2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Exploiting Vulnerabilities: Universal Adversarial Attacks on Vision-Language-Action Models in Robotics

Songhua Yang[1,2], Ziyu Liu[1], Yuanwei Liu[1], Xuetao Li[1,2], Xuanye Fei[3], He Huang[3], Zheng Wang[1], Miao Li[1,2,*]

***Abstract*—Recently, Vision-Language-Action (VLA) models have revolutionized robotic manipulation by seamlessly integrating visual perception, language understanding, and action generation in an end-to-end learning framework. However, since these models are designed to interact directly with the physical world and humans, their security is critical, and even small vulnerabilities can lead to catastrophic failures. In this work, we propose the Universal Adversarial Object, a sphere with optimized surface texture that significantly degrades task success rates when placed within the robot's field of view. Specifically, our approach introduces a multi-level attack framework that jointly disrupts trajectory planning, task execution, and action control. We validate our method in both simulated and real-world robotic settings. Experimental results demonstrate that the adversarial object reduces the average task success rates by 31.2%-39.9% for two representative VLA models (Pi0 and RDT), with success rates dropping to near zero in complex scenarios.**

***Index Terms*—Vision-Language-Action models, adversarial attack, robotic security, universal adversarial object**

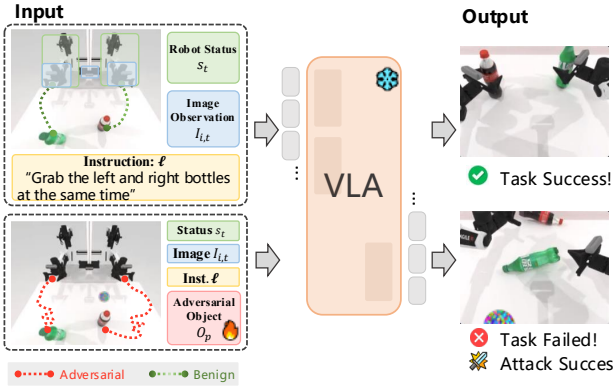

Fig. 1: An example of implementing an attack on VLA with universal adversarial objects. The normal inputs to the VLA include the robot status S_t, image observation $I_{i,t}$, and instruction l, which enable the robot to successfully perform the grasping task. However, by applying an adversarial attack, an adversarial object O_p is added to these normal inputs, causing the robot's grasping attempt to fail.

I. Introduction

Vision-Language-Action (VLA) models have emerged as a transformative paradigm in robotics, enabling robots to seamlessly translate visual observations and natural language instructions into physical actions through end-to-end learning frameworks [12], [13]. Unlike Large Language Models (LLMs) or Vision-Language Models (VLMs), which operate in the digital domain, VLA models directly control physical robots that manipulate objects and interact with humans [1], [2]. This unique capability significantly increases the security risks associated with VLA models. However, research on adversarial attacks and security evaluations for VLA is virtually non-existent, with almost all studies focused solely on performance improvements rather than addressing the critical security vulnerabilities these models face [14].

Adversarial attack research serves as a critical tool for evaluating model robustness and identifying security vulnerabilities before deployment [22]. Although physical adversarial attacks have successfully compromised traditional robotic systems - from attacking imitation learning policies [32] to deceiving reinforcement learning agents [33] - these methods were designed for relatively simple modular architectures. Previous work has shown that adversarial patches [23], [24] and objects [28] can fool perception systems in autonomous vehicles [29], yet these approaches cannot be directly transferred to VLA models. The fundamental difference lies in VLA's architectural complexity: Their multimodal fusion mechanisms, diffusion-based generation processes, and temporal decision-making create novel vulnerability patterns that require entirely new attack strategies to understand and ultimately defend against.

In this work, we present the first systematic analysis of mainstream VLA models' vulnerabilities to physical adversarial attacks, bridging the gap between theoretical security concerns and practical threats. We propose a Universal Adversarial Object (UAO)—a seemingly innocuous 3cm textured sphere—that, when strategically placed within the robot's workspace, severely disrupts VLA model decision-making across diverse tasks and viewing conditions. As illustrated in Figure 1, this simple object induces dramatic behavioral corruptions: robots exhibit severe trajectory deviations, erratic oscillations, and catastrophic task failures, with success rates plummeting by up to 70% in precision manipulation tasks. Our multi-level attack framework simultaneously targets trajectory planning, task execution, and motion control, exploiting the

[1]School of Computer Science, Wuhan University.
[2]School of Robotics, Wuhan University.
[3]School of Power and Mechanical Engineering, Wuhan University.
* Corresponding author: Miao Li (E-mail: limiao@whu.edu.cn).

hierarchical nature of VLA processing to maximize disruption while maintaining physical realizability and visual inconspicuousness.

Specifically, we design a multi-level attack framework that disrupts the entire perception-to-action decision chain of VLA models: trajectory planning, task execution, and action control. Given that VLA models are based on diffusion architectures [35] with inaccessible internal gradients, we adopt a black-box optimization strategy. To ensure the attack's sustained efficacy, we integrate temporal characteristics and the SPSA optimization method to enhance efficiency. Additionally, we employ data augmentation techniques to guarantee the robustness and physical feasibility of the adversarial objects under various environmental conditions.

We thoroughly evaluate our approach on two representative VLA models (Pi0 and RDT) across 13 robotic manipulation tasks with varying complexities. Experimental results show that, in simulation, the adversarial object reduces task success rates by 31.2%-39.9%, and in more complex environments, the success rate drops to nearly zero (RDT: 2.1%, Pi0: 1.9%). Moreover, we demonstrate UAO on a real dual-arm robotic platform, achieving a sim-to-real transfer rate of 81.4%-82.2%. Even when the UAO is visible in only a single camera view, it leads to a considerable decline, highlighting its viewpoint robustness.

The main contributions of this work include:

(1) Comprehensive revelation of VLA model adversarial vulnerability. We show through large-scale experiments that VLA models are highly vulnerable to adversarial objects, which can severely disrupt decision-making.

(2) Multi-level attack framework and UAO generation. We introduce a multi-level attack framework that generates robust adversarial objects, overcoming the limitations of traditional attack methods.

(3) Validation from simulation to real-world deployment. We validate the attack effectiveness in both simulated and real-world environments, demonstrating its practical threat to VLA model safety.

II. Related Works

Vision-Language-Action Models.

Vision-Language-Action (VLA) models represent a paradigm shift in robotic learning, unifying visual perception, language understanding, and action generation within end-to-end frameworks [1], [2]. Early VLA explorations primarily constructed hierarchical frameworks leveraging the perceptual and reasoning capabilities of Vision-Language Models (VLMs) [3]–[6]. Subsequent advances encoded actions as an additional modality, achieving superior generalization through joint VLA training and enabling end-to-end learning for complex manipulation tasks [7]–[11].

Recent VLA models have achieved significant breakthroughs in architectural design and training strategies [2]. In particular, Pi0 [12] introduces innovative policy learning methods that unify various manipulation tasks through large-scale demonstration data. RDT [13] incorporates diffusion models for action generation, producing more precise trajectories through iterative denoising. Although these models demonstrate impressive performance on multiple benchmarks [21], existing research focuses primarily on improving capabilities, overlooking systematic security analysis [14].

Adversarial Attacks in Robotic.

Adversarial attacks, which mislead deep learning models through carefully crafted input perturbations, have emerged as critical tools to evaluate and improve model robustness [22]. The progression from digital to physical-world attacks has exposed the pervasive vulnerability of deep models to adversarial perturbations. Physical adversarial attacks have proven particularly significant, and methods such as adversarial patches [23], [24] and adversarial objects [28] demonstrate the feasibility of deceiving vision models in real-world environments. These attacks pose serious security threats across domains including autonomous driving [29], facial recognition [26], and object detection [30].

Robotic systems present unique security challenges, requiring continuous decision-making and physical interaction in dynamic environments. Early adversarial research primarily targeted traditional small-scale robotic models, such as imitation learning [32] and reinforcement learning policies [33]. However, the emergence of VLA models marks a fundamental shift toward large-scale, multimodal, end-to-end architectures, introducing novel attack surfaces and security challenges. The complex multimodal interactions and diffusion-based architectures of VLA models render traditional attack methods ineffective. Although Wang et al. [25] pioneered the exploration of vulnerabilities in VLA, their analysis remained limited to specific models and tasks. In contrast, we propose the first universal physical adversarial attack framework that systematically compromises VLA models across different architectures and validates its effectiveness on real robotic platforms.

III. Methodology

A. Preliminary

Currently, VLA models leverage end-to-end learning, directly transforming multimodal inputs into robot control commands [1], [2]. Specifically, a VLA model $\mathcal{M}$ receives several types of inputs at each decision timestep t. These include task instructions $\mathcal{L}$ in natural language describing the objectives; visual observations $\{\mathbf{I}_{i,t}\}_{i=1}^{N_{cam}}$ from multiple viewpoints; and the robot's proprioceptive state $\mathbf{s}_t \in \mathcal{S}$, encompassing joint angles and end-effector positions.

The output of the VLA model comprises two components: a predicted action sequence $\mathbf{a}_{t:t+k} = \{\mathbf{a}_t, \mathbf{a}_{t+1}, ..., \mathbf{a}_{t+k}\}$ where k denotes the prediction horizon and a task completion indicator $r \in \{0, 1\}$. In this work, we focus on models operating on a 7-DoF dual-arm Aloha platform [15]. At each timestep, actions are represented as:

$$\mathbf{a}_t = [\Delta P_x, \Delta P_y, \Delta P_z, \Delta R_x, \Delta R_y, \Delta R_z, g]$$

where $\Delta P_x, \Delta P_y, \Delta P_z$ and $\Delta R_x, \Delta R_y, \Delta R_z$ denote relative position and rotation changes along the x, y, and z axes respectively, and g is a binary gripper state.

B. Problem Definition

We formalize the adversarial attack as finding a universal visual perturbation that induces task failure without modifying model parameters $\mathcal{M}$, language instructions $\mathcal{L}$, or robot states $\mathcal{S}$. Specifically, we design an adversarial patch $\mathbf{P} \in \mathbb{R}^{H\times W\times 3}$, where H and W define spatial resolution. This patch is projected onto a sphere of radius r via a spherical mapping function $\phi : \mathbb{R}^{H\times W\times 3} \to \mathbb{S}^2(r)$, forming a universal adversarial object $\mathbf{O_p}$. When placed in position $\mathbf{c} \in \mathbb{R}^3$ within the robot workspace, this object appears in camera views $\mathbf{I}_{i,t}$ and disrupts task execution through visual manipulation.

We employ black-box optimization to generate this universal patch, accessing only model outputs without internal parameters or gradients. Our objective is to find an optimal patch $\mathbf{P}^*$ such that the resulting trajectory $\tau^{adv} = \{\mathbf{a}_1^{adv}, ..., \mathbf{a}_T^{adv}\}$ significantly deviates from the clean trajectory τ^{clean} and induces task failure ($r^{adv} = 0$). To achieve this, we design a multi-objective loss function incorporating trajectory deviation, task failure, action perturbation, and visual naturalness.

C. Multi-level Attack Framework for VLA

The generation of VLA actions involves multiple processing levels, from pixel-level visual perception to task-level goal understanding and trajectory-level action planning [10], [16]. Based on this observation, we design a hierarchical attack framework that jointly disrupts these levels for comprehensive adversarial impact.

1) Trajectory Deviation Attack: Robotic manipulation tasks exhibit strong temporal dependencies, with VLA models demonstrating distinct phases during long-horizon execution [17], [18]: coarse positioning in initial stages, primary manipulation in intermediate phases, and precise control for task completion. We design a time-weighted trajectory deviation loss:

$$\mathcal{L}_{traj}(\mathbf{P}) = \frac{1}{T}\sum_{t=1}^{T} w_t \cdot \|\tau_{clean}^t - \tau_{adv}^t\|_2$$

where τ_{clean}^t and τ_{adv}^t denote end-effector positions at time t under clean and adversarial conditions, respectively, with $\|\cdot\|_2$ computing Euclidean distance. The temporal weight function $w_t = \exp(\alpha \cdot t/T)$ assigns exponentially increasing importance to later timesteps, concentrating attack energy on critical decision moments. Parameter α controls the growth rate, with larger values emphasizing late-stage precision control.

2) Task Failure Attack: The success rate is the primary metric for robotic manipulation, yet its discrete binary nature prevents direct gradient optimization. Moreover, diffusion-based VLA models preclude direct access to internal reward signals. To address this challenge, we propose a contrastive-inspired task failure loss [19]:

$$\mathcal{L}_{task}(\mathbf{P}) = \sum_{i=1}^{N} r_i^{clean} \cdot (1 - r_i^{adv}) + \beta \cdot \max(0, r_i^{adv} - r_i^{clean})$$

The first term targets originally successful tasks, generating negative loss when they fail under adversarial conditions. The second term penalizes unintended improvements where failed tasks become successful. The asymmetric penalty coefficient $\beta > 1$ reflects the attack's directional nature.

3) Action Perturbation Attack: Through imitation learning, VLA models acquire smooth action primitives by mimicking expert demonstrations [10], [13]. We exploit this by inducing high-frequency directional changes that disrupt the diffusion denoising process. Our action perturbation loss is:

$$\mathcal{L}_{shake}(\mathbf{P}) = -\frac{1}{T-2}\sum_{t=2}^{T-1} \|\mathbf{v}_{t+1} - \mathbf{v}_t\|_2$$

where $\mathbf{v}_t = (\tau_t^{adv} - \tau_{t-1}^{adv})/\|\tau_t^{adv} - \tau_{t-1}^{adv}\|_2$ represents the normalized direction of action. This loss maximizes the directional differences between adjacent timesteps, generating oscillatory patterns. Given velocity constraints in robotic systems, we perturb direction rather than magnitude. Geometrically, $\|\mathbf{v}_{t+1} - \mathbf{v}_t\|_2 \in [0, 2]$, where 0 indicates a linear action and 2 represents a 180-degree reversal.

4) Regularization Constraints: To ensure physical realizability and inconspicuousness, we introduce regularization constraints [27], [31]:

$$\mathcal{L}_{reg}(\mathbf{P}) = \|\mathbf{P}\|_\infty + \gamma \cdot TV(\mathbf{P})$$

The infinity norm $\|\mathbf{P}\|_\infty$ limits the maximum pixel intensity within the standard RGB range. Regularization of total variation $TV(\mathbf{P}) = \sum_{i,j}\sqrt{(P_{i+1,j} - P_{i,j})^2 + (P_{i,j+1} - P_{i,j})^2}$ promotes spatial smoothness by penalizing adjacent pixel differences. This smoothness yields multiple benefits: natural texture appearance that reduces detection risk, robustness to printing errors and lighting variations, and prevention of high-frequency local optima.

5) Overall Optimization Objective: In summary, our total loss function is defined as follows:

$$\begin{aligned}\mathcal{L}_{total}(\mathbf{P}) = \lambda_1\mathcal{L}_{traj}(\mathbf{P}) + \lambda_2\mathcal{L}_{task}(\mathbf{P}) \\ +\lambda_3\mathcal{L}_{shake}(\mathbf{P}) + \lambda_4\mathcal{L}_{reg}(\mathbf{P})\end{aligned}$$

The weight coefficients reflect relative importance, with empirically determined values: $\lambda_1 = 0.30$, $\lambda_2 = 0.45$, $\lambda_3 = 0.20$, $\lambda_4 = 0.05$. Task failure receives the highest weight as the primary objective, whereas trajectory deviation and action perturbation serve as auxiliary targets. Regularization maintains minimal weight for necessary physical constraints.

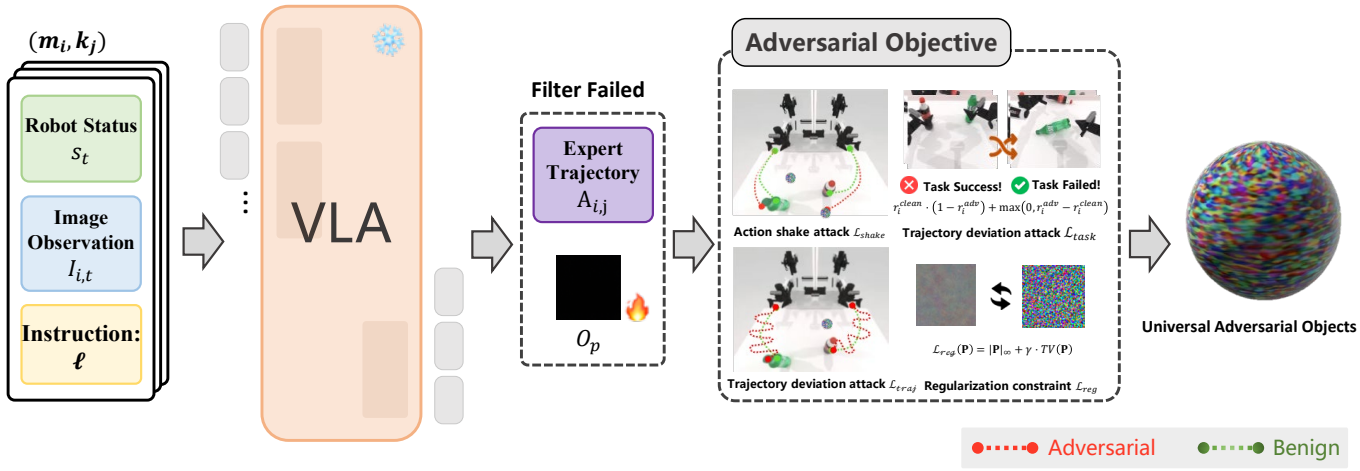

Fig. 2: The framework for generating universal adversarial objects via multi-level attack optimization. Starting from expert demonstrations, we filter successful trajectories and optimize adversarial textures using four complementary losses that target different aspects of robot behavior—from high-level task failure to low-level motion perturbations. The resulting texture pattern is mapped onto a sphere to create a physically universal adversarial object.

D. Optimization Strategy

Considering the training cost, we develop an efficient black-box optimization method achieving effective attacks within limited query budgets.

1) Gradient Estimation: Given the inaccessibility of internal gradients in VLA models, we adopt Simultaneous Perturbation Stochastic Approximation (SPSA) [20], which efficiently estimates high-dimensional gradients through merely two function evaluations:

$$\nabla_{\mathbf{P}}\mathcal{L} \approx \frac{\mathcal{L}(\mathbf{P}+c\Delta)-\mathcal{L}(\mathbf{P}-c\Delta)}{2c}\cdot\Delta^{-1}$$

where the perturbation vector $\Delta \in \{-1,+1\}^{H\times W\times 3}$ has elements sampled independently of the Bernoulli distribution. Step size c follows an adaptive schedule: larger values for initial exploration, gradually decreasing for fine-tuning.

2) Data Augmentation: During manipulation, camera viewpoints continuously change as the arm moves, altering the patch's appearance and potentially occluding regions. To ensure attack effectiveness under dynamic conditions, we apply augmentations during optimization:

$$\mathbf{P}_{aug} = \mathcal{T}_{rot}(\theta)\circ\mathcal{T}_{scale}(s)\circ\mathcal{T}_{shift}(\delta)(\mathbf{P})$$

where the rotation angle $\theta \sim \mathcal{U}(0, 2\pi)$ ensures orientation invariance, the scale factor $s \sim \mathcal{U}(0.8, 1.2)$ handles distance variations, and the cyclic shift δ simulates the rotation of the sphere. Color enhancement adjusts contrast $\alpha \sim \mathcal{U}(0.8, 1.2)$ and brightness $\beta \sim \mathcal{U}(-0.1, 0.1)$ to provide light stability.

3) Optimization Process: We employ alternating optimization across models and tasks to ensure broad attack effectiveness. For each model-task pair $(\mathbf{m}, \mathbf{k})$, we evaluate clean performance as baseline, then use SPSA to estimate gradients and update the patch—requiring only two evaluations per iteration. Tasks that initially fail are filtered out. The optimization is implemented in RoboTwin simulator [21] with PyOpenGL[1] for sphere texture mapping. Algorithm 1 details the complete process, generating universal patches effective in multiple VLA models and tasks.

Algorithm 1 Universal Adversarial Object Optimization Process.

Input: Model set $\mathcal{M}$, task set $\mathcal{K}$, iterations N
Initialize: $\mathbf{P} \sim \mathcal{U}(-0.01, 0.01)^{H\times W\times 3}$
for $n = 1, 2, \cdots, N$ **do**
  for $(m, k) \in \mathcal{M} \times \mathcal{K}$ **do**
    $r^{clean}, \tau^{clean} \leftarrow$ Evaluate$(m, k, \emptyset)$
    if $r^{clean} = 0$ **then**
      continue
    end if
    $\mathbf{P}_{aug} \leftarrow \mathcal{T}(\mathbf{P})$ {Augmentation}
    $\Delta \sim \{-1, +1\}^{H\times W\times 3}$
    $\mathcal{L}^{\pm} \leftarrow \mathcal{L}_{total}$(Evaluate$(m, k, \mathbf{P}_{aug} \pm c\Delta)$)
    $\mathbf{P} \leftarrow \mathbf{P} - \eta \cdot \frac{\mathcal{L}^{+}-\mathcal{L}^{-}}{2c} \cdot \Delta^{-1}$ {SPSA update}
  end for
end for
Output: Optimized patch $\mathbf{P}^*$

IV. Experiment and Analysis

This section verifies the proposed UAO attack method through systematic experiments. Section IV-A first introduces the experimental setup, and Section IV-B presents the main attack results. Subsequently, Section IV-C analyzes the contribution of each loss component through ablation experiments, Section IV-D explores the impact of key hyperparameters via parameter analysis, Section IV-E tests the view robustness

[1]https://github.com/mcfletch/pyopengl

through multi-view evaluation, and finally, Section IV-F conducts real-world validation.

A. Experiment Setup

1) Target Models: We select Pi0 [12] and RDT [13] as the target VLA models. Pi0 uses a Transformer-based architecture, with the pre-trained PaLI-Gemma [34] backbone, and generates continuous action trajectories using Flow Matching. RDT, with a Diffusion Transformer architecture [35] and 1.2 billion parameters, is pre-trained on 46 multi-robot datasets (over 1 million trajectories) and fine-tuned with 6000 dual-arm operation data. Both models are extensively pre-trained and fine-tuned, showcasing excellent generalization in dual-arm manipulation tasks.

2) Tasks and Environment: We use the RoboTwin simulator as the training and evaluation platform, designed specifically for dual-arm coordination, and fully supports Pi0 and RDT models. The simulator includes a benchmark with expert trajectories. From 50 tasks, we select 13 representative ones with high success rates, ranging from basic grasping to assembly and dual-arm coordination tasks. Each task has an Easy and Hard mode—Easy uses a clean desktop, while Hard introduces random lighting, material, and irrelevant objects. The experiments use the standard Aloha dual-arm configuration, with 7 degrees of freedom per arm and three fixed cameras (one overhead, two at the end) for multi-view observation. The UAO $\mathbf{O}_p$ is generated by mapping optimized patch $\mathbf{P}$ onto the surface of a 3cm radius sphere, randomly placed in the workspace during each evaluation, ensuring visibility in at least two camera views.

3) Evaluation Metrics: We use two complementary metrics to evaluate the attack's effect. The first is task success rate, which measures the proportion of successfully completed tasks in both clean and adversarial environments, reflecting the overall impact of the attack. The second is Trajectory Discrepancy (TD), which quantifies deviations in the robot's action:

$$TD = \frac{1}{T}\sum_{t=1}^{T}\frac{\|\tau_{clean}^{t} - \tau_{adv}^{t}\|_2}{\|\tau_{clean}^{T} - \tau_{clean}^{0}\|_2}$$

where τ_{clean}^{t} and τ_{adv}^{t} represent the end-effector positions at time t in the clean and adversarial environments. The denominator normalizes by the total trajectory length, making deviations comparable between tasks. TD quantifies the impact of the perturbation even if the task is not completely failed. Each task is evaluated with 10 different random seeds, and the mean result is reported.

4) Implementation Details: The adversarial patch $\mathbf{P}$ is set at a resolution of 128×128 pixels and projected onto the surface of a 3 cm radius sphere using spherical mapping. The simulation and real-world objects are shown in Figure 3. The optimization process uses the SPSA algorithm with an initial perturbation step size of $c = 0.01$, decaying at a rate of 0.95, and a learning rate of $\eta = 0.001$. For each model-task pair, we train for 500 iterations, with optimization taking approximately one week on an NVIDIA A100 80G GPU. The optimized patch remains fixed during the testing phase.

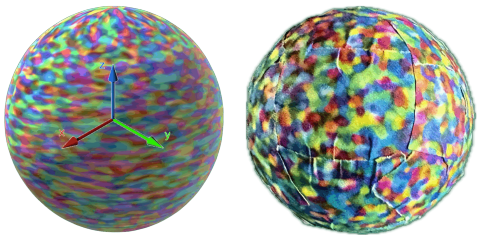

Fig. 3: The image of universal adversarial objects in simulator and real-world.

B. Main Results

Table I presents comprehensive evaluation results of our UAO against the RDT and Pi0 models in 13 representative manipulation tasks. The experiments reveal pervasive vulnerabilities in state-of-the-art VLA models, with a single 3cm textured sphere inducing average success rate drops of 31.2%-39.9% across all scenarios. Beyond binary task failure, the trajectory deviation metric quantifies systematic behavioral corruption, demonstrating 19%-27% positional errors throughout execution sequences even in partially successful attempts. This dual measurement reveals that adversarial perturbations not only prevent task completion but fundamentally distort the learned visuomotor mappings underlying robotic control.

The data exposes a critical vulnerability gradient correlated with environmental complexity and task precision requirements. In Hard mode, where visual distractors and lighting variations are introduced, both models catastrophically fail with near-zero success rates (RDT: 2.1%, Pi0: 1.9%), compared to 25.7%-36.4% residual performance in Easy mode. This dramatic amplification suggests that VLA models develop increased dependence on visual features when navigating complex scenes, inadvertently expanding their attack surface. Task-specific analysis reveals heightened susceptibility in precision-demanding operations: Grab Roller (70% drop for Pi0-Hard), Adjust Bottle (68% drop for RDT-Hard), and dual-arm coordination tasks consistently exhibit vulnerability rates exceeding 50%. Conversely, tasks with larger error tolerance such as Press Stapler and Put Object Cabinet demonstrate relative resilience, though still experiencing substantial degradation (16%-29% drops). Notably, Pi0 exhibits marginally higher baseline performance but suffers greater absolute degradation under attack, while RDT's diffusion-based architecture shows no inherent robustness advantage despite its iterative refinement mechanism. These findings underscore a fundamental security gap in current VLA architectures: the same multimodal integration that enables impressive generalization simultaneously creates exploitable dependencies that can be systematically compromised through carefully crafted visual perturbations.

TABLE I: Attack performance on different VLA models and tasks. SR: Success Rate, ΔSR: Success Rate Drop, TD: Trajectory Deviation. All metrics are averaged over 10 independent runs.

Task	RDT								Pi0							
	Easy				Hard				Easy				Hard			
	Clean SR	Adv. SR	ΔSR↓	TD	Clean SR	Adv. SR	ΔSR↓	TD	Clean SR	Adv. SR	ΔSR↓	TD	Clean SR	Adv. SR	ΔSR↓	TD
Adjust Bottle	83%	38%	45%	0.24	73%	5%	68%	0.29	88%	44%	44%	0.26	58%	2%	56%	0.31
Beat Block Hammer	75%	35%	40%	0.22	39%	3%	36%	0.27	45%	22%	23%	0.18	19%	0%	19%	0.25
Click Alarmclock	62%	30%	32%	0.19	10%	0%	10%	0.21	61%	28%	33%	0.20	13%	0%	13%	0.23
Grab Roller	72%	33%	39%	0.21	45%	4%	41%	0.26	94%	46%	48%	0.27	78%	8%	70%	0.32
Dump Bin Bigbin	66%	28%	38%	0.20	30%	2%	28%	0.25	81%	35%	46%	0.26	26%	1%	25%	0.28
Open Laptop	57%	25%	32%	0.19	34%	3%	31%	0.24	87%	40%	47%	0.27	44%	2%	42%	0.29
Press Stapler	43%	21%	22%	0.15	22%	1%	21%	0.22	60%	32%	28%	0.18	31%	2%	29%	0.26
Put Object Cabinet	31%	15%	16%	0.13	20%	1%	19%	0.21	70%	35%	35%	0.21	16%	0%	16%	0.24
Shake Bottle Horiz.	82%	37%	45%	0.25	53%	3%	50%	0.29	97%	48%	49%	0.28	53%	2%	51%	0.31
Shake Bottle	76%	36%	40%	0.22	43%	2%	41%	0.27	95%	45%	50%	0.28	62%	4%	58%	0.30
Pick Dual Bottles	40%	18%	22%	0.15	15%	0%	15%	0.20	59%	28%	31%	0.19	10%	0%	10%	0.18
Place Burger Fries	52%	23%	29%	0.17	25%	1%	24%	0.23	78%	36%	42%	0.24	6%	0%	6%	0.15
Open Microwave	35%	15%	20%	0.14	22%	1%	21%	0.22	82%	39%	43%	0.25	48%	3%	45%	0.28
Average	57.1%	25.7%	**32.3%**	**0.19**	33.5%	2.1%	**31.2%**	**0.25**	75.1%	36.4%	**39.9%**	**0.24**	35.4%	1.9%	**33.8%**	**0.27**

TABLE II: Ablation study on different attack components. w/o indicates that the component is removed.

Method	RDT (Easy)		RDT (Hard)		Pi0 (Easy)		Pi0 (Hard)	
	ΔSR↓	TD	ΔSR↓	TD	ΔSR↓	TD	ΔSR↓	TD
Full Attack	**32.3**%	0.19	**31.2**%	0.25	**39.9**%	0.24	**33.8**%	0.27
w/o $\mathcal{L}_{traj}$	22.9%	0.12	24.8%	0.16	26.8%	0.15	26.2%	0.18
w/o $\mathcal{L}_{task}$	18.6%	0.18	18.2%	0.23	22.4%	0.22	18.6%	0.25
w/o $\mathcal{L}_{shake}$	25.7%	0.14	27.9%	0.19	32.9%	0.18	30.1%	0.21
w/o $\mathcal{L}_{reg}$	32.3%	0.20	31.2%	0.26	39.2%	0.25	33.7%	0.28
Only $\mathcal{L}_{task}$	14.8%	0.08	14.9%	0.11	18.7%	0.10	15.3%	0.13
Only $\mathcal{L}_{traj}$	11.4%	0.16	11.1%	0.21	15.9%	0.19	11.7%	0.23
Only $\mathcal{L}_{shake}$	8.2%	0.11	7.7%	0.15	11.6%	0.14	8.1%	0.17

TABLE III: Attack performance under different camera visibility conditions. The adversarial object is placed to be visible in different numbers of cameras.

Visibility	RDT (Easy)		RDT (Hard)		Pi0 (Easy)		Pi0 (Hard)	
	ΔSR↓	TD	ΔSR↓	TD	ΔSR↓	TD	ΔSR↓	TD
3 Cameras (All)	**32.3**%	0.19	**31.2**%	0.25	**39.9**%	0.24	**33.8**%	0.27
2 Cameras	26.1%	0.15	24.7%	0.20	31.2%	0.19	26.9%	0.22
1 Camera	15.8%	0.09	14.3%	0.12	18.6%	0.11	15.2%	0.13
Top Camera Only	12.4%	0.08	11.8%	0.11	14.9%	0.10	12.7%	0.12
Wrist Cameras Only	17.9%	0.10	16.2%	0.13	21.3%	0.12	17.4%	0.14

C. Ablation Study

To systematically assess the contributions of each component in our multi-level attack framework, we conducted comprehensive ablation experiments by selectively removing individual loss terms. The results in Table II reveal a hierarchical importance structure among the components. The task failure loss $\mathcal{L}_{task}$ emerges as the most critical element, with its removal causing the success rate drop to plummet from 32.3% to 18.6%, underscoring its fundamental role in directly optimizing the primary attack objective. The trajectory deviation loss $\mathcal{L}_{traj}$ demonstrates substantial impact as well, contributing approximately 9.4 percentage points to the overall attack effectiveness by inducing systematic positional errors throughout the manipulation sequence. The action perturbation loss $\mathcal{L}_{shake}$ proves essential for disrupting the smooth action primitives learned through imitation, with its absence reducing attack potency by 6.6%. While the regularization term $\mathcal{L}_{reg}$ exhibits minimal quantitative impact on success rates, it serves a crucial qualitative role in maintaining physical realizability and visual inconspicuousness, preventing the optimization from converging to easily detectable high-frequency patterns. Most notably, when restricted to single loss components, the attack effectiveness catastrophically degrades (ΔSR merely 8.2%-18.7%), demonstrating that the synergistic interaction between multiple attack vectors is fundamental to compromising VLA models' robust decision-making pipeline.

D. Patch Parameter Analysis

To understand the sensitivity of our attack to key hyperparameters, we systematically investigated the influence of patch size and regularization strength on both attack effectiveness and trajectory perturbation, as illustrated in Figure 4. The relationship between patch size and attack potency exhibits a distinctive nonmonotonic pattern: moderate patches (3cm radius) achieve optimal performance by maintaining sufficient visual saliency while remaining within the robots' operational field of view throughout task execution. Smaller patches (1-2cm) suffer from reduced visual impact due to limited pixel coverage in the captured images, while larger patches (5cm) paradoxically weaken the attack by frequently exceeding camera boundaries during dynamic manipulation, resulting in intermittent visibility. The regularization parameter γ reveals an intriguing trade-off between stealth and disruption—increased regularization strength enhances visual naturalness through smoother texture transitions, yet counterintuitively amplifies trajectory deviation from 0.08 to 0.19, suggesting that constrained perturbations force the optimization to exploit more subtle but kinematically disruptive features. This phenomenon indicates that VLA models may be particularly vulnerable to smooth, naturalistic perturbations that align with expected environmental textures while fundamentally corrupting the learned visuomotor mappings.

E. Multi-view Robustness Evaluation

In real-world robotic operations, the visibility of objects in cameras changes with the arm's movement. Table III evaluates the effectiveness of the attack under different view-

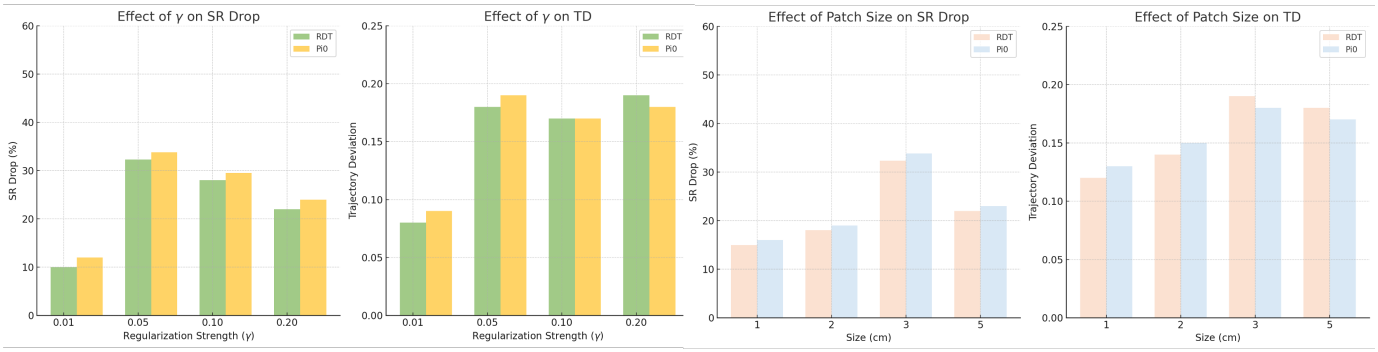

Fig. 4: Effect of Patch Size and Regularization Strength on SR Drop and Trajectory Deviation. Moderate patch sizes (3cm) maximize attack impact while increased regularization ($\gamma > 0.1$) significantly diminishes success rate reduction but paradoxically increases trajectory perturbation.

TABLE IV: Real-world validation of the adversarial attack on physical Aloha robot platform. Transfer Rate indicates the percentage of attack effectiveness retained from simulation.

Task	RDT				Pi0			
	Clean SR	Adv. SR	ΔSR↓	Transfer Rate	Clean SR	Adv. SR	ΔSR↓	Transfer Rate
Grab Roller	55%	23%	32%	82.1%	76%	36%	40%	**83.3%**
Adjust Bottle	65%	28%	37%	82.2%	72%	35%	37%	**84.1%**
Pick Dual Bottles	28%	10%	18%	81.8%	45%	20%	25%	**80.6%**
Place Burger Fries	38%	15%	23%	79.3%	62%	28%	34%	**81.0%**
Press Stapler	30%	12%	18%	81.8%	46%	23%	23%	**82.1%**
Average	43.2%	17.6%	25.6%	81.4%	60.2%	28.4%	31.8%	**82.2%**

point conditions. Even when the adversarial object is visible in only one camera, the success rate decreases by 14.3%-18.6%, which has significant implications for real-world attack scenarios. When visible in two or three cameras, the attack's effectiveness increases (24.7%-31.2% and 31.2%-39.9%, respectively). Notably, the attack is more effective from the wrist cameras (16.2%-21.3%) than from the top camera (11.8%-14.9%), likely because wrist cameras provide more critical visual information for task execution. These results show that the adversarial object does not require precise control over its appearance across multiple viewpoints to be effective, validating its utility in dynamic real-world environments.

F. Real-world Validation

To validate the attack in a real-world physical environment, we selected five representative tasks and deployed the adversarial patches optimized in simulation onto the Aloha dual-arm robot platform. The UAO was 3D printed into a 3cm radius sphere, with the optimized texture applied. Table IV shows the sim-to-real transferability of the attack. Despite challenges such as printing inaccuracies, dynamic lighting, and camera calibration errors, the attack still maintained significant effects. The success rates for RDT and Pi0 models dropped by 81.4% and 82.2%, respectively, compared to simulation, demonstrating the robustness of our adversarial object to real-world noise. In particular, tasks of varying complexity exhibited similar transfer rates (79%-84%), confirming the robustness of our attack method.

V. Conclusion

In this work, we provide the first systematic analysis of the vulnerabilities in recent VLA models under adversarial attacks. By introducing a universal adversarial object attack method, we highlight that even state-of-the-art VLA models are susceptible to significant security risks. Our proposed multi-level attack framework demonstrates strong generalization across diverse tasks and models, achieving successful transfer from simulated environments to real-world scenarios.

In the future, we will focus on both offensive and defensive strategies. On the offensive side, further exploration of the various attack modalities can help uncover additional vulnerabilities in VLA systems. On the defensive side, developing robust defense mechanisms, including adversarial training and real-time detection systems, will be crucial to safeguarding the security and reliability of VLA technologies.

ACKNOWLEDGMENT

This work was supported by the Key Research Project of Wuhan City 2024060788020073. The Learning Algorithms & Soft Manipulation Laboratory of Wuhan University supported the robot in this paper.

References

[1] Y. Ma, Z. Song, Y. Zhuang, J. Hao, and I. King, "A survey on vision-language-action models for embodied ai," *arXiv preprint arXiv:2405.14093*, 2024.

[2] R. Sapkota, Y. Cao, K. I. Roumeliotis, and M. Karkee, "Vision-language-action models: Concepts, progress, applications and challenges," *arXiv preprint arXiv:2505.04769*, 2025.

[3] Y. Hu, F. Lin, T. Zhang, L. Yi, and Y. Gao, "Look before you leap: Unveiling the power of gpt-4v in robotic vision-language planning," in *First Workshop on Vision-Language Models for Navigation and Manipulation at ICRA 2024*.

[4] X. Li, M. Zhang, Y. Geng, H. Geng, Y. Long, Y. Shen, R. Zhang, J. Liu, and H. Dong, "Manipllm: Embodied multimodal large language model for object-centric robotic manipulation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 18 061–18 070.

[5] S. Huang, I. Ponomarenko, Z. Jiang, X. Li, X. Hu, P. Gao, H. Li, and H. Dong, "Manipvqa: Injecting robotic affordance and physically grounded information into multi-modal large language models," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024, pp. 7580–7587.

[6] Y. Jin, D. Li, J. Shi, P. Hao, F. Sun, J. Zhang, B. Fang *et al.*, "Robotgpt: Robot manipulation learning from chatgpt," *IEEE Robotics and Automation Letters*, vol. 9, no. 3, pp. 2543–2550, 2024.

[7] Y. Jiang, A. Gupta, Z. Zhang, G. Wang, Y. Dou, Y. Chen, L. Fei-Fei, A. Anandkumar, Y. Zhu, and L. Fan, "Vima: General robot manipulation with multimodal prompts," *arXiv preprint arXiv:2210.03094*, vol. 2, no. 3, p. 6, 2022.

[8] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu *et al.*, "Rt-1: Robotics transformer for real-world control at scale," *arXiv preprint arXiv:2212.06817*, 2022.

[9] B. Zitkovich, T. Yu, S. Xu, P. Xu, T. Xiao, F. Xia, J. Wu, P. Wohlhart, S. Welker, A. Wahid *et al.*, "Rt-2: Vision-language-action models transfer web knowledge to robotic control," in *Conference on Robot Learning*. PMLR, 2023, pp. 2165–2183.

[10] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi *et al.*, "Openvla: An open-source vision-language-action model," *arXiv preprint arXiv:2406.09246*, 2024.

[11] J. Wen, Y. Zhu, J. Li, Z. Tang, C. Shen, and F. Feng, "Dexvla: Vision-language model with plug-in diffusion expert for general robot control," *arXiv preprint arXiv:2502.05855*, 2025.

[12] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter *et al.*, "π: A vision-language-action flow model for general robot control," *CoRR*, 2024.

[13] S. Liu, L. Wu, B. Li, H. Tan, H. Chen, Z. Wang, K. Xu, H. Su, and J. Zhu, "Rdt-1b: a diffusion foundation model for bimanual manipulation," in *The Thirteenth International Conference on Learning Representations*.

[14] D. Liu, M. Yang, X. Qu, P. Zhou, Y. Cheng, and W. Hu, "A survey of attacks on large vision–language models: Resources, advances, and future trends," *IEEE Transactions on Neural Networks and Learning Systems*, 2025.

[15] Z. Fu, T. Z. Zhao, and C. Finn, "Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation," *arXiv preprint arXiv:2401.02117*, 2024.

[16] M. U. Din, W. Akram, L. S. Saoud, J. Rosell, and I. Hussain, "Vision language action models in robotic manipulation: A systematic review," *arXiv preprint arXiv:2507.10672*, 2025.

[17] R. Takano, H. Oyama, and M. Yamakita, "Continuous optimization-based task and motion planning with signal temporal logic specifications for sequential manipulation," in *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2021, pp. 8409–8415.

[18] J. Y. M. Y. E. Goldman, "Evaluating π0 on long-horizon manipulation tasks."

[19] G. Koch, R. Zemel, R. Salakhutdinov *et al.*, "Siamese neural networks for one-shot image recognition," in *ICML deep learning workshop*, vol. 2, no. 1. Lille, 2015, pp. 1–30.

[20] J. C. Spall, "Multivariate stochastic approximation using a simultaneous perturbation gradient approximation," *IEEE transactions on automatic control*, vol. 37, no. 3, pp. 332–341, 2002.

[21] Y. Mu, T. Chen, S. Peng, Z. Chen, Z. Gao, Y. Zou, L. Lin, Z. Xie, and P. Luo, "Robotwin: Dual-arm robot benchmark with generative digital twins (early version)," in *European Conference on Computer Vision*. Springer, 2024, pp. 264–273.

[22] C. Zhang, P. Benz, C. Lin, A. Karjauv, J. Wu, and I. S. Kweon, "A survey on universal adversarial attack," in *International Joint Conference on Artificial Intelligence 2021*. Association for the Advancement of Artificial Intelligence (AAAI), 2021, pp. 4687–4694.

[23] Y. Liu, H. Wei, C. Jia, R. Xiao, W. Ruan, X. Wei, J. T. Zhou, and Z. Wang, "Projattacker: A configurable physical adversarial attack for face recognition via projector," in *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025, pp. 21 248–21 257.

[24] T. B. Brown, D. Mané, A. Roy, M. Abadi, and J. Gilmer, "Adversarial patch," 2018. [Online]. Available: https://arxiv.org/abs/1712.09665

[25] T. Wang, C. Han, J. C. Liang, W. Yang, D. Liu, L. X. Zhang, Q. Wang, J. Luo, and R. Tang, "Exploring the adversarial vulnerabilities of vision-language-action models in robotics," 2025. [Online]. Available: https://arxiv.org/abs/2411.13587

[26] Y. Liu, H. Wei, C. Jia, R. Xiao, W. Ruan, X. Wei, J. T. Zhou, and Z. Wang, "Projattacker: A configurable physical adversarial attack for face recognition via projector," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 21 248–21 257.

[27] I. Evtimov, K. Eykholt, E. Fernandes, T. Kohno, B. Li, A. Prakash, A. Rahmati, and D. Song, "Robust physical-world attacks on machine learning models," *arXiv preprint arXiv:1707.08945*, vol. 2, no. 3, p. 4, 2017.

[28] T. Tsai, K. Yang, T.-Y. Ho, and Y. Jin, "Robust adversarial objects against deep learning models," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 01, 2020, pp. 954–962.

[29] Y. Deng, X. Zheng, T. Zhang, C. Chen, G. Lou, and M. Kim, "An analysis of adversarial attacks and defenses on autonomous driving models," in *2020 IEEE international conference on pervasive computing and communications (PerCom)*. IEEE, 2020, pp. 1–10.

[30] H. Wei, Z. Wang, K. Zhang, J. Hou, Y. Liu, H. Tang, and Z. Wang, "Revisiting adversarial patches for designing camera-agnostic attacks against person detection," *Advances in Neural Information Processing Systems*, vol. 37, pp. 8047–8064, 2024.

[31] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, and D. Song, "Robust physical-world attacks on deep learning visual classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1625–1634.

[32] Y. Jia, C. M. Poskitt, J. Sun, and S. Chattopadhyay, "Physical adversarial attack on a robotic arm," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9334–9341, 2022.

[33] F. Bai, R. Liu, Y. Du, Y. Wen, and Y. Yang, "Rat: Adversarial attacks on deep reinforcement agents for targeted behaviors," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 15, 2025, pp. 15 453–15 461.

[34] G. Team, T. Mesnard, M. Hardin, M. S. Kale, J. Love *et al.*, "Gemma: Open models based on gemini research and technology," *arXiv preprint arXiv:2403.08295*, 2024.

[35] W. Peebles and S. Xie, "Scalable diffusion models with transformers," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 4195–4205.