Understanding the Effects of Visual 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 studies on language models have reported an inverse scaling effect, where textual distractors lead to longer but less effective reasoning. To 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, numerical, and spatial dimensions. Our analyses reveal that visual distractors behave fundamentally differently from textual ones: although inverse scaling still occurs, adding visual distractors decreases accuracy without increasing reasoning length. We further show that tracking attribute counts within reasoning traces offers key insights into the interaction between distractors, reasoning length, and accuracy. Finally, we propose a prompting strategy to mitigate distractor-driven predictions in reasoning models, and extend it to an established visual bias benchmark, namely Waterbirds.

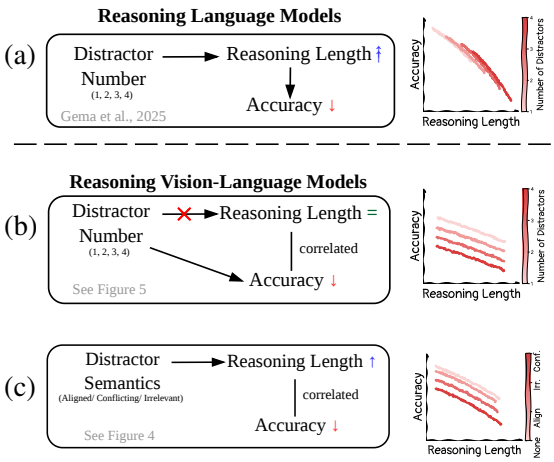


Figure 1: **Inverse scaling in reasoning LMs vs. VLMs.** (a) In reasoning LMs, adding more textual distractors amplifies the test-time inverse scaling effects. (b) In reasoning VLMs, adding visual distractors decreases the accuracy but does not increase the reasoning length. Instead, the entire length-accuracy curve is shifted downward. (c) The semantics of visual distractors (e.g., aligned, irrelevant, conflicting) primarily affects the degree of accuracy drop, with the most severe degradation occurring when distractors are negatively spuriously correlated with the target object.

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, 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 language-only models (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

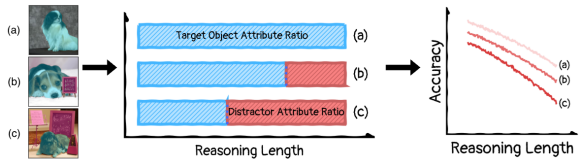


Figure 2: **Larger distractor areas increase distractor-related attributes and lead to performance degradation.** As the relative spatial scale of distractors to target objects grows (from (a) being smallest to (c) being largest), the proportion of distractor-related attributes within the reasoning trace increases. Conversely, the total attributes count remains roughly constant. This leads to a downward shift of the inverse scaling curve.

information in the context (i.e., distractors) consistently induces the inverse scaling. 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 ***visual distractors*** induce analogous failure modes in reasoning VLMs. Our motivation is twofold. First, visual inputs inherently contain substantial irrelevant information, e.g., background objects or contextual noise. Second, visual and textual modalities often contribute unevenly to VLM predictions, with textual inputs frequently dominating visual cues (Deng et al., 2025).

To systematically examine the effect of visual distractors on inverse scaling of reasoning VLMs, we construct Idis (Images with distractors), a new VQA benchmark for controlled analysis. Idis comprises over 50,000 natural and synthetic images featuring target objects alongside one or more distractors. Each image is derived from a clean, background-free base image and edited using Gemini 2.5 Flash Image (Google, 2025) to insert distractors with systematically varied semantic relevance, quantity, and spatial scale. Because distractors are added to the same base image without modifying the target object, our dataset effectively forms a controlled intervention setup, enabling causal analysis of distractor effects.

Using Idis, we first confirm that reasoning VLMs exhibit test-time inverse scaling. However, we observe two key deviations from patterns reported for reasoning LMs with textual distractors: (i) visual distractors degrade accuracy without increasing reasoning length (Fig. 1b) and (ii) the semantic relationship between the distractor and the target object significantly affects performance (Fig. 1c).

To investigate the mechanism behind the accuracy drop in finding (i), we perform an attribute-level analysis of the reasoning traces. Our analysis suggests a potential limitation of current reasoning VLMs: they may fail to assess whether the visual attributes they extract are actually relevant to the target object. As distractors increase in ***spatial scale*** (i.e., in size and number), reasoning VLMs incorporate more distractor-related attributes into their reasoning traces (Fig. 2). As a result, predictions become dominated by ***attribute counts*** rather than their semantic relevance to the target.

Building on this hypothesis, we propose a simple prompt-based strategy that encourages reasoning VLMs to focus on attributes of the target object, thereby reducing their reliance on spurious distractor cues. We show that this strategy improves robustness to distractors and that both the identified limitation and the effectiveness of the approach generalize to the Waterbirds dataset (Sagawa et al., 2020), where distractors arise from background context rather than localized objects.

Our key contributions are as follows:

- We introduce Idis, the first dataset designed to systematically study how visual distractors affect inverse scaling trends in reasoning VLMs.
- We identify a distinct inverse scaling pattern specific to vision-language models and provide a hypothesis for its underlying mechanisms through in-depth attribute-level analysis.
- Motivated by this hypothesis, we propose a prompting strategy that mitigates the negative effects of distractors in reasoning VLMs.

2 Related work

Overthinking and inverse scaling. Recent work shows that reasoning models often overthink, generating excessive tokens for only marginal performance gains (Sui et al., 2025; Chen et al., 2025; Yeo et al., 2025; Luo et al., 2025). More recently, inverse scaling—where accuracy decreases as more tokens are generated—has also been observed. Cuadron et al. (2025) report such effects in interactive settings (e.g., agentic tasks), with similar trends confirmed 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 consistently arises when distractors—such as irrelevant text or code—are introduced. We extend this line of work

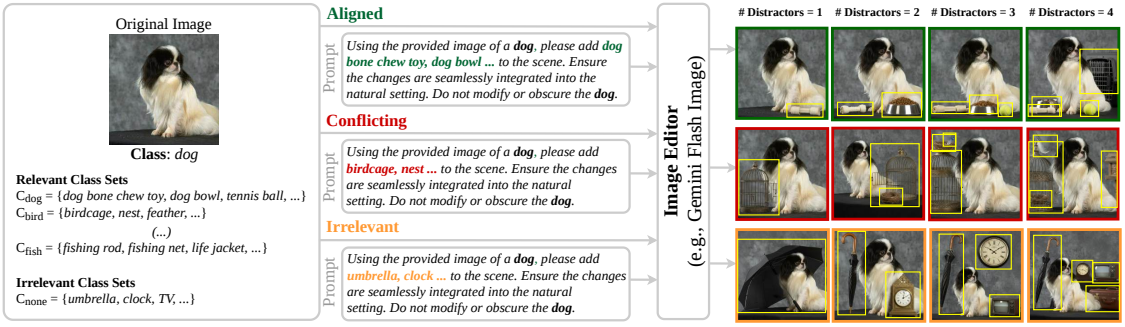


Figure 3: **Dataset generation pipeline for Idis.** We generate images with distractors by editing an image with object-free background. In particular, we prompt Gemini 2.5 Flash Image with instructions to add distractor objects selected from the set of aligned, conflicting, or irrelevant distractors. Here, each set is defined by a set of keywords, which are either correlated with the target object class (aligned), correlated with other classes (conflicting), or not correlated (irrelevant). This pipeline can generate multiple images with various choices of visual distractors, while keeping the target object consistent. Yellow boxes indicate distractor regions (not included in original images).

to reasoning VLMs by introducing the notion of visual distractors for VQA and studying their impact on test-time scaling.

Distractors in vision-language models. Several prior works have examined how distractors affect the predictive accuracy of reasoning VLMs, showing that various forms of distraction—such as GUI pop-up boxes (Ma et al., 2025), textual overlays (Deng et al., 2025), and entirely unrelated images (Cai et al., 2025)—can substantially degrade performance. Our work differs in three key ways. First, we study how distractors affect scaling behavior, rather than accuracy alone. Second, we focus on distracting objects within the original image, rather than inserting unrelated images. Third, we systematically control multiple dimensions of distractor severity—semantic, numeric, and spatial—to enable a more precise analysis of failure modes.

Spurious correlation and visual biases. We control distractor relevance by selecting objects with varying degrees of spurious correlation (or *bias*), i.e., correlations that appear meaningful but are not (Ribeiro et al., 2016; Sagawa et al., 2020). Visual models are known to rely heavily on such spurious cues when trained on biased data (Sagawa et al., 2020; Nam et al., 2020). Although foundation models are generally more robust (Goyal et al., 2022), recent studies show that vision-language models still exhibit biases (e.g., gender and racial) (Fraser and Kiritchenko, 2024; Xiao et al., 2025; Raj et al., 2024; Ruggeri et al., 2023; Janghorbani and De Melo, 2023). Our paper extends this line of work by examining how such biases affect test-time scaling in vision-language models.

3 Idis: Images with distractors

To analyze how visual distractors affect test-time scaling of reasoning VLMs, we introduce ***Idis*** (Images with distractors), a visual question-answering (VQA) dataset where the task is to classify the target object appearing in an image. We adopt this minimal task design because reasoning models tend to overthink simpler tasks more (Chen et al., 2025), thus providing an ideal setup to study the failure modes of reasoning models.

3.1 Data generation pipeline

Idis is constructed by editing images of target objects to introduce controlled visual distractors. Given a base image, we use a text-conditioned image editing model to synthesize new images with additional objects, based on the given prompt that describes which distractors to add (Figure 3).

Crucially, multiple distractor-conditioned samples are generated by intervening on the same base image, where the target object remains unchanged and only distractor objects are added. This controlled intervention allows us to isolate the causal effect of distractors on model predictions.

Base dataset. We use the "original" split of the ImageNet-9 dataset (Xiao et al., 2020), which contains 4,050 curated images from ImageNet (Deng et al., 2009). The dataset provides images with a single salient foreground object and no background clutter, allowing precise control over the number of distracting objects in the image.

Image editing. Naïvely compositing distractors into the base image, e.g., via direct overlaying,

leads VLMs to detect artifacts rather than reasoning about content. They produce reasoning traces such as *"the image might be a composite or edited photo"*. To ensure realistic edits while preserving the consistency of the target object with the original image, we use Gemini 2.5 Flash Image (a.k.a., "nano banana") (Google, 2025) for generative editing. To insert k distractors into an image, we select k distinct distractor classes from aligned, conflicting, or irrelevant categories (see Section 3.2). Then, we prompt the image generation model to add selected distractor classes to the image.

Idis-manual. 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 overlaying 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.

Other details. We provide omitted details, including the exact generation process, dataset statistics, and the prompts used, in Appendix A.

3.2 Distractor design across three dimensions

Semantic relationship. We consider three semantic categories of distracting objects, corresponding to the type of spurious correlation between the distractor and the target object. In particular, we adopt the following categorization of correlated features from the debiasing literature (Nam et al., 2020):

- Aligned: The distractor is positively spuriously correlated with the target class. (e.g., a bird cage is an aligned distractor to the class "bird.")
- Conflicting: The distractor is negatively spuriously correlated with the target class, by being positively correlated with other classes. (e.g., a bird cage may be a conflicting distractor to the class "vehicle.")
- Irrelevant: The distractor is not strongly spuriously correlated with any target class. (e.g., a TV may not be strongly correlated with any classes in the base dataset.)

The categories can be characterized by specifying two sets. First, we define the *relevant class set* $\mathcal{C}_y$ as a set of classes that are positively correlated with the target class y. We select four relevant classes for each y, where each relevant class is characterized by a keyword or a phrase. For example, the relevant class for the target class "bird" is $\mathcal{C}_{\text{bird}} = \{\text{bird cage}, \text{nest}, \text{feather}, \text{bird feeder}\}$. Second, we define the *irrelevant class set* $\mathcal{C}_{\text{none}}$ as a set of classes that are not relevant for any target class. In particular, we use $\mathcal{C}_{\text{none}} = \{\text{umbrella}, \text{clock}, \text{TV}, \text{suitcase}\}$. Given these sets, an aligned distractor for a sample of class y is simply $\mathcal{C}_y$, and the conflicting distractor will be $\mathcal{C}_{y'}$ for any $y \neq y'$. The irrelevant distractor will be the elements of $\mathcal{C}_{\text{none}}$.

The semantic relationship is most positive for aligned distractors and most negative for conflicting distractors. Consequently, it is reasonable to expect that aligned distractors may reinforce the model's correct decision, whereas conflicting distractors may bias the model toward the class positively associated with those distractors. Irrelevant distractors let us assess the robustness of the reasoning process to semantically unrelated information.

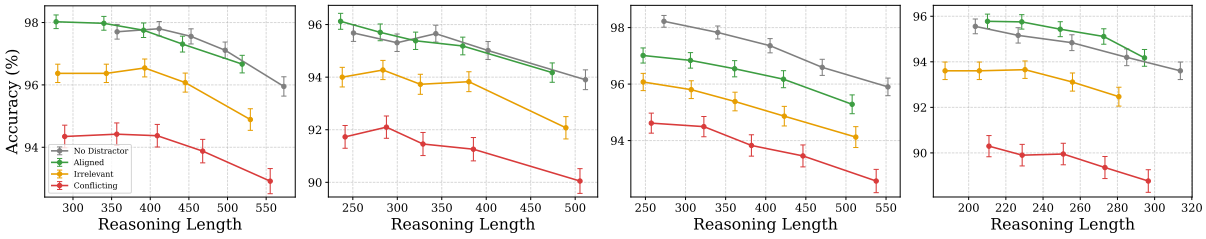
Number. We add between one and four distractors to each image. When multiple distractors are included, they are drawn from the same semantic category (e.g., aligned) but differ in their fine-grained classes. For instance, when adding two aligned distractors to an image containing a dog, we might include a dog bowl and a wooden kennel.

Size. In the Idis-manual dataset, we vary each distractor's size across three levels: small, medium, and large. The small setting corresponds to a distractor whose width is 25% of the image width, while the medium and large settings correspond to 35% and 45%, respectively.

4 Experimental setup

Similar to Gema et al. (2025), we focus on the *sequential scaling*. That is, models scale by generating a longer reasoning trace, rather than utilizing parallel reasoning traces. For analysis, we sample five responses for each question, rank them by reasoning length, and compute the accuracy of each rank over all questions.

Models. We evaluate four open-weight frontier reasoning VLMs at the model scales of 7–9B parameters, with various 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). Here-



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

Figure 4: **Inverse scaling in reasoning VLMs, with various types of semantic relationship of visual distractors to the target object.** We examine the inverse scaling trend across four reasoning VLMs, comparing the no distractor baseline with the cases of inserting aligned, irrelevant, or conflicting distractor (four distractors each). 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.

after, we refer to these models without specifying their parameters.

Reasoning budget. Unlike reasoning language models (Gema et al., 2025), 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.

Metrics. Following Gema et al. (2025), we primarily focus on the interplay between three elements: (1) the accuracy on the target task; (2) the reasoning length; (3) the number of distractors added to the image. Stepping further, we also measure and utilize the following metrics for analysis:

- Area of each object: We measure the area of the target and distracting objects appearing in the image, and how they affect the accuracy and the reasoning length. Precisely, we use the LangSAM to generate the mask for the pixels corresponding to the textual description of each object (Medeiros, 2025), and count the number of pixels.
- Proportion of key attributes: We parse the visual attributes that are mentioned in the reasoning trace, and analyze their relationship to each target or distractor class. To parse the reasoning trace, we provide the trace and structured instructions to the DeepSeek-V3.2-Exp (DeepSeek-AI, 2025).

5 Results

In this section, we provide the following results:

- An empirical observation on how visual distractors affect the inverse scaling trends vary with distractor semantics and number (Section 5.1)
- An attribute-level analysis showing that reasoning VLMs are overly guided by attribute counts, causing distractors with larger spatial scale to dominate the reasoning trace (Section 5.2)
- A formal hypothesis that reasoning VLMs lack a mechanism to reliably prioritize target-relevant attributes over distractor attributes (Section 5.3)

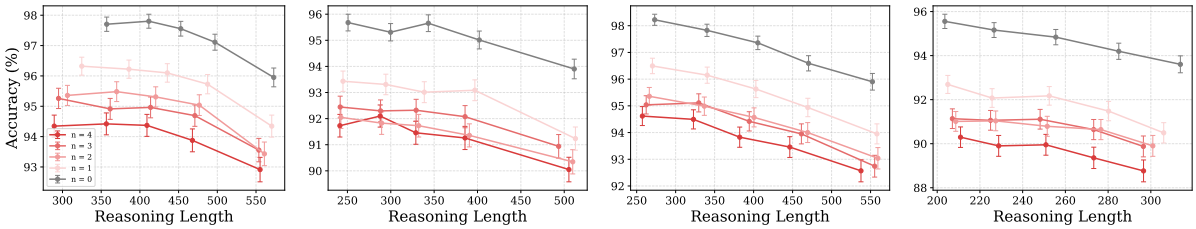
5.1 Observation: Scaling of reasoning VLMs

We now demonstrate that the inverse scaling phenomenon indeed takes place on reasoning VLMs predicting on the Idis dataset. We first put particular focus on its interplay with the semantic properties of the distractors. Then, we highlight the difference with language-only reasoning models (Gema et al., 2025) in how the number of distractors affects the test-time scaling trends.

Inverse scaling and distractor semantics. In Fig. 4, we visualize the relationship between the reasoning length and the accuracy of four different VLMs, under various choices of semantic relationship between the distracting objects and the target object. In particular, we compare the cases of: (1) no distractors, (2) aligned distractors, (3) irrelevant distractors, and (4) conflicting distractors. For (2,3,4), we add four distracting objects that are all drawn from the same semantic category.

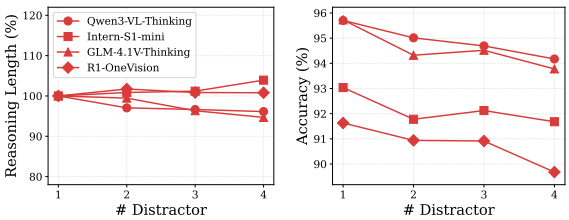
In the plot, we first observe that the inverse scaling indeed occurs in all setups considered. That is, a longer reasoning is associated with lower accuracy. Another observation is that distractor semantics have a substantial effect on the scaling curves, shifting them along the y-axis. The downward 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.

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



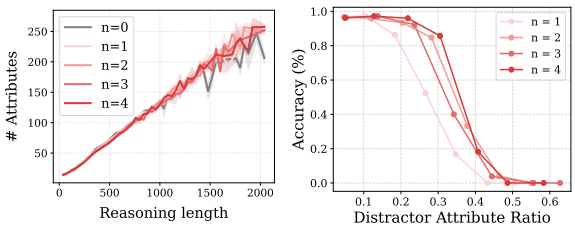
(a) Qwen3-VL-Thinking (b) Intern-S1-mini (c) GLM-4.1V-Thinking (d) R1-OneVision

Figure 5: **Effect of distractor number on inverse scaling in reasoning VLMs.** Increasing the number of visual distractors (from zero to four) in reasoning VLMs does not extend the reasoning length but instead shifts the length–accuracy curve downward, in contrast to reasoning LMs. This suggests that visual distractors primarily reduce accuracy without altering the reasoning process itself.



(a) Reasoning Length (b) Accuracy

Figure 6: **Adding more distractors drops accuracy without extending reasoning length.** (a) shows that the distractor count does not have any significant or consistent effect on the reasoning length over various models. In contrast, (b) shows that accuracy drops as we add more distractors, consistently over all models.



(a) Reasoning length (b) Accuracy

Figure 7: **Attribute statistics strongly predict reasoning length and accuracy.** (a) Attribute count is linearly correlated with reasoning length, largely independent of the number of distractors. (b) Accuracy drops sharply as the distractor attribute ratio increases. Results are shown for Qwen3-VL-Thinking (See Appendix E).

Effect of the number of distractors. Next, we study how varying the number of distractors affects the scaling trend. In reasoning LMs (Gema et al., 2025), adding more textual distractors intensifies test-time inverse scaling. In contrast, Fig. 5 shows that additional visual distractors consistently shift the length–accuracy curve downward without substantially changing the range of reasoning lengths.

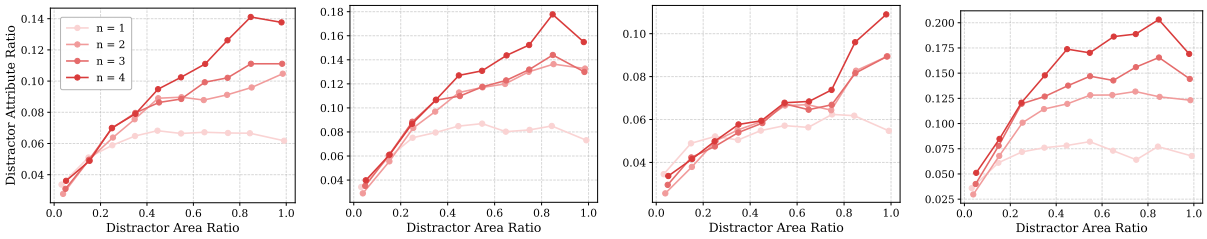
This observation is further supported by Fig. 6a. We observe that as we increase the number of distractors, the reasoning length remains similar, with less than 10% change even with four distractors. On the other hand, as can be seen in Fig. 6b, the accuracy consistently decreases as we increase the number of distractors. This trend is similar to what has been observed for the reasoning LMs.

> ***Takeaway#1***: Visual distractors intensify the test-time inverse scaling of reasoning VLMs with a distinct pattern, where adding more distractors lowers accuracy while keeping reasoning length unchanged.

5.2 Analyses: Attributes in reasoning traces

To understand why Takeaway#1 only occurs for visual distractors, we conduct an ***attribute-level analysis*** of the reasoning trace, motivated by prior works that analyze reasoning traces to better understand model behavior (Deng et al., 2025; Shojaee et al., 2025; Qian et al., 2025). Specifically, we utilize a language model to parse the visual attributes that appear in the reasoning trace, defined as literal words or phrases that describe observable properties, objects, or class-related features. We then track the associated object in the image (see Section 4 for more details).

The attribute statistics also appear empirically important. Fig. 7 shows that the quantity is strongly correlated with both reasoning length and accuracy. In particular, Fig. 7b shows that the prediction accuracy is almost determined solely by the fraction of attributes that are about the distractor. In particular, the accuracy drops to near zero whenever the distractor attribute ratio exceeds 50%, yet remains above 97% whenever the ratio is below 20%.



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

Figure 8: **A larger distractor-to-target area ratio leads to the generation of more distractor attributes.** Across all four reasoning VLMs, the fraction of attributes related to the distractors tends to increase whenever the spatial area of the distracting objects (with respect to the area of target object) increases, up to some saturation point. Interestingly, the number of distractors seems to play a certain role in determining the saturation point of the increase. In particular, adding more distractors leads to a higher saturation point, for all four models.

> ***Takeaway#2***: Attribute statistics strongly predict both reasoning length and accuracy, presumably due to a lack of a mechanism that can discern meaningful attributes from distracting ones.

Given the Takeaway#2, one may ask whether reasoning VLMs have the ability to focus on extracting meaningful attributes from the image. If this is the case, VLMs may still be able to make accurate predictions by textualizing only the highly relevant information to the reasoning trace.

Unfortunately, our experiments suggest that this may not be the case. Across models, the distractor attribute ratio is strongly shaped by distractor area (Fig. 8). In particular, whenever the ratio of the distractor pixels (relative to the target object pixels) increases, the total number of distractor attributes increases as well, up to some saturation point. Interestingly, the saturation point seems to increase as we increase the number of distractors.

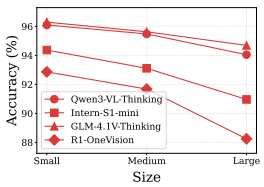


Figure 9: Accuracy drops with distractor size on Idis-manual.

Our experiments on the Idis-manual dataset support this claim as well. As shown in Fig. 9, the model accuracy decreases quite severely whenever we increase the size of the distracting objects, from small to large. While the textual description and semantics of the distracting objects may be essentially the same, the varying size of distracting objects may play a crucial role in making the final prediction. We also provide an attribute analysis for the Idis-manual dataset in Appendix E. Thus, we conclude as follows:

> ***Takeaway#3***: The spatial area of distracting objects is a driving factor that determines the ratio of distractor attributes in the reasoning trace, which is presumably due to a lack of an attribute extraction mechanism that can effectively discern meaningful visual features from distracting ones.

5.3 Hypothesis

Recall that in Section 5.2, we have made two central claims to explain the vulnerability of reasoning VLMs against visual distractors. Precisely, we argue that this reasoning failure stems from two limitations: (i) the lack of a mechanism to disregard the attributes corresponding to the distracting objects, when making the final conclusion (Takeaway#2), and (ii) the lack of a mechanism to prevent the extraction of visual attributes from distracting objects in the first place (Takeaway#3).

Our hypothesis is as follows: reasoning VLMs are guided by the attribute counts in the reasoning trace, and the spatial scale (numbers and size) of each object plays a dominant role in determining the number of corresponding attributes.

In other words, we hypothesize that reasoning VLMs implicitly associate visual salience with task relevance, lacking higher-level conceptual grounding. When more or larger distractors are present, the model treats them as additional evidence and allocates more reasoning capacity to describing these regions. This is consistent with our observation that accuracy drops as the spatial scale of distractors increases (Figs. 5 and 9). Our findings highlight the need for mechanisms that improve attribute extraction and encourage reasoning VLMs to focus on attributes genuinely relevant to the target.

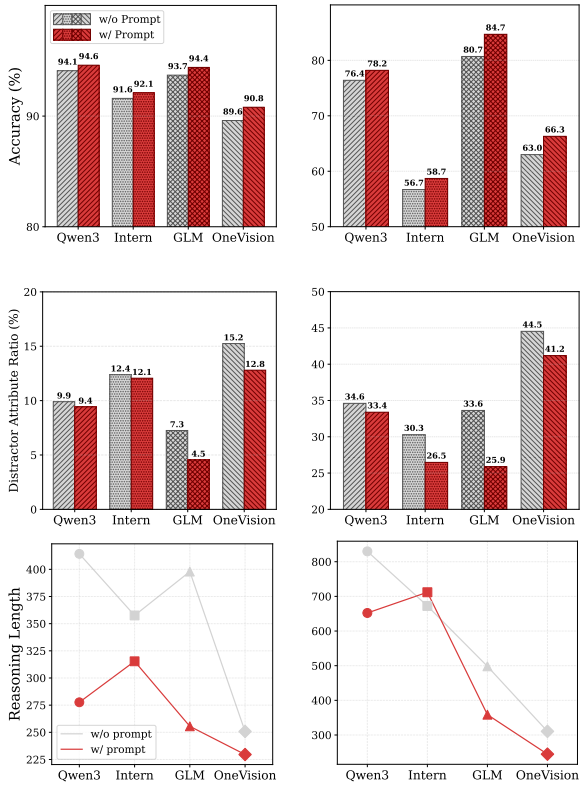
6 Mitigation strategy

Based on our hypothesis from Section 5.3, we ask whether reasoning failures can be mitigated by encouraging the model to focus more on attributes that are genuinely relevant to the target object. To this end, we design a prompt-based strategy that explicitly steers reasoning VLMs toward target object attributes. We then extend this investigation from the controlled setting of Idis to the Waterbirds dataset (Sagawa et al., 2020), a standard benchmark for visual bias where the bird label is spuriously correlated with the background environment. Waterbirds is a particularly suitable benchmark for our hypothesis because the distractor is not a localized object but a broad background cue, enabling us to examine whether the same limitation and mitigation strategy carry over to a more realistic bias setting. Through this extension, we test both the generality of our interpretation and the practical usefulness of the prompting strategy.

6.1 Attribute-guided prompting

To mitigate the reasoning failures discussed above, we adopt a prompting strategy to steer the chain-of-thought to target object attributes. Specifically, for the Idis dataset, where multiple distracting objects are present in the scene, we append the instruction: *"Base your reasoning only on the target object's own visual attributes."* For the Waterbirds dataset, where spurious correlations arise primarily from background cues, we instead use: *"Think step by step based on the foreground bird's attributes."*

As summarized in Fig. 10, we compare the performance of reasoning VLMs with and without the proposed prompt strategy on both Idis and Waterbirds, along three axes: accuracy, distractor-related attribute ratio, and reasoning length. On both datasets, the prompt strategy consistently improves accuracy. Moreover, it lowers the distractor-related attribute ratio, reflecting reduced reliance on environment-related cues. Notably, prompting the models to reason based on the target-object attributes yields reasoning lengths that are comparable to, or even shorter than, those obtained without prompting. These findings suggest that attribute-guided prompting can function as an effective, training-free debiasing method at test time for reasoning VLMs. It serves as an initial step toward future work in developing debiasing techniques for reasoning VLMs.



(a) Idis (b) Waterbirds

Figure 10: **Prompting improves accuracy, while reducing reliance on spurious attributes and average reasoning length.** Top: the prompt strategy consistently improves accuracy across all models when distractors are negatively spuriously correlated with the target object. Middle: the use of prompting reduces the ratio of distractor-related attributes, indicating less reliance on spurious cues irrelevant to the target object. Bottom: prompting results in average reasoning lengths that are either comparable to or substantially shorter than those without prompting.

7 Conclusion

We extend the study of test-time inverse scaling to reasoning VLMs by introducing Idis, a VQA dataset that systematically varies the semantic relevance, quantity, and spatial scale of visual distractors. We show that accuracy drops are especially severe under semantically conflicting distractors, and that unlike in reasoning LMs, visual distractors reduce accuracy without substantially increasing reasoning length. Our attribute-level analysis reveals a lack of a mechanism to prioritize target-relevant attributes over distracting ones. Based on these insights, we explore prompt-based bias mitigation on both Idis and Waterbirds. We hope this work offers useful analytical tools for improving the robustness of reasoning VLMs.

Limitations

Our analyses focus on a VQA domain, which provides a controlled setup to analyze the effects of visual distractors. Extending this framework to more complex reasoning-heavy tasks, such as agentic decision making, multi-step planning, or mathematical reasoning, remains challenging and is an essential next step. Another important direction is to move beyond purely visual distractors and investigate how textual and visual distractors interplay in multimodal settings, where language descriptions, visual context, and spurious cues jointly shape the behavior of reasoning VLMs.

References

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*.

Kathleen C Fraser and Svetlana Kiritchenko. 2024. Examining gender and racial bias in large vision–language models using a novel dataset of parallel images. In *EACL*.

Aryo Pradipta 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. *arXiv preprint arXiv:2507.14417*.

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/.

Priya Goyal, Quentin Duval, Isaac Seessel, Mathilde Caron, Ishan Misra, Levent Sagun, Armand Joulin, and Piotr Bojanowski. 2022. Vision models are more robust and fair when pretrained on uncurated images without supervision. *arXiv preprint arXiv:2202.08360*.

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*.

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

Sepehr Janghorbani and Gerard De Melo. 2023. Multimodal bias: Introducing a framework for stereotypical bias assessment beyond gender and race in vision–language models. In *EACL*.

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: Debiasing classifier from biased classifier. *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. *NeurIPS*.

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

Chahat Raj, Anjishnu Mukherjee, Aylin Caliskan, Antonios Anastasopoulos, and Ziwei Zhu. 2024. Biasdora: Exploring hidden biased associations in vision-language models. In *Findings of EMNLP*.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "why should i trust you?" explaining the predictions of any classifier. In *KDD*.

Gabriele Ruggeri, Debora Nozza, and 1 others. 2023. A multi-dimensional study on bias in vision-language models. In *Findings of ACL*.

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. *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. 2020. Noise or signal: The role of image backgrounds in object recognition. In *ICLR*.

Yisong Xiao, Xianglong Liu, QianJia Cheng, Zhenfei Yin, Siyuan Liang, Jiapeng Li, Jing Shao, Aishan Liu, and Dacheng Tao. 2025. GenderBias-VL: Benchmarking gender bias in vision language models via counterfactual probing: Y. xiao et al. *IJCV*.

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. *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*.

N (Distractors)	Resolution	Images per semantic			
		Conflicting	Irrelevant	Aligned	Total
Idis					
0	224×224	–	–	–	4,050
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
Idis-manual					
1	224×224	4,050	4,050	4,050	12,150
Overall Total	—	**20,250**	**20,250**	**20,250**	**64,800**

Table 1: **Statistics of Idis and Idis-manual datasets.** Both subsets are summarized by distractor count (N), semantic type, and resolution. Idis-manual includes only N=1 samples at 224×224, while Idis spans N=0–4 with 1024×1024 resolution for $N \geq 1$.

A Implementation details

A.1 Prompts for dataset generation

To generate each distractor-conditioned sample in the Idis dataset, 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**.

A.2 Dataset detail

Dataset statistic. Our Idis and Idis-manual datasets contain a set of images across nine semantic classes. Each class contributes 450 images, yielding 4,050 images per semantic split. As summarized in Table 1, where each distractor configuration ($N \geq 1$) provides 4,050 images for each of the three semantic types—Conflicting, Irrelevant, and Aligned—resulting in 12,150 images per N level. The Idis-manual subset includes only the N=1 setting at 224×224 resolution, similarly containing 4,050 images per semantic type. Overall, the combined dataset comprises 64,800 images.

Defined object. As shown in Table 2, 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.

Idis-manual. We place distractors so that they do not occlude the target object. When avoiding overlap is not possible, we place the distractors in the top-right corner of the image.

A.3 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

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

	Model Configuration	
Model	**Reasoning**	**Non-Reasoning**
Qwen3-VL	Qwen/Qwen3-VL-8B-Thinking	Qwen/Qwen3-VL-8B-Instruct
Intern-S1	internlm/Intern-S1-mini	internlm/Intern-S1-mini*
GLM-4.1V	zai-org/GLM-4.1V-9B-Thinking	"don't think" prompting†
Qwen2.5-VL	Fancy-MLLM/R1-Onevision-7B-RL	Qwen/Qwen2.5-VL-7B-Instruct

Table 3: **Reasoning and non-reasoning model configurations.** Intern-S1* provides a built-in option to turn reasoning mode on or off. GLM-4.1V† does not provide a non-reasoning checkpoint; we disable reasoning via a "don't think" suppression prompt at inference time.

together with structured, class-aware instructions to a specialized large language model (DeepSeek-V3.2-Exp) (See Table 4). 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, objects, morphological cues, or class-related features. We define 10 attribute categories corresponding to the nine semantic classes in Idis (i.e., dog, bird, vehicle, reptile, carnivore, insect, instrument, primate, fish) plus an 'other' category. 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. Multi-word expressions such as long tail are treated as a single attribute. Each extracted phrase is then assigned to exactly one of the ten predefined categories.

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 5.

Defined metrics.

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

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

$$r_{\text{attr}} = \frac{\#\text{Distractor Attributes}}{\#\text{Total Attributes}}. \tag{1}$$

The *Distractor Area Ratio* measures the relative spatial extent of distractors compared with the target object:

$$r_{\text{area}} = \frac{A_{\text{dist}}}{A_{\text{dist}} + A_{\text{target}}}. \tag{2}$$

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

A.4 Model information

The detailed model configurations for both reasoning and non-reasoning settings are summarized in Table 3. Some models naturally provide separate checkpoints for reasoning and non-reasoning behavior (e.g., Qwen3-VL and Qwen2.5-VL). For GLM-4.1V, we construct the non-reasoning variant by applying a "don't think" prompt, effectively turning off reasoning during inference. In contrast, Intern-S1 provides a built-in option to explicitly turn off reasoning turns, allowing clean control over its non-reasoning mode without additional prompting.

A.5 Hardware

We conducted all inference experiments on a single NVIDIA RTX A6000 and a single RTX A6000 ADA GPU. All model evaluations were performed using precision BF16.

A.6 Prompts

Inference prompt. We employ different inference prompts depending on the dataset and whether the model is configured for reasoning or non-reasoning behavior:

- Waterbirds
  - Reasoning model prompt

    > ***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.

  - Non-reasoning model prompt

    > ***Prompt***: [Question] Is the bird in the image a waterbird or a landbird? At the end, provide your final answer and clearly indicate it using $< answer > X < /answer >$ format.

  - 'don't think' prompt

    > ***Prompt***: [Question] Is the bird in the image a waterbird or a landbird? Don't think. Directly provide your answer and clearly indicate your final answer using $< answer > X < /answer >$ format.

  - Prompt strategy

    > ***Prompt***: [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.

- Idis
  - Reasoning model prompt

    > ***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.

  - Non-reasoning model prompt

    > ***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. At the end, provide your final answer and clearly indicate it using $< answer > X < /answer >$ format.

  - Prompt strategy

    > ***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. First identify the main object. Base your reasoning only on that object'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.

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 4.
- Waterbirds attribute extraction: The biological/environmental attribute extraction prompt shown in Table 5.

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 4: **System prompt for Idis dataset.**

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 5: **System prompt for Waterbirds dataset.**

B Qualitative examples

B.1 Effect of 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>

B.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.

B.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

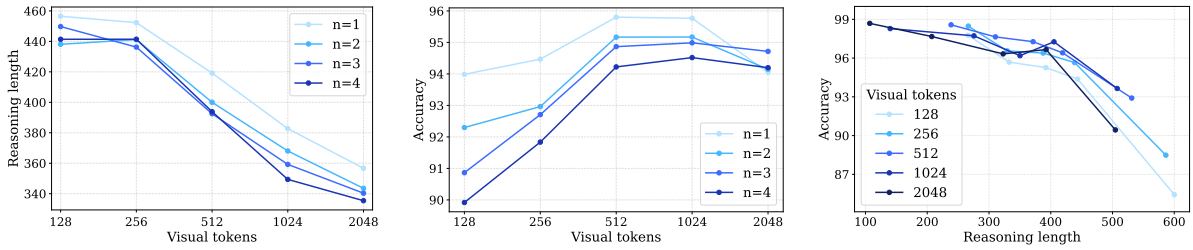
Model Output:

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

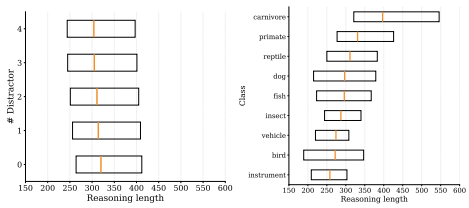
Model Output:

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 11: **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 12: **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.

C 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.

C.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. 12, 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.

C.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. 11a, 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. 11b 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. 11c, 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. 11b

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.

C.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.

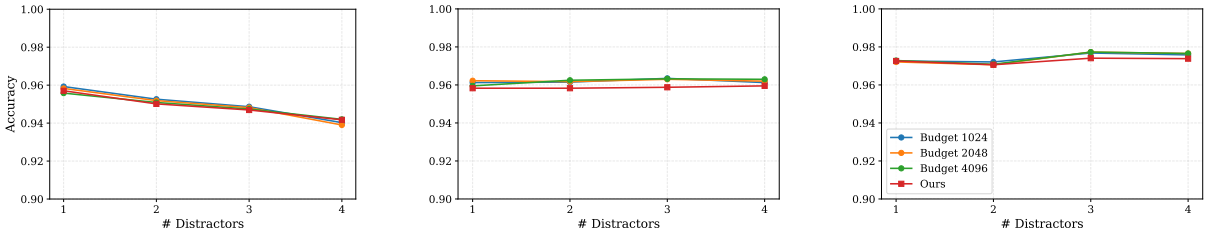
Across the four distractor conditions, the average response lengths were $\{397.4, 378.5, 370.7, 361.0\}$ tokens for 1 through 4 distractors, respectively. At the single-image level, this corresponds to an average per-step slope of 11.8 tokens per additional distractor and an average reduction of 36.6 tokens when increasing the number of distractors from 1 to 4. 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. In other words, 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.

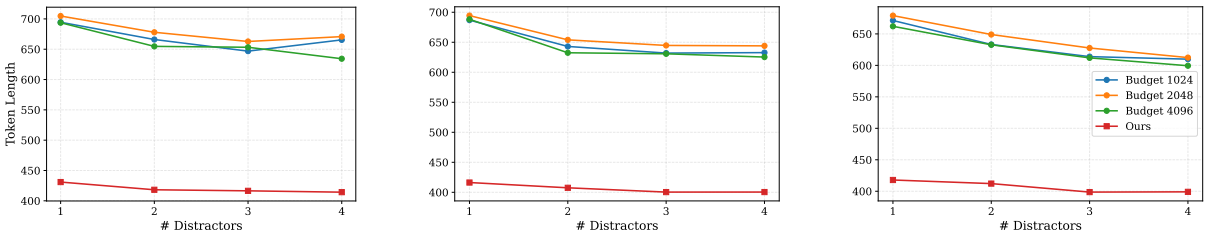
D Controlled reasoning budgets

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. 13 and Fig. 14, 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. 13 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 13: **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 14: **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").

Size	Qwen3-VL-Thinking Acc.	Intern-S1-mini Acc.	GLM-4.1V-Thinking Acc.	R1-OneVision 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 6: **Accuracy results on the Idis-manual dataset.** Accuracy across three distractor-size conditions (Small, Medium, Large) for four reasoning VLMs.

E Additional experimental results

E.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. 15 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. Finally, Fig. 16 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.

E.2 Full results on Idis-manual dataset

In this subsection, we present additional quantitative results on the Idis-manual dataset, where we explicitly control the distractor size. As shown in Fig. 17, 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 6 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

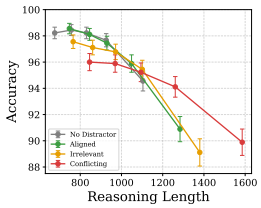
Table 7: **Table results of four vision-language models on the Waterbirds dataset.** We report accuracy and average reasoning length for aligned, conflicting, and overall conditions across non-reasoning VLMs, reasoning VLMs, and prompt-strategy settings.

E.3 Detailed results for debiasing experiments

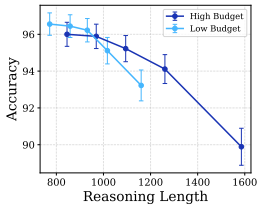
Table 7 presents the full results of our debiasing experiments on the Waterbirds dataset, extending the summary trends shown in Fig. 10b. All four reasoning VLMs with the prompt strategy show better performance on the conflicting. These detailed results confirm that our debiasing prompt can mitigate spurious-correlation failures on Waterbirds by steering the reasoning VLMs to reason primarily based on attributes of the target object rather than spurious cues.

F Experiments on proprietary model

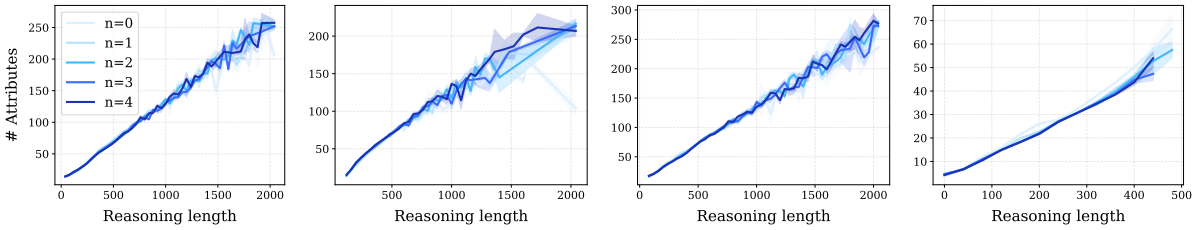
To examine whether test-time inverse scaling generalizes beyond open-weight models, we conduct additional experiments on Gemini 3 Flash, a proprietary SOTA model, using the Idis dataset.



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.

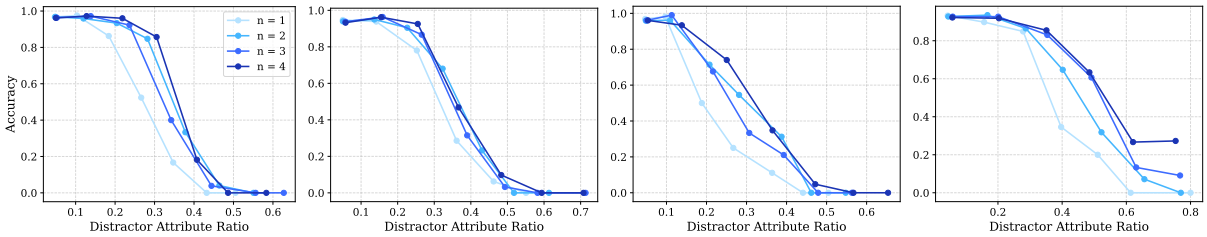


Furthermore, by varying Gemini 3 Flash's native thinking budget, we find that the "High budget" setting, which encourages longer reasoning, consistently yields lower accuracy than the "Low budget" setup across all reasoning lengths. These results suggest that test-time inverse scaling is a broader phenomenon.



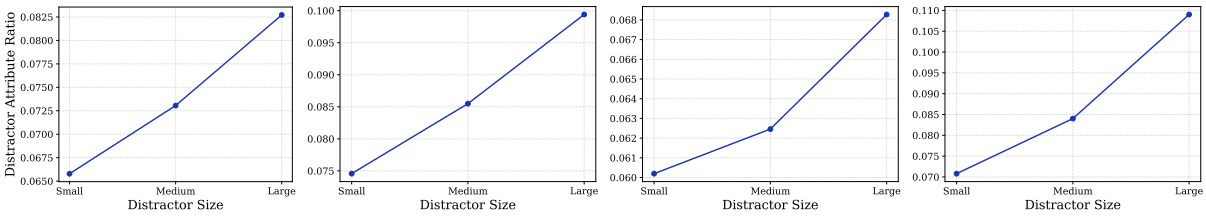
(a) Qwen3-VL-Thinking (b) Intern-S1-mini (c) GLM-4.1V-Thinking (d) R1-OneVision

Figure 15: **Extended visualization to other models of Fig. 7a.** 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 16: **Extended visualization to other models of Fig. 7b.** 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 17: **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.