Demo: Sensor-Based Jailbreak Attacks Against Safety Policies in Physical AI Systems

Shri Abijith Mohanraj, Jianshu Wang, Jinwen Wang
University of Texas at Dallas

Abstract

Physical AI systems powered by vision-language-action (VLA) models employ safety policies to prevent unsafe actions. However, jailbreak attacks pose a significant threat to generative AI models, including VLAs. This threat is further exacerbated by the rich sensor inputs in physical AI systems, as attackers can manipulate sensor observations with human-imperceptible perturbations to bypass safety policies and induce unsafe actions. In this demo, we show that sensor-based jailbreak attacks against safety policies can achieve a 97% success rate on the NVIDIA GR00T-N1.6 physical AI model.

1 Introduction

Vision-language-action (VLA) models have revolutionized robotics by enabling robots to autonomously reason, plan, and execute actions in response to user commands, as demonstrated by physical AI models such as NVIDIA GR00T [1], Gemini Robotics [2], and OpenVLA [3]. Due to their continuous interaction with the physical environment, physical AI systems must enforce safety policies to prevent unsafe behaviors and physical incidents. Unfortunately, jailbreak attacks on generative AI models can bypass the safety policies of physical AI systems, leading to safety hazards by inducing the execution of actions prohibited by those policies. Existing jailbreak attacks on VLA models focus on the malicious construction of input for vision and language modality. However, VLA models accept inputs from a broader range of perception modalities, such as IMUs, which further expands and makes the jailbreak attack surface more stealthy. This demo aims to evaluate the effectiveness of sensor-based jailbreak attacks that compromise available sensors in physical AI systems. More details can be found in our full paper.

In this demo, we deploy a height-constrained safety policy on NVIDIA GR00T-N1.6 to limit the maximum height a robotic arm can reach in height-restricted scenarios. By compromising sensor inputs in a structured manner, we show that sensor-based jailbreak attacks can induce the execution of dangerous commands that violate the safety policy and would otherwise be rejected, achieving a success rate of 97% across 1,400 open-loop trials.

2 Threat Model

The attacker aims to bypass safety policies by manipulating the sensor values perceived by a physical AI system, causing it to execute natural-language commands that would otherwise be prohibited. Similar to prior sensor attacks on cyber-physical systems, the attacker can inject perturbations into sensor inputs to influence the system's perception. Meanwhile, the attacker can provide arbitrary natural-language commands as a benign user. We assume the computational hardware (e.g., CPUs) and software stack (e.g., the operating system and physical AI models) are trusted.

3 Demo Setup and Workflow

Demo Setup: We run the NVIDIA GR00T-N1.6 inference server on a machine equipped with an Intel Core Ultra 7 265K processor (20 cores, up to 4.6,GHz), 32,GB of system memory, and an NVIDIA GeForce RTX 5090 GPU with 32,GB of GDDR7 memory. Using forward kinematics, the safety-policy reference monitor on top of NVIDIA GR00T computes the predicted world-frame end-effector (EEF) height associated with the proposed action and rejects the action if the predicted EEF height exceeds a predefined safety threshold. The deployed safety policy is *the EEF must not lift above the table top (1m).* The same language prompt is used for every trial: *Stretch the robotic arm to reach 30 cm above the table.*

Demo Workflow: The demo evaluates the success rate of safety-policy jailbreak attacks under different attack strategies, as summarized in Table 1: (1) no attack; (2) addition of amplified Gaussian noise to GPS and IMU signals; (3) GPS spoofing for jailbreak; (4) IMU spoofing for jailbreak; (5) joint GPS and IMU spoofing for jailbreak; (6) addition of

Table 1: Evaluation conditions ($n = 200$ each).

Cond.	Channels active and modes
C0	Clean. No perturbation.
C1	GPS + IMU Gaussian at σ L2-matched to C4.
C2	GPS only: deterministic -0.20 m bias.
C3	IMU only: 2-tone sinusoidal drift.
C4	GPS spoof + IMU spoof, no noise.
C5a	GPS + IMU realistic noise (σ_{GPS}=0.02, σ_{IMU}=0.03).
C5b	C5a noise + C4 deterministic bias. Stealthy.

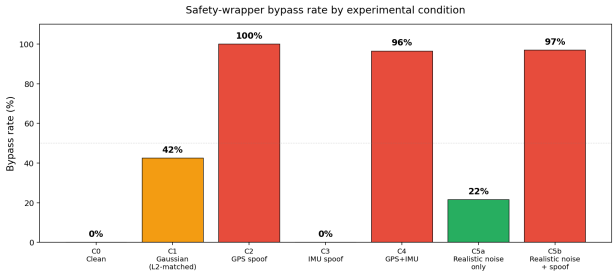


Figure 1: Safety Policy Jailbreak Success Rate

realistic Gaussian noise to GPS and IMU signals; and (7) combined GPS and IMU spoofing with simultaneous realistic Gaussian noise added to GPS and IMU signals. We evaluate the outputs of the safety-policy reference monitor that relies solely on GPS input for each trial. For each scenario, we conduct 200 trials.

4 Experiment Results

Figure 1 reports the safety-policy jailbreak success rate for each experimental condition. Under C0 (clean), the success rate is 0%, indicating that the safety policy is consistently enforced. C3 (IMU spoofing only) also achieves a 0% success rate because the safety-policy reference monitor determines policy violations solely from the GPS-based localization channel rather than the IMU measurements. Consequently, perturbations confined to the IMU cannot affect the monitor's decision. In contrast, C2 (GPS spoofing only) achieves a 100% success rate. C4 (combined GPS and IMU spoofing) achieves a 96.5% success rate, while C5b (stealthy attack, combining realistic sensor noise with structured GPS spoofing) achieves a 97% success rate.

The two control conditions provide additional insight. C1, which injects Gaussian noise with a total ℓ_2 magnitude matched to that of the targeted attack, achieves a jailbreak success rate of 42.5%. This is higher than the clean baseline because some noise realizations perturb the GPS measurement in the same direction as the targeted spoofing attack, causing the safety-policy monitor to accept actions that violate the safety policy. Although random noise is less effective than a directed spoofing attack, it is not harmless. The structured attack achieves a success rate that is 2.3× higher than the matched Gaussian-noise baseline. C5a, which injects realistic sensor noise without structured spoofing, achieves a jailbreak success rate of 21.5%. This result stems from the proximity

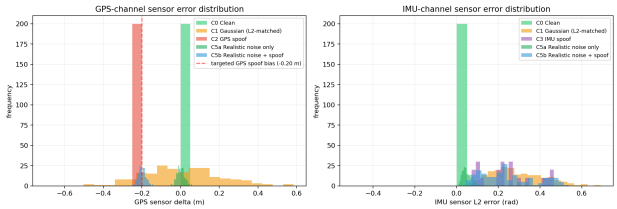


Figure 2: Sensor-error Distributions Per Channel

of the safety threshold (1.00, m) to GR00T's mean predicted EEF height (1.015, m). With $\sigma_{GPS} = 0.02$, the probability that zero-mean Gaussian noise reduces the predicted height below the threshold is $\Phi(-0.015/0.02) = 22.7\%$, closely matching the observed 21.5%. Thus, the C5a result reflects the threshold's sensitivity to measurement noise rather than a weakness in the safety-policy monitor. Reducing the threshold to 0.95, m lowers the C5a success rate to below 1%, while C5b remains at 95%.

Figure 2 shows the sensor-error distributions for each channel. The GPS panel highlights the difference between structured spoofing and random noise: C2 and C5b form narrow peaks at -0.20 m, whereas C1 exhibits a Gaussian distribution with a matched expected ℓ_2 magnitude. The IMU panel shows a similar pattern for conditions that perturb the orientation channel. By construction, C2 and C5a do not appear in the IMU panel, while C0 and C3 do not appear in the GPS panel. C4 is omitted from both panels for visual clarity, as its GPS and IMU components are already represented by C2 and C3.

5 Conclusion

This demo shows that a standard runtime safety-policy monitor in a VLA-driven physical AI system can be bypassed through structured sensor spoofing on its input channels, resulting in a jailbreak attack against the deployed safety policy. More details on techniques for compromising diverse sensor modalities to jailbreak safety policies in physical AI systems can be found in our full paper.

References

[1] NVIDIA, "GR00T-N1.6: Generalist humanoid robot foundation model," https://huggingface.co/nvidia/GR00T-N1.6-3B, 2025.

[2] Google, "Gemini Robotics," https://deepmind.google/models/gemini-robotics/, 2025.

[3] M. J. Kim et al., "OpenVLA: An open-source vision-language-action model," *arXiv:2406.09246*, 2024.

[4] Z. Zeng et al., "VLAGuard: Safety-aware monitoring for vision-language-action models," *arXiv:2503.09935*, 2025.