Mental Imagery That Matters: Curing Latent Collapse in Multimodal Reasoning

Mi Luo[1] **Rohan Dugad**[1] **Zihui Xue**[1] **Alex Dimakis**[2, 3] **Kristen Grauman**[1]
[1]The University of Texas at Austin [2]UC Berkeley [3]Bespoke Labs

Abstract

Mental simulation, the ability to reason by constructing and updating internal visual states, is central to flexible human cognition, but remains elusive in current multimodal large language models. Latent multimodal reasoning offers a promising route toward this capability by interleaving text with continuous visual latent blocks, allowing models to maintain internal visual workspaces without decoding them into pixels. In this paper, we ask whether these latent visual states actually matter. Through a direct perturbation analysis across representative latent-reasoning MLLMs, we find that they often do not: dropping, replacing, or shuffling generated visual latents has only minor effects on the model's own final-answer probability, while equivalent perturbations to the linguistic reasoning trace are substantially more damaging. We call this failure mode *Mental Simulation Collapse* (MSC): the model appears to perform interleaved visual reasoning, but routes most answer-relevant computation through text. We further diagnose that MSC follows naturally from existing training objectives: visual alignment can make generated latents resemble helper-image embeddings, but it does not require the decoder to rely on them; subsequent text-only supervision further leaves the semantic role of the latent pathway unidentifiable. To address this, we introduce **AnchorSim** (**Anchor**ed Mental **Sim**ulation), a contrastive distillation framework that trains models to make their generated visual latents consequential. Across multiple visual reasoning benchmarks, AnchorSim improves performance, while diagnostic re-evaluation shows that visual perturbations now meaningfully affect predictions. Our results expose a fundamental failure mode of latent multimodal reasoning and demonstrate that mental simulation must be anchored to model behavior in order to matter.[1]

1 Introduction

Human visual reasoning often unfolds as an internal simulation rather than a purely linguistic process [32]. When people solve tasks involving spatial layouts, object transformations, physical dynamics, or navigation, they construct and update visual states in their mind: rotating a shape, tracing a path through a maze, or imagining how one object affects another [4, 2]. These intermediate states are not merely memories retrieved once and verbalized. They are manipulated, inspected, and revised over time [37]. For multimodal large language models, this suggests a natural goal: visual reasoning should not have to be compressed entirely into text [60]. Models should maintain and consult task-relevant visual states throughout their reasoning trajectory.

Recent work has moved toward this goal through *interleaved multimodal reasoning*, where a model alternates between natural-language reasoning and generated visual representations. Some approaches make these representations explicit by producing intermediate images or video frames [55, 29, 7, 22, 14, 25]. While interpretable, such methods are computationally expensive and require substantial image-generation machinery. Others rely on external visual tools, such as drawing, cropping,

[1]Project page: https://vision.cs.utexas.edu/projects/AnchorSim/

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

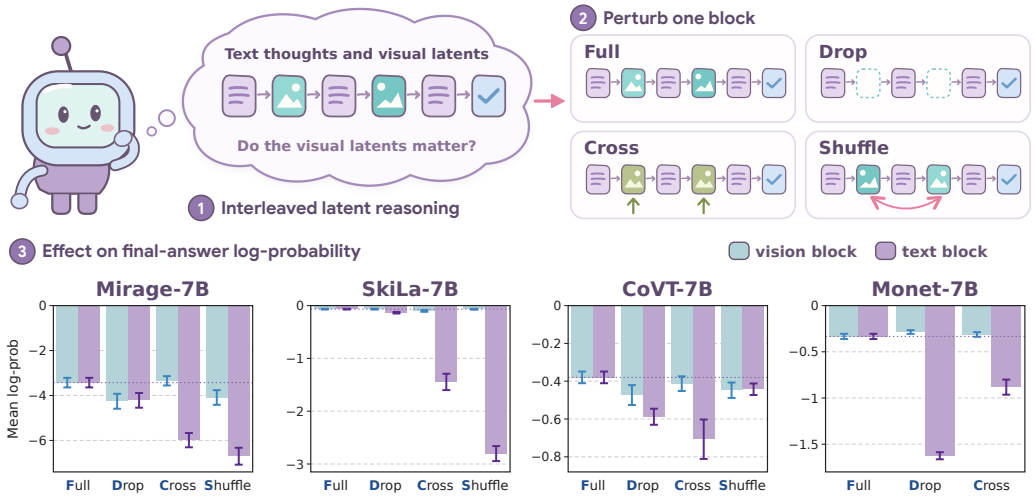


Figure 1: **Vision vs. text perturbation impact across four multimodal latent-reasoning models.** Each panel reports the mean log-probability (±1 SEM) of the model's final-answer span. We compare four conditions applied independently to the *vision block* (blue) and the *reasoning text block* (purple): **F**ull (unchanged), **D**rop (removed), **C**ross (replaced from another sample), and **S**huffle (permuted). Step 2 shows them on the vision block; the charts apply them to both blocks in turn. A larger drop from **F**ull indicates that the perturbed block carries more answer-relevant information. In most conditions, perturbing the text block degrades the answer substantially more than perturbing the visual block, indicating that computation is routed primarily through the linguistic stream.

detection, or code execution, but are limited by the operations exposed through the tool interface [64, 62, 10, 21, 33, 67, 18]. A more lightweight alternative is to reason directly in latent space [61, 23, 34, 42, 48, 8]. Instead of rendering pixels, the model emits continuous visual latent blocks between text spans, producing trajectories of the form $\langle \texttt{text}_0, \texttt{vis}_0, \texttt{text}_1, \texttt{vis}_1, \ldots, \texttt{answer} \rangle$. In principle, these latent blocks could serve as compact mental images: internal visual workspaces that preserve information too cumbersome to express in language, and that later reasoning can use. This premise assumes that generated visual latents actually matter. If answers rely entirely on surrounding text, the model is merely decorating a linguistic chain-of-thought with inert visual placeholders. We therefore ask: do latent-reasoning MLLMs causally use generated visual states to produce the final answer?

Surprisingly, for existing systems, the answer is largely no. Evaluating four recent latent-reasoning MLLMs by perturbing their generated trajectories reveals that dropping, replacing, or shuffling visual latents barely affects the final-answer probability. Conversely, equivalent perturbations to the linguistic trace are highly damaging, as shown in Fig. 1. These visual states act as decorative placeholders rather than load-bearing computations, a failure mode we term *Mental Simulation Collapse* (MSC). We further diagnose that MSC follows naturally from existing training pipelines [61, 8]. While initial alignment makes latents resemble helper images, subsequent text-only supervision only evaluates the final text distribution. Consequently, a decoder can achieve high likelihoods by routing information entirely through the linguistic stream and ignoring the vision. Making latents appear visual is therefore insufficient; the objective must force predictions to depend on them.

Motivated by this, we introduce **AnchorSim** (*Anchored Mental Simulation*) to train interleaved multimodal models to rely on their generated visual pathway. During training, AnchorSim contrasts a latent-aware forward pass with a latent-blind pass (where visual inputs and latents are masked). An anti-collapse objective pushes the latent-aware predictions away from the latent-blind distribution, with stop-gradient on the blind branch, forcing the model to encode consequential information in the visual states rather than relying solely on text. To stabilize these continuous latents, this contrastive pressure is combined with distillation from an exponential moving average teacher. At inference, the auxiliary blind branch is discarded, and the model generates standard interleaved trajectories.

We evaluate AnchorSim across nine visual reasoning benchmarks against state-of-the-art latent [34, 42, 61, 23] and RL-enhanced [19, 46] MLLMs. AnchorSim achieves the best average, improves over the base model by 13.5 points, and ranks first on seven of nine benchmarks. Crucially, diagnostic evaluations and ablations confirm these gains stem from genuine visual grounding: perturbing the

visual latents now meaningfully impacts predictions. These findings demonstrate that AnchorSim does not merely produce visual states in form, but mitigates collapse by making them functional components of multimodal reasoning. Our work provides the community with both a critical diagnostic framework to identify simulation collapse and a principled blueprint for building models that genuinely reason across modalities.

2 Related Work

Latent Reasoning Latent reasoning studies how models can perform intermediate computation in hidden or continuous states rather than expressing every step as natural language. In language models, this idea appears in approaches that introduce thinking tokens, recurrent latent states, or additional test-time computation before decoding the final answer [15, 13, 38]. These methods aim to increase reasoning capacity without requiring long explicit chains of text [51, 50]. Recent multimodal work extends this idea to visual reasoning by interleaving text with visual thought tokens or latent image-conditioned states [61, 23, 34, 42, 48, 8, 35]. Systems such as Mirage [61] suggest that multimodal models can reason through internal visual representations without always rendering pixels. Our work follows this broader direction, but focuses on a different question: not how to insert latent thoughts into the trajectory, but whether those thoughts are actually used. We show that existing latent visual reasoning models can emit visual latents while leaving the final prediction largely insensitive to them, a failure mode we call *Mental Simulation Collapse*. AnchorSim contributes to latent reasoning by adding an explicit anti-collapse training signal: it contrasts predictions made with the visual pathway against predictions made when that pathway is removed, encouraging latent thoughts to become causally relevant rather than merely present.

Visual Reasoning Recent progress in vision-language models [3, 24, 5, 20] has substantially improved visual perception, but robust reasoning over visual inputs remains a persistent challenge. Motivated by the success of chain-of-thought prompting [51] in language models, many works extend textual reasoning strategies to multimodal tasks, asking models to verbalize intermediate steps before producing an answer [19, 11, 46]. While effective in some settings, purely textual reasoning can drift away from the visual evidence [31] and be a poor fit for problems that require tracking spatial structure [43, 36], object transformations [63, 59], or intermediate visual states [39, 58]. This limitation has led to a growing body of work on visual forms of reasoning, including methods that generate explicit image-like thoughts [57, 41, 52, 47, 65, 49], use unified multimodal models to interleave text and visual outputs [55, 29, 7, 22, 14, 25], introduce latent reasoning tokens [61, 48, 23, 34], or call external tools such as drawing, cropping, detection, and code execution [64, 62, 10, 21, 33, 67, 18]. These approaches broaden the reasoning interface beyond language, but each comes with trade-offs: pixel-level generation can be expensive, tool-based reasoning depends on brittle predefined operations, and latent reasoning is efficient but may not guarantee that the visual states are actually used.

Our work contributes to this latent visual reasoning line by shifting the focus from producing visual thoughts to making them consequential. Rather than treating latent visual states as useful merely because they are interleaved with text or aligned to visual embeddings, we directly test whether perturbing them changes the model's prediction. This reveals Mental Simulation Collapse, a failure mode in which models emit visual latents but route most answer-relevant computation through the linguistic stream. AnchorSim addresses this gap with a training objective that contrasts latent-aware and latent-blind predictions, explicitly making it suboptimal for the model to ignore its generated visual pathway. In doing so, our work complements recent efforts on spatial and multi-step visual reasoning benchmarks and training data, while providing a general anti-collapse mechanism for making latent visual reasoning functionally grounded rather than only architecturally present. Concurrent work, CapImagine [27], finds that perturbing visual latents barely changes benchmark accuracy and shifts reasoning to explicit text. In contrast, we apply matched perturbations to both the visual and textual streams, analyze why the training objective permits collapse, and keep the latent pathway while penalizing the model for ignoring it.

3 Does the Mental Imagery Matter? Diagnosing Mental Simulation Collapse

A recent line of works [61, 8] propose that interleaving *visual latent blocks* into an autoregressive reasoning trajectory can give multimodal models a form of mental imagery: dedicated image-

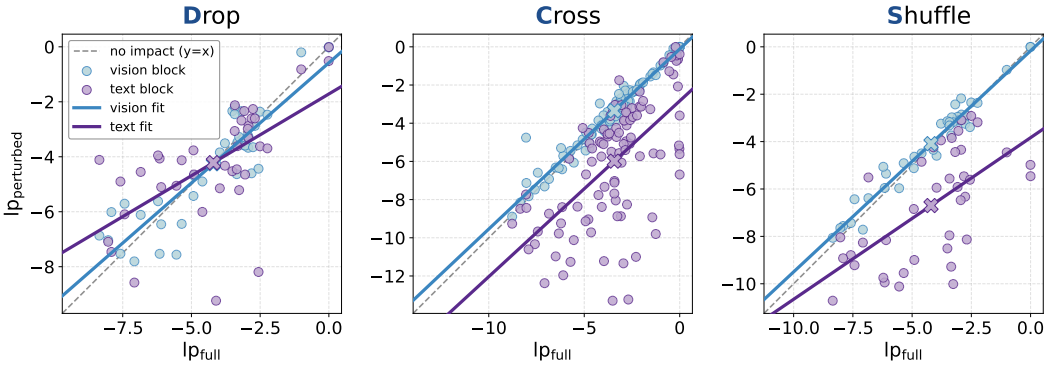


Figure 2: **Per-sample vision vs. text perturbation impact for Mirage-7B.** Each panel plots the perturbed final-answer log-probability ($y = \mathrm{lp}_{\mathrm{perturbed}}$) against the unperturbed baseline ($x = \mathrm{lp}_{\mathrm{full}}$) for the *vision block* (blue) and *reasoning text block* (purple). Points further below the $y{=}x$ diagonal (dashed) and lower ordinary-least-squares fit slopes (solid lines) indicate a larger performance drop. Across all conditions, the text fit has a lower slope than the vision fit, demonstrating that perturbing the linguistic chain degrades the answer more than perturbing the visual latents.

conditioned hidden states that the model can consult while reasoning. The idea is compelling, but it leaves open a basic question. When a model emits a trajectory of the form:

$$\langle \texttt{text}_0, \texttt{vis}_0, \texttt{text}_1, \texttt{vis}_1, \ldots, \texttt{text}_n, \texttt{answer} \rangle,$$

do the visual latent blocks actually affect the answer, or does the model rely mostly on the surrounding language? We answer this question with a direct perturbation test and find that, across representative latent-reasoning systems, the generated visual blocks contribute substantially less to the model's own answer than the linguistic blocks do. We refer to this failure mode as **Mental Simulation Collapse** (MSC): *the model's trajectory exhibits the surface form of visual reasoning, yet the answer-relevant computation is carried almost entirely by the textual stream.*

3.1 Quantifying Latent Dependence: A Perturbation Probe

Setup We start from four recent latent-reasoning MLLMs: MIRAGE-7B [61], SKILA-7B [42], COVT-7B [34], and MONET-7B [48]. For each model, we randomly select 100 samples for a pivot study. The samples are drawn from CoMT [6], BLINK [12], V* [53], and MMVP [44]+V* [53], respectively, for the four models. We use different benchmarks because, for some model–benchmark pairs, the model exhibits insufficient multi-hop reasoning behavior, defined as having fewer than two reasoning blocks. The probing process consists of two steps: (i) **Trajectory capture:** We record the model's greedy-decoded trajectory along with the hidden states at every visual latent slot. (ii) **Teacher-forced scoring.** We feed the captured trajectory back through the model under teacher forcing and record $\mathrm{lp}_{\mathrm{full}}$, the log-probability assigned to the answer span.

To localize where the answer-relevant computation resides, we re-score the same answer span after perturbing the captured trajectory, applying perturbations independently on the vision side or the text side. Vision-side perturbations target only the visual latent hidden states, while text-side perturbations target only the natural-language reasoning segments; in both cases, the rest of the trajectory is left intact. We consider four conditions: (i) `Full` leaves the trajectory unchanged and serves as the reference, (ii) `Drop` removes the block content, (iii) `Cross` replaces each block with the corresponding block from a donor sample, and (iv) `Shuffle` permutes the model's own blocks across positions, preserving real content but moving it to the wrong locations. The answer span itself is never modified, so any shift in log-probability reflects the contribution of the perturbed block to the model's prediction. Implementation details of each perturbation are provided in Appendix A.

Vision Blocks Barely Move the Answer Figure 1 summarizes the result. In most conditions, the blue vision bars under `Drop`, `Cross`, and `Shuffle` remain close to the `Full` reference, while the purple text bars fall noticeably lower. The same pattern appears consistent across all four models and their corresponding benchmarks: removing, replacing, or permuting visual latent blocks costs the model little log-probability on its own answer, whereas applying the same perturbations to the linguistic reasoning blocks is typically more damaging. This asymmetry is striking because the

vision-side perturbations directly target the latent states these architectures were designed to introduce. If the models were genuinely using their visual mental states as intermediate reasoning variables, their predictions should be highly sensitive to corruption of those states. Instead, the answer remains largely intact when the visual blocks are altered, suggesting that the model can bypass them and recover the answer from the textual stream.

Per-sample Evidence: the Gap is Not Driven by Outliers. To verify that the aggregate results in Figure 1 are not driven by outliers, we examine MIRAGE-7B on a per-sample basis in Figure 2. Each panel plots $\mathrm{lp}_{\mathrm{perturbed}}$ against $\mathrm{lp}_{\mathrm{full}}$ for a single perturbation condition, with the dashed $y{=}x$ diagonal marking no impact and points farther below the diagonal indicating larger degradation. For each category, we fit an ordinary least-squares regression line to summarize the overall trend. The per-sample analysis reinforces the aggregate finding in Figure 1. For every perturbation condition, the text fit has a lower slope than the vision fit, indicating that the typical example loses more answer log-probability when its text block is perturbed than when its visual latent block is perturbed. The vision slopes (0.87 for `Drop`, 0.94 for `Cross`, and 0.93 for `Shuffle`) are closer to the no-impact slope of 1 than the corresponding text slopes (0.59, 0.92, and 0.68). The gap is largest under `Drop` and `Shuffle`, the two conditions that most directly disrupt the structure of the reasoning trajectory. The asymmetry is therefore not driven by a small number of outliers: at the level of individual trajectories, the visual latent content is largely inert.

Taken together, these observations show that the visual latent blocks emitted by current latent-reasoning models behave less like load-bearing intermediate computations and more like decorative positions. Their hidden states can be dropped, swapped with another sample's, or shuffled out of order with only minor consequence for the model's own answer, whereas comparable perturbations to the natural-language reasoning chain produce much larger drops. This asymmetry echoes prior findings on text-only reasoning, where the *structure* of the reasoning chain matters [26] and the reasoning ability can be improved even with invalid individual content [45]. By the same diagnostic, the natural-language stream behaves like a genuine reasoning chain, and the visual latent stream does not. Although these architectures are capable of routing information through dedicated visual blocks, they have learned to channel the answer-relevant signal through language instead. MSC does not imply that these models gain nothing from latent blocks: they can still help through additional serial computation, a more structured reasoning trajectory, and changed attention to the input image, rather than through the content of the latents (Appendix F).

3.2 Why Latent Visual Reasoning Collapses

The previous paragraphs demonstrated consistent evidence of Mental Simulation Collapse (MSC) across different latent block designs (token-pad, sketch-token, hidden-state, and multi-encoder variants), multiple benchmarks, and a range of trajectory lengths, suggesting that the problem is not a peculiarity of any single architecture, dataset, or decoding format. Instead, it points to a fundamental training-level failure: current objectives permit models to generate visual latents without making future predictions causally depend on them.

Latent-reasoning methods [61, 8] typically follow a two-stage training recipe. Let $z_\theta = G_\theta(q, \mathbf{v}, t_{\mathrm{pre}})$ denote the visual latent generated from query q, image $\mathbf{v}$, and textual prefix t_{pre}. The decoder then predicts the answer through $p_\theta(y \mid q, \mathbf{v}, t_{\mathrm{pre}}, z_\theta)$. Training proceeds with

$$\mathcal{L}_{\mathrm{Stage1}} = \ell_{\mathrm{align}}(z_\theta, \hat{z}) + \gamma \mathcal{L}_{\mathrm{text}}, \qquad \mathcal{L}_{\mathrm{Stage2}} = \mathcal{L}_{\mathrm{text}} = -\log p_\theta(y \mid q, \mathbf{v}, t_{\mathrm{pre}}, z_\theta). \tag{1}$$

Stage 1 aligns z_θ to a helper-image embedding $\hat{z}$, while Stage 2 removes the alignment loss and optimizes only the text likelihood. The implicit assumption is that Stage 1 establishes visual grounding and Stage 2 preserves it while adapting the latent for reasoning. This objective leaves the latent pathway underdetermined at both stages: Stage 2 cannot distinguish latent-dependent reasoning from latent-independent shortcuts, while Stage 1 constrains how the latent looks, not whether it is used.

Text-based supervision does not identify latent dependence. In both stages, the text loss observes only the output distribution, not the causal pathway producing it. Consequently, latent usage is not identifiable under text-only supervision. In particular, for any latent-dependent solution, a sufficiently expressive decoder can achieve nearly identical likelihood while ignoring the latent:

$$p_{\theta'}(y \mid q, \mathbf{v}, t, z) \approx p_\theta(y \mid q, \mathbf{v}, t, G_\theta(q, \mathbf{v}, t)) \qquad \forall z. \tag{2}$$

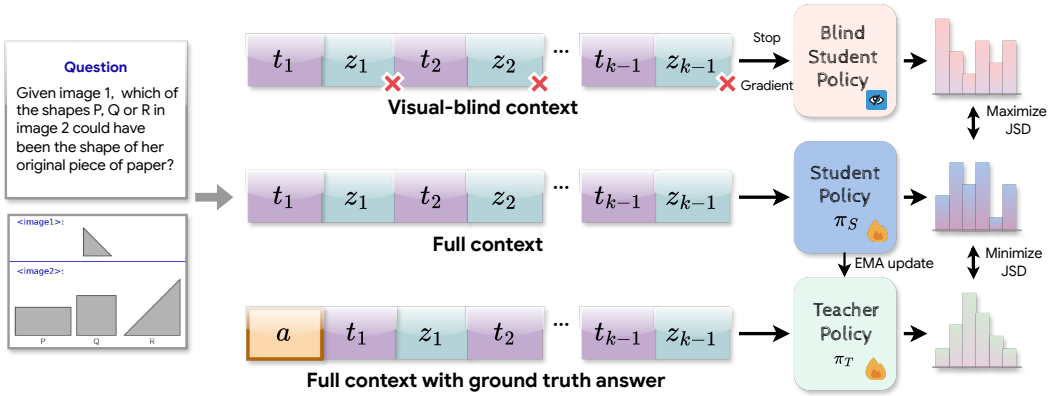


Figure 3: Overview of AnchorSim. Given an initial image and query, the model produces an interleaved reasoning trajectory consisting of textual thoughts t_k and intermediate visual simulations z_k. AnchorSim trains the student policy with two complementary signals. First, a visual-blind branch masks the previous generated visual state z_{k-1} and is stopped from receiving gradients; the student is encouraged to diverge from this blind prediction, preventing "lazy text" reasoning that ignores visual latents. Second, a simulation-anchored teacher, updated as an EMA of the student and conditioned on the ground-truth answer, provides reliable reasoning targets under the full context. The student is optimized to minimize the JSD to the teacher while maximizing the JSD from its visual-blind counterpart, thereby anchoring textual reasoning to the generated visual simulations.

Thus, the objective cannot distinguish whether the latent acts as a visual workspace, a non-visual scratchpad, or an unused placeholder. Moreover, collapse is stable: once the decoder becomes invariant to z, gradients through the latent pathway become weak or uninformative, removing the signal needed to maintain meaningful latent computation.

From an information-theoretic perspective [40], collapse corresponds to $I_\theta(y; z \mid q, \mathbf{v}, t) \approx 0$. The conditional mutual information $I_\theta(y; z \mid q, \mathbf{v}, t)$ measures how much the latent z tells us about the output y beyond what is already determined by the query q, image $\mathbf{v}$, and text prefix t; it equals zero precisely when y and z are conditionally independent given $(q, \mathbf{v}, t)$, i.e., when $p_\theta(y \mid q, \mathbf{v}, t, z) = p_\theta(y \mid q, \mathbf{v}, t)$. This is exactly the collapse condition: when the output distribution is invariant to z, the latent contributes no useful information, precisely the pattern observed in the probe experiments.

Such unidentifiability is not resolved by the visual alignment in the first stage. The alignment loss ($\ell_{\text{align}}(z_\theta, \hat{z})$) compares z_θ to a target embedding $\hat{z}$, and never inspects whether downstream computation uses the latent. Even when alignment is fully achieved and $z_\theta \approx \hat{z}$, the decoder may still satisfy $p_\theta(y \mid q, \mathbf{v}, t, z_\theta) \approx p_\theta(y \mid q, \mathbf{v}, t)$, leaving the latent visually plausible yet functionally inert. Adding visual supervision constrains how the latent *looks* but not how it is *used*, so the same condition $I_\theta(y; z \mid q, \mathbf{v}, t) \approx 0$ persists.

4 Anchored Mental Simulation (AnchorSim)

The empirical probe and theoretical analysis point to the same key takeaway: mental simulation fails not because current models lack visual latent states, but because training gives the decoder no reason to make its future predictions depend on those states. The perturbation results show that visual blocks are largely non-causal in practice, while the objective-level analysis explains why this behavior is allowed: text likelihood can be optimized through latent-independent shortcuts. Therefore, an effective method must do more than generate visually plausible intermediate states; it must explicitly make the textual reasoning distribution sensitive to the presence and content of those states. This motivates our **AnchorSim** (Anchored Mental Simulation), which directly targets the collapse mechanism by contrasting full-context predictions against visual-blind predictions and distilling reasoning that remains anchored to the generated visual simulations.

AnchorSim is a lightweight training algorithm that encourages causal dependence between the model's reasoning and its visual latents. It is built on two core components: (i) *Contrastive Latent Anchoring*, which enforces a measurable difference between the model's predictions with and without

visual content; and (ii) *Anchored Reasoning Distillation*, which supplies high-quality reasoning that coherently bridges each latent to the final answer. Figure 3 illustrates the overall framework.

Given an initial visual input $\mathbf{v}$ and a textual query q, the model produces an interleaved reasoning trajectory $\mathcal{T} = (t_1, z_1, \ldots, t_K, z_K, a)$, where t_k represents a text reasoning chunk, z_k represents a generated intermediate visual state (mental simulation), and a is the final answer. When predicting the k-th text chunk, the *full context* is defined as $C_{<k} = (q, \mathbf{v}, t_1, z_1, \ldots, t_{k-1}, z_{k-1})$. We further define an *visual-blind context* $\tilde{C}_{<k}$, wherein all previous visual state $\{z_1, z_2, \ldots, z_{k-1}\}$ is masked from the attention mechanism: $\tilde{C}_{<k} = C_{<k} \setminus \{z_1, z_2, \ldots, z_{k-1}\}$.

4.1 Contrastive Latent Anchoring

The diagnosis in Section 3.2 identified the core problem: standard next-token training does not require the model's predictions to depend on the visual latent stream, so the model is free to ignore it whenever language priors alone yield a plausible continuation. Contrastive Latent Anchoring closes this loophole directly. For every reasoning step, we form two next-token distributions from the student policy π_S: one conditioned on the full context $C_{<k}$, and one conditioned on the visual-blind context $\tilde{C}_{<k}$ in which all preceding latents $\{z_1, \ldots, z_{k-1}\}$ have been masked. We then penalize agreement between the two distributions, since it implies that the latent stream contributed nothing:

$$\mathcal{L}_{\text{anchor}}(\theta) = -\,\mathbb{E}_{\mathcal{T},k}\left[\frac{1}{|t_k|}\sum_{n=1}^{|t_k|}\text{JSD}\Big(\pi_S(\cdot \mid C_{<k}, t_{k,<n}) \,\Big\|\, \text{sg}\big[\pi_S(\cdot \mid \tilde{C}_{<k}, t_{k,<n})\big]\Big)\right], \tag{3}$$

where π_S is the student policy, JSD denotes the Jensen–Shannon divergence [28] and $\text{sg}[\cdot]$ stops gradients through the visual-blind branch. Minimizing $\mathcal{L}_{\text{anchor}}$ enforces a measurable gap between predictions with and without the visual stream.

4.2 Anchored Reasoning Distillation

To ensure the text remains logically sound while anchored to the visual state, we distill high-quality reasoning pathways from a privileged teacher π_T. Rather than acting as a static target, the teacher shares the student's architecture and is parameterized by an exponential moving average (EMA) of the student's weights $\bar{\theta} \leftarrow \alpha \cdot \bar{\theta} + (1-\alpha)\cdot\theta$, motivated by [17].

The teacher is conditioned on the ground-truth answer a, which is injected into its prompt as a privileged prefix [66] [9]. This access lets the teacher reliably interpret each generated latent z_{k-1} and produce textual reasoning that coherently bridges the visual state to the correct answer. We distill this answer-anchored reasoning into the student via

$$\mathcal{L}_{\text{distill}}(\theta) = \mathbb{E}_{\mathcal{T},k}\left[\frac{1}{|t_k|}\sum_{n=1}^{|t_k|}\text{JSD}\Big(\pi_T(\cdot \mid C_{<k}, a, t_{k,<n}) \,\Big\|\, \pi_S(\cdot \mid C_{<k}, t_{k,<n})\Big)\right], \tag{4}$$

where the student and teacher consume the same student-generated latents z_{k-1}; the teacher's role is purely interpretive, supplying a learning target that uses the latents faithfully.

4.3 The AnchorSim Training Objective

We train the model end-to-end by minimizing the *AnchorSim* loss, which is a weighted sum of the distillation and anchoring objectives from the previous subsections:

$$\mathcal{L}_{\text{AnchorSim}}(\theta) = \mathcal{L}_{\text{distill}}(\theta) + \lambda\,\mathcal{L}_{\text{anchor}}(\theta), \tag{5}$$

where $\lambda > 0$ scales the anchoring penalty. Contrastive anchoring encourages the latents to carry answer-relevant, image-conditioned information that affects subsequent predictions, while answer-conditioned distillation teaches the model to use that content productively.

5 Experiments

Training Strategy We adopt QWEN2.5-VL-7B [3] as our base model, following common practice in prior works [61, 34, 23]. We post-train it in two stages: **Stage 1** warms up the model for multimodal

Table 1: Comparison with proprietary and open-source baselines across 9 public visual reasoning benchmarks. Best and second-best results are highlighted in **bold underlined** and **bold** style, respectively. We report BLINK with both the Jigsaw subset and overall average, CV-Bench with its 2D and 3D splits, and Avg as the average performance over 9 benchmarks, where BLINK uses its overall average and CV-Bench uses the mean of its 2D and 3D splits. *Results are reported from prior work.

Model	BLINK		V*	MMVP	EMMA	VisuLogic
	Jigsaw	Avg				
GPT-4o [20]*	-	**60.0**	66.0	70.7	-	-
Claude-4-Sonnet [1]*	-	39.6	15.2	-	-	-
Vision-r1 [19]*	27.3	51.0	80.1	72.6	24.3	15.2
PixelReasoner [46]*	52.7	-	80.6	-	23.5	23.4
LVR [23]*	52.7	52.5	81.7	72.0	-	-
CoVT [34]	62.7	56.0	78.0	**77.3**	22.7	24.6
SkiLa [42]	45.3	53.0	**84.3**	75.3	21.5	24.1
QWEN-2.5-VL-7B [3]	61.3	55.5	77.5	74.0	24.3	25.6
QWEN-2.5-VL-7B (interleaved)	**64.7** (+3.4)	58.0 (+2.5)	79.1 (+1.6)	75.7 (+1.7)	**25.5** (+1.2)	**25.8** (+0.2)
ANCHORSIM-7B (Ours)	**66.0** (+4.7)	**59.4** (+3.9)	**82.7** (+5.2)	**78.3** (+4.3)	**26.0** (+1.7)	**27.5** (+1.9)

Model	CV-Bench		SAT-R	VSP	CoMT	Avg
	2D	3D				
Vision-r1 [19]*	8.0	37.6	-	-	-	-
CoVT [34]	76.0	**84.7**	64.7	6.0	27.8	48.6
SkiLa [42]	**76.7**	74.4	56.7	4.3	32.3	47.5
QWEN-2.5-VL-7B [3]	76.6	73.3	59.7	9.3	39.5	48.9
QWEN-2.5-VL-7B (interleaved)	**76.7** (+0.1)	77.3 (+4.0)	**65.3** (+5.6)	**74.8** (+65.5)	**56.3** (+16.8)	**59.7** (+10.8)
ANCHORSIM-7B (Ours)	**80.4** (+3.8)	**86.4** (+13.1)	**68.7** (+9.0)	**78.0** (+68.7)	**58.0** (+18.5)	**62.4** (+13.5)

interleaved reasoning following Mirage [61]; **Stage 2** further post-trains the Stage 1 checkpoint with our proposed AnchorSim objective. Both stages use the same training mixture, drawn from Zebra-CoT [22], SAT [36], CoMT [6], and VSP [54]. Since SAT does not provide interleaved reasoning chains, we supervise only its final-answer tokens during training. Detailed dataset statistics are reported in the Appendix. We refer to the resulting model as ANCHORSIM-7B.

Benchmarks We evaluate on 9 public visual reasoning benchmarks that may benefit from latent-space mental imagery, organized into four families: (i) *perception and fine-grained visual search*: BLINK [12], MMVP [44], and V* [53]; (ii) *2D and 3D spatial reasoning*: CV-Bench-2D, CV-Bench-3D [43], and SAT-Real [36]; (iii) *abstract and multimodal STEM reasoning*: EMMA [16] and VisuLogic [56]; and (iv) *visual planning and step-wise manipulation*: VSP [54] and CoMT [6]. Together, these benchmarks cover low-level perception, spatial and depth reasoning, diagram-grounded STEM problem solving, abstract pattern induction, planning, and multi-step visual-state tracking.

Implementation Details We use an EMA teacher with decay rate $\alpha = 0.99$. The latent block size is set to 8 tokens by default. Stage 1 is trained for 3 epochs, after which we follow Mirage [61] and fine-tune for 1 additional epoch with text-only supervision. Stage 2 is initialized from the Stage 1 checkpoint and trained for 1 further epoch with AnchorSim. All training is performed on 8 NVIDIA H200 GPUs with a per-GPU batch size of 4.

Baselines We compare ANCHORSIM-7B against both proprietary and open-source MLLMs. For proprietary systems, we include GPT-4o [20] and Claude-4-Sonnet [1]. For open-source baselines, we compare with two families: (i) reinforcement-learning-enhanced MLLMs, namely Vision-R1 [19] and PixelReasoner [46]; and (ii) recent latent-space reasoning models with publicly released checkpoints, including LVR [23], CoVT [34], and SkiLa [42]. For CoVT, we use the released COVT-7B-SEG_DEPTH_DINO checkpoint throughout. We also compare against our base model QWEN2.5-VL-7B [3] and the Stage 1 model trained with Mirage [61]. Together, these baselines span a broad spectrum of contemporary vision-language systems: from large commercially deployed models to lightweight community releases—ensuring that our evaluation is both representative of the strongest available competitors. Further details on the evaluation setup are provided in Appendix C.

5.1 Main Results

Anchor-Based Simulation Improves Visual Reasoning Table 1 compares ANCHORSIM-7B with proprietary models and recent open-source visual reasoning baselines across the 9 benchmarks spanning perception, spatial reasoning, abstract logic, STEM reasoning, planning, and step-wise manipulation. ANCHORSIM-7B achieves the strongest overall performance, attaining the best

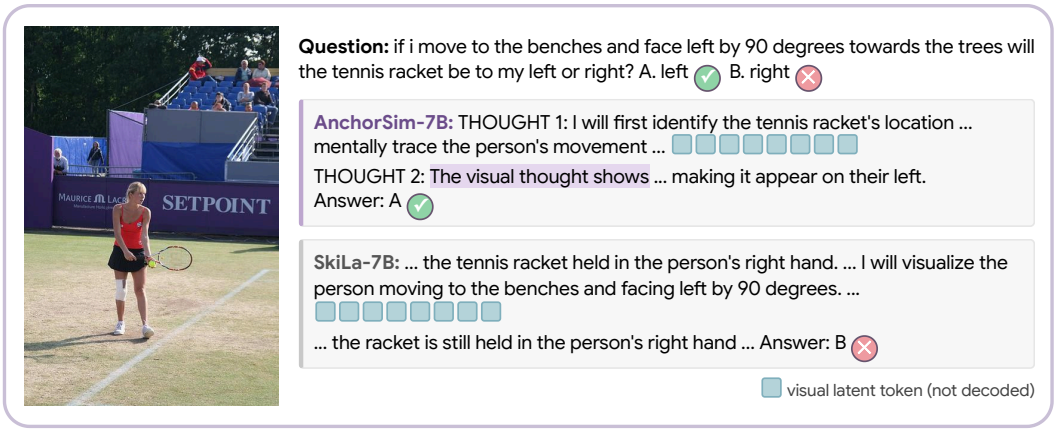


Figure 4: **Qualitative example on SAT-R.** AnchorSim-7B achieves better visual–text interleaving and correctly predicts the answer. Blue squares denote visual latent tokens.

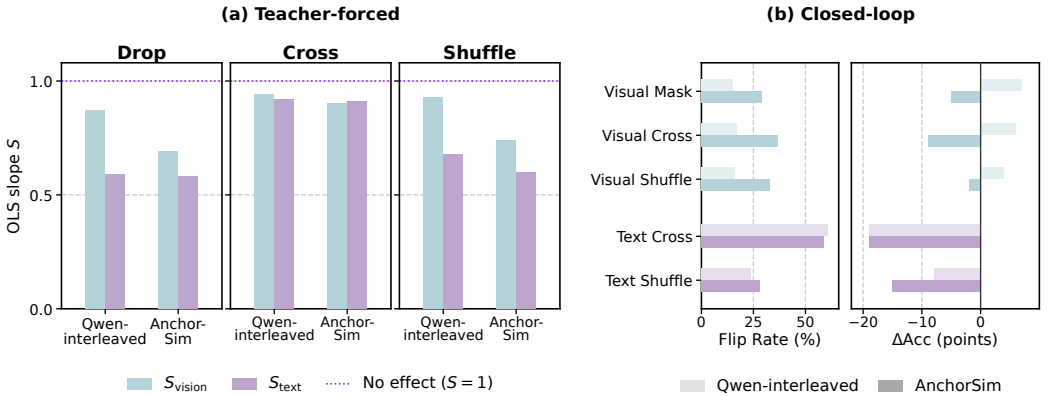


Figure 5: Perturbation analysis. (a) OLS slope S between perturbed and unperturbed answer log-probabilities for Qwen-interleaved and AnchorSim, where $S \approx 1$ indicates little effect. (b) Closed-loop analysis, where the model regenerates the rest of the trajectory after the perturbation.

average score of 62.4% and outperforming the Qwen2.5-VL-7B base by 13.5 points and the Stage 1 interleaved model by 2.7 points. The gains are consistent across reasoning categories: AnchorSim-7B ranks first on all benchmarks except BLINK and V*, where it ranks second. This demonstrates strong generalization beyond any single benchmark family. The large improvements on VSP and CoMT in particular suggest that AnchorSim substantially benefits tasks requiring multi-step visual state tracking and planning. Compared with the Stage 1 Mirage model, AnchorSim-7B further improves performance on nearly all benchmarks, indicating that the proposed Stage 2 objective provides complementary gains by strengthening grounded latent-space visual reasoning. Figure 4 compares AnchorSim-7B and SkiLa-7B on SAT-R, where AnchorSim-7B predicts the correct answer and SkiLa-7B does not. Additional examples are provided in Appendix H.

5.2 Does AnchorSim Mitigate MSC?

Teacher-Forced Perturbation We apply the same perturbation analysis as in Section 3 to Qwen-interleaved and AnchorSim. The base Qwen2.5-VL-7B does not generate visual latents, so the same intervention cannot be applied to it. As shown in Figure 5(a), AnchorSim makes visual perturbations more damaging while sensitivity to the text stream remains similar, reducing the gap $S_{\text{vision}} - S_{\text{text}}$ by 61% under `Drop` and 44% under `Shuffle`. Under `Cross`, the near-uniform downward shift is largely absorbed by the regression intercept, so slopes stay close to 1 for both models. These results indicate that AnchorSim mitigates MSC. We next test whether this holds when the model regenerates its trajectory.

Closed-Loop Perturbation The analysis above holds the downstream reasoning fixed and measures direct answer dependence. To capture effects that propagate through subsequent reasoning, we

perturb either the first visual latent block or the reasoning-text segment immediately following it in greedy rollouts of Qwen-interleaved and ANCHORSIM, and then regenerate all subsequent text, visual latent blocks, and the final answer. For the visual stream, we replace `Drop` with `Mask`, which prevents subsequent tokens from attending to the selected latent content, because after a literal deletion the model can immediately generate a replacement block. We report Flip Rate, the percentage of perturbations that change the final answer, and the change in accuracy relative to each model's unperturbed accuracy (52% for Qwen-interleaved and 60% for ANCHORSIM). As shown in Figure 5(b), Qwen-interleaved flips only 15–17% of answers under visual perturbations and its accuracy does not decrease, whereas for ANCHORSIM, perturbing the latents flips 29–37% of answers and reduces accuracy by up to 9 points. In contrast, flip rates under text perturbations remain similar across the two models. The low visual sensitivity of the baseline is therefore not an artifact of teacher forcing. Together, these results show that ANCHORSIM makes visual latents consequential both for the answer and for the regenerated reasoning, mitigating MSC.

5.3 Ablation Study

Effectiveness of Each Loss Term We first ablate the contribution of each loss term in our objective. Compared with the QWEN2.5-VL-7B base model, training with the visual-distillation loss $\mathcal{L}_{\text{distill}}$ consistently improves performance across all three benchmarks, yielding noticeable gains on BLINK, V*, and CV-Bench. Further adding the contrastive loss $\mathcal{L}_{\text{anchor}}$ boosts accuracy on every benchmark, with the largest marginal gain observed on V*, where $\mathcal{L}_{\text{anchor}}$ adds $+4.2$ points over the distillation-only variant. These results indicate that the two losses are complementary: distillation preserves task-relevant visual information, while the contrastive learning term further sharpens the learned representations by forcing the model to use them.

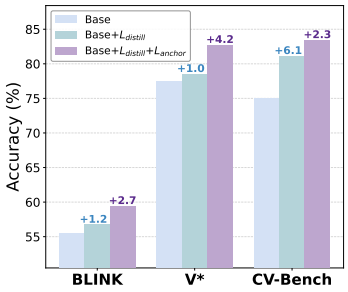


Figure 6: **Loss ablation.**

Scalability with Number of Latent Tokens We next study how performance scales with the number of visual latent tokens emitted per reasoning block. Introducing latent tokens improves over the no-latent Qwen base for every tested block size and benchmark, indicating that compact latent visual representations are beneficial even when only a small number of tokens are used. The trend, however, is not monotonic: BLINK and CV-Bench improve as the block size increases up to 8 tokens but degrade when further increased to 16, with CV-Bench dropping most noticeably. In contrast, V* continues to show a small improvement from 8 to 16 tokens. We therefore adopt 8 latent tokens as the default setting, due to the best overall balance.

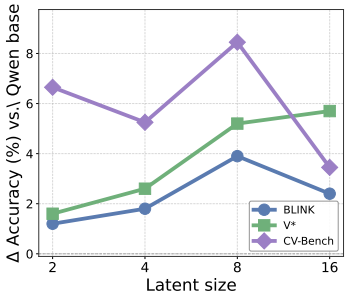


Figure 7: **Effect of latent size.**

6 Conclusions and Limitations

We identify *Mental Simulation Collapse* (MSC), where latent-reasoning multimodal models produce visual states that resemble mental simulation but barely affect final predictions, while answer-relevant computation is routed through text. To address this, we propose **AnchorSim** (**Anchor**ed Mental **Sim**ulation), a contrastive distillation framework that trains models to rely on generated visual latents instead of bypassing them through text-only shortcuts. Our results show that anchoring visual latents to model behavior improves visual reasoning and mitigates collapse by making latent simulations functional components of the reasoning process. ANCHORSIM is representation-agnostic: it does not require individual latent tokens to correspond to image crops or to be directly decodable, and we define their content operationally as answer-relevant, image-conditioned information. AnchorSim is evaluated mainly at the 7B scale MLLM like QWEN2.5-VL on benchmark-style tasks, so future work should test its scalability to larger models and real-world multimodal decision-making. More broadly, we see making internal visual representations functionally necessary, rather than merely present, as a step toward models that can genuinely simulate before they act.

Acknowledgements This research has been supported by computing support on the Vista GPU Cluster through the Center for Generative AI (CGAI) and the Texas Advanced Computing Center (TACC) at the University of Texas at Austin. UT Austin is supported by NSF AI Institute for Foundations of Machine Learning (IFML) 2019844, NSF Grants 2019844, 2112471, AF 1901292, CNS 2148141, Tripods CCF 1934932, Tripods 2217069, the Texas Advanced Computing Center (TACC) and research gifts by Western Digital, Wireless Networking and Communications Group (WNCG) Industrial Affiliates Program, UT Austin Machine Learning Lab (MLL), and Cisco.

We thank the anonymous reviewers for their constructive feedback.

References

[1] Anthropic. Claude sonnet 4. https://claude.ai, 2025. Large language model.

[2] Mahmoud Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba Komeili, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, Francois Robert Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, and Nicolas Ballas. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

[3] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[4] Adrien Bardes, Quentin Garrido, Jean Ponce, Michael Rabbat, Yann LeCun, Mahmoud Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video. *arXiv:2404.08471*, 2024.

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

[6] Zihui Cheng, Qiguang Chen, Jin Zhang, Hao Fei, Xiaocheng Feng, Wanxiang Che, Min Li, and Libo Qin. Comt: A novel benchmark for chain of multi-modal thought on large vision-language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 23678–23686, 2025.

[7] Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, Guang Shi, and Haoqi Fan. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025.

[8] Shuai Dong, Siyuan Wang, Xingyu Liu, Chenglin Li, Haowen Hou, and Zhongyu Wei. Interleaved latent visual reasoning with selective perceptual modeling. *arXiv preprint arXiv:2512.05665*, 2025.

[9] Rohan Dugad, Mi Luo, and Kristen Grauman. Watching itself watch: Self-auditing visual reliance for video reasoning. *Advances in Neural Information Processing Systems*, 2026.

[10] Yue Fan, Xuehai He, Diji Yang, Kaizhi Zheng, Ching-Chen Kuo, Yuting Zheng, Sravana Jyothi Narayanaraju, Xinze Guan, and Xin Eric Wang. Grit: Teaching mllms to think with images. *arXiv preprint arXiv:2505.15879*, 2025.

[11] Kaituo Feng, Kaixiong Gong, Bohao Li, Zonghao Guo, Yibing Wang, Tianshuo Peng, Benyou Wang, and Xiangyu Yue. Video-r1: Reinforcing video reasoning in mllms. *arXiv preprint arXiv:2503.21776*, 2025.

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

[13] Sachin Goyal, Ziwei Ji, Ankit Singh Rawat, Aditya Krishna Menon, Sanjiv Kumar, and Vaishnavh Nagarajan. Think before you speak: Training language models with pause tokens. *arXiv preprint arXiv:2310.02226*, 2023.

[14] Jiawei Gu, Yunzhuo Hao, Huichen Will Wang, Linjie Li, Michael Qizhe Shieh, Yejin Choi, Ranjay Krishna, and Yu Cheng. Thinkmorph: Emergent properties in multimodal interleaved chain-of-thought reasoning. *arXiv preprint arXiv:2510.27492*, 2025.

[15] Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. Training large language models to reason in a continuous latent space. *arXiv preprint arXiv:2412.06769*, 2024.

[16] Yunzhuo Hao, Jiawei Gu, Huichen Will Wang, Linjie Li, Zhengyuan Yang, Lijuan Wang, and Yu Cheng. Can mllms reason in multimodality? emma: An enhanced multimodal reasoning benchmark. *arXiv preprint arXiv:2501.05444*, 2025.

[17] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[18] Jack Hong, Chenxiao Zhao, ChengLin Zhu, Weiheng Lu, Guohai Xu, and Xing Yu. Deepeyesv2: Toward agentic multimodal model. *arXiv preprint arXiv:2511.05271*, 2025.

[19] Wenxuan Huang, Bohan Jia, Zijie Zhai, Shaosheng Cao, Zheyu Ye, Fei Zhao, Zhe Xu, Xu Tang, Yao Hu, and Shaohui Lin. Vision-r1: Incentivizing reasoning capability in multimodal large language models. *arXiv preprint arXiv:2503.06749*, 2025.

[20] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[21] Chaoya Jiang, Yongrui Heng, Wei Ye, Han Yang, Haiyang Xu, Ming Yan, Ji Zhang, Fei Huang, and Shikun Zhang. Vlm-r3: Region recognition, reasoning, and refinement for enhanced multimodal chain-of-thought. *arXiv preprint arXiv:2505.16192*, 2025.

[22] Ang Li, Charles Wang, Deqing Fu, Kaiyu Yue, Zikui Cai, Wang Bill Zhu, Ollie Liu, Peng Guo, Willie Neiswanger, Furong Huang, et al. Zebra-cot: A dataset for interleaved vision language reasoning. *arXiv preprint arXiv:2507.16746*, 2025.

[23] Bangzheng Li, Ximeng Sun, Jiang Liu, Ze Wang, Jialian Wu, Xiaodong Yu, Hao Chen, Emad Barsoum, Muhao Chen, and Zicheng Liu. Latent visual reasoning. *arXiv preprint arXiv:2509.24251*, 2025.

[24] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024.

[25] Chengzu Li, Wenshan Wu, Huanyu Zhang, Yan Xia, Shaoguang Mao, Li Dong, Ivan Vulić, and Furu Wei. Imagine while reasoning in space: Multimodal visualization-of-thought. *arXiv preprint arXiv:2501.07542*, 2025.

[26] Dacheng Li, Shiyi Cao, Tyler Griggs, Shu Liu, Xiangxi Mo, Eric Tang, Sumanth Hegde, Kourosh Hakhamaneshi, Shishir G Patil, Matei Zaharia, et al. Llms can easily learn to reason from demonstrations structure, not content, is what matters! *arXiv preprint arXiv:2502.07374*, 2025.

[27] You Li, Chi Chen, Yanghao Li, Fanhu Zeng, Kaiyu Huang, Jinan Xu, and Maosong Sun. Imagination helps visual reasoning, but not yet in latent space. *arXiv preprint arXiv:2602.22766*, 2026.

[28] Jianhua Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, 37(1):145–151, 2002.

[29] Kai Liu, Jungang Li, Yuchong Sun, Shengqiong Wu, Jianzhang Gao, Daoan Zhang, Wei Zhang, Sheng Jin, Sicheng Yu, Geng Zhan, et al. Javisgpt: A unified multi-modal llm for sounding-video comprehension and generation. *arXiv preprint arXiv:2512.22905*, 2025.

[30] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[31] Mi Romy Luo, Zihui Sherry Xue, Alex Dimakis, and Kristen Grauman. When thinking drifts: Evidential grounding for robust video reasoning. *Advances in Neural Information Processing Systems*, 38:83696–83727, 2025.

[32] Bence Nanay. *Mental Imagery*. Oxford University Press, Oxford, 2023.

[33] Siqu Ou, Hongcheng Liu, Pingjie Wang, Yusheng Liao, Chuan Xuan, Yanfeng Wang, and Yu Wang. Bridging the dynamic perception gap: Training-free draft chain-of-thought for dynamic multimodal spatial reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 6560–6578, 2025.

[34] Yiming Qin, Bomin Wei, Jiaxin Ge, Konstantinos Kallidromitis, Stephanie Fu, Trevor Darrell, and XuDong Wang. Chain-of-visual-thought: Teaching vlms to see and think better with continuous visual tokens. *arXiv preprint arXiv:2511.19418*, 2025.

[35] Arijit Ray, Ahmed Abdelkader, Chengzhi Mao, Bryan A. Plummer, Kate Saenko, Ranjay Krishna, Leonidas Guibas, and Wen-Sheng Chu. Mull-tokens: Modality-agnostic latent thinking, 2025.

[36] Arijit Ray, Jiafei Duan, Reuben Tan, Dina Bashkirova, Rose Hendrix, Kiana Ehsani, Aniruddha Kembhavi, Bryan A Plummer, Ranjay Krishna, Kuo-Hao Zeng, et al. Sat: Spatial aptitude training for multimodal language models. *arXiv preprint arXiv:2412.07755*, 3, 2024.

[37] Alan Richardson. *Defining Mental Imagery*, page 1–12. Springer Berlin Heidelberg, 1969.

[38] Zhenyi Shen, Hanqi Yan, Linhai Zhang, Zhanghao Hu, Yali Du, and Yulan He. Codi: Compressing chain-of-thought into continuous space via self-distillation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 677–693, 2025.

[39] Yansong Tang, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. Coin: A large-scale dataset for comprehensive instructional video analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1207–1216, 2019.

[40] Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

[41] Jingqi Tong, Yurong Mou, Hangcheng Li, Mingzhe Li, Yongzhuo Yang, Ming Zhang, Qiguang Chen, Tianyi Liang, Xiaomeng Hu, Yining Zheng, et al. Thinking with video: Video generation as a promising multimodal reasoning paradigm. *arXiv preprint arXiv:2511.04570*, 2025.

[42] Jintao Tong, Jiaqi Gu, Yujing Lou, Lubin Fan, Yixiong Zou, Yue Wu, Jieping Ye, and Ruixuan Li. Sketch-in-latents: Eliciting unified reasoning in mllms. *arXiv preprint arXiv:2512.16584*, 2025.

[43] Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, Austin Wang, Rob Fergus, Yann LeCun, and Saining Xie. Cambrian-1: A fully open, vision-centric exploration of multimodal llms, 2024.

[44] Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal llms, 2024.

[45] Boshi Wang, Sewon Min, Xiang Deng, Jiaming Shen, You Wu, Luke Zettlemoyer, and Huan Sun. Towards understanding chain-of-thought prompting: An empirical study of what matters. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2717–2739, 2023.

[46] Haozhe Wang, Alex Su, Weiming Ren, Fangzhen Lin, and Wenhu Chen. Pixel reasoner: Incentivizing pixel-space reasoning with curiosity-driven reinforcement learning. *arXiv preprint arXiv:2505.15966*, 2025.

[47] Maijunxian Wang, Ruisi Wang, Juyi Lin, Ran Ji, Thaddäus Wiedemer, Qingying Gao, Dezhi Luo, Yaoyao Qian, Lianyu Huang, Zelong Hong, et al. A very big video reasoning suite. *arXiv preprint arXiv:2602.20159*, 2026.

[48] Qixun Wang, Yang Shi, Yifei Wang, Yuanxing Zhang, Pengfei Wan, Kun Gai, Xianghua Ying, and Yisen Wang. Monet: Reasoning in latent visual space beyond images and language. *arXiv preprint arXiv:2511.21395*, 2025.

[49] Ruisi Wang, Zhongang Cai, Fanyi Pu, Junxiang Xu, Wanqi Yin, Maijunxian Wang, Ran Ji, Chenyang Gu, Bo Li, Ziqi Huang, et al. Demystifing video reasoning. *arXiv preprint arXiv:2603.16870*, 2026.

[50] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

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

[52] Thaddäus Wiedemer, Yuxuan Li, Paul Vicol, Shixiang Shane Gu, Nick Matarese, Kevin Swersky, Been Kim, Priyank Jaini, and Robert Geirhos. Video models are zero-shot learners and reasoners. *arXiv preprint arXiv:2509.20328*, 2025.

[53] Penghao Wu and Saining Xie. V*: Guided visual search as a core mechanism in multimodal llms. *arXiv preprint arXiv:2312.14135*, 2023.

[54] Qiucheng Wu, Handong Zhao, Michael Saxon, Trung Bui, William Yang Wang, Yang Zhang, and Shiyu Chang. Vsp: Assessing the dual challenges of perception and reasoning in spatial planning tasks for vlms. *arXiv preprint arXiv:2407.01863*, 2024.

[55] Yecheng Wu, Zhuoyang Zhang, Junyu Chen, Haotian Tang, Dacheng Li, Yunhao Fang, Ligeng Zhu, Enze Xie, Hongxu Yin, Li Yi, et al. Vila-u: a unified foundation model integrating visual understanding and generation. *arXiv preprint arXiv:2409.04429*, 2024.

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

[57] Yi Xu, Chengzu Li, Han Zhou, Xingchen Wan, Caiqi Zhang, Anna Korhonen, and Ivan Vulić. Visual planning: Let's think only with images, 2025.

[58] Zihui Xue, Joungbin An, Xitong Yang, and Kristen Grauman. Progress-aware video frame captioning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 13639–13650, 2025.

[59] Zihui Xue, Kumar Ashutosh, and Kristen Grauman. Learning object state changes in videos: An open-world perspective. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18493–18503, 2024.

[60] Sherry Yang, Jacob Walker, Jack Parker-Holder, Yilun Du, Jake Bruce, Andre Barreto, Pieter Abbeel, and Dale Schuurmans. Video as the new language for real-world decision making. *arXiv preprint arXiv:2402.17139*, 2024.

[61] Zeyuan Yang, Xueyang Yu, Delin Chen, Maohao Shen, and Chuang Gan. Machine mental imagery: Empower multimodal reasoning with latent visual tokens. *arXiv preprint arXiv:2506.17218*, 2025.

[62] Zuhao Yang, Sudong Wang, Kaichen Zhang, Keming Wu, Sicong Leng, Yifan Zhang, Bo Li, Chengwei Qin, Shijian Lu, Xingxuan Li, et al. Longvt: Incentivizing" thinking with long videos" via native tool calling. *arXiv preprint arXiv:2511.20785*, 2025.

[63] Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B Tenenbaum. Clevrer: Collision events for video representation and reasoning. *arXiv preprint arXiv:1910.01442*, 2019.

[64] Haoji Zhang, Xin Gu, Jiawen Li, Chixiang Ma, Sule Bai, Chubin Zhang, Bowen Zhang, Zhichao Zhou, Dongliang He, and Yansong Tang. Thinking with videos: Multimodal tool-augmented reinforcement learning for long video reasoning. *arXiv preprint arXiv:2508.04416*, 2025.

[65] Huanyu Zhang, Wenshan Wu, Chengzu Li, Ning Shang, Yan Xia, Yangyu Huang, Yifan Zhang, Li Dong, Zhang Zhang, Liang Wang, et al. Latent sketchpad: Sketching visual thoughts to elicit multimodal reasoning in mllms. *arXiv preprint arXiv:2510.24514*, 2025.

[66] Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. *arXiv preprint arXiv:2601.18734*, 2026.

[67] Ziwei Zheng, Michael Yang, Jack Hong, Chenxiao Zhao, Guohai Xu, Le Yang, Chao Shen, and Xing Yu. Deepeyes: Incentivizing" thinking with images" via reinforcement learning. *arXiv preprint arXiv:2505.14362*, 2025.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [N/A].
- [N/A] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for [N/A]).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will also be asked to include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While [Yes] is generally preferable to [No], it is perfectly acceptable to answer [No] provided a proper justification is given (e.g., error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering [No] or [N/A] is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly outline the key objectives and outcomes of the research, including the problem being addressed, the proposed approach or methodology, and the significance of the findings.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

Justification: Yes, we discussed the failure case and the limitations of our work.

Guidelines:

- The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A].

   Justification: Not applicable, as this paper does not include any theoretical results.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide all the necessary information to reproduce the main experimental results in the Experiments section.

   Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The code and data will be released.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper includes comprehensive information on key training configurations.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes].

   Justification: For Fig 1, we include error bars. We conducted multi-run evaluations for a subset of the big experiments and observed no significant variance in the results. Due to the high computational cost—approximately 384 H200 GPU hours per run—it was not feasible to perform multiple runs for all experiments. Nonetheless, the stability observed in the sampled multi-run tests supports the reliability of the reported results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Please refer to the Experiment section.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS Code of Ethics and confirm that the research conducted in this paper adheres to its guidelines.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Please refer to the discussion provided in the supplementary material.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [Yes]

Justification: We take responsible release seriously and ensure that any code or evaluation data we release avoids including sensitive content or misuse-enabling functionality. Our work is intended for academic benchmarking and insight into model behavior.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: Any pretrained models referenced are publicly available and subject to their original licenses and usage policies.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: We will provide guideline documentation as part of the release.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This study does not involve any human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A].

    Justification: This study does not involve any human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A].

    Justification: LLM is used only for writing, editing, or formatting purposes.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.

Contents

A Additional Details for the Vision vs. Text Perturbation Study 22

B Benchmark Details 23

C Implementation Details 23

D Additional Perturbation Results 23

E Training and Inference Cost 24

F Why Do Models with MSC Still Improve? 24

G Societal Impact 25

H Additional Qualitative Examples 26

A Additional Details for the Vision vs. Text Perturbation Study

This section provides implementation details for the perturbation probe introduced in Section 3. The goal of the probe is to measure whether generated visual latent blocks causally affect the model's own final-answer probability, relative to the surrounding linguistic reasoning trace.

Fair text-side perturbations. A naive comparison between vision-side and text-side `Drop` is unfair for multi-hop trajectories. A typical latent-reasoning trajectory contains an initial planning segment, alternating visual latent blocks and between-block reasoning text, and a final segment that often explicitly states the answer, such as *"Therefore, the answer is B"*. Vision-side `Drop` removes only the latent hidden states while leaving the entire linguistic scaffold intact. In contrast, naively dropping all text would also remove the answer-stating tail, degrading performance for reasons unrelated to intermediate reasoning.

To make the comparison more balanced, for multi-hop models we define text-side `Drop` as removing only the *between-block reasoning text*, while preserving both the initial planning segment and the final answer-stating tail. This mirrors the asymmetry of vision-side `Drop`, where only intermediate visual content is neutralized. Monet emits a single visual block and does not contain between-block text, so we use the naive full-text `Drop` variant for that model.

Filtering and perturbation conditions. For `Drop` and `Shuffle`, we restrict evaluation to trajectories with at least two visual blocks whenever the perturbation requires a meaningful within-trajectory permutation. This applies to Mirage, SkiLa, and CoVT. Monet emits one visual block per trajectory, so vision-side `Shuffle` is omitted.

We evaluate four models on their corresponding benchmarks with 100 samples per model: Mirage-7B on CoMT, SkiLa-7B on BLINK multi-hop, CoVT-7B on V*Bench, and Monet-7B on MMVP + V*Bench. The perturbation conditions are:

- **Full**: the original captured trajectory is scored without modification.
- **Drop**: the target block content is removed or masked.
- **Cross**: the target block is replaced with the corresponding block from another sample.
- **Shuffle**: the model's own blocks are permuted across positions.

The answer span itself is never modified during scoring. Therefore, changes in final-answer log-probability reflect how much the perturbed component contributes to the model's prediction.

Mirage-7B per-sample statistics. For the Mirage-7B per-sample perturbation analysis in Figure 2, `Drop` and `Shuffle` are restricted to multi-hop trajectories with $n_{\text{latents}} \geq 2$ ($n = 39$), while `Cross` uses all 100 samples. The ordinary-least-squares fit slopes are:

$$\begin{aligned} \texttt{Drop}: &\quad \text{vision} = 0.87, \quad \text{text} = 0.59, \\ \texttt{Cross}: &\quad \text{vision} = 0.94, \quad \text{text} = 0.92, \\ \texttt{Shuffle}: &\quad \text{vision} = 0.93, \quad \text{text} = 0.68. \end{aligned}$$

Across all three perturbations, the text fit has a lower slope than the vision fit, indicating that the linguistic trace carries more answer-relevant signal than the generated visual latents.

B Benchmark Details

We evaluate on public visual reasoning benchmarks spanning perception, spatial reasoning, abstract logic, STEM reasoning, planning, and step-wise manipulation.

Perception and fine-grained visual search. BLINK [12] evaluates 14 core visual perception skills. We report both the overall average and the jigsaw subset, which requires visual completion and fine-grained part-whole reasoning. MMVP [44] contains CLIP-blind image pairs designed to expose perceptual failures in modern vision-language models. V^* [53] evaluates fine-grained visual search in high-resolution images, where correct answers often depend on small or peripheral details.

2D and 3D spatial reasoning. CV-Bench [43] contains 2,638 items derived from COCO, ADE20K, and Omni3D. Its 2D split tests spatial relations and counting, while its 3D split evaluates depth ordering and relative distance. We report these as CV-Bench-2D and CV-Bench-3D. SAT-Real [36] is the real-image split of the Spatial Aptitude Test, containing 300 items across action-consequence, ego-movement, goal-aiming, object-movement, and perspective-taking categories.

Abstract and multimodal STEM reasoning. EMMA [16] is a multimodal STEM benchmark covering chemistry, coding, mathematics, and physics. Solving EMMA problems often requires combining symbolic reasoning with iterative grounding in diagrams. VisuLogic [56] contains 1,000 abstract visual-logic puzzles inspired by Raven's progressive matrices, testing pattern induction over shapes, colors, and transformations.

Visual planning and step-wise manipulation. VSP [54] is a 400-item maze-planning benchmark on 3×3 to 6×6 grids. We evaluate predictions by simulating the generated U/D/L/R plan against the maze and count success only if the agent reaches the goal without stepping into a hole. CoMT [6] is a 400-item Chain-of-Manipulations benchmark that tests step-wise visual editing reasoning over creation, deletion, selection, and update operations.

C Implementation Details

Training and optimization. We use Qwen2.5-VL-7B as the base model. Stage 1 follows the interleaved multimodal warmup recipe used by Mirage, while Stage 2 applies the AnchorSim objective described in Section 4. We optimize with AdamW [30] using a peak learning rate of 1×10^{-6}. Unless otherwise stated, we use a latent block size of 8 tokens and EMA decay rate $\alpha = 0.99$.

Evaluation. All benchmark evaluations use the same prompting and decoding settings across comparable models. For planning-style tasks such as VSP, we evaluate by executing the predicted action sequence in the environment rather than relying on exact string matching. For free-form tasks such as CoMT, predictions are normalized before answer comparison.

D Additional Perturbation Results

This section reports per-sample perturbation plots for additional latent-reasoning baselines. As shown in Figures 8, 9, and 10, the same text-over-vision asymmetry observed for Mirage-7B also appears in

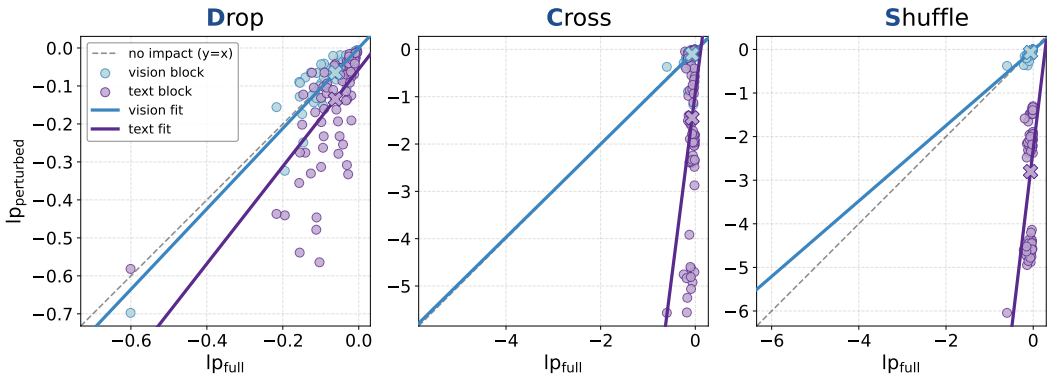


Figure 8: **Per-sample vision vs. text perturbation impact for SkiLa-7B on the multi-hop subset of BLINK.** The layout, axes, and color encoding match Figure 2. Each point compares the perturbed final-answer log-probability against the full-trajectory baseline. `Drop` and `Shuffle` are restricted to trajectories with $n_{\text{sketches}} \geq 2$ ($n = 88$), while `Cross` uses all 100 samples. Across conditions, text perturbations produce larger answer log-probability drops than sketch-hidden-state perturbations, indicating that the linguistic reasoning chain carries more answer-relevant computation than the visual sketch states.

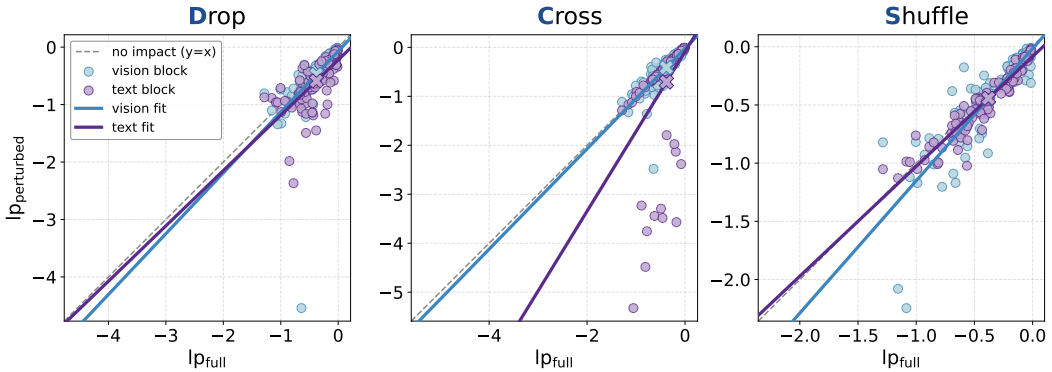


Figure 9: **Per-sample vision vs. text perturbation impact for CoVT-7B on V*Bench.** CoVT emits a fixed three-encoder visual-block template per trajectory, so all perturbation conditions are evaluated on the full 100-sample set. Vision-side perturbations leave many samples close to the no-impact diagonal, whereas text-side perturbations move a substantial subset below the diagonal. This shows that the model's final-answer probability is more sensitive to corrupted reasoning text than to corrupted SAM, depth, and DINO pad hidden states.

SkiLa-7B, CoVT-7B, and Monet-7B. These results complement the aggregate perturbation results in the main text and show that Mental Simulation Collapse is not specific to a single model architecture or benchmark.

E Training and Inference Cost

AnchorSim adds no inference overhead: the visual-blind branch and the EMA teacher are used only during training and are discarded after Stage 2, so AnchorSim-7B has the same architecture, parameter count, and latent format as the Stage 1 model. During training, the two auxiliary passes do not require gradients, so neither requires a backward pass or a separate training run.

F Why Do Models with MSC Still Improve?

We found baselines with MSC can still improve over the base model, plausibly through channels other than the content of the visual latents. Latent blocks add serial computation, like pause tokens [13]; they structure the reasoning trajectory into multi-step segments [26, 45]; and Stage 1 alignment with helper-image embeddings may change how the model attends to the input image. Our diagnostics do

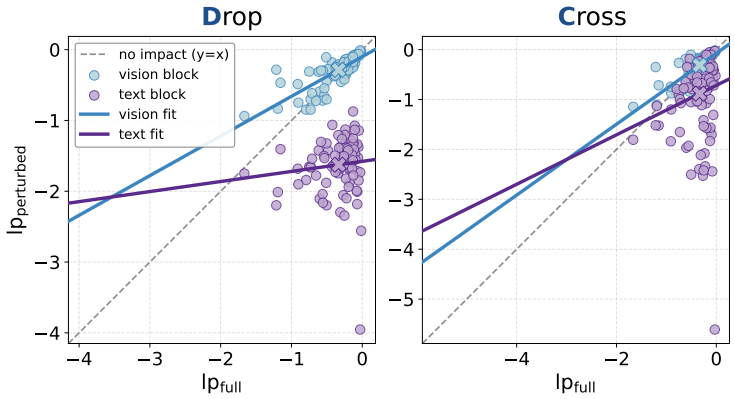


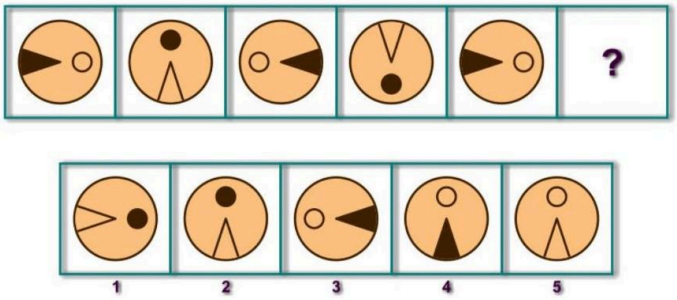
Figure 10: **Per-sample vision vs. text perturbation impact for Monet-7B on MMVP + V*Bench.** Monet emits a single visual latent block per trajectory, so vision-side `Shuffle` is undefined and omitted. `Drop` and `Cross` are evaluated on all 100 samples. Since Monet has no between-block text segments, text-side `Drop` uses the naive full-text variant. Both vision and text channels carry signal in this single-hop setting, but text perturbations cause larger log-probability drops.

not isolate these effects, but all three are consistent with MSC: these models may work in spite of their mental images rather than because of them. AnchorSim preserves these structural benefits while also making the latent content consequential.

G Societal Impact

This work aims to make multimodal reasoning systems more reliable by ensuring that generated visual latents are not merely decorative, but are actually used in downstream prediction. By mitigating *Mental Simulation Collapse*, AnchorSim may improve applications that require faithful reasoning over visual inputs, including assistive technologies, educational tools, scientific diagram understanding, spatial planning, and analysis of complex visual content. More causally grounded visual reasoning may also improve interpretability, since perturbation-based diagnostics can help determine whether a model relies on visual evidence rather than unsupported linguistic priors.

At the same time, stronger multimodal reasoning systems introduce risks. Models that better understand images and visual structure may be misused for privacy-invasive analysis, surveillance, or automated decision-making in sensitive domains. They may also inherit or amplify biases present in training data, especially when visual cues correlate with demographic or social attributes. AnchorSim addresses one failure mode of latent multimodal reasoning, but it does not guarantee fairness, privacy, or safety. Responsible deployment requires careful dataset curation, bias evaluation, privacy safeguards, human oversight, and domain-specific validation before use in high-stakes settings.



Question: Do you see the figures inside these boxes? They form a pattern. Choose the figure in the answer row below that continues the pattern. A. 1 B. 2 C. 3 D. 4 E. 5

AnchorSim-7B: THOUGHT 0: The problem presents a sequence of images, each containing a circle with an inner shape. I need to identify the pattern governing these shapes to find the missing image.

THOUGHT 1: Let's analyze the pattern of the inner shapes. The inner shapes in the first column are a triangle, a circle, and a triangle. The inner shapes in the second column are a circle, a triangle, and a circle. This suggests the inner shape in the missing panel should be a circle.

THOUGHT 2: Now, let's examine the outer shapes. The outer shapes in the first column are a circle, a circle, and a circle. The outer shapes in the second column are a circle, a circle, and a circle. This indicates the outer shape in the missing panel should also be a circle.

THOUGHT 3: Combining the patterns, the missing image should have an outer circle and an inner circle. Comparing this with the answer options, option 2 matches this description. The answer is B.

SkiLa-7B: The problem presents a sequence of images, each composed of an outer orange circle and an inner shape. I need to identify the pattern governing these shapes to determine the missing image. To discern the pattern, I will analyze the sequence of images, focusing on the outer orange circle and the inner shapes. I will generate a visual representation to highlight the progression of the inner shapes.

The generated image shows the progression of the inner shapes. The pattern indicates that the inner shape rotates clockwise by 90 degrees with each step. The missing image should have the inner shape rotated 90 degrees clockwise from the last image in the sequence. Based on the identified pattern, the inner shape in the missing image should be a triangle pointing upwards. This corresponds to option 5.

visual latent token (not decoded)

Figure 11: **Qualitative example on EMMA.** The model solves a diagram-grounded STEM problem by combining textual reasoning with visual grounding. AnchorSim encourages the reasoning chain to revisit visual latents when deriving the final answer, rather than relying only on language priors.

H Additional Qualitative Examples

We provide additional qualitative examples to illustrate how AnchorSim behaves across different types of visual reasoning tasks. Figure 11 shows a diagram-grounded STEM example from EMMA, where the model must combine symbolic reasoning with visual evidence. Figure 12 demonstrates spatial mental simulation on SAT-Real, while Figure 13 highlights fine-grained visual search on V^*. Figure 14 presents a low-level visual discrimination case from MMVP, and Figure 15 shows 2D spatial-relation reasoning on CV-Bench. Finally, Figure 16 illustrates a multi-hop reasoning trajectory in which intermediate latent visual states support later textual reasoning before the final answer.



Question: I need to go to the houseplant. if i turn right by 100 degrees, will i facing away from it?
A. yes B. no

AnchorSim-7B: THOUGHT 0: To answer the question, I need to locate the houseplant in the image and determine its position relative to the pool table, which serves as the reference point for the 100-degree turn.

THOUGHT 1: I will highlight the houseplant with a bounding box to clearly identify its location within the room.

THOUGHT 2: The red bounding box clearly outlines the houseplant on the left side of the image. The pool table is visible on the right.
THOUGHT 3: If I start at the pool table and turn 100 degrees to the right, I will be facing the wall directly opposite the houseplant. Therefore, I would be facing away from it.

The final answer is A.

SkiLa-7B: To determine if turning right by 100 degrees will face away from the houseplant, I need to identify the houseplant's location and the direction of the turn. I will visualize the houseplant's position relative to the room's layout and the proposed turn.

The visual thought shows the houseplant on the left side of the room. If I turn right by 100 degrees from the current orientation, I will be facing towards the back wall, which is perpendicular to the houseplant's location. Therefore, I will not be facing away from the houseplant. Answer is B.

visual latent token (not decoded)

Figure 12: **Qualitative example on SAT-R.** The task requires mentally simulating a spatial or viewpoint transformation from a real image. The example illustrates how latent visual states can support intermediate spatial reasoning before the model commits to the final answer.



Question: What is the color of the motorcycle? A. orange B. red C. white. D. black ✓

AnchorSim-7B: THOUGHT 0: To determine the motorcycle's color, I need to locate it within the provided image and examine its details.

THOUGHT 1: The motorcycle is small in the overall scene. To accurately identify its color, I will generate a zoomed-in view of the area where it is located.

□□□□□□□□

THOUGHT 2: The zoomed-in image clearly shows the motorcycle. Its body is predominantly black, with some red and silver accents.

THOUGHT 3: Based on the visual evidence from the detailed view, the primary color of the motorcycle is black.

Answer: D. black ✓

SkiLa-7B: The question asks for the color of a motorcycle visible in the image. I need to locate the motorcycle and identify its color. To accurately determine the motorcycle's color, I will focus on the lower left portion of the image where the vehicles are parked. A closer view will help.

□□□□□□□□

The zoomed-in image clearly shows a motorcycle parked next to a red car. Its color is distinctly visible. Based on the visual evidence from the detailed view, the motorcycle is red. Answer is B. red. ✗

□ visual latent token (not decoded)

Figure 13: **Qualitative example on V*.** The question depends on fine-grained visual search over a high-resolution image. AnchorSim helps the model preserve visually relevant details in latent form, allowing the final response to be grounded in localized evidence.



Question: In the image, there is one daisy that is under the shadow of a taller daisy.

A. Correct B. Incorrect

AnchorSim-7B: THOUGHT 0: I need to examine the image to identify any daisies and determine if any are casting a shadow on another.

THOUGHT 1: To accurately assess the shadows, I will focus on the central part of the image where the flowers are most prominent.

THOUGHT 2: The zoomed-in view clearly shows two daisies. Neither flower appears to be casting a shadow onto the other.

THOUGHT 3: Based on this observation, the statement that there is one daisy under the shadow of another is incorrect.

ANSWER: B. Incorrect

SkiLa-7B: The problem asks to determine if there is a daisy in the image that is in shadow. I need to examine the image for any daisies that appear to be partially obscured or in shadow. To clearly identify the daisies and their relative positions, I will focus on the central part of the image where the flowers are most prominent.

The zoomed-in view clearly shows two daisies. One daisy is positioned behind the other, and its petals are slightly darker, indicating it is in shadow. Based on this observation, there is indeed a daisy that is under the shadow of another daisy. Answer is A. Correct.

visual latent token (not decoded)

Figure 14: **Qualitative example on MMVP.** MMVP contains image pairs that expose failures in low-level visual discrimination. This example shows how anchoring the reasoning process to visual latents can help distinguish visually similar alternatives.



Question: How many televisions are in the image? A. 3 B. 1 ✓ C. 2 D. 0

AnchorSim-7B: THOUGHT 0: I need to examine the provided image to locate any televisions present.

THOUGHT 1: To clearly identify the object in question, I will highlight the potential television with a bounding box.



THOUGHT 2: The red bounding box clearly outlines a framed object on the left side of the window, which is identifiable as a television.

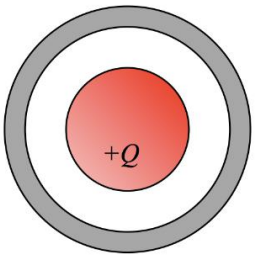
THOUGHT 3: Based on the visual evidence, there is only one television visible in the image.

THOUGHT 4: The image contains a single television, which is a framed picture on the wall.
Final Answer: B ✓

CoVT-7B: Because the segmentation of the image is , the depth map of the image is , the perception feature of the image is , and the edge map of the image is . Answer is D. ✗

visual latent token (not decoded)

Figure 15: **Qualitative example on CV-Bench-2D.** The model must reason about planar spatial relations and object layout. The generated latent states provide compact visual anchors that support relation tracking throughout the reasoning trajectory.



Question: A non-conducting sphere of uniformly-distributed charge is surrounded by a conducting shell with no net charge. Which of the following statements is false?
A. There is an electric field inside the non-conducting sphere.
B. There is an electric field in the space between the sphere and the shell.
C. There is an electric field above the outer surface of the conducting shell.
D. All points in the conducting shell are at the same electric potential.
E. All points in the non-conducting sphere are at the same potential.

AnchorSim-7B: THOUGHT 0: The problem asks to identify the statement that is false regarding the electric field and potential in the given system. The system consists of a positively charged non-conducting sphere and a concentric conducting shell. I need to analyze the electric field and potential in different regions of this system.

THOUGHT 1: To understand the electric field distribution, I will first visualize the electric field lines. The problem states the non-conducting sphere has a positive charge, so I will sketch the electric field lines originating from its surface.

THOUGHT 2: The sketch shows that the electric field lines originate from the surface of the non-conducting sphere. Since the conducting shell is concentric and has no net charge, it will not create any electric field lines that originate from the sphere. The shell will only affect the electric field lines that pass through it.

THOUGHT 3: Next, I will visualize the electric field lines within the conducting shell. ... I will sketch this.

THOUGHT 4: The second sketch confirms that ... any external electric field. The electric field lines are continuous and form closed loops within the shell. THOUGHT 5: Now, I will visualize the ... but inside the outer shell. I will sketch the electric field lines in this region.

THOUGHT 6: The third sketch shows that ... This confirms that ... THOUGHT 7: Finally, I will visualize the electric field lines ... will be perpendicular to it. I will sketch this.

THOUGHT 8: The fourth sketch shows that ... THOUGHT 9: Based on the visual analysis .... Answer is E.

visual latent token (not decoded)

Figure 16: **Qualitative example of multi-hop visual reasoning.** The model decomposes the problem into multiple intermediate reasoning steps, each paired with latent visual states. This example illustrates the intended behavior of interleaved visual reasoning: later text should depend on earlier visual simulations rather than bypassing them.