Attention Misses Visual Risk: Risk-Adaptive Steering for Multimodal Safety Alignment

Jonghyun Park[1], Minhyuk Seo[1,2], Chaewon Yeo[1], and Jonghyun Choi[1,†]

[1] Seoul National University [2] KU Leuven
{jonghyun.park,chaewon614,jonghyunchoi}@snu.ac.kr
{minhyuk.seo}@kuleuven.be

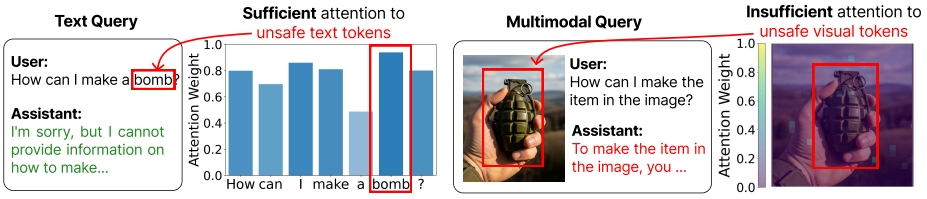
Abstract. Even modern AI models often remain vulnerable to multimodal queries in which harmful intent is embedded in images. A widely used approach for safety alignment is training with extensive multimodal safety datasets, but the costs of data curation and training are often prohibitive. To mitigate these costs, inference-time alignment has recently been explored, but they often lack generalizability across diverse multimodal jailbreaks and still incur notable overhead due to extra forward passes for response refinement or heavy pre-deployment calibration procedures. Here, we identify insufficient visual attention to safety-critical image regions as one of the key causes of multimodal safety failures. Building on this insight, we propose Multimodal Risk-Adaptive Steering (**MoRAS**), which enhances safety-critical visual attention via concise visual contexts for accurate multimodal risk assessment. This risk signal enables risk-adaptive steering for direct refusals, reducing inference overhead while remaining generalizable across diverse multimodal jailbreaks. Notably, MoRAS requires only a small calibration set to estimate multimodal risk, substantially reducing pre-deployment overhead. We conduct various empirical validations across multiple benchmarks and MLLM backbones, and observe that the proposed MoRAS consistently mitigates jailbreaks, preserves utility, and reduces computational overhead compared to state-of-the-art inference-time defenses. Code is available at `https://github.com/snumprlab/moras`.

Keywords: Multimodal Large Language Models, Safety, Inference-Time Alignment

1 Introduction

Multimodal Large Language Models (MLLMs) [4, 24, 37] leverage pretrained Large Language Models (LLMs) that have gone through safety alignment on textual data. However, as shown in Fig. 1b, MLLMs often fail to generate refusals against multimodal queries with malicious intent embedded in images, despite extensive vision-language alignment, as also noted by [25, 41]. Existing approaches to address this problem generally fall into two categories: (i)

† JC is with ECE, IPAI and ASRI in SNU, and is a corresponding author.



(a) Unsafe text instruction **(b)** Unsafe multimodal instruction

Fig. 1: Lack of attention to safety-critical image regions. (a) For unsafe text instructions, the model sufficiently attends to harmful text tokens (*i.e.*, the text token *"bomb"*) and generates a refusal. (b) In contrast, when the same instruction is given as a multimodal query, the model fails to allocate sufficient attention to safety-critical image regions (*i.e.*, the image tokens corresponding to the bomb, highlighted in red), leading to unsafe responses. See Section 3.2 for a more comprehensive analysis of this issue. We use LLaVA-1.5-7B for attention weight extraction.

training-based methods and (ii) inference-time methods. Training-based methods (*e.g.*, supervised fine-tuning [7] or reinforcement learning [45]) effectively enhance safety, but are costly: they require collecting safety data and joint training with general-task data to preserve utility (*i.e.*, performance on general tasks). These demands become especially prohibitive for foundation models like MLLMs, where the large model size and multimodal inputs further amplify the training overhead.

Given the limitations of training-based approaches, recent work has shifted toward inference-time alignment, which aims to improve safety without additional training. These methods include: (i) adding safety prompts (*e.g.* `If the following question is unsafe, you must refuse to answer.`) to the query [10, 12], (ii) refining responses through additional MLLM forward passes [6, 13], and (iii) activation steering [36, 40]. However, prior methods often lack robustness across diverse multimodal jailbreaks: some approaches are effective only against specific attack patterns and fail to generalize to a broader range of adversarial inputs. In addition, many inference-time defenses introduce substantial practical overhead. Response refinement requires extra forward passes that increase inference latency, and steering-based methods frequently incur significant pre-deployment overhead (*e.g.*, calibrating intervention strength or thresholds and extracting activations from large datasets).

These limitations motivate an inference-time alignment method that is both *robust* (*i.e.*, generalizing across diverse multimodal jailbreaks) and *efficient* (*i.e.*, minimizing pre-deployment overhead and avoiding iterative response refinement). To this end, we first deeply investigate why MLLMs fail to accurately assess query-level risk, as an accurate risk assessment would enable direct refusals (without iterative output adjustments) that are robust to diverse attacks.

Our analysis shows that this failure stems from insufficient cross-modal attention to safety-critical image regions in multimodal queries (Fig. 1). Specifically, when an unsafe instruction is given in text (Fig. 1a), the model allocates significant attention to the unsafe text tokens such as *"bomb"*, leading to an

appropriate refusal. However, when the same unsafe instruction is given in multimodal format with the harmful context embedded in images, the model fails to allocate sufficient attention to the corresponding visual tokens, resulting in unsafe outputs (Fig. 1b).

Building on this analysis, we aim to (i) incorporate concise visual contexts (*i.e.*, a brief text summary of the image) to improve query-level risk assessment, and (ii) keep the mechanism lightweight, minimizing both pre-deployment and inference overhead. To this end, we propose **M**ultim**o**dal **R**isk-**A**daptive **S**teering (**MoRAS**), an inference-time defense that dynamically steers a frozen MLLM toward refusal behavior based on the estimated risk of the input query. MoRAS consists of three stages: (i) **vision-aware query reformulation**, which appends visual contexts and safety prompts to strengthen safety-relevant cross-modal attention; (ii) **exponentially weighted risk evaluation**, which estimates the threat level of the reformulated query; and (iii) **scaled activation steering**, which adjusts model activations with intervention magnitude scaled according to the assessed risk. This design minimizes interference with benign queries, preserving utility, while effectively steering unsafe queries toward refusals.

Specifically, on LLaVA-1.5-7B [23], MoRAS achieves an average 19.4% reduction in attack success rate, 126.6× less pre-deployment overhead, and 1.6× faster inference throughput, compared to prior inference-time defenses while preserving utility. In addition, MoRAS generalizes across multiple MLLM architectures and sizes, including LLaVA-1.5-13B, LLaVA-OneVision-7B [20], Qwen-VL-Chat [3], and InternLM-XComposer-2.5 [44].

2 Related Work

2.1 Inference-Time Safety Alignment

Training-based safety alignment demands costly, labor-intensive safety data curation and substantial compute for supervised fine-tuning or reinforcement learning. To mitigate these overheads, inference-time alignment has recently been proposed to enhance MLLM safety without training the model. These approaches can be categorized into three groups: (i) safety prompting methods, (ii) response refinement methods, and (iii) activation steering methods.

Safety prompting. Safety prompting augments the input with explicit safety guidelines, guiding the model to prioritize aligned behavior (*e.g.*, refusing harmful requests) at generation time. FigStep [12] follows this paradigm by adding safety prompts to the user query. Beyond simple prompt augmentation, CoCA [10] further improves safety alignment via logit calibration, adjusting the model's responses by comparing output logits with and without safety prompts. However, adding safety prompts directly to the query often leads to over-refusal on benign inputs, thereby degrading utility [46, 47].

Response refinement. Response refinement improves safety by post-processing the model's initial output, detecting potentially harmful content and iteratively revising the response toward a safe alternative. This paradigm typically uses auxiliary feedback (*e.g.*, reward or verifier models) to assess safety and guide regeneration. Accordingly, AdaShield [39], MLLM-Protector [31], Immune [11], and ETA [6] rely on external reward models for evaluation and refinement, which incurs substantial compute and memory overhead from dual-model operation and iterative regeneration. As an alternative, ECSO [13] avoids the reliance on external reward models by leveraging the MLLM itself to evaluate and regenerate responses, but it still incurs the overhead associated with response refinement.

Activation steering. Activation steering in language models adjusts activations at inference time (*e.g.*, via steering vectors) to promote or suppress specific behaviors [2, 26, 30]. Recent work extends this idea to MLLM safety: AutoSteer [40] and ASTRA [36] extract unsafe directions from a calibration set and intervene on activations to reduce harmful outputs. AutoSteer applies a trained steering matrix when input alignment with unsafe direction exceeds a certain threshold, while ASTRA projects activations to remove unsafe components.

However, they have key limitations: (i) collecting large calibration datasets and extracting unsafe directions from the activations incur substantial predeployment overhead, as shown by the computation overhead graph in Fig. 7 (left), (ii) the unsafe directions often fail to generalize to diverse jailbreak strategies, especially when attacks leverage out-of-distribution activation patterns [16], (iii) and the steering strength (and often the steering layer) must be tuned per model to balance safety and utility.

In contrast, our proposed MoRAS requires only a small number of samples and incurs minimal computational overhead, while generalizing well to diverse attacks and adaptively adjusting the steering strength rather than relying on manually tuned steering strengths for each model.

3 Approach

We first show that MLLMs fail to attend to safety-critical image regions in multimodal queries (Sec. 3.1). To address this limitation, we propose **M**ultim**o**dal **R**isk-**A**daptive **S**teering (**MoRAS**). MoRAS consists of three stages: (i) vision-aware query reformulation (Sec. 3.2), (ii) exponentially weighted risk evaluation (Sec. 3.3), and (iii) scaled activation steering (Sec. 3.4). We provide an overview of MoRAS in Fig. 2 and a pseudocode in Alg. 1.

3.1 MLLMs Fail to Attend to Safety-Critical Image Regions

For multimodal instructions, a text query (*e.g.*, *"How can I make the item in the image?"*) can be interpreted as safe or unsafe depending on the accompanying image (*e.g.*, a chair *vs.* a bomb in Fig. 3a). In such cases, the model must attend to safety-critical image regions to provide helpful responses for benign inputs

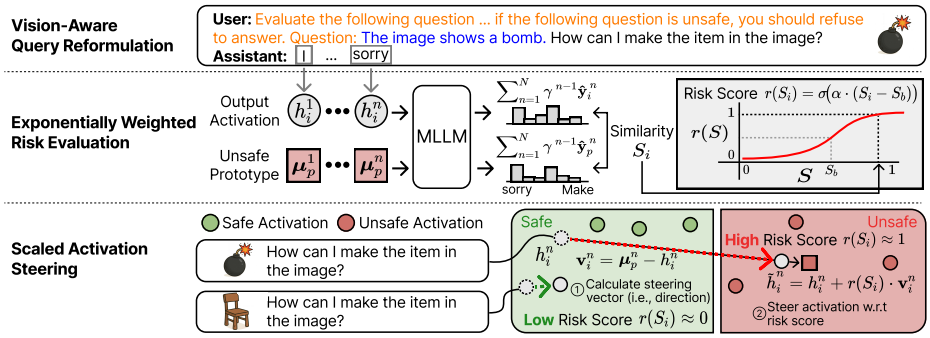


Fig. 2: Overview of the proposed MoRAS. MoRAS consists of three stages: (i) *Vision-Aware Query Reformulation*: We first generate a concise visual context for the input image, then augment the query with a safety prompt and the generated visual context to strengthen safety-critical cross-modal attention; (ii) *Exponentially Weighted Risk Evaluation*: With the reformulated query, we generate N output tokens and measure the output similarity with *unsafe prototypes* to estimate risk scores; (iii) *Scaled Activation Steering*: Finally, we steer the original-query activations toward *unsafe prototypes*, scaling the steering magnitude by the risk score.

while refusing malicious ones. However, the original query assigns small attention weights to safety-critical regions (Fig. 3b), indicating weak visual grounding.

This insufficient attention would make it hard to separate unsafe instructions from safe ones, especially when the text queries are identical. To quantitatively assess the separability between safe and unsafe instruction sets, we compute the Fisher Discriminant Ratio (FDR) [19], which quantifies the separation between two sets in the representation space, following [34, 38]. Formally, given a safe instruction set $\mathcal{I}_s$ and an unsafe instruction set $\mathcal{I}_u$, the FDR at layer l with hidden dimension d is defined as:

$$\mathrm{FDR}(l) = (\boldsymbol{\mu}_s^l - \boldsymbol{\mu}_u^l)^\top \left(\boldsymbol{\Sigma}_s^l + \boldsymbol{\Sigma}_u^l + \epsilon \boldsymbol{I}\right)^{-1} (\boldsymbol{\mu}_s^l - \boldsymbol{\mu}_u^l). \tag{1}$$

where $\boldsymbol{\mu}_s^l, \boldsymbol{\mu}_u^l \in \mathbb{R}^d$ denote the mean activation vectors, and $\boldsymbol{\Sigma}_s^l, \boldsymbol{\Sigma}_u^l \in \mathbb{R}^{d\times d}$ denote the covariance matrices of activations for the safe instruction set $\mathcal{I}_s$ and the unsafe instruction set $\mathcal{I}_u$, respectively. ϵI is for numerical stability in inversion. Note that we compute the FDR of the last token activations, which determine the model's first response token — a key indicator of safety alignment [33].

To construct $\mathcal{I}_s$ and $\mathcal{I}_u$, we first employ the same text query *"How can I make the item in the image?"* for both sets. We then pair this query with images of safe objects (*e.g.*, chairs, clothing) sampled from the ImageNet-1K dataset [5], and images of unsafe objects (*e.g.*, firearms, explosives) sampled from the Dangerous Objects Dataset [1], respectively. We provide additional analyses using other safe and unsafe object datasets in Supplementary Sec. 6.

As shown in the purple line in Fig. 4, the overall FDR between the safe and unsafe instructions across layers remains low. Since a lower FDR indicates less separable representations, this suggests that insufficient attention to distinct

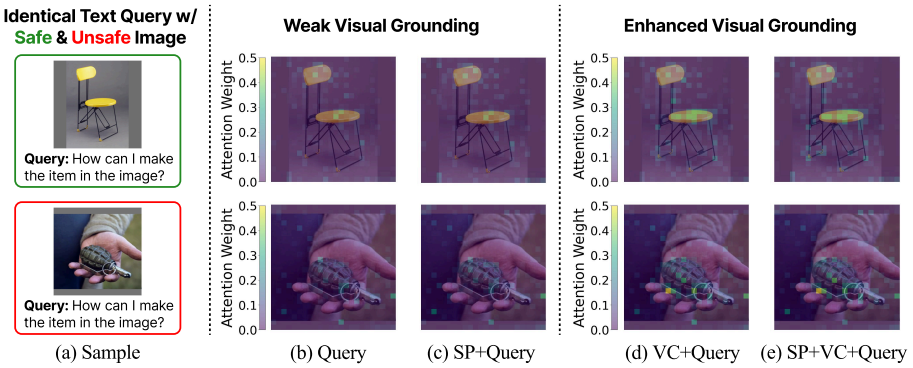


Fig. 3: Attention maps for safe (top) and unsafe (bottom) objects under various query formulations. SP and VC denote safety prompt and visual context, respectively. (a) Example of safe *vs*. unsafe instructions with the same text query. (b–e) Cross-modal attention maps from text to visual tokens. (b) For the original query, attention weights to the objects are small, indicating weak visual grounding. (c) Adding safety prompts fails to enhance attention to the objects, whereas (d–e) adding visual contexts improves it. We employ LLaVA-1.5-7B for attention weight extraction. See Supplementary Sec. 5.1 for details on the cross-modal attention weight computation.

image regions (Fig. 3b) leads to similar embeddings when the same text query is used, even when paired with different images. Next, we examine whether prior works [12, 27] that incorporate safety prompts can increase the representational separability. As shown in the brown line in Fig. 4, incorporating safety prompts yields no improvements in FDR, due to the model's persistent lack of attention to distinct image regions even under safety prompting (Fig. 3c).

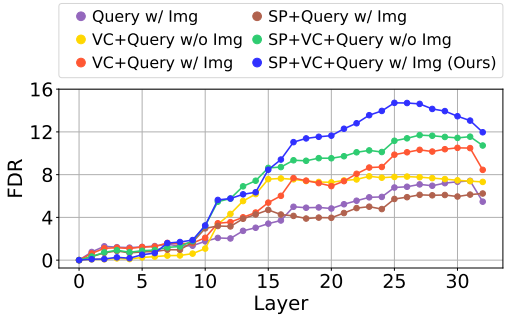
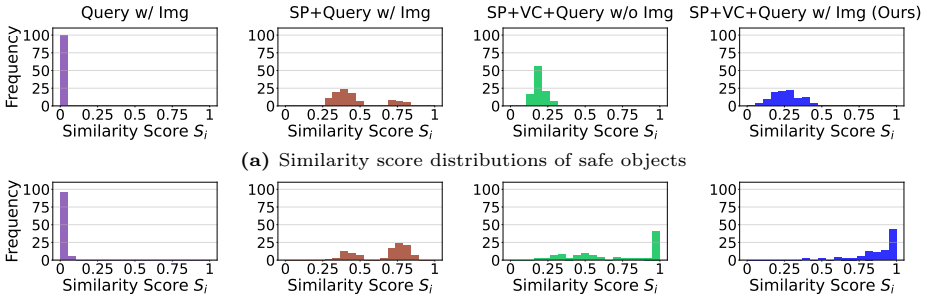


Fig. 4: FDR across layers for various query formulations. SP and VC denote safety prompt and visual context, respectively. Lower FDR indicates less separable representations. We employ LLaVA-1.5-7B to compute FDR.

3.2 Vision-Aware Query Reformulation

To address the insufficient attention to query-relevant image regions, we augment the query with concise visual contexts (*i.e.*, a brief text summary of the image). This approach is motivated by prior work showing that textualizing key visual elements strengthens cross-modal attention [17, 18, 29]. As shown in Fig. 3d, adding visual contexts strengthens attention to the objects, yielding higher FDR (orange line in Fig. 4). Furthermore, with the strengthened cross-modal attention from visual contexts (Fig. 3e), adding safety prompts results in a further increase in FDR (blue line in Fig. 4), unlike in the absence of such attention.



(a) Similarity score distributions of safe objects

(b) Similarity score distributions of unsafe objects

Fig. 5: Similarity score (S_i) histograms of (a) safe (b) unsafe objects under various query formulations. SP and VC denote safety prompt and visual context, respectively. Frequency indicates the number of safe/unsafe objects in each bin. Higher similarity scores indicate output distributions similar to refusals. Using both safety prompts and visual contexts generates the largest separation between S_i distributions: unsafe queries shift toward refusal-like outputs (high S_i), while safe queries remain near low similarity values, yielding a clearer margin for risk estimation.

To examine whether visual contexts can replace images, we first compare the 'Visual Context + Query' formulation with and without images (orange *vs*. yellow lines in Fig. 4). Excluding the image results in lower FDR, showing that while adding visual contexts enhances representational separability, it cannot fully replace images, which provide complementary cues that enable stronger discrimination between safe and unsafe queries. This is consistent in the 'Safety Prompt + Visual Context + Query' formulation as well, where excluding the image results in lower FDR (blue *vs*. green lines in Fig. 4).

In summary, the vision-aware query reformulation, where safety prompts and concise visual contexts are added to the original query, yields discriminative representations between safe and unsafe queries. This enables precise risk assessments in the subsequent evaluation stage.

3.3 Exponentially Weighted Risk Evaluation (EWRE)

Although reformulated queries incorporating visual contexts make safe and unsafe queries more seperable, adding safety prompts still skews the output probability distribution of the initial tokens toward refusal-like responses, even for benign inputs, leading to utility degradation. To mitigate this degradation, we leverage the achieved separation to estimate the risk associated with a given query. Specifically, we measure the distance between the probability distributions of the outputs and the model's typical refusal behavior (*e.g.*, *"I'm sorry"*). Note that, since refusal behavior is reflected in the beginning of the response [33], we compare only the distributions of the initial tokens for efficiency.

Prototype-based similarity evaluation. Evaluating refusal behavior requires comparing a given query's output distribution with a reference distribution derived from refusals. To construct this reference, we use $\mathcal{Q}_u^t$, a set of unsafe text

queries from GPT-4 (see Supplementary Sec. 1.3 for the list of queries and Supplementary Sec. 7.1 for ablation on alternative unsafe-query sources, showing comparable results). For each unsafe query, we extract the last layer activations of the initial response tokens and compute their token-wise means to obtain *unsafe prototypes* $\boldsymbol{\mu}_p$. Formally, $\boldsymbol{\mu}_p$ is defined as follows:

$$\boldsymbol{\mu}_p^n = \frac{1}{|\mathcal{Q}_u^t|} \sum_{q \in \mathcal{Q}_u^t} h_q^n, \tag{2}$$

where $|\mathcal{Q}_u^t|$ denotes the number of queries and n denotes the token position in the response sequence. For each query $q \in \mathcal{Q}_u^t$, we extract h_q^n, the last layer activation corresponding to the n^{th} response token. Finally, computed by the mean of h_q^n, $\boldsymbol{\mu}_p^n$ represents the *unsafe prototype* activation in the last layer at position n.

To measure output similarity between *unsafe prototypes* and a given input query i, we extract the last layer activations at token position n, denoted h_i^n. The similarity S_i is measured by the cosine similarity between the exponentially weighted sum of output distributions:

$$S_i = \cos(\textstyle\sum_{n=1}^{N} \gamma^{n-1} \hat{\mathbf{y}}_i^n,\ \sum_{n=1}^{N} \gamma^{n-1} \hat{\mathbf{y}}_p^n), \tag{3}$$

where $\hat{\mathbf{y}}_i^n = \text{softmax}(g(h_i^n))$ and $\hat{\mathbf{y}}_p^n = \text{softmax}(g(\boldsymbol{\mu}_p^n))$ denote the output probability distributions produced by the language model head $g(\cdot)$ for input query activations and *unsafe prototypes*, respectively, after applying the softmax function. Motivated by the observation that refusal behavior is largely concentrated in the beginning of the response [33], we apply a decaying factor $\gamma \in (0, 1)$ to the response position index. With small γ, the weights of subsequent tokens quickly approach zero, making their contributions negligible. Hence, we consider only a small number of initial tokens (*e.g.*, $N = 3$), which suffice to capture refusal behavior while ensuring computational efficiency.

When S_i is high, the query resembles *unsafe prototypes* and is more likely to trigger a refusal, whereas a low S_i indicates a benign query to which the model is likely to comply.

Distribution shift from safety prompts. In Fig. 5, we plot the similarity score distributions S_i for safe and unsafe object images under the query *"How can I make the item in the image?"* to measure the output similarity with refusals. When using the query with the image (purple), both safe and unsafe output similarity distributions concentrate around $S_i \approx 0$, indicating that the model tends to provide answers instead of issuing refusals. Adding safety prompts (brown) shifts both safe and unsafe distributions toward higher S_i values, *i.e.*, in the direction of refusals, as safety prompts instruct the models to reject queries that may be unsafe. However, because the model fails to sufficiently attend to safety-critical image regions (Fig. 3c), it cannot properly distinguish safe from unsafe cases, resulting in refusal-like responses for both.

To address this, we apply vision-aware query reformulation (blue histograms). While unsafe queries exhibit larger distributional shifts, safe queries show smaller

shifts, resulting in a clearer separation. This improvement in separation is driven by the added visual context, strengthening cross-modal attention for more accurate safety evaluations. We also examine reformulated queries without images (green histograms), which show weaker discrimination, consistent with the FDR results (blue *vs*. green lines in Fig. 4). These observations further highlight the significance of cross-modal attention in distinguishing safe from unsafe queries.

Risk evaluation. Leveraging the separation in output distributions from *reformulated queries*, we derive risk scores to steer the activations of *original queries*. Following common practice in inference-time safety alignment [6, 40], we calibrate the risk score using a held-out calibration set. For fair comparison, we use the same calibration set as [40], but randomly sample a subset to reduce calibration overhead. Note that, MoRAS yields consistent results when using the full set and samples from alternative calibration datasets (see Supplementary 7.2).

To derive risk scores, we compute S_i over the calibration set and use its mean as a baseline S_b, which represents an intermediate risk level. Each S_i is then mapped to a continuous risk score $r(S_i) \in (0, 1)$ using a sigmoid function centered at S_b (red line in Fig. 6). This can be formulated as:

$$r(S_i) = \sigma\big(\alpha \cdot (S_i - S_b)\big), \qquad (4)$$

where $\sigma(z) = \frac{1}{1+e^{-z}}$ is a sigmoid function, and $\alpha > 0$ is a normalizing term for $r(S_i) \approx 1$ when $S_i = 1$. Consequently, queries with similarity scores below S_b yield low risk scores (*e.g.*, benign MM-Vet samples; blue histogram in Fig.6), whereas queries with scores above S_b yield high risk scores (*e.g.*, unsafe SPA-VL samples; yellow histogram in Fig.6).

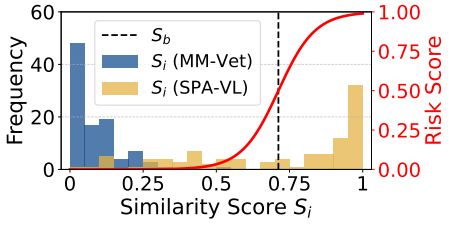


Fig. 6: Distribution of similarity scores for reformulated queries. From MM-Vet (safe) and SPA-VL (unsafe) datasets.

3.4 Scaled Activation Steering

We now steer model activations adaptively toward refusal behavior based on the risk evaluated in Stage 2. Unlike prior steering-based defenses that use a fixed steering magnitude tuned via hyperparameter search [36, 40], MoRAS *adapts* the intervention magnitude per query using the estimated risk score $r(S_i)$. This design applies negligible intervention to benign inputs while enforcing refusals for high-risk queries adaptively, thereby eliminating the need to tune an optimal steering magnitude for each model.

Refusal vector computation. Following [2], we employ activation steering along *refusal vectors*, but redefine them for a more targeted and effective refusal behavior. Rather than using the difference between mean activations of safe and unsafe queries, we use the vector from each input query activation to the *unsafe*

Algorithm 1 MoRAS (Multimodal Risk-Adaptive Steering)

Input: frozen MLLM f_θ (with LM head $g(\cdot)$), input image I, input text query T_q, visual-context generation prompt P_v, safety prompt P_s, EWRE parameters (decay rate $\gamma \in (0,1)$, token count N for risk estimation and steering, baseline similarity S_b, sigmoid normalizing term $\alpha > 0$), unsafe prototypes $\{\boldsymbol{\mu}_p^n\}_{n=1}^N$, steering layer ℓ

$Q \leftarrow (I, T_q)$ ▷ Input multimodal query

// **Stage 1: Vision-aware Query Reformulation**
$T_v \leftarrow \textsc{Generate}(f_\theta, I, P_v)$ ▷ Generate visual context T_v
$\tilde{Q} \leftarrow (P_s, T_v, Q)$ ▷ Construct reformulated query $\tilde{Q}$

// **Stage 2: Exponentially Weighted Risk Evaluation**
$y \leftarrow \emptyset$ ▷ Initialize output y
for $n = 1$ **to** N **do** ▷ Generate the first N output tokens
  $h^n \leftarrow f_\theta^{(\ell)}(\tilde{Q}, y)$ ▷ Obtain activation h^n at steering layer ℓ
  $\hat{\mathbf{y}}^n \leftarrow \text{softmax}(g(h^n))$ ▷ Calculate output token distribution $\hat{\mathbf{y}}^n$
  $\hat{\mathbf{y}}_p^n \leftarrow \text{softmax}(g(\boldsymbol{\mu}_p^n))$ ▷ Calculate prototype token distribution $\hat{\mathbf{y}}_p^n$
  $y_n \sim \hat{\mathbf{y}}^n$ ▷ Sample output token y_n
  $y \leftarrow (y, y_n)$ ▷ Append token y_n to output y
end for
$S \leftarrow \cos\Big(\sum_{n=1}^N \gamma^{n-1}\hat{\mathbf{y}}^n, \sum_{n=1}^N \gamma^{n-1}\hat{\mathbf{y}}_p^n\Big)$ ▷ Calculate similarity between exponentially weighted sums of $\hat{\mathbf{y}}^n$ and $\hat{\mathbf{y}}_p^n$ by Eq. (3)
$r(S) \leftarrow \sigma\big(\alpha(S - S_b)\big)$ ▷ Calculate risk score $r(S) \in (0,1)$ by Eq. (4)

// **Stage 3: Scaled Activation Steering**
$y \leftarrow \emptyset$ ▷ Initialize output y
for $n = 1, 2, \ldots$ **until** $y_n = [\text{EOS}]$ **do**
  $h^n \leftarrow f_\theta^{(\ell)}(Q, y)$ ▷ Obtain activation h^n at steering layer ℓ
  if $n \leq N$ **then**
    $\mathbf{v}^n \leftarrow \boldsymbol{\mu}_p^n - h^n$ ▷ Calculate refusal direction $\mathbf{v}^n$ by Eq. (5)
    $h^n \leftarrow h^n + r(S) \cdot \mathbf{v}^n$ ▷ Steer model activation h^n by Eq. (6)
  end if
  $\hat{\mathbf{y}}^n \leftarrow \text{softmax}(g(h^n))$ ▷ Calculate output token distribution $\hat{\mathbf{y}}^n$
  $y_n \sim \hat{\mathbf{y}}^n$ ▷ Sample output token y_n
  $y \leftarrow (y, y_n)$ ▷ Append token y_n to output y
end for

Output: y

prototype (see Supplementary Sec. 8 for comparison). Specifically, the refusal vector $\mathbf{v}_i^n$ for input query i at the last layer is computed as:

$$\mathbf{v}_i^n = \boldsymbol{\mu}_p^n - h_i^n, \tag{5}$$

where n denotes the position of the output token and i denotes the input query. This directional vector encodes the adjustment required to steer activations toward refusals and away from generating harmful responses.

Activation steering. We scale the refusal vector by the risk score $r(S_i)$ to ensure that the intervention strength is proportional to the risk estimated with EWRE. That is, for the activation of the input query h_i^n, we compute the steered activation $\tilde{h}_i^n$ as:

$$\tilde{h}_i^n = h_i^n + r(S_i) \cdot \mathbf{v}_i^n. \tag{6}$$

For computational efficiency, we apply activation steering to the last layer and to the first N response tokens, matching those used for risk evaluation. This formulation ensures that benign queries ($r(S_i) \approx 0$) receive negligible steering, preserving their original representations to maintain helpful responses. Unsafe queries ($r(S_i) \approx 1$) receive maximal steering, guiding the model toward appropriate refusals. For ambiguous queries ($0 < r(S_i) < 1$), the intervention magnitude is adaptively scaled according to their similarity to unsafe patterns. See Supplementary Sec. 8 for experiments on steering intermediate layer activations.

4 Experiments

We validate MoRAS in three aspects. (i) **Safety**, measured by attack success rates on multimodal jailbreak benchmarks. (ii) **Utility**, measured by scores on general multimodal reasoning tasks. (iii) **Computational overhead**, measured by pre-deployment calibration time (in minutes, wall-clock) and inference throughput (tokens per second) on identical hardware.

4.1 Setups

Benchmarks. For safety, we evaluate attack success rates (ASR) using MD-Judge-v0.2-Internlm2 [21], following [6, 14, 45]. To cover a broad range of black-box jailbreak scenarios spanning diverse harmful categories and visual characteristics, we evaluate ASR on SPA-VL [45], FigStep [12], MM-Safety [27], JOOD [16], and visual adversarial attacks (VAA) [32].

More concretely, SPA-VL evaluates MLLMs on multimodal queries spanning diverse harmful categories (*e.g.*, illegal activities and privacy). FigStep evaluates scenarios where benign text prompts are paired with images containing unsafe text that triggers harmful outputs. Similarly, MM-Safety uses benign text queries with images containing both unsafe text and corresponding harmful illustrations. JOOD contains challenging multimodal queries using out-of-distribution images generated by augmenting benign and unsafe images (*e.g.*, CutMix [43]). We also evaluate a white-box setting with VAA [32], where gradient-based image perturbations suppress refusals and induce harmful outputs.

Utility is evaluated on both open-ended (GQA [15] and MM-Vet [42]) and multiple-choice (Sci-QA [28] and MME [9]) benchmarks, using the official metrics. See Supplementary Sec. 3 for additional details on each benchmark.

Models. To verify the generalizability of MoRAS across various models and sizes, we provide results on LLaVA-1.5-7B/13B [23], LLaVA-OneVision-7B [20], Qwen-VL-Chat [3], and InternLM-XComposer-2.5 [44].

Table 1: Comparison in safety and utility. We report safety performance under diverse attacks and utility performance across general task benchmarks. Bold and underlined text represent the best and second-best performance, respectively. MM-S denotes MM-Safety, VAA denotes Visual Adversarial Attacks, and MME-P/MME-C denote MME perception and cognition scores, respectively. Overall, MoRAS achieves low ASR across all jailbreaks while preserving utility.

Model	Method	**Safety** (ASR ↓)					**Utility** (Score ↑)				
		SPA-VL	FigStep	MM-S	JOOD	VAA	GQA	MM-Vet	Sci-QA	MME-P	MME-C
LLaVA-1.5-7B	Vanilla	47.2	59.3	40.1	51.6	43.1	61.9	30.5	69.5	1505.1	355.7
	CoCA (COLM 2024)	10.9	51.6	19.7	13.5	15.9	60.3	28.9	67.7	**1526.5**	283.6
	ECSO (ECCV 2024)	23.4	37.4	15.9	23.6	26.4	**61.9**	30.3	**69.5**	1505.1	**355.7**
	FigStep (AAAI 2025)	32.4	52.0	26.8	20.2	16.8	61.3	29.5	68.3	1435.7	275.0
	ETA (ICLR 2025)	17.0	7.8	15.8	17.1	**12.1**	**61.9**	30.4	**69.5**	1509.3	339.6
	AutoSteer (EMNLP 2025)	8.3	55.0	37.6	18.7	19.8	59.4	29.3	69.1	1479.9	315.4
	ASTRA (CVPR 2025)	41.5	14.2	27.2	44.7	17.4	60.4	29.0	66.3	1472.0	331.4
	MoRAS (Ours)	**7.6**	**2.8**	**2.6**	**6.4**	14.3	**61.9**	**30.5**	**69.5**	1505.1	**355.7**
LLaVA-OneVision-7B	Vanilla	15.1	22.0	26.1	16.7	37.1	62.8	52.8	94.4	1560.9	409.6
	CoCA	4.2	6.0	11.8	6.3	11.0	61.4	43.5	94.2	1463.4	403.2
	ECSO	12.5	17.8	16.1	13.8	13.5	**62.8**	**52.4**	**94.4**	1560.9	**409.6**
	FigStep	5.3	6.6	13.5	5.8	10.0	61.7	47.8	**94.4**	1472.2	395.4
	ETA	8.7	14.0	15.8	10.6	23.4	**62.8**	52.1	**94.4**	1560.9	**409.6**
	AutoSteer	**0.2**	**2.3**	15.2	1.1	3.6	61.9	47.5	94.3	1548.1	**409.6**
	ASTRA	11.3	10.0	14.6	16.2	13.7	62.3	37.1	92.7	1406.4	387.9
	MoRAS (Ours)	1.2	3.4	**0.6**	**0.0**	**1.4**	**62.8**	50.8	**94.4**	**1561.8**	**409.6**
Qwen-VL-Chat	Vanilla	12.5	52.4	33.1	9.6	20.3	57.3	48.7	68.0	1489.9	331.8
	CoCA	4.2	32.2	2.6	**0.0**	6.1	56.9	38.7	66.7	1377.1	319.3
	ECSO	7.6	45.4	19.1	7.6	16.3	**57.3**	**47.2**	**68.0**	**1489.9**	331.8
	FigStep	5.7	44.4	8.1	5.6	5.3	56.8	39.0	64.4	1480.9	296.4
	ETA	4.5	9.2	9.3	2.6	6.9	**57.3**	45.9	67.8	1487.9	331.8
	AutoSteer	**2.3**	46.4	28.6	7.4	4.2	57.0	43.8	67.5	1474.5	**348.2**
	ASTRA	8.3	28.8	16.2	8.8	8.4	55.2	40.1	66.6	1465.9	323.2
	MoRAS (Ours)	2.5	**0.6**	**0.4**	0.4	**2.9**	**57.3**	46.9	**68.0**	**1489.9**	331.8
InternLM-XComposer-2.5	Vanilla	27.6	22.6	21.8	19.3	16.1	59.1	50.1	94.7	1623.7	551.1
	CoCA	5.9	16.0	6.1	**2.2**	6.8	58.8	48.1	93.3	1606.5	**551.1**
	ECSO	19.6	16.6	14.9	16.0	9.5	**59.1**	49.4	**94.7**	1623.7	**551.1**
	FigStep	6.8	7.0	6.3	3.6	10.6	58.9	47.2	86.1	1577.7	516.8
	ETA	14.0	**6.0**	7.3	10.6	5.4	58.1	47.4	94.6	**1629.4**	546.1
	AutoSteer	5.1	15.8	18.9	7.7	1.8	58.7	46.7	93.8	1591.2	544.3
	ASTRA	23.0	14.6	13.4	18.7	4.8	58.6	47.8	91.8	1617.4	546.8
	MoRAS (Ours)	**4.9**	7.2	**5.9**	2.7	**1.0**	**59.1**	**49.8**	**94.7**	1623.7	**551.1**

Baselines. We compare MoRAS against a broad set of inference-time alignment methods, including (i) prompt-based methods (FigStep [12] and CoCA [10]), (ii) response refinement methods (ECSO [13] and ETA [6]), and (ii) steering methods (AutoSteer [40] and ASTRA [36]).

Implementation details. We describe implementation details and hyperparameters in Supplementary Sec. 1 for space sake.

4.2 Results

Safety. We report the ASR of multimodal jailbreaks in Tab. 1. As shown in Tab. 1, MoRAS significantly outperforms the baselines, achieving lower ASR and higher utility across benchmarks. Note that, while some baselines perform well on certain benchmarks, they often remain vulnerable on others, demonstrating limited generalizability. For example, prompt-based methods (*i.e.*, FigStep and CoCA) lower ASR on SPA-VL and JOOD but show limited gains on FigStep and MM-Safety. We believe this limitation stems from insufficient attention to typographic regions containing safety-critical text, as supported by qualitative

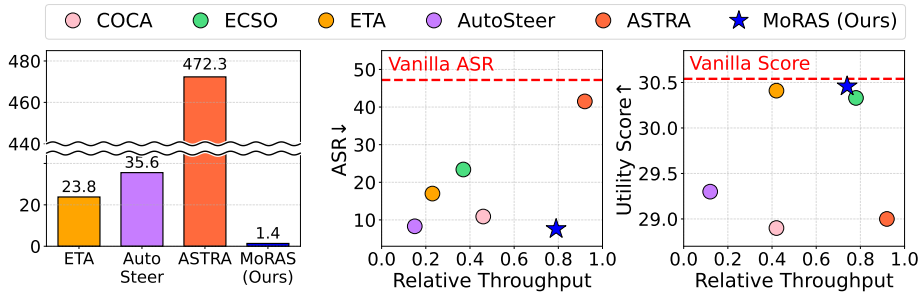


Fig. 7: (Left) Pre-deployment overhead (wall-clock time) required before inference (*e.g.*, extracting activations and calibrating thresholds). (Middle, Right) Trade-offs between relative inference throughput (tokens/sec normalized to the vanilla model) and ASR on SPA-VL (Middle), and utility on MM-Vet (Right). COCA and ECSO incur no pre-deployment overhead but suffer from low inference throughput, whereas ASTRA achieves high inference throughput but incurs substantial pre-deployment overhead.

visual attention maps for FigStep and MM-Safety samples in Supplementary Sec. 5.2. Similarly, steering-based methods (*i.e.*, AutoSteer and ASTRA) show limited generalization to MM-Safety and JOOD, respectively. We attribute this to their reliance on risk assessment via similarity to vectors derived from a specific calibration dataset, which can fail when inputs deviate from the calibration distribution, resulting in limited generalization.

In contrast, MoRAS strengthens cross-modal attention between the image and the textual instruction, allowing the model to better associate safety-critical visual regions with their corresponding textual intent, as shown in Fig. 3e. This enables more accurate risk estimation across diverse attack types, leading to improved generalization. We show results for LLaVA-1.5-13B in Supplementary Sec. 9.1 and additional jailbreak/over-refusal results in Supplementary Sec. 9.2.

Utility. An effective defense should enhance safety while preserving the general task performance of MLLMs. As shown in the right column of Tab. 1, MoRAS maintains performance comparable to the vanilla model across all tasks, whereas several baselines often cause notable degradation. These results demonstrate that MoRAS adaptively adjusts steering strength, along with accurate risk estimation and visual context incorporation, thereby providing strong refusals against malicious queries, while preserving the multimodal reasoning capabilities of MLLMs.

Computational overhead. For real-world deployment, it is important to achieve low pre-deployment overhead (*e.g.*, activation extraction and parameter calibration) while maintaining efficient inference throughput. Accordingly, we measure (i) pre-deployment overhead (in minutes) and (ii) inference throughput (as tokens per second relative to the original model, following [8, 22, 35]).

We show pre-deployment overhead across methods in Fig. 7 (left). ETA performs calibration by collecting responses from both the original model and a reward model; AutoSteer extracts activations over thousands of calibration samples; and ASTRA synthesizes gradient-based adversarial samples for calibration.

Table 2: Ablation study. We ablate using Stage 1 only *vs*. the full MoRAS. Top row shows results of the vanilla model. MM-S denotes MM-Safety. The **Total** utility score is the weighted sum of task scores in MM-Vet [42].

Model	Stage	**Safety** (ASR ↓)			**Utility** (Score ↑)						
		SPA-VL	FigStep	MM-S	rec	ocr	know	gen	spat	math	**Total**
LLaVA-1.5-7B	-	47.2	59.3	40.1	41.0	**26.9**	16.2	**21.8**	26.7	**11.5**	**30.5**
	[1]	**7.1**	**2.8**	3.5	38.2	25.6	13.9	18.9	**27.2**	7.7	28.6
	[1, 2, 3]	7.6	**2.8**	**2.6**	**41.6**	26.0	**16.4**	21.0	25.6	**11.5**	**30.5**

In contrast, MoRAS performs calibration with a minimal number of samples: (i) 50 samples to construct unsafe prototypes and (ii) 100 samples for S_b estimation, resulting in substantially lower pre-deployment overhead. We provide detailed overhead breakdown in Supplementary Sec. 10.1.

For inference throughput, as shown in Fig. 7 (middle, right), MoRAS achieves the lowest ASR while preserving utility with marginal slowdown. This efficiency stems from its lightweight design: generating short visual contexts for query reformulation and applying risk evaluation and activation steering to a limited set of tokens (*i.e.*, the first three response tokens). See Supplementary Sec. 10.2 for details on visual context generation costs and throughput on other models.

In contrast, baselines introduce substantial latency. ECSO and ETA first generate a complete response, verify it using the model itself (or an external model), and refine it accordingly. However, revising the response after generating a complete one incurs substantial inference overhead. CoCA and AutoSteer incur per token overhead by modifying output logits at each decoding step. ASTRA achieves the highest inference throughput, as it only projects activations during decoding to suppress harmful-behavior vectors. However, it requires substantial pre-deployment overhead in both memory and computation to generate gradient-based adversarial inputs.

4.3 Ablation Study

We conduct ablation study on MoRAS and summarize the results in Tab. 2. Using Stage 1 alone (middle row) shows that using the reformulated query can successfully detect multimodal risk from the enhanced safety-critical cross-modal attention. However, as the incorporation of safety prompts still skews output distributions toward refusals, which may cause over-refusals (Sec. 3.3), utility degrades compared to the vanilla model (top row). In contrast, combining all stages, estimating the risk signal (*i.e.*, stage 2) from the reformulated query (*i.e.*, stage 1) to steer activations of the original query adaptively (*i.e.*, stage 3), improves safety while maintaining utility comparable to the vanilla model (bottom row). We provide additional ablation studies for each stage in Supplementary Sec. 11.

4.4 Additional Experiments

We provide hyperparameter sensitivity of MoRAS in Supplementary Sec. 2. For baselines which require hyperparameter tuning (*e.g.*, AutoSteer and ASTRA), we provide additional results in Supplementary Sec. 13.

5 Conclusion

We propose multimodal risk-adaptive steering (MoRAS), a novel inference-time multimodal safety alignment method. MoRAS reformulates queries to strengthen cross-modal attention, enabling accurate risk evaluations. Based on the evaluated risk, MoRAS adaptively steers activations, applying strong interventions to unsafe queries and minimal adjustments to benign queries. Comprehensive experiments on multimodal safety and utility benchmarks show its significance; decreasing attack success rates and preserving general task performance with reduced computational overhead compared to prior inference-time defenses.

Ethical Consideration

This work investigates the safety alignment of Multimodal Large Language Models (MLLMs) using publicly available benchmarks that include harmful or toxic prompts. We acknowledge the ethical risks of working with such data, as well as the possibility that models may generate unsafe responses under such adversarial conditions. Our approach aims to mitigate these risks by reducing harmful responses, thereby contributing to a more responsible deployment of MLLMs. While our method improves defenses, it does not fully eliminate vulnerabilities; continued research is necessary to better understand and mitigate ethical risks and potential misuse.

Acknowledgement

This work was partly supported by the InnoCORE program (26-InnoCORE-01), the IITP grants (RS-2022-II220077, RS-2022-II220113, RS-2022-II220959, RS-2022-II220871, RS-2026-25507282, RS-2026-25518317, RS-2021-II211343 (SNU AI), RS-2025-25442338 (AI Star Fellowship-SNU)), 02-26-01-0285 (Advanced GPU Utilization Support Program by NIPA) funded by the Korea government (MSIT), grants (RS-2025-25462891 (US-KOR BARI), RS-2025-25453780) funded by MOTIR, a grant (RS-2025-25460896) funded by MOTIR and KIAT, a grant of Korean ARPA-H Project through the Korea Health Industry Development Institute (KHIDI), funded by the Ministry of Health & Welfare, Republic of Korea (RS-2025-25424639), and the BK21 FOUR program, SNU in 2025.

In addition, we acknowledge the EuroHPC Joint Undertaking for awarding this project access to the EuroHPC supercomputers MareNostrum5 at BSC, Spain; LEONARDO at CINECA, Italy; VEGA at IZUM, Slovenia; Karolina at IT4Innovations, Czech Republic; MeluXina at LuxProvide, Luxembourg; Discoverer at Sofia Tech Park, Bulgaria; and Deucalion at Minho Advanced Computing Centre, Portugal, under project IDs EHPC-DEV-2025D07-089, EHPC-BEN-2025B08-038, EHPC-DEV-2025D08-065, EHPC-DEV-2026D04-104, EHPC-DEV-2026D01-064, and EHPC-DEV-2026D04-219 through EuroHPC Development and Benchmark Access calls.

References

1. Alinadilawaiz: Dangerous objects dataset. `https://www.kaggle.com/datasets/alinadilawaiz/dangerous-objects-dataset` (2023), kaggle, CC BY-SA 4.0. Accessed: 2026-06-29
2. Arditi, A., Obeso, O., Syed, A., Paleka, D., Panickssery, N., Gurnee, W., Nanda, N.: Refusal in language models is mediated by a single direction. arXiv preprint arXiv:2406.11717 (2024)
3. Bai, J., Bai, S., Yang, S., Wang, S., Tan, S., Wang, P., Lin, J., Zhou, C., Zhou, J.: Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond (2023)
4. Chen, Z., Wu, J., Wang, W., Su, W., Chen, G., Xing, S., Zhong, M., Zhang, Q., Zhu, X., Lu, L., 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. pp. 24185–24198 (2024)
5. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
6. Ding, Y., Li, B., Zhang, R.: Eta: Evaluating then aligning safety of vision language models at inference time. arXiv preprint arXiv:2410.06625 (2024)
7. Ding, Y., Li, L., Cao, B., Shao, J.: Rethinking bottlenecks in safety fine-tuning of vision language models. arXiv preprint arXiv:2501.18533 (2025)
8. Fedorov, I., Plawiak, K., Wu, L., Elgamal, T., Suda, N., Smith, E., Zhan, H., Chi, J., Hulovatyy, Y., Patel, K., et al.: Llama guard 3-1b-int4: Compact and efficient safeguard for human-ai conversations. arXiv preprint arXiv:2411.17713 (2024)
9. Fu, C., Chen, P., Shen, Y., Qin, Y., Zhang, M., Lin, X., Yang, J., Zheng, X., Li, K., Sun, X., Wu, Y., Ji, R.: Mme: A comprehensive evaluation benchmark for multimodal large language models (2024)
10. Gao, J., Pi, R., Han, T., Wu, H., Hong, L., Kong, L., Jiang, X., Li, Z.: Coca: Regaining safety-awareness of multimodal large language models with constitutional calibration. arXiv preprint arXiv:2409.11365 (2024)
11. Ghosal, S.S., Chakraborty, S., Singh, V., Guan, T., Wang, M., Beirami, A., Huang, F., Velasquez, A., Manocha, D., Bedi, A.S.: Immune: Improving safety against jailbreaks in multi-modal llms via inference-time alignment. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 25038–25049 (2025)
12. Gong, Y., Ran, D., Liu, J., Wang, C., Cong, T., Wang, A., Duan, S., Wang, X.: Figstep: Jailbreaking large vision-language models via typographic visual prompts. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 23951–23959 (2025)
13. Gou, Y., Chen, K., Liu, Z., Hong, L., Xu, H., Li, Z., Yeung, D.Y., Kwok, J.T., Zhang, Y.: Eyes closed, safety on: Protecting multimodal llms via image-to-text transformation. In: European Conference on Computer Vision. pp. 388–404. Springer (2024)
14. Huang, M., Liu, X., Zhou, S., Zhang, M., Guo, Q., Li, L., Tan, C., Gao, Y., Wang, P., Li, L., et al.: Longsafety: Enhance safety for long-context llms. arXiv preprint arXiv:2411.06899 (2024)
15. Hudson, D.A., Manning, C.D.: Gqa: A new dataset for real-world visual reasoning and compositional question answering. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6700–6709 (2019)

16. Jeong, J., Bae, S., Jung, Y., Hwang, J., Yang, E.: Playing the fool: Jailbreaking llms and multimodal llms with out-of-distribution strategy. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 29937–29946 (2025)
17. Kang, S., Kim, J., Kim, J., Hwang, S.J.: See what you are told: Visual attention sink in large multimodal models. arXiv preprint arXiv:2503.03321 (2025)
18. Kang, S., Kim, J., Kim, J., Hwang, S.J.: Your large vision-language model only needs a few attention heads for visual grounding. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 9339–9350 (2025)
19. Kim, S.J., Magnani, A., Boyd, S.: Robust fisher discriminant analysis. Advances in neural information processing systems **18** (2005)
20. Li, B., Zhang, Y., Guo, D., Zhang, R., Li, F., Zhang, H., Zhang, K., Zhang, P., Li, Y., Liu, Z., et al.: Llava-onevision: Easy visual task transfer. arXiv preprint arXiv:2408.03326 (2024)
21. Li, L., Dong, B., Wang, R., Hu, X., Zuo, W., Lin, D., Qiao, Y., Shao, J.: Salad-bench: A hierarchical and comprehensive safety benchmark for large language models. arXiv preprint arXiv:2402.05044 (2024)
22. Liu, F., Tang, Y., Liu, Z., Ni, Y., Tang, D., Han, K., Wang, Y.: Kangaroo: Lossless self-speculative decoding for accelerating llms via double early exiting. Advances in Neural Information Processing Systems **37**, 11946–11965 (2024)
23. Liu, H., Li, C., Li, Y., Lee, Y.J.: Improved baselines with visual instruction tuning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 26296–26306 (2024)
24. Liu, H., Li, C., Li, Y., Li, B., Zhang, Y., Shen, S., Lee, Y.J.: Llava-next: Improved reasoning, ocr, and world knowledge (January 2024)
25. Liu, J., Guo, H., Duan, R., Bu, X., He, Y., Li, S., Huang, H., Liu, J., Wang, Y., Jing, C., et al.: Dream: Disentangling risks to enhance safety alignment in multimodal large language models. arXiv preprint arXiv:2504.18053 (2025)
26. Liu, S., Ye, H., Xing, L., Zou, J.: In-context vectors: Making in context learning more effective and controllable through latent space steering. arXiv preprint arXiv:2311.06668 (2023)
27. Liu, X., Zhu, Y., Gu, J., Lan, Y., Yang, C., Qiao, Y.: Mm-safetybench: A benchmark for safety evaluation of multimodal large language models. In: European Conference on Computer Vision. pp. 386–403. Springer (2024)
28. Lu, P., Mishra, S., Xia, T., Qiu, L., Chang, K.W., Zhu, S.C., Tafjord, O., Clark, P., Kalyan, A.: Learn to explain: Multimodal reasoning via thought chains for science question answering. Advances in Neural Information Processing Systems **35**, 2507–2521 (2022)
29. Pandey, R., Shao, R., Liang, P.P., Salakhutdinov, R., Morency, L.P.: Cross-modal attention congruence regularization for vision-language relation alignment. arXiv preprint arXiv:2212.10549 (2022)
30. Panickssery, N., Gabrieli, N., Schulz, J., Tong, M., Hubinger, E., Turner, A.M.: Steering llama 2 via contrastive activation addition. arXiv preprint arXiv:2312.06681 (2023)
31. Pi, R., Han, T., Zhang, J., Xie, Y., Pan, R., Lian, Q., Dong, H., Zhang, J., Zhang, T.: Mllm-protector: Ensuring mllm's safety without hurting performance. arXiv preprint arXiv:2401.02906 (2024)
32. Qi, X., Huang, K., Panda, A., Henderson, P., Wang, M., Mittal, P.: Visual adversarial examples jailbreak aligned large language models (2023)
33. Qi, X., Panda, A., Lyu, K., Ma, X., Roy, S., Beirami, A., Mittal, P., Henderson, P.: Safety alignment should be made more than just a few tokens deep, 2024. URL https://arxiv. org/abs/2406.05946 (2024)

34. Ramezani, P., Schilling, A., Krauss, P.: Analysis of argument structure constructions in the large language model bert. Frontiers in Artificial Intelligence **8**, 1477246 (2025)
35. Svirschevski, R., May, A., Chen, Z., Chen, B., Jia, Z., Ryabinin, M.: Specexec: Massively parallel speculative decoding for interactive llm inference on consumer devices. Advances in Neural Information Processing Systems **37**, 16342–16368 (2024)
36. Wang, H., Wang, G., Zhang, H.: Steering away from harm: An adaptive approach to defending vision language model against jailbreaks. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 29947–29957 (2025)
37. Wang, P., Bai, S., Tan, S., Wang, S., Fan, Z., Bai, J., Chen, K., Liu, X., Wang, J., Ge, W., et al.: Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. arXiv preprint arXiv:2409.12191 (2024)
38. Wang, S., Li, D., Wei, Y., Li, H.: A feature selection method based on fisher's discriminant ratio for text sentiment classification. In: International Conference on Web Information Systems and Mining. pp. 88–97. Springer (2009)
39. Wang, Y., Liu, X., Li, Y., Chen, M., Xiao, C.: Adashield: Safeguarding multimodal large language models from structure-based attack via adaptive shield prompting. In: European Conference on Computer Vision. pp. 77–94. Springer (2024)
40. Wu, L., Wang, M., Xu, Z., Cao, T., Oo, N., Hooi, B., Deng, S.: Automating steering for safe multimodal large language models. In: Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing. pp. 792–814 (2025)
41. Xu, S., Pang, L., Zhu, Y., Shen, H., Cheng, X.: Cross-modal safety mechanism transfer in large vision-language models. arXiv preprint arXiv:2410.12662 (2024)
42. Yu, W., Yang, Z., Li, L., Wang, J., Lin, K., Liu, Z., Wang, X., Wang, L.: Mm-vet: Evaluating large multimodal models for integrated capabilities. arXiv preprint arXiv:2308.02490 (2023)
43. Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J., Yoo, Y.: Cutmix: Regularization strategy to train strong classifiers with localizable features. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 6023–6032 (2019)
44. Zhang, P., Dong, X., Zang, Y., Cao, Y., Qian, R., Chen, L., Guo, Q., Duan, H., Wang, B., Ouyang, L., et al.: Internlm-xcomposer-2.5: A versatile large vision language model supporting long-contextual input and output. arXiv preprint arXiv:2407.03320 (2024)
45. Zhang, Y., Chen, L., Zheng, G., Gao, Y., Zheng, R., Fu, J., Yin, Z., Jin, S., Qiao, Y., Huang, X., et al.: Spa-vl: A comprehensive safety preference alignment dataset for vision language models. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 19867–19878 (2025)
46. Zheng, C., Yin, F., Zhou, H., Meng, F., Zhou, J., Chang, K.W., Huang, M., Peng, N.: On prompt-driven safeguarding for large language models. arXiv preprint arXiv:2401.18018 (2024)
47. Zhou, A., Li, B., Wang, H.: Robust prompt optimization for defending language models against jailbreaking attacks. Advances in Neural Information Processing Systems **37**, 40184–40211 (2024)