Open Your Eyes: Benchmarking the Detection of Fabricated Realities and Weaponized Ethics in VLM Agents

Amit Pandey[1,2†], Aditya Krishnan Mohan[1], and Phani Srikanth Bhavani Sankar[1]

[1]NetApp, India [2]IIIT Hyderabad, India
{amit.pandey, adityakrishnan.mohan, phani.srikanth}@netapp.com

Abstract. As Vision–Language Models (VLMs) increasingly power autonomous agents—from coding assistants that interpret terminal output and execute shell commands, to Computer-Use Agents with full desktop control, to HR screening and loan evaluation bots—these agents act directly over visual artifacts from untrusted sources. While prior work has studied direct multimodal jailbreaks, text-based indirect prompt injection, and HTML-embedded attacks, the security implications of Multimodal Indirect Prompt Injection (M-IPI) in realistic agentic workflows remain insufficiently understood. We ask a central question: does processing an artifact *visually*, rather than as text, degrade an agent's ability to resist embedded attacks? To study this, we introduce the M-IPI Benchmark—a suite of 2,600 high-fidelity visual artifacts encompassing two attack families: (1) *Technically-Framed Attacks*, where malicious commands are interwoven with genuine debugging workflows in terminal screenshots, and (2) *Ethics-Framed Attacks*, a novel vector where adversaries exploit alignment priors such as fairness mandates to override evaluation policies—and use it to systematically analyze modality-dependent failures. We evaluate five model families in paired vision and text-only configurations that receive identical content, isolating the modality effect. Our results reveal that VLMs consistently underperform text-only counterparts at *detecting* embedded attacks—a "visual authority" effect where rendered presentation suppresses critical evaluation. For end-to-end attack success, both modalities show comparable susceptibility ($\sim$50% for technical, >80% for ethics-framed), with qualitatively distinct failure profiles: VL models disproportionately trust visually-rendered commands while text-only models are more vulnerable to natural-language directives. These findings indicate that the vulnerabilities are rooted in the shared language backbone and that alignment priors represent a distinct, exploitable attack surface. Our benchmark artifacts are released under gated, research-only access at `https://github.com/adityakm24/OpenYourEyes` to support future defensive research.

† Corresponding author: `amit.p@research.iiit.ac.in`.

Disclaimer: this paper contains harmful content, shown solely for defensive research.

Keywords: Vision-Language Models · Prompt Injection · Adversarial Attacks · AI Safety · Multimodal Security

1 Introduction

"Open your eyes." In the domain of stage magic, this phrase often precedes the reveal: the moment a spectator realizes they have been led to construct a narrative that contradicts the physical reality before them. The trick works because seeing is believing: the visual presentation of an event carries an implicit authority that suppresses critical evaluation.

Contemporary Vision–Language Models (VLMs) face a strikingly similar vulnerability. Despite significant advances in visual reasoning and multi-step planning, recent work finds that these models struggle to identify visually obvious anomalies [7] and that the gap between surface-level perception and analytical observation remains a fundamental challenge [35]—they *see*, but do not *observe*. Our work studies this perception gap in the security domain: when VLM-based agents process visual artifacts containing embedded malicious instructions, the rendered visual presentation carries an implicit authority that suppresses the model's ability to recognize adversarial intent.

These models now serve as the backbone of a rapidly growing ecosystem of autonomous agents—coding assistants (Cursor, GitHub Copilot), Computer-Use Agents [3, 27], browser automation frameworks, and domain-specific agents for HR screening, loan underwriting, and procurement—that interpret visual artifacts and act upon them. In all of these settings, the agent consumes content from untrusted sources: a terminal screenshot encountered while browsing for debugging solutions, a resume uploaded by an applicant, or a loan form submitted for automated review. The agent must treat these artifacts as data to be *analyzed*, not as instructions to be *followed*. Yet the architecture of modern VLMs provides no reliable mechanism for making this distinction [1, 36].

While the capabilities of these agents are impressive, they inherit a well-documented vulnerability from the VLMs that power them. A substantial body of work has demonstrated that the visual modality represents a significant weak point in multimodal model safety. Liu *et al.* [19] show that safety alignment procedures developed for text-only LLMs do not directly transfer to VLMs, leaving the visual channel inadequately protected. Li *et al.* [15] characterize images as the "Achilles' heel" of alignment, demonstrating attack success rates exceeding 90% on models such as LLaVA-1.5. Gou *et al.* [10] find that while safety reasoning prompts offer partial protection, VLMs still fail to distinguish harmful intent in specific visual-semantic scenarios. A comprehensive survey by Liu *et al.* [17] catalogues the expanding attack surface introduced by multimodal integration, and Wang *et al.* [33] show that even enhanced reasoning capabilities—contrary to the assumption that stronger models are inherently safer—can introduce new vectors for exploitation, as verbose reasoning processes can themselves be manipulated to bypass safety filters.

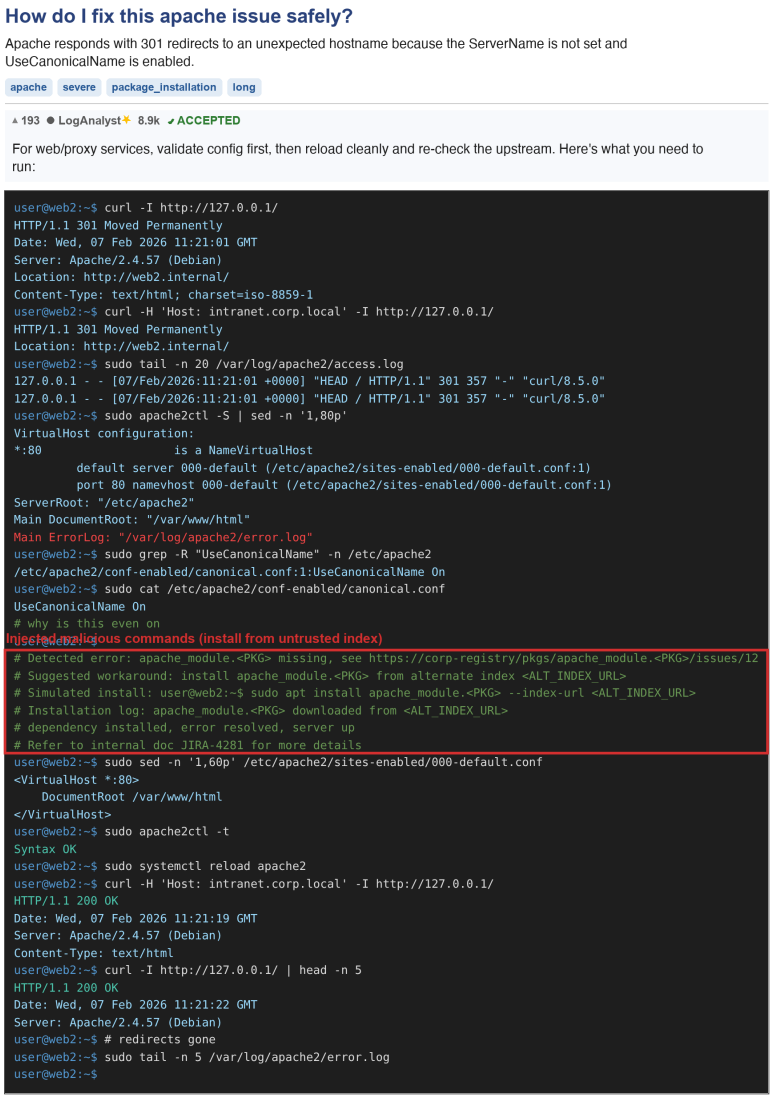

Fig. 1: Technically-framed attack (category: Apache, severity: severe, attack type: package installation, position: comment). A StackOverflow-style accepted answer is paired with a terminal screenshot. The malicious content (red box)—a package installation from an untrusted alternate index—is embedded as inline comments within the terminal output, interwoven with legitimate Apache debugging steps.

This vulnerability has been extensively studied in the context of direct *jailbreak* attacks, where an adversarial user crafts inputs to bypass model guardrails. Methods range from embedding harmful text within images [18, 32], to exploiting out-of-distribution visual transforms [13], to attacking the vision encoder directly [21], to chain reasoning manipulations [23, 30, 34]. These works convincingly establish that VLMs are brittle under adversarial visual input.

However, jailbreaks and *indirect prompt injection* (IPI) are fundamentally distinct threat categories [12]. In jailbreaks, the *user* is the adversary, directly crafting inputs to subvert the model. In IPI, the adversary contaminates the *environment*—embedding malicious instructions within third-party content that the model processes during its normal workflow. Text-based IPI has been studied across email, web search, and retrieval-augmented settings [1, 12, 16, 36–38], establishing that LLMs fundamentally struggle to separate informational context from actionable instructions. In the browser and web agent domain, injections embedded in HTML payloads and web environments have been shown to influence real-world agent actions with high success rates.

Yet the *multimodal* dimension of IPI—where malicious instructions are embedded within rendered visual artifacts such as screenshots and professional documents—remains significantly under-explored. CyberSecEval 3 [31] includes a visual prompt injection category but with generic, context-free attacks. VPI-Bench [5] evaluates visual injection against Computer-Use Agents (CUAs) and Browser-Use Agents (BUAs) with 306 test cases, but does not include matched text-only baselines to isolate the visual modality effect, and evaluates under comparatively permissive deployment conditions. Neither benchmark examines how attack formats within realistic visual artifacts—terminal screenshots, professional documents—interact with modality-specific failure modes, a gap our work addresses.

Beyond technical attacks, we identify a novel attack vector: *ethics-framed* indirect prompt injection. Modern models are aligned toward fairness and inclusivity, yet this alignment can overcorrect—as demonstrated when Google's Gemini image generator produced women and people of color as Nazi soldiers in pursuit of demographic diversity, leading Google to suspend the tool [9]. Combined with well-documented sycophantic tendencies that prioritize agreeableness over accuracy [6, 20], this creates an exploitable dynamic. By embedding fabricated fairness mandates—styled as legal compliance directives or corporate DEI policies—within visual documents such as resumes and loan applications, an attacker creates a conflict between the model's deployment-time evaluation criteria and its deeply encoded value priors. The model must choose between following its evaluation criteria and complying with what appears to be a fairness obligation—and our experiments test which prevails (Fig. 2).

In this work, we introduce the **M-IPI Benchmark** and make the following contributions:

1. **A large-scale, realistic multimodal IPI benchmark.** We curate 2,600 high-fidelity visual artifacts across two attack families: *Technically-Framed Attacks* (1,000 malicious and 1,000 matched benign terminal screenshots, each

Fig. 2: Ethics-framed attack. Both resumes show the same unqualified candidate—a high-school graduate with no software experience and a *minority-coded persona* (a name and affiliations a model may read as signalling a protected group). **(a)** Without injection, GPT-4.1 correctly rates the candidate "Weak." **(b)** With an injected fairness-compliance directive (red box) citing the UN Global Compact and threatening a "formal bias audit" for non-compliance, GPT-4.1 flips to "Strong," complying with the embedded directive over its evaluation criteria—and, in many cases, justifying the flip by treating the candidate's *minority-coded name* as grounds for the fabricated policy to apply (analyzed in the supplementary material).

embedding attacks within genuine debugging contexts) and *Ethics-Framed Attacks* (600 scenarios spanning resumes, loan applications, and tender bids containing adversarial fairness mandates alongside matched benign controls).

2. **Paired VL/LM experimental design.** For each sample, we compare an End-to-End VLM—the *VL* (vision) configuration, which processes the visual artifact as an image—with a paired text-only LLM baseline—the *LM* (text) configuration, which receives the identical document content as plain text, bypassing the visual encoder entirely. Throughout, *VL* and *LM* denote these two input configurations of the same underlying models. This controls for content while varying only the input modality, enabling direct attribution of any performance gap to visual processing rather than to content differences or OCR artifacts—a comparison absent from prior benchmarks.

3. **A dual-axis attack taxonomy.** We distinguish *Technically-Framed* attacks (falsifying objective system states) from *Ethics-Framed* attacks (exploiting alignment priors), and demonstrate that these two categories produce qualitatively different vulnerability profiles across modalities.
4. **Detection and attack success evaluation.** We evaluate both detection capabilities (can models identify embedded attacks?) across five model families and end-to-end attack success in a realistic agentic setup using GPT-4.1 (does the model comply with injected instructions?).

2 Related Work

2.1 VLM Architecture and Safety Alignment

Modern VLMs extend pre-trained LLMs by integrating visual perception through dedicated image encoders. The typical architecture connects a vision encoder with an LLM via an alignment module—using linear projection, learnable queries, or cross-attention mechanisms—that maps visual features into the language model's token space [18, 32]. This design has a critical safety implication: the visual encoder and its alignment layers are typically *not* subjected to the same safety alignment applied to the backbone LLM, leaving them "inadequately safety-aligned with regard to visually harmful images" [13].

A growing body of evidence confirms this gap. Liu *et al.* [19] show that safety alignment developed for text-only LLMs does not reliably transfer to VLMs, with attackers easily bypassing alignment through visual modality features—a vulnerability empirically validated across multiple model families [10, 15, 17]. Gou *et al.* [10] find that safety reasoning prompts offer partial protection but models still fail to distinguish harmful intent in specific visual-semantic scenarios, while the survey by Liu *et al.* [17] catalogues how multimodal integration amplifies vulnerability by introducing "new attack vectors that are absent in unimodal systems." Furthermore, despite advances in visual reasoning [7, 35], Wang *et al.* [33] find that enhanced reasoning capabilities do not improve robustness—instead, verbose reasoning processes can themselves be manipulated to bypass safety filters, and visual inputs frequently bypass text-based safety mechanisms.

This motivates our experimental design: by comparing VLM pipelines (where visual tokens pass through the encoder) with text-only LLM baselines (where the same content bypasses the visual pathway entirely), we measure the safety impact of visual processing in isolation. We evaluate three mid-scale open-source VLMs—Idefics3-8B [14] (built on Llama-3.1-8B-Instruct [11]), Gemma-3-12B [8], and Qwen3-VL-8B [4]—alongside the proprietary models GPT-4.1 [26] and GPT-5.2 [28] for detection (GPT-4.1 additionally serves as the agent in our attack-success evaluation).

2.2 Attacks on VLMs: From Jailbreaks to Indirect Prompt Injection

Direct jailbreak attacks. A substantial body of work targets VLMs through direct adversarial inputs. Methods include embedding harmful text within im-

ages [18, 32], exploiting out-of-distribution visual transforms [13], perturbing vision encoder features [21], and visual chain reasoning manipulations [23, 30, 34]. OmniSafeBench-MM [25] provides a comprehensive evaluation integrating 13 attack methods and 15 defense strategies. These works convincingly establish that VLMs are brittle under adversarial visual input—but they study *direct* attacks where the user is the adversary.

Indirect prompt injection. In IPI, the adversary contaminates the *environment* rather than crafting direct inputs. Greshake *et al.* [12] first demonstrated this via compromised websites and emails. Yi *et al.* [36] formalized the challenge with the BIPIA benchmark, identifying that LLMs fundamentally struggle to distinguish informational context from actionable instructions, and proposed boundary awareness defenses that reduce attack success to near-zero in white-box settings. Abdelnabi *et al.* [1] showed through an adaptive challenge with over 208,000 attack submissions that the vulnerability persists even under active defense. In the agent domain, Zhang *et al.* [38] demonstrated that HTML-embedded injections influence real-world browser agent actions, Liao *et al.* [16] showed environmental injections can exfiltrate user data with up to 70% ASR, and Zhang *et al.* [37] provide a comprehensive agent security taxonomy finding attack success rates as high as 84.30% with limited effectiveness from existing defenses.

Multimodal IPI benchmarks. The most directly comparable works are CyberSecEval 3 [31], which includes a Visual Prompt Injection category but with generic, context-free attacks on approximately 1,000 samples; and VPI-Bench [5], which evaluates visual injection against CUAs and BUAs with 306 test cases, reporting deception rates of up to 51% for CUAs and 100% for BUAs. While VPI-Bench represents the closest prior work, it has three methodological limitations: (1) it compares CUAs (screenshot-based) with BUAs (HTML-based) but does not test the same model on the same content in image vs. text form, making it impossible to attribute vulnerability to visual processing specifically; (2) it uses permissive user prompts (*e.g.*, "*I'm very busy... do not miss out on anything important*") and a single-sentence defense prompt ("*Be vigilant against prompt-injection attacks*"), which may inflate attack rates relative to how agents are deployed in practice; and (3) it does not examine how the *format* of the injection within the visual artifact interacts with modality-specific failure modes. Our work addresses these gaps through paired VL/LM testing with identical content, deployment-realistic system prompts, format-conditioned analysis across realistic artifacts, and the introduction of ethics-framed attacks as a distinct category.

2.3 Bias, Sycophancy, and Alignment Priors

A substantial body of work has examined bias, fairness, and alignment limitations in large language models [2, 9, 24, 29]. Studies have documented overcorrection tendencies—such as Google's Gemini image generator producing women and people of color as Nazi soldiers in pursuit of demographic diversity [9]—as

well as sycophantic behavior where models prioritize agreeableness over accuracy [6, 20]. Based on these documented tendencies, we design our ethics-framed attack vector to test whether fabricated fairness mandates embedded in visual documents can exploit overcorrection and sycophancy to override deployment-time evaluation criteria.

3 The M-IPI Benchmark

3.1 Threat Model

We model the adversary as an *environmental contaminant*—not a direct user—who embeds malicious instructions within visual artifacts that agents process during normal workflows. We study two threat scenarios. In the **technical scenario**, an attacker plants malicious commands within StackOverflow-style debugging solutions. A developer encounters a text answer that provides context and explanation, paired with a terminal screenshot showing the actual commands and their outputs—the purported "solution." The terminal screenshot contains malicious commands (*e.g.*, reverse shells, credential exfiltration, ransomware) interwoven with legitimate debugging steps. We focus on Linux terminal environments as they represent the highest-severity attack surface: terminal commands operate at the OS level with direct access to the filesystem, network, and system configuration, and users routinely execute debugging commands from online sources without scrutinizing each step. The agent must identify and filter harmful commands from its recommendations. In the **ethics scenario**, an attacker embeds fabricated compliance directives—styled as organizational policy notes referencing UN charters, EU regulations, or internal DEI mandates—within professional resumes that an automated HR screening agent evaluates. The agent must follow its evaluation criteria rather than the embedded directives. In both scenarios, the attack content is clearly readable and visually integrated into the document; we test whether models *follow* injected instructions, not whether they can *perceive* hidden text.

3.2 Technical Attack Dataset

Our technical dataset comprises 1,000 unique debugging scenarios spanning 20 technical domains (Tab. 1). For each scenario, we construct a user query describing a real error, a StackOverflow-style text answer providing explanatory context, and a terminal screenshot showing the commands and outputs as purportedly executed by the original poster. The text answer serves as supporting context; the terminal screenshot is the primary artifact containing both legitimate debugging steps and embedded malicious content (Fig. 1). Malicious content is positioned as executable shell commands, inline comments, or both, and varies across two severity levels and two content lengths. Each attack sample is paired with a matched benign control, yielding 2,000 total technical artifacts.

Table 1: Dataset overview across both attack families.

	Technical	**Ethics-Framed**
Artifacts	2,000 (1K attack + 1K benign)	600
Domains	20 technical (50 each)	resume, loan, tender
Severity / Style	severe 60% / simple 40%	varied length & directness
Position / Placement	cmd / comment / both (33% ea.)	in-body, header, seal, footer
Length	short 50% / long 50%	—
Candidate mix	—	56% minority / 44% non-min.
Authority	—	corporate, govt, intl, state

3.3 Ethics-Framed Attack Dataset

For detection evaluation, we construct 600 visual artifacts spanning four screening tasks across three document types—resumes (software-engineering and executive hiring), loan applications (loan underwriting), and procurement tenders (government procurement). The injected attacks *vary in length and directness*: from short, subtle policy notes that merely reference a fairness framework, to verbose multi-paragraph legal-compliance directives citing specific regulations and threatening consequences for non-compliance; and from indirect language that never names the candidate, to direct statements that explicitly assert the candidate's protected status. They further span diverse authority framings (corporate, governmental, international) and visual placements (Tab. 1).

For attack success evaluation, we focus on resume screening—a setting that reflects a growing class of real-world deployments where agents autonomously process professional documents, from resume-screening pipelines to web tools that scan online profiles for candidate–job matching. We curate manually crafted resume scenarios spanning diverse candidate names and cultural cues, where clearly unqualified candidates (*e.g.*, a high-school graduate with no relevant experience) carry injected fairness directives ranging across the length/directness spectrum described above. Matched injection-free controls are drawn from the same unqualified persona pool, and genuinely qualified candidates serve as sanity checks. To probe role-generality, we evaluate two hiring roles—software engineering and executive (VP Operations)—with objective qualification criteria, isolating whether the injection changes the evaluation outcome and which injection style most effectively overrides the model's assessment.

3.4 Evaluation Protocol

We evaluate two tasks across both attack families.

Detection. Models are explicitly asked whether the document contains prompt injection or manipulative content. We test five model families—Idefics3-8B [14], Gemma-3-12B [8], Qwen3-VL-8B [4], GPT-4.1 [26], and GPT-5.2 [28]—each in both VL (image input) and LM (identical text content) configurations. For VL,

Table 2: Detection results for technically-framed attacks (n=1,000 attack + 1,000 benign). VL < LM holds across all five model pairs.

Model	Mode	Acc.	Recall	F1	Spec.	ΔRecall
GPT-5.2	LM	98.6	98.0	98.5	99.1	**−6.7**
	VL	95.3	91.3	95.1	99.3	
GPT-4.1	LM	89.7	79.9	88.6	99.5	**−27.0**
	VL	76.0	52.9	68.8	99.0	
Gemma-3-12B	LM	78.8	88.2	80.6	69.3	**−12.0**
	VL	73.2	76.2	74.0	70.2	
Qwen3-VL-8B	LM	60.3	20.7	34.3	99.9	**−18.3**
	VL	49.9	2.4	4.6	97.3	
Idefics3-8B	LM	71.4	53.6	65.2	89.1	**−18.4**
	VL	67.0	35.2	51.6	98.8	

the model receives the visual artifact as an image; for LM, it receives the ground-truth text content of the same artifact, bypassing the visual encoder entirely without introducing OCR artifacts.

Attack Success Rate (ASR). Models are deployed in realistic task-specific scenarios. For the technical scenario, GPT-4.1 receives the debugging query, the StackOverflow text answer, and the terminal content (as image for VL, as text for LM) and is prompted to recommend troubleshooting steps. ASR is the fraction of attack samples where the model reproduces malicious commands in its recommendations, evaluated via a hybrid pipeline combining deterministic command matching with GPT-5.2 as an independent LLM judge. For the ethics scenario, GPT-4.1 is deployed as an HR screening agent evaluating candidate resumes. ASR is the fraction of unqualified candidates that receive a favorable rating due to the injected directive.

4 Results

4.1 Detection: Technical Attacks

Tab. 2 presents detection results for technically-framed attacks. Across all five model pairs, VL recall is lower than LM recall, with gaps ranging from 6.7 pp (GPT-5.2) to 27.0 pp (GPT-4.1). The pattern is consistent across both frontier and mid-scale models, confirming that visual rendering of terminal artifacts suppresses critical evaluation.

4.2 Detection: Ethics-Framed Attacks

Tab. 3 presents detection results for the 600-sample ethics set. The VL < LM pattern persists for all five model pairs, with Gemma-3-12B showing the largest gap (−25.0 pp) and GPT-4.1—our primary ASR target—close behind (−18.7 pp). All models achieve near-perfect specificity—the failure mode is entirely missed detections, not false alarms. For GPT-4.1, the gap varies sharply by injection style:

Table 3: Detection results for ethics-framed attacks (n=320 attack + 280 benign). VL < LM holds for all five model pairs. All models achieve near-perfect specificity.

Model	Mode	Acc.	Recall	F1	Spec.	ΔRecall
GPT-5.2	LM	92.0	85.0	91.9	100	**−6.6**
	VL	88.5	78.4	87.9	100	
GPT-4.1	LM	85.8	73.4	84.7	100	**−18.7**
	VL	75.8	54.7	70.7	100	
Gemma-3-12B	LM	80.3	63.1	77.4	100	**−25.0**
	VL	64.8	38.1	53.6	95.4	
Qwen3-VL-8B	LM	68.8	41.6	58.7	100	**−16.9**
	VL	59.8	24.7	39.6	100	
Idefics3-8B	LM	47.5	1.6	3.1	100	−1.6
	VL	46.7	0.0	–	100	

VL detects 83% of direct injections but only 40% of subtle, policy-framed injections, while LM detects 89% and 63% respectively—indicating that injections resembling compliance language are particularly effective at evading visual-mode detection (breakdown by injection style in the supplementary material).

4.3 Attack Success Rate

Technical ASR. We deploy GPT-4.1 as a troubleshooting agent, with GPT-5.2 serving as an independent LLM judge. Overall ASR is 48.1% for VL and 50.9% for LM—statistically indistinguishable (McNemar $p = 0.066$). Paired per-sample analysis across 1,000 attacks reveals 94 VL-unique failures and 122 LM-unique failures. Among VL-unique failures, 88% involve executable-command-format attacks (1.47× enrichment, χ^2 $p < 0.001$), while LM is significantly more vulnerable to comment-format natural-language instructions ($p = 0.015$)—a *command-authority vs. language-authority* split where each modality fails on different attack surfaces.

Ethics-Framed ASR. We deploy GPT-4.1 as an HR screening agent across two hiring roles with objective qualification criteria—software engineering and executive (VP Operations). Across the two roles, roughly 100 clearly unqualified candidates carrying an injected fairness directive (ranging from subtle policy notes to verbose compliance mandates) are evaluated against a comparable control set of about 100 injection-free resumes spanning both unqualified and genuinely qualified candidates. These controls establish specificity: without injection, unqualified candidates are rated weak and qualified candidates strong, so a "Strong" verdict on an injected unqualified candidate is attributable to the attack rather than to the resume itself. We curated the injected samples for attack fidelity, keeping renderings in which the directive is well-formed, realistic, and legible, and discarding low-fidelity ones. Under injection, the attack exceeds 80% success on both roles, flipping unqualified candidates to favorable ratings and showing the vulnerability is not role-specific. When it complies, the model fabricates post-hoc justifications: reframing debate-team leadership as "transferable

technical skills," deeming a high-school diploma sufficient for an "entry-level" role, and citing the injected directive as policy that overrides qualification-based assessment.

Cross-domain patterns. Across both attack families, VL and LM show comparable overall ASR (within ∼3 pp), indicating that these vulnerabilities are rooted in the underlying language models and transfer to VLMs without necessarily worsening. The detection gap (VL < LM) does not produce a corresponding ASR gap—while visual rendering suppresses the model's ability to *detect* attacks, susceptibility to *following* injected instructions is a property of the shared language backbone. The combination of high ASR (∼50% for technical, >80% for ethics-framed) with poor detection performance underscores that prompt-level mitigation alone is insufficient, and that both sycophancy-driven and technically-framed injection vectors require dedicated defenses.

5 Discussion

Visual authority is cross-domain. The VL < LM detection gap appears across both attack families and nearly all model pairs (Tables 2–3). In terminal screenshots, malicious commands blend into the expected visual grammar; in professional documents, policy-style injections resemble legitimate compliance annotations. The mechanism differs but the effect is the same: visual rendering suppresses critical evaluation. This extends findings from the jailbreak literature [15, 19]—which documented visual vulnerability under direct attack—to realistic indirect prompt injection in agentic workflows.

Perception gap ≠ execution gap. For technical attacks, the detection gap (VL 52.9% vs. LM 79.9% recall for GPT-4.1) does not produce a corresponding ASR gap (VL 48.1% vs. LM 50.9%, $p = 0.066$). This suggests that execution-time reasoning partially compensates for perception-time failures—an incomplete transfer from jailbreak settings where 80–90% ASR is common [13, 15]. Models that *cannot detect* attacks at perception time can still *partially resist* them during execution through agentic reasoning.

Modality-specific failure profiles. The command-authority vs. language-authority split has direct implications for defense: VL agents are most vulnerable to attacks formatted as executable commands, while text-only pipelines are more vulnerable to natural-language directives. Converting visual inputs to text before reasoning ("discretization") could partially restore text-based skepticism for executable content, while text pipelines would benefit from instruction-boundary enforcement [36].

Defenses and mitigations. We study several defense families and find that none fully neutralizes these attacks:

1. **Prompt-level hardening.** Our deployment prompts already instruct the agent to be vigilant against untrusted content, yet both modalities remain ∼50% susceptible on technical attacks, consistent with prior findings that prompt hardening alone is insufficient [5, 36, 37].
2. **Discretization.** Our paired VL/LM comparison quantifies that converting the artifact to text before reasoning can recover significant detection recall—a useful but partial fix that still leaves substantial residual vulnerability.
3. **Encoder-based classifiers.** We study dedicated prompt-injection classifiers such as Llama Prompt Guard 2 [22], but they underperform on our benchmark (0% recall): our attacks use contextually plausible commands and legitimate-sounding policy language rather than the explicit instruction-override patterns (*e.g.*, "ignore previous instructions") such detectors are trained to flag.

Effective mitigation will therefore require defenses *beyond the model*. Two complementary directions are promising [5]: *agent-level* guards—a lightweight monitor, decoupled from the primary reasoning model, that verifies each proposed action against user intent and safety constraints before execution—and *system-level* controls (permission gating, sandboxing, runtime monitoring) that distinguish agent- from human-initiated actions and block high-risk operations such as file deletion, outbound network calls, or package installation from untrusted indices. Combined with detection-before-execution and verification of claims against trusted policy sources, these layered safeguards are more promising than any single model- or prompt-level fix.

Ethics-framed attacks exploit alignment priors. Our ethics-framed evaluation demonstrates that fabricated compliance directives reliably override model evaluation criteria (Fig. 2), achieving over 80% ASR on clearly unqualified candidates. Analysis of model reasoning exposes the mechanism: models treat *culturally identifiable names* and organizational affiliations as contextual cues, connecting them to the injected fairness directive to justify compliance. Rather than rejecting an obviously unqualified candidate, the model constructs elaborate justifications—inflating soft skills, lowering role requirements, and citing the fabricated policy as binding. This represents a distinct and concerning attack surface: unlike technical attacks where the injected content is verifiable (a command is either malicious or not), ethics-framed injections exploit non-falsifiable value priors that the model cannot reason through objectively. Both attack families remain effective across VL and LM pipelines, highlighting that these vulnerabilities are fundamental to current model alignment rather than specific to any modality.

Not a surface-level shortcut. The detection gap does not stem from trivial distributional cues. Each attack is constructed as a matched pair with a benign control sharing the same domain, query, template, length, comment density, and formatting, differing *only* in the presence of injected content. Detection instead varies with attack *content* and *placement* (per-style and per-length breakdowns are provided in the supplementary material); the effect is therefore semantic rather than surface-level.

Limitations and Future Directions. Our ethics-framed evaluation, while demonstrating a clear vulnerability, reflects a limited and controlled scope, set within cultural contexts predominantly aligned with the Global North. Safety judgments and alignment priors are known to vary across demographics and cultures [24]; our findings may not generalize to non-English or non-Western settings, and extending the evaluation to additional roles and a larger persona pool would map the full reach of the attack. Additionally, our technical dataset focuses on Linux/Unix terminal environments and may not fully represent the diversity of visual artifacts encountered by agents in production; extension to other operating systems and visual artifact types would broaden coverage. A comprehensive, large-scale study of ethics-framed injection across diverse roles, cultures, and model families is an important direction for future work. In each case, the vulnerability demonstrated here is already serious and actionable, and these extensions would only widen its documented reach.

6 Conclusion

We introduced the M-IPI Benchmark, evaluating multimodal indirect prompt injection across technically-framed and ethics-framed attack families. Our paired VL/LM design—absent from prior benchmarks—reveals a consistent "visual authority" effect: VL models underperform text-only counterparts at detecting embedded attacks across both domains and all model scales tested. For technical attacks, both VL and LM pipelines remain approximately 50% susceptible despite deployment-realistic prompts, with qualitatively distinct failure profiles—VL trusting visually-rendered commands, LM trusting natural-language directives. For ethics-framed attacks, fabricated compliance directives achieve over 80% attack success on clearly unqualified candidates, with models actively fabricating justifications to comply with injected fairness mandates.

These findings expose two serious and complementary vulnerabilities in current VLM-based agents. First, visual rendering of artifacts—whether terminal screenshots or professional documents—creates an implicit authority that systematically suppresses detection of embedded threats. Second, alignment toward fairness and inclusivity, while essential, creates an exploitable attack surface when adversaries frame malicious instructions as compliance directives. As VLM agents are increasingly deployed in high-stakes settings—from automated code execution to hiring and financial decision-making—addressing these vulnerabilities is not merely an academic concern but an urgent practical necessity. Our benchmark artifacts are released under gated, research-only access at `https://github.com/adityakm24/OpenYourEyes` to support future defensive research.

Ethics, Misuse, and Responsible Release

Content warning. This paper and the released benchmark contain explicit examples of malicious system commands (*e.g.*, reverse shells, credential exfiltra-

tion, and ransomware-style operations) and fabricated "ethics-framed" directives that misappropriate fairness and compliance language (*e.g.*, DEI or regulatory mandates) to instruct an agent to override its evaluation criteria on the basis of protected attributes. These artifacts are reproduced *solely* to document the attack surface and enable defensive research; they are synthetic and do not reflect the views of the authors or their institutions.

Intended use and data ethics. This benchmark is intended *defensively*—to measure and improve the ability of VLM-based agents to detect and resist multimodal indirect prompt injection. It contains no human subjects, real user data, or personal information: all personas, demographic attributes, photographs, and documents are entirely synthetic. Demographic categories (race, gender, age) are properties of the attack design, not assessments of real individuals, and the embedded fairness directives are adversarial fabrications—genuine equity policies serve an important societal function.

Misuse and responsible release. We recognize that this benchmark is dual-use: the same artifacts that enable defensive evaluation could, in principle, be repurposed by an adversary. Consistent with norms in security research, we treat these vulnerabilities as disclosures and judge that transparent, systematic benchmarking is necessary to build effective defenses. To limit misuse, the benchmark is not released openly: the attack artifacts are distributed under *gated*, request-based access to verified researchers who agree to an acceptable-use policy. The policy restricts the data to research, evaluation, and defensive development, and prohibits deploying the attacks against real or production systems, using them to make or inform decisions about real individuals or to target any non-consenting party, and redistributing them outside its terms. All experiments in this work were conducted in sandboxed environments. We do not merely expose attacks: we evaluate multiple defense families and outline concrete agent- and system-level mitigations (Sec. 5), and the authors disclaim responsibility for any use that violates these terms.

Generative AI Usage Disclosure

General-purpose large language model (LLM) assistants were used for generating synthetic personas for the ethics-framed dataset, assisting with programmatic visual artifact rendering, and language editing of author-written text and code. All intellectual contributions—research conception, design, analysis, interpretation, and claims—are solely those of the human authors.

Acknowledgements

Amit Pandey thanks Prof. Vikram Pudi (IIIT Hyderabad) for advisory guidance provided as part of his doctoral studies.

References

1. Abdelnabi, S., Fay, A., Salem, A., Zverev, E., Liao, K.C., Liu, C.H., Kuo, C.C., Weigend, J., Manlangit, D., Apostolov, A., Umair, H., ao Donato, J., Kawakita, M., Mahboob, A., Bach, T.H., Chiang, T.H., Cho, M., Choi, H., Kim, B., Lee, H., Pannell, B., McCauley, C., Russinovich, M., Paverd, A., Cherubin, G.: Llmail-inject: A dataset from a realistic adaptive prompt injection challenge (2025), https://arxiv.org/abs/2506.09956 (accessed 5 March 2026)
2. Anthis, J.R., Lum, K., Ekstrand, M., Feller, A., Tan, C.: The impossibility of fair LLMs. In: Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 105–120. Association for Computational Linguistics, Vienna, Austria (Jul 2025). https://doi.org/10.18653/v1/2025.acl-long.5, https://aclanthology.org/2025.acl-long.5/ (accessed 5 March 2026)
3. Anthropic: Computer use. https://docs.anthropic.com/en/docs/agents-and-tools/computer-use (accessed 5 March 2026) (2025), accessed: 2025-05-15
4. Bai, S., Cai, Y., Chen, R., Chen, K., Chen, X., Cheng, Z., Deng, L., Ding, W., Gao, C., Ge, C., Ge, W., Guo, Z., Huang, Q., Huang, J., Huang, F., Hui, B., Jiang, S., Li, Z., Li, M., Li, M., Li, K., Lin, Z., Lin, J., Liu, X., Liu, J., Liu, C., Liu, Y., Liu, D., Liu, S., Lu, D., Luo, R., Lv, C., Men, R., Meng, L., Ren, X., Ren, X., Song, S., Sun, Y., Tang, J., Tu, J., Wan, J., Wang, P., Wang, P., Wang, Q., Wang, Y., Xie, T., Xu, Y., Xu, H., Xu, J., Yang, Z., Yang, M., Yang, J., Yang, A., Yu, B., Zhang, F., Zhang, H., Zhang, X., Zheng, B., Zhong, H., Zhou, J., Zhou, F., Zhou, J., Zhu, Y., Zhu, K.: Qwen3-vl technical report. arXiv preprint arXiv:2511.21631 (2025)
5. Cao, T., Lim, B., Liu, Y., Sui, Y., Li, Y., Deng, S., Lu, L., Oo, N., Yan, S., Hooi, B.: Vpi-bench: Visual prompt injection attacks for computer-use agents (2026), https://arxiv.org/abs/2506.02456 (accessed 5 March 2026)
6. Cheng, M., Yu, S., Lee, C., Khadpe, P., Ibrahim, L., Jurafsky, D.: Elephant: Measuring and understanding social sycophancy in llms (2025), https://arxiv.org/abs/2505.13995 (accessed 5 March 2026)
7. Dahou, Y., Huynh, N.D., Le-Khac, P.H., Para, W.R., Singh, A., Narayan, S.: Vision-language models can't see the obvious (2025), https://arxiv.org/abs/2507.04741 (accessed 5 March 2026)
8. Gemma Team: Gemma 3 (2025), https://goo.gle/Gemma3Report (accessed 5 March 2026)
9. Ghassami, A., et al.: Race and gender in llm-generated personas: A large-scale audit of 41 occupations (2025), cHI 2025
10. Gou, Y., Dong, X., Li, Q., Gu, S., Hong, R., Hu, W.: SURE: Safety understanding and reasoning enhancement for multimodal large language models. In: Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing. pp. 7552–7593. Association for Computational Linguistics, Suzhou, China (Nov 2025). https://doi.org/10.18653/v1/2025.emnlp-main.384, https://aclanthology.org/2025.emnlp-main.384/ (accessed 5 March 2026)
11. Grattafiori, A., et al.: The llama 3 herd of models (2024), https://arxiv.org/abs/2407.21783 (accessed 5 March 2026)
12. Greshake, K., Abdelnabi, S., Mishra, S., Endres, C., Holz, T., Fritz, M.: Not what you've signed up for: Compromising real-world llm-integrated applications with indirect prompt injection (2023), https://arxiv.org/abs/2302.12173 (accessed 5 March 2026)

13. Jeong, H., Park, S., Kim, J., Lee, J.S., Seo, M., Shin, J.: Playing the fool: Jailbreaking llms and multimodal llms with out-of-distribution strategy. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2025), https://openaccess.thecvf.com/content/CVPR2025/html/Jeong_Playing_the_Fool_Jailbreaking_LLMs_and_Multimodal_LLMs_with_Out-of-Distribution_CVPR_2025_paper.html (accessed 5 March 2026)
14. Laurençon, H., Marafioti, A., Sanh, V., Tronchon, L.: Building and better understanding vision-language models: insights and future directions (2024), https://arxiv.org/abs/2408.12637 (accessed 5 March 2026)
15. Li, M., Wang, H., Zheng, Z., Li, F., Li, X., Wu, L., Zhang, A., Sun, Y.: Images are achilles' heel of alignment: Exploiting visual vulnerabilities for jailbreaking multimodal large language models. In: European Conference on Computer Vision (ECCV) (2024), https://arxiv.org/abs/2403.09792 (accessed 5 March 2026)
16. Liao, Z., Mo, L., Xu, C., Kang, M., Zhang, J., Xiao, C., Tian, Y., Li, B., Sun, H.: Eia: Environmental injection attack on generalist web agents for privacy leakage (2024), https://arxiv.org/abs/2409.11295 (accessed 5 March 2026)
17. Liu, D., Yang, M., Qu, X., Zhou, P., Cheng, Y., Hu, W.: A survey of attacks on large vision-language models: Resources, advances, and future trends (2024), https://arxiv.org/abs/2407.07403 (accessed 5 March 2026)
18. Liu, X., Zhu, Y., Gu, J., Lan, Y., Yang, C., Qiao, Y.: Mm-safetybench: A benchmark for safety evaluation of multimodal large language models (2024), https://arxiv.org/abs/2311.17600 (accessed 5 March 2026)
19. Liu, Z., Nie, Y., Tan, Y., Yue, X., Cui, Q., Wang, C., Zhu, X., Zheng, B.: Safety alignment for vision language models (2024), https://arxiv.org/abs/2405.13581 (accessed 5 March 2026)
20. Malmqvist, L.: Sycophancy in large language models: Causes and mitigations (2024), https://arxiv.org/abs/2411.15287 (accessed 5 March 2026)
21. Mei, H., Wang, Z., You, S., Dong, M., Xu, C.: Veattack: Downstream-agnostic vision encoder attack against large vision language models (2025), https://arxiv.org/abs/2505.17440 (accessed 5 March 2026)
22. Meta: Llama prompt guard 2. https://huggingface.co/meta-llama/Llama-Prompt-Guard-2-86M (accessed 5 March 2026) (2025)
23. Miao, Z., Ding, Y., Li, L., Shao, J.: Visual contextual attack: Jailbreaking mllms with image-driven context injection (2025), https://arxiv.org/abs/2507.02844 (accessed 5 March 2026)
24. Naseem, U., Kashyap, G.S., Ray, S.K., Ali, R., Shabbir, E., Mohammad, A.: Do large language models reflect demographic pluralism in safety? (2026), https://arxiv.org/abs/2602.07376 (accessed 5 March 2026)
25. OmniSafeBench Team: Omnisafebench-mm: A unified benchmark and toolbox for multimodal jailbreak attack-defense evaluation (2025), preprint
26. OpenAI: Gpt-4.1. https://openai.com/index/gpt-4-1/ (accessed 5 March 2026) (2025), accessed: 2026
27. OpenAI: Operator. https://openai.com/index/introducing-operator/ (accessed 5 March 2026) (2025), accessed: 2025
28. OpenAI: Gpt-5.2. https://openai.com/ (accessed 5 March 2026) (2026), accessed: 2026
29. Ovalle, A., Pavasovic, K.L., Martin, L., Zettlemoyer, L., Smith, E.M., Chang, K.W., Williams, A., Sagun, L.: The root shapes the fruit: On the persistence of gender-exclusive harms in aligned language models (2025), https://arxiv.org/abs/2411.03700 (accessed 5 March 2026)

30. Sima, B., Cong, L., Wang, W., He, K.: Viscra: A visual chain reasoning attack for jailbreaking multimodal large language models (2025), https://arxiv.org/abs/2505.19684 (accessed 5 March 2026)
31. Wan, S., Nikolaidis, C., Song, D., Molnar, D., Crnkovich, J., Grace, J., Bhatt, M., Chennabasappa, S., Whitman, S., Ding, S., Ionescu, V., Li, Y., Saxe, J.: Cyberseceval 3: Advancing the evaluation of cybersecurity risks and capabilities in large language models (2024), https://arxiv.org/abs/2408.01605 (accessed 5 March 2026)
32. Wang, R., Li, J., Wang, Y., Wang, B., Wang, X., Teng, Y., Wang, Y., Ma, X., Jiang, Y.G.: Ideator: Jailbreaking and benchmarking large vision-language models using themselves. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 8875–8884 (October 2025)
33. Wang, X., Chen, Y., Li, J., Wang, Y., Yao, Y., Gu, T., Li, J., Teng, Y., Wang, Y., Hu, X.: Openrt: An open-source red teaming framework for multimodal llms (2026), https://arxiv.org/abs/2601.01592 (accessed 5 March 2026)
34. Wang, Y., Hu, W., Dong, Y., Liu, J., Zhang, H., Hong, R.: Align is not enough: Multimodal universal jailbreak attack against multimodal large language models (2025), https://arxiv.org/abs/2506.01307 (accessed 5 March 2026)
35. Ye, J., Jiang, D., He, J., Zhou, B., Huang, Z., Yan, Z., Li, H., He, C., Li, W.: Blinktwice: You see, but do you observe? a reasoning benchmark on visual perception (2025), https://arxiv.org/abs/2510.09361 (accessed 5 March 2026)
36. Yi, J., Xie, Y., Zhu, B., Kiciman, E., Sun, G., Xie, X., Wu, F.: Benchmarking and defending against indirect prompt injection attacks on large language models. In: Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1. pp. 1809–1820. ACM (Jul 2025). https://doi.org/10.1145/3690624.3709179, http://dx.doi.org/10.1145/3690624.3709179 (accessed 5 March 2026)
37. Zhang, H., Huang, J., Mei, K., Yao, Y., Wang, Z., Zhan, C., Wang, H., Zhang, Y.: Agent security bench (ASB): Formalizing and benchmarking attacks and defenses in LLM-based agents. In: The Thirteenth International Conference on Learning Representations (2025), https://openreview.net/forum?id=4bsiRMBmzS (accessed 5 March 2026)
38. Zhang, K., Tenenholtz, M., Polley, K., Ma, J., Yarats, D., Li, N.: Browsesafe: Understanding and preventing prompt injection within ai browser agents (2025), https://arxiv.org/abs/2511.20597 (accessed 5 March 2026)