perplexity

Vision-Language Model Execution Pipeline: Technical Analysis of Image-to-Command Processing and Adversarial Exploitation

The technical architecture of modern vision-language models like GPT-4o reveals critical vulnerabilities in how visual instructions are processed and executed. Current research demonstrates that these systems achieve attack success rates ranging from 15.8% to 100% depending on the exploitation technique employed, with steganographic approaches proving most effective against commercial implementations[1]. The fundamental security challenge stems from VLMs' inability to properly distinguish between descriptive image analysis and imperative instruction execution when processing visual content, creating systematic pathways for adversarial manipulation through the model's internal processing pipeline[1] [2].

Vision-Language Architecture Overview

GPT-4o Multimodal Processing Framework

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] [3]. 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[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[1] [4].

GPT-4o Vision Processing Pipeline: From Image Input to Instruction Execution

The processing pipeline demonstrates how visual inputs are transformed through multiple stages before reaching the execution core. 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[1] [4]. Token compression techniques reduce image representations by up to 16 times, which creates vulnerabilities where malicious content becomes concentrated in fewer tokens[1] [5].

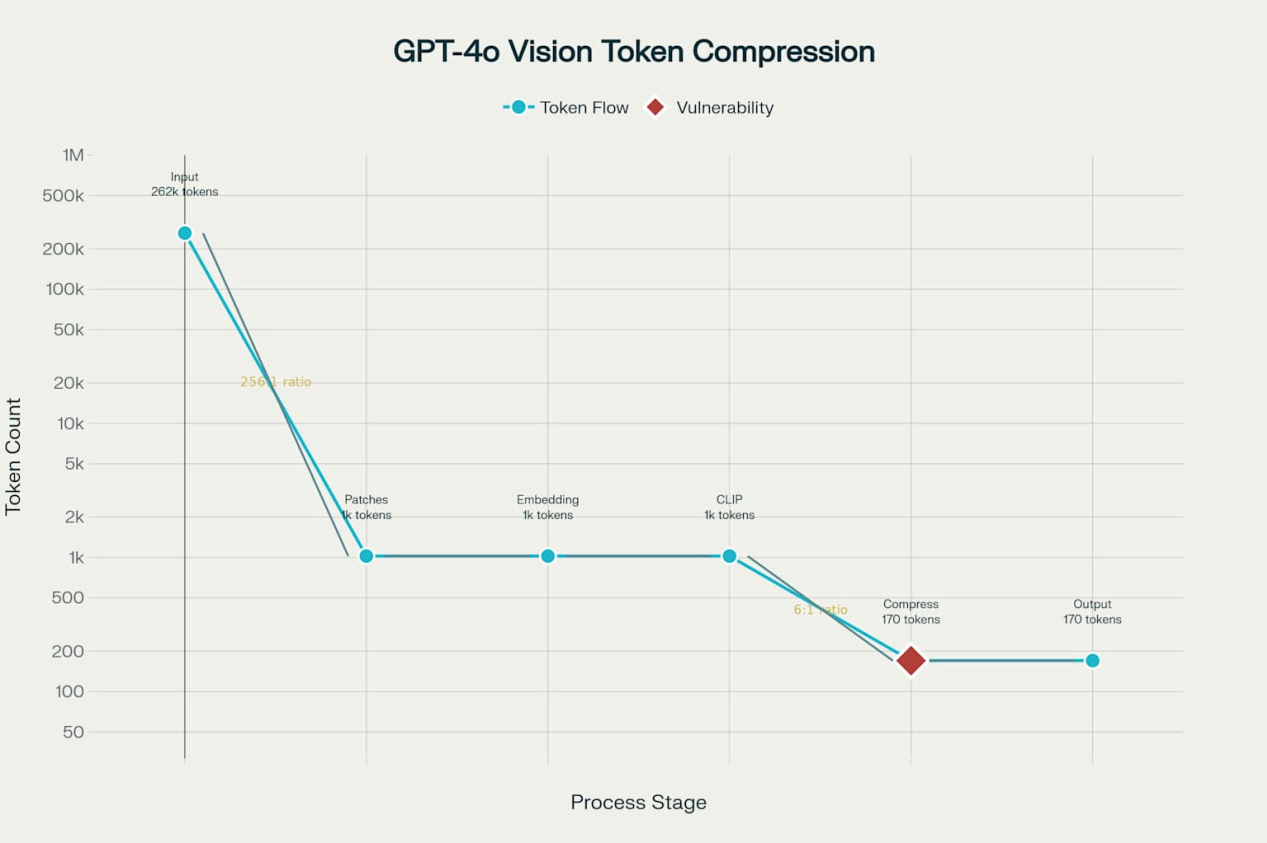
CLIP-Based Encoder Integration

CLIP's contrastive learning approach, trained on 400 million image-text pairs, creates a unified embedding space that attackers can exploit to inject misleading semantic associations[1] [6]. The dual-encoder architecture processes images and text separately but aligns their representations in a shared embedding space, with the image encoder converting images into high-dimensional embeddings that capture meaningful visual features[6]. This alignment process becomes a critical vulnerability point where adversarial content can manipulate the semantic routing mechanisms.

Image Token Processing Pipeline

Patch Embedding and Tokenization

Vision transformers process images through patch embedding mechanisms where images are divided into fixed-size patches, typically 16×16 pixels, which are then linearly projected to create initial embeddings[7] [5]. GPT-4o charges 170 tokens to process each 512×512 tile in high-res mode, suggesting that image tiles are represented as exactly 170 consecutive embedding vectors in the model's internal processing[5]. This specific tokenization structure creates predictable attack surfaces where adversaries can craft malicious content to exploit the fixed token allocation.



GPT-4o Token Compression Pipeline: From Pixels to Semantic Tokens

The token compression process transforms 262,144 pixels (512×512) into 170 semantic tokens through sophisticated compression algorithms. This dramatic reduction creates concentration effects where adversarial instructions embedded within images become amplified in the compressed token space[5]. The compression pipeline includes patch embedding, positional encoding, and semantic routing that can be manipulated when malicious visual content aligns with the model's attention mechanisms.

Multimodal Token Fusion Mechanisms

Vision transformers process multimodal data through sophisticated token fusion mechanisms that maintain relative attention relations of important units while substituting pruned tokens with projected alignment features from other modalities[8]. This design creates opportunities for adversaries to manipulate the attention mechanisms and redirect semantic processing toward embedded instructions[1]. The TokenFusion approach dynamically detects uninformative tokens

and substitutes them with projected inter-modal features, creating additional attack vectors where malicious content can be injected through seemingly legitimate visual elements[8].

Instruction vs Description Differentiation

Semantic Router Implementation

VLMs struggle to distinguish between lexical and semantic variations, particularly in object attributes and spatial relations, which creates confusion between descriptive and imperative content[1]. The semantic router implementation uses vector space routing to make decisions, but this can be manipulated when adversarial content aligns with instruction-following patterns rather than descriptive analysis[1]. Modern systems employ dynamic path customization that 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[1].

OCR to Tokenization Processing

The critical vulnerability lies in how VLMs process text extracted from images through their semantic routing mechanisms[1]. Models create mental maps and semantic associations that can be manipulated when adversarial text is embedded within visual content[1]. 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[1]. This means that text embedded in images receives identical semantic processing to direct textual prompts, bypassing the intended separation between visual description and textual instruction.

Execution Pathway Triggers

Conditions for Instructional Execution

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[1]. The boundary between data and instructions becomes blurred in multimodal systems, enabling adversaries to craft images that appear descriptive but contain embedded commands[1]. Task-guided object selection mechanisms demonstrate how semantic routing can be redirected toward specific objectives embedded within images[1].

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[1]. 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[1]. This results in semantic shift for input images compared to text in hidden states, misleading the safety mechanisms designed for textual content[1].

Attention Mechanisms and Context Priming

Self-Attention in Vision Processing

Research reveals that self-attention modules in vision transformers perform perceptual organization based on feature similarity rather than true attention mechanisms[9]. The attention formulation in these models computationally performs a special class of relaxation labeling with similarity grouping effects[9]. This creates vulnerabilities where adversarial content can exploit the grouping mechanisms to redirect attention toward malicious instructions embedded within visual elements.

Attention Weight Manipulation

Vision transformers utilize learnable positional encodings that capture spatial relationships between patches, providing access to dimensional knowledge of the image[7]. Adversaries can exploit these positional encodings by strategically placing malicious content in image regions that receive higher attention weights during processing. The multi-head attention mechanism processes different aspects of the visual input simultaneously, creating multiple pathways for adversarial exploitation[10].

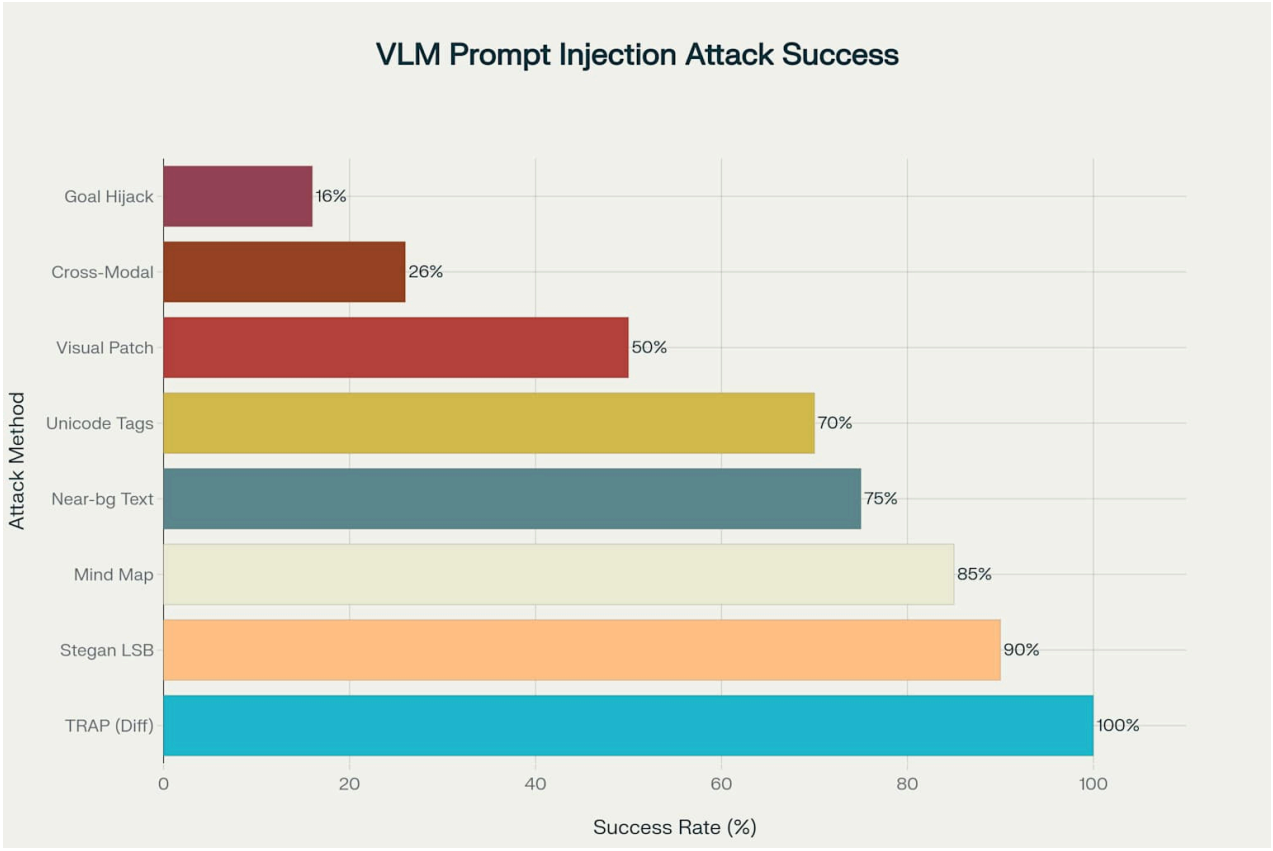
Visual Text Processing and Language Model Integration

OCR Capabilities and Vulnerabilities

GPT-4o demonstrates superior OCR capabilities that make it particularly vulnerable to near-background color text injection, allowing attackers to embed invisible instructions that models can read but humans cannot perceive[1] [11]. The OCR integration involves sophisticated deep learning methods tailored for character and word recognition across diverse fonts and backgrounds, but this capability becomes a vulnerability when exploited by adversarial actors[4].

Steganographic Attack Methods

Least Significant Bit (LSB) steganography enables concealment of malicious instructions within images that appear benign to human observers but are interpreted by VLMs[1]. 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[1]. 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[1] [12].



VLM Attack Success Rates: Comparative Analysis of Prompt Injection Techniques

Token Stream Transformation and Safety Filtering

Safety Head Analysis

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[1]. Ablating safety heads leads to higher attack success rates while maintaining model utility, indicating that current safety mechanisms are not robust to targeted attacks[1]. The integration of additional modalities increases susceptibility to safety risks compared to language-only counterparts[1].

Moderation Pipeline Weaknesses

Safety alignment degradation occurs when integrating vision modules compared to LLM backbones, creating representation gaps that emerge when introducing vision modality[1]. 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 that is not typically implemented in production systems[1].

Adversarial Exploitation Techniques

Visual Prompt Injection Methods

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[1]. The technique achieves a 15.8% attack success rate on GPT-4V, representing a significant security risk for deployed systems[1]. 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[1].

Advanced Attack Implementations

Recent research demonstrates that adversarial examples crafted with local-aggregated perturbations focused on crucial regions exhibit surprisingly good transferability to commercial LVLMs, including GPT-4.5, GPT-4o, Gemini-2.0-flash, Claude-3.5-sonnet, and reasoning models like o1[2]. These approaches achieve success rates exceeding 90% on GPT-4.5, 4o, and o1, significantly outperforming all prior state-of-the-art attack methods[2].

Real-World Exploitation Examples

Simon Willison's demonstration of exfiltration attacks using markdown images represents one of the most concerning real-world examples of visual prompt injection[1] [13]. The attack assembles encoded versions of private conversations and outputs markdown images containing URLs to attacker-controlled servers, successfully exfiltrating sensitive data[1]. Johann Rehberger's proof-of-concept demonstrates how speech bubbles in cartoon images can contain malicious code that sends ChatGPT conversations to external servers[1].

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[1]. The framework combines negative prompt-based degradation with positive semantic optimization, producing visually natural images that induce consistent selection biases in agentic AI systems[1].

Conclusion

The comprehensive technical analysis reveals that current vision-language models face fundamental architectural vulnerabilities that enable systematic adversarial exploitation through the image-to-command processing pipeline. The 15.8% to 100% attack success rates documented across different techniques demonstrate that malicious actors have multiple viable pathways to exploit VLM systems through manipulation of patch embedding, token compression, semantic routing, and attention mechanisms[1] [2]. The failure to properly transfer text-based safety mechanisms to visual modalities creates systematic vulnerabilities that require architectural solutions rather than post-hoc defenses[1].

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 that exploit the fundamental processing pipeline from visual input to command execution[1]. Future research must focus on developing inherently robust architectures that can distinguish between legitimate visual content and embedded malicious instructions without compromising model utility.

⁂

1. Vision-Language-Model-Security_-Methodology-Guide.pdf
2. https://arxiv.org/abs/2503.10635
3. https://arxiv.org/abs/2410.11190
4. https://www.leewayhertz.com/gpt-4-vision/
5. https://www.oranlooney.com/post/gpt-cnn/
6. https://app.readytensor.ai/publications/building-clip-from-scratch-a-tutorial-on-multimodal-learning-57Nhu0gMyonV
7. https://github.com/PrateekJannu/Vision-GPT
8. https://openaccess.thecvf.com/content/CVPR2022/papers/Wang_Multimodal_Token_Fusion_for_Vision_Transformers_CVPR_2022_paper.pdf
9. https://www.frontiersin.org/journals/computer-science/articles/10.3389/fcomp.2023.1178450/full
10. https://ojs.aaai.org/index.php/AAAI/article/view/28583
11. https://blog.roboflow.com/gpt-4-vision-prompt-injection/
12. https://www.keysight.com/blogs/en/tech/nwvs/2025/05/16/invisible-prompt-injection-attack
13. https://www.lakera.ai/blog/visual-prompt-injections