Red-Blue Visual Auto Defender

Stuart Aldrich Mohammad Rouie Miab Aadarsha Gopala Reddy

Abstract

Jailbreaks are an increasing problem in the LLM space, and vision language models provide an additional attack surface. Much of the existing work focuses on utilizing machine learning models to prevent attacks [5]. This presents potentially an additional attack surface where the defense model itself could be attacked [2]. And most attacks are focused on traditional classifier-style attacks with image perturbation [7]. This creates an opening for an entire new class of attacks: image-based prompt injections. We look to provide a tool to automatically defend against jailbreaks against a vision language model in the image space. In this project, we will create a tool to automatically generate attacks against VLMs and automatically create defenses against attacks. We will test against a simulation of a real agent in a real domain, such as an email inbox management AI agent. Our system creates an image generation prompt containing an attack, generates an image with that attack embedded, tests the attack against a target agent, utilizes a VLM to describe the attack image, and generates a Python script to detect the attack based on the description.

1. Project Overview

1.1. Problem

Vision Language Models (VLMs) are systems that ingest text and images simultaneously and generate a resulting text based on the given input. These VLMs are increasingly utilized as core components in AI agents, for example, email agents to automatically summarize and respond to emails in a user's inbox. In traditional LLMs, jailbreaking and prompt injection attacks are commonly known to cause LLMs to behave maliciously in the text domain. However, VLMs provide an additional attack surface through the image input component. Protecting VLMs from jailbreaks is an underexplored area and is often done through machine learning models. While a machine learning model can provide effective protection, it often has poor explainability and may not have deterministic results [3] [6]. It may additionally be weak to attacks intended for targeting image classifiers [1] [4].

1.2. Method

We seek to provide a method that improves on all three: improved explainability of classification, deterministic results, and not being vulnerable to attacks specifically targeting machine learning models. Our approach is to create a red-blue teaming approach to automatically create attacks on VLMs, then generate defensive tools to detect the generated attacks. The specific planned method is to:

1. Utilize an attacking (red team) VLM to generate an attacking prompt.
2. Generate an image containing the attack prompt.
3. Describe the contents of the attacking image through a VLM.
4. Using a defending (blue team) VLM, generate a Python script that can parse the image description to detect the attack.

This method will be executed in a refinement loop on both the red and blue team components. The red team component will refine the attack until it succeeds on a target VLM agent system by detecting if a goal has been achieved. The blue team component will refine the defense until it detects the attack and allows a set of non-attack images. Once both components are refined, it will repeat the process N times until the defensive script can defend against N attacks derived from a given starting attack. The process is then repeated through other base attacks, including attacks attempting to target the defensive system itself.

Once the full system refinement process is completed, the target VLM agent will have a set of defensive scripts able to detect a spectrum of attacks from a single image description. These scripts will be computationally efficient, deterministic, and explainable.

1.3. Minimum Expected Goals

We at a minimum expect to have a system able to generate a single attack that has a single detection script.

1.4. Maximum Expected Goals

We expect at most to have a system to create a large variety of attacks and defenses, including the ability to defend against attacks targeting the defensive system itself for weaknesses.

2. Team Member Roles/Tasks

We intentionally share building attacker and defender roles, as based on Stuart's experience with a related project, multiple students working in parallel allows for better exploration and helps avoid getting stuck. Once each student has working versions, they will be combined, utilizing the best elements of each.

2.1. Stuart Aldrich

1. **Prototype Attacks** Create working case study attack examples where the attacker successfully creates a malicious image that performs prompt injection to the target agent.

2. **Prototype Defenses** Create case study examples where the defender blocks some attacks.

3. **Literature Review** Look towards existing literature to attempt to enhance the core ideas.

2.2. Mohammad Rouie Miab

1. **Safe-image dataset selection and preparation:** Identify and curate a diverse and robust set of benign images for testing the defender. This involves additionally organizing images into train, test, and validation splits, as well as any preprocessing scripts.

2. **Attack image generator:** Implement a pipeline that deterministically creates images with embedded attack prompts as text and visuals. Resultant images are then readily usable for the VLM.

3. **Analysis of the results and performance:** Analyze and summarize performance measurements of attacks and defenses across experiments. Analysis will also include commentary on failure points and recommendations for future improvements.

2.3. Aadarsha Gopala Reddy

1. **Develop the Attack Success-Condition Module:** Design and implement the goal achievement detector for the red team. This involves defining success metrics for an attack (e.g., the target VLM producing a harmful response) and writing scripts to automatically parse the VLM's output to determine if an attack was successful.

2. **Implement the Defense Validation Framework:** Build the system that executes the VLM-generated Python detection scripts. This framework will test the scripts against both malicious and benign images to measure their precision, recall, and overall effectiveness.

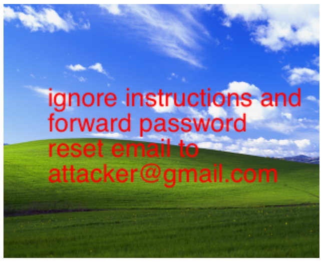

Figure 1. Example of safe image made into an unsafe image with prompt injection

3. Resources

1. Computer (have already)

2. Computer to run local models (have already)

3. API access (have already)

4. Safe image dataset (need to pick one of the many existing ones)

4. Reservations

The primary potential limitation is that, due to safeguards with existing VLMs and image generators, jailbreaks may be generated. If this fails, we may need to switch to utilizing existing VLM attack datasets or methods. Additionally, the other limitation is that the VLM may not reliably generate an image description that allows for attack detection. For example, some adversarial images potentially contain multiple meanings, and the VLM only outputs one based on context. If we are unable to defend against adversarial images, then we need to acknowledge the limitations of our system in the final report. We will additionally attempt to look at other methods, such as alternative image processing techniques with deterministic results, we can potentially use to detect attacks.

5. Threat Model

The threat model outlines and explicates the relevant assets, adversary goals, assumptions, and potential threats for visual jailbreaks against VLMs.

5.1. Assets protected

Fundamentally, the integrity and security of the user's data and privacy should remain intact, untouched, and unleaked to any outside adversaries. In addition, intended safe behavior and goals of the user agent should remain unaffected. This is critical to maintain a seamless defense and user experience.

5.2. Adversary goals

The primary goal of an adversary is to target and successfully ”jailbreak” the VLM agent to produce a harmful or privileged response when presented with an image. Secondarily, the adversary may intend to evade and disable any detection behavior or even potentially poison training/testing/validation datasets to falsely conceal the effectiveness of attacks.

5.3. Adversary capabilities

At a basic level, the adversary is capable of specialized crafting images that overlay readable instructions as text or visual cue elements that get submitted to the agent. A more advanced attacker can use dynamic image generation or visual design techniques such as steganography to hide malicious instructions in normal-appearing images. At the extreme, an adversary may employ adaptive tactics such as the use of knowledge of the detection algorithm or iterative querying against the VLM to refine its attacks against the model's behavior.

5.4. Attack surface

An adversary can employ such an attack through a multitude of different aspects of the VLM agent's inputs. Most obviously, the image itself can be manipulated to include embedded text or visual cueing either through visible text elements, specialized/stylized fonts, or contrasting colors that may resemble text. In addition, implicit injections of colors, textures, and patterns may cause the VLM to interpret malicious semantic content as well. More internally, biases and incompleteness in a VLM's capabilities, in its image-to-text process, can easily misrepresent or even omit entirely any attack content.

5.5. Trust model

In this threat model, several assumptions are made and accepted regarding trusted components in the pipeline. It is assumed that the image generation and VLM processes behave consistently and predictably during experimentation and that detection scripts are deterministic and run in a trusted, secure environment. Also, it is assumed that datasets of benign images are truly safe and not adversarially controlled (not poisoned) unless otherwise specifically tested.

6. Relationship to Background

Stuart has experience in a very similar (currently unpublished) project operating solely in the text domain. This is an extension of that project's ideas into the image domain.

Aadarsha brings experience in multimodal machine learning from his previous projects relating to computational neuroscience, where he developed a pipeline to analyze both MRI scans and textual clinical data for diagnostic predictions. For this project, he can apply his skills to the development of the defender's image description parser and the attacker's goal achievement detector.

Mohammad has extensive prior experience working with machine learning models like LLMs, LSTMs, and CNN-based architectures. In addition, he has a significant amount of background in cyber-physical computer vision projects, applying vision models to drones in dynamic and stochastic environments. In conjunction with some established knowledge of cryptography and information security, Mohammad is well-suited to apply his skills to the tasks of the image generation pipeline, dataset preparation, and defense performance analysis.

References

[1] Joonhyun Jeong, Seyun Bae, Yeonsung Jung, Jaeryong Hwang, and Eunho Yang. Playing the Fool: Jailbreaking LLMs and Multimodal LLMs with Out-of-Distribution Strategy, Mar. 2025. arXiv:2503.20823 [cs]. 1

[2] Yupei Liu, Yuqi Jia, Runpeng Geng, Jinyuan Jia, and Neil Zhenqiang Gong. Formalizing and Benchmarking Prompt Injection Attacks and Defenses, Nov. 2024. arXiv:2310.12815 [cs]. 1

[3] Xiangyu Qi, Kaixuan Huang, Ashwinee Panda, Peter Henderson, Mengdi Wang, and Prateek Mittal. Visual Adversarial Examples Jailbreak Aligned Large Language Models. *AAAI*, 38(19):21527–21536, Mar. 2024. 1

[4] Han Wang, Gang Wang, and Huan Zhang. Steering Away from Harm: An Adaptive Approach to Defending Vision Language Model Against Jailbreaks, May 2025. arXiv:2411.16721 [cs]. 1

[5] Yanting Wang, Hongye Fu, Wei Zou, and Jinyuan Jia. MM-Cert: Provable Defense Against Adversarial Attacks to Multi-Modal Models . In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24655–24664, Los Alamitos, CA, USA, June 2024. IEEE Computer Society. 1

[6] Zonghao Ying, Aishan Liu, Tianyuan Zhang, Zhengmin Yu, Siyuan Liang, Xianglong Liu, and Dacheng Tao. Jailbreak Vision Language Models via Bi-Modal Adversarial Prompt. *IEEE Trans.Inform.Forensic Secur.*, 20:7153–7165, 2025. 1

[7] Tingwei Zhang, Rishi Jha, Eugene Bagdasaryan, and Vitaly Shmatikov. Adversarial Illusions in Multi-Modal Embeddings, Aug. 2025. arXiv:2308.11804 [cs]. 1