

Seeing but Not Thinking: Routing Distraction in Multimodal Mixture-of-Experts

Haolei Xu^{1,2*}, Haiwen Hong^{2*†}, Hongxing Li^{1,2}, Rui Zhou¹, Yang Zhang¹,
Longtao Huang², Hui Xue², Yongliang Shen^{1‡}, Weiming Lu^{1‡}, Yueting Zhuang¹

¹Zhejiang University, ²Alibaba Group
{xuhaolei, syl, luwm}@zju.edu.cn, honghaiwen.hhw@alibaba-inc.com

Abstract

Multimodal Mixture-of-Experts (MoE) models have achieved remarkable performance on vision-language tasks. However, we identify a puzzling phenomenon termed *Seeing but Not Thinking*: models accurately perceive image content yet fail in subsequent reasoning, while correctly solving identical problems presented as pure text. Through systematic analysis, we first verify that cross-modal semantic sharing exists in MoE architectures, ruling out semantic alignment failure as the sole explanation. We then reveal that visual experts and domain experts exhibit layer-wise separation, with image inputs inducing significant routing divergence from text inputs in middle layers where domain experts concentrate. Based on these findings, we propose the *Routing Distraction* hypothesis: when processing visual inputs, the routing mechanism fails to adequately activate task-relevant reasoning experts. To validate this hypothesis, we design a routing-guided intervention method that enhances domain expert activation. Experiments on three multimodal MoE models across six benchmarks demonstrate consistent improvements, with gains of up to 3.17% on complex visual reasoning tasks. Our analysis further reveals that domain expert identification locates cognitive functions rather than sample-specific solutions, enabling effective transfer across tasks with different information structures.

1 Introduction

Mixture-of-Experts (MoE) architectures (Cai et al., 2025) have become the dominant paradigm for scaling large vision-language models (Lin et al., 2024; Wang et al., 2025b; Kuang et al., 2025; Tang et al., 2025; Ding et al., 2026), powering a wide range of downstream multimodal applications (Lu et al., 2026a, 2025). By activating only a sparse subset of

* Equal Contribution.

† Project Leader.

‡ Corresponding Author.

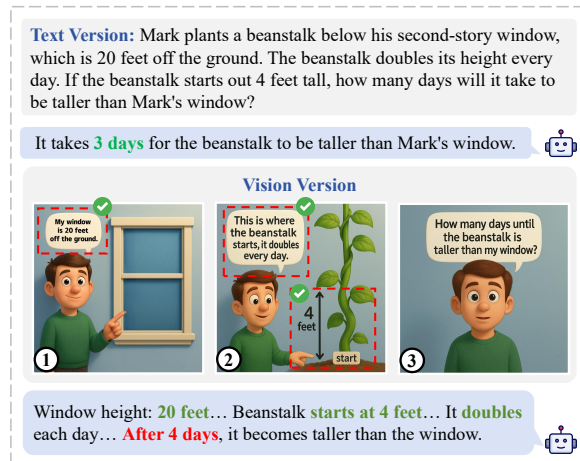


Figure 1: Illustration of the *Seeing but Not Thinking* phenomenon. See Appendix B for details.

experts for each input, MoE models efficiently handle the intricate interactions between visual and textual information while maintaining computational tractability. However, beneath this success lies a puzzling phenomenon that challenges our fundamental understanding of how these models integrate perception and reasoning.

Consider a simple scenario illustrated in Figure 1 and Appendix B: when presented with a grade-school mathematics problem (Cobbe et al., 2021) as an image (Yuan et al., 2025), Qwen3-VL-30B-A3B (Bai et al., 2025) accurately extracts all numerical values and textual content, yet produces an incorrect answer due to reasoning errors. When the identical problem is presented as pure text, the same model solves it correctly with ease. We term this phenomenon *Seeing but Not Thinking*: the model perceives visual content accurately but fails to reason correctly, despite possessing the requisite capability on semantically equivalent text inputs. This raises a fundamental question: *what factor causes multimodal MoE models to fail at reasoning when visual inputs are correctly perceived?*

To systematically quantify this phenomenon

while minimizing interference from perceptual errors, we construct rigorously controlled experiments based on the MATH500 (Hendrycks et al., 2021) dataset. We render all pure-text problems as high-resolution images (detailed in Appendix C) to ensure visual inputs are clear and legible. To pinpoint the source of these failures, we conduct error analysis on samples that succeed on text but fail on images. The results are striking: 68.2% to 73.1% of failures stem from reasoning errors, while only 26.9% to 31.8% are attributable to perception errors (Table 6). This confirms that visual inputs degrade reasoning performance even when the content is correctly perceived. This finding aligns with conclusions from recent benchmarks (Zhang et al., 2024b; van Sprang et al., 2025).

A natural hypothesis is cross-modal semantic alignment failure: visual information, though correctly perceived, may fail to align with the textual semantic space at the representation level. Prior work has demonstrated that dense-architecture VLMs achieve cross-modal semantic sharing (Wu et al., 2024; Shukor and Cord, 2024), but whether MoE-based VLMs possess the same property remains unexplored. We design cross-modal concept intervention experiments that manipulate hidden state representations across modalities. Our results reveal a clear inverted U-shaped pattern across layers: intervention success rates are low in early layers where visual features have not yet aligned, peak in middle layers where semantic sharing occurs, and decline in terminal layers where output distributions are already determined. This finding confirms that MoE architectures also exhibit cross-modal semantic sharing, indicating that semantic alignment failure alone cannot account for the observed reasoning degradation.

If semantic alignment is not the primary bottleneck, what other factors might contribute? We examine the routing mechanism, the core component distinguishing MoE from dense models. Through systematic analysis of expert activation patterns, we uncover two critical findings. First, experts exhibiting high activation on visual inputs concentrate in early and terminal layers, while domain-specific reasoning experts cluster in middle layers. Second, image inputs induce significant routing divergence from text inputs precisely in these middle layers. Crucially, greater routing divergence correlates with lower reasoning accuracy across our controlled conditions. These observations lead us to propose the *Routing Distraction* hypothesis: when

processing visual inputs, the routing mechanism fails to adequately activate task-relevant reasoning experts, instead directing computation toward less suitable experts. The visual modality does not impair intrinsic reasoning capabilities; rather, it causes suboptimal expert selection, preventing full utilization of domain-specific reasoning capacity.

To validate this hypothesis, we design a routing-guided intervention that enhances domain expert activation during inference. The core idea is straightforward: if routing distraction significantly contributes to reasoning failures, then explicitly increasing the activation weights of domain-relevant experts should recover reasoning performance. We evaluate on three MoE models across six benchmarks spanning semantically equivalent and natural visual scenarios. The soft routing guidance strategy yields consistent improvements. Even on tasks involving complex geometric figures and function graphs where visual information cannot be replaced by text, enhancing domain expert activation helps models better integrate perception with reasoning, with gains of up to 3.17%. Further analysis reveals that domain expert identification is robust to information completeness of text references: as long as the reference elicits target domain reasoning patterns, identified experts transfer effectively to visual tasks with different information structures. This suggests that expert identification locates computational units responsible for cognitive functions, rather than memorizing sample-specific solutions.

Our contributions are threefold. First, we systematically characterize the *Seeing but Not Thinking* phenomenon and demonstrate its prevalence across multiple state-of-the-art multimodal MoE models. Second, we provide mechanistic insights into this phenomenon through the *Routing Distraction* hypothesis, revealing the layer-wise separation between visual and domain experts and the routing divergence induced by visual inputs. Third, we propose and validate a routing-guided intervention method that effectively mitigates routing distraction, achieving consistent improvements across diverse benchmarks and model scales.

2 Related Work

Multimodal Semantic Sharing Vision-language models achieve cross-modal understanding by connecting visual encoders with large language models. Although image and text embeddings exhibit separated distributions in the shared space (Liang

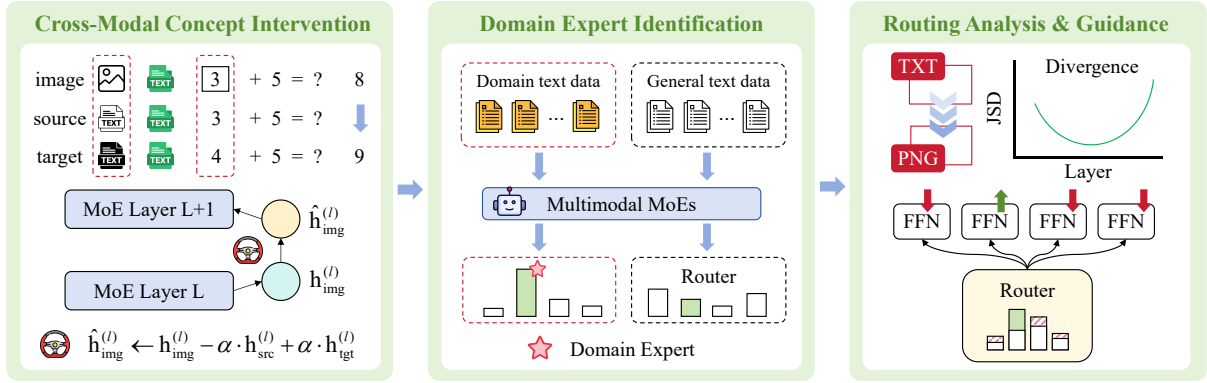


Figure 2: **Overview of our work.** We first conduct cross-modal concept intervention to verify semantic sharing in MoE architectures (left, §3.1), then identify domain experts by comparing activation frequencies on domain-specific versus general data (middle, §3.2), and finally analyze routing divergence across modalities and apply routing guidance to enhance domain expert activation (right, §3.3-§4).

et al., 2022; Schrodi et al., 2024), semantic sharing arise at deeper representation levels. Image representations can transfer to frozen language models through a single linear projection (Merullo et al., 2022), and representations from different modalities may converge toward a shared statistical model (Huh et al., 2024). In VLMs specifically, visual and text tokens activate similar LLM weights despite being distinct in representation space (Shukor and Cord, 2024), and semantically equivalent text-image inputs can be aligned into modality-invariant task vectors (Luo et al., 2024). More broadly, shared semantic spaces have been observed across diverse modalities and languages (Wendler et al., 2024; Bandarkar et al., 2025; Wu et al., 2024).

MoE Expert Specialization Mixture-of-Experts models scale effectively through sparse activation. Early MoE models such as OpenMoE (Xue et al., 2024) did not exhibit significant domain specialization due to the design of their load-balancing objectives. In modern MoE architectures, improved training strategies make domain specialization an expected outcome (see Appendix F). This specialization manifests at multiple granularities: expert differentiation intensifies with layer depth (Lo et al., 2025), cognitive experts control meta-level reasoning operations (Wang et al., 2025a), and safety-related refusal behavior concentrates in a small subset of experts (Lai et al., 2025).

Routing Intervention Inference-time routing intervention (Wu et al., 2025; Ding et al., 2025) has emerged as a promising direction. R2-T2 refines expert selection by shifting routing weights toward correctly predicted samples (Li et al., 2025). SC-

MoE enhances reasoning by contrasting output distributions between selected and unselected experts (Shi et al., 2024). SteerMoE identifies key experts through routing differences and adjusts routing logits for lightweight behavior control (Fayyaz et al., 2025). Dynamic routing mechanisms can also enable complex tasks to activate more experts (Huang et al., 2024).

3 Analyzing Routing Distraction

We systematically probe the mechanisms underlying the *Seeing but Not Thinking* phenomenon. Our analysis proceeds in three stages: we first verify that cross-modal semantic sharing exists in MoE architectures (§3.1), then examine the spatial distribution of specialized experts across layers (§3.2), and finally characterize routing behavior differences across modalities (§3.3). All experiments are conducted on Qwen3-VL-30B-A3B using controlled datasets that minimize perceptual confounds.

3.1 Cross-Modal Semantic Sharing in MoE

Experimental Design. To examine whether cross-modal semantic sharing exists in MoE architectures, we design a concept intervention experiment that directly alters hidden states across modalities. We construct an arithmetic completion task where the input consists of a digit image followed by a textual arithmetic expression (e.g., an image of "3" followed by "+ 2 ="). We extract hidden state vectors for the source digit S and target digit T from pure text inputs, denoted as $\mathbf{h}_{src}^{(l)}$ and $\mathbf{h}_{tgt}^{(l)}$ at layer l . We then perform the following

intervention on the hidden states of image tokens:

$$\hat{\mathbf{h}}_{img}^{(l)} \leftarrow \mathbf{h}_{img}^{(l)} - \alpha \cdot \mathbf{h}_{src}^{(l)} + \alpha \cdot \mathbf{h}_{tgt}^{(l)} \quad (1)$$

where $\alpha = 1$ controls intervention strength. This operation removes the source concept’s semantic vector from the image representation while adding the target concept’s vector. If the model’s output changes to match the answer for the target digit, the intervention is deemed successful. We randomly generate 100 test instances with simple digit images to ensure perception is not a confounding factor.

Results. Figure 3(a) shows intervention success rates across layers. The results exhibit a clear inverted U-shaped pattern: early layers show low success rates, indicating that visual features have not yet aligned with the textual semantic space; middle layers (8–42) show significantly elevated success rates exceeding 90%, suggesting that the two modalities achieve substantial semantic sharing in this region; terminal layers show a sharp decline in intervention effectiveness, likely because the model has already committed to its output distribution. This pattern confirms that MoE-based vision-language models exhibit cross-modal semantic sharing in middle layers, consistent with findings in dense architectures (Wu et al., 2024; Shukor and Cord, 2024). This result rules out semantic alignment failure as the sole explanation for *Seeing but Not Thinking*, motivating us to explore other factors specific to MoE architectures.

3.2 Layer-wise Expert Specialization

Having established that semantic alignment is preserved across modalities, we now investigate another distinctive aspect of MoE architectures: expert specialization. We examine how different types of experts are distributed across layers and whether this distribution reveals structural patterns relevant to the observed phenomenon.

Quantifying Expert Specialization. We first measure the degree of expert specialization at each layer using the Gini coefficient. For a sequence of length L , let $\mathbf{p}_{l,t}$ denote the expert probability vector for the t -th token at layer l . The average expert importance at layer l is $\mathbf{q}_l = \frac{1}{L} \sum_{t=1}^L \mathbf{p}_{l,t}$, with $q_{l,i}$ representing the importance of expert i . The Gini coefficient is computed as:

$$G_l = \frac{\sum_{i=1}^E \sum_{j=1}^E |q_{l,i} - q_{l,j}|}{2E \sum_{k=1}^E q_{l,k}} \quad (2)$$

where E is the total number of experts. Higher values indicate greater concentration of computation among fewer experts. As shown in Figure 3(b), early layers exhibit lower Gini coefficients while middle and terminal layers show elevated values, indicating that expert functional specialization intensifies in deeper layers.

Identifying Domain Experts. To locate experts responsible for domain-specific reasoning, we compare activation frequencies between domain data and general data. Define the Top-K activation frequency of expert $E_{l,i}$ on dataset D as:

$$\Phi(E_{l,i}, D) = \frac{1}{N_D} \sum_{t \in D} \mathbf{1}[E_{l,i} \in \text{TopK}(x_t)] \quad (3)$$

where N_D is the total token count. The frequency difference $\Delta\Phi_{l,i} = \Phi(E_{l,i}, D_{\text{dom}}) - \Phi(E_{l,i}, D_{\text{gen}})$ captures domain-specific activation patterns. Experts with $\Delta\Phi_{l,i} > \tau$ are set as domain experts. Using GSM8K as domain data and Alpaca as general data with $\tau = 0.3$, we find that math experts cluster predominantly in middle layers (Figure 4).

Identifying Visual Experts. To locate experts associated with visual processing, we render Alpaca text as images and compute frequency differences between the image and text versions. With $\tau = 0.2$ (we also tried $\tau = 0.3$ but found too few experts), we identify visual experts that are preferentially activated for image inputs. These experts concentrate in early and terminal layers, exhibiting minimal overlap with math experts in middle layers.

Key Finding. This analysis reveals a critical structural trait: visual experts and domain experts exhibit layer-wise separation. Visual experts cluster in early layers (for initial visual encoding) and terminal layers (preparing modality-specific outputs), while domain experts concentrate in middle layers where cross-modal semantic sharing occurs (§3.1). This spatial segregation raises a natural question: when processing visual inputs, does the routing mechanism in middle layers adequately activate the domain experts necessary for reasoning? We investigate this question in the following section.

3.3 Routing Divergence Across Modalities

The layer separation between visual and domain experts implies routing behavior in middle layers may be critical for reasoning performance. We now directly examine how expert activation patterns differ between image and text inputs, and whether such differences correlate with reasoning degradation.

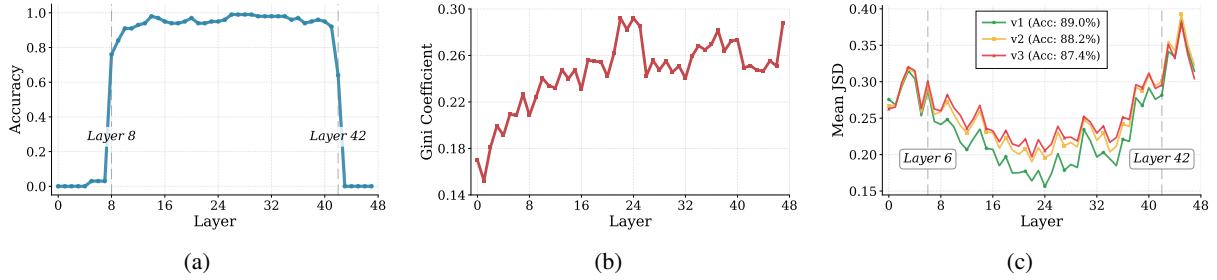


Figure 3: Analysis of routing mechanisms in multimodal MoE models. (a) Cross-modal semantic sharing verification showing inverted U-shaped intervention success rates. (b) Expert specialization quantification using Gini coefficients. (c) Routing divergence across modalities for three image versions.

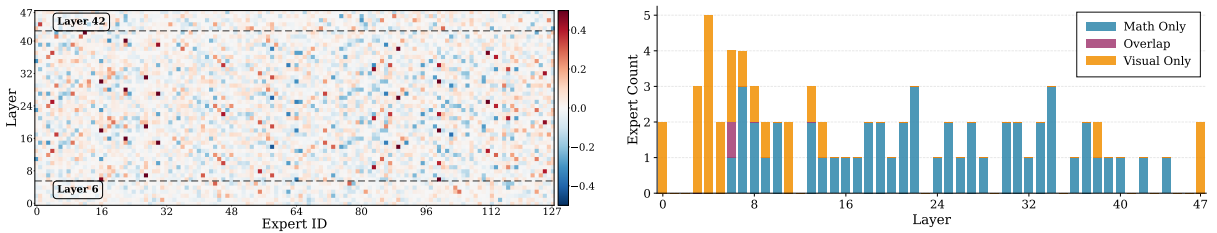


Figure 4: Layer-wise distribution of domain experts and visual experts. Left: Heatmap showing activation frequency differences (red: higher on math data; blue: higher on general data), with deep red concentrated in layers 6–42. Right: Expert counts per layer, where Overlap indicates experts identified as both math and visual experts.

Experimental Setup. We conduct analysis on the MATH500 dataset with semantically equivalent text and image versions. Three image versions (v1/v2/v3) are constructed with increasing visual complexity. Error analysis confirms that reasoning errors rather than perception errors dominate these failures (Table 6), ensuring strict control over perceptual factors. To explicitly measure how visual inputs alter expert selection compared to text inputs for the same problem, we calculate the divergence at the sample level. Let $\Phi_l(x)$ denote the expert activation frequency distribution at layer l for input x . We quantify the average routing divergence as:

$$Div_l = \frac{1}{N} \sum_{i=1}^N JSD(\Phi_l(x_i^{txt}), \Phi_l(x_i^{img})) \quad (4)$$

where N is the total sample count, and $JSD(\cdot)$ represents the Jensen-Shannon Divergence. JSD is computed over prompt-phase tokens only, as generation-phase tokens are exclusively textual.

Results. Figure 3(c) shows JSD across layers for the three image versions. Two patterns emerge. First, JSD exhibits a U-shaped distribution: early and terminal layers show larger divergence (expected due to visual encoding and output preparation), while middle layers show smaller divergence. Second, and more critically, the three curves diverge primarily in the middle layer (6–42) region

while remaining nearly identical in early and terminal layers. This indicates that visual complexity predominantly affects routing behavior in middle layers, precisely where domain experts concentrate.

Correlation with Reasoning Accuracy. Despite similar perception error rates, the three image versions exhibit different reasoning performance: v1/v2/v3 achieve 89.0%/88.2%/87.4% respectively, versus 92.8% for pure text. Notably, versions with lower reasoning accuracy exhibit greater JSD in middle layers. This correlation suggests that routing divergence in middle layers, rather than perceptual quality, contributes to reasoning degradation.

The Routing Distraction Hypothesis. Synthesizing the findings from §3.1–§3.3, we propose the Routing Distraction hypothesis: when processing visual inputs, the MoE routing mechanism fails to adequately activate task-relevant domain experts in middle layers, instead directing computation toward other experts less suited for the reasoning task. This hypothesis explains why models can perceive correctly yet reason incorrectly: semantic alignment is preserved, but the computational resources required for reasoning are not fully mobilized. While our controlled analysis focuses on *semantically equivalent scenarios* where perception is not a confounding factor, we expect this

hypothesis to generalize to *natural visual scenarios* involving complex figures and diagrams. In the following sections, we validate this hypothesis across both scenario types.

4 Routing-Guided Intervention

Based on the Routing Distraction hypothesis, we propose a simple intervention strategy: if insufficient activation of domain experts contributes to reasoning failures, then explicitly enhancing their routing weights should recover performance.

4.1 Domain Expert Identification

Following the method in §3.2, we find domain experts by comparing activation rates between domain-specific data and general-purpose data (Alpaca). This requires constructing text references that elicit the target domain’s reasoning patterns.

Text reference construction depends on the scenario type. For semantically equivalent scenarios, we directly use the original text problems before rendering. For natural visual scenarios, we adopt task-appropriate proxies that elicit similar domain reasoning patterns, such as text-only problem versions or model-generated descriptions (§5.1).

Given text references, we compute $\Delta\Phi_{l,i}$ for each expert and apply threshold τ to obtain the domain expert set $\mathcal{E}_{\text{domain}}$. This procedure requires only 20 randomly sampled examples. Notably, the model need not solve these samples, nor must the text reference be strictly equivalent to the visual task. We analyze this robustness property in §5.3.

4.2 Routing Weight Adjustment

During inference, we enhance routing weights of identified domain experts. We investigate two strategies with a random baseline as control.

Soft Intervention. We apply moderate additive enhancement to domain expert logits:

$$r'_{l,k} \leftarrow r_{l,k} + \lambda \cdot s(r_l), \quad \forall E_{l,k} \in \mathcal{E}_{\text{domain}} \quad (5)$$

where $r_{l,k}$ is the original routing logit, $s(r_l)$ is the standard deviation of all expert logits at layer l , and λ is the enhancement coefficient. This formulation preserves the router’s flexibility to adjust based on specific inputs while systematically increasing domain expert activation probability.

Hard Intervention. We force domain expert logits to the layer maximum:

$$r'_{l,k} \leftarrow \max_j r_{l,j} + \delta, \quad \forall E_{l,k} \in \mathcal{E}_{\text{domain}} \quad (6)$$

where $\delta \sim \mathcal{N}(0, 10^{-4})$ introduces small perturbations to prevent identical logits, particularly relevant for architectures with Top-1 routing.

Random Baseline. To verify that improvements stem from activating specific domain experts rather than routing perturbation itself, we randomly select the same number of experts at each layer and apply identical enhancement as Soft intervention.

5 Experiments

5.1 Experimental Setup

Models. We evaluate three multimodal MoE models spanning different scales: Qwen3-VL-30B-A3B (Bai et al., 2025), Kimi-VL-16B-A3B (Team et al., 2025), and Llama4-Scout-109B-A17B (Meta AI, 2025). Detailed model architectures are provided in Appendix D.

Benchmarks. We construct two complementary evaluation scenarios. *Semantically Equivalent Scenarios* include MATH500 and GPQA-Diamond (chemistry, physics subsets) (Rein et al., 2024) rendered as high-resolution images, providing controlled environments where perception is not a confounding factor. *Natural Visual Scenarios* include MathVerse (vision-only version) (Zhang et al., 2024a), MATH-Vision (Wang et al., 2024), and GSM8K-V (Yuan et al., 2025), with text references constructed from the official text-only version for MathVerse, model-generated problem descriptions for MATH-Vision (Appendix E.2), and corresponding GSM8K text problems for GSM8K-V.

Implementation. We use vLLM (Kwon et al., 2023) with EasySteer (Xu et al., 2025) for inference-time intervention. Intervention configurations are summarized in Table 1. All models use $\tau = 0.3$ for domain expert identification. Intervention layers are selected based on findings from §3.2: middle layers for Qwen3-VL and Llama4-Scout, early and middle layers for Kimi-VL where domain experts emerge earlier. Due to slight non-determinism in vLLM, we run 16 trials with greedy decoding and report average accuracy verified by xVerify (Chen et al., 2025). Additional details are provided in Appendix G.

5.2 Main Results

Table 2 presents reasoning accuracy across all models and benchmarks. We analyze the results from three perspectives.

Model	Intervention Layers	τ	λ
Kimi-VL-16B-A3B	[0, 20]	0.3	0.5
Qwen3-VL-30B-A3B	[6, 42]	0.3	0.5
Llama4-Scout-109B-A17B	[8, 40]	0.3	0.2

Table 1: Intervention configurations for each model. Intervention layers are specified as inclusive ranges.

Effectiveness in Semantically Equivalent Scenarios. These setup directly test the Routing Distraction hypothesis by isolating reasoning from perception. The Soft intervention achieves consistent improvements across all models. Kimi-VL improves from 52.30% to 54.54% (+2.24%) on math and from 29.51% to 32.49% (+2.98%) on physics. Qwen3-VL shows gains of 1.22% on math and 2.15% on chemistry. These results confirm that enhancing domain expert activation effectively mitigates routing distraction when perception is fixed.

Generalization to Natural Visual Scenarios. A key question is whether routing distraction extends to scenarios with complex visual elements. Kimi-VL rises from 35.41% to 38.58% (+3.17%) on MathVerse; Qwen3-VL improves from 55.92% to 57.57% (+1.65%) on MATH-Vision. These gains indicate enhancing domain expert activation helps models better integrate visual information. Improvements on GSM8K-V are modest, suggesting that when task difficulty concentrates in the perception phase (extracting information from multiple scene images), routing guidance offers less benefit.

Comparison of Intervention Strategies. Soft intervention performs most consistently, balancing enhanced domain expert activation with preserved routing flexibility. Hard intervention shows mixed results: while it sometimes yields improvements (e.g., Kimi-VL on physics), it causes severe degradation on Llama4-Scout, even underperforming the Random baseline. We attribute this to Llama4’s Top-1 routing mechanism, where forcing expert logits throughout generation frequently disrupts the original routing decisions. The Random baseline generally produces no improvement, confirming that gains stem from activating correct domain experts rather than routing perturbation.

5.3 Analysis

Robustness of Expert Identification. A critical question is whether domain expert identification demands semantically equivalent text refer-

ences. This is particularly relevant for natural visual benchmarks where perfect text equivalents do not exist. Taking MathVerse as an example, its official text-only version cannot fully convey spatial relationships and geometric configurations, rendering many problems unsolvable from text alone. Indeed, Qwen3-VL achieves 67.26% accuracy on text-only, while achieving 69.29% on vision-only, indicating that visual information provides independent value.

However, experts identified using this partial text reference still yield substantial improvements: vision-only accuracy rises to 71.20% (+1.91%) with Soft intervention. This result carries two implications. First, visual information provides unique value that text misses; routing guidance helps models leverage this information rather than regressing to text-based reasoning paths. Second, expert identification is robust to information completeness. The procedure locates computational units responsible for domain-specific cognitive functions, not sample-specific solution paths. As long as the text reference elicits target domain reasoning patterns, identified experts transfer effectively to visual tasks with different information structures.

Domain Specificity. While expert identification is robust to information completeness, it remains sensitive to domain mismatch. When using GSM8K (elementary arithmetic) instead of MathVerse text-only as the reference for MathVerse evaluation, improvements diminish substantially or even reverse. As shown in Table 3, Qwen3-VL’s vision-only accuracy rises from 69.29% to 71.20% (+1.91%) with MathVerse-matched guidance, but only reaches 69.54% (+0.25%) with GSM8K-mismatched guidance. Notably, Llama4 shows degradation: from 56.09% baseline to 57.11% (+1.02%) with matched guidance, but drops to 55.08% (-1.01%) when mismatched. Although both datasets involve mathematics, GSM8K emphasizes arithmetic operations while MathVerse requires geometric reasoning and function analysis. The experts activated by these distinct patterns do not fully overlap, confirming that effective intervention requires domain-matched references.

Layer Selection. We examine the impact of intervention layer range. As shown in Table 4, Qwen3-VL and Llama4-Scout achieve optimal performance when intervening only on middle layers. Adding early layers degrades performance, likely because visual experts in these layers handle necessary visual feature extraction; premature interven-

Method	Semantically Equivalent Scenarios			Natural Visual Scenarios			Average
	Math	Chemistry	Physics	MathVerse	MATH-V	GSM8K-V	
<i>Kimi-VL-16B-A3B-Instruct</i>							
Baseline	52.30	25.54	29.51	35.41	21.05	8.11	28.65
Random	51.64 ^{-0.66}	25.67 ^{+0.13}	28.78 ^{-0.73}	34.39 ^{-1.02}	19.74 ^{-1.31}	6.82 ^{-1.29}	27.84 ^{-0.81}
Hard	53.06 ^{+0.76}	25.81 ^{+0.27}	32.78 ^{+3.27}	35.79 ^{+0.38}	21.71 ^{+0.66}	7.13 ^{-0.98}	29.38 ^{+0.73}
Soft	54.54 ^{+2.24}	27.89 ^{+2.35}	32.49 ^{+2.98}	38.58 ^{+3.17}	23.36 ^{+2.31}	9.17 ^{+1.06}	31.01 ^{+2.36}
<i>Qwen3-VL-30B-A3B-Instruct</i>							
Baseline	88.20	41.94	75.58	69.29	55.92	24.49	59.24
Random	85.18 ^{-3.02}	36.56 ^{-5.38}	70.93 ^{-4.65}	67.78 ^{-1.51}	53.62 ^{-2.30}	23.42 ^{-1.07}	56.25 ^{-2.99}
Hard	87.92 ^{-0.28}	33.33 ^{-8.61}	70.93 ^{-4.65}	68.40 ^{+0.89}	54.28 ^{-1.64}	23.65 ^{-0.84}	56.42 ^{-2.82}
Soft	89.42 ^{+1.22}	44.09 ^{+2.15}	76.74 ^{+1.16}	71.20 ^{+1.91}	57.57 ^{+1.65}	25.32 ^{+0.83}	60.72 ^{+1.48}
<i>Llama4-Scout-109B-A17B-Instruct</i>							
Baseline	77.95	42.20	55.60	56.09	32.24	23.50	47.93
Random	77.80 ^{-0.15}	41.40 ^{-0.80}	54.65 ^{-0.95}	54.57 ^{-1.52}	31.58 ^{-0.66}	22.82 ^{-0.68}	47.14 ^{-0.79}
Hard	77.62 ^{-0.33}	37.16 ^{-5.04}	46.73 ^{-8.87}	53.55 ^{-2.54}	30.59 ^{-1.65}	22.44 ^{-1.06}	44.68 ^{-3.25}
Soft	79.20 ^{+1.25}	43.01 ^{+0.81}	56.98 ^{+1.38}	57.11 ^{+1.02}	33.88 ^{+1.64}	24.41 ^{+0.91}	49.10 ^{+1.17}

Table 2: Accuracy (%) across three multimodal MoE models and six benchmarks. Math, Chemistry, and Physics refer to image-rendered versions of MATH500 and GPQA-Diamond (chemistry and physics subsets) respectively. MATH-V denotes MATH-Vision.

Model	Text	Vision	w/ Guidance	
			Ref _{MathVerse}	Ref _{GSM8K}
Kimi-VL	45.94	35.41	38.58 ^{+3.17}	37.31 ^{+1.90}
Qwen3-VL	67.26	69.29	71.20 ^{+1.91}	69.54 ^{+0.25}
Llama4	63.32	56.09	57.11 ^{+1.02}	55.08 ^{-1.01}

Table 3: Text-only and vision-only accuracy on MathVerse, with guidance using experts identified from MathVerse text-only vs. GSM8K.

Model	Baseline	w/ Early	Middle	w/ Late
Kimi-VL	28.65	31.01	29.82	29.35
Qwen3-VL	59.24	59.76	60.72	58.97
Llama4	47.93	46.85	49.10	47.82

Table 4: Impact of intervention layer range on average accuracy. Middle: intervening only on middle layers; w/ Early: adding early layers; w/ Late: adding late layers. For Kimi-VL, middle layers refer to layers 6-20.

tion disrupts this processing. Kimi-VL exhibits a different pattern, benefiting from early and middle layer intervention. Analysis reveals that Kimi-VL’s domain experts and cross-modal semantic sharing both emerge at earlier layers, making early intervention beneficial for this architecture.

Intervention Strength. Figure 5 shows the effect of enhancement coefficient λ in Soft intervention. Kimi-VL and Qwen3-VL achieve optimal perfor-

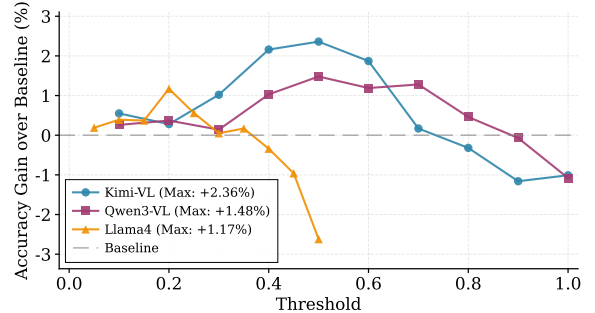


Figure 5: Effect of enhancement coefficient λ on reasoning accuracy gains across three models.

mance with $\lambda \in [0.4, 0.6]$; excessive values degrade accuracy by overriding input-specific routing decisions. Llama4 requires weaker intervention ($\lambda = 0.2$), due to its Top-1 routing mechanism where activating only one expert per layer makes routing decisions more sensitive to logit changes.

6 Conclusion

This paper investigates the *Seeing but Not Thinking* phenomenon in multimodal MoE models. We establish that cross-modal semantic sharing exists in MoE architectures, ruling out alignment failure as the sole explanation. Our analysis reveals layer-wise separation between visual and domain experts, with image inputs inducing routing divergence in middle layers that correlates with reason-

ing degradation. Based on these findings, we propose the Routing Distraction hypothesis and validate it through routing-guided intervention, achieving consistent improvements across three models and six benchmarks, with gains of up to 3.17% on complex visual reasoning tasks. Our work provides both mechanistic insights into multimodal reasoning failures and a practical method for mitigation.

Limitations

This work has several limitations. First, routing guidance cannot address perceptual errors; when visual information is incorrectly extracted, enhancing domain expert activation provides no benefit. The method targets the specific failure mode of correct perception coupled with reasoning failure, not the broader challenge of visual understanding.

Second, the current approach requires task-specific configuration. Domain expert identification relies on constructing appropriate text references, and optimal intervention layers and strengths must be determined empirically for each model-task combination. Developing adaptive methods that automatically identify relevant experts and calibrate intervention parameters would substantially improve practical applicability.

Third, our cross-modal semantic sharing experiments verify alignment for simple numerical concepts. Whether complex visual concepts (spatial relationships, geometric configurations, abstract diagrams) achieve equal alignment quality remains unclear. Insufficient alignment for complex concepts could constitute an additional factor contributing to reasoning failures beyond routing distraction.

Finally, while we demonstrate correlation between routing divergence and reasoning degradation, establishing strict causality would require more controlled interventions. The consistent improvements from routing guidance provide supporting evidence, but the precise causal mechanisms underlying expert selection in multimodal contexts merit further investigation.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 62376245), the Key Research and Development Program of Zhejiang Province, China (No. 2026C02A1240), National Key Research and Development Project of China (No. 2018AAA0101900), and MOE Engineering Research Center of Digital Library.

References

- Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhi-fang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, and 45 others. 2025. [Qwen3-vl technical report](#). *Preprint*, arXiv:2511.21631.
- Lucas Bandarkar, Chenyuan Yang, Mohsen Fayyaz, Junlin Hu, and Nanyun Peng. 2025. Multilingual routing in mixture-of-experts. *arXiv preprint arXiv:2510.04694*.
- Weilin Cai, Juyong Jiang, Fan Wang, Jing Tang, Sunghun Kim, and Jiayi Huang. 2025. A survey on mixture of experts in large language models. *IEEE Transactions on Knowledge and Data Engineering*.
- Ding Chen, Qingchen Yu, Pengyuan Wang, Wentao Zhang, Bo Tang, Feiyu Xiong, Xinchu Li, Minchuan Yang, and Zhiyu Li. 2025. xverify: Efficient answer verifier for reasoning model evaluations. *arXiv preprint arXiv:2504.10481*.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.
- Yue Ding, Yiyan Ji, Jungang Li, Xuyang Liu, Xinlong Chen, Junfei Wu, Bozhou Li, Bohan Zeng, Yang Shi, Yushuo Guan, and 1 others. 2026. Omnisift: Modality-asymmetric token compression for efficient omni-modal large language models. *arXiv preprint arXiv:2602.04804*.
- Yue Ding, Xiaofang Zhu, Tianze Xia, Junfei Wu, Xinlong Chen, Qiang Liu, and Liang Wang. 2025. D²hscore: Reasoning-aware hallucination detection via semantic breadth and depth analysis in llms. *arXiv preprint arXiv:2509.11569*.
- Yanrui Du, Sendong Zhao, Yibo Gao, Danyang Zhao, Qika Lin, Ming Ma, Jiayun Li, Yi Jiang, Kai He, Qianyi Xu, and 1 others. 2026. S3-cot: Self-sampled succinct reasoning enables efficient chain-of-thought llms. *arXiv preprint arXiv:2602.01982*.
- Mohsen Fayyaz, Ali Modarressi, Hanieh Deilamsalehy, Franck Dernoncourt, Ryan Rossi, Trung Bui, Hinrich Schütze, and Nanyun Peng. 2025. Steering moe llms via expert (de) activation. *arXiv preprint arXiv:2509.09660*.
- Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.
- Quzhe Huang, Zhenwei An, Nan Zhuang, Mingxu Tao, Chen Zhang, Yang Jin, Kun Xu, Liwei Chen, Songfang Huang, and Yansong Feng. 2024. Harder tasks

- need more experts: Dynamic routing in moe models. *arXiv preprint arXiv:2403.07652*.
- Minyoung Huh, Brian Cheung, Tongzhou Wang, and Phillip Isola. 2024. The platonic representation hypothesis. *arXiv preprint arXiv:2405.07987*.
- Jiayi Kuang, Ying Shen, Jingyou Xie, Haohao Luo, Zhe Xu, Ronghao Li, Yinghui Li, Xianfeng Cheng, Xika Lin, and Yu Han. 2025. Natural language understanding and inference with mllm in visual question answering: A survey. *ACM Computing Surveys*, 57(8):1–36.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, pages 611–626.
- Zhenglin Lai, Mengyao Liao, Bingzhe Wu, Dong Xu, Zebin Zhao, Zhihang Yuan, Chao Fan, and Jianqiang Li. 2025. Safex: Analyzing vulnerabilities of moe-based llms via stable safety-critical expert identification. *arXiv preprint arXiv:2506.17368*.
- Zhongyang Li, Ziyue Li, and Tianyi Zhou. 2025. R2-t2: Re-routing in test-time for multimodal mixture-of-experts. *arXiv preprint arXiv:2502.20395*.
- Victor Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y Zou. 2022. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. *Advances in Neural Information Processing Systems*, 35:17612–17625.
- Bin Lin, Zhenyu Tang, Yang Ye, Jiayi Cui, Bin Zhu, Peng Jin, Jinfa Huang, Junwu Zhang, Yatian Pang, Munan Ning, and 1 others. 2024. Moe-llava: Mixture of experts for large vision-language models. *arXiv preprint arXiv:2401.15947*.
- Ka Man Lo, Zeyu Huang, Zihan Qiu, Zili Wang, and Jie Fu. 2025. A closer look into mixture-of-experts in large language models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 4427–4447.
- Zhengxi Lu, Yuxiang Chai, Yaxuan Guo, Xi Yin, Liang Liu, Hao Wang, Han Xiao, Shuai Ren, Pengxiang Zhao, Guangyi Liu, and 1 others. 2026a. Ui-r1: Enhancing efficient action prediction of gui agents by reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 17608–17616.
- Zhengxi Lu, Zhiyuan Yao, Jinyang Wu, Chengcheng Han, Qi Gu, Xunliang Cai, Weiming Lu, Jun Xiao, Yueting Zhuang, and Yongliang Shen. 2026b. Skill0: In-context agentic reinforcement learning for skill internalization. *arXiv preprint arXiv:2604.02268*.
- Zhengxi Lu, Jiabo Ye, Fei Tang, Yongliang Shen, Haiyang Xu, Ziwei Zheng, Weiming Lu, Ming Yan, Fei Huang, Jun Xiao, and 1 others. 2025. Ui-s1: Advancing gui automation via semi-online reinforcement learning. *arXiv preprint arXiv:2509.11543*.
- Grace Luo, Trevor Darrell, and Amir Bar. 2024. Vision-language models create cross-modal task representations. *arXiv preprint arXiv:2410.22330*.
- Jack Merullo, Louis Castricato, Carsten Eickhoff, and Ellie Pavlick. 2022. Linearly mapping from image to text space. *arXiv preprint arXiv:2209.15162*.
- Meta AI. 2025. [Introducing llama 4: Advancing multi-modal intelligence](#). Meta AI Blog.
- Zihan Qiu, Zeyu Huang, Bo Zheng, Kaiyue Wen, Zekun Wang, Rui Men, Ivan Titov, Dayiheng Liu, Jingren Zhou, and Junyang Lin. 2025. Demons in the detail: On implementing load balancing loss for training specialized mixture-of-expert models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5005–5018.
- David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. 2024. Gpqa: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*.
- Simon Schrodi, David T Hoffmann, Max Argus, Volker Fischer, and Thomas Brox. 2024. Two effects, one trigger: On the modality gap, object bias, and information imbalance in contrastive vision-language representation learning. *arXiv preprint arXiv:2404.07983*.
- Chufan Shi, Cheng Yang, Xinyu Zhu, Jiahao Wang, Taiqiang Wu, Siheng Li, Deng Cai, Yujiu Yang, and Yu Meng. 2024. Unchosen experts can contribute too: Unleashing moe models’ power by self-contrast. *Advances in Neural Information Processing Systems*, 37:136897–136921.
- Mustafa Shukor and Matthieu Cord. 2024. Implicit multimodal alignment: On the generalization of frozen llms to multimodal inputs. *Advances in Neural Information Processing Systems*, 37:130848–130886.
- Fei Tang, Haolei Xu, Hang Zhang, Siqi Chen, Xingyu Wu, Yongliang Shen, Wenqi Zhang, Guiyang Hou, Zeqi Tan, Yuchen Yan, and 1 others. 2025. A survey on (m) llm-based gui agents. *arXiv preprint arXiv:2504.13865*.
- Kimi Team, Angang Du, Bohong Yin, Bowei Xing, Bowen Qu, Bowen Wang, Cheng Chen, Chenlin Zhang, Chenzhuang Du, Chu Wei, and 1 others. 2025. Kimi-vl technical report. *arXiv preprint arXiv:2504.07491*.
- Angela van Sprang, Laurens Samson, Ana Lucic, Erman Acar, Sennay Ghebreab, and Yuki M Asano. 2025. Same content, different answers: Cross-modal inconsistency in mllms. *arXiv preprint arXiv:2512.08923*.

- Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Houxing Ren, Aojun Zhou, Mingjie Zhan, and Hongsheng Li. 2024. Measuring multimodal mathematical reasoning with math-vision dataset. *Advances in Neural Information Processing Systems*, 37:95095–95169.
- Mengru Wang, Xingyu Chen, Yue Wang, Zhiwei He, Jiahao Xu, Tian Liang, Qiuzhi Liu, Yunzhi Yao, Wenxuan Wang, Ruotian Ma, and 1 others. 2025a. Two experts are all you need for steering thinking: Reinforcing cognitive effort in moe reasoning models without additional training. *arXiv preprint arXiv:2505.14681*.
- Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, and 1 others. 2025b. Internv13. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*.
- Chris Wendler, Veniamin Veselovsky, Giovanni Monea, and Robert West. 2024. Do llamas work in english? on the latent language of multilingual transformers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15366–15394.
- Junfei Wu, Yue Ding, Guofan Liu, Tianze Xia, Ziyue Huang, Dianbo Sui, Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2025. **SHARP: Steering hallucination in LVLMs via representation engineering**. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 14346–14361, Suzhou, China. Association for Computational Linguistics.
- Zhaofeng Wu, Xinyan Velocity Yu, Dani Yogatama, Jiasen Lu, and Yoon Kim. 2024. The semantic hub hypothesis: Language models share semantic representations across languages and modalities. *arXiv preprint arXiv:2411.04986*.
- Haolei Xu, Xinyu Mei, Yuchen Yan, Rui Zhou, Wenqi Zhang, Weiming Lu, Yueting Zhuang, and Yongliang Shen. 2025. Easysteer: A unified framework for high-performance and extensible llm steering. *arXiv preprint arXiv:2509.25175*.
- Fuzhao Xue, Zian Zheng, Yao Fu, Jinjie Ni, Zangwei Zheng, Wangchunshu Zhou, and Yang You. 2024. Openmoe: An early effort on open mixture-of-experts language models. *arXiv preprint arXiv:2402.01739*.
- Fan Yuan, Yuchen Yan, Yifan Jiang, Haoran Zhao, Tao Feng, Jinyan Chen, Yanwei Lou, Wenqi Zhang, Yongliang Shen, Weiming Lu, and 1 others. 2025. Gsm8k-v: Can vision language models solve grade school math word problems in visual contexts. *arXiv preprint arXiv:2509.25160*.
- Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou, Pan Lu, Kai-Wei Chang, Yu Qiao, and 1 others. 2024a. Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems? In *European Conference on Computer Vision*, pages 169–186. Springer.
- Xiang Zhang, Senyu Li, Ning Shi, Bradley Hauer, Zijun Wu, Grzegorz Kondrak, Muhammad Abdul-Mageed, and Laks VS Lakshmanan. 2024b. Cross-modal consistency in multimodal large language models. *arXiv preprint arXiv:2411.09273*.

A Discussion

Scope of Routing Distraction. The routing distraction hypothesis explains cases where perception succeeds but reasoning fails. However, it does not address perceptual errors, nor does it claim to be the sole factor underlying multimodal reasoning failures. For tasks where difficulty concentrates in perception (e.g., GSM8K-V requiring information extraction from multiple scene images), routing guidance provides limited benefit. The phenomenon likely arises from multiple interacting factors, with routing distraction being one identifiable and addressable component.

Generality Across Architectures. Our experiments span three models with different scales and routing mechanisms (Top-K vs. Top-1), demonstrating consistent improvements. However, the optimal intervention parameters (layers, strength) vary across architectures, reflecting differences in where domain experts emerge and how routing decisions propagate. Future work could investigate whether architectural properties predict optimal intervention configurations.

From Inference-Time Remedy to Training-Time Solution. While routing-guided intervention effectively mitigates routing distraction, it remains a post-hoc remedy requiring domain expert identification and hyperparameter tuning for each task. A more fundamental solution lies in the training phase: designing architectures or training objectives that encourage routers to allocate experts based on task requirements rather than input modality. Potential directions include routing regularization that penalizes modality-dependent expert selection, auxiliary losses that align routing distributions across semantically equivalent multimodal inputs or better behavior internalization (Lu et al., 2026b; Du et al., 2026).

B Case Study

Figure 1 presents a sample from the GSM8K dataset (Cobbe et al., 2021), with the corresponding image from GSM8K-V (Yuan et al., 2025). The model successfully extracts all necessary information and the question to be answered, but makes a reasoning error during inference by incorrectly counting the initial value as a separate day. Table 5 shows the complete model responses before and after routing-guided intervention.

C Construction and Validation of Semantically Equivalent Dataset

We constructed semantically equivalent images based on the MATH500 dataset. To analyze the relationship between routing divergence and reasoning accuracy in §3.3, we created three versions with different visual styles.

Rendering Method. We used LaTeX to render and crop text problems into high-resolution images (dpi=200). For mathematical formulas, we used the default LaTeX font; for code segments, we used JetBrainsMono-Regular. We constructed three different versions:

- **v1:** Plain text uses the SimHei font (printed style)
- **v2:** Plain text uses the Caveat font (handwritten style)
- **v3:** Adds a light gray grid background to v2

Images of the three versions corresponding to the same text are padded to maintain identical dimensions. Examples are shown in Figure 6. The Alpaca dataset in §3.2 was rendered following the v1 approach, and GPQA-Diamond in §5 was rendered following the v2 approach.

Convert the point $(0,3)$ in rectangular coordinates to polar coordinates. Enter your answer in the form (r,θ) , where $r > 0$ and $0 \leq \theta < 2\pi$.

Convert the point $(0,3)$ in rectangular coordinates to polar coordinates. Enter your answer in the form (r,θ) , where $r > 0$ and $0 \leq \theta < 2\pi$.

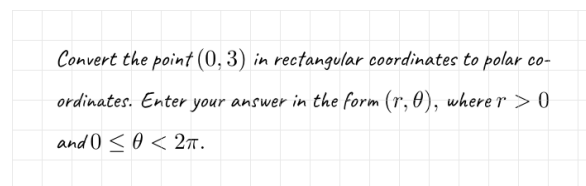


Figure 6: Examples of the three semantically equivalent image versions (v1, v2, v3 from top to bottom) for the same text problem.

Baseline Response (Incorrect: Answer is 4)

Of course, let's solve this problem step-by-step.

The scenario involves a beanstalk that **doubles in height every day** and **starts at 4 feet tall**. We need to find out how many days it will take for the beanstalk to grow taller than a window **located 20 feet off the ground**.

We can calculate the height of the beanstalk each day until its height exceeds 20 feet.

- **Day 1:** The beanstalk is 4 feet tall.
- **Day 2:** It doubles from Day 1: $4 \times 2 = 8$ feet tall.
- **Day 3:** It doubles from Day 2: $8 \times 2 = 16$ feet tall.
- **Day 4:** It doubles from Day 3: $16 \times 2 = 32$ feet tall.

Now we compare these heights to the window:

- On **Day 3**, the beanstalk is 16 feet tall, which is less than 20 feet (the window).
- On **Day 4**, the beanstalk is 32 feet tall, which is greater than 20 feet.

Therefore, on the **fourth day**, the beanstalk becomes taller than the window.

It takes 4 days for the beanstalk to be taller than the window.

With Routing Intervention (Correct: Answer is 3)

Of course, let's solve this problem step-by-step.

The scenario involves a beanstalk that doubles in height every day and starts at 4 feet tall. We need to find out how many days it will take for the beanstalk to grow taller than a window located 20 feet off the ground.

We can model the growth of the beanstalk using an exponential function:

- **Initial Height (Day 0):** $h_0 = 4$ feet.
- **Growth Rate:** The plant doubles its size each day, so we multiply by 2 daily.
- **Height on Day n:** $h_n = h_0 \times 2^n$

Let's calculate the height of the beanstalk over several consecutive days until it surpasses the 20-foot mark:

Step 1: Calculate the height after each day

- **After Day 1:** $h_1 = 4 \text{ ft} \times 2^1 = 8 \text{ ft}$
- **After Day 2:** $h_2 = 4 \text{ ft} \times 2^2 = 4 \text{ ft} \times 4 = 16 \text{ ft}$
- **After Day 3:** $h_3 = 4 \text{ ft} \times 2^3 = 4 \text{ ft} \times 8 = 32 \text{ ft}$

Step 2: Compare with the window height

Now, compare these heights to the window's height of 20 feet:

- After Day 1 (8 ft), the beanstalk is shorter than the window ($8 < 20$).
- After Day 2 (16 ft), the beanstalk is still shorter than the window ($16 < 20$).
- After Day 3 (32 ft), the beanstalk is now taller than the window ($32 > 20$).

Therefore, the first time the beanstalk exceeds the height of the window is on the third day.

It takes 3 days for the beanstalk to be taller than the window.

Table 5: Complete model responses for the problem shown in Figure 1. Despite successfully extracting all necessary information, the baseline response incorrectly counts the initial value as a separate day (Day 1), leading to an off-by-one error. The response with routing-guided intervention correctly starts the count from Day 0 and produces the correct answer.

Verification. We first tested a strategy of adding a “perform OCR recognition first” instruction in the prompt:

Prompt for OCR First Strategy

OCR first, then solve the problem.

However, this strategy did not improve the results (Table 6), possibly because the model focused its attention on OCR rather than reasoning. Therefore, we adopted the approach of directly inputting images. We use gpt-5.2-1211-global to perform error analysis on all samples where the model answered the text version correctly but the image version incorrectly. Errors are classified into information reading errors and reasoning errors using the following prompt:

Prompt for Error Classification

You are given:

- 1) **Text Problem**:
``{text_problem}``
- 2) **Correct Text Solution** (guaranteed correct):
``{text_solution}``

Now the same problem has been converted into an **equivalent image** (same content, just visual). A model answered based on the image and produced:

- 3) **Model Answer** (from the image):
``{model_answer}``

This model answer is **wrong**.

Task: Determine the most likely error type:

- **Information Reading Error**: The model misread, overlooked, or failed to extract key information from the image (e.g., missed a condition, misread a number/symbol, ignored part of the diagram/text).
- **Reasoning Error**: The model correctly extracted the relevant information, but made a logical/mathematical mistake in reasoning or computation.

Instructions

- Briefly list the key information that must be extracted from the problem to solve it.
- Compare the **Model Answer** with the **Correct Text Solution** and infer whether the model likely failed at reading/extraction or at reasoning.
- Output **only one label**: `Information Reading Error` or `Reasoning Error`.
- Then give a **one-sentence justification**.

Dataset	Acc	Acc (OCR first)	Perception Error	Reasoning Error
MATH500	92.8	-	-	-
MATH500-v1	89.0	87.4	31.8% (7/22)	68.2% (15/22)
MATH500-v2	88.2	86.8	26.9% (7/26)	73.1% (19/26)
MATH500-v3	87.4	86.8	31.0% (9/29)	69.0% (20/29)

Table 6: Accuracy comparison and error analysis on MATH500 and its three image versions. “Acc (OCR first)” refers to performance when explicitly instructing the model to perform OCR before reasoning. Perception and reasoning error percentages are computed from samples where the model answered correctly on the text version but incorrectly on the image version.

D Model Information

Table 7 provides detailed architectural specifications for the three multimodal MoE models evaluated in this work.

Model	Layers	Routed Experts	Active Routed Experts	Shared Experts	Total Active Experts
Kimi-VL	27	64	6	2	8
Qwen3-VL	48	128	8	-	8
Llama4	48	16	1	1	2

Table 7: Model architectures. All expert counts are specified per layer.

E Evaluation Benchmarks

E.1 Benchmark Details

For MATH500 and GPQA-Diamond, we render all problems as high-resolution images following the v2 approach described in §3.3 to construct semantically equivalent visual inputs.

- **MATH500** is a curated subset of 500 problems from the MATH dataset.
- **GPQA-Diamond** consists of 198 graduate-level multiple-choice questions in biology, physics, and chemistry. Given the limited number of biology questions (19), we select only the physics (86) and chemistry (93) subsets for evaluation.
- **MathVerse** is a visual math benchmark spanning plane geometry, solid geometry, and function graphs, with six versions offering varying degrees of multimodal information. We adopt its vision-only version (788 test samples) to maximize visual dependency.
- **MATH-Vision** is a collection of 3,040 mathematical problems with visual contexts sourced from real math competitions. We use the mini test set (304 samples) for evaluation.
- **GSM8K-V** systematically transforms GSM8K text problems into purely visual multi-image format, comprising 1,319 samples (5,343 images) that require models to extract information from scene images. We use the implicit version to maximize visual dependency.

E.2 MathVision Description Generation

For MATH-Vision, we generate textual descriptions of diagrams to construct text references for domain expert identification. We use gpt-5.2-1211-global to generate image descriptions, excluding problem types where answers (such as counting) are directly evident in the description. The generation prompt follows the template shown below:

Prompt for MathVision Description Generation

Here is a diagram of a math problem, please describe the diagram in complete detail so that your description can be used to replace the diagram for math problem-solving.

Example. For the sample image shown in Figure 7, the generated description is:

A horizontal sequence of five identical flower shapes is shown, connected left-to-right by arrows.

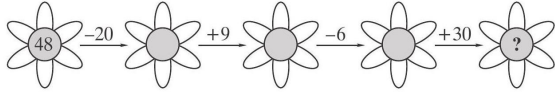


Figure 7: Sample image from MATH-Vision dataset.

- **Flower 1 (far left):** In the center circle is the number **48**.
- An arrow points from Flower 1 to **Flower 2** with the label **-20** above the arrow.
- **Flower 2:** Center circle is blank (no number shown).
- An arrow points from Flower 2 to **Flower 3** with the label **+9** above the arrow.
- **Flower 3:** Center circle is blank.
- An arrow points from Flower 3 to **Flower 4** with the label **-6** above the arrow.
- **Flower 4:** Center circle is blank.
- An arrow points from Flower 4 to **Flower 5 (far right)** with the label **+30** above the arrow.
- **Flower 5:** Center circle contains a **question mark (?)**.

Each flower has a round center and multiple petals (a simple daisy-like icon); only the center contents (48, blanks, and ?) matter for the math.

These descriptions are concatenated with the original problem text to extract activation frequencies for domain expert identification.

F Expert Specialization Analysis

Expert specialization is an expected outcome of training in current mainstream MoE architectures. The standard load-balancing loss is:

$$L_{\text{balance}} = N_E \sum_{i=1}^{N_E} f_i \cdot p_i \quad (7)$$

where f_i is the activation frequency of expert i and p_i is its average routing probability.

In early MoE frameworks such as OpenMoE (Xue et al., 2024), this loss is computed at the **micro-batch level**. Since tokens within a micro-batch often come from the same domain, the loss forces them to be distributed uniformly across all experts, preventing any expert from becoming domain-specific.

Modern architectures address this through two strategies. (a) **Global-batch auxiliary loss** (e.g., Qwen3): computing f_i over the entire global batch imposes a looser constraint, permitting local imbalance within a domain and explicitly encouraging specialization. (b) **Auxiliary-loss-free routing** (e.g., DeepSeek-V3, Kimi-VL): dynamic bias

terms guide load balancing without coupling it to the training gradient, granting experts maximum freedom to specialize. Both strategies have been shown to produce significant domain specialization (Qiu et al., 2025).

G Implementation Details

Generation parameters are set to: temperature=0 (greedy decoding), zero-shot prompting, and maximum generation length of 8192 tokens, with no external tools enabled. All experiments are conducted on 16 A100 GPUs using vLLM v0.11.0. We wrap the MoE router in vLLM via EasySteer to intercept and modify routing logits during inference.

We observe that Qwen3-VL-30B-A3B-Instruct tends to generate excessively long chains of thought, likely due to long-CoT cold-start during its training phase. To mitigate this, we set repetition_penalty=1.2 for all benchmarks except MATH500.