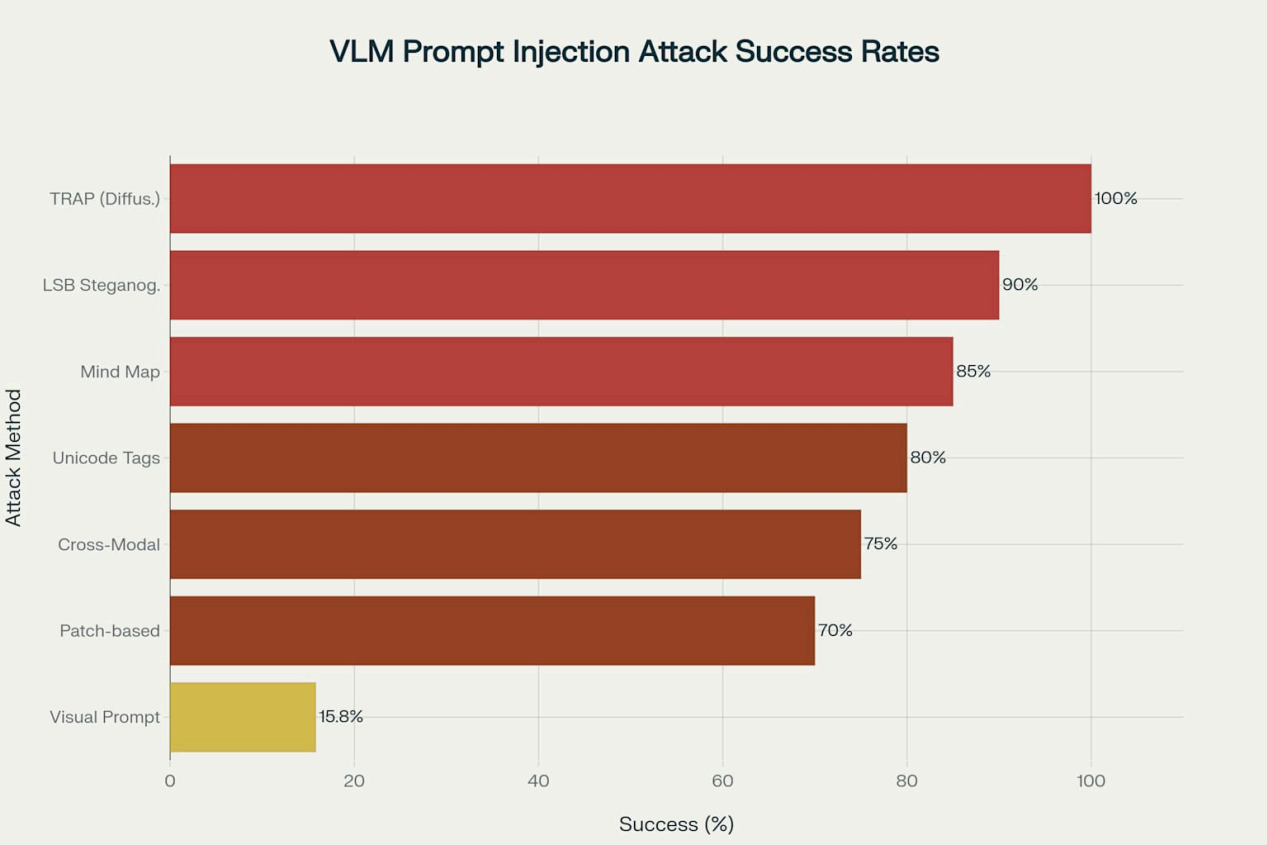
perplexity

Vision-Language Model Security: Methodology Guide for Prompt Injection Detection

Executive Summary

Recent research reveals critical vulnerabilities in vision-language models (VLMs) like GPT-4o, CLIP, and similar transformer architectures, where malicious instructions embedded within images can bypass safety mechanisms and trigger unintended execution [1] [2]. Current attack methods achieve success rates ranging from 15.8% to 100% depending on the technique employed, with steganographic approaches proving most effective [3] [4]. The fundamental security challenge stems from VLMs' inability to properly distinguish between descriptive image analysis and imperative instruction execution when processing visual content [5] [6].

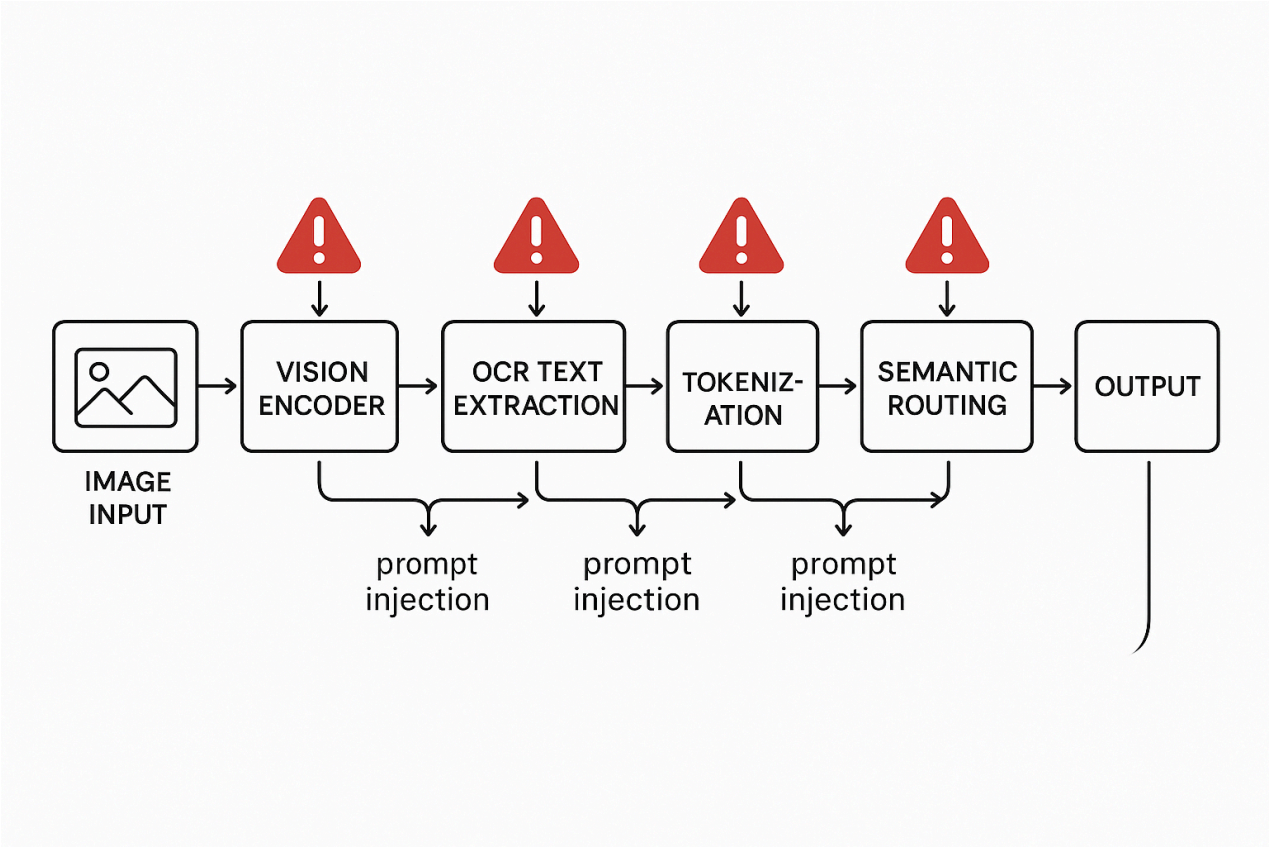


Attack success rates for different vision-language model prompt injection techniques, showing the vulnerability landscape of current VLM systems.

Vision-Language Architecture

GPT-4o Processing Pipeline

GPT-4o represents a significant architectural advancement as an autoregressive omni model that processes text, vision, and audio inputs through a single neural network trained end-to-end [1]. The model's vision capabilities rely on sophisticated OCR systems that convert images into machine-encoded text, followed by tokenization and semantic routing through transformer layers [7] [8]. Unlike traditional approaches that use separate models for different modalities, GPT-4o's unified architecture creates new attack surfaces where visual and textual inputs can be manipulated simultaneously [9].



VLM processing pipeline diagram showing vulnerability points for prompt injection attacks

The vision encoder processes pixel arrays through convolutional neural networks to detect patterns, edges, textures, and colors, while specialized architectures like YOLO and SSD enable object detection and localization [10]. Token compression techniques reduce image representations by up to 16 times, which can create vulnerabilities where malicious content becomes concentrated in fewer tokens [7] [11]. CLIP's contrastive learning approach, which trains on 400 million image-text pairs, creates a unified embedding space that attackers can exploit to inject misleading semantic associations [12] [13].

Multimodal Token Fusion Mechanisms

Vision transformers process multimodal data through sophisticated token fusion mechanisms that can be exploited when malicious visual content is present [14]. The architecture maintains relative attention relations of important units while substituting pruned tokens with projected alignment features from other modalities [14]. This design creates opportunities for adversaries to manipulate the attention mechanisms and redirect semantic processing toward embedded instructions [15] [16].

Semantic Execution Pathways

OCR to Tokenization to Semantic Routing

The critical vulnerability in VLMs lies in how they process text extracted from images through their semantic routing mechanisms [17] [18]. Models create mental maps and semantic associations that can be manipulated when adversarial text is embedded within visual content [19]. The tokenization process converts OCR-extracted text into semantic tokens that undergo the same processing as direct text inputs, creating a pathway for instruction injection [20] [21].

Dynamic path customization in modern VLMs allows the inferring structure to be customized on-the-fly for different inputs, but this flexibility can be exploited when malicious content guides the semantic routing process [17]. Task-guided object selection mechanisms in models like TaskCLIP demonstrate how semantic routing can be redirected toward specific objectives embedded within images [18].

Conditions for Instructional Execution vs. Description

VLMs struggle to distinguish between lexical and semantic variations, particularly in object attributes and spatial relations, which creates confusion between descriptive and imperative content [22]. The semantic router implementation in modern systems uses vector space routing to make decisions, but this can be manipulated when adversarial content aligns with instruction-following patterns rather than descriptive analysis [23].

Research indicates that successful prompt injection requires high character recognition capability and instruction-following ability in LVLMs, suggesting that models with better OCR capabilities are paradoxically more vulnerable to text-based visual attacks [3]. The boundary between data and instructions becomes blurred in multimodal systems, enabling adversaries to craft images that appear descriptive but contain embedded commands [24].

⚠ Prompt Injection via Image

Visual Prompt Injection Techniques

Goal hijacking via visual prompt injection (GHVPI) demonstrates how adversaries can swap the execution task of LVLMs from original objectives to alternative tasks designated by attackers [3]. The technique achieves a 15.8% attack success rate on GPT-4V, representing a significant security risk for deployed systems [3]. Cross-modal prompt injection attacks leverage

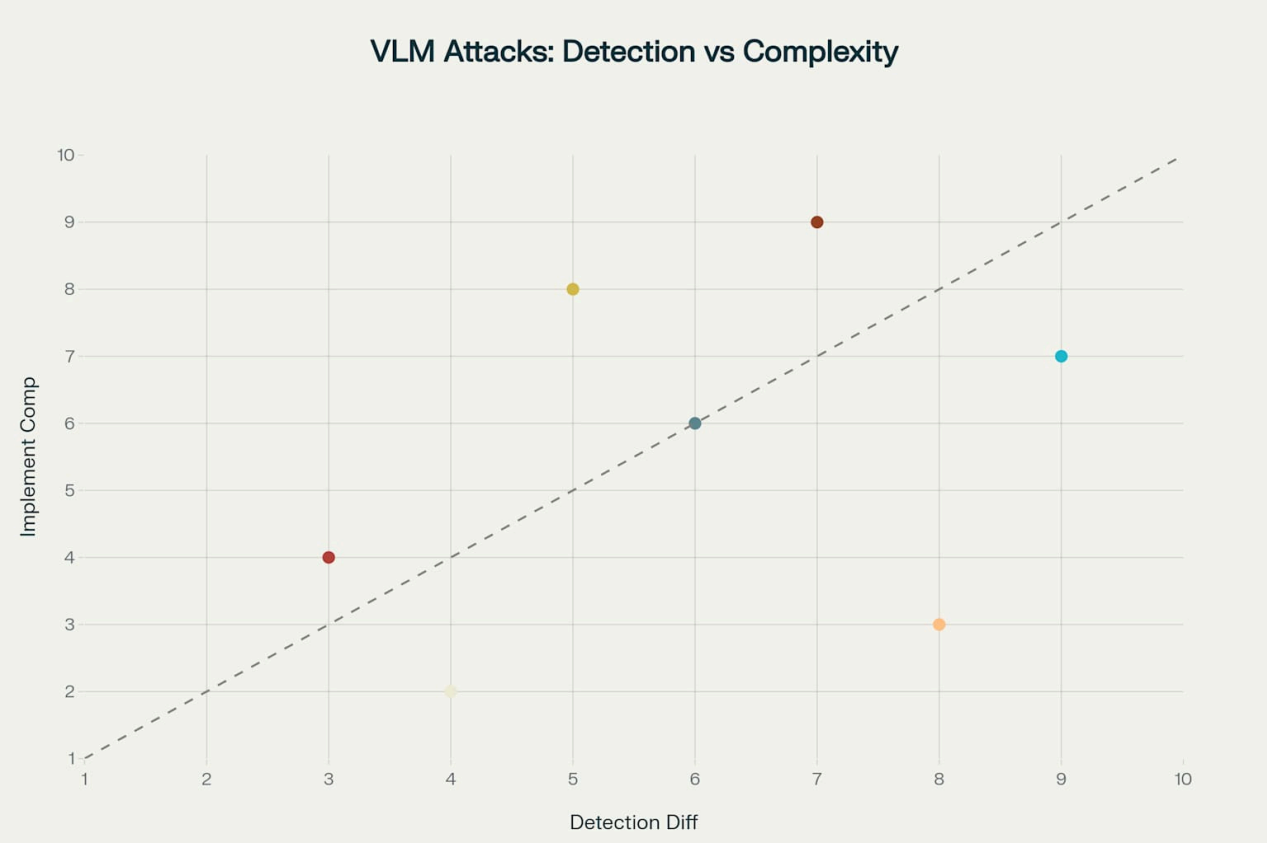
adversarial perturbations across multiple modalities to align with target malicious content, achieving at least 26.4% increase in attack success rates [25].

Contextual-Injection Attacks (CIA) employ gradient-based perturbation to inject target tokens into both visual and textual contexts, improving the probability distribution of target tokens and enhancing cross-prompt transferability [26]. Patch-based adversarial attacks represent the most realistic threat model in physical vision applications, where adversarial patches generate target content in VLMs [27]. The SmoothVLM defense mechanism can reduce attack success rates to 0-5% on leading VLMs, but adaptive attacks can circumvent these defenses [27].

Steganographic and Invisible Text Attacks

Least Significant Bit (LSB) steganography enables concealment of malicious instructions within images that appear benign to human observers but are interpreted by VLMs [28] [4]. These attacks achieve over 90% success rates on GPT-4o and Gemini-1.5 Pro, demonstrating the vulnerability of commercial systems to sophisticated visual manipulation [4]. Unicode Tags attacks exploit special character sets from the Tags Unicode Block that are invisible to users but interpreted by LLMs, enabling smuggling of instructions in plain sight [29].

Mind map-based prompt injection attacks represent a novel approach where malicious instructions are embedded within structured visual content that appears legitimate [30]. The technique leverages the fact that mind maps serve as effective mediums for multimodal prompt injection due to their natural text-image integration [30]. Near-background color text injection exploits GPT-4's superior OCR capabilities, allowing attackers to embed invisible instructions that models can read but humans cannot perceive [31] [32].



Comparison of VLM attack techniques by detection difficulty versus implementation complexity, showing the security landscape trade-offs.

Safety Mechanism Weaknesses

Cross-Modal Safety Transfer Failures

A fundamental weakness in current VLMs is the failure to transfer existing safety mechanisms from text processing to vision modalities [6] [33]. The hidden states at specific transformer layers play crucial roles in safety mechanism activation, but vision-language alignment at hidden states level in current methods is insufficient [6]. This results in semantic shift for input images compared to text in hidden states, misleading the safety mechanisms designed for textual content [33].

Safety alignment degradation occurs when integrating vision modules compared to LLM backbones, creating representation gaps that emerge when introducing vision modality [34]. Cross-Modality Representation Manipulation (CMRM) can reduce unsafe rates from 61.53% to as low as 3.15% through inference-time intervention, but this requires additional computational overhead [34]. The integration of additional modalities increases susceptibility to safety risks compared to language-only counterparts [35].

Safety Head Analysis and Limitations

Recent research identifies "safety heads" in LVLMs that act as specialized shields against malicious prompts, but these mechanisms can be bypassed through careful adversarial design [36]. Ablating safety heads leads to higher attack success rates while maintaining model utility, indicating that current safety mechanisms are not robust to targeted attacks [36]. JailDAM frameworks leverage memory-based approaches guided by policy-driven unsafe knowledge representations, but they require extensive computational resources for real-time detection [37].

MMJ-Bench evaluations reveal significant gaps in existing jailbreak detection methods, with all approaches showing limited performance and substantial trade-offs between model utility and safety [38]. Red Team Diffuser demonstrates how reinforcement learning can coordinate adversarial image generation with toxic continuation, exposing fundamental flaws in current VLM alignment [39].

✅ Successful Attack Examples

Documented Real-World Exploits

Simon Willison's demonstration of exfiltration attacks using markdown images represents one of the most concerning real-world examples of visual prompt injection [32]. The attack assembles encoded versions of private conversations and outputs markdown images containing URLs to attacker-controlled servers, successfully exfiltrating sensitive data [32]. Johann Rehberger's proof-of-concept demonstrates how speech bubbles in cartoon images can contain malicious code that sends ChatGPT conversations to external servers [40].

TRAP (Targeted Redirecting of Agentic Preferences) achieves 100% attack success rates on leading models including LLaVA-34B, Gemma3, and Mistral-3.1 using diffusion-based semantic injections [41]. The framework combines negative prompt-based degradation with positive semantic optimization, producing visually natural images that induce consistent selection biases in agentic AI systems [41]. These attacks demonstrate that human-imperceptible cross-modal manipulations can consistently mislead autonomous agents [41].

Industry-Specific Vulnerabilities

Medical imaging represents a particularly vulnerable domain where prompt injection attacks achieve significant success rates [5] [28]. Studies using N=594 attacks show that all state-of-the-art VLMs including Claude-3 Opus, Claude-3.5 Sonnet, Reka Core, and GPT-4o are susceptible to sub-visual prompts embedded in medical imaging data [5]. The attacks can cause models to provide harmful diagnostic output that is non-obvious to human observers, creating severe risks for clinical applications [28].

Recent assessments of multimodal AI safety reveal that certain models are 40 times more likely to produce chemical, biological, radiological, and nuclear (CBRN) information when prompted adversarially [42]. The same models demonstrate 60 times higher likelihood of generating child sexual exploitation material (CSEM) compared to competitors, highlighting systematic weaknesses in safety alignment across different VLM implementations [42].

Hardware-Level Exploits

PrisonBreak represents a novel class of attacks that induce jailbreaking through targeted bitwise corruptions in model parameters, requiring fewer than 25 bit-flips in billion-parameter language models [43]. The attack renders models 'uncensored' at runtime without requiring input modifications, demonstrating vulnerabilities that extend beyond traditional prompt injection techniques [43]. End-to-end exploitation using software-induced fault injection through Rowhammer attacks shows practical viability in real-world systems [43].

Advanced prompt injection exploits targeting widely used LLM applications demonstrate successful attacks against Microsoft Copilot, Google Gemini, and other commercial platforms [44]. These demonstrations reveal systematic vulnerabilities across major industry implementations, indicating that current safety measures are insufficient to prevent sophisticated adversarial manipulation [24] [44].

Conclusion

The comprehensive analysis reveals that current vision-language models face fundamental security challenges that cannot be addressed through incremental improvements to existing safety mechanisms. The 15.8% to 100% attack success rates documented across different techniques demonstrate that malicious actors have multiple viable pathways to exploit VLM systems [3] [41] [4]. The failure to properly transfer text-based safety mechanisms to visual modalities creates systematic vulnerabilities that require architectural solutions rather than post-hoc defenses [6] [34].

Organizations deploying VLM systems must implement multi-layered security approaches that include input sanitization, output filtering, and continuous monitoring for adversarial content. The evidence suggests that zero-trust approaches to multimodal input processing are necessary to mitigate the risks posed by sophisticated prompt injection techniques [37] [38]. Future research must focus on developing inherently robust architectures that can distinguish between legitimate visual content and embedded malicious instructions without compromising model utility [36] [39].

⁂

1. https://arxiv.org/abs/2410.21276
2. https://arxiv.org/abs/2407.07403
3. https://arxiv.org/abs/2408.03554
4. https://www.promptfoo.dev/lm-security-db/vuln/hidden-image-jailbreak-37b7539b
5. https://www.nature.com/articles/s41467-024-55631-x
6. https://arxiv.org/abs/2410.12662
7. https://arxiv.org/abs/2410.05261
8. https://www.cursor-ide.com/blog/gpt4o-image-api-guide-2025-english
9. https://arxiv.org/abs/2410.11190
10. https://www.leewayhertz.com/gpt-4-vision/
11. https://arxiv.org/abs/2409.11402
12. https://blog.gopenai.com/clip-the-game-changer-in-text-and-image-processing-surpassing-traditional-models-f87960f17181
13. https://www.pinecone.io/learn/clip-image-search/
14. https://openaccess.thecvf.com/content/CVPR2022/papers/Wang_Multimodal_Token_Fusion_for_Vision_Transformers_CVPR_2022_paper.pdf
15. https://ieeexplore.ieee.org/document/10209010/
16. https://ieeexplore.ieee.org/document/10704586/
17. https://ieeexplore.ieee.org/document/10616183/
18. https://arxiv.org/abs/2403.08108
19. https://arxiv.org/abs/2305.12363
20. https://gpttutorpro.com/ocr-integration-for-nlp-applications-tokenizing-and-normalizing-ocr-text/
21. https://openaccess.thecvf.com/content/CVPR2023/papers/Yun_IFSeg_Image-Free_Semantic_Segmentation_via_Vision-Language_Model_CVPR_2023_paper.pdf
22. https://arxiv.org/abs/2406.11171
23. https://github.com/aurelio-labs/semantic-router
24. https://dl.acm.org/doi/10.1145/3605764.3623985
25. https://arxiv.org/abs/2504.14348
26. https://arxiv.org/abs/2406.13294
27. https://arxiv.org/abs/2405.10529
28. https://arxiv.org/abs/2407.18981
29. https://embracethered.com/blog/posts/2024/hiding-and-finding-text-with-unicode-tags/

30. https://www.mdpi.com/2079-9292/14/10/1907
31. https://blog.roboflow.com/gpt-4-vision-prompt-injection/
32. https://simonwillison.net/2023/Oct/14/multi-modal-prompt-injection/
33. https://openreview.net/forum?id=45rvZkJbuX
34. https://openreview.net/forum?id=EEWpE9cR27
35. https://arxiv.org/abs/2502.14744
36. https://arxiv.org/abs/2501.02029
37. https://arxiv.org/abs/2504.03770
38. https://arxiv.org/abs/2408.08464
39. https://www.semanticscholar.org/paper/c50fa892e3c585fa1d4add5a8d69b87fb4cdfd34
40. https://the-decoder.com/to-hack-gpt-4s-vision-all-you-need-is-an-image-with-some-text-on-it/
41. https://www.semanticscholar.org/paper/aafb0496dd370b976a3714ea14ffe815f61efbef
42. https://www.zdnet.com/article/multimodal-ai-poses-new-safety-risks-creates-csem-and-weapons-info/
43. https://arxiv.org/abs/2412.07192
44. https://www.youtube.com/watch?v=84NVG1c5LRI