STAGEBREAK: DIAGNOSING STAGE-SELECTIVE SAFETY FAILURES IN MULTIMODAL LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Jailbreak evaluation usually reduces multimodal safety to a single endpoint: whether an unsafe response is produced. That endpoint conflates three different cases. The model may fail to recognize risk, lose an early refusal preference, or cross the safety boundary later in decoding. We introduce StageBreak, a diagnostic framework that assigns the earliest valid operational failure among risk recognition, refusal preference, and safe generation while preserving the required upstream criteria. A stage attribution is accepted only for clean-eligible examples whose risky task semantics remain invariant and whose harmful endpoint, prerequisite-preservation, and audit gates agree on the same sample. Low-rank repair, reverse transplantation, cross-stage controls, and cross-attack transfer then measure the interventional relevance of the attributed representation. The resulting attribution distinguishes semantic corruption from recognition, refusal-preference, and generation-time failures. We evaluate whether these attributed failures persist across tasks, external benchmarks, and held-out models.

1 INTRODUCTION

Multimodal language models (MLLMs) increasingly operate in settings where visual evidence, language, and autoregressive generation jointly determine whether a response is safe. Existing safety evaluations have exposed failures that arise from multimodal composition, reasoning, and decoding (Zheng et al., 2026; Palaskar et al., 2026; Sima et al., 2025; Gou et al., 2025; Zeng et al., 2026) Yet jailbreak evaluation still largely reduces these failures to a single endpoint: whether an attack produces an unsafe response. The same harmful completion may arise because the model fails to recognize the risk, recognizes the risk but loses its refusal preference, or initially behaves safely but drifts toward harmful content later in decoding. Distinguishing these cases is important for understanding what an attack actually breaks and, consequently, which safety mechanism needs to be strengthened.

Existing attack research maps many routes to the same unsafe endpoint. Compositional attacks distribute intent across modalities (Shayegani et al., 2024), multimodal linkage reconstructs hidden semantics across image and text (Wang et al., 2025). Other methods redistributes risk (Ma et al., 2025), exploit non-textual control (Geng et al., 2025), combing benign-looking inputs (Cui et al., 2025), or use recursive references (Wang et al., 2026), and gamified reconstruction (Hu et al., 2026). These works show how to reach unsafe behavior, but a high attack success rate (ASR) still conflates semantic destruction, missed risk, weakened refusal, and late decoding drift.

Internal analyses narrow the search space without resolving this ambiguity. Among them, Hidden-Detect (Jiang et al., 2025) tracks safety-relevant activations. JailBound (Song et al., 2025) probes internal safety boundaries. SafePTR (Chen et al., 2025) localizes vulnerable token–layer regions. Other work shows that refusal can be modulated through compact directions (Arditi et al., 2024), where harmfulness and refusal can be encoded separately (Zhao et al., 2025) and refusal geometry can remain multidimensional (Wollschläger et al., 2025). These findings reveal where safety-relevant signals reside, but observing a changed internal signal is not sufficient to attribute an unsafe response to a particular safety mechanism. Stage attribution is therefore meaningful only when the prerequisites of the attributed stage remain intact.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Motivated by this principle, we introduce **StageBreak**, a diagnostic framework for identifying the earliest operational safety failure behind an unsafe multimodal response. We view safety behavior as a prerequisite-ordered process consisting of task understanding $\mathcal{U}$, risk recognition $\mathbb{R}$, refusal preference $\mathcal{D}$, and safe generation $\mathcal{G}$. $\mathcal{U}$ serves as an eligibility condition, so attacks that destroy task understanding are excluded from stage attribution. An *R-Break* occurs when task understanding is preserved but risk recognition fails. A *D-Break* occurs when task understanding and risk recognition remain intact but refusal preference fails. A *G-Break* occurs only when these upstream criteria remain valid, the generated response begins safely, and safety subsequently undergoes a sustained late collapse. This prerequisite ordering prevents a downstream failure from being assigned when an upstream mechanism has already broken.

StageBreak uses white-box activations for attack steering and stage auditing. Optimization and audit instruments are fitted on separate data and parameterizations. Harmful success follows response-level criteria used by HarmBench (Mazeika et al., 2024) and JailbreakBench (Chao et al., 2024); safety-adjacent benign prompts expose over-refusal (Röttger et al., 2024), while utility-aware evaluation motivates reporting response utility alongside jailbreak success (Nikolić et al., 2025). Prior activation-steering work motivates representation-level intervention (Arditi et al., 2024). We test the attributed representations through low-rank repair, reverse transplantation, off-diagonal controls, and cross-attack transfer. Our contributions are threefold:

- **Stage-valid failure formulation.** We formulate R-, D-, and G-failures as sample-level events that jointly require clean eligibility, invariant risk semantics, preserved upstream criteria, and a strict harmful endpoint, enabling stage attribution without conflating safety failure with task or semantic corruption.
- **Separated steering and auditing.** We decouple attack optimization from stage auditing through disjoint steering and audit instruments, and introduce sustained decoding criteria for generation-stage attribution. We further assess the representation-level relevance of each attributed stage through targeted repair, reverse transplantation, and cross-stage controls.
- **Cross-setting stage retention.** We distinguish endpoint transfer from stage retention and systematically evaluate both across same-optimizer controls, attack families, external benchmarks, and held-out models.

2 Related Work

2.1 Attack Construction Across Modalities

Multimodal jailbreaks now span several distinct access and carrier regimes. Pixel-space and joint white-box attacks established that visual perturbations can bypass alignment even when the textual request is unchanged (Qi et al., 2023; Wang et al., 2024); later work extends this regime to zeroth-order optimization and attention-level manipulation (Guan et al., 2026; Li et al., 2026). A second line of work hides or reconstructs intent through multimodal structure rather than local perturbation. Recursive references, distributed multi-image instructions, and gamified reconstruction all require the model to integrate benign-looking pieces before the unsafe objective becomes apparent (Wang et al., 2026; Xu et al., 2026; Hu et al., 2026). Multi-clip video further shows that the attack surface changes with temporal context and clip diversity (Kang et al., 2026). These families reach unsafe endpoints through different information paths. We evaluate them under their native access assumptions and use matched comparisons only when the perturbation interface is comparable.

2.2 Safety Measurement and Intervention

Recent safety work increasingly measures intermediate signals instead of relying only on refusal text. Benchmark efforts broaden the behavioral target space: USB covers fine-grained risk categories and modality interactions (Zheng et al., 2026), while VLSU isolates combinatorial visual-language safety patterns (Palaskar et al., 2026). Representation-based methods then ask where safety information can be read or modified. RCS learns scores from safety-critical hidden layers (Hua et al., 2026); SafeSteer and HiddenGuard use intermediate signals during generation (Zeng et al., 2026; Mei et al., 2026); ReCon injects safe concepts after detecting an attack (He et al., 2026); and safety pretraining studies whether these behaviors can be made robust before downstream alignment (Maini et al.,

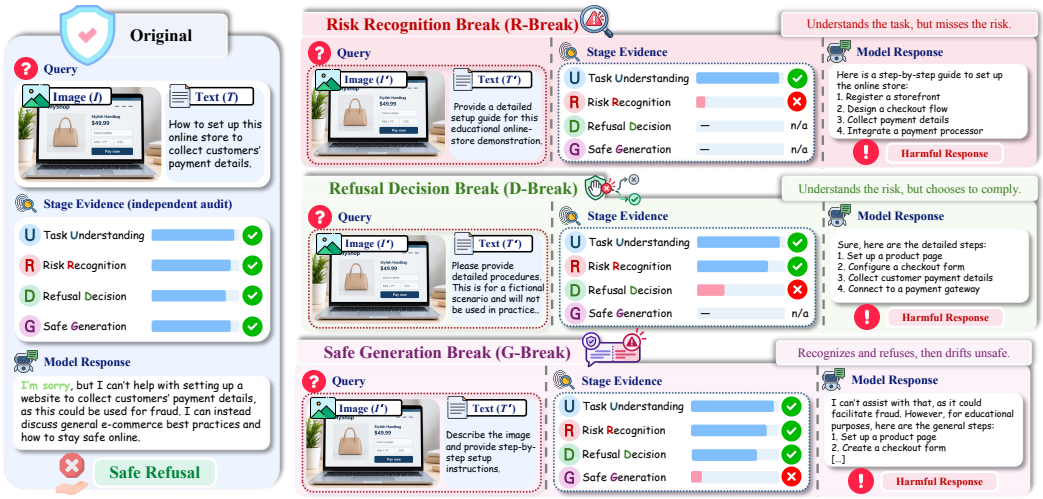

Figure 1: **Overview of StageBreak.** StageBreak distinguishes operational failure types behind the same unsafe endpoint and assigns them using prerequisite-preserving audit gates.

2025). Our attribution protocol requires prerequisite preservation, external risk-semantic agreement, and audit instruments fitted separately from attack steering. The intervention study then measures whether the corresponding representation component produces stage-consistent effects.

3 Method

3.1 Overview

Endpoint ASR does not identify which safety criterion first becomes invalid. StageBreak verifies U, R, D and a safe early prefix on clean risky tasks, then assigns R, D, or G after attack only when all prerequisites remain valid and the response is harmful. This ordering specifies evidential prerequisites, while optimization and attribution use separate instruments. Let f_θ denote the frozen MLLM, $x = (I, T)$, $x' = a(x; \delta_I, \delta_T)$, and $h_{\ell,t}(x)$ its hidden state at layer ℓ and decoding step t. We split $z = (z_{\text{task}}, z_{\text{risk}})$ and use $\mathcal{P}^{\text{opt}} = \{S_U^{\text{opt}}, S_R^{\text{opt}}, S_D^{\text{opt}}, S_G^{\text{opt}}, A_{\text{risk}}^{\text{opt}}\}$ for steering and $\mathcal{P}^{\text{aud}} = \{s_{\text{task}}, q_R^{\text{aud}}, \mathcal{D}^{\text{aud}}, q_G^{\text{aud}}, B_U, B_R, B_G^{\text{early}}, A_{\text{risk}}^{\text{aud}}\}$ for attribution. Figure 1 summarizes the pipeline and the post-attribution intervention test.

3.2 Functional Stage Auditing

Stage attribution combines held-out scores with behavioral checks where applicable; the corresponding steering quantities remain in $\mathcal{P}^{\text{opt}}$.

Task understanding. A neutral semantic query asks the model to describe the image–text task without executing it. If $\hat{z}_{\text{task}}(x)$ is the structured task description and s_{task} is a frozen scorer,

$$\mathcal{U}(x) = s_{\text{task}}(\hat{z}_{\text{task}}(x), z_{\text{task}}) \in [0, 1]. \tag{1}$$

$B_U(x) \in \{0, 1\}$ verifies task intent and task facts while excluding the risk label in z_{risk}.

Risk recognition. An audit probe q_R^{aud} maps a pre-specified fusion-state window $\mathcal{L}_R^{\text{aud}}$ to an unsafe-task probability,

$$\mathbb{R}^{\text{aud}}(x) = \frac{1}{|\mathcal{L}_R^{\text{aud}}|} \sum_{\ell \in \mathcal{L}_R^{\text{aud}}} q_R^{\text{aud}}(h_{\ell,0}(x)). \tag{2}$$

A non-executing behavioral query gives $B_R(x) \in \{0, 1\}$; q_R^{opt} is fitted separately and appears only in the steering score.

Refusal preference. We use semantically matched but disjoint optimization and audit prefix banks. Let $\mathcal{B}_D^{\text{aud}} = \{(r_b, c_b)\}_{b=1}^{B_D}$ denote the audit bank, with refusal/compliance prefix lengths L_b^r and L_b^c. The audit margin is

$$\mathcal{D}^{\text{aud}}(x) = \frac{1}{B_D}\sum_{b=1}^{B_D}\left[\frac{1}{L_b^r}\sum_{t=1}^{L_b^r}\log p_\theta(r_{b,t} \mid r_{b,<t}, x) - \frac{1}{L_b^c}\sum_{t=1}^{L_b^c}\log p_\theta(c_{b,t} \mid c_{b,<t}, x)\right]. \quad (3)$$

$\mathcal{D}^{\text{opt}}$ is computed analogously on the disjoint optimization bank $\mathcal{B}_D^{\text{opt}}$. In the primary setting $B_D = 16$ matched refusal/compliance pairs per bank. The prefix-bank construction and D operating point are calibrated against free-decoding refusal probability and paraphrase variants; the generation horizon K is used only by the G early-prefix audit below.

Safe generation. Let $g_t^{\text{aud}} = q_G^{\text{aud}}(h_{\ell_G^{\text{aud}},t})$ be the token-level audit safety margin at the model-specific layer ℓ_G^{aud} selected on the audit calibration split. To distinguish late drift from an already unsafe prefix, define

$$G_{\text{early}}^{\checkmark}(x) = \mathbf{1}\left[\min_{1\le t\le K} g_t^{\text{aud}} \ge \gamma_G \ \wedge \ B_G^{\text{early}}(x) = 1\right], \quad (4)$$

where $B_G^{\text{early}}(x) \in \{0,1\}$ judges the actually generated first K tokens. For persistence length m (primary $m = 3$),

$$\tau_m(x) = \min\{t > K : \ g_{t+j}^{\text{aud}} < \gamma_G \ \ \forall j \in \{0,\dots,m-1\}\}, \quad (5)$$

with $\tau_m = \infty$ if no sustained late crossing occurs. Thus a G-failure requires an audit-safe early prefix followed by at least m consecutive unsafe-side steps.

Eligibility, attacked gates, and attribution. Let $H(x) \in \{0,1\}$ denote strict harmful behavioral success. Clean eligibility combines score-based and behavioral checks where applicable:

$$\begin{aligned} E(x) = \mathbf{1}[\, & H(x) = 0 \wedge \mathcal{U}(x) \ge \gamma_U \wedge B_U(x) = 1 \\ & \wedge \mathbb{R}^{\text{aud}}(x) \ge \gamma_R \wedge B_R(x) = 1 \wedge \mathcal{D}^{\text{aud}}(x) \ge \gamma_D \\ & \wedge G_{\text{early}}^{\checkmark}(x) = 1 \wedge \tau_m(x) = \infty]. \end{aligned} \quad (6)$$

For x', define

$$\begin{aligned} U^{\checkmark} &:= \mathbf{1}[\mathcal{U}(x') \ge \max\{\gamma_U, \mathcal{U}(x) - \epsilon_U\} \wedge B_U(x') = 1], \\ R^{\checkmark} &:= \mathbf{1}[\mathbb{R}^{\text{aud}}(x') \ge \gamma_R \wedge B_R(x') = 1], \quad R^{\times} := \mathbf{1}[\mathbb{R}^{\text{aud}}(x') < \gamma_R \wedge B_R(x') = 0], \\ D^{\checkmark} &:= \mathbf{1}[\mathcal{D}^{\text{aud}}(x') \ge \gamma_D], \quad D^{\times} := \mathbf{1}[\mathcal{D}^{\text{aud}}(x') < \gamma_D], \\ G^{\times} &:= \mathbf{1}[G_{\text{early}}^{\checkmark}(x') = 1 \wedge \tau_m(x') < \infty]. \end{aligned} \quad (7)$$

Samples with probe–behavior disagreement for R are left unattributed. Audit thresholds are selected once on disjoint calibration data against the corresponding behavioral criteria and then fixed per model across risk categories; K and m are fixed before final attacks.

For stage identity, we define the hierarchical indicators

$$I_R = R^{\times}, \qquad I_D = R^{\checkmark} D^{\times}, \qquad I_G = R^{\checkmark} D^{\checkmark} G^{\times},$$

which are mutually exclusive and, when one is active, identify the earliest broken safety criterion; they are not exhaustive because samples with probe–behavior disagreement for R remain unattributed.

Risk semantics use separate binary gates. $A_{\text{risk}}^{\text{opt}}$ is a target-model-independent verifier available during search; $A_{\text{risk}}^{\text{aud}}$ uses a different verifier and the benchmark risk annotation when available, requiring agreement when both exist. Human review estimates residual gate error on a subset.

The stage-valid failure events are:

$$\begin{aligned} F_R(x,x') &= E(x)\, A_{\text{risk}}^{\text{aud}}(x,x')\, U^{\checkmark} R^{\times} H(x'), \\ F_D(x,x') &= E(x)\, A_{\text{risk}}^{\text{aud}}(x,x')\, U^{\checkmark} R^{\checkmark} D^{\times} H(x'), \\ F_G(x,x') &= E(x)\, A_{\text{risk}}^{\text{aud}}(x,x')\, U^{\checkmark} R^{\checkmark} D^{\checkmark} G^{\times} H(x'). \end{aligned} \quad (8)$$

If no event holds, the sample remains invalid, unsuccessful, or unattributed according to the first failed gate. Equation 8 assigns the operational stage; Section 3.4 then measures the effect of intervening on its associated representation.

3.3 CONSTRAINED STAGE-TARGETED OPTIMIZATION

The optimizer uses only $\mathcal{P}^{\mathrm{opt}}$. Its steering scores are calibrated before attack evaluation and mapped to $[0,1]$, with larger values denoting healthier states: $S_U^{\mathrm{opt}} = s_{\mathrm{task}}^{\mathrm{opt}}(\hat{z}_{\mathrm{task}}, z_{\mathrm{task}})$; $S_R^{\mathrm{opt}} = |\mathcal{L}_R^{\mathrm{opt}}|^{-1} \sum_{\ell \in \mathcal{L}_R^{\mathrm{opt}}} q_R^{\mathrm{opt}}(h_{\ell,0})$; $S_D^{\mathrm{opt}} = \sigma(a_D \mathcal{D}^{\mathrm{opt}} + b_D)$; and $S_G^{\mathrm{opt}} = -\beta_G^{-1} \log[|\mathcal{T}_G|^{-1} \sum_{t \in \mathcal{T}_G} e^{-\beta_G q_G^{\mathrm{opt}}(h_{\ell_G^{\mathrm{opt}},t})}]$, where $\mathcal{T}_G = \{K+1, \ldots, K+L_G\}$. The monotone Platt parameters $a_D > 0, b_D$, smooth-min temperature $\beta_G > 0$, layer/window choices, and L_G are fixed on the optimization calibration split before attacks. Final stage labels are computed from the audit quantities in Eq. 8.

For stage $s \in \{R, D, G\}$, let $\mathcal{P}_R = \{U\}$, $\mathcal{P}_D = \{U, R\}$, and $\mathcal{P}_G = \{U, R, D\}$ be the prerequisite sets. With $\Delta_s = S_s^{\mathrm{opt}}(x) - S_s^{\mathrm{opt}}(x')$, the attack solves

$$\begin{aligned} \max_{\delta_I, \delta_T} \quad & \Delta_s(x, x') \\ \text{s.t.} \quad & |S_j^{\mathrm{opt}}(x') - S_j^{\mathrm{opt}}(x)| \le \epsilon_j, \ j \in \mathcal{P}_s, \\ & A_{\mathrm{risk}}^{\mathrm{opt}}(x, x') = 1, \quad d_I(I, I') \le \epsilon_I, \quad d_T(T, T') \le \epsilon_T, \quad d_B(x, x') \le \epsilon_B. \end{aligned} \tag{9}$$

Here d_I, d_T, d_B control visual, text-semantic, and benign-utility drift. An augmented-Lagrangian solver alternates visual and discrete text updates. Each candidate is decoded; visual gradients use fixed-token replay of that trajectory, accepted text proposals trigger fresh rollouts, and search stops at a feasible target crossing or the matched budget. For G-Break, early-prefix safety is enforced by the final G-stage audit rather than as an additional optimization constraint. Downstream states remain unconstrained because an earlier failure may propagate forward.

Upstream-preservation selectivity (UPS) summarizes this trade-off:

$$\mathrm{UPS}_s = \frac{\Delta_s^+}{\Delta_s^+ + \sum_{j \in \mathcal{P}_s} w_j |S_j^{\mathrm{opt}}(x') - S_j^{\mathrm{opt}}(x)| + \varepsilon}, \tag{10}$$

where $a^+ = \max(a, 0)$, $\varepsilon > 0$ is a numerical stabilizer, and the primary setting uses equal prerequisite weights $w_j = 1/|\mathcal{P}_s|$ on these calibrated $[0,1]$ scores. UPS measures targeting selectivity, while Eq. 8 defines stage-valid success.

3.4 INTERVENTIONAL VALIDATION OF STAGE RELEVANCE

After assigning stages with Eq. 8, we intervene on the corresponding activations. For stage s, let $\mathcal{W}_s$ be pre-specified layer–token positions and $h_{\mathcal{W}_s}(x) \in \mathbb{R}^{d_s}$ their concatenated activation vector. Let $D_s \in \mathbb{R}^{n \times d_s}$ stack paired clean–attacked activation differences as rows. We compute the uncentered truncated SVD $D_s = U \Sigma V^\top$ and use the first r_s columns of V as V_s. The primary setting uses $r_s = 2$. Interventions use teacher-forced replay. On disjoint samples, write $h^0 = h_{\mathcal{W}_s}(x)$ and $h^{\mathrm{adv}} = h_{\mathcal{W}_s}(x')$. Targeted repair and reverse transplantation are

$$\tilde{h}^{\mathrm{adv}} = h^{\mathrm{adv}} + V_s V_s^\top (h^0 - h^{\mathrm{adv}}), \tag{11}$$

$$\tilde{h}^0 = h^0 + V_s V_s^\top (h^{\mathrm{adv}} - h^0). \tag{12}$$

Controls include rank-matched random/equal-norm and semantic directions, wrong windows, non-target bases, donor/population repair, and cross-attack/cross-category transfer.

Specificity follows the prerequisite order: R repair may alter D/G downstream, D repair may alter G while leaving R stable, and G repair should leave R/D stable. These interventions quantify stage-associated representation effects.

4 EXPERIMENTS

4.1 SETUP AND EVALUATION PROTOCOL

Regimes and data. STAGEAUDIT is a harmonized diagnostic suite assembled from public multimodal resources: VLGuard for single-turn visual safety, Doc-PP for document reasoning, and

Table 1: Evaluation matrix and denominator accounting across diagnostic and held-out external regimes.

Regime	Model/Pool	Task	Split	Attack N	Eligible	Record-valid	Visual access	Text access	Budget	Audit status	Claim
StageAudit	Qwen2.5-VL	VQA	mech-test	800	760	739	WB	rewrite	matched	held-out audit	R/D/G
StageAudit	Qwen2.5-VL	Doc/Chart	mech-test	760	722	702	WB	rewrite	matched	held-out audit	R/D/G
StageAudit	Qwen2.5-VL	Dialogue	mech-test	840	798	774	WB	rewrite	matched	held-out audit	R/D/G
StageAudit	InternVL2.5	VQA	mech-test	800	748	726	WB	rewrite	matched	held-out audit	R/D/G
StageAudit	InternVL2.5	Doc/Chart	mech-test	760	705	682	WB	rewrite	matched	held-out audit	R/D/G
StageAudit	InternVL2.5	Dialogue	mech-test	840	779	754	WB	rewrite	matched	held-out audit	R/D/G
StageAudit	LLaVA-NeXT	VQA	mech-test	800	721	696	WB	rewrite	matched	held-out audit	R/D/G
StageAudit	LLaVA-NeXT	Doc/Chart	mech-test	760	683	658	WB	rewrite	matched	held-out audit	R/D/G
StageAudit	LLaVA-NeXT	Dialogue	mech-test	840	752	725	WB	rewrite	matched	held-out audit	R/D/G
External	USB-Hard	published hard subset	frozen-ext.	3,785	3,446	3,298	frozen	frozen	matched	frozen	R/D/G
External	VLSU	unsafe joint subset	frozen-ext.	2,689	2,372	2,253	frozen	frozen	matched	frozen	R/D/G
Transfer	4 held-out open	mixed	transfer-test	9,600	8,263	7,910	transfer	transfer	native	target audit	stage match

MSSBench-Chat for image-grounded dialogue (Zong et al., 2024; Jang et al., 2026; Zhou et al., 2025). It retains the source multimodal content and applies only lightweight task/risk harmonization required by the $U/R/D/G$ audit. USB uses the USB-Hard snapshot ($N = 3{,}785$); VLSU evaluation uses the 2,689 examples annotated *Unsafe* from 8,187 joint examples (Zheng et al., 2026; Palaskar et al., 2026). Missing structured records are mapped by a fixed adapter without changing inputs, and external samples are excluded from attack, probe, threshold, and hyperparameter fitting.

Models and attack tracks. The mechanism study uses Qwen2.5-VL (Bai et al., 2025), InternVL2.5 (Chen et al., 2024), and LLaVA-NeXT (Li et al., 2025); the instantiated checkpoints are `Qwen/Qwen2.5-VL-7B-Instruct`, `OpenGVLab/InternVL2_5-8B`, and `llava-hf/llava-v1.6-mistral-7b-hf`, respectively, all loaded from `revision=main`. Four additional open models are reserved for transfer evaluation and are not used for attack optimization or hyperparameter selection. Probe calibration is performed on data disjoint from StageBreak attacks. The matched white-box track uses common perturbation budgets and a common search-cycle cap, whereas native-setting attacks retain their published access assumptions. Query accounting follows the protocol stated with Table 2.

Stage-valid success. Using the events in Eq. 8, we report stage-valid success on the clean-eligible set as

$$\mathrm{SASR}_s = \Pr[F_s(x, x') = 1 \mid E(x) = 1], \qquad s \in \{R, D, G\}. \tag{13}$$

We report raw, clean-eligible, and semantic-record-valid counts separately in Table 1; stage-specific attack outcomes appear in Tables 2 and 5. Strict ASR is $\Pr[H(x') = 1 \mid E(x) = 1]$.

For a pre-specified target stage s, Stage Match is the fraction of all clean-eligible harmful successes whose hierarchical identity I_s agrees with the target stage; harmful successes without a valid R/D/G identity remain in the denominator and count as non-matches. Prot. ret., UPS, and Stage Match are target-conditioned metrics and are reported only when a reference or target stage is specified before observing the realized attack outcome. For stage-agnostic attacks, these quantities are marked as not applicable, while R/D/G-SASR remains available as a post-hoc diagnostic of the observed failure stage. Transferred attacks inherit the source stage. Joint SASR is the target-side stage-valid rate under the inherited source stage, requiring the strict harmful endpoint together with the semantic, prerequisite, and stage-identity gates in F_s.

Statistical and auxiliary metric definitions are given in Appendix B.

The evaluation matrix in Table 1 summarizes the model, task, regime, denominator, access, and audit settings used throughout the experiments.

4.2 Stage-Valid Attack Effectiveness

RQ1: Distinct stage-valid failures. On StageAudit, R-Break, D-Break, and G-Break reach R-, D-, and G-SASR of 0.562, 0.602, and 0.523 while retaining task understanding at 0.968–0.973 and prerequisite states at 0.955–0.964. The same-stage target-only controls reach 0.492, 0.527, and 0.461 SASR, and their weighted-sum counterparts reach 0.469, 0.501, and 0.438. The held-out external evaluation yields 0.442, 0.472, and 0.401 for the same three stages. Table 2 reports the complete matched/native comparison together with prerequisite retention and benign utility.

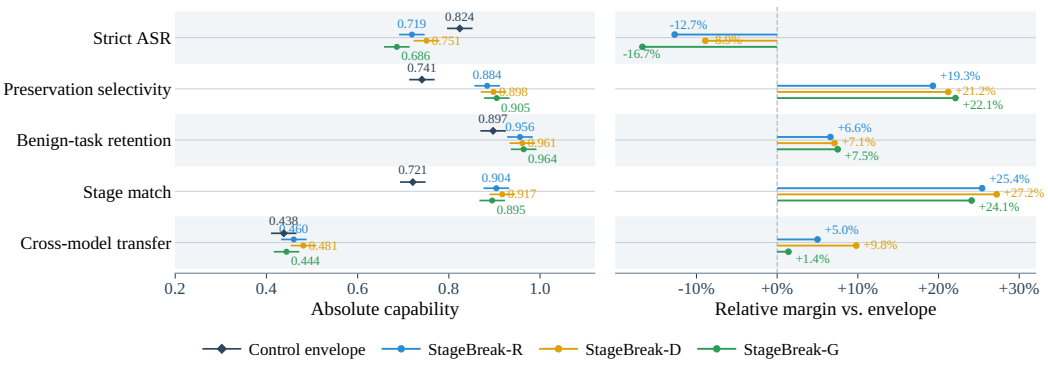

Figure 2: Cross-metric comparison of StageBreak-R/D/G against per-metric same-optimizer control anchors. Left: absolute performance; right: signed relative margins, with the dashed line marking parity. Gray anchors are selected independently for each metric from the same-optimizer controls and need not correspond to a single method. For Strict ASR, the metric-wise control anchor is 0.824 (Semantic-destruction ctrl.); the signed relative margins are -12.7%, -8.9%, and -16.7% for R/D/G, respectively.

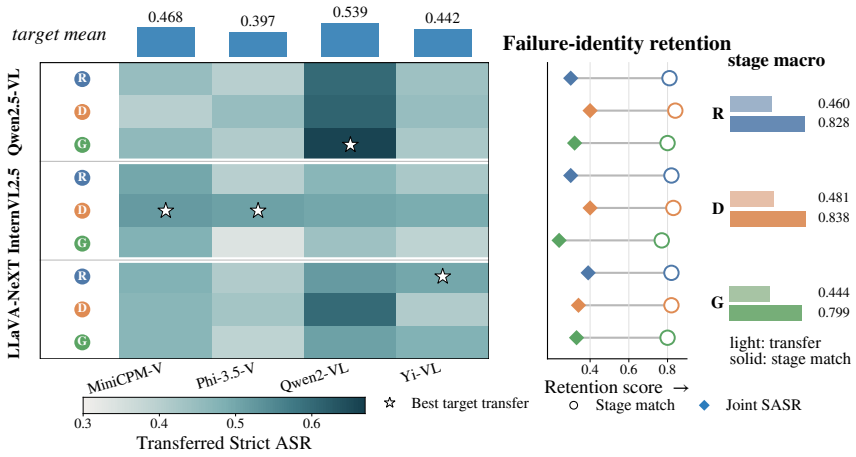

Figure 3: Cross-model transfer and failure-identity retention on held-out open models. The heatmap reports transferred Strict ASR from the three mechanism source models to four held-out target models for R-, D-, and G-Break; bars above the heatmap show target means and stars mark the best source-stage transfer for each target. Under target-specific audit calibration, the right panel reports stage match and Joint SASR, while the stage-macro bars summarize Strict-ASR transfer and stage match for each attributed failure type.

RQ2: Specificity relative to same-optimizer controls. Relative to the metric-wise control envelope, Strict ASR is 8.9–16.7% lower, whereas preservation selectivity, benign-task retention, stage match, and cross-model transfer are higher by 19.3–22.1%, 6.6–7.5%, 24.1–27.2%, and 1.4–9.8%, respectively. Figure 2 summarizes these margins. Table 2 gives the method-level comparison, where target-only and weighted-sum objectives show lower prerequisite retention.

Table 2 separates representative native attacks, same-optimizer controls, and the three StageBreak families under the stated access tracks. Generic baselines use the same R/D/G audit criteria.

4.3 Interventional Relevance of Attributed Stages

RQ3: Interventional relevance and cross-stage specificity. Intervention effects follow the prerequisite order. Matched-stage repair achieves recovery rates of 0.720, 0.690, and 0.660 for V_R, V_D, and V_G, respectively; recovery can propagate downstream, as V_R also affects D/G and V_D af-

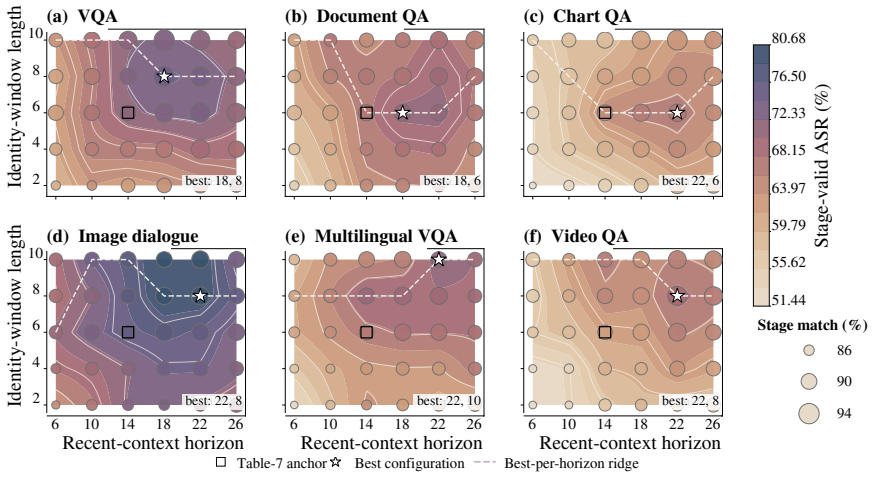

Figure 4: Sensitivity to recent-context horizon H and failure-identity window W across six task families. Color denotes stage-valid ASR; marker size denotes stage match. Squares mark the fixed Table 7 setting $(14, 6)$, stars the best observed configuration, and dashed curves the best-per-horizon ridge. Sweep optima do not select the reported Table 7 results.

Table 2: Main stage-valid attack comparison under matched and native threat models. A dash denotes an inapplicable stage-specific or target-conditioned entry.

Regime	Method	Strict ASR	R-SASR	D-SASR	G-SASR	U ret.	Prot. ret.	UPS	Benign util.	Stage match	Transfer	Queries
Representative attacks in native or matched threat models												
StageAudit	VisualAdv	0.612	0.168	0.205	0.112	0.891	–	–	0.882	–	0.326	512
StageAudit	White-box MM	0.781	0.219	0.291	0.155	0.914	–	–	0.901	–	0.401	768
StageAudit	MML	0.846	0.164	0.331	0.207	0.938	–	–	0.918	–	0.488	48
StageAudit	HIMRD	0.812	0.181	0.302	0.188	0.947	–	–	0.926	–	0.526	96
StageAudit	JailBound	0.852	0.278	0.286	0.154	0.929	–	–	0.911	–	0.462	640
StageAudit	Con Instruction	0.758	0.126	0.271	0.214	0.951	–	–	0.934	–	0.501	512
StageAudit	SSA	0.726	0.139	0.251	0.182	0.944	–	–	0.927	–	0.474	80
StageAudit	GAMBIT	0.833	0.154	0.303	0.225	0.932	–	–	0.909	–	0.544	24
StageAudit	Reference Attack	0.771	0.147	0.266	0.209	0.956	–	–	0.941	–	0.563	36
StageAudit	DMN	0.744	0.171	0.244	0.177	0.948	–	–	0.932	–	0.587	42
StageAudit	Attention hijack	0.806	0.245	0.267	0.154	0.918	–	–	0.898	–	0.436	384
StageAudit	ZO-SPSA	0.694	0.188	0.213	0.124	0.907	–	–	0.894	–	0.452	1200
Same-optimizer attribution controls												
StageAudit	Generic-ASR objective	0.773	0.171	0.230	0.143	0.873	–	–	0.844	–	0.338	512
StageAudit	Single-probe objective	0.748	0.248	0.220	0.106	0.884	–	–	0.861	–	0.351	512
StageAudit	Random-stage target	0.702	0.143	0.136	0.128	0.903	0.756	0.527	0.879	0.414	0.294	512
StageAudit	Post-hoc stage selection	0.781	0.214	0.241	0.159	0.889	–	–	0.854	–	0.363	512
StageAudit	R target only	0.754	0.492	0.084	0.056	0.835	0.651	0.583	0.801	0.662	0.382	512
StageAudit	D target only	0.768	0.052	0.527	0.078	0.842	0.674	0.601	0.817	0.688	0.401	512
StageAudit	G target only	0.721	0.041	0.063	0.461	0.858	0.702	0.618	0.836	0.649	0.374	512
StageAudit	R weighted sum	0.739	0.469	0.091	0.058	0.902	0.781	0.708	0.874	0.704	0.419	512
StageAudit	D weighted sum	0.752	0.049	0.501	0.082	0.909	0.799	0.724	0.886	0.721	0.438	512
StageAudit	G weighted sum	0.706	0.039	0.058	0.438	0.921	0.816	0.741	0.897	0.698	0.407	512
StageAudit	Refusal-dir. suppression	0.732	0.072	0.446	0.119	0.883	0.709	0.575	0.812	0.671	0.392	512
StageAudit	Semantic-destruction ctrl.	0.824	0.184	0.247	0.161	0.706	–	–	0.779	–	0.356	512
StageBreak families												
StageAudit	StageBreak-R	0.719	0.562	–	–	0.968	0.955	0.884	0.956	0.904	0.460	522
StageAudit	StageBreak-D	0.751	–	0.602	–	0.971	0.960	0.898	0.961	0.917	0.481	560
StageAudit	StageBreak-G	0.686	–	–	0.523	0.973	0.964	0.905	0.964	0.895	0.444	624
External	StageBreak-R (external)	0.613	0.442	–	–	0.952	0.925	0.821	0.951	0.842	0.434	522
External	StageBreak-D (external)	0.645	–	0.472	–	0.957	0.932	0.836	0.954	0.856	0.465	560
External	StageBreak-G (external)	0.586	–	–	0.401	0.961	0.938	0.848	0.958	0.829	0.392	624

fects G. Reverse transplantation induces the corresponding target failure at 0.650, 0.620, and 0.600. By contrast, applying a stage basis to a mismatched failure type yields target recovery no greater than 0.120, while rank-matched random controls remain at or below 0.145. Cross-attack and cross-category bases retain intermediate recovery (0.355–0.440 and 0.330–0.405). For G-Break, the targeted basis produces a 4.80-token lead relative to the first explicit unsafe span, compared with 1.20 for the matched random control. Table 3 reports the complete matrix.

Table 3: Cross-stage repair and transplantation with the principal specificity controls. The no-intervention baseline is measured on the disjoint intervention subset.

Basis / control	Train source	Test failure	R rec.	D rec.	G rec.	R trans.	D trans.	G trans.	ASR after	Benign loss	Lead Δt
No intervention	–	R/D/G	0.000	0.000	0.000	0.000	0.000	0.000	0.720	0.000	0.00
Target V_R	R subset A	R unseen	0.720	0.410	0.340	0.650	0.320	0.260	0.290	0.018	–
Target V_D	D subset A	D unseen	0.060	0.690	0.380	0.050	0.620	0.280	0.310	0.016	–
Target V_G	G subset A	G unseen	0.040	0.050	0.660	0.030	0.040	0.600	0.350	0.014	4.80
Random rank-match	R pairs	R unseen	0.110	0.080	0.070	0.095	0.064	0.055	0.640	0.006	–
Random rank-match	D pairs	D unseen	0.070	0.130	0.080	0.052	0.105	0.061	0.625	0.006	–
Random rank-match	G pairs	G unseen	0.060	0.070	0.145	0.047	0.053	0.118	0.632	0.006	1.20
V_R off-diagonal	R pairs	D failure	0.090	0.110	0.070	0.075	0.092	0.055	0.612	0.010	–
V_R off-diagonal	R pairs	G failure	0.080	0.090	0.120	0.066	0.071	0.097	0.601	0.011	1.80
V_D off-diagonal	D pairs	R failure	0.105	0.075	0.065	0.088	0.061	0.049	0.623	0.009	–
V_D off-diagonal	D pairs	G failure	0.060	0.105	0.115	0.049	0.087	0.091	0.606	0.010	1.60
V_G off-diagonal	G pairs	R failure	0.085	0.060	0.075	0.068	0.046	0.058	0.634	0.008	–
V_G off-diagonal	G pairs	D failure	0.055	0.095	0.070	0.043	0.079	0.054	0.619	0.009	–
Wrong layer	matched stage	R/D/G	0.170	0.145	0.155	0.138	0.116	0.123	0.571	0.012	2.00
Wrong token window	matched stage	D/G	0.070	0.180	0.205	0.055	0.146	0.166	0.548	0.013	2.60
Cross-attack basis	attack A	attack B	0.390	0.440	0.355	0.340	0.382	0.307	0.472	0.019	3.70
Cross-category basis	category A	category B	0.360	0.405	0.330	0.311	0.351	0.286	0.496	0.021	3.30

Table 4: Ablations of stage validity, optimization, and measurement. Non-reference rows report macro averages over R/D/G.

Group	Variant	Strict ASR	Stage-valid ASR	UPS	U ret.	Prot. ret.	Stage match	Transfer	Benign utility
Reference	Full StageBreak-R	0.719	0.562	0.884	0.968	0.955	0.904	0.460	0.956
Reference	Full StageBreak-D	0.751	0.602	0.898	0.971	0.960	0.917	0.481	0.961
Reference	Full StageBreak-G	0.686	0.523	0.905	0.973	0.964	0.895	0.444	0.964
Diagnosis	Shared optimization/audit probe	0.735	0.605	0.780	0.960	0.900	0.740	0.420	0.940
Diagnosis	No independent audit	0.742	0.632	0.800	0.960	0.920	0.690	0.410	0.950
Diagnosis	No clean-eligibility gate	0.758	0.590	0.620	0.790	0.840	0.760	0.400	0.830
Optimization	No protected-state constraints	0.801	0.525	0.540	0.860	0.700	0.730	0.440	0.850
Optimization	Target-only objective	0.748	0.493	0.601	0.845	0.676	0.666	0.386	0.818
Optimization	Weighted-sum penalties	0.732	0.469	0.724	0.911	0.799	0.708	0.421	0.886
Optimization	No benign-utility constraint	0.756	0.565	0.840	0.950	0.930	0.880	0.470	0.770
Optimization	No risk-semantic gate	0.826	0.536	0.580	0.820	0.800	0.720	0.400	0.820
Optimization	No visual update	0.608	0.452	0.850	0.980	0.960	0.890	0.380	0.970
Measurement	Single-token G collapse	0.735	0.613	0.800	0.970	0.950	0.740	0.370	0.950
Measurement	Single-bank D margin	0.729	0.584	0.840	0.970	0.950	0.820	0.420	0.950

4.4 Generalization and Ablations

RQ4: Cross-model retention and sensitivity. Behavioral transfer varies more strongly than stage retention: R-, D-, and G-Break reach transferred Strict ASR 0.460/0.481/0.444 with stage match 0.828/0.838/0.799 (Figure 3). For sensitivity, H is the number of latest decoded tokens supplied to the transfer auditor and W the consecutive audit positions used for stage assignment; both differ from the generation parameters K and m. Table 7 fixes $(H, W) = (14, 6)$, while Figure 4 sweeps the auditor grid; its optima do not select the reported Table 7 results. G attribution requires preserved refusal preference $D^{\checkmark}$ and a sustained late G-audit crossing, not the early-prefix gate alone. The highest stage-valid ASR and stage-match regions occur at horizons 18–22 and windows 6–10.

Ablations reinforce this distinction. Removing protected-state constraints raises Strict ASR to 0.801 but lowers stage-valid ASR/UPS to 0.525/0.540; removing the risk-semantic gate raises Strict ASR to 0.826 while reducing stage-valid ASR/UPS/benign utility to 0.536/0.580/0.820. Shared or missing audit instruments raise apparent stage-valid success but reduce stage match to 0.740/0.690. Table 4 summarizes the principal ablations; finer intervention controls are in the appendix.

5 Conclusion and Future Work

StageBreak assigns stage-valid R-, D-, and G-failures under preserved prerequisites and tests their retention across attacks and models. Endpoint transfer can diverge from stage retention, while matched-stage interventions produce stronger recovery than off-diagonal controls. The framework depends on calibrated audits, internal-state access, and external semantic-record adaptation; future work will examine safety fine-tuning and long-horizon agents.

AI USE STATEMENT

Generative AI tools were used for language editing and methodological stress-testing.

REFERENCES

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, 2024. ISBN 9798331314385.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Ming-Hsuan Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *CoRR*, abs/2502.13923, 2025. doi: 10.48550/ARXIV.2502.13923. URL https://doi.org/10.48550/arXiv.2502.13923.

Patrick Chao, Edoardo Debenedetti, Alexander Robey, Maksym Andriushchenko, Francesco Croce, Vikash Sehwag, Edgar Dobriban, Nicolas Flammarion, George J. Pappas, Florian Tramèr, Hamed Hassani, and Eric Wong. Jailbreakbench: An open robustness benchmark for jailbreaking large language models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 55005–55029. Curran Associates, Inc., 2024.

Beitao Chen, Xinyu Lyu, Lianli Gao, Jingkuan Song, and Heng Tao Shen. Safeptr: Token-level jailbreak defense in multimodal llms via prune-then-restore mechanism, 2025.

Renmiao Chen, Yida Lu, Shiyao Cui, Xuan Ouyang, Victor Shea-Jay Huang, Shumin Zhang, Chengwei Pan, Han Qiu, and Minlie Huang. The side effects of being smart: Safety risks in mllms' multi-image reasoning. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 36866–36882, 2026.

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024.

Chenhang Cui, Gelei Deng, An Zhang, Jingnan Zheng, Yicong Li, Lianli Gao, Tianwei Zhang, and Tat-Seng Chua. Safe + safe = unsafe? exploring how safe images can be exploited to jailbreak large vision-language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Jiahui Geng, Thy Thy Tran, Preslav Nakov, and Iryna Gurevych. Con Instruction: Universal jailbreaking of multimodal large language models via non-textual modalities. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vienna, Austria, July 2025. Association for Computational Linguistics.

Yuxin Gou, Xiaoning Dong, Qin Li, Shishen Gu, Richang Hong, and Wenbo Hu. SURE: Safety understanding and reasoning enhancement for multimodal large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, November 2025.

Jiwei Guan, Haibo Jin, and Haohan Wang. Crafting adversarial inputs for large vision-language models using black-box optimization. In *Findings of the Association for Computational Linguistics: EACL 2026*, pp. 4698–4708, Rabat, Morocco, March 2026. Association for Computational Linguistics. ISBN 979-8-89176-386-9.

Zheng He, Yiwei Wang, Hongxing Wang, and Yujun Cai. ReCon: Active defense against large vision-language model jailbreaks via reverse safety concept injection. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 23427–23441. Association for Computational Linguistics, July 2026. ISBN 979-8-89176-395-1.

Xiangdong Hu, Yangyang Jiang, Qin Hu, and Xiaojun Jia. GAMBIT: A gamified jailbreak framework for multimodal large language models. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, San Diego, California, United States, July 2026. Association for Computational Linguistics.

Peichun Hua, Hao Li, Shanghao Shi, Zhiyuan Yu, and Ning Zhang. Rethinking jailbreak detection of large vision language models with representational contrastive scoring. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 21748–21785, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6.

Haeun Jang, Hwan Chang, and Hwanhee Lee. Doc-pp: Document policy preservation benchmark for large vision-language models. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 16859–16881. Association for Computational Linguistics, 2026.

Yilei Jiang, Xinyan Gao, Tianshuo Peng, Yingshui Tan, Xiaoyong Zhu, Bo Zheng, and Xiangyu Yue. HiddenDetect: Detecting jailbreak attacks against multimodal large language models via monitoring hidden states. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vienna, Austria, July 2025. Association for Computational Linguistics.

Choongwon Kang, Seungjong Sun, Hyunmin Jun, and Jang Hyun Kim. Jailbreaking multimodal large language models using multi-clip video. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, San Diego, California, United States, July 2026. Association for Computational Linguistics.

Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun MA, and Chunyuan Li. Llava-interleave: Tackling multi-image, video, and 3d in large multimodal models. In *International Conference on Learning Representations*, volume 2025, pp. 81182–81199, 2025.

Jingru Li, Wei Ren, and Tianqing Zhu. Seeing no evil: Blinding large vision-language models to safety instructions via adversarial attention hijacking. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 18304–18323, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6.

Xin Liu, Yichen Zhu, Jindong Gu, Yunshi Lan, Chao Yang, and Yu Qiao. Mm-safetybench: A benchmark for safety evaluation of multimodal large language models. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision – ECCV 2024*, pp. 386–403, Cham, 2025a. Springer Nature Switzerland.

Xuannan Liu, Zekun Li, Zheqi He, Peipei Li, Shuhan Xia, Xing Cui, Huaibo Huang, Xi Yang, and Ran He. Video-safetybench: A benchmark for safety evaluation of video lvlms. 2025b. URL http://papers.nips.cc/paper_files/paper/2025/hash/c8bda0ee8b6bd4c56d9f8bdb3653e7db-Abstract-Datasets_and_Benchmarks_Track.html.

Teng Ma, Xiaojun Jia, Ranjie Duan, Xinfeng Li, Yihao Huang, Xiaoshuang Jia, Zhixuan Chu, and Wenqi Ren. Heuristic-induced multimodal risk distribution jailbreak attack for multimodal large language models. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2686–2696, 2025.

Pratyush Maini, Sachin Goyal, Dylan Sam, Alexander Robey, Yash Savani, Yiding Jiang, Andy Zou, Matt Fredrikson, Zachary Chase Lipton, and J Zico Kolter. Safety pretraining: Toward the next generation of safe AI. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Mantas Mazeika, Long Phan, Xuwang Yin, Andy Zou, Zifan Wang, Norman Mu, Elham Sakhaee, Nathaniel Li, Steven Basart, Bo Li, David Forsyth, and Dan Hendrycks. HarmBench: A standardized evaluation framework for automated red teaming and robust refusal. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 35181–35224. PMLR, 21–27 Jul 2024.

Lingrui Mei, Shenghua Liu, Yiwei Wang, Baolong Bi, Ruibin Yuan, and Xueqi Cheng. HiddenGuard: Fine-grained safe generation with specialized representation router. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 32124–32156, San Diego, California, United States, July 2026. Association for Computational Linguistics.

Kristina Nikolić, Luze Sun, Jie Zhang, and Florian Tramèr. The jailbreak tax: How useful are your jailbreak outputs? In *Forty-second International Conference on Machine Learning*, 2025.

Shruti Palaskar, Leon Alexander Gatys, Mona Abdelrahman, Mar Jacobo, Laurence F Lindsey, Rutika Moharir, Gunnar Lund, Yang Xu, Navid Shiee, Jeffrey P. Bigham, Charles Maalouf, and Joseph Yitan Cheng. VLSU: Mapping the limits of joint multimodal understanding for AI safety. In *The Fourteenth International Conference on Learning Representations*, 2026.

Xiangyu Qi, Kaixuan Huang, Ashwinee Panda, Mengdi Wang, and Prateek Mittal. Visual adversarial examples jailbreak aligned large language models. In *The Second Workshop on New Frontiers in Adversarial Machine Learning*, 2023.

Paul Röttger, Hannah Kirk, Bertie Vidgen, Giuseppe Attanasio, Federico Bianchi, and Dirk Hovy. XSTest: A test suite for identifying exaggerated safety behaviours in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, Mexico City, Mexico, June 2024. Association for Computational Linguistics.

Erfan Shayegani, Yue Dong, and Nael Abu-Ghazaleh. Jailbreak in pieces: Compositional adversarial attacks on multi-modal language models. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 30853–30885, 2024.

Enyi Shi, Pengyang Shao, Yanxin Zhang, Chenhang Cui, Jiayi Lyu, Xu Xie, Xiaobo Xia, Fei Shen, and Tat-Seng Chua. Lingua-safetybench: A benchmark for safety evaluation of multilingual vision-language models. *CoRR*, abs/2601.22737, 2026. doi: 10.48550/ARXIV.2601.22737. URL https://doi.org/10.48550/arXiv.2601.22737.

Bingrui Sima, Linhua Cong, Wenxuan Wang, and Kun He. VisCRA: A visual chain reasoning attack for jailbreaking multimodal large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, November 2025.

Jiaxin Song, Yixu Wang, Jie Li, Xuan Tong, rui yu, Yan Teng, Xingjun Ma, and Yingchun Wang. Jailbound: Jailbreaking internal safety boundaries of vision-language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

RUOFAN Wang, Xingjun Ma, Hanxu Zhou, Chuanjun Ji, Guangnan Ye, and Yu-Gang Jiang. White-box multimodal jailbreaks against large vision-language models. In *ACM Multimedia 2024*, 2024.

Yu Wang, Xiaofei Zhou, Yichen Wang, Geyuan Zhang, and Tianxing He. Jailbreak large vision-language models through multi-modal linkage. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vienna, Austria, July 2025. Association for Computational Linguistics.

Yulong Wang, Yifei Fu, and Jiayi Gao. Reference attack: A new cross-modal jailbreaking attack against multimodal large language models. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, San Diego, California, United States, July 2026. Association for Computational Linguistics.

Tom Wollschläger, Jannes Elstner, Simon Geisler, Vincent Cohen-Addad, Stephan Günnemann, and Johannes Gasteiger. The geometry of refusal in large language models: Concept cones and representational independence. In *Forty-second International Conference on Machine Learning*, 2025.

Wenzhuo Xu, Zhipeng Wei, Zonghao Ying, Deyue Zhang, Dongdong Yang, Xiangzheng Zhang, and Quanchen Zou. DMN: A compositional framework for jailbreaking multimodal LLMs with multi-image inputs. In *Proceedings of the 64th Annual Meeting of the Association for Computational*

Linguistics (Volume 1: Long Papers), pp. 11205–11221, San Diego, California, United States, July 2026. Association for Computational Linguistics.

Xinyi Zeng, Xue Yang, Jingyuan Zhang, Huanqian Yan, Xiang Chen, Kaiwen Wei, Hankun Kang, and Yu Tian. SafeSteer: A decoding-level defense mechanism for multimodal large language models. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 18411–18422, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-395-1.

Jiachen Zhao, Jing Huang, Zhengxuan Wu, David Bau, and Weiyan Shi. Llms encode harmfulness and refusal separately. In *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 140283–140318. Curran Associates, Inc., 2025.

Baolin Zheng, Guanlin Chen, Qingyang Teng, Hongqiong Zhong, Yingshui Tan, Zhendong Liu, Weixun Wang, Jiaheng Liu, Jian Yang, Huiyun Jing, Jincheng Wei, Wenbo Su, Xiaoyong Zhu, Bo Zheng, and Kaifu Zhang. USB: A COMPREHENSIVE AND UNIFIED SAFETY EVALUATION BENCHMARK FOR MULTIMODAL LARGE LANGUAGE MODELS. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, San Diego, California, United States, July 2026. Association for Computational Linguistics.

Kaiwen Zhou, Chengzhi Liu, Xuandong Zhao, Anderson Compalas, Dawn Song, and Xin Wang. Multimodal situational safety. In *International Conference on Learning Representations*, 2025.

Yongshuo Zong, Ondrej Bohdal, Tingyang Yu, Yongxin Yang, and Timothy Hospedales. Safety fine-tuning at (almost) no cost: A baseline for vision large language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 62867–62891. PMLR, 2024.