



(a) Qwen3-VL-Thinking (b) Intern-S1-mini (c) GLM-4.1V-Thinking (d) R1-OneVision

Figure 19: **Larger distractors lead to higher distractor-attribute ratios.** Across all four reasoning VLMs, the proportion of distractor-related attributes increases as distractor size grows from small to medium to large. This indicates that larger distractors capture more of the model's attention and contribute more heavily to the attribute composition.